

**Федеральное государственное бюджетное образовательное**  
учреждение высшего образования  
«Ижевский государственный технический университет  
имени М.Т. Калашникова»

На правах рукописи

Николаева Юлия Викторовна

**МЕТОДЫ И АЛГОРИТМЫ ИНТЕЛЛЕКТУАЛЬНОЙ СИСТЕМЫ  
ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ ТРЕЙДЕРОВ ФИНАНСОВЫХ  
РЫНКОВ**

05.13.10 – Управление в социальных и экономических системах

Диссертация на соискание ученой степени  
кандидата технических наук

Научный руководитель:  
Лялин Вадим Евгеньевич,  
д.т.н., д.э.н., д.г.-м.н., профессор

Ижевск, 2018

## ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	5
1 ОБЗОР МЕТОДОВ АНАЛИЗА ФИНАНСОВЫХ РЫНКОВ И ИНТЕЛЛЕКТУАЛЬНОГО АНАЛИЗА ДАННЫХ.....	16
1.1 Постановка задачи управления на финансовых рынках на основе теории управления социально-экономическими системами.....	16
1.2 Классические методы анализа финансового рынка.....	21
1.2.1 Фундаментальный анализ.....	24
1.2.2 Технический анализ финансовых рынков.....	26
1.3 Эконометрический анализ финансовых рынков.....	29
1.4 СППР и методы интеллектуального анализа данных .....	34
1.4.1 Системы поддержки принятия решений.....	34
1.4.2 Исторический обзор развития интеллектуальных методов анализа данных .....	36
1.4.3 Современное состояние методов интеллектуального анализа данных и средств искусственного интеллекта.....	42
1.4.3.1 Деревья решений.....	46
1.4.3.2 Метод ближайшего соседа.....	48
1.4.3.3 Метод опорных векторов.....	50
1.4.3.4 Нейронные сети.....	52
Выводы по главе, цель и задачи исследования.....	61
2 РАЗРАБОТКА МЕТОДОВ И АЛГОРИТМОВ ИНТЕЛЛЕКТУАЛЬНОЙ СИСТЕМЫ ПРИНЯТИЯ РЕШЕНИЙ ТРЕЙДЕРОМ.....	66
2.1 Алгоритмы методов технического анализа.....	67
2.1.1 Индекс относительной силы.....	68
2.1.2 Конвергенция-дивергенция скользящих средних.....	69
2.2 Алгоритмы метода эконометрического моделирования.....	72
2.2.1 Модель обобщенной условной авторегрессионной гетероскедстичности.....	72
2.2.2 Метод максимального правдоподобия.....	76
2.3 Алгоритм анализа с применением метода искусственного интеллекта.....	82
2.3.1 Нормировка данных для нейронной сети.....	84
2.3.2 Алгоритмы обучения многослойного перцептрона.....	87
2.3.2.1 Алгоритм обратного распространения ошибки.....	88
2.3.2.2 Альтернативные методы обучения многослойного перцептрона.....	91

2.3.3	Распознавание кризисных паттернов с использованием нейронных сетей.....	96
2.4	Алгоритмы с использованием элементов фундаментального анализа.....	102
2.4.1	Исследование влияния экономического календаря на волатильность котировок.....	102
2.4.2	Алгоритм анализа повышений волатильности при совершении экономических событий.....	113
2.5	Методы и алгоритмы нейросетевой классификации рыночных ситуаций.....	116
2.5.1	Проектирование архитектуры нейронной сети.....	119
2.5.2	Метод генерации выходной выборки для задачи нейросетевой классификации рыночных ситуаций.....	127
2.5.3	Методика поддержки принятия решений трейдера и ее алгоритм классификации рыночных ситуаций с использованием многослойного перцептрона и методов анализа финансовых рынков .....	130
	Выводы по главе.....	134
3	ПРОЕКТНЫЕ РЕШЕНИЯ ПО РАЗРАБОТКЕ ИНТЕЛЛЕКТУАЛЬНОЙ СИСТЕМЫ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ ФИНАНСОВЫХ РЫНКОВ.....	136
3.1	Проектирование архитектуры системы.....	136
3.2	Проектирование системы с использованием case-средств.....	137
3.2.1	Функциональное проектирование системы.....	138
3.2.2	Детальное проектирование системы.....	142
3.3	Проектирование базы данных.....	150
3.3.1	Концептуальная модель базы данных.....	152
3.3.2	Логическая модель базы данных.....	156
	Выводы по главе.....	161
4	РЕАЛИЗАЦИЯ ИСППР И РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТОВ.....	163
4.1	Сравнение с аналогами ИСППР «НейроПрофит» .....	163
4.2	Разработка ИСППР «НейроПрофит».....	166
4.2.1	Пользовательские интерфейсы.....	166
4.2.2	Примеры функционирования системы.....	173
4.3	Исследование работоспособности системы.....	176
	Выводы по главе.....	183
	ЗАКЛЮЧЕНИЕ.....	185
	СПИСОК СОКРАЩЕНИЙ.....	187
	СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	188
	ПРИЛОЖЕНИЕ А. СВИДЕТЕЛЬСТВА О РЕГИСТРАЦИИ ПРОГРАММЫ ДЛЯ ЭВМ.....	201

ПРИЛОЖЕНИЕ Б. АКТ ВНЕДРЕНИЯ РЕЗУЛЬТАТОВ ДИССЕРТАЦИОННОЙ РАБОТЫ....	
.....	203
ПРИЛОЖЕНИЕ В. ИСПОЛЬЗОВАНИЕ ВЕБ-ТЕХНОЛОГИЙ В СИСТЕМЕ .....	204
ПРИЛОЖЕНИЕ Г. ИСХОДНЫЙ КОД АЛГОРИТМОВ РАБОТЫ ИСППР.....	221
ПРИЛОЖЕНИЕ Д. ЭКСПЕРИМЕНТАЛЬНЫЕ ДАННЫЕ .....	228



## ВВЕДЕНИЕ

**Актуальность темы исследования.** В рыночной экономике важной причиной для проведения исследований различных социально-экономических систем является обеспечение стабильного роста экономики отдельных предприятий и стабилизация экономики страны в целом. Вопросы управления социально-экономическими системами и принятия решений в них носят ярко выраженную дисциплинарную направленность и являются наиболее актуальными темами исследований в современных науках об управлении. Поскольку финансовые рынки являются яркими примерами организационных систем, интерес к исследованию их функционирования и управления ими поддерживается на высоком уровне. Особенно, учитывая, что размещение временно свободных денежных средств на финансовых рынках приобрело в последние десятилетия большую популярность. При грамотном подходе оно приносит большую прибыль, нежели банковские депозиты или хранение денег в валюте.

В условиях текущей экономической реальности в России торговля на финансовых рынках становится все более непредсказуемой и, как следствие, рискованной. Торговля на финансовых рынках в современной действительности, как на фондовых и срочных рынках российских биржах, так и на международных товарных биржах, на валютном рынке, требует от трейдера значительно более глубоких знаний, чем 10-15 лет назад, умений пользоваться большим количеством инструментов анализа, быстроты реакции при реагировании на изменения и принятия обоснованных решений.

Принятие инвестиционных решений на финансовом рынке является вопросом исследования данной работы. В этом плане умение хорошо прогнозировать самым непосредственным образом превращается в прибыль. Если инвестор предполагает, что цена акции или валюты вырастет, он покупает акции, валюту, надеясь продать их позже по более высокой цене. И, наоборот, прогнозируя падение цен, инвестор продаёт акции, чтобы впоследствии выкупить их обратно по более низкой цене. Задача инвестора или профессионального трейдера в том, чтобы правильно предугадать направление будущего изменения цены – роста или падения.

Зачастую, самой распространенной практикой принятия инвестиционных решений широкой аудиторией является практика интуитивной торговли. Это тем более парадоксально, что основная масса игроков рынка достаточно неплохо знакома с основами анализа и в теории могла бы неплохо зарабатывать на биржевой игре. Однако статистика показывает, что подавляющее большинство трейдеров проигрывает.

Безусловно, принятие инвестиционного решения на финансовом рынке на основе фундаментального и технического анализа является рациональным и более прибыльным, чем на ос-

нове интуитивных предположений. Но так как финансовый рынок является быстроразвивающейся динамической системой, закономерности и тенденции, выявленные с помощью подходящих в данный момент методов анализа и прогнозирования, через некоторый промежуток времени становятся не эффективными для применения в реальной торговой стратегии.

Нарастающая скорость развития биржевой торговли безусловно является стимулом, подталкивающим к развитию все более совершенных многообразных методов анализа данных, порождаемых финансовыми рынками, как сложными организационными системами. Для анализа многокритериальной информации, из которой складывается текущее состояние рыночной системы, перед аналитиком стоит задача выбора наиболее удобных, распространенных, в то же время достоверных инструментов для поддержки принятия им решения о направлении заключения сделки с тем или иным финансовым инструментом.

На смену интуитивной торговле, основанной на собственных предположениях, приходят мощные инструменты системного анализа состояний рыночных систем, основанных на самых современных разработках в сфере искусственного интеллекта. Все большее число профессиональных трейдеров встают на сторону технического анализа, предполагающего независимость временного ряда котировок конкретного финансового инструмента от остальных. Ориентированность на самоописание временного ряда, то есть предположение, что все зависимости и настроения рынка уже сами собой включены в динамику временного ряда, ведет к уменьшению сторонников использования фундаментального анализа рынков, высчитывающие величину капитализации, оборотов, чистой прибыли и других экономических показателей компаний-эмитентов для определения истинной стоимости их акций. Временной ряд каждого финансового инструмента формируется как динамическое коллективное поведение рыночного сообщества, основываясь на реакциях участников рынка на происходящие события. Еще с развития волновой теории Эллиотта укрепляется мнение, что в самих временных рядах финансовых инструментов содержатся скрытые зависимости и закономерности, выявление которых гарантирует высокую степень предсказуемости поведения рынка. С появлением нейросетевого аппарата у сторонников данных идей появились реальные возможности утвердиться в своем мнении и на практике применять нейросетевые модели для определения внутренних не очевидных закономерностей развития динамической системы – рынка конкретного финансового инструмента.

Нарастающая скорость развития и автоматизации отрасли биржевой торговли, потребность в современном и эффективном инструментарии для анализа ситуаций на финансовых рынках, ведет к бурному развитию методов интеллектуального анализа данных, развитию информационных систем управления и исследований в области систем поддержки принятия решений (СППР) для торговли на финансовых рынках. Распространенными классами СППР для

анализа данных финансовых рынков и поддержки принятия решений на финансовых рынках являются торговые платформы, например, MetaTrader (Кипр), MetaStock (США), QUIK (Россия, Новосибирск), и торговые советники, такие как TSLab (Россия, Москва), Wall Street Forex Robot Evolution (США). Большинство торговых платформ обладают возможностями технического анализа, просмотра экономических событий, ручного выбора параметров и заключения сделки. Использование торговых советников зачастую сводится к выработке определенной стратегии на конкретном финансовом инструменте и временном интервале, не применимые на других параметрах. Но у существующих СППР отсутствует возможность задания ожидаемой прибыли и учета повышения волатильности под влиянием экономических событий. Разработка и использование методов анализа и прогнозирования поведения финансового инструмента, учитывающих данные возможности, являются актуальной сферой приложения математических, эконометрических знаний, а также навыков проектирования и разработки информационных систем.

**Степень разработанности темы исследования.** Исторически анализ финансовых рынков сводился к поиску математических зависимостей во временных рядах котировок финансовых инструментов, начали появляться технические индикаторы, тем или иным образом порождаемые исходной системой. Далее бурно развивалась область исследований по анализу временных рядов, и стало возможным прогнозировать данные временные ряды, сначала с применением классических методов статистического анализа, затем методов авторегрессионного анализа, адаптивных методов, основанных на скользящих средних.

Исследованием методов прогнозирования временных рядов посвящено большое количество научных работ. Начало исследований в данной области положили работы таких авторов как G.E.P. Box, G.M. Jenkins, D.R. Brillinger, C. Holt, P. Winters [11, 113, 114, 121, 134]. Русскоязычные авторы, занимающимися данной тематикой: Лукашин Ю.П., Афанасьев В.Н., Канторович Г.Г. [52, 8, 40, 41, 42, 43, 44]. Среди прочих встречаются работы и по прогнозированию именно финансовых временных рядов, например, Осминин К.П., Орлов Ю.Н. [81, 82, 83].

С развитием методов распознавания образов и нейросетевой методологии стал возможен принципиально новый подход к анализу финансовых рынков, отличающийся от прогнозирования временных рядов, такой как классификация рыночных ситуаций.

Методами распознавания образов и классификации с использованием нейронных сетей занимались F. Rosenblatt, D.E. Rumelhart, G.E. Hinton, R.J. Williams, T. Kohonen [129, 130, 131, 120, 122]. В России данные методы и подходы развивали такие ученые, как А. И. Галушкин, Горбань А.Н., В.Л.Дунин-Барковский, Е.М.Миркес [115, 21, 24, 25, 32, 61]. С развитием теоретических основ начали появляться работы, описывающие применение нейросетевого аппарата в различных прикладных областях, в том числе в анализе финансовых рынков, например, работы

Бэстенс Д.-Э., ван ден Берг В.-М., Ежов А. А., Шумский С.А., Ширяев В. И., Бугорский В.Н., Сергиенко А.Г. [17, 33, 106, 14].

С усложнением математического инструментов анализа рынков, трейдеры оказываются в ситуации, когда им необходимо отслеживать все большее количество различных индикаторов и показателей, выявлять для себя значимость данных критериев в общей модели анализа, и, соответственно, им все сложнее становится принимать окончательное решение. Данные факторы стали причиной широкого распространения в последнее десятилетие торговых роботов, которые по заранее запрограммированным алгоритмам анализируют данные и сами заключают сделки на финансовых рынках, полностью исключая трейдера из процесса принятия решений. Но такие системы не принесли ожидаемого от них эффекта, так как они не учитывают, что рынок по своей сути является психологически обоснованной системой, некоторые сигналы которой способен распознать только человек, являющийся источником данных психологических колебаний. Поэтому именно трейдер выступает в системе «анализ-торговля на финансовом рынке» в качестве лица принимающего решения (ЛПР), а задачей аналитических информационных систем является помощь в принятии решений в сложных условиях при слабоструктурированной информации о предметной области, каковыми являются финансовые рынки. Существующие системы поддержки принятия решений, такие как торговые советники и многофункциональные торговые платформы, в большей степени обеспечивают трейдера рекомендациями по вхождению в сделку, не выдавая информацию о том, как долго находится в сделке и при достижении какой прибыли выходить из сделки, а также не учитывают риск повышения волатильности под влиянием изменений макроэкономических показателей, экономических новостей и событий. В данной ситуации трейдеру сложно выстроить прибыльную торговую стратегию, поскольку велик риск заключения убыточных сделок, и даже при заключении потенциально прибыльной сделки по торговому сигналу, рекомендации, несвоевременное ее закрытие может приводить к потере ожидаемой прибыли.

Таким образом, объективно сложилось *противоречие*, заключающееся в том, что существующие методы и системы поддержки принятия решений, основанные на интеллектуальном анализе данных, не обеспечивают пользователя информацией о критериях закрытия сделки и рисках повышения волатильности под влиянием экономических событий и новостей, тогда как при практическом применении данных систем пользователю при открытии сделки по выданной СППР рекомендации необходимо понимать, при достижении какой прибыли необходимо закрывать сделку, а также анализировать риски изменений котировок, связанные с выходом экономических новостей и событий разной степени важности.

**Объектом исследования** являются финансовые рынки, как сложные системы, и порождаемые ими данные по изменениям цен финансовых инструментов, представляющие собой одномерные временные ряды.

**Предметом исследования** является экономико-математическое, алгоритмическое, программное обеспечение анализа финансовых рынков; методы классификации ситуаций на финансовых рынках; математическое описание алгоритмов методов анализа временных рядов.

**Цель и основные задачи работы.** Целью исследования является повышение эффективности принятия управленческих решений участников рынка, заключающееся в достижении ими положительных результатов при торговле, за счет применения интеллектуальной системы поддержки принятия решений трейдеров финансовых рынков.

Для достижения главной цели были поставлены следующие **задачи**:

1. Исследование и анализ существующих методов и средств анализа финансовых рынков и временных рядов, и выбор наиболее удовлетворяющих параметрам эффективности и программной реализуемости.
2. Исследование, анализ и выбор методов и средств интеллектуальных систем поддержки принятия решений, подходящих для задач классификации.
3. Разработка метода генерации выходной выборки для обучения нейронной сети с учетом ожидаемой прибыли и повышения волатильности.
4. Разработка методики поддержки принятия решений трейдера по направлению сделки с финансовым инструментом, основанной на нейросетевой методологии с применением методов технического, фундаментального анализа и эконометрического моделирования.
5. Проектирование и разработка интеллектуальной системы поддержки принятия решений на языке высокого уровня, исследование ее работоспособности на реальных примерах.

**Методы исследования.** В работе использованы методы системного анализа, математического моделирования, методы теории вероятности, математической статистики, фундаментального и технического анализа, нейросетевого моделирования, теории принятия решений, методы разработки настольных и веб-приложений.

При выполнении поставленных задач возникли проблемы, связанные с выбором методологического обеспечения работы:

- выбор современных моделей прогнозирования временных рядов, являющихся компромиссными между эффективностью и технической простотой, учитывающие специфику временного ряда цены финансового инструмента;
- выбор адекватного метода оценивания прогнозных моделей, применению которого не противоречит структура и особенности данного временного ряда;

- выбор лучших индикаторов технического анализа для финансового рынка на основе популярности их применения;
- выбор финансовых инструментов, являющихся популярными на своем рынке и позволяющих на своем примере оценить поведение других инструментов данного рынка;
- выбор топологии нейронной сети, подходящей в большей мере поставленным задачам;
- выбор метода обучения нейронной сети, оптимизирующего процесс обучения;
- выбор подходящих средств проектирования ИС, языка и среды программирования, системы управления базами данных на основе удобства применения.

**Научная новизна** полученных результатов определяется комплексными исследованиями, направленными на разработку методов, алгоритмов и программных средств анализа и классификации ситуаций на финансовых рынках, в ходе которых:

1. Разработан метод генерации выходной выборки для обучения нейронной сети с учетом повышения волатильности, использующийся для формирования выходной выборки при обучении нейронной сети в задаче классификации рыночных ситуаций, отличающийся возможностью задания ожидаемой прибыли, позволяющий исключить влияние на эффективность работы ИСППР квалификации эксперта и учесть риск повышения волатильности, связанный с влиянием макроэкономических показателей и новостей.

2. Разработаны методика поддержки принятия решений трейдера по направлению сделки с финансовым инструментом, заключающаяся в нейросетевой классификации рыночных ситуаций с применением методов технического, фундаментального анализа и эконометрического моделирования, и алгоритм ее реализации в системе поддержки принятия решений на финансовых рынках, основанные на применении нейронной сети, на входы которой подаются данные, полученные в результате: построения эконометрической модели, использования технических индикаторов, анализа реакции на события экономического календаря, отличие которого заключается в обучении нейросети на основе оценок с учетом ожидаемой прибыли от сделки и повышения волатильности. Методика позволяет проводить анализ многокритериальной информации о текущем состоянии рынка конкретного финансового инструмента, учитывая основные подходы к анализу финансовых рынков: прогнозирования временных рядов, технического анализа, фундаментального анализа (в части анализа влияния на повышение волатильности изменений макроэкономических показателей, выхода экономических новостей).

3. Реализована интеллектуальная система поддержки принятия решений «Нейро-Профит», функциональное и детальное проектирование системы, программное, информационное и математическое обеспечение системы и результаты ее экспериментального использования, отличающаяся наличием:

- информационных связей, позволяющих ЛПП задавать параметры анализа в виде ожидаемой прибыли от сделки с конкретным финансовым инструментом на выбранном временном интервале;

- модуля фундаментального анализа, позволяющего выявлять для выбранного финансового инструмента значимые новости и события, из экономического календаря, полученного с помощью веб-сервиса, а также анализировать коэффициент повышения волатильности на данные события;

- кэша данных, позволяющего обращаться к результатам технического, эконометрического, фундаментального анализа, при повторном использовании системы поддержки принятия решений для выбранного финансового инструмента и временного интервала с изменением параметра ожидаемой прибыли для получения альтернативной рекомендации, сокращая временные затраты при повторной подготовке данных системы для нейросетевого анализа;

- базой данных, хранящей результаты обучения нейронной сети для различных финансовых инструментов и временных интервалов, оцененных пользователем, которая позволяет подбирать ранее сохраненные веса нейронной сети, подходящей для выбранных параметров, снижая время обучения нейронной сети при многократном использовании системы для повторяющихся параметров.

**На защиту выносятся** результаты исследования и разработки методики классификации рыночных ситуаций и основанной на ней ИСППР, обеспечивающей повышение прибыльность торговой деятельности трейдера и получению им рекомендаций по покупке-продаже финансовых инструментов, в том числе:

1. Разработанный метод генерации выходной выборки для обучения нейронной сети для задачи классификации рыночных ситуаций с учетом повышения волатильности.

2. Разработанные методика поддержки принятия решений трейдера по направлению сделки с финансовым инструментом, заключающаяся в нейросетевой классификации рыночных ситуаций, и алгоритм анализа финансовых рынков, основанные на нейросетевом подходе с применением методов технического, фундаментального и эконометрического анализа, с учетом ожидаемой прибыли от сделки и повышения волатильности.

3. Реализованная интеллектуальная система поддержки принятия решений «НейроПрофит», функциональное и детальное проектирование системы, принципы построения программного, информационного и математического обеспечения системы.

4. Результаты экспериментального использования ИСППР «НейроПрофит», оценка эффективности классификации рыночных ситуаций системой, работающей на основе разработанной методики.

**Практическая значимость работы** заключается в применении на практике интеллектуальной СППР, функционирующей на основе разработанных алгоритмов и методов, а именно:

1. Использование разработанной методики и алгоритма ее реализации при управлении стратегией заключения сделок позволяет трейдерам увеличить достоверность выдаваемых рекомендаций на 8-26% по сравнению с аналогичными алгоритмами нейросетевого анализа рыночных ситуаций.

2. Разработанный метод генерации выходной выборки с учетом ожидаемой прибыли обеспечивает трейдера не только информацией по направлению открытия сделки, но и явным количественным критерием для закрытия сделки с прибылью, увеличивая количество прибыльных торговых операций с 75 до 83 %.

3. Разработанный метод генерации выходной выборки позволяет минимизировать влияние квалификации эксперта в процессе обучения нейросети (практически свести к 0 вероятность ухудшения результатов обучения нейросети из-за некомпетентности учителя).

4. Разработанная архитектура ИСППР позволяет сохранять наилучшие случаи обучения нейронной сети на конкретных параметрах, ускоряя в 2 раза повторное обучение нейросети на аналогичных параметрах при последующих использованиях ИСППР.

**Соответствие паспорту специальности.** Согласно паспорту специальности 05.13.10 Управление в социальных и экономических системах диссертационная работа соответствует п. 4 «Разработка методов и алгоритмов решения задач управления и принятия решений в социальных и экономических системах» в части разработки метода генерации выходной выборки для обучения нейронной сети с учетом ожидаемой прибыли и повышения волатильности, п.5 «Разработка специального математического и программного обеспечения систем управления и механизмов принятия решений в социальных и экономических системах» и п. 10 «Разработка методов и алгоритмов интеллектуальной поддержки принятия управленческих решений в экономических и социальных системах» в части разработки методики поддержки принятия решений трейдера по направлению сделки с финансовым инструментом, заключающейся в нейросетевой классификации рыночных ситуаций, и интеллектуальной системы поддержки принятия решений трейдеров финансовых рынков.

**Апробация работы.** Основные результаты диссертации и материалы исследования докладывались на следующих конференциях:

- Научно-техническая конференция аспирантов, магистрантов и молодых ученых «Молодые ученые – ускорению научно-технического прогресса в XXI веке» (ИжГТУ, 15-18 марта 2011);



- Межвузовская студенческая научная конференция «Communication of Students, Master Students and Postgraduates in Academic, Scientific and Professional Areas» (ИжГТУ, 27 апреля - 3 мая 2011 г.);
- «Молодые ученые – ускорению научно-технического прогресса в XXI веке, III Всероссийской научно-технической конференции аспирантов, магистрантов и молодых ученых с международным участием» (Ижевск, 22-23 апреля 2015 г);
- «Молодые ученые – ускорению научно-технического прогресса в XXI веке, IV Всероссийской научно-технической конференции аспирантов, магистрантов и молодых ученых с международным участием» (Ижевск, 20-22 апреля 2016 г);
- Международная научно–практическая конференция «Новая наука: от идеи к результату» (Сургут, 22.10.2016 г);
- Международная научно-практическая конференция «Новая наука: стратегии и векторы развития» (Магнитогорск, 08.05.2017 г);
- «Научные революции: сущность и роль в развитии науки и техники» (Пермь, 08.05.2017 г.).

**Публикации.** Некоторые аспекты и результаты научного исследования опубликованы в 15 научных работах, в том числе 8 статей в журналах, из которых 7 в рекомендованных ВАК РФ периодических изданиях, и 7 тезисов докладов.

**Достоверность и обоснованность результатов** диссертации подтверждается: экспериментами по применению разработанного алгоритма и интеллектуальной системы поддержки принятия решений к данным котировок курсов валют, акций, фьючерсов; корректным использованием законов и положений теории нейронных сетей; подтверждением результатов проведенных исследований на основе положений математической статистики; рецензированием печатных работ, их обсуждением на научно-технических конференциях, а также экспертизой для авторского свидетельства разработанной ИСППР.

Список используемых источников и литературы включает в себя как монографии зарубежных ученых, учебные и методические пособия русских авторов, включая работы автора представленного диссертационного исследования, так и различные интернет-ресурсы, другие средства массовой информации.

**Реализация результатов работы.** Результаты диссертационной работы по разработке интеллектуальной системы поддержки принятия решений трейдера финансовых рынков представлены на выставке инноваций 2016 (весенняя сессия) бизнес-инкубатора ИжГТУ им. М.Т. Калашникова. Разработанная интеллектуальная система поддержки принятия решений «НейроПрофит» и база данных зарегистрированы в Реестре программ для ЭВМ: свидетельства

№ 2016661369, 2016621361. Свидетельства о государственной регистрации программы для ЭВМ приведены в Приложении А. Система используется автором для индивидуальной инвестиционной деятельности на финансовых рынках. Отдельные результаты диссертационного исследования используются при проведении лекционных, лабораторных и практических занятий студентов по дисциплинам «Методы оптимизации», «Базы данных», «Интеллектуальные информационные системы» на кафедре ЗИКС ФГБОУ ВО «Ижевский государственный технический университет имени М. Т. Калашникова». Акт внедрения приведен в Приложении Б.

**Личный вклад соискателя.** Все исследования, результаты которых изложены в диссертации, получены лично соискателем в процессе научных исследований и экспериментов. Из совместных публикаций в диссертацию включен только тот материал, который непосредственно принадлежит соискателю.

**Структура и объем диссертационной работы.** Объем диссертационной работы составляет 230 страницы, включая 60 рисунков и 30 таблиц. Работа состоит из введения, четырех глав, заключения, списка сокращений, списка использованных источников, включающего 136 наименования, в дополнение к работе приводятся 5 приложений.

**Введение** содержит обоснование актуальности темы, формулировку цели и задач исследования, основные положения, выносимые на защиту, определяет научную новизну, практическую значимость, содержание и методы выполнения работы.

**В первой главе** приведена постановка задачи управления на финансовых рынках, как организационных системах. Рассмотрены распространенные методы анализа финансовых рынков, выявлены преимущества и недостатки. Приведен обзор подходящих методов и средств интеллектуальных систем поддержки принятия решений для задачи классификации информации о состоянии системы рынка выбранного финансового инструмента. Более подробно рассмотрены теоретические основы нейросетевой методологии.

**Вторая глава** отведена описанию математических методов анализа и классификации состояний на рынке финансового инструмента. Приводятся алгоритмы вычисления значений выбранных технических индикаторов, обоснование выбора прогнозной модели условной обобщенной авторегрессионной гетероскедестичности, численный метод нахождения ее коэффициентов. Рассматриваются алгоритмы применения многослойного перцептрона в качестве классификатора рыночных ситуаций. Приводится алгоритм реализации в СППР разработанной комбинированной методики нейросетевой классификации ситуаций на финансовых рынках. Описаны результаты исследования влияния экономического календаря на волатильность котировок. Описаны метод генерации выходной выборки для обучения нейронной сети с учетом

ожидаемой прибыли и повышение волатильности, алгоритм исключения кризисных паттернов из входной выборки для обучения нейронной сети.

**В третьей главе** приведены проектные решения по разработке интеллектуальной системы поддержки принятия решений на основе описанной методики нейросетевой классификации рыночных ситуаций. Приведены этапы функционального и детального проектирования, проектирования базы данных. Даны практические рекомендации по проектированию информационных систем типа СППР.

**В четвертой главе** приведены примеры функционирования и тестирования разработанной системы. Рассмотрено сравнение с аналогичными СППР на финансовых рынках. Описано решение проблем, возникших при проверке работоспособности системы.

**В заключении** приведены основные результаты проведенного теоретического и экспериментального исследования.

**В приложении** приведены свидетельства о регистрации программы для ЭВМ, акт внедрения результатов диссертационного исследования, описание использования веб-технологий в системе. Представлены экспериментальные данные и исходный код некоторых алгоритмов работы ИСППР.

## **1 ОБЗОР МЕТОДОВ АНАЛИЗА ФИНАНСОВЫХ РЫНКОВ И ИНТЕЛЛЕКТУАЛЬНОГО АНАЛИЗА ДАННЫХ**

Финансовый рынок является системой отношений, возникающей в процессе обмена экономическими благами с использованием денег в качестве актива-посредника [100].

На финансовом рынке происходит мобилизация капитала, осуществление обменных денежных операций, размещение финансовых средств в бизнесе и производстве, предоставление кредитов и займов на различные сроки. В совокупности спрос и предложение на капитал заемщиков и кредиторов из различных стран образуют мировой финансовый рынок.

Финансовый рынок подразделяется на:

- Рынок капиталов (акционерного и облигаций, векселей);
- Денежный рынок;
- Рынок производных инструментов (деривативов);
- Валютный рынок (форекс).

Анализ финансовых рынков – комплексная и сложная задача, представляющая собой анализ большого количества данных, среди которых выделяют котировки финансовых инструментов, объемы торгов, экономические новости и события, значения макроэкономических показателей. Большинство экспертов в данной области сходятся во мнении, что финансовый рынок представляет собой не столько четко выверенную статистическую систему, сколько довольно сильно подверженный влиянию психологических факторов «организм». Но несмотря на непредсказуемость психологической составляющей, трейдеры и аналитики вот уже второе столетие не оставляют попыток разработки все более новых и совершенных методов и инструментов для прогнозирования поведения рынка и анализа цен финансовых инструментов.

Изначально попытки анализа финансовых рынков сводились к анализу котировок финансовых инструментов, как одномерных динамических временных рядов. Среди современных методов анализа рынков выделяют методы системного анализа, интеллектуальные методы анализа данных, средства искусственного интеллекта и их реализацию в предметно-ориентированных системах поддержки принятия решений.

### **1.1 Постановка задачи управления на финансовых рынках на основе теории управления социально-экономическими системами**

Финансовые рынки являются одной из разновидностей сложных социально-экономических систем, управление которыми требует разработки специализированного инструментария, направленного как на анализ многокритериальной информации, так и на поддержку принятия решений.

По мере развития и совершенствования социальных и экономических институтов их анализ и моделирование с помощью стандартных математических методов и алгоритмов становилось все более трудновыполнимой задачей, что привело к появлению целой отрасли научных знаний, связанной с рассмотрением их как систем. Моделирование сложно формализуемых системных процессов становилось возможным благодаря появлению таких научных дисциплин как системный анализ, информационно-управляющие системы, прикладная информатика, управление и принятие решений.

Если рассматривать систему с точки зрения кибернетических наук, то в качестве системы может выступать простые или сложные, детерминированные или вероятностные комплексы из взаимосвязанных компонентов. При этом характер детерминированности или вероятности определяется наличием управления в системах или самоуправления. Также основной характеристикой системы является ее способность поддерживать целостных характер. Все эти определения и условия свойственны и финансовым рынкам, как одним из представителей социально-экономических систем, что позволяет говорить о возможности анализа финансовых рынков с помощью подходов к управлению социально-экономическими системами.

Основополагающим понятием теории управления социально-экономическими системами является такая категория (в смысле предельно общего понятия), как «организация», что позволяет перейти от рассмотрения управления социально-экономическими системами к теории управления организационными системами.

Организационная система является объединением объектов (людей, систем, регламентов), действующих совместно для достижения определенной цели по определенным правилам. Под управлением понимается осуществление специализированной, направленной деятельности для достижения определенного результата. А под механизмом (или технологией) управления понимается совокупность действий, направленных на принятие управленческих решений. Среди функций управления можно выделить следующие: прогнозирование, планирование, организация, учет, контроль и другие. Объектом исследований в теории управления организационных систем является собственно сами системы, а инструментом исследования является математическое моделирование [79, с. 4-9]. Подробнее о организационных системах и управлении ими как социально-экономическими и активными системами см. [78, 15].

Финансовый рынок можно рассматривать как динамические многоэлементные организационные системы с неопределенностью, в которых имеется несколько участников (трейдеры, брокерские компании, банки, инвестиционные фонды), многократно принимающих решения одновременно и независимо в условиях ограниченности обладания полной и исчерпывающей информацией о текущей ситуации на рынке. Неопределённость в финансовых рынках заключа-

ется в том, что фактически не представляется возможным в приемлемый срок для принятия оперативного управленческого решения обработать всю информацию, которая является существенными свойствами системы и описывающие ее состояние, и, соответственно которые преобразуются в параметры для принятия решений. Поэтому системы управления финансовыми рынками принимают допущение, заключающееся в анализе и обработке только ограниченного определенного набора параметров, которые принимаются за описание «текущей ситуации на финансовом рынке выбранного финансового инструмента».

Для экономических систем, каковой является рассматриваемая система «рынков выбранного финансового инструмента», свойственна также открытость, то есть возможность взаимодействия со средой. Финансовые рынки в физическом и информационном смысле взаимодействуют со средой, обмениваясь материальными ресурсами и постоянными информационными потоками с внешними задействованными объектами. При этом у внешней среды финансовых рынков большая степень неопределенности. Если рассматривать каждый отдельный рынок финансового инструмента, как отдельную систему, то во внешнюю среду такой системы также будут входить и рынки других финансовых инструментов. Ради уменьшения этой неопределенности и используются такие функции управления как мониторинг, анализ текущего состояния и прогнозирование на финансовых рынках.

Если рассматривать классификацию механизмов управления с точки зрения предмета управления, то есть какой элемент системы подлежит изменениям в процессе функционирования системы, можно выделить, например, управление составом или структурой, управление порядком функционирования, информационное управление. В данном случае под информационным управлением понимается управление той информацией о системе, которой могут обладать участники организационной системы на момент принятия управленческого решения. Для таких организационных систем, как финансовые рынки, информационное управление является критически важной частью управления системой. Обладание и анализ полноценной многокритериальной информации о текущей ситуации на финансовом рынке выбранного финансового инструмента является одним из факторов успеха при принятии управленческого решения по заключению сделок по покупке или продаже финансовых инструментов.

Одним из подвидов информационного управления в классификации по функциям управления является прогнозирование, которое выделяется как одна из технологий управления.

С помощью прогнозирования можно строить в том числе и экономические прогнозы (например, макроэкономических показателей страны или региона) и прогнозы состояния отдельных рынков. По технологии функции прогнозирования, приведенной в работе [10, с. 47], сначала необходимо исследовать информацию для анализа, выявить необходимость в прогно-

зировании, объект и цель прогнозирования, собрать данные для анализа и провести прогнозирование с помощью выбранных методов. Результат прогнозирования может быть визуализирован различными методами и предоставлен для принятия управленческого решения.

Каждый из описанных шагов можно расписать и для обоснования необходимости в анализе и управлении такой социально-экономической системой, как финансовый рынок. Необходимость прогнозирования в данной области особо сильна, поскольку успешность или не успешность прогнозов по направлению движения тех или иных рынков ценных бумаг, товарных и валютных рынков ведет не только к выигрышу конкретного трейдера или брокера, но и в сумме к стабильности или дестабилизации экономики отдельных стран и мировой экономике.

Объектом прогноза на рынках могут быть как цены выбранных акций или фьючерсов в следующий момент времени, так и направление движение рынка в следующее заданное количество временных интервалов. Направление движения рынков можно в обобщенном виде характеризовать как отнесение динамики изменения котировок за определенный будущий интервал времени к одному из трех классов: повышающий тренд (ситуация на рынке в данном случае подходит для покупки), понижающий тренд (ситуация, подходящая для продажи), боковой тренд (ситуация ожидания).

При анализе и работе с любой социально-экономической системой менеджеры постоянной сталкиваются с проблемой принятия управленческих решений, заключающиеся в выборе из различных альтернатив в направлении развития бизнеса или социального института, заключении сделок, инвестирования, снижения рисков и прочих. При этом выбранное менеджером решение должно приближать к поставленным им или системой более высокого порядка целям. Решение может приниматься спонтанно, поскольку принятие решения в целом это волевой и психологический процесс. Лицо, принимающее решения может основываться на различных подходах, например, выносить решение на основе своего или экспертного опыта, или обращаться к более сложному математическому аппарату и логическим алгоритмам для определения рационального и более оптимального решения. Если рассматривать социально-экономические системы как организационные системы, в них выделяют центр – игрок, делающий ход первым, устанавливающий правила игры для других, и агентов – игроков, делающий последующие ходы с учетом выбора первого игрока [79, с. 27]. При рассмотрении финансового рынка как организационной системы участники (трейдеры, брокерские компании) являются по сути агентами, так как первое решение о состоянии системы, задавшее направление развития системы было принято еще на стадии зарождения рынка каждого конкретного финансового инструмента.

В каждом конкретном случае участник того или иного рынка выбирает из своего множества допустимых действий с финансовым инструментом. Обозначим за  $A$  такое множество вариантов:  $A = \{\text{продажа; покупка; ожидание}\}$ . Выбранное действие обозначим за  $y$ , при этом после выбора действия с финансовым инструментом оно приводит к определенному результату  $z$  ( $z$  принадлежит множеству допустимых результатов деятельности  $A_0$ ). Для действий по заключению сделки с финансовыми инструментами логично рассматривать множество допустимых финансовых результатов, к которым привела данная сделка  $A_0 = \{\text{получена прибыль; убыток; не получена ни прибыль, ни убыток}\}$ .

При выборе действия участник рынка имеет предпочтение  $R_{A_0}$  из множества допустимых результатов торговой деятельности, которое заключается в получении прибыли от совершаемых сделок с финансовыми инструментами.

При выборе действия участник рынка с учетом своих предпочтений и функции изменения результата в зависимости от окружающей среды  $\Theta$  (текущей ситуацией на рынке выбранного финансового инструмента), информация о которой отражена в множестве  $I$ , причем множество  $I$  может входить в множество всей обстановки  $\Theta$ , но в обстановке также включается вся информация, которой не обладает по тем или иным причинам лицо принимающее решение (ЛПР).

Задача анализа финансового рынка с целью принятия рационального решения его участником сводится к построению такой целевой функции  $W_I(\cdot)$ , которая будет отражать зависимость результата  $z$  от выбранной стратегии действия  $y$  при изначально заданном информационном описании состояния системы  $I$  (подробнее о модели принятия решений в активных системах см. [80]).

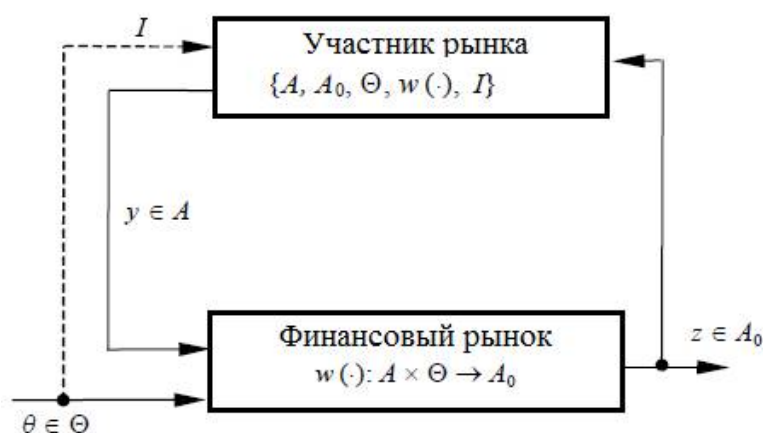


Рисунок 1.1 – Структурная схема модели принятия решения участником рынка

Для формирования данной целевой функции при управлении организационной системой могут использоваться различные подходы к анализу данных и математические модели. Подходы к анализу финансовых рынков для принятия управленческого решения разделяются как на более формализованные: технический анализ, экономико-математическое моделирование, так и



на менее формализованные, например, фундаментальный анализ. Среди формализованных можно выделить особый класс подходов интеллектуального анализа данных  $I$ , среди которых выделяется нейросетевая методология. Таким образом с помощью нейросетевой динамической системы можно смоделировать функцию принятия решения по заключению сделки для управления такой организационной системой, как финансовый рынок выбранного финансового инструмента. На основе нейросетевой динамической системы может быть реализована интеллектуальная система поддержки принятия решений участников рынка.

## 1.2 Классические методы анализа финансовых рынков

Аналитиков финансовых рынков по своим взглядам изначально можно разделить на два направления в вопросе прогнозирования будущего финансового рынка: фундаменталисты и чартисты (от слова «chart» – в переводе с английского «график»). Специалисты, придерживающиеся первого направления, строят свои прогнозы на основе различных фундаментальных факторов и событий, происходящих в мире и стране. Это могут быть новости самой различной направленности, начиная с террористических актов и заканчивая выходом макроэкономических показателей какого-либо государства. Приверженцы второго направления используют для анализа рынков графики динамики цен, находя закономерности в их динамике и прогнозируя изменение цен по различным правилам технического анализа, используя различные его инструменты, такие как индикаторы и уровни. Фундаментальный и технический анализ – оба направления по сути основываются на статистических данных о рынках, но находятся на противоположенных сторонах. Фундаментальный анализ углубляется в причины происхождения волнений, формирования и изменения трендов на финансовых рынках. В техническом анализе считается, что в результатах функционирования самого рынка – графиках цен, находят свое отражение все происходящие в мире события.

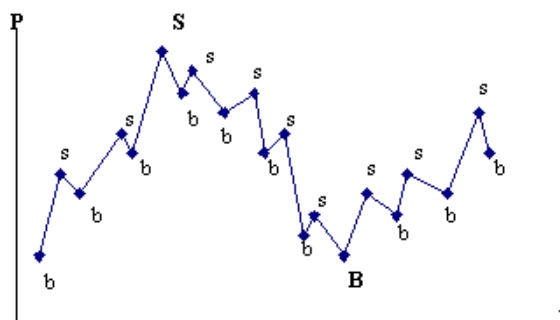


Рисунок 1.2 – Модель циклов в графиках цен

На рисунке 1.2 представлен график изменения цен на некоторый гипотетический финансовый инструмент. При предположении о заранее известном некотором количестве последующих изменений цен на рынке данного финансового инструмента будет существовать только од-

на оптимальная стратегия: продавать финансовые инструменты на пиках цен (s) и, соответственно, совершать покупку на впадинах (b). Технический анализ ставит перед собой цель определения точек разворотов в изменениях цены, концентрируясь не на возможности определения каждой из точки разворота, а на определении существенных разворотов цен (S и B на рисунке 1.2). Анализ таких моментов возможен в случае, если в рядах цен вместе со случайными изменениями присутствуют и устойчивые закономерности их сдвига (тренды), которые оказывают влияние на цены финансового инструмента в течение определенных интервалов времени. При уточнении понятия можно определить технический анализ как искусство определения на ранних этапах разворота тренда и следования выявленному тренду до тех пор, пока взвешенная оценка не покажет, что тренд опять развернулся [53, с. 3-4].

Фундаментальный анализ, в отличие от технического, содержит в себе основную информацию о рынке и позволяет найти причину изменения цен. Его можно понимать, как изучение событий, происходящих за графиками курса финансового инструмента.

История технического анализа берет начало в Японии, где еще три столетия тому назад применялась такая форма представления движения цены, как японские свечи, получившая широкое распространение в настоящее время в биржевых торговых системах [98]. История применения технического анализа в западных странах насчитывает немногим более века.

В начале девяностых годов девятнадцатого века в Wall Street Journal появилась статья Чарльза Доу, где изложены принципы, с помощью которых, можно было вступать в сделки на покупку и продажу с минимальным риском. И им была сформулирована теория, признанная классической в современной действительности, и ее принципы используются в том или ином виде во многих методах технического анализа, теорема определения направлений движения рынка акций. В 1932 году Робертом Реа теория была доведена до ее логического завершения в книге "Теория Доу".

В тридцатых годах (1930-40 гг) публиковались статьи в двух разных направлениях. Первое (например, Вайкофф и Шабакер) представляло диаграммы как графическую запись биржевого спроса и предложения. Вторые (например, Эллиотт и Ганн) занимались поиском скрытых закономерностей на графиках. Труды Ральфа Эллиотта были оформлены в Волновую Теорию, открывшую для технического анализа использование чисел Фибоначчи. В 1948 году опубликована книга "Технический анализ тенденций акций" в соавторстве Эдвардса и Маги. В ней они определили такие известные и по сей день модели, как треугольники, прямоугольники, "голова и плечи" и прочие графические модели, а также отразили идеи о линии тенденций, уровнях поддержки и сопротивления. В начале шестидесятых появились и распространились скользящие средние (СС). Основателями биржевого метода СС были Ричард Дончиан и Дж. М.

Хрест. Дончиан разработал методику биржевой стратегии, основанную на использовании нескольких скользящих средних, а Хрест в книге "Чудо-прибыльность своевременных сделок с акциями" описал принцип использования СС для сделок с акциями.

С появлением компьютеров технический анализ стал принимать более математизированную форму, начали с большой быстротой развиваться аналитические методы, использующие математическую аппроксимацию и фильтрацию. К этому времени относится создание Дж. Вэлс Уайдлером «Дирекционной системы», появление индикаторов: Процентный разброс Вильямса, Баланс объема Джозефа Гранвилла. В начале 80-х Джон Мерфи и Роберт Прихтер систематизировали методы и оформили технический анализ, как отдельную прикладную науку.

В девяностых распространилась тенденция следования за «гуру» (культовые фигуры рынка, которые, сделав себе громкое имя на нескольких удачных сделках на основе собственных прогнозов, начинали рассылать рекомендации по направлению сделок по тем или иным акциям). Но следование рынка за рекомендациями «гуру» приводило к последующему обвалу цен на рынке (или росту, если рекомендованное направление сделок было иным). Подробнее об этом феномене можно прочитать в книге Александра Элдера "Trading for a living" [110]. Подобное следование за мнением значимого игрока рынка и последующие обвалы свидетельствуют о важной и весомой «психологической составляющей» в модели функционирования финансовых рынков как систем.

Довольно новое направление в прогнозировании деятельности финансовых рынков – эконометрическое моделирование. Эконометрические методы основаны на выявлении сложных зависимостей временных рядов цен финансового инструмента, построении математических моделей. В данных математических моделях котировки акций и фьючерсов представляются в виде временных рядов, динамику которых необходимо предсказать, например, на основе предыдущих значений, что приводило к использованию авторегрессионных рядов, либо построение моделей зависимости котировок, как основного фактора, от различных критериальных факторов, таких как объем продаж, отраслевые индексы. По своей сути данные методы близки к индикаторам технического анализа. Но являются более сложными и требующими специальной математической подготовки инструментами по сравнению с большинством технических индикаторов и прочих способов технического анализа, поэтому они менее доступны любому инвестору или трейдеру.

Не смотря на свои различия все методы анализа финансовых рынков придерживаются единых постулатов:

1. Рынок учитывает все. Суть постулата заключается в том, что любой фактор, влияющий на стоимость ценной бумаги: экономический, политический, психологический, – заранее учтен

и уже отражен на ценовом графике. Главным следствием этой предпосылки является необходимость внимательного отслеживания и изучения динамики цен. Эта предпосылка конфликтует с фундаментальным анализом, в котором основное внимание уделено изучению факторов, после анализа которых представляются выводы относительно движения рынка.

2. Движение цен подчинено тенденциям. Это предположение стало основой для создания всех методик технического анализа, поскольку рынок подверженный тенденциям в отличие от хаотичного рынка можно анализировать.

3. История повторяется. Технический анализ и исследования динамики финансовых рынков неразрывно связаны с изучением их психологической составляющей. Курс ценной бумаги представляет собой соглашение. Это цена, по которой один инвестор согласен купить, а второй – продать. Величина, которая устраивает и покупателя, и продавца обусловлена их ожиданиями. Если инвестор убежден, что в будущем курс бумаги будет подниматься, то он будет покупать; а если он убежден в его падении, он будет продавать. Таким образом прогнозирование цен связано с человеческими ожиданиями.

#### 1.1.1 Фундаментальный анализ

Фундаментальный анализ развивался наряду со вторым важным направлением в анализе финансовых рынков – техническим анализом. Его последователи утверждают, что настоящие причины движения цен на тот или иной финансовый инструмент могут быть объяснены и спрогнозированы, основываясь на изучении различных макроэкономических показателей и типичные реакции рынка на публикацию тех или иных данных.

Фундаментальный анализ используется как для прогнозирования развития рынка на длительных промежутках времени (месяц, год), так и в некоторой модификации может использоваться и для анализа вхождения в короткие позиции, что используется реже и вызывает споры среди аналитиков.

Основным трудом по фундаментальному анализу американской школы является классический труд Б. Грэма и Д. Додда «Анализ ценных бумаг» 1934 года. В нем авторы опираются на макроэкономические индикаторы и индексы деловой активности рынка. Изначально, фундаментальный анализ применялся только для оценки стоимости компаний-эмитентов и, соответственно, ее акций, при этом анализировались такие финансовые показатели компании, как чистая прибыль, EBITDA, чистая стоимость и денежный поток. Инвесторы, основываясь на данных показателях, принимали решение о недооцененности или переоцененности цен акций компаний. В настоящее время понятие фундаментального анализа значительно расширилось, и теперь в него входят и анализ экономических, политических событий и даже природных ката-

клизмов. А с развитием средств вычислительной техники и интернет-технологий экономические события стали доступны инвесторам и трейдерам в режиме реального времени, что ускоряет реакцию рынка на то или иное событие.

Основные факторы, влияющие на изменение цен финансового инструмента, можно разбить на три основные группы:

- политические явления, кризисные и природные явления;
- финансовая политика;
- экономические события.

К первой группе относятся такие события, как встречи основных политических лидеров стран, анализ рынков которых производится, различные саммиты, переговоры по заключению крупных международных сделок. Также кризисные явления, как финансового, так и социального или природного характера: падение рынков, крах крупных банков, введение военного положения, народные волнения, революции, террористические акты, и даже наводнения и землетрясения, – могут оказывать большое влияние на экономику исследуемых стран.

Ко второй группе в первую очередь можно отнести важнейший показатель – ключевая процентная ставка страны, здесь же рассматривают и остальные составляющие финансовой политики: бюджетная, налоговая, инвестиционная и кредитно-денежная политика.

К третьей группе относятся публикация макроэкономических показателей и индикаторов. Макроэкономический фундаментальный анализ имеет целью выявить общие тенденции, характеризующие экономику страны в целом и крупные ее секторы. Выделяют основные макроэкономические показатели в следующих сферах:

- валовой внутренний продукт, производство и услуги;
- рынок труда, доходы и занятость;
- инфляция;
- фондовые рынки.

Фундаменталисты выделяют новости ожидаемые (например, публикация плановых макроэкономических показателей) и случайные (такие как стихийные бедствия, террористические акты), с длинным (месяц, год) и коротким (неделя, день, внутри дня) циклом жизни факторов, на мировые новости, новости отдельного государства и даже отдельного предприятия [45].

Несмотря на долгую и успешную историю применения фундаментального анализа, многие современные рыночные аналитики воспринимают его как рудимент, основываясь на положении «рынок учитывает все», утверждающее, что все макроэкономические показатели уже заложены в самих изменениях котировок исследуемого финансового инструмента. Также подвергается критике возможность оценки результатов фундаментального анализа, так как в случае

если фундаментальный прогноз оправдывается, во-первых, это можно объяснить с точки зрения технического анализа, во-вторых, это может быть случайностью. Согласно исследованию известного трейдера Ларри Вильямса, прогнозы аналитиков с Уолл-стрит в отношении изменения цен финансовых инструментов сбываются лишь в 34 % случаев, что даже меньше, чем если бы прогнозы делались случайным образом [103].

Но даже самый скептический критик, все же не может не согласиться с тем, что основные экономические новости и события оказывают определенное влияние на котировки финансовых инструментов. Поэтому в современной действительности редкий трейдер не использует в своей повседневной торговле информацию об экономических новостях и событиях, тем более, что с появлением интернет-экономических календарей, такая возможность стала максимально доступна.

### 1.1.2 Технический анализ финансовых рынков

Технический анализ – метод анализа и прогнозирования изменений цен (котировок) финансового инструмента в будущем на основе анализа их изменений в прошлом. В его основе лежит анализ временных рядов цен, отображаемых на графиках изменений цен («чартах» – от английского *chart*). В техническом анализе дополнительно может использоваться информация об объемах торгов, количестве покупок и продаж конкретного финансового инструмента в отдельный период времени и прочие статистические данные о финансовом рынке. Наиболее часто методы технического анализа используются для анализа цен, изменяющихся свободно, например, на биржах, финансовых рынках.

Возникновению технического анализа предшествовали наблюдения изменений цен на различных финансовых рынках на протяжении нескольких веков. В конце 19-го века американский журналист Чарльз Доу опубликовал серию статей о рынках ценных бумаг, которые легли в основу его теории. Целью теории Доу является выявление изменений в различных колебаниях рынка. Выявленный тренд временного ряда предполагается существующим до момента, пока не обнаружится точка его разворота. Теория описывает лишь направление тренда, и не прогнозирует продолжительности тренда или размеров изменений внутри него.

Согласно данной теории, большинство акций практически все время следует определенному внутреннему тренду рынка. Для оценки подобного "состояния рынка" Доу сконструировал два индекса, один из которых называется в настоящее время «Индустриальным индексом Доу-Джонса» (DJI), который рассчитывался сперва по ценам 12 (в настоящее время 30) акций крупных компаний, так называемых "голубых фишек", второй индекс – «Индекс железных дорог Доу-Джонса», строившийся по ценам 12 акций железнодорожных компаний. Теория Доу предусматривает положение, что изменения в ценах закрытия отражают совокупную оценку и

стремления всех участников рынка, как действительных, так и потенциальных. Данная теория основывается на соотношении цен и объемов продаж, определяет направление тренда по уровням поддержки. Индустриальный индекс и Индекс железных дорог должны рассматриваться совместно, то есть изменения, происходящие в одном из индексов, должны подтверждаться изменениями в другом [53, с. 13-19].

Большое влияние на развитие технического анализа оказала волновая теория Эллиота. Основа принципа: изменения финансовых рядов имеют циклический характер. Концепция основана на последовательности чисел Фибоначчи. Используя характер естественных циклов Эллиотт показал на 80-летнем периоде существование устойчивых циклов из 8 волн: 5 из которых поднимают рынок, а затем 3 опускают его (рисунок 1.3).

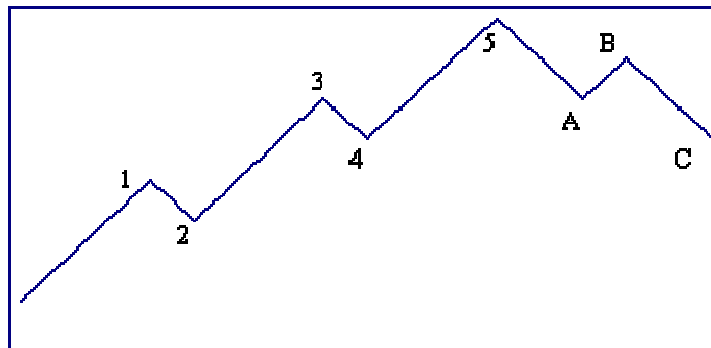


Рисунок 1.3 – Цикл Эллиота

Основной самый продолжительный цикл в теории Эллиотта – Большой Суперцикл, состоящий из 8 волн Суперцикла, каждая из которых в свою очередь делится на 8 волн меньшего цикла и т.д. Процесс дробления продолжается, выявляя главные, промежуточные, минутные волны.

По теории Эллиотта полный возрастающий-убывающий («бычий-медвежий») цикл заключен между впадинами «медвежьих» фаз и состоит из 8 волн. Возрастающая фаза состоит из пяти волн: трех волн продвижения вверх (первая, третья и пятая – волны импульса) и двух волн коррекции или консолидации (вторая и четвертая – волны коррекции). Убывающая фаза состоит из трех волн: двух волн импульса и одной волны коррекции.

Развитие компьютерных технологий во второй половине 20-го века способствовало усовершенствованию инструментов и методов анализа, а также появлению новых методов, использующих возможности вычислительной техники. В настоящее время появляются новые индикаторы рынков, развиваются методики выявления зависимостей в изменениях цен, формулируются новые теории анализа и прогнозирования финансовых рынков, совершенствуются стратегии торговли финансовыми инструментами. Но в основе всех самых современных методов техниче-

ского анализа лежат логика и принципы поведения игроков финансовых рынков, заложенные в, ставших уже классическими, теориях Доу и Эллиота.

В настоящее время торговые системы предлагают трейдерам большое количество технических индикаторов. Под техническим индикатором понимается некая функция, математическая зависимость, построенная на основе статистических показателей торгов, анализ поведения которой позволяет отследить изменения текущих тенденций на финансовом рынке.

Выделяют несколько групп технических индикаторов:

- индикаторы тенденций – подтверждают тенденцию рынка;
- осцилляторы – предсказывающие разворотные точки для текущей тенденции;
- каналные индикаторы – показывающие пределы текущей тенденции.

Наиболее популярными в группе индикаторов тенденций являются классические скользящие средние, также используют различные модификации методов пересечения разнопериодных скользящих средних (Аллигатор), и индикатор схождения-расхождения скользящих средних (MACD).

К широко используемым осцилляторам относятся Williams %R, индекс относительной силы (RSI), Стохастический осциллятор, которые показывают состояния перекупленности или перепроданности финансового инструмента, используя определенные статистические закономерности. Самым распространенным каналным индикатором являются линии Боллинджера. К этой же группе можно отнести линию тренда, линии поддержки и сопротивления, показывающие психологически значимые уровни.

У каждого профессионального трейдера есть свой комплекс технических индикаторов, который он использует для анализа того или иного финансового рынка и/или инструмента. Зачастую, именно одновременное использование нескольких технических индикаторов из разных групп могут привести к наилучшему результату по анализу ситуации на рынке и заключению торговых операций.

Минусами технического подхода к анализу рынков является непредсказуемость реакций рынка, которые могут возникать при публикации различного рода экономических показателей и совершения политических, экономических событий, касающихся стран-эмитентов анализируемого финансового рынка, которые никак не учитываются при построении индикаторов технического анализа. И зачастую, трейдеры, не учитывающие перечисленные экономические события, попадают в неожиданные убыточные позиции, хотя по всем индикаторам их сделка была открыта в необходимом направлении. В связи с этим, большинство технических аналитиков в некоторой степени применяют в своей работе и инструменты фундаментального анализа.



### 1.3 Эконометрический анализ финансовых рынков

Для трейдера главной целью является предугадать направление движения цен финансовых инструментов, для этого в его распоряжение есть, в большинстве случаев, только графики изменения цен, которые по своей сути являются временными рядами. Проблемами анализа и прогнозирования временных рядов занимается научная дисциплина эконометрика.

Термин эконометрика введен Рагнаром Фришем в 1930 году для обозначения нового направления научных исследований, возникшего из необходимости научно-обоснованного подтверждения и доказательства концепций и выводов экономической теории результатами количественного анализа рассматриваемых процессов. Основной задачей эконометрики является построение эконометрических моделей, описывающих взаимообусловленное развитие социально-экономических процессов, на основе информации, отражающей распределение их уровней во времени или в пространстве однородных объектов. Подобные модели могут быть использованы для анализа и прогнозирования общих закономерностей и конкретных количественных характеристик рассматриваемых процессов, для определения управляющих воздействий [99, стр. 7].

При использовании эконометрики в исследовании предполагается, что закономерности моделируемого процесса образуются под влиянием ряда других факторов. Обобщенная форма эконометрической модели, описывающей закономерности развития такого процесса, обозначенного переменной  $y$ , в зависимости от уровней, воздействующих на него внешних явлений, факторов  $x_i, i=1, 2, \dots, n$ , может быть представлена следующим уравнением:

$$y_t = f(\alpha, x_t) + \varepsilon_t, \quad (1.1),$$

где  $f(\alpha, x_t)$  – функционал, выражающий взаимосвязи между уровнями переменных  $y_t$  и  $x_t$  в моменты времени  $t=1, 2, \dots, T$  (или на интервалах  $(t, t+1)$ );  $x_t = (x_{1t}, x_{2t}, \dots, x_{nt})$  – вектор значений независимых переменных (факторов) в момент  $t$ ;  $\alpha = (\alpha_0, \alpha_1, \dots, \alpha_n)$  – вектор параметров модели; параметр  $\alpha_i$  выражает степень влияния фактора  $x_i$  на переменную  $y$  на всем рассматриваемом интервале  $(1, T)$ ;  $\alpha_0$  – постоянная модели;  $\varepsilon_t$  – случайная ошибка модели в момент  $t$ , в отношении свойств и характеристик которой выдвигают дополнительные предположения.

Достаточно часто их значения интерпретируются как детерминированные величины. В этом случае при ошибке модели, обладающей свойствами "белого шума"\* , наблюдаемые значения  $y_t$  можно рассматривать как условное распределение переменной  $y$  при заданных значениях

---

\* Т. е. ошибка обладает нулевым математическим ожиданием  $M[\varepsilon_t]=0$ , ее дисперсия постоянна на всех участках рассматриваемого периода времени, а разновременные значения  $\varepsilon_t$  и  $\varepsilon_{t-j}, j=1, 2, \dots$ ; независимы.

$x_{it}$ ,  $i=1, 2, \dots, n$ ;  $t=1, 2, \dots, T$ . Математическим ожиданием такого распределения является функционал  $f(a, x)$  [99, с. 9].

$$\hat{y}_t = M[y_t | x_{1t}, x_{2t}, \dots, x_{nt}] = f(a, x_t) \quad (1.2)$$

В общем случае процедуру построения эконометрической модели можно представить, как последовательность взаимосвязанных этапов. Основные этапы:

1. Анализ специфических свойств рассматриваемых явлений и процессов и обоснование класса моделей, подходящих для описания данных явлений и процессов.
2. Оценка параметров выбранной модели на основании исходных данных, выражающих уровни показателей (переменных) в различные моменты времени или на совокупности однородных объектов.
3. Проверка качества построенной модели, обоснование вывода о возможности (целесообразности) ее использования в ходе эконометрического исследования.
4. При выводе о нецелесообразности использования построенной эконометрической модели в дальнейших исследованиях следует вернуться к пункту 1 и попытаться построить более качественную модификацию модели [99, с. 15].

Одним из наиболее распространенных способов моделирования тенденции временного ряда является построение аналитической функции, характеризующую зависимость уровней ряда от времени, или функции тренда. Данный способ называют аналитическим выравниванием временного ряда. По внешнему виду графика зависимости уровней ряда от времени можно сделать заключение о присутствии или отсутствии тренда.

Временной ряд является совокупностью значений какого-либо показателя (переменной) за несколько последовательных моментов или периодов времени.

В анализе временных рядов предполагается, что данные содержат систематическую составляющую, состоящую в некоторых случаях из нескольких компонент, и случайный шум, то есть ошибку, который затрудняет обнаружение регулярных компонент. Большинство методов исследования временных рядов включает различные способы фильтрации шума, позволяющие увидеть регулярную составляющую более отчетливо.

Зависимость от времени может принимать разные формы (см. рисунок 1.4), поэтому для формализации тренда можно применять различные функции:

– линейный:  $\hat{y}_t = a + b \cdot t$ ;

– гипербола:  $\hat{y}_t = a + b/t$ ;

– экспоненциальный тренд:  $\hat{y}_t = e^{a+bt}$

– полином  $k$ -ого порядка:  $\hat{y}_t = a + b_1 \cdot t + b_2 \cdot t^2 + \dots + b_k \cdot t^k$ .

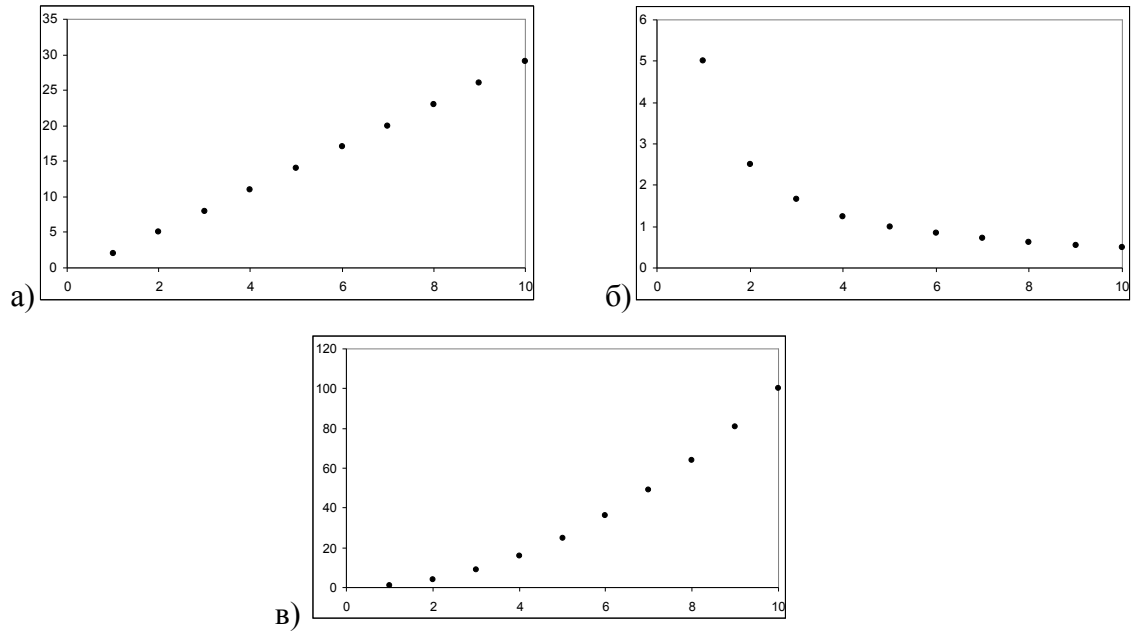


Рисунок 1.4 – Зависимость уровней ряда от времени: а) линейная; б) гиперболическая; в) степенная.

Параметры каждого из перечисленных выше трендов можно определить МНК (метод наименьших квадратов), используя в качестве независимой переменной время  $t = 1, 2, \dots, n$ , а в качестве зависимой переменной – фактические уровни временного ряда  $y_t$ .

Величина достоверности аппроксимации может быть определена с помощью коэффициента детерминации. Коэффициент детерминации ( $R^2$ ) рассматривается как мера качества регрессионной модели, описывающей связь между зависимой и независимыми переменными модели. Он показывает, какая доля дисперсии результативного признака объясняется влиянием независимых переменных.

$$R^2 = 1 - \frac{\sum (y - \hat{y}_t)^2}{\sum (y - \bar{y})^2} \quad (1.3)$$

Индекс детерминации используется для проверки существенности в целом уравнения регрессии по  $F$ -критерию Фишера:

$$F = \frac{R^2}{1 - R^2} \cdot \frac{n - m - 1}{m} \quad (1.4),$$

где  $R^2$  – индекс детерминации;

$n$  – число наблюдений;

$m$  – число параметров при переменной  $t$ .

Величина  $m$  характеризует число степеней свободы для факторной суммы квадратов, а  $(n - m - 1)$  – число степеней свободы для остаточной суммы квадратов. Нулевая гипотеза  $F$ -критерия заключается в равенстве факторной и остаточной дисперсий на одну степень свободы.

Табличное значение  $F$ -критерия – это максимальная величина отношения дисперсий, которая может иметь место при случайном их расхождении для данного уровня вероятности наличия нулевой гипотезы. Вычисленное значение  $F$ -отношения признается достоверным, если оно больше табличного. В этом случае нулевая гипотеза об отсутствии связи признаков отклоняется и делается вывод о существенности этой связи. Если же величина окажется меньше табличной  $F_{\text{факт}} < F_{\text{табл}}$ , то нулевая гипотеза отклоняется, уравнение регрессии считается статистически не значимым [34, с. 48-53, 85, 234].

Применение модели временных рядов нашли широкое распространение в исследованиях финансовых рынков, в анализе динамики финансовых показателей, прогнозировании цен на различные товары, курсов акций, соотношений курсов валют. Общим для всех моделей временных рядов является предположение о том, что текущее значение процесса  $y_t$  в значительной степени предопределено его предыдущей историей, т. е. величина переменной (показателя)  $y_t$  генерируется значениями  $y_{t-1}, y_{t-2}, \dots$  согласно специфичным для этого временного ряда закономерностям. Математически это допущение выражается следующим уравнением:

$$y_t = f(y_{t-1}, y_{t-2}, \dots) + \varepsilon_t \quad (1.5),$$

где  $\varepsilon_t$  – случайная ошибка модели в момент  $t$ .

Линейные модели временных рядов применяются для описания стационарных процессов (обычно принимается стационарность второго порядка). Для стационарного процесса  $n$ -го порядка характерно постоянство значений всех моментов порядка  $n$  и ниже на всех временных отрезках, входящих в интервал  $t=1, 2, \dots, T$ . На практике проверяют совпадение на двух временных промежутках оценки математических ожиданий, дисперсий и коэффициентов автокорреляции  $i$ -го порядка.

Для проверки соответствия ряда стационарному процессу применяют различные специализированные тесты: непараметрические, полупараметрические и параметрические тесты. Одними из используемых параметрических критериев при проверке процесса на стационарность являются критерии Стьюдента и Фишера. Среди непараметрических тестов используют такие как тест Манна-Уитни, Тест Сиджела-Тьюки, сериальные критерии стационарности [99, с. 264-273].

Стоит отметить, что большинство финансовых временных рядов редко проявляют свойства стационарности. Поэтому при их анализе с помощью методов эконометрического моделирования применяют преобразования нестационарных временных рядов в стационарные. Примерами таких преобразований могут служить взятие конечных разностей, логарифмирование, расчет темпов прироста. Таким образом, получая стационарные временные ряды из исходных

получаем ряд, особенности которого могут быть объяснены характером его автокорреляционной функции. В общем случае все множество стационарных рядов второго порядка можно разделить на несколько групп, для каждой из которых можно построить адекватную модель: модель авторегрессии, модель скользящего среднего или смешанные модели авторегрессии-скользящего среднего.

Использование модели авторегрессии (AR( $k$ )) предполагает, что закономерность стационарного процесса второго порядка может быть выражена в виде линейной комбинации некоторого количества предыдущих его значений и случайной ошибки:

$$y_t = \alpha_1 y_{t-1} + \alpha_2 y_{t-2} + \alpha_k y_{t-k} + \varepsilon_t \quad (1.6),$$

где  $y_t, y_{t-i}, i=1, 2, \dots, k$  – значения переменной  $y$  в соответствующие моменты времени;  $k$  – порядок модели;  $\alpha_1, \dots, \alpha_k$  – коэффициенты модели;  $\varepsilon_t$  – случайная ошибка.

Решение задачи подбора адекватных коэффициентов модели может быть выполнено с использованием уравнений Юла-Уокера, коэффициенты автокорреляции полученные из данной системы уравнений называют коэффициенты модели авторегрессии Юла-Уокера. Качество оценок Юла-Уокера может быть проверено исследованием ряда ошибок  $\varepsilon_t$ : если ее свойства близки к характеристикам белого шума, оценки принимаются достаточными.

В моделях скользящего среднего (СС( $m$ )) текущее значение ряда представляют в виде линейной комбинации значений ошибки.

$$y_t = \varepsilon_t - \beta_1 \varepsilon_{t-1} - \beta_2 \varepsilon_{t-2} \dots - \beta_m \varepsilon_{t-m} \quad (1.7),$$

где  $\beta_1, \beta_2, \dots, \beta_m$  – параметры модели.

Системы уравнений для нахождения коэффициентов модели, по сравнению с системой уравнений Юла-Уокера, является нелинейной, и ее решение требует применения специальных итеративных процедур расчетов.

Развитие исследований временных рядов показало, что довольно эффективным является использование моделей смешанного типа, которые сочетают в себе свойства моделей авторегрессии и скользящего среднего. Общий вид модели авторегрессии-скользящего среднего (АРСС( $k, m$ )):

$$y_t = \alpha_1 y_{t-1} + \alpha_2 y_{t-2} + \alpha_k y_{t-k} + \varepsilon_t - \beta_1 \varepsilon_{t-1} - \beta_2 \varepsilon_{t-2} \dots - \beta_m \varepsilon_{t-m} \quad (1.8),$$

где  $\alpha_1, \dots, \alpha_k, \beta_1, \beta_2, \dots, \beta_m$  – коэффициенты модели,  $\varepsilon_t$  – случайная ошибка,  $k$  – порядок авторегрессии;  $m$  – порядок скользящего среднего [99, с. 302-314].

Расширением модели АРСС для нестационарных временных рядов, преобразуемые взятием разностей от исходного временного ряда, являются модели проинтегрированная модель ав-

торегрессии-скользящего среднего (АРПСС( $k, d, m$ ), ARIMA, модель Бокса-Дженкинса, 1976). Отличие от модели АРСС заключается в порядке интегрирования ряда  $d$ , то есть порядок взятие разностей от исходного ряда. Общий вид уравнения модели АРПСС( $k, d, m$ ):

$$\Delta^d y_t = c + \alpha_1 \Delta^d y_{t-1} + \alpha_2 \Delta^d y_{t-2} + \alpha_k \Delta^d y_{t-k} + \varepsilon_t - \beta_1 \varepsilon_{t-1} - \beta_2 \varepsilon_{t-2} \dots - \beta_m \varepsilon_{t-m} \quad (1.9),$$

Дальнейшими шагами в анализе финансовых временных рядов стали попытки работы с предсказуемостью ошибки ряда и прогнозированию его дисперсии, что привело к появлению таких моделей как ARCH (Энгл, 1982) и GARCH (Боллерслев, 1986) – моделей, основанных на зависимости условной дисперсии от предыдущих значений самой дисперсии и исходного временного ряда.

Таким образом анализ и прогнозирование временных рядов является довольно разработанной темой для научных изысканий, берущих свое начало еще у основ эконометрики. Напомним, что важнейшей целью проведения эконометрического анализа является построение модели, на основе которой можно совершать прогнозирование. Качество получаемого прогноза определяется характером прогнозируемых величин, исходные данные, статистическая модель, описывающая данные, метод, которым оценивается модель, цели прогнозирования.

Возможности применения эконометрических методов для прогнозирования финансовых рынков очень широки, и данное направление в настоящее время находит все новые подтверждения и доказательства в научных исследованиях.

#### **1.4 СППР и методы интеллектуального анализа данных**

##### **1.4.1 Системы поддержки принятия решений**

Система поддержки принятия решений (СППР) (англ. Decision Support System, DSS) – компьютерная автоматизированная система, целью которой является помощь людям, принимающим решение в сложных условиях для полного и объективного анализа предметной деятельности [93]. Именно такими свойствами обладают процедура принятия решений на финансовых рынках, и, соответственно, применение подобного рода систем является очень востребованным в сфере анализа финансовых рынков.

Современные системы поддержки принятия решения используются для решения различного рода задач в повседневной управленческой деятельности, являются инструментом, призванным оказать помощь лицам, принимающим решения (ЛПР). С помощью СППР могут быть решены некоторые неструктурированные или слабоструктурированные задачи, в том числе подходят для выбора решений в многокритериальных задачах.

Теория систем поддержки принятия решений находится на пересечении мультидисциплинарных исследований, включающих теории баз данных, искусственного интеллекта, интерак-

тивных компьютерных систем, методов имитационного моделирования и различных методов анализа данных.

В настоящее время нет общепринятого определения СППР, так как архитектура и особенности реализации СППР существенно зависят от вида задач, для решения которых она разрабатывается, от имеющихся данных о моделируемой системе, информации и знаний, а также от пользователей системы.

СППР обладает следующими четырьмя основными характеристиками [47]:

1. СППР использует и данные, и модели;
2. СППР предназначены для помощи менеджерам в принятии решений для слабоструктурированных и неструктурированных задач;
3. Они поддерживают, а не заменяют, выработку решений менеджерами;
4. Цель СППР – улучшение эффективности решений.

Можно говорить о максимально удовлетворяющей понятию СППР системе, если она соответствует следующим параметрам:

- способна оперировать со слабоструктурированными решениями;
- может быть применена ЛПР различного уровня;
- может быть адаптирована для группового или индивидуального использования;
- может поддерживать взаимозависимые и последовательные решения;
- поддерживает 3 фазы процесса решения: интеллектуальную часть, проектирование и выбор;
- поддерживает различные методы решения;
- способна адаптироваться к изменениям как предприятия, и его окружения;
- проста в использовании пользователями и модификации системы разработчиками;
- улучшает эффективность процесса принятия решений;
- дает возможность пользователю управлять процессом принятия решений;
- может быть использована эволюционно, адаптируясь к новым требованиям;
- может быть легко построена по сформулированной логике и архитектуре;
- поддерживает моделирование;
- позволяет использовать накопленные знания [133].

В зависимости от обрабатываемых данных СППР условно можно разделить на оперативные и стратегические. Оперативные СППР используются в случаях необходимости моментального реагирования на изменения текущей ситуации в управлении экономическими и социальными процессами компании. Стратегические СППР используются в случае необходимости анализа значительных объемов разнородной информации, которая может поступать из различных

источников. Основной целью СППР подобного рода является поиск наиболее приемлемых вариантов развития бизнеса компаний с учетом влияния разных факторов, среди которых можно выделить конъюнктуру различных рынков, являющихся целевыми для конкретной компании, изменения рынков капиталов, финансовых рынков, законодательства и т.п.

В некоторых определениях упоминается возможность: включения в состав СППР функциональных возможностей искусственного интеллекта. Существует связанное понятие – Business Intelligence Tools (инструментальные средства бизнес-интеллекта) – программное обеспечение, предоставляющее пользователям возможность наблюдения и использования больших объемов сложных данных.

Можно выделить три типа инструментальных средств BI:

1. Средства многомерного анализа (On-line Analytical Processing, OLAP) – программное обеспечение, которое дает пользователю возможность наблюдать данные в различных измерениях, направлениях и сечениях.
2. Инструментальные средства запросов (Query Tools) – программное обеспечение, которое позволяет сформировать запросы к данным по содержанию или образцу (паттерну).
3. Инструментальные средства поиска данных (Data Mining Tools) – программное обеспечение, осуществляющее автоматический поиск важных образцов (паттернов), или неявных зависимостей в данных.

В сфере бизнеса, управления и финансов можно выделить различные классы аналитических задач, для решения которых требуется использование статистических методов:

- горизонтальный или временной анализ;
- трендовый анализ (прогноз);
- вертикальный или структурный анализ;
- сравнительный или пространственный анализ;
- анализ относительных показателей;
- факторный анализ.

Для задач анализа финансовых рынков наиболее подходящими являются методы поиска знаний и полезной информации в большом наборе анализируемых данных, осуществляющие выявление некоторых сложно параметризуемых паттернов, образцов, по которым действует исследуемая система.

#### 1.4.2 Исторический обзор развития интеллектуальных методов анализа данных

Data Mining (добыча, извлечение данных) – собирательное название, используемое для обозначения совокупности методов обнаружения в данных ранее неизвестных, нетривиальных, практически полезных и доступных для интерпретации знаний, необходимых для принятия ре-



шений в различных сферах человеческой деятельности.

Основой Data Mining являются различные методы классификации, прогнозирования и моделирования, основанные на применении таких средств и методологий, как деревья решений, генетические алгоритмы, эволюционное программирование, искусственные нейронные сети, нечеткая логика. К методам Data Mining также относятся статистические методы: дескриптивный, корреляционный и регрессионный анализ, факторный анализ, дисперсионный анализ, компонентный анализ, дискриминантный анализ, анализ и прогнозирование временных рядов.

Важной особенностью методов Data Mining является возможность наглядного представления результатов произведенных с их помощью вычислений, что позволяет пользователям, не обладающим специальной математической подготовкой, также в полной мере использовать инструментарий Data Mining. В то же время, применение статистических методов анализа данных требует хорошего владения теорией вероятностей и математической статистикой.

Основоположителем области Data Mining является Григорием Пятецким-Шапиро, который провел специализированный семинар по данной тематике в 1989 году. Григорий Пятецкий-Шапиро заинтересовался вопросом: существует ли возможность автоматически находить определённые правила, для ускорения некоторых сложных запросов к крупным базам данных. Им было предложено два термина – Data Mining («извлечение данных») и Knowledge Discovery In Data («открытие знаний в базах данных»). В 1993 году начала выходить рассылка «Knowledge Discovery Nuggets», а в 1994 году был открыт доступ к одному из первых сайтов по Data Mining [118].

Появление технологий DM&KDD (Data Mining and Knowledge Discovery in Databases) обусловлено накоплением больших объемов информации в компьютерных базах данных, которые стало невыгодно хранить, и использование которых традиционными способами вызывало затруднения. Возникает необходимость часто модифицировать накопленные данные из-за быстрой смены аппаратного и программного обеспечения баз данных, при этом не редко встречаются потери и искажение информации. Одним из способов для исправления подобных ситуаций является создание информационных хранилищ данных. Второй подход связан со сжатием данных и выделением из них наиболее полезной информации, такой подход более интересен для специалистов в области искусственного интеллекта, так как связан с решением проблемы выявления новых знаний.

Data Mining представляет собой область знаний, возникшую на стыке таких наук как математическая статистика, распознавание образов, машинное обучение, искусственный интеллект, теория баз данных и др. (см. рисунок 1.5).

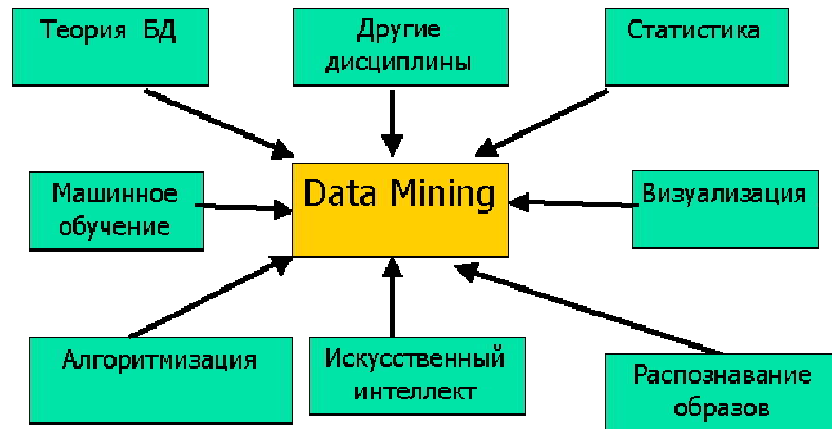


Рисунок 1.5 – Data Mining как мультидисциплинарная область [36]

Задача анализа данных изначально ставилась следующим образом: имеется достаточно большая база данных, и предполагается, что в ней существуют «скрытые знания». Необходимо разработать методы обнаружения этих знаний, скрытых в больших объемах исходных «сырых» данных [118].

Под обнаружением знаний из «сырых» данных понимаются получение:

- ранее неизвестных знаний, которые должны быть новыми (а не подтверждающими какие-то ранее полученные или заранее известные сведения);
- нетривиальных знаний, которые нельзя просто так увидеть (при непосредственном визуальном анализе данных или при вычислении простых статистических характеристик);
- практически полезных знаний, которые представляют ценность для исследователя или потребителя;
- доступных для интерпретации знаний с возможностью их визуализации для пользователя в довольно понятной и удобной форме, а также знания, которые можно разъяснить в терминах предметной области.

Проблемой извлечения и структурирования знаний интересовались еще намного раньше. Первые работы по теории распознавания и классификации по прецедентам появились в 30-х годах двадцатого века, они были основаны на байесовской теории принятия решений (Нейман, Пирсон), на применении разделяющих функций к задаче классификации (Фишер), на решении вопросов проверки гипотез (Вальд) [39]. В 50-х годах начали появляться первые нейросетевые модели распознавания образов (перцептрон Розенблата), связанные с успехами в области моделирования головного мозга. В конце 60-х годов были разработаны и детально исследованы различные подходы для решения задач распознавания в рамках статистических, перцептронных моделей, и моделей с разделяющими функциями.

Большой вклад в развитие теории распознавания и классификации внесли советские и, в последующем, российские ученые: Айзерман, Браверман, Розоноэр (метод потенциальных

функций), Вапник, Червоненкис (статистическая теория распознавания, метод «обобщенный портрет»), Мазуров (метод комитетов), Ивахненко (метод группового учета аргументов), Загоруйко (алгоритмы таксономии и анализа знаний), Лбов (логические методы распознавания и поиска зависимостей). В начале 60-х академик РАН Журавлев предложил алгоритм распознавания, состоящий в логическом методе эффективного решения задач распознавания при условиях малого числа обучающих прецедентов. На текущий момент в данной области теории распознавания ведутся исследования членом корреспондентом РАН Рудаковым (общая теория проблемно-ориентированного алгебраического синтеза корректных алгоритмов), членом корреспондентом РАН Матросовым (статистическое обоснование алгебраического подхода), Рязановым (оптимизация моделей классификации, коллективные решения задач кластерного анализа), Дюковой (асимптотически-оптимальные логические алгоритмы), Сенько (алгоритмы взвешенного статистического распознавания), Асланян (логические алгоритмы распознавания) и другими исследователями в России, СНГ и зарубежья.

Также в России и ведущих зарубежных странах активно ведутся разработки программных систем анализа данных. К подобного рода разработкам относятся статистические пакеты обработки данных и визуализации (SPSS, STADIA, STAT GRAPHICS, STATISTICA, Олимп: Стат-Эксперт Prof., и другие), которые используют для анализа данных различные инструменты математической статистики: проверка статистических гипотез, регрессионный анализ, дисперсионный анализ, анализ временных рядов, и др. Применение статистического программного обеспечения является эффективным инструментом анализа данных, и, в первую очередь, на начальных этапах исследований, когда находятся значения различных усредненных показателей, проверяется статистическая достоверность различных гипотез, выявляются корреляционные и регрессионные зависимости.

Несмотря на широкое распространение, статистические подходы имеют и ряд существенных недостатков. Позволяя оценить статистическую достоверность значения прогнозируемой величины, гипотезы или зависимости, сами методы имеют некоторые ограничения. В первую очередь, подобные методы оперируют усредненными по выборке величинами, которые являются достаточно приближительными по отношению к реальным анализируемым или прогнозируемым параметрам. Следующим ограничением является использование в статистических моделях понятий «случайных событий», «функций распределения случайных величин», хотя в реальной ситуации различные параметры и величины объектов исследования являются определяемыми по некоторым законам, то есть детерминированными. При применении статистических методов необходимо наличие конечного числа наблюдений для обоснованности полученного результата, но данное число может превышать имеющиеся для анализа данные о исследуемом объекте

или процессе. Таким образом в ситуациях анализа недостаточных или непредставительных данных, а также на начальных этапах накопления данных, статистические подходы становятся практически неэффективны для анализа и прогнозирования данных.

Поэтому в последние годы начали появляться узкоспециализированные пакеты для интеллектуального анализа данных. Они ориентированы на определенный узкий круг практических задач. В алгоритмической основе таких систем лежат какие-либо альтернативные модели, использующие нейронные сети, деревья решений, нечеткую логику, ограниченный перебор, и т.п. Ограничение узкоспециализированных пакетов заключается в следующем: во-первых, заложенные в них подходы не являются универсальными относительно различных размерностей задач, структурированности и противоречивости данных, величины шума; во-вторых, созданные и подстроенные под решение конкретных задач, такие системы могут оказаться совершенно неприменимыми для решения других задач; в-третьих, большое количество задач, интересных практическому пользователю, оказываются шире возможностей одного отдельного подхода. Для примера, пользователю может быть важно иметь численную характеристику надежности некоторого прогноза, но деревья решений не применимы для ее вычисления. Логические методы распознавания используются для выявления логических закономерностей в данных, или для прогнозирования, но в случае линейных зависимостей между признаками и прогнозируемой величиной точность прогноза, сделанного подобной «линейной машиной», может быть заметно выше.

Таким образом, при создании программного обеспечения для решения задач с использованием Data Mining необходимо использовать разнообразные подходы, что позволяет повысить шансы подбора необходимого из имеющихся алгоритмов для обеспечения наиболее точного решения интересующих пользователя задач на новых данных. Также важным свойством систем анализа данных должно быть наличие средств автоматического решения задач не только одним, но и несколькими алгоритмами одновременно (коллективы алгоритмов).

Теоретическими основами для практической реализации решения задач анализа данных коллективами алгоритмов являлись разработки в ВЦ РАН в рамках алгебраического подхода для решения задач распознавания (логическая и алгебраическая коррекция алгоритмов) в 1976-1980 и комитетного синтеза классификаций для задач кластерного анализа (автоматической классификации) в 1981-1982 годах.

В 1980-х годах агентство Gartner Group, занимающееся анализом рынков информационных технологий, ввело термин "Business Intelligence" (BI), бизнес-интеллект [36]. Этот термин предложен для описания различных концепций и методов, которые улучшают бизнес решения путем использования систем поддержки принятия решений. В 1996 году данный термин был

уточнен: Business Intelligence – программные средства, функционирующие в рамках предприятия и обеспечивающие функции доступа и анализа информации, которая находится в хранилище данных, а также обеспечивающие принятие правильных и обоснованных управленческих решений.

BI-системы также пересекаются с понятием систем поддержки принятия решений (СППР, DSS, Decision Support System). То есть данные термины до определенной степени являются взаимозаменяемыми, эти системы обоих классов превращают данные в информацию, на основе которой можно принимать решения.

Gartner Group определяет состав рынка систем Business Intelligence как набор программных продуктов следующих классов:

- средства построения хранилищ данных (data warehousing);
- системы оперативной аналитической обработки (OLAP);
- информационно-аналитические системы (Enterprise Information Systems, EIS);
- средства интеллектуального анализа данных (data mining);
- инструменты для выполнения запросов и построения отчетов (query and reporting tools).

Традиционные методы анализа данных (статистические методы) и OLAP в основном ориентированы на проверку заранее сформулированных гипотез и на первоначальный разведочный анализ, составляющий основу оперативной аналитической обработки данных (OLAP), в то время как одно из основных положений Data Mining – поиск неочевидных, нетривиальных, «скрытых» закономерностей. Инструменты Data Mining способны находить подобные закономерности самостоятельно и строить гипотезы о взаимосвязях между исследуемыми параметрами.

При этом методы Data Mining оперирует реальными значениями в отличие от большинства статистических методов, которые для выявления взаимосвязей в данных используют концепцию средних по выборке значений, приводящую к операциям над несуществующими величинами.

При рассмотрении Data Mining в краткосрочной перспективе, очевидно, что развитие этой технологии направлено в основном к бизнес-областям, в которых необходимы решения, основанные на знаниях, в которых имеется большое количество доступных и значимых данных в ситуации изменчивости окружающей среды, и принятие правильных решений в которых приносит высокие дивиденды.

Финансовые рынки являются как раз такой областью, обладающей всеми приведенными выше особенностями. Поэтому применение инструментов Data Mining для анализа финансовых рынков является перспективной областью исследований.

Совершенствование методологии анализа данных, извлечения знаний из «сырых» данных детерминировано все ускоряющимся развитием информационных и компьютерных технологий, и увеличения в связи с этим объемов информации, которую требуется хранить и обрабатывать. Достаточно долго дисциплина Data Mining не признавалась полноценной самостоятельной областью анализа данных, иногда ее относят к одному из применений статистики. Но технология Data Mining постоянно развивается, привлекает к себе все больший интерес, как со стороны научного мира, так и со стороны применения достижений технологии в бизнесе.

1.4.3. Современное состояние методов интеллектуального анализа данных и средств искусственного интеллекта

С развитием и совершенствованием методов анализа данных, научное сообщество совершало неоднократные попытки создания искусственного интеллекта и с каждым шагом все ближе подходит к своей цели. Замена человека-специалиста на системы искусственного интеллекта, в частности на экспертные системы, системы поддержки принятия решений, позволит существенно ускорить и удешевить сервисные и производственные процессы. Системы искусственного интеллекта всегда объективны и результаты их работы не зависят от одномоментного настроения и ряда других субъективных факторов, присущих человеку.

Искусственный интеллект – это:

- научное направление, в рамках которого ставятся и решаются задачи аппаратного или программного моделирования тех видов человеческой деятельности, которые традиционно считаются интеллектуальными.
- свойство интеллектуальных систем выполнять функции, например, творческие, которые традиционно считаются прерогативой человека [38].

Одними из быстро и активно развивающихся направлений искусственного интеллекта являются средства интеллектуального анализа данных (ИАД, Data Mining).

Первым направлением развития средств Data Mining выделяют методы статистической обработки данных:

- предварительный анализ природы статистических данных (проверка гипотез стационарности, нормальности, независимости, оценка вида функции распределения и ее параметров);
- выявление связей и закономерностей (регрессионный, корреляционный анализ);
- многомерный статистический анализ (дискриминантный, факторный анализ, кластер-анализ);
- динамические модели и прогнозирование временных рядов.

Вторым направлением являются интеллектуальные методы оптимизации, основанные на принципах саморазвивающихся систем – методы нейронных сетей, эволюционного и генетического программирования.

Однако развитие новых технологий необратимо вызывает столкновения с новыми трудностями и проблемами. Например, решения, полученные интеллектуальными методами анализа данных, бывают сложно визуализируемы, что может усложнить интерпретацию результатов экспертам.

К третьему направлению ИАД относят традиционные методы решения оптимизационных задач: методы исследования операций, включающие в себя различные виды математического программирования, динамическое программирование, принцип максимума Понтрягина, методы теории систем массового обслуживания. Стандартные пакеты прикладных программ, например, Math CAD и MatLab включают в себя реализацию большинства из этих методов.

В четвертое направление ИАД входят средства, которые можно условно назвать экспертными, то есть такие средства неразрывно связаны с использованием опыта эксперта. К подобным методам можно отнести метод «ближайшего соседа». Альтернативным подходом к выбору решения является построение последовательных логических выводов. Примером данного подхода является метод дерева решений, в каждом узле которого эксперт осуществляет простой логический выбор. В зависимости от принятого решения, поиск продвигается по одной из ветвей дерева и в конце концов приходит к конечной ветви, отвечающей конкретному окончательному решению по поставленной задаче.

Классификация является довольно простой, но часто решаемой с помощью методов Data Mining задачей. Под классификацией понимается системное распределение изучаемых предметов, явлений, процессов по родам, видам, типам, по каким-либо существенным, важным признакам для удобства их исследования; группировка исходных понятий и величин, расположение их в определенном порядке, отражающем степень этого сходства.

Классификация является закономерностью, позволяющей делать вывод относительно определения характеристик конкретной группы. Таким образом, для проведения классификации должны присутствовать признаки, характеризующие группу, к которой принадлежит тот или иной исследуемый объект.

Классификация относится к методам обучения с учителем (supervised learning), то есть контролируемому или управляемому обучению, что означает необходимость подтверждения отнесения объектов к группе некоторым внешним учителем.

Задачей классификации часто называют предсказание зависимой переменной на основе выборки непрерывных или категориальных переменных, что может быть отнесено к задаче классификации рыночных ситуаций (в понимании текущего состояния системы рынка финансового инструмента).

Классификация может быть одномерной (по одному признаку) и многомерной (по двум или нескольким признакам). Целью процесса классификации является построение модели, использующей независимых переменных в качестве входных параметров и получает значение зависимой величины. Таким образом сам процесс классификации можно представить, как последовательное разбиение множества объектов на классы по определенному критерию. Классификатором в данном случае называется сущность, определяющая, к какому из predetermined классов принадлежит объект по вектору признаков.

Задача анализа финансовых рынков также может быть сведена к классификации рыночных ситуаций. Под классификацией рыночных ситуаций понимается описание набора данных (котировок, технических индикаторов, фундаментальных показателей) на рынке финансового инструмента, предрасполагающих к открытию сделки в одном из направлений в конкретный момент времени, например, ситуаций, при которых покупка или продажа финансового инструмента является выгодной, или ситуация неопределенности, когда по поведению рынка нельзя утверждать, в каком направлении далее последует динамика изменений котировок финансового инструмента.

Для проведения классификации с помощью математических методов необходимо иметь формальное описание объекта, которым можно оперировать, используя математический аппарат классификации. Таким описанием в нашем случае выступает набор информации о системе.

Набор исходных данных (или выборку данных) разбивают на два множества: обучающее и тестовое. Обучающее множество (training set) – множество, включающее данные, использующиеся для обучения (построения) модели. Такое множество содержит входные и выходные значения примеров. Выходные значения предназначены для обучения модели. Тестовое (test set) множество также содержит входные и выходные значения примеров. Здесь выходные значения используются для проверки работоспособности модели.

Процесс классификации состоит из двух этапов: построения модели и ее использования.

1. Построение модели: описание множества predetermined классов.

- каждый пример из набора входных данных относится к определенному классу (выходные данные);
- входные данные берутся из обучающего множества, выходные данные в него записываются;
- полученная модель представляется в виде классификационных правил, дерева решений или математической формулы.

2. Использование модели: классификация новых или неизвестных значений.

- новые значения не должны ходить в обучающее множество;
- тестовое множество не должно повторять или пересекаться с обучающим множеством;



– производится оценка точности модели:

а) известные выходные значения из тестового примера сравниваются с результатами использования полученной модели для каждого из набора входных данных;

б) уровень точности оценивается как процент правильно классифицированных примеров в тестовом множестве;

– в случае получения допустимой точности модели, возможно использование модели для классификации новых примеров, класс которых неизвестен.

Для оценки точности классификации может использоваться кросс-проверка. Кросс-проверка – процедура оценки точности классификации на данных из тестового множества. Точность классификации тестового множества сравнивается с точностью классификации обучающего множества. При получении приблизительно равных результатов по точности классификация тестового и обучающего множества, делается вывод о прохождении моделью кросс-проверки.

При оценке методов классификации следует учитывать следующие характеристики:

1. Скорость – характеризует время, которое требуется на построение модели и ее использование.
2. Робастность – устойчивость к нарушениям исходных предпосылок, то есть оценивается возможность работы с зашумленными данными и пропусками в значениях в исходных данных.
3. Интерпретируемость – характеризует возможность понимания построенной модели аналитиком, в том числе возможности наглядного представления полученных результатов.
4. Надежность – предусматривает примерно одинаковые результаты по модели на относительно чистых данных и данных с шумами и выбросами [36].

Для решения задач классификации используются различные методы. Основные из них:

- классификация с помощью деревьев решений;
- байесовская (наивная) классификация;
- классификация при помощи искусственных нейронных сетей;
- классификация методом опорных векторов;
- статистические методы, в частности, линейная регрессия;
- классификация при помощи метода ближайшего соседа;
- классификация при помощи генетических алгоритмов.

Рассмотрим более подробно некоторые распространенные и наиболее актуальные методы классификации.

### 1.4.3.1 Деревья решений

Деревья решений – один из методов автоматического анализа данных. Первые идеи создания деревьев решений восходят к работам Ховленда и Ханта в конце 50-х годов XX века. Работой, которая явилась важной в развитии данного направления, явилась книга Hunt, Marin J., Stone, P.J "Experiments in Induction", 1966 г.

Деревья решений – это способ представления правил в иерархической, последовательной структуре, где каждому объекту соответствует единственный элемент (узел), предоставляющий решение.

Под правилом понимается логическая конструкция, представленная в виде: "если ..., то ...". Пример такой структуры представлен на рисунке 1.6.

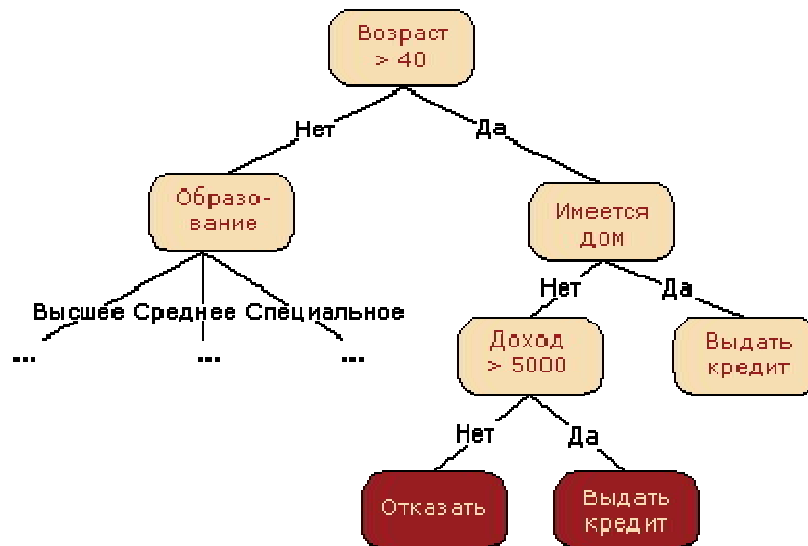


Рисунок 1.6 – Пример дерева решений

Пусть задано некоторое обучающее множество  $T$ , содержащее объекты (примеры), каждый из которых характеризуется  $m$  атрибутами (характеристиками), причем один из них указывает на принадлежность объекта к определенному классу.

Идею построения деревьев решений из множества  $T$ , приведем в интерпритации Р. Куинлена (R. Quinlan).

Пусть через  $\{C_1, C_2, \dots, C_k\}$  обозначены классы (значения метки класса), тогда существуют 3 ситуации:

1. множество  $T$  содержит один или более примеров, относящихся к одному классу  $C_k$ . Тогда дерево решений для  $T$  – это лист, определяющий класс  $C_k$ ;
2. множество  $T$  не содержит ни одного примера, т.е. пустое множество. Тогда это снова лист, и класс, ассоциированный с листом, выбирается из другого множества отличного от  $T$ , скажем, из множества, ассоциированного с родителем;

3. множество  $T$  содержит примеры, относящиеся к разным классам. В этом случае следует разбить множество  $T$  на некоторые подмножества. Для этого выбирается один из признаков, имеющий два и более отличных друг от друга значений  $O_1, O_2, \dots, O_n$ .  $T$  разбивается на подмножества  $T_1, T_2, \dots, T_n$ , где каждое подмножество  $T_i$  содержит все примеры, имеющие значение  $O_i$  для выбранного признака. Это процедура будет рекурсивно продолжаться до тех пор, пока конечное множество не будет состоять из примеров, относящихся к одному и тому же классу.

На сегодняшний день существует значительное число алгоритмов, реализующих деревья решений CART, C4.5, NewId, ITrule, CHAID, CN2 и т.д. Но наибольшее распространение и популярность получили следующие два:

- CART (Classification and Regression Tree) – это алгоритм построения бинарного дерева решений – дихотомической классификационной модели. Каждый узел дерева при разбиении имеет только двух потомков. Алгоритм решает задачи классификации и регрессии.
- C4.5 – алгоритм построения дерева решений, количество потомков у узла при этом не имеет ограничений. Не умеет работать с непрерывным целевым полем, поэтому решает только задачи классификации.

При построении деревьев решений особое внимание уделяется следующим вопросам: выбору критерия атрибута, по которому пойдет разбиение, остановки обучения и отсечения ветвей [28].

Деревья принятия решений предоставляют удобную аналитику визуализацию данных, но в тоже время они обладают рядом недостатков при решении сложных задач, таких как анализ финансового рынка:

- они используют алгоритм «жадности», при котором выбирается решение на каждом узле, что не может обеспечить оптимальность всего дерева целиком;
- при конструировании дерева может возникнуть проблема излишней глубины, что может привести к переобучению и недостоверности предоставляемых деревом результатов;
- при большом количестве критериев теряется свойство репрезентативности дерева;
- при больших деревьях решений возникают проблемы XOR, четности и мультиплексарности;
- присваивание большего веса атрибутам, которые имеют большое количество уровней [27].

Перечисленные недостатки не позволяют использовать деревья решений для анализа ситуаций на финансовых рынках, так как данная задача является многокритериальной и для ее решения необходима обработка большого количества разнородных данных.

### 1.4.3.2 Метод ближайшего соседа

Метод "ближайшего соседа" ("nearest neighbour") относится к классу методов, работа которых основывается на хранении данных в памяти для сравнения с новыми элементами [54]. При появлении новой записи для прогнозирования находятся отклонения между этой записью и подобными наборами данных, и наиболее подобная (или ближний сосед) идентифицируется. Данный метод по своей сути относится к категории «обучение без учителя», то есть является «самообучающейся» технологией, благодаря чему рабочие характеристики каждой базы прецедентов с течением времени и накоплением примеров улучшаются.

Пусть требуется провести классификацию заданного множества объектов методом ближайшего соседа. Расстояние между двумя классами определяется как расстояние между ближайшими их представителями (рассчитывается матрица расстояний между объектами, см. рисунок 1.7).

На каждом шаге в матрице расстояний ищется минимальное значение, соответствующее расстоянию между двумя наиболее близкими кластерами. Найденные кластеры объединяются, образуя новый кластер. Эта процедура повторяется до тех пор, пока не будут объединены все кластеры.

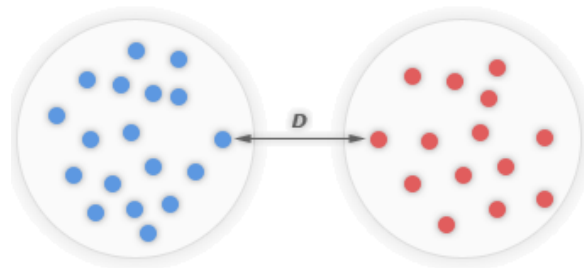


Рисунок 1.7 – Расстояние между классами

Изначально каждый объект представляет собой отдельный кластер: 1, 2, ...,  $n$ . В целом метод ближайшего соседа является простейшим метрическим классификатором, основанным на оценивании сходства объектов. Объект, подлежащий классификации, относится к тому классу, которому принадлежат ближайшие к нему объекты в обучающей выборке.

Простота использования является положительным качеством данного метода.

Суть метода заключается в следующем:

1. Пусть задана обучающая выборка пар «объект-ответ»:

$$X^m = \{(x_1, y_1), \dots, (x_m, y_m)\} \quad (1.10)$$

2. На множестве задана функция расстояния  $\rho(x, x')$ , причем чем больше значение этой функции, тем менее схожи объекты.
3. Произвольный объект  $u$  порождает свою перенумерацию выборки:

$$a(u) = \arg \max_{y \in Y} \sum_{i=1}^m [y(x_{i,u}) = y] \cdot w(i, u) \quad (1.11),$$

где  $w(i, u)$  заданная весовая функция, которая оценивает степень важности  $i$ -го соседа для классификации объекта  $u$ .

4. Для различных модификаций методов ближайшего соседа весовая функция задается различным образом:

$$w(i, u) = [i = 1] \quad (1.12),$$

для простейшего метода ближайшего соседа;

$$w(i, u) = [i \leq k] \quad (1.13),$$

для метода  $k$ -ближайших соседей, где  $k$  – количество соседних объектов, признающиеся ближайшими;

$$w(i, u) = [i \leq k] q^i \quad (1.14),$$

для метода  $k$ -экспоненциально взвешанных ближайших соседей, где  $k$  – количество соседних объектов, признающиеся ближайшими,  $q$  – коэффициент взвешивания, полагается  $q < 1$ .

В алгоритме ближайшего соседа существует проблема выбора количества соседей. При  $k=1$  алгоритм ближайшего соседа неустойчив к шумовым выбросам: он даёт ошибочные классификации не только на самих объектах-выбросах, но и на ближайших к ним объектах других классов. При  $k=m$  алгоритм чрезмерно устойчив и вырождается в константу. На практике оптимальное значение параметра  $k$  определяют по критерию скользящего контроля, чаще всего – методом исключения объектов по одному. Такой метод перебора и исключения является довольно трудоемким и долгим [55].

При использовании метода ближайшего соседа особое внимание следует уделять выбору меры расстояния между объектами. На основе нее формируется начальная матрица расстояний, определяя дальнейший процесс классификации. Зачастую мерой берется обыкновенное евклидово расстояние, что порождает некоторые сомнения в том, действительно ли все признаки вносят одинаковый вклад в общий вес. Данную меру расстояний довольно проблематично адекватно выделить для сложных систем, таких как финансовые рынки. Также при использовании метода на больших объемах данных, встают технические вопросы о необходимости хранения слишком большого количества данных и оптимальном алгоритме поиска среди них  $k$  ближайших соседей, которые могут быть решены прореживание выборки, удалением неинформативных объектов, что, во-первых, требует дополнительных временных и ресурсных затрат со стороны аналитика, во-вторых, может привести к потере данных и, как следствие, уменьшению надежности выполненной классификации.

### 1.4.3.3 Метод опорных векторов

Метод опорных векторов (SVM – support vector machines) – это набор схожих алгоритмов вида «обучение с учителем», использующихся для задач классификации и регрессионного анализа [58]. Этот метод относится к линейным классификаторам.

Основная идея метода опорных векторов заключается в переводе исходных векторов в пространство более высокой размерности и поиск разделяющей гиперплоскости с максимальным зазором в этом пространстве. Две параллельных гиперплоскости строятся по обеим сторонам гиперплоскости, разделяющей классы. Разделяющей гиперплоскостью будет гиперплоскость, максимизирующая расстояние до двух параллельных гиперплоскостей. Чем больше расстояние между этими параллельными гиперплоскостями, тем меньше будет каждый объект данных представлен как вектор (точка) в пространстве (последовательность чисел). Исследуется возможность разделения точек гиперплоскостью размерностью  $p-1$ . Это типичный случай линейной разделимости. Гиперплоскостей может быть довольно много, но необходимо найти такую гиперплоскость, чтобы расстояние от нее до ближайшей точки было максимальным. Расстояние между двумя ближайшими точками, располагающимися по разные стороны гиперплоскости, является максимальным. Существование такой гиперплоскости вызывает наибольший интерес, ее называют оптимальная разделяющая гиперплоскость, а соответствующий ей линейный классификатор – оптимальный разделяющий классификатор.

Пусть точки имеют вид:  $(x_1, c_1), (x_2, c_2), \dots, (x_n, c_n)$ , где  $c_i$  принимает значение 1 или  $-1$ , в зависимости от того, к какому классу принадлежит точка  $x_i$ . Каждое  $x_i$  – это  $p$ -мерный вещественный вектор, обычно нормализованный значениями  $[0,1]$  или  $[-1,1]$ . Если точки не будут нормализованы, то точка с большими отклонениями от средних значений координат точек слишком сильно повлияет на классификатор. Данное множество рассматривается, как учебная коллекция, в которой для каждого элемента уже задан класс, к которому он принадлежит. От метода опорных векторов требуется классификация аналогичным образом. Для этого строится разделяющая гиперплоскость (рисунок 1.8).

Ближайшие к параллельным гиперплоскостям точки называются опорными векторами.

Оптимальная разделяющая гиперплоскость для метода опорных векторов, построенная на точках из двух классов. Ближайшие к параллельным гиперплоскостям точки называются опорными векторами.

$$w \cdot x - b = 0 \quad (1.15),$$

где вектор  $w$  – перпендикулярен к разделяющей гиперплоскости.

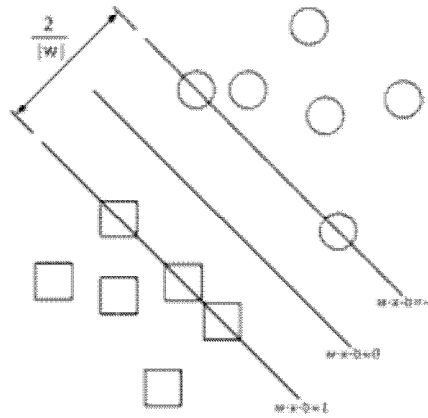


Рисунок 1.8 – Оптимальная разделяющая гиперплоскость

Параметр  $b$  зависит от кратчайшего расстояния гиперплоскости до начала координат. Если параметр  $b$  равен нулю, гиперплоскость проходит через начало координат, что ограничивает решение. При поиске оптимального разделения рассмотрим опорные вектора и гиперплоскости параллельные оптимальной и ближайšie к опорным векторам двух классов. Можно показать, что эти параллельные гиперплоскости могут быть описаны следующими уравнениям.

$$w \cdot x - b = 1, \quad w \cdot x - b = -1 \quad (1.16).$$

Если учебное множество является линейно разделимым, то гиперплоскости можно выбирать так, чтобы между ними не лежала ни одна точка обучающей выборки с максимизацией расстояния между гиперплоскостями. Ширина полосы между ними может быть представлена как  $2/\|w\|$ . Для исключения всех точек из полосы, необходимо проверить для всех  $i$ :

$$\begin{cases} w \cdot x_i - b \geq 1, c_i = 1 \\ w \cdot x_i - b \leq -1, c_i = -1 \end{cases} \quad (1.17).$$

Проблема построения оптимальной разделяющей гиперплоскости сводится к минимизации  $\|w\|$ , при условии (1.16). Это задача квадратичной оптимизации, методы решения которых проработаны. Среди них можно выделить, последовательное квадратичное программирование, метод Ньютона, метод сопряженных направлений.

Основной идеей метода опорных векторов является перевод исходных векторов в пространство более высокой размерности, что накладывает некоторые ограничения на максимально допустимую размерность вектора исходных данных. Данная идея берет свое начало из предположений, что не все исследуемые множества данных являются линейно разделимыми, что требуется для применения метода разделения гиперплоскостями. При изначально высокой размерности исходного вектора перевод его в еще более высокоразмерное пространство ведет к

усложнению алгоритмов анализа данной системы и требует больших вычислительных мощностей.

#### 1.4.3.4 Нейронные сети

Искусственные нейронные сети (ИНС) – математические модели, и их программные или аппаратные реализации, построенные по принципу организации и функционирования биологических нейронных сетей (сетей нервных клеток живого организма, в том числе человека).

ИНС представляют собой систему соединённых и взаимодействующих между собой простых процессоров (искусственных нейронов). Такие процессоры обычно довольно просты, особенно в сравнении с процессорами, используемыми в персональных компьютерах. Каждый процессор подобной сети имеет дело только с сигналами, которые он периодически получает, и сигналами, которые он периодически посылает другим процессорам.

С точки зрения машинного обучения, нейронная сеть представляет собой частный случай методов распознавания образов, дискриминантного анализа, методов кластеризации и т. п. С математической точки зрения, обучение нейронных сетей – это многопараметрическая задача нелинейной оптимизации. С точки зрения кибернетики, нейронная сеть используется в задачах адаптивного управления. А с точки зрения искусственного интеллекта, ИНС является основой философского течения коннективизма и основным направлением в структурном подходе по изучению возможности построения (моделирования) естественного интеллекта с помощью компьютерных алгоритмов.

Нейронные сети не программируются в привычном смысле этого слова, они подвергаются обучению. Возможность обучения является одним из существенных преимуществ нейронных сетей перед традиционными алгоритмами. Технически обучение заключается в нахождении коэффициентов связей (весов) между нейронами. В процессе обучения нейронная сеть способна выявлять сложные зависимости между входными данными и выходными, а также выполнять различного рода обобщения. При успешном обучении нейросеть может вернуть верный результат на основании данных, которые отсутствовали в обучающей выборке, а также «зашумленных», неполных или частично искаженных данных.

В процессе обучения сеть в определенно заданном порядке просматривает обучающую выборку. Порядок просмотра может быть последовательным, случайным и т. д. Некоторые сети, обучающиеся без учителя, например, сети Хопфилда просматривают выборку только один раз. Другие, например, сети Кохонена, а также сети, обучающиеся с учителем, просматривают выборку множество раз, при этом один полный проход по выборке называется эпохой обучения. При обучении с учителем набор исходных данных делят на две части: обучающая и тестовая.



вая выборки, принцип разделения на которые может быть произвольным, но тестовое множество не может превышать по количеству примеров обучающее. Обучающие данные подаются сети для обучения, а проверочные используются для расчета ошибки сети. Таким образом, если на проверочных данных ошибка уменьшается, то сеть действительно выполняет обобщение. Если ошибка на обучающих данных продолжает уменьшаться, а ошибка на тестовых данных увеличивается, значит, сеть перестала выполнять обобщение и просто «запоминает» обучающие данные. Это явление называется переобучением сети или оверфиттингом. В таких случаях обучение обычно прекращают. В процессе обучения могут проявиться другие проблемы, такие как паралич или попадание сети в локальный минимум поверхности ошибок. Невозможно заранее предсказать проявление той или иной проблемы, равно как и дать однозначные рекомендации к их разрешению.

Классификация нейронными сетями по характеру обучения:

- обучение с учителем – выходное обучающее пространство решений нейронной сети известно;
- обучение без учителя – нейронная сеть формирует выходное пространство решений только на основе входных воздействий. Такие сети называют самоорганизующимися;
- обучение с подкреплением – система назначения штрафов и поощрений от среды.

Обучить нейронную сеть – значит, сообщить ей исходные данные и результаты анализа, по которым ей необходимо самой выстроить необходимые взаимосвязи для аналогичного анализа на подобных данных (см. рисунок 1.9).

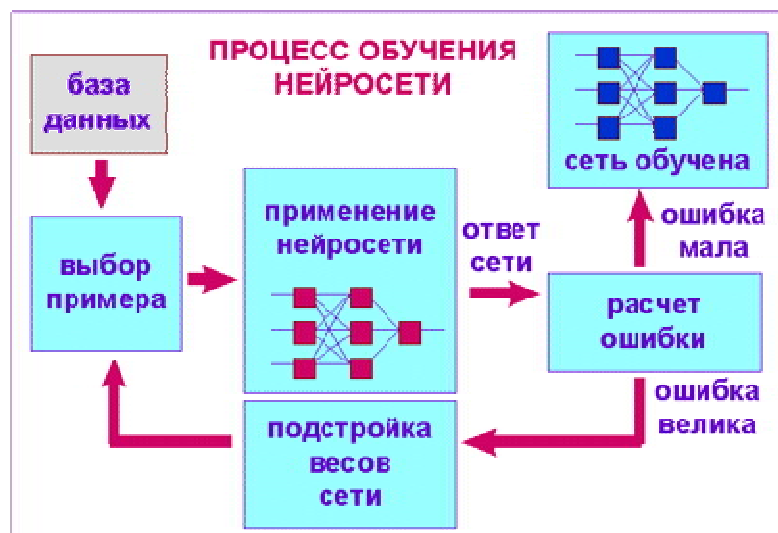


Рисунок 1.9 – Концептуальная модель процесса обучения нейросети

Оказывается, что после многократного предъявления примеров веса нейронной сети стабилизируются, причем нейронная сеть дает правильные ответы на все (или почти все) примеры из базы данных [66]. В таком случае говорят, что нейронная сеть выучила все примеры,

нейронная сеть обучена или натренирована. В программных реализациях можно видеть, что в процессе обучения величина ошибки (сумма квадратов ошибок по всем выходам) постепенно уменьшается. Когда величина ошибки достигает нуля или приемлемого малого уровня, тренировку останавливают, а полученную нейронную сеть считают натренированной и готовой к применению на новых данных, нуждающихся в анализе.

Нейронная сеть представляет собой сеть с конечным числом слоев, состоящих из отдельных элементов, подобных нейронам с различными типами связей между слоями. Количество нейронов в слоях выбирается таким образом, чтобы быть достаточным для предоставления необходимого качества решения задачи. Желаемое количество слоев такое, чтобы быть минимальным при уменьшении времени решения задачи. [115, с. 23]

Искусственный нейрон – это устройство (элемент нейронной сети) с большим количеством входов и одним выходом. Простой нейрон имеет два режима работы: обучение и использование. При обучении нейрон может обучаться срабатывать (или нет) для определенных входных наборов. В режиме использования, когда предоставленный входной набор обнаружен при вводе, его связанный вывод становится текущим выводом. Если входной набор не принадлежит предоставленному списку входных наборов, используется правило отказа, чтобы определить, срабатывать или нет (рисунок 1.10).

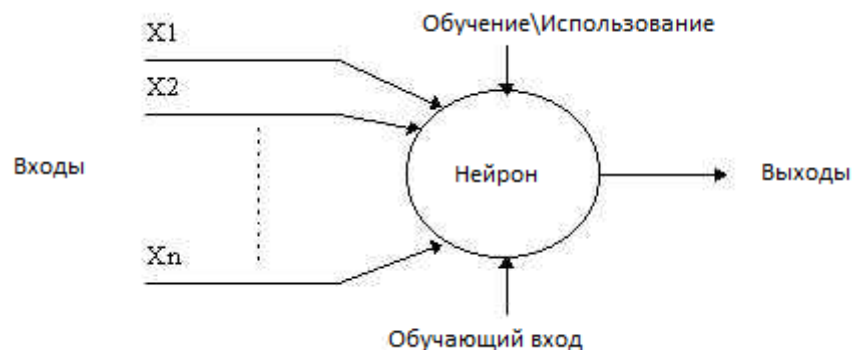


Рисунок 1.10 – Простой нейрон

Более сложный нейрон (рисунок 1.11) был разработан в 1943 нейрофизиологом Уорреном Маккаллоком и логиком Уолтером Питсом, называется моделью Маккаллока и Питтса (МКП) [123]. Отличием от предыдущей модели является то обстоятельство, что входы являются «взвешенными», то есть тот эффект, который каждый вход имеет при принятии решений, зависит каждого от веса определенного входа. Вес входа представляет число, которое при умножении на вход дает взвешенный вход.

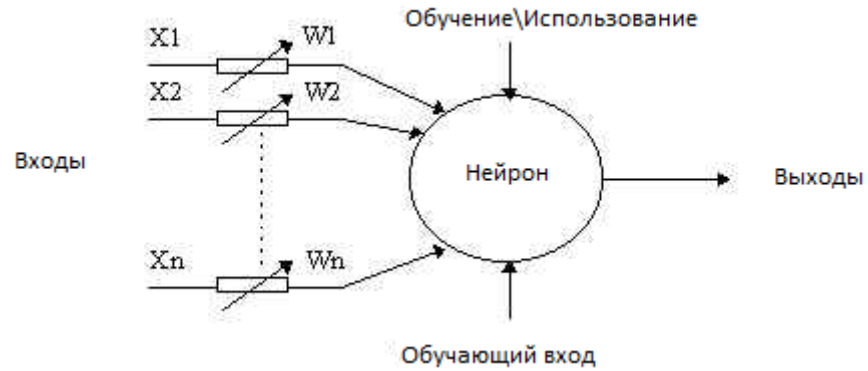


Рисунок 1.11 – Нейрон МКП

В математических терминах, нейрон срабатывает тогда и только тогда, когда:

$$x_1 \cdot w_1 + x_2 \cdot w_2 + \dots + x_n \cdot w_n > T \quad (1.18)$$

Добавление входных весов и порога делает этот нейрон очень гибким и мощным. У нейрона МКП есть возможность адаптироваться к определенной ситуации, изменяя свои веса и/или порог. Существуют различные алгоритмы, которые позволяют нейрону «адаптироваться»; наиболее используемые – правило Дельты и обратного распространения ошибки.

В еще более сложных нейронах взвешенная сумма не просто сравнивается с пороговым значением ( $T$ ), а она преобразуется в выходное значение с помощью особой функции, называемой функцией активации.

Функция активации (активационная функция) – функция, с помощью которой вычисляется выходной сигнал в искусственном нейроне. В качестве аргумента функция принимает сигнал  $Y$ , получаемый на выходе входного сумматора. Ниже приведены наиболее часто используемые функции активации.

#### 1. Единичный скачок или жесткая пороговая функция

Простая кусочно-линейная функция (рисунок 1.12). Если входное значение меньше порогового, то значение функции активации равно нижней границе, иначе – верхней границе.

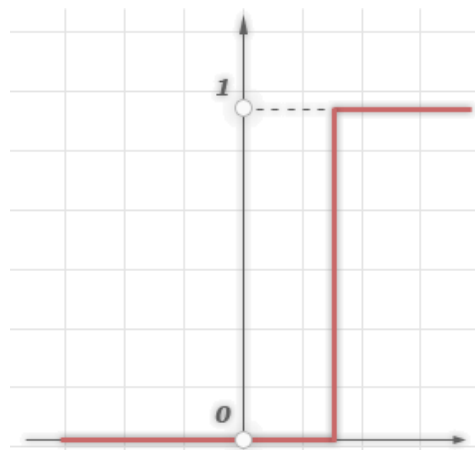


Рисунок 1.12 – Жесткая пороговая функция

## 2. Линейный порог (гистерезис)

Более сложная кусочно-линейная функция (рисунок 1.13). Состоит из 2 участков, на которых функция активации тождественно равна минимально допустимому и максимально допустимому значению, и также имеет участок, где функция строго монотонно возрастает.

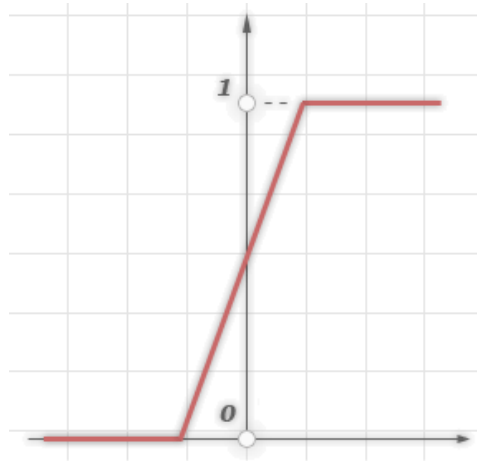


Рисунок 1.13 – Гистерезис

## 3. Сигмоидальная функция или сигмоид

Монотонно возрастающая всюду дифференцируемая S-образная нелинейная функция с насыщением (рисунок 1.14). С помощью сигмоида можно усиливать слабые сигналы и избегать насыщения от сравнительно сильных сигналов. Также в 1973 году Гроссберг обнаружил, что применение данной функции активации позволяет решить дилемму шумового насыщения.

Слабые сигналы необходимо усиливать для придания подходящего для использования выходного сигнала. Однако при использовании больших коэффициентов усиления может возникнуть ситуация насыщения выхода шумами усилителей, которые присутствуют в любой физически реализованной сети. Сильные входные сигналы в свою очередь также будут приводить к насыщению усилительных каскадов, минимизируя возможность использования выхода. Для обработки как сильных, так и любых сигналов применяются сигмоидальные функции активации.

Одной из сигмоидальных функций активации является логистическая функция, задаваемая следующим выражением:

$$OUT = \frac{1}{1 + e^{-\alpha Y}} \quad (1.19)$$

где  $\alpha$  – параметр наклона сигмоидальной функции. При изменении данного параметра получаются функции с различной крутизной.

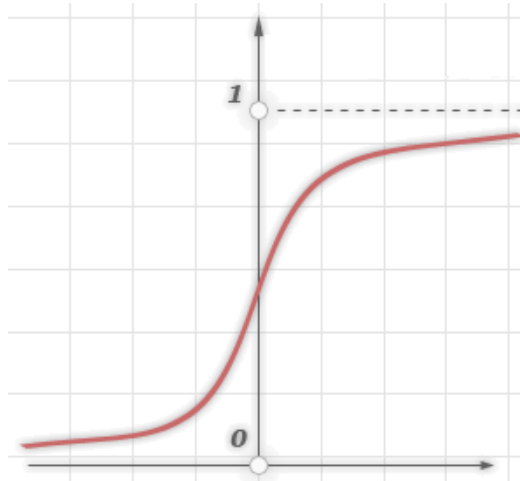


Рисунок 1.14 – Сигмоид

Еще одним примером сигмоидальной функции активации может служить гиперболический тангенс (рисунок 1.15):

$$OUT = th\left(\frac{Y}{\alpha}\right) \quad (1.20),$$

где  $\alpha$  – параметр, влияющий на наклон функции [104].

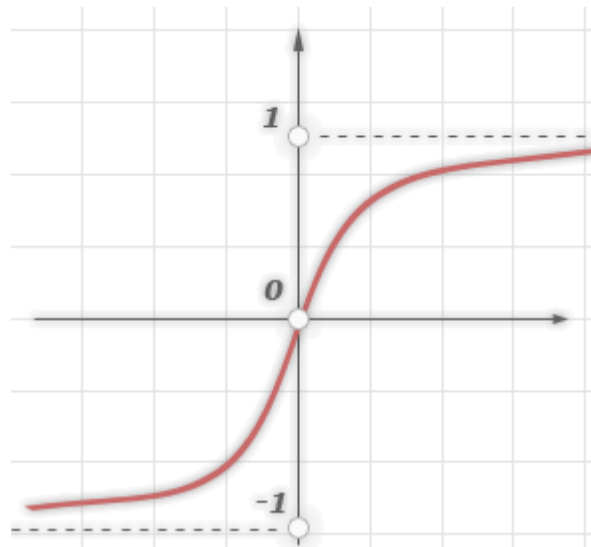


Рисунок 1.15 – Гиперболический тангенс

Более подробно о функциях активации можно прочитать в книгах Бене Кросе, Патрика ван дер Смэгта [123, с.15-16], Александра Галушкина [115, с.35-40].

Таким образом математически нейрон представляет собой взвешенный сумматор (см. рисунок 1.16).

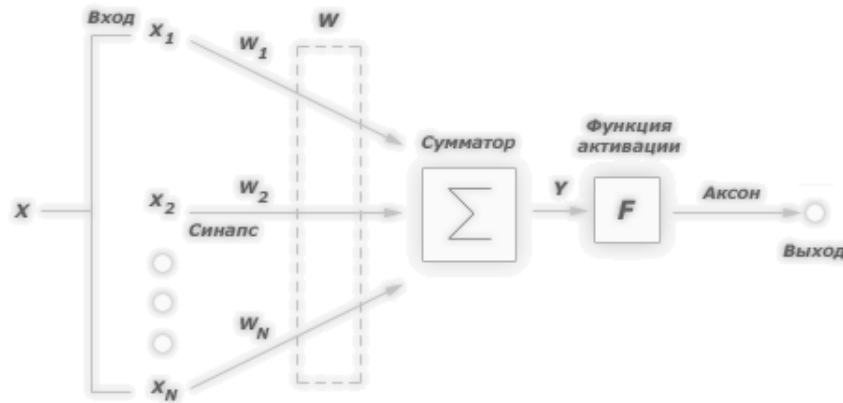


Рисунок 1.16 – Модель искусственного нейрона

Выход данного сумматора определяется через его входы и матрицу весов:

$$y=f(u), \text{ где } u = w_0x_0 + \sum_{i=1}^n w_i x_i \quad (1.21).$$

Здесь  $x$  и  $w_i$  – соответственно сигналы на входах нейрона и веса входов, функция  $u$  называется индуцированным локальным полем, а  $f(u)$  – передаточной функцией. Возможные значения сигналов на входах нейрона считают заданными в интервале  $[0,1]$ . Дополнительный вход  $x_0$  и соответствующий ему вес  $w_0$  используются для инициализации нейрона. Под инициализацией подразумевается смещение активационной функции нейрона по горизонтальной оси, то есть формирование порога чувствительности нейрона [112].

Передаточная функция  $f(u)$  определяет зависимость сигнала на выходе нейрона от взвешенной суммы сигналов на его входах. В большинстве случаев она является монотонно возрастающей и имеет область значений  $[-1,1]$  или  $[0,1]$ .

Выделяют различные топологии нейронных сетей для решения задач классификации.

Сети с прямой связью являются универсальным средством аппроксимации функций, что позволяет их использовать в решении задач классификации. При этом еще Минским было доказано, что при использовании однослойного перцептрона, возможно решить только линейно разделимые задачи, поэтому для более сложных пространств признаков рекомендуется использовать многослойные перцептроны, позволяющие выявлять не только сами характеристики исследуемых объектов, но и характеристики характеристик.

Оригинальная модель «Перцептрон» была разработана Франком Розенблатом в 1958 году [129]. Модель Розенבלата состояла из трех слоев. В 1969 году Мавином Минским и Сэймором Пейпером [124] был опубликован критический анализ перцептрона, указывая много значительных недостатков перцептрона, и для того времени интерес к перцептрону пропал.

Интерес к нейронным сетям был возрожден в 1986 году, когда Дэвид Румельхарт, Геоффри Хинтон и Рональд Уильямс публикуют «Обучение внутренних представлений методом обрат-

ного распространения ошибки» [130]. Они предложили многослойный перцептрон с нелинейной, но дифференцируемой функцией отклика. Многослойный перцептрон на рисунке 1.17 имеет входной слой (слева) с тремя нейронами, один скрытый слой (в середине) с тремя нейронами и один выходной слой с тремя нейронами.

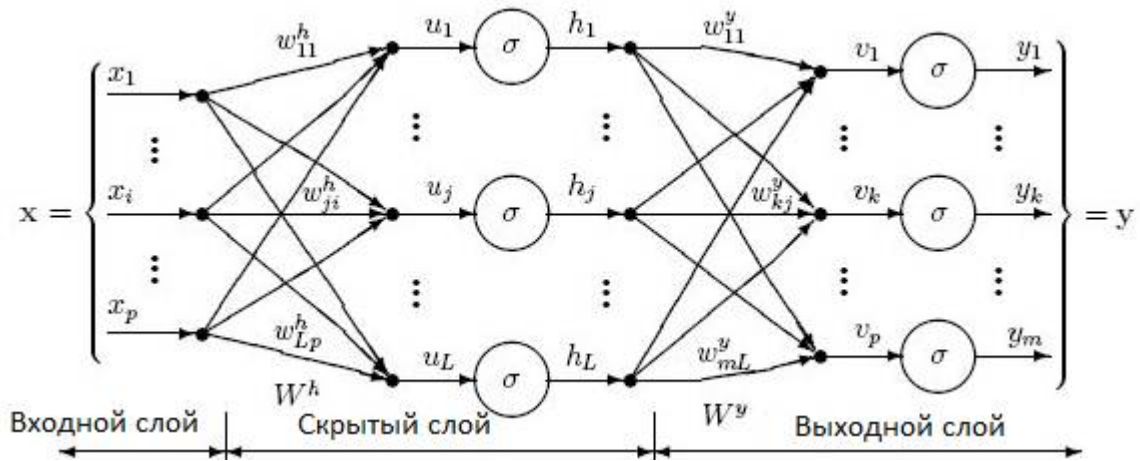


Рисунок 1.17 – Многослойный перцептрон

Входной слой – вектор величин предикатных переменных ( $x_1 \dots x_p$ ) предоставляется на входной слой. Входной слой (или обработка до входного слоя) стандартизует эти переменные, так что каждая переменная была ранжирована в пределах от -1 до 1.

Скрытый (промежуточный) слой – поступающая на скрытый слой величина от каждого входного нейрона умножается на вес ( $w_{ji}$ ), и взвешенные величины суммируются, образуя величину  $u_j$ . Взвешенная сумма  $u_j$  поступает на передаточную функцию  $\sigma$ , на выходе которой получается величина  $h_j$ .

Выходной слой – поступающая на нейрон выходного слоя от каждого нейрона скрытого слоя величина умножается на вес ( $w_{kj}$ ), и полученные в результате взвешенные величины суммируются и образуют общую величину  $v_j$ . Взвешенная сумма  $v_j$  поступает на передаточную функцию  $\sigma$ , на выходе которой величина  $u_k$ . Величина  $u$  является выходом сети [125].

Цель процесса обучения найти такой набор значений весов, при которых выходы нейронной сети будут как можно ближе к действительным целевым значениям. Геометрическую интерпретацию обучения перцептрона показана в лекциях Генивава Орра, Ники Шраудорфа и Фреда Кумминса [116]. Многослойные перцептроны успешно применяются для решения разнообразных сложных задач и имеют следующих три отличительных признака.

Свойство 1. Каждый нейрон сети имеет нелинейную функцию активации

Важно подчеркнуть, что такая нелинейная функция должна быть гладкой (т.е. всюду дифференцируемой). Наличие нелинейности играет важную роль, так как в противном случае отображение «вход-выход» сети можно свести к обычному однослойному перцептрон.

### Свойство 2. Несколько скрытых слоев

Многослойный перцептрон содержит один или несколько слоев скрытых нейронов, не являющихся частью входа или выхода сети. Эти нейроны позволяют сети обучаться решению сложных задач, последовательно извлекая наиболее важные признаки из входного образа.

### Свойство 3. Высокая связность

Многослойный перцептрон обладает высокой степенью связности, реализуемой посредством синаптических соединений.

Многослойные перцептроны, использующие алгоритм обратного распространения, являются стандартным алгоритмом для любого обучаемого с учителем процесса распознавания образов [125]. В настоящее время многослойные перцептроны обычно применяются в распознавании речи, распознавании изображения, и в программном обеспечении машинного перевода. Так же они могут применяться для задач классификации любой сложности.

Для решения специфических типов задач классификации, таких как распознавание образов при различном искажении исходных данных (например, при непривычном расположенных изображений, текстов) может использоваться сверточная нейронная сеть, предложенная Яном Лекуном [135]. Особенностью данной архитектуры является наличие операции свертки, которая заключается в том, что каждый фрагмент образа (изображения, текста) умножается поэлементно на матрицу свертки, результат суммируется и записывается поэлементно в позицию образа. Сверточная сеть может состоять из нескольких различных типов слоев: сверточные, субдескрипторизирующие, полносвязанные. В сверточном слое нейроны, использующие одни и те же веса, объединяются, и каждый нейрон такого объединения связан с некоторым количеством нейронов предыдущего слоя. Нейроны субскредитирующего слоя уменьшают размерность, что довольно важно при распознавании пиксельных изображений или больших объемов текстов, где размерность измеряется сотнями или больше признаков. Данная модель нейронной сети была разработана на основе исследования зрительной коры кошки, в котором выяснилось, что существуют довольно простые клетки, которые реагируют на прямые линии, и более сложные клетки, которые реагируют уже на однонаправленное движение различных линий. Идея состоит в аналогичном чередовании слоев более простых и отвечающих за более сложное восприятие: сверточных и субскредитирующих [136]. Подобного рода топологии нейронных сетей выгодно использовать при распознавании образов довольно большой размерности (более 100 параметров), так как благодаря своей структуре они имеют возможность ее уменьшения с помощью специальных элементов. Минусом сверточных топологий нейронных сетей является их сложность для понимания и визуализации процесса классификации, значительные трудозатраты при



реализации, увеличение времени на обучение и работу таких сетей по сравнению с сетями более простой топологии и меньшим количеством слоев.

Нейронные сети адаптивного резонанса, основанные на теории, разработанной Стивеном Гроссбергом и Гейлом Карпентером, основной идеей которой является предположение, что распознавание образов является результатом нисходящей информации (припоминание прототипов) и восходящей сенсорной информации, также могут применяться для решения задач распознавания образов [4]. Данный тип нейронных сетей предполагает обучение без учителя с помощью специальных полей сравнения и распознавания, параметров бдительности, благодаря этой особенности четкое разделение на период обучения и использование сети практически невелируется. В данного рода сетях выделяется входной слой, нейроны которого связаны с выходами не одной, а парой синапсов: кратковременной и долговременной памяти. Долговременная память отвечает за необходимость создания еще одного кластера, если все выходы оказались нулевыми при вычислении количественного сходства, величина которого в конечном итоге оказывает значительное влияние на результаты кластеризации по тем или иным признакам (выбор сетью их за существенные или нет) [67]. Также минусом данной топологии является большая степень зависимости от порядка предъявления обучающей выборки, упорядочивание которой каждой раз на практике в динамических изменяющихся системах, таких как финансовые рынки, является затруднительным.

### **Выводы по главе, цель и задачи исследования**

1. Финансовый рынок – сложная, многокритериальная система, порождающая, как следствие своего функционирования, временные ряды изменений котировок акций, фьючерсов, валют. Финансовые рынки являются открытыми социально-экономическими системами, задача управление в которых важна с точки зрения стабилизации инвестиционных процессов в стране. Финансовые рынки являются примерами организационных систем с неопределенностью, то есть неполной информации о системе, доступной для анализа. Данная система характеризуется большим количеством численной информации, такой как цены, объемы торгов, капитализация, значения макроэкономических показателей, а также информации, отличающейся от численного представления, такие как экономические и политические встречи, кризисные или природные явления. Анализ информации о текущем состоянии рынка выбранного финансового инструмента может быть произведен с помощью различных методов. Классическими методами анализа информации о рынках являются технический и фундаментальный анализ, более современными экономико-математическое моделирование, заключающееся в прогнозировании временных рядов котировок финансовых инструментов. Развитие информационных наук привело к появле-

нию нового направления интеллектуального анализа данных, среди которых довольно перспективной является нейросетевая методология, практическое применение которой в настоящее время имеет широкое

2. Основные исследования в области анализа финансовых рынков ведутся в разрезе 3 подходов: фундаментальный, технический и эконометрический анализ. Каждый из подходов предоставляет инструменты для анализа финансового рынка, рассматривая его с разных сторон, как экономически обоснованную систему, как набор закономерностей котировок и как временной ряд, который может быть смоделирован на основе своих предыдущих значений. Все подходы исторически доказали свою обоснованность, и имеют место быть в стратегии анализа финансовых рынков. Но также каждый из них и обладает некоторыми минусами, которые могут быть нивелированы использованием инструментов остальных двух подходов. Современные трейдеры и аналитики финансового рынка предпочитают учитывать в своей работе все вышеперечисленные характеристики данной многофакторной системы, выбирая для анализа того или иного финансового инструмента определенный набор методов каждого из приведенных подходов к анализу финансового рынка.

Таким образом, целью данной работы является разработка нового алгоритма анализа финансовых рынков, использующего все преимущества каждого из подходов к его анализу, позволяющей с наибольшей вероятностью определять закономерности рынка, что будет способствовать достижению трейдерами положительных результатов при торговле.

3. Не все методы подходов к анализу финансовых рынков являются релевантными по отношению к краткосрочному прогнозированию и моделированию финансовых временных рядов. Так, например, стоит отметить, что применение фундаментального анализа является очень спорным в современной действительности на таких изменчивых и динамических рынках, как валютные и фьючерсные рынки. Но в тоже время, следует учитывать, что важные экономические события все же оказывают влияние на котировки, и, если и не могут объяснить или предсказать изменение тенденций, по крайней мере влияют на увеличение волатильности в момент их публикации. При применении индикаторов технического анализа следует выбирать наиболее распространенные, так как это повышает количество сделок, совершаемых по их сигналам, что ведет к увеличению вероятности их надежности. Также следует иметь арсенал из нескольких индикаторов разных групп, например, индикатор тренда и осциллятор. А при построении эконометрических моделей обязательно нужно помнить о нестационарности финансовых временных рядов, и учитывать данную особенность при подборе моделей, отдавая предпочтение моделям, предоставляющим инструменты преобразования нестационарных временных рядов в

стационарные, либо ориентированным на работу с временными рядами, не удовлетворяющими свойствам стационарности второго порядка.

4. В настоящее время проблематика искусственного интеллекта активно обсуждается в научных кругах, развиваются существующие и выделяются все новые подходы и методики обработки, анализа данных, извлечения из них необходимых знаний, прорабатываются в научных исследованиях оптимальные методики анализа. Одним из наиболее распространенных и перспективных методов анализа данных, особенно в области финансовых рынков, является применение искусственных нейронных сетей для выявления зависимостей и установления связей.

Не оставляет сомнения, что методы интеллектуального анализа данных могут быть использованы для решения задачи анализа рыночных ситуаций. Данную задачу можно сформулировать, как нахождение определенных паттернов (классов) по множеству входных данных (цены, индикаторы, показатели), которыми было бы удобно пользоваться аналитикам и трейдерам финансового рынка. Учитывая, что основная проблема при торговле на финансовых рынках, которая встает перед трейдерами, в какой момент заходить на рынок со сделкой в том или ином направлении. То есть особо важным является выделение таких классов рыночных ситуаций, как ситуация, подходящая для покупки, ситуация, подходящая для продажи, а также определение такого состояния рынка, при котором выход на него является крайне рискованным, когда рынок находится в неопределенном состоянии и тенденции, определяющие его дальнейшую динамику еще не сформировались.

В связи с этим, в главе рассматривалась именно такая группа методов интеллектуального анализа данных, как методы классификации.

5. Проведено исследование основных методов, имеющих широкое распространение для решения реальных задач, такие как деревья решений, метод ближайшего соседа, метод опорных векторов. Первые два метода, несмотря на свои очевидные преимущества, будь то наглядность процедуры классификации, или простота в использовании, соответственно, не подходят для решения задачи классификации рыночных ситуаций, в первую очередь из-за своих ограниченных возможностей по работе с большими объемами входной выборки (многокритериальность, количество входных векторов).

Методы опорных векторов и нейросетевая классификация видятся более подходящими для решения задачи классификации рыночных ситуаций. Оба эти метода сводятся к решению задач оптимизации, для решения которых имеется большое количество численных и итерационных методов. Но учитывая, что для метода опорных векторов существует определенная проблема с выбором ядра в линейно неразделимых пространствах, и что нет никаких предпосылок считать по умолчанию пространство входных данных финансового рынка линейно разделимы-

ми, выбор в методе был сделан в пользу нейронных сетей. Нейронные сети решают проблему анализа линейно неразделимых пространств увеличением количества слоев сети [65, 97]. Соответственно, для решения задачи классификации в данной работе будет использована нейросетевая методология, а именно нейронная сеть, подходящей для задач распознавания образов и классификации топологии.

6. Задачу классификации или распознавания образов может быть решена с помощью различных топологий нейронных сетей. Многослойные нейронные перцептроны – классические сети для задачи классификации, за счет своих способностей к аппроксимации остаются ведущим инструментом среди нейронных сетей для задач классификации с учителем на среднеразмерных пространствах. Сверточные нейронные сети обладают мощным потенциалом при распознавании образов в условиях искажения информации и высокой размерности входных и выходных множеств (например, распознавание картографических снимков, распознавание рукописных текстов), но их использование в более простых задачах с меньшей размерностью является не совсем рациональным, так как их реализация более трудоемка, обучение более затянато, и теряется возможность визуального пояснения процесса обучения. Так же существуют варианты топологий нейронных сетей для распознавания образов с возможностями самообучения, то есть обучения без учителя, например, сети адаптивного резонанса. Использование данных сетей для задачи классификации рыночных ситуаций является избыточным, так как в данном случае классы уже предварительно обозначены. Таким образом, наиболее подходящей топологией нейронной сети для задачи классификации с учителем рыночных ситуаций является многослойный перцептрон.

Учитывая проведенный обзор и исследование методов анализа финансовых рынков и интеллектуального анализа данных, целью данной диссертационной работы является повышение эффективности принятия управленческих решений участников рынка, заключающееся в достижении ими положительных результатов при торговле, за счет применения интеллектуальной системы поддержки принятия решений трейдеров финансовых рынков.

Для достижения главной цели исследования в работе были поставлены следующие задачи:

1. Исследование и анализ существующих методов и средств анализа финансовых рынков и временных рядов, и выбор наиболее удовлетворяющих параметрам эффективности и программной реализуемости.
2. Исследование, анализ и выбор методов и средств интеллектуальных систем поддержки принятия решений, подходящих для задач классификации
3. Разработка метода генерации выходной выборки для обучения нейронной сети с учетом ожидаемой прибыли и повышения волатильности.

4. Разработка методики анализа финансовых рынков, основанной на нейросетевой методологии с применением методов технического, фундаментального анализа и эконометрического моделирования, и алгоритма ее реализации в системе поддержки принятия решений.

5. Проектирование и разработка интеллектуальной системы поддержки принятия решений на языке высокого уровня, исследование ее работоспособности на реальных примерах.

## 2 РАЗРАБОТКА МЕТОДОВ И АЛГОРИТМОВ ИНТЕЛЛЕКТУАЛЬНОЙ СИСТЕМЫ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ ТРЕЙДЕРОМ

Любой финансовый рынок – это открытая, динамически изменяющаяся сложная многофакторная система. Каждый игрок рынка: будь то профессиональный трейдер, инвестор, крупная финансовая компания либо случайный покупатель, заключающий единичный контракт на покупку или продажу финансового инструмента, – каждый вносит свой вклад в направление движения рынка, и совершение любой следующей сделки может оказаться переломным моментом в смене тренда. Рынок складывается из мнения всех агентов, участвующих в его функционировании, он интегрирует и аккумулирует в себе все точки зрения на его состояние в конкретный момент и выдает соответствующую реакцию, последовательность которых и является поведением рынка, то есть фактически динамикой изменения цены на данный финансовый инструмент.

Для прогнозирования состояния финансового рынка необходимо комплексно подходить к анализу его компонентов, то есть для верного прогноза требуется проанализировать стратегии торговли каждого его участника и выяснить в какую сторону склоняется преобладающая их часть. Но такого рода анализ на практике невозможен, поскольку одновременно участниками рынка могут являться сотни тысяч человек, территориально разделенных между собой на тысячи километров, и нет никакой гарантии, что каждый из них имеет доступ к всемирной сети интернет. К тому же, подобный анализ, займет слишком много времени, и скорее всего, полученная в результате него информация будет уже не актуальна и не пригодна для совершения эффективной торговой сделки. Выходом в сложившейся ситуации является полагаться на статистически значимое интегральное обобщенное мнение покупателей и продавцов финансовых инструментов, находящихся на рынках в конкретный момент времени.

Так как большинство игроков рынка являются сторонниками самых распространенных методологий его анализа: технический и фундаментальный анализа, эконометрическое моделирование, нейросетевой анализ, – и основывают свои торговые стратегии на полученных с их помощью прогнозах, автором представленного диссертационного исследования принято решение разработать методику анализа финансовых рынков для интеллектуальной системы поддержки принятия решений, включающий в себя все вышеназванные методы.

Нейросетевая методология находит все новые успешные применения в практике управления и принятия решений, в том числе в финансовой и торговой сферах. Лежащая в ее основе теория нелинейных адаптивных систем доказала свою полезность при выработке прогнозов в целом ряде отраслей экономики и финансов [132], а также потому что, как показано в преды-

душей главе, нейросетевая классификация является наиболее подходящим методом для задачи классификации рыночных ситуаций (состояний системы), именно поэтому она выбрана основной разработанной методики, и уже в ее рамках применены другие подходы к анализу финансовых рынков. Относительно простые с точки зрения математической интерпретации технические методики и алгоритмически более развитая эконометрическая модель встраиваются в принцип действия нейронной сети, улучшая качество исходной информации и повышая уровень надежности выводимых результатов.

Как описывалось в первой главе, для анализа рынка с помощью подхода технического анализа, наилучшие результаты показывает одновременное применение нескольких технических индикаторов из разных групп, например, индикатор тренда и осциллятор. Согласно данному утверждению, выбраны два индикатора технического анализа, применение которых доказывает свою эффективность на протяжении многих лет, а также подтверждена автором работы экспериментальным путем, и которые являются явными фаворитами в использовании их профессиональными трейдерами.

Также в процессе разработки методики возникал вопрос о способе применения в системе фундаментального анализа. Так как использование показателей фундаментального анализа для прогнозирования направления движения рынка является довольно спорным и не доказывает своей эффективности на коротких позициях, но несомненным остается тот факт, что рынок все же показывает реакцию в ожидании и в момент публикации экономических данных, принято решение рассматривать именно вопрос влияния экономических событий на увеличение волатильности в ряду определенного финансового инструмента.

Отобрана в результате исследований наиболее оптимальная по соотношению качества и сложности реализации эконометрическая модель, учитывающая свойства нестационарности временного ряда котировок финансового инструмента и изменения волатильности, модель обобщенной авторегрессионной условной гетероскедестичности.

Рассмотрим подробнее, каждый из выбранных методов анализа и их участие в общей методике.

## **2.1 Алгоритмы методов технического анализа**

В техническом анализе в качестве базового временного ряда используются ряды значений цены финансового инструмента за некоторый промежуток времени, объема торговли и числа открытых позиций.

Основным инструментом численных или аналитических методов является индикаторы.

Индикаторы технического анализа относятся к численным его методам и делятся на три группы: отслеживающие тренды, осцилляторы и характеристические.

Отслеживающие тренды включают: скользящие средние, MACD, MACD-гистограммы, Систему направлений (*Directional System*), индикатор Равновесного объема (OBV), Аккумуляции/Распыления и др. Они относятся к классу сопровождающих или запаздывающих - их разворот следует за разворотом тренда.

Осцилляторы помогают выявить точки разворота тренда. К ним относятся: Стохастик-индикатор, Показатель скорости изменения, Сглаженный Показатель скорости изменения, Индексы момента (*Momentum*), RSI, луч Элдера, Индекс Силы, %R Вильямса, Канальный индекс товаров (CCI) и др. Осцилляторы относятся к сопровождающим или опережающим индикаторам, изменения в них часто предшествуют действительным изменениям цен на рынке.

Характеристические индикаторы показывают внутреннее соотношение сил между медведями и быками. Их можно разделить на две группы: макроэкономические и индикаторы настроения рынка. Индикаторы первой группы анализируют общую экономическую ситуацию и изменения "макроэкономической среды" потенциально влияющей на рынок. Ко второй группе относятся индикаторы "ширины" рынка (индикаторы фондового рынка) и "психологические" индикаторы. Индикаторы этой группы пытаются определить по определенным внешним признакам направление внутренних, "глубинных" процессов на рынке [53, с. 58].

### 2.1.1 Индекс относительной силы

Индекс относительной силы (Relative Strength Index - RSI) был разработан Уэллесом Уайльдером Мл. (Welles J.Wilder Jr.). Он ввел его в статье, опубликованной в журнале «Commodities» (в настоящее время «Futures») в июне 1978 года. Это следующий за ценами осциллятор, который колеблется в диапазоне от 0 до 100, оценивающий силу операционных инициатив, отслеживая изменения в ценах закрытия. RSI - всегда опережающий или одновременный и никогда - отстающий индикатор.

$$RSI = 100 - \frac{100}{1 + RS} \quad (2.1),$$

где RS – относительная сила акции.

$$RS = \frac{S_u}{S_d} \quad (2.2),$$

где  $S_u$  – среднее значение увеличений за сутки цен закрытий за некоторое количество дней;  $S_d$  – среднее значение уменьшений за сутки цен закрытий за некоторое количество дней.



Фигуры пиков и впадин RSI не слишком зависят от длительности расчетного интервала. Сигналы индикатора становятся более отчетливыми на коротких интервалах в 7, 9 дней.

Фактически индекс равен процентному выражению доли всех повышений цен в итоговой сумме всех изменений цен за расчетный (базовый) период. Если обозначить сумму всех повышений за  $X$ , а понижений за  $Y$ , то

$$RSI = 100 \cdot \frac{X}{X + Y} \quad (2.3).$$

Пусть цены при этом выросли на величину  $C$ :  $X - Y = C$ , если при этом они росли непрерывно, то есть, не встречая сопротивления ( $Y = 0, X = C$ ), то  $RSI = 100\% X/(X+0) = 100\%$ .

Если же на протяжении всего расчетного периода медведи оказывали сильное сопротивление и цены совершали множество движений, то  $X$  и  $Y$  окажутся достаточно большими. Из  $Y = X - C$  следует

$$RSI = 100\% X/(2X - C) \rightarrow 50\%.$$

Контрольные значения для RSI часто берутся равными 30% и 70%. Сигналы, подаваемые данным индикатором:

- 1) Когда RSI поднимается выше верхнего контрольного уровня, индикатор показывает, что восходящий тренд силен, но рынок перенасыщен покупками и входит в область продаж. Означает начало убывающего («медвежьего») тренда.
- 2) Когда RSI опускается ниже нижнего контрольного уровня, индикатор показывает, что сильны продающие, но рынок перенасыщен продажами и входит в область покупок. Означает начало возрастающего («бычьего») тренда [53, с. 81-83].

Алгоритм вычисления RSI:

1. Если нынешняя цена больше предыдущей, то вычислить повышение цены, равное разности нынешней цены и предыдущей. Иначе вычислить понижение цены, равное разности предыдущей цены и нынешней.
2. Вычислить суммарные повышения  $X$  и понижения  $Y$  цен за выбранный период.
3. Вычислить значение индекса, равное процентному выражению отношения сумм всех повышений к сумме всех изменений цены.

### 2.1.2 Конвергенция-дивергенция скользящих средних

Скользящее среднее значение – это средняя цена финансового инструмента за определенный период. Период расчета скользящего среднего выбирается по усмотрению аналитика. Традиционно скользящее среднее используется для наблюдения за изменением цен. Обычно

инвесторы покупают, если цена бумаг поднимается выше скользящего среднего, и продают, когда она падает ниже него.

Существует три типа скользящих средних СС: простые СС, взвешенные СС и экспоненциальные СС.

Простое среднее скользящее.

Простое СС (moving average), или среднее арифметическое значение вычисляется по следующей формуле:

$$MA = \frac{\sum_{i=1}^n P_i}{n} \quad (2.4),$$

где  $P_i$  – цена  $i$ -го дня,  $n$ -порядок скользящей средней, то есть период расчета.

Взвешенное среднее скользящее.

Для того чтобы как-то решить проблему “удельного веса” средних значений цен, некоторые аналитики применяют взвешенные скользящие средние (weighted moving average). Они рассчитываются по следующей формуле:

$$WMA = \frac{\sum_{i=1}^n P_i \cdot W_i}{\sum_{i=1}^n W_i} \quad (2.5),$$

где  $W_i$  – вес  $i$ -го компонента (цены).

Экспоненциальное среднее скользящее.

Экспоненциальное скользящее среднее (exponential moving average) имеет более сложное построение, чем ВСС или простое СС, которое позволяет ЭСС устранить два недостатка, присущие простому СС. Во-первых, ЭСС придает гораздо большее значение показателям последних дней. Поэтому он является взвешенным. Но, хотя предшествующей динамике цен придается меньший вес, при вычислении используются все данные по ценам за весь период действия рынка ценной бумаги.

Математическая формула для расчета экспоненциального скользящего среднего является рекурсивной и при значении коэффициента сглаживания равном  $t$  имеет вид:

$$EMA_t = k \cdot P_t + (1 - k) \cdot EMA_{t-1} \quad (2.6),$$

где  $P_t$  – цена закрытия текущего торгового периода (дня),  $EMA(n-1)$  – значение экспоненциального скользящего среднего, рассчитанного для предыдущего торгового периода и  $k$  – регулирующий коэффициент.

Начальное значение  $EMA(I)$  принимается равным цене самого первого торгового периода, принимаемого в рассмотрение, то есть  $P(I)$ . Чем больше значение регулирующего коэффициента  $k$ , тем лучше кривая экспоненциального скользящего среднего аппроксимирует график, так как больше значение придается текущим ценам [95].

Джеральдом Аппелем в конце 1970-ых годов был построен более совершенный индикатор, чем скользящая средняя. Индикатор MACD (англ. *Moving Average Convergence/Divergence* – конвергенция/дивергенция скользящих средних) используют для проверки силы и направления тренда, а также определение разворотных точек. Строится на основе скользящих средних. Существует две модификации индикатора MACD: линейный MACD и MACD-гистограмма.

Дивергенция – расхождение направленности индикатора и графика цены таким образом, что более высокий максимум цены не подтверждается более высоким максимумом на индикаторе MACD (медвежья дивергенция) или наоборот, более низкий минимум не подтверждается минимумом на индикаторе (бычья дивергенция). Конвергенция является обратным понятием дивергенции и обозначает схождение направленности индикатора и графика цены.

Индикатор состоит из трех ЭСС и представляется на графике в виде двух линий, чьи пересечения служат сигналом к проведению операций.

КДСС состоит из двух кривых:

КДСС-линия. Строится на основе двух ЭСС, ее вычисляют как разность между 12-ти и 26-ти дневными скользящими средними цены акции. Полученная величина может быть, как выше, так и ниже нуля. Первая линия изменяется сравнительно быстро.

Сигнальная линия. Строится на основе сглаживания КДСС-линии третьей ЭСС. Обычно период сглаживания экспоненциального скользящего среднего индикатора выбирают 9 дней. Вторая линия меняется медленнее первой. Такое пересечение линий, при котором КДСС-линия становится над Сигнальной линией, означает сигнал покупать. Пересечение линий, при котором КДСС-линия уходит под Сигнальную линию, означает сигнал продавать.

Индикатор MACD – это разность между двумя скользящими средними цены. Если короткое скользящее среднее превосходит длинное (т.е. MACD поднимается выше нуля) – значит, ожидания инвесторов приобретают бычий характер (т.е. линии спроса/предложения сместились вверх). Дополнительное использование 9-ти дневного скользящего среднего MACD позволяет более точно определить момент подобных изменений в ожиданиях инвесторов.

Расстояние между быстрой (КДСС-линией) и медленной (сигнальной) линиями MACD в течение всего времени изменяется, эти изменения дают неплохие дополнительные сигналы. КДСС-гистограмма (MACD-histogramm) позволяет глубже понять, каково соотношение сил между склонными к покупке и продаже. КДСС-гистограмма строится по следующей формуле:

$$\text{КДСС-гистограмма} = \text{КДСС-линия} - \text{Сигнальная линия} \quad (2.7).$$

Правила использования КДСС-гистограмм:

1. Покупать, когда КДСС-гистограмма перестает понижаться и наметился сдвиг вверх.
2. Открывать короткую позицию на продажу, когда КДСС-гистограмма перестает расти и появился сдвиг вниз.
3. Дивергенция между КДСС-гистограммой и ценами определяет главные точки разворота рынка, при этом проявляется не при любых разворотах.
4. Открываться коротко на продажу, когда КДСС-гистограмма начала опускаться вниз со своей второй более низкой вершины, в то время как цены находятся на новом пике.
5. Покупать, когда КДСС-гистограмма начинает подниматься со своей второй более высокой впадины, в то время как цены упали на новый нижний уровень [53, с. 58-63].

Алгоритм расчета КДСС-гистограммы:

1. Вычислить двенадцати- и двадцати шестипериодную экспоненциальную скользящую среднюю по ценам финансового инструмента.
2. Вычислить значение КДСС-линии.
3. Вычислить значение Сигнальной линии, как девятипериодную экспоненциальную скользящую среднюю от КДСС-линии.
4. Вычислить значение КДСС-гистограммы, как разность между КДСС-линией и Сигнальной линией.

## 2.2 Алгоритмы метода эконометрического моделирования

### 2.2.1 Модель обобщенной условной авторегрессионной гетероскедастичности

Колебания цен отдельного финансового инструмента создают последовательность значений, представляющих собой временной ряд. Для анализа временного ряда используют методы эконометрического моделирования.

Решению задачи прогнозирования на основе моделей временных рядов посвящено большое количество научных работ. Исследованиям в данной области занимались Бокс Дж., Дженкинса Г., Ивченко Г.И., Боровиков В.П., Канторович Г.Г., Оссовского С., и др. [11, 13, 40].

Модель обобщенной авторегрессионной условной гетероскедастичности (Generalized Autoregressive Conditional Heteroskedasticity, GARCH) – эконометрическая модель для прогнозирования временных рядов, составленная на предположении о наличии гетероскедастичности остатков по ней. Под гетероскедастичностью понимается нестационарная дисперсия (волатильность), то есть неодинаковая в различные моменты времени. Здесь понятие волатильности включает в себя нестабильность условий, которое ведет к изменчивости моделируемого вре-

менного ряда. Условность предполагает наличие зависимости наблюдений от недавних значений. Авторегрессию является механизмом, с помощью которого происходит связь текущих и прошлых наблюдений. Резюмируя, GARCH – метод моделирования финансовых временных рядов, который использует предыдущие значения дисперсии для объяснения ее будущих значений. Основные идеи данной модели разработаны Энгельсом Р. и Нельсоном Д. [119, 126].

Многие финансовые временные ряды имеют в своей структуре устойчивые закономерности, такие как значимая положительность эксцесса (так называемые «толстые хвосты» в распределении) и «кластерность», разделенность на некоторые кластеры, группы высокой, средней и низкой волатильности. Учитывая данные закономерности при GARCH моделировании обеспечивается довольно высокая точность прогнозов значений дисперсии. GARCH модели применяются в различных областях, например, риск-менеджмент, портфолио-менеджмент, размещение активов, ценообразование опционов, обмен валюты.

Высокая значимость GARCH эффектов (гетероскедастичности) для финансовых рынков, не только для отдельных финансовых инструментов, но и для портфелей акций, биржевых индексов, валют и фьючерсов.

Несмотря на свое распространение в применении в управлении рисками и портфелями финансовых инструментов GARCH модели все же имеют некоторые ограничения:

- являются одной из частей решения, так как применяют для рядов доходностей финансовых инструментов, а для принятия финансовых решений необходим полный анализ ситуации;
- практически не способны учесть в модели феномены, связанные с непредсказуемыми рыночными изменениями, вызванными выходом экономических новостей и событий, и другими непредвиденными ситуациям, что приводит к значительным изменениям структуры временного ряда;
- гетероскедастичность не может полностью объяснить абсолютно все распределения с «толстыми хвостами».

В то же время GARCH модели работают с особенностями, характерными именно для финансовых временных рядов:

- «Толстые хвосты» – распределение временных рядов финансовых инструментов показывает более толстые хвосты, чем в случае нормального или гауссовского распределения.
- «Кластерность» волатильности – финансовые временные ряды обычно проявляют данную особенность, когда большие изменения приводят к последующим большим изменениям, а малые изменения – к малым. Знак последующего изменения (положительный или отрицательный) предсказать невозможно.

При рассмотрении финансовых временных рядов как последовательности случайных наблюдений, то эта последовательность порождается стохастическим процессом, и в ней может присутствовать некоторая степень корреляции в своих значениях. Данную корреляционную структуру можно использовать для прогноза будущих значений временного ряда, используя предыдущие наблюдения. Используя корреляционную структуру, временной ряд раскладывается на детерминированную составляющую (т.е. прогнозную часть) и случайную компоненту (т.е. ошибку, неопределенность, связанную с прогнозом).

Понятие GARCH модели включает в себя условную и безусловную дисперсию  $\{\varepsilon_t\}$ . Условность предполагает зависимость от предыдущих значений. GARCH модели характеризуются условным распределением  $\varepsilon_t$ . Модель дисперсии для GARCH:

$$D_{t-1}(y_t) = M(\varepsilon_t) = \sigma_t^2 \quad (2.8),$$

$$\text{где} \quad \sigma_t^2 = k + \sum_{i=1}^P G_i \sigma_{t-i}^2 + \sum_{j=1}^Q A_j \varepsilon_{t-j}^2 \quad (2.9).$$

При моделировании финансовых временных рядов предполагается, что ошибки прогноза некоррелированы между собой и имеют нулевое среднее:

$$M(\varepsilon_t \varepsilon_T) = \begin{cases} \sigma_t^2, & \text{если } t = T \\ 0, & \text{если } t \neq T \end{cases} \quad (2.10).$$

Ошибки прогноза некоррелированы, но вместе с тем не являются независимыми. Ошибки прогноза  $\{\varepsilon_t\}$  выражаются формулой:  $\varepsilon_t = \sigma_t z_t$ , где  $\sigma_t$  – условное стандартное отклонение;  $z_t$  – независимая одинаково распределенная случайная величина.

Существуют следующие ограничения для параметров условной дисперсии:

$$\begin{aligned} \sum_{i=1}^P G_i + \sum_{j=1}^Q A_j < 1, \quad k > 0, \\ G_i \geq 0, \quad i = 1 \dots P \\ A_j \geq 0, \quad j = 1 \dots Q \end{aligned} \quad (2.11).$$

Первое ограничение, ограничение стационарности, является необходимым и достаточным для существования предела дисперсии процесса  $\{\varepsilon_t\}$ . Остальные ограничения являются достаточными для обеспечения строго положительности условной дисперсии.

Формула для безусловной дисперсии процесса  $\{\varepsilon_t\}$  [18, с.229 -234]:

$$\sigma_t^2 = M(\varepsilon_t^2) = \frac{k}{1 - \sum_{i=1}^P G_i - \sum_{j=1}^Q A_j} \quad (2.12).$$

Для анализа и оценки GARCH-модели будем использовать гибкую модель для описания условного среднего – АРПСС модель, заключающую в себе авторегрессию (AR), скользящие средние (MA). Данная модель разработана Боксом и Дженкинсом в 1976 г.

При моделировании временных рядов будем условную дисперсию воссоздавать с помощью стандартного GARCH процесса с гауссовскими ошибками, то есть условную дисперсию можно описать формулой (2.9).

При анализе графиков финансовых инструментов становится очевидным, что нет постоянного среднего уровня цен, то есть данные временные ряды обладают свойством нестационарности. Хотя GARCH модели предоставляют возможность работы с нестационарными временными рядами, но в силу нестационарности таких рядов рекомендуется работать с производными рядами доходностей анализируемых финансовых инструментов.

Три этапа GARCH моделирования: предварительный анализ, оценка параметров модели, проверка модели на адекватность статистическим данным. На первом этапе производится исследование финансового временного ряда: производится преобразование цен в доходности, определяется автокорреляция в значениях ряда, выявляется гетероскедастичность в доходностях, проверяется пригодность временного ряда для GARCH-модели.

Для качественной проверки временного ряда на присутствие коррелированности значений используются автокорреляционная (АКФ) и частная автокорреляционная (ЧАКФ) функции. Для количественной оценки корреляции применяются методы проверки статистических гипотез, например, LBQ и ARCH тесты.

Нулевой гипотезой является то, что модель подобрана адекватно, то есть не выделяется последовательной корреляции в значениях временного ряда для определенных лагов.

Альтернативной гипотезой является утверждение, что модель неадекватна исходным данным.

LBQ тест основан на Q-статистике:

$$Q = N(N+1) \sum_{k=1}^L \frac{r_k^2}{N-k} \quad (2.13),$$

где  $N$  – размер выборки,

$L$  – число автокорреляционных лагов, включаемых в статистику,

$r_k^2$  – оценка выборочного коэффициента корреляции с лагом  $k$ .

Q-статистика имеет асимптотическое распределение  $\chi^2$  с  $L$  степенями свободы. При отсутствии в значениях временного ряда автокорреляции для соответствующих лагов Q-статистика будет мала. Нулевая гипотеза отвергается при превышении Q-статистикой критического значения  $\chi_L^2(\alpha)$ , где  $\alpha$  – заданный уровень значимости.

ARCH тест (Engle's ARCH test) используется для проверки временных рядов на присутствие в них ARCH эффектов.

Нулевая гипотеза: значения временного ряда независимы и имеют одинаковое Гауссово распределение, то есть ARCH эффекты отсутствуют.

Альтернативная гипотеза: во временном ряду присутствуют ARCH эффекты порядка  $M$ .

ARCH тест проверяет временной ряд на наличие в нем ARCH эффектов порядка  $M$  посредством регрессирования квадратов значений временного ряда на константу и  $M$  квадратов предыдущих значений временного ряда:

$$\varepsilon_t^2 = \alpha_0 + \alpha_1 \varepsilon_{t-1}^2 + \dots + \alpha_M \varepsilon_{t-M}^2 \quad (2.14).$$

В случае отсутствие во временном ряде ARCH или GARCH эффектов квадрат выборочного коэффициента корреляции  $R^2$  (коэффициент детерминации) будет мал. Статистика  $TR^2$  ( $T$  – число квадратов значений временного ряда, включаемых в регрессию) имеет асимптотическое распределение  $\chi^2$  с  $M$  степенями свободы. Если  $TR^2$  превышает критическое значение  $\chi_M^2(\alpha)$ , где  $\alpha$  – заданный уровень значимости, то нулевая гипотеза отвергается.

При проверке ARCH эффектов GARCH( $P, Q$ ) процесс эквивалентен ARCH( $P+Q$ ) процессу. Присутствие гетероскедастичности, наличие ARCH-эффектов в значениях временного ряда делает его подходящим для GARCH моделирования [18, с. 235-238].

Оценки параметров модели являются оценками максимального правдоподобия. Численные методы их нахождения описаны в следующем подразделе.

### 2.2.2 Метод максимального правдоподобия

При составлении эконометрической модели одной из наиболее важных итераций является оценивание неизвестных ее параметров. Существуют два распространенных метода оценивания параметров модели: метод наименьших квадратов и метод максимального правдоподобия. Первый из перечисленных методов имеет ряд ограничений для использования, главное из которых: условное по факторам математическое ожидание ошибок по модели должно быть равно нулю [57], то есть ошибки, получаемые путем сравнения прогнозного и реального значения, имеют нормальное (Гауссово) распределение. Но на практике редко встречаются такие временные ряды, построенные для которых модели, выдавали бы нормально распределенные ошибки. Не яв-



ляются исключением в этом смысле и временные ряды цен финансовых инструментов. Поэтому для оценки параметров модели, строящейся для данных рядов, необходимо применять более сложный, но и более точный метод максимального правдоподобия.

Метод максимального правдоподобия или метод наибольшего правдоподобия (ММП, ML, MLE – Maximum Likelihood Estimation) – метод оценивания неизвестного параметра путём максимизации функции правдоподобия. Основывается на предположении о том, что вся информация о статистической выборке содержится в функции правдоподобия [56].

Пусть  $Y$  – реализация  $N$ -мерной случайной величины, распределенной как:

- а)  $P_\theta(x)$  (вероятность) – в случае дискретного распределения.
- б)  $p_\theta(x)$  (плотность) – в случае непрерывного распределения.

Здесь  $P_\theta(x)$  ( $p_\theta(x)$ ) характеризует семейство распределений задаваемое параметром  $\theta \in \Theta$ ,  $\Theta \subset \rho^m$  – пространство параметров. В эконометрии принято говорить об этом семействе распределений как о порождающем данные процессе (ПДП). Будем считать, что рассматриваемый вектор наблюдений порожден распределением из этого семейства с параметром  $\theta_0 \in \Theta$ , которое будем называть истинным распределением, а  $\theta_0$  – истинным параметром.

Функция  $L(Y, \theta) = P_\theta(Y)$  (соответственно  $L(Y, \theta) = p_\theta(Y)$ ) называется функцией правдоподобия.

Оценкой максимального правдоподобия ( $\hat{\theta}$ ), сокращенно оценкой МП, называется решение задачи

$$L(Y, \theta) \rightarrow \max_{\theta \in \Theta} \quad (2.15).$$

Такой метод оценивания называют методом максимального правдоподобия.

Обычно удобнее пользоваться логарифмической функцией максимального правдоподобия

$$l(Y, \theta) = \ln(L(Y, \theta)) \quad (2.16).$$

Логарифм – (бесконечно) дифференцируемая возрастающая функция: поэтому можно находить оценки МП решая задачу

$$l(Y, \theta) \rightarrow \max_{\theta \in \Theta} \quad (2.17).$$

В частном случае вектор наблюдений состоит из независимых случайных величин. При этом

$$L(Y, \theta) = \prod_i L_i(Y_i, \theta) \quad (2.18),$$

$$l(Y, \theta) = \sum_i l_i(Y_i, \theta) \quad (2.19).$$

В целом, вектор наблюдений  $Y$  состоит из зависимых между собой и неодинаково распределенных случайных величин, то есть не является выборкой в распространенном смысле слова. Равенство в общем случае тоже будет верным если обозначить

$$L(Y, \theta) = p_\theta(Y_i | Y_{i-1}, \dots, Y_1) \text{ и } l(Y, \theta) = \ln(L(Y, \theta)).$$

Так задается разбиение функции правдоподобия на вклады отдельных наблюдений.

Поскольку  $Y$  – случайная величина, то функция правдоподобия – случайная величина при данном значении параметров. Оценка максимального правдоподобия является функцией вектора наблюдений:  $\hat{\theta} = \hat{\theta}(Y)$ , поэтому это тоже случайная величина. Соответственно, точно так же случайными величинами является значение функции правдоподобия в максимуме  $\hat{L}(Y) = L(Y, \hat{\theta})$  и многие другие рассматриваемые далее величины (градиент, гессиан и т. п.).

Пусть функция правдоподобия дифференцируема по  $\theta$  и достигает максимума во внутренней точке ( $\hat{\theta} \in \text{int}(\Theta)$ ), тогда оценка МП  $\hat{\theta}$  должна удовлетворять следующему условию первого порядка:

$$\frac{\partial L}{\partial \theta}(Y, \hat{\theta}) = 0 \text{ или } \frac{\partial l}{\partial \theta}(Y, \hat{\theta}) = 0.$$

Таким образом, градиент логарифмической функции правдоподобия  $g(\theta)$  при  $\theta = \hat{\theta}$  должен быть равен нулю.

Для максимального правдоподобия оценок, удовлетворяющих этим уравнениям правдоподобия, необходимо и достаточно выполнение условий второго порядка (предполагая, что функция правдоподобия дважды дифференцируема). То есть, матрица Гессе (гессиан) логарифмической функции правдоподобия должна быть всюду отрицательно определена. Матрица Гессе  $H$  по определению есть матрица вторых производных:

$$H_{ij}(Y, \theta) = \frac{\partial^2 l}{\partial \theta_i \partial \theta_j}, \quad i, j = 1, \dots, m \quad (2.20).$$

С помощью матричного дифференцирования можно записать гессиан в виде

$$H = \frac{\partial^2 l}{\partial \theta \partial \theta^T} \quad (2.21).$$

В некоторых моделях функция правдоподобия неограниченна сверху и не существует оценок максимального правдоподобия по приведенному выше описанию. Альтернативный ме-

тод определения оценок максимального правдоподобия называют ими корни уравнения правдоподобия, являющиеся локальными максимумами функции правдоподобия. Существуют модели, для которых такие оценки состоятельны.

Информационной матрицей для вектора наблюдений размерностью  $N$  будем называть матрицу

$$I^N(\theta) = E(g(Y, \theta)g^T(Y, \theta)) \quad (2.22).$$

Пусть  $\varphi(Y)$  есть некоторая функция вектора наблюдений  $Y$ . Тогда ее математическое ожидание равно

$$E(\varphi(Y)) = \int_y \varphi(Y)L(\theta_0, Y)dY \quad (2.23),$$

где  $y$  обозначает пространство элементарных событий (пространство переменной  $Y$ ).

Таким образом, можно переписать определение информационной матрицы в виде

$$I(\theta) = \int_y g(Y, \theta)g^T(Y, \theta)L(Y, \theta)dY \quad (2.24).$$

Асимптотическая информационная матрица есть предел

$$I^\infty(\theta) = \lim_{N \rightarrow \infty} \frac{1}{N} I^N(\theta) \quad (2.25).$$

Если наблюдения независимы и одинаково распределены, то информация растет пропорционально количеству наблюдений [105, с.54-57].

Метод максимального правдоподобия используются для нахождения способа расчетов, а затем уже доказательства, какими свойствами обладает этот метод по отношению к более широкому классу распределений.

ММП для случая регрессии с нормально распределенными ошибками можно получить МНК, который обладает «хорошими» свойствами и при ошибках, которые уже не имеют нормального распределения, но при этом эффективность уменьшается. Обратная связь между этими методами заключается в том, что МНК используют как вычислительную процедуру для нахождения оценки МП и построения тестов. Такое техническое использование МНК называют вспомогательной регрессией. Вслед за Дэвидсоном и Мак-Кинноном будем использовать термин искусственной регрессии, если вспомогательная регрессия может быть применена и для нахождения оценок, и для проверки гипотез относительно полученных оценок и проверки правильности спецификации модели.

Функцию правдоподобия разбивается по вкладам отдельных наблюдений:  $l(Y, \theta) = \sum_i l_i(Y_i, \hat{\theta})$ . То же самое можно проделать с градиентом. Определим матрицу вкладов в градиент отдельных наблюдений  $G$  как

$$G_{ij}(\theta) = \frac{\partial l_i}{\partial \theta_j}(\theta) \quad (2.26).$$

$$\text{При этом } \sum_i G_{ij} = \sum_i \frac{\partial l_i}{\partial \theta_j} = \frac{\partial}{\partial \theta_j} \sum_i l_i = \frac{\partial l}{\partial \theta_j} = g_j.$$

Выражение для информационной матрицы в случае линейной регрессии с нормально распределенными ошибками прямо по определению:

$$E(G_0^T G_0) = E \sum_i (G_{0i}^T G_{0i}) = E \left( \sum_i (G_{0i})^T \left( \sum_i G_{0i} \right) \right) = \sum_i (g_0 g_0^T).$$

Последнее выражение по определению есть информационная матрица. Таким образом, получается [105, с.62-65]

$$E(G_0^T G_0) = I_0 \quad (2.27).$$

Рассмотрим семейство универсальных алгоритмов вычисления оценок максимального правдоподобия, тесно связанных с рассмотренными способами получения матрицы  $\hat{I}$ . Такого рода алгоритмы являются итеративными градиентными методами, где  $t$ -й шаг алгоритма задается формулой

$$\theta^{t+1} = \theta^t + (\hat{I}^t)^{-1} g(\theta^t) \quad (2.28).$$

Стационарная точка этого процесса  $\theta^{t+1} = \theta^t$  будет удовлетворять уравнениям правдоподобия  $g = 0$  и (с соответствующими оговорками) будет оценкой максимального правдоподобия.

Если в качестве  $\hat{I}^t$  взять информационную матрицу в точке оценок  $I(\theta^t)$ , то мы получаем метод, называемый «method of scoring»:

$$\theta^{t+1} = \theta^t + I(\theta^t)^{-1} g(\theta^t) \quad (2.29).$$

Если в качестве  $\hat{I}^t$  взять минус гессиан  $-H(\theta^t)$  то мы получаем классический метод Ньютона:

$$\theta^{t+1} = \theta^t - H(\theta^t)^{-1} g(\theta^t) \quad (2.30).$$

Шаг метода ВПГ (внешнего произведения градиентов, OPG) можно получить с помощью искусственной регрессии, в которой зависимой переменной будет вектор, составленный из единиц (обозначим его  $\mathbf{1}$ ), а матрицей регрессоров – матрица  $G(\theta^t)$ . Если  $\Delta\theta^t$  – оценки коэффициентов в этой вспомогательной регрессии на  $t$ -м шаге, то итерацию выразим как

$$\theta^{t+1} = \theta^t + \Delta\theta^t \quad (2.31),$$

$$\text{где } \Delta\theta^t = (G(\theta^t)^T G(\theta^t))^{-1} G(\theta^t)^T \mathbf{1} \quad (2.32).$$

Последний алгоритм является самым простым, но он сходится очень медленно. Учитывая, что обычно при использовании данного метода  $(G(\theta^t)^T G(\theta^t))^{-1}$  берут в качестве оценки ковариационной матрицы оценок, то его использование нежелательно.

Существуют различные модификации основной идеи метода.

Шаг алгоритма вычисляется, домножая исходный шаг на параметр  $\lambda$  :

$$\theta^{t+1} = \theta^t + \lambda(\hat{I}^t)^{-1} g(\theta^t) \quad (2.33).$$

Параметр  $\lambda$  выбирается так, чтобы максимизировать по нему функцию правдоподобия в точке  $\theta^{t+1}$  [105, с. 20-23]:

$$\lambda = \arg \max \lambda(\theta^t + \lambda(\hat{I}^t)^{-1} g(\theta^t)) \quad (2.34).$$

Алгоритм вычисления значения по модели GARCH:

1. Преобразовать цены в доходы от финансовых инструментов:

$$z_t = \Delta x_t = \Delta^2 y_t = x_t - x_{t-1} = y_t - 2y_{t-1} + y_{t-2}$$

2. Вычисление значений по модели АРПСС(1,1):  $\hat{z}_t = c + \alpha_1 \cdot z_t + \beta_1(z_t - \bar{z}_{t-1})$

3. Вычисление ошибок по АРПСС:  $e_t = z_t - \hat{z}_t$

4. Вычисление градиентов остатков  $d\varepsilon / d\theta$  :

$$\frac{d\varepsilon_t}{d\theta} = (-1, -y_{t-1}, -\varepsilon_{t-1})$$

5. Вычислить градиенты условных дисперсий  $H_n$  :

– первая строка состоит из нулей, только последний элемент равен 1;

– для остальных строк вычисляются по формуле:  $\frac{d\sigma_t^2}{d\theta} = (0, 0, 0, 1, \sigma_{t-1}^2, \varepsilon_{t-1}^2, 0)$

6. Вычислить  $Q, S, \xi, \nu$  :

$$Q_t = \frac{1}{\sigma_t} \cdot \frac{d\varepsilon_t}{d\theta},$$

$$S_t = \frac{1}{\sigma_t^2} \cdot \frac{d\sigma_t^2}{d\theta},$$

$$\xi_t = \frac{\varepsilon_t}{\sigma_t},$$

$$v_t = \xi_t^2 - 1.$$

7. Оценить искусственную регрессию и получить направление изменения параметров  $\Delta$ :

$$8. \quad \Delta = (Q^{tT} \cdot Q^t + \frac{1}{2} \cdot S^{tT} \cdot S^t) \cdot (Q^{tT} \cdot \xi^t + \frac{1}{2} \cdot S^{tT} \cdot v^t)$$

9. Вычислить новые коэффициенты:  $Q_1 = Q_0 + \lambda \cdot \Delta$

10. Вычислить значения, получаемые по уравнению GARCH:  $\hat{z}_t = q_1 + q_2 \cdot z_t + q_3 \cdot e_t$

### 2.3 Алгоритмы анализа с применением методов искусственного интеллекта

Обработка исследуемых цен финансовых инструментов методами технических индикаторов (RSI, MACD) и получение прогнозного значения по GARCH-модели являются лишь подготовительным этапом перед началом интеллектуального анализа ситуаций на финансовых рынках, составляющего ядро разработанного механизма. Для данных целей выбран популярный в настоящее время, широко применяемый, активно развивающийся метод нейронных сетей.

Теория искусственных нейронных сетей была создана и получила свое развитие в работах таких ученых, как Розенблатт Ф., Минский М., Пейперт С., Кохонен Т, Гроссберг С. [117, 122, 124, 129]. Вопросам теории и практического использования искусственных нейронных сетей посвящены работы таких ученых России, как Галушкин А.И., Горбань А.Н., Россиев Д.А., Шумский С.А., Ежов А.А., Миркес Е.М. [115, 21, 24, 25, 33, 61].

Поскольку сети с прямой связью, такие как многослойный перцептрон (описание в 1 главе), являются универсальным средством аппроксимации функции, их с успехом можно использовать для решения задач классификации. В диссертации использована более всего подходящая топология нейронных сетей для задачи классификации – многослойный перцептрон. Многослойными перцептронами называют нейронные сети прямого распространения. Входной сигнал в таких сетях распространяется в прямом направлении, от слоя к слою (подробнее о многослойном перцептроне см. [128]). Богатые возможности отображения особенно важны в тех случаях, когда на основе нескольких оценок строится высокоуровневая процедура принятия решений. Задача классификации может быть решена на сети с одним выходным нейроном.

Имеются различные проблемы, затрагивающие составление и обучение многослойного перцептрона [125]:

### 1. Выбор количества скрытых слоев.

Почти для всех задач одного промежуточного слоя достаточно. Два слоя необходимы при моделировании данных с разрывами. При проектировании нейронной сети для классификации рыночных ситуаций решено начать с одного скрытого слоя.

### 2. Сколько нейронов использовать в каждом скрытом слое.

Если используется недостаточное количество нейронов, сеть будет не в состоянии моделировать сложные данные. Если используются слишком много нейронов, увеличивается время обучения, и сеть перестает соответствовать данным. Когда происходит переобучение, сеть начинает моделировать случайный шум в данных. Необходимое количество нейронов в скрытом слое будет устанавливаться в ходе проведения экспериментальной части.

### 3. Нахождение глобального оптимального решения, избежав локального минимума.

Если изобразить график ошибок как функцию весов, то он будет соответствовать поверхности с большим количеством локальных минимумов, двумерное представление которой приведено на рисунке 2.1.

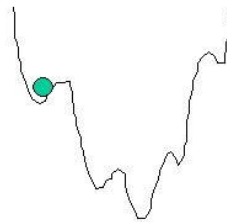


Рисунок 2.1 – Поверхность ошибок МСП

В обычной нейронной сети на практике может быть намного больше двух измерений, в каждом из которых поверхность имеет большое число локальных минимумов. Оптимизационные методы (пошаговый спуск и сопряженный градиент) высоко восприимчивы к нахождению локального минимума, если они начинают исследование вблизи локального минимума.

### 4. Схождение к оптимальному решению в приемлемые сроки.

Так как проектируемая система поддержки принятия решений будет работать в режиме реального времени, и нейронная сеть будет обучаться также в реальном времени, то при проектировании будут заданы определенные временные рамки, в которые нужно будет укладываться. Если время обучения проектируемой нейросети будет выходить за эти рамки, то встанет вопрос об изменении ее архитектуры.

### 5. Проверка нейронной сети на переобучаемость.

Отслеживание этого вопроса будет производиться в процессе тестирования нейронной сети.

При применении нейронных сетей как классификаторов при анализе финансовых рынков необходимо помнить о важности выбора алгоритма обучения, предварительной предобработки данных и формировании обучающей выборки. Процесс обучения сети представляет собой задачу минимизации ошибки обучения (разницы между ответами взвешанного сумматора и предъявленным выходным образом) по средствам подбора необходимых весов каждого из слоев сети. Общий алгоритм обучения сети можно представить следующим образом:

1. Инициализация весов сети рандомизированными малыми значениями.
2. Подача на вход сети вектора обучающей выборки  $X$  и вычисление сигнала  $NET$  от каждого нейрона.

$$NET_j = \sum_{i=1}^n w_{ij} x_i \quad (2.35).$$

3. Вычисление значения пороговой функции нейрона для сигнала  $NET$  для каждого нейрона:

$$OUT_j = \begin{cases} 1, & NET_j > \theta_j \\ 0, & NET_j \leq \theta_j \end{cases} \quad (2.36),$$

где  $\theta_j$  представляет собой порог, соответствующий  $j$ -му нейрону (пример для самого простого варианта пороговой функции).

4. Вычисляется ошибка для каждого нейрона:

$$e_j = Tr_j - OUT_j \quad (2.37),$$

где  $Tr_j$  – требуемый выход нейрона.

5. Модификация веса нейронов:

$$w_{ij}(t+1) = w_{ij}(t) + \alpha x_i e_j \quad (2.38),$$

где  $\alpha$  - коэффициент, позволяющий регулировать скорость обучения сети.

Задача обучения многослойных сетей по сути сводится к задаче многокритериальной безусловной оптимизации, для решения которой применяются такие методы, как метод оптимизации Ньютона, методы наискорейшего спуска, в том числе градиентные методы, и их специализированные модификации для нейронных сетей, такие как алгоритм обратного распространения ошибки.

### 2.3.1 Нормировка данных для нейронных сетей

На практике в процессе нейросетевого анализа предобработка данных может стать наиболее трудоемким элементом нейросетевого анализа. Причем, знание основных принципов и приемов предобработки данных не менее, а может быть даже более важно, чем знание собственно



нейросетевых алгоритмов. Хотя предобработка не связана непосредственно с нейросетями, она является одним из ключевых элементов этой информационной технологии. Успех обучения нейросети может решающим образом зависеть от того, в каком виде представлена информация для ее обучения. Нормировка данных – это первая ступень предобработки, поэтому она является важным этапом предобработки данных для нейросети. От результатов правильной нормировки данных будет зависеть весь последующий процесс обучения и анализа функционирования нейронной сети.

Нормировка – это корректировка ряда (вектора) значений в соответствии с некоторыми функциями преобразования, с целью сделать их более удобными для анализа и (или) сравнения. Нормировка данных требуется, когда несовместимость единиц измерений переменных может отразиться на результатах (как в случае обучения нейросети), и рекомендуется в тех случаях, когда итоговые отчеты могут быть улучшены, если выразить результаты в определенных понятных/совместимых единицах.

Так на входы проектируемой нейронной сети будут подаваться разнородные данные:

- цена финансового инструмента в денежных единицах (рубли, доллары), либо в пунктах (например, как на валютном рынке);
- значение индикатора RSI в пределах от 0 до 100;
- значение индикатора MACD в пределах от -10 до 10.

Чтобы при обучении нейросети они имели равный вес и вносили равный вклад в функцию ошибок, необходимо провести нормировку этих данных.

Важно учесть, что, если с точки зрения операции умножения значения  $\pm 1$  равноправны, между 0 и 1 имеется существенная асимметрия: нулевые значения не дают никакого вклада в градиент ошибки. Таким образом, выбор схемы кодирования входов влияет на процесс обучения. В силу логической равноправности обоих значений входов, более предпочтительной выглядит симметричная кодировка:  $\{-1;1\}$ , сохраняющая это равноправие в процессе обучения.

Для нормировки основной массы данных и одновременного ограничения диапазона возможных значений можно использовать для предобработки данных сигмоидальные функции активации нейронов. Например, нелинейное преобразование с использованием функции гиперболического тангенса:

$$\tilde{x}_i = th\left(\frac{y}{\alpha}\right) \quad (2.39),$$

где  $y = \frac{x_i - \bar{x}_i}{\sigma_i}$ , а  $\alpha$  – коэффициент наклон сигмоидальной функции, нормирует основную

массу данных и дает возможность разместить данные в интервале  $\tilde{x}_i \in [-1;1]$ . Это предупреждает опасность асимметрии и неравномерного вклада входов при вычислении градиента во время обучения, что может произойти при нормировании с помощью логистической функции, так как применяя ее все данные размещаются в единичном интервале  $\tilde{x}_i \in [0;1]$ .

Для подбора наиболее оптимального способа нормировки проведем экспериментальную нормировку курса акций ОАО «Газпром» и соответствующие им значения индикатора технического анализа RSI (Relative Strength Index) (см. рисунок 2.2), которые готовим для подачи на входы нейронной сети. Эти данные являются разнородными, так как курс акций изменяется в диапазоне от 196 до 216 рублей, а значения индикатора от 0 до 100 пунктов.

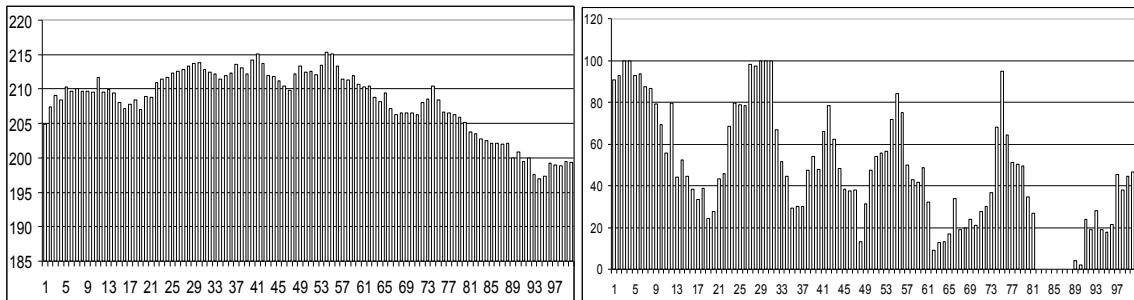


Рисунок 2.2 – Ежечасные значения цен акций ОАО «Газпром» (слева) и RSI (справа) с 1.02.2011 по 16.02.2011 на ММВБ.

Нормировка с помощью использования активационной функции нейронов гиперболического тангенса предусматривает подбор коэффициента наклона. При коэффициенте наклона  $\alpha=1$  обнаруживаются всплески в данных (см. рисунок 2.3).

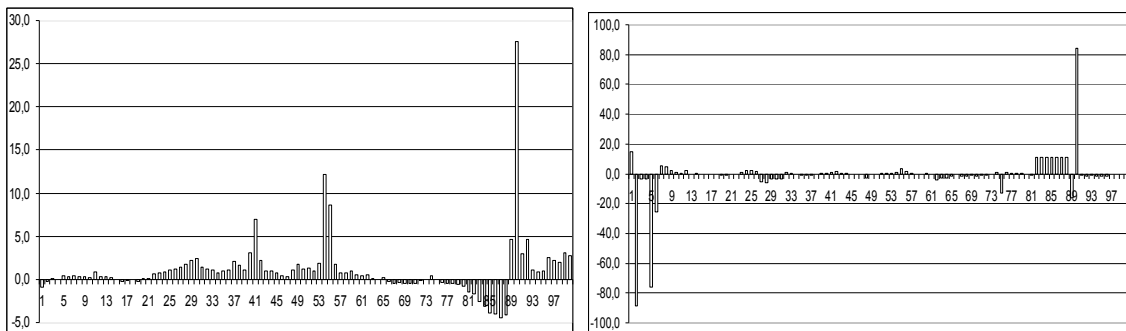


Рисунок 2.3 – Нелинейная нормировка цен акций ОАО «Газпром» (слева) и RSI (справа) с применением гиперболического тангенса с коэффициентом наклона 1.

Для устранения всплесков и приведения всех данных в интервал  $\tilde{x}_i \in [-1;1]$  необходимо применить следующий алгоритм:

- выяснить максимальное по модулю значение из данных при нормировке гиперболическим тангенсом с коэффициентом наклона 1;
- выяснить для этого значения величину  $y$  из формулы  $y_{max}$ ;
- вычислить значение коэффициента наклона по формуле:

$$\alpha = \frac{y_{max} \cdot 4}{\pi} \quad (2.40),$$

исходя из того, что  $tg(\frac{\pi}{4}) = 1$ .

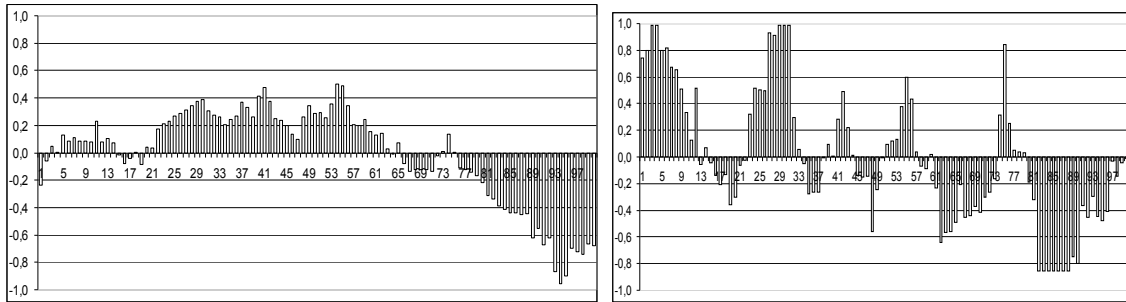


Рисунок 2.4 – Нелинейная нормировка цен акций ОАО «Газпром» (слева) и RSI (справа) с применением гиперболического тангенса с подобранными коэффициентами наклона.

Таким образом из разнородных данных получаем нормированные относительно средних величин выборки и обе группы данных расположены в интервале  $[-1;1]$  (см. рисунок 2.4), что увеличивает информативную составляющую данных, то есть информационную энтропию. Нормированные таким образом цены акций ОАО «Газпром» и индикатора RSI становятся пригодными для нейросетевого анализа. Они позволят эффективно обучить нейронную сеть для классификации рыночных ситуаций. Каждое новое данное, как цены, так и индикатора будет вносить равный вклад в процесс обучения [76].

### 2.3.2 Алгоритмы обучения многослойного перцептрона

В целом задача обучения многослойного перцептрона сводится к поиску минимума функции ошибки  $E(w)$ , зависящей от набора всех синаптических весов сети  $w$ . В основе большинства известных сегодня алгоритмов обучения нейросетей лежит метод градиентной оптимизации – итерационное изменение синаптических весов, постепенно понижающее ошибку обработки нейросетью обучающих примеров. Причем изменения весов происходят с учетом локального градиента функции ошибки. Задача обучения сводится к задаче нахождения глобального минимума функции многих переменных. При выборе алгоритма обучения возникает 2 проблемы: во-первых, вычислительные затраты на нахождение градиентов функций многих переменных могут быть оказаться слишком велики, что значительно замедлит процесс обучения, во-вторых, существует проблема схождения алгоритма к локальному минимуму и застреванию

в нем. Для решения первой проблемы используют алгоритмы, минимизирующие количество итераций для расчета градиента, для решения второй проблемы в алгоритм вводят элементы стохастической оптимизации.

### 2.3.2.1 Алгоритм обратного распространения ошибки

Самый известный пример алгоритма обучения нейронных сетей и, в частности, многослойного перцептрона – алгоритм обратного распространения (error back-propagation), предложенный Руммельхартом и Хинтоном в 1986 г. [131]. Современные алгоритмы второго порядка, такие как спуск по сопряженным градиентам и Левенберга-Марквардт существенно быстрее при решении многих задач, но обратное распространение все еще имеет преимущества в некоторых обстоятельствах, и является самым легким алгоритмом для понимания.

Суть алгоритма обратного распространения заключается в распространении по сети навстречу потоку сигналов разницы между желаемыми и фактическими выходами нейросети, определяемые на выходном слое нейронов. В итоге каждый нейрон способен определить вклад каждого своего веса в суммарную ошибку сети простым умножением невязки этого нейрона на значение соответствующего входа. Простейшее правило обучения соответствует методу наискорейшего спуска – изменения синаптических весов пропорциональны вкладу каждого веса в общую ошибку.

В обратном распространении, вычисляется вектор градиента поверхности ошибок. Этот вектор указывает вдоль линии самого крутого спуска от текущей точки, таким образом, мы знаем, что, если мы пройдем это "короткое" расстояние, мы уменьшим ошибку. Последовательность таких шагов (замедляющийся около основания) в конечном счете найдет минимум некоторого вида. Трудным является выбор размера шагов [127].

Большие шаги могут сходиться быстрее, но также могут переступить через решение или уйти в неправильном направлении. Напротив, очень маленькие шаги могут найти правильное направление, но они требуют большого количества итераций. Практически, размер шага пропорционален наклону и к специальной константе: уровень обучения.

Исследования в области рационализации вопросов вычисления градиента функции многих переменных, какой и является математическая постановка функционирования и обучения нейронной сети, занимались Румельхарт, Хинтон и Вильямс, а также представители русской школы Барцева и Охонина. Подробный обзор полученных результатов представлен в труде Горбань А.Н. «Нейроинформатика» в главе 3 «Быстрое дифференцирование, двойственность и обратное распространение ошибки» [23].

Основная идея заключается в следующем: если представить поиск градиента  $\text{grad}H$  как некоторый двойственный процесс над структурой вычисления  $H$ , где  $H$  сложная функция, являющаяся суперпозицией функций малого числа переменных, и воспользоваться правилом дифференцирования сложных функций, сохраняя промежуточные результаты, то вычисление всей совокупности  $\partial H / \partial x_i$  ( $i=1, \dots, n$ ) получается не многим сложнее, чем одной из этих функций, так как они состоят из одинаковых блоков. Вычисление градиента сложной функции представляется как вычислительный процесс, в ходе которого сигналы распространяются в обратном направлении от выходных элементов к входным. И также вычисление сложной функции будет складываться из прохождения каждого из элементов сети за счет структуры связей между элементами. Таким образом одна и та же структура нейронной сети используется как для ее функционирования, так и для ее обучения. Сами вычисления в обоих случаях распределены по конкретным нейронам, то есть каждый нейрон производит вычисления при поступлении к нему сигналов от его входов и выходов.

Алгоритм обратного распространения ошибки следующий:

1. Инициализировать синаптические веса случайными значениями близкими к 0.
2. Выбрать очередную обучающую пару (входные и выходное значения) из обучающего множества; подать входной вектор на вход сети.

При этом нейроны последовательно от слоя к слою функционируют по следующим формулам:

- скрытый слой

$$x_j = \sum_j w_{ij} x_j^{\alpha}, \quad y_j = \sum_j \sigma(x_j) \quad (2.41),$$

- выходной слой

$$x_k = \sum_k v_{ij} y_j, \quad y_k = \sum_k \sigma(x_k) \quad (2.42),$$

где  $\sigma(x)$  – функция активации,

$x_j, x_k$  – входы;

$w_{ij}, v_{ij}$  – веса;

$y_j, y_k$  – выходы.

3. Вычислить выход сети.
4. Вычислить разность между выходом сети и требуемым выходом (целевым вектором обучающей пары).

Функция ошибки для данного входного набора вычисляется по следующей формуле:

$$E = \frac{1}{2} \sum_k (y_k - Y_k^\alpha)^2 \quad (2.43),$$

где  $y_k$  – выходы выходного слоя;

$Y_k^\alpha$  – реальные выходы. [123, с. 34-35]

Данная функция подлежит минимизации. Классический градиентный метод оптимизации состоит в итерационном уточнении аргумента согласно формуле:

$$v_{jk}(t+1) = v_{jk}(t) - h \cdot \frac{\partial E}{\partial v_{jk}} \quad (2.44),$$

где параметр  $h$  имеет смысл скорости обучения и выбирается достаточно малым для сходимости метода.

Дифференцируя функцию получаем:

$$\frac{\partial E}{\partial y_k} = \delta_k = (y_k - Y_k^\alpha) \quad (2.45);$$

$$\frac{\partial E}{\partial x_k} = \frac{\partial E}{\partial y_k} \cdot \frac{\partial y_k}{\partial x_k} = \delta_k \cdot y_k (1 - y_k) \quad (2.46);$$

$$\frac{\partial E}{\partial v_{jk}} = \frac{\partial E}{\partial y_k} \cdot \frac{\partial y_k}{\partial x_k} \cdot \frac{\partial x_k}{\partial v_{jk}} = \delta_k \cdot y_k (1 - y_k) \cdot y_j \quad (2.47).$$

Таким образом, все необходимые величины для подстройки весов выходного слоя получены.

5. Подкорректировать веса сети для минимизации ошибки.

Согласно градиентному методу веса равны:

$$w_{ij}(t+1) = w_{ij}(t) - h \cdot \frac{\partial E}{\partial w_{ij}} \quad (2.48).$$

Вычисления производных выполняются по тем же формулам, за исключением некоторого усложнения формулы для ошибки  $d_j$ .

$$\frac{\partial E}{\partial y_j} = \delta_j = \sum_k \frac{\partial E}{\partial x_k} \cdot \frac{\partial x_k}{\partial y_j} = \sum_k \delta_k \cdot y_k (1 - y_k) \cdot v_{jk} \quad (2.49);$$

$$\frac{\partial E}{\partial x_k} = \frac{\partial E}{\partial y_k} \cdot \frac{\partial y_k}{\partial x_k} = \delta_k \cdot y_k (1 - y_k) \quad (2.50);$$

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial y_j} \cdot \frac{\partial y_j}{\partial x_j} \cdot \frac{\partial x_j}{\partial w_{ij}} = \delta_j \cdot y_j (1 - y_j) \cdot x_i^\alpha = \left[ \sum_k \delta_k \cdot y_k (1 - y_k) \cdot v_{jk} \right] \cdot \left[ y_j (1 - y_j) \cdot x_i^\alpha \right] \quad (2.51).$$

При вычислении  $\delta_j$  здесь и был применен принцип обратного распространения ошибки: частные производные берутся только по переменным последующего слоя.

По полученным формулам модифицируются веса нейронов скрытого слоя. Если в нейронной сети имеется несколько скрытых слоев, процедура обратного распространения применяется последовательно для каждого из них, начиная со слоя, предшествующего выходному, и далее до слоя, следующего за входным. При этом формулы сохраняют свой вид с заменой элементов выходного слоя на элементы соответствующего скрытого слоя.

6. Повторять шаги с 2 по 5 для каждого вектора обучающего множества до тех пор, пока ошибка на всем множестве не достигнет приемлемого уровня, который задается критерием остановки обучения.

Особенность обратного распространения ошибки состоит в том, что для ее оценки для нейронов скрытых слоев можно принять взвешенную сумму ошибок последующего слоя.

Появление алгоритма обратного распространения ошибки стало знаковым событием в области развития нейронных сетей, так как он реализует вычислительно эффективный метод обучения многослойного перцептрона, и он развеял пессимизм относительно обучения многослойных сетей, воцарившийся в результате публикации Минского [124] в 1969 году.

Применение сигмоидальных функций активации облегчают вычислительную процедуру алгоритма обратного распространения ошибок за счет возможности выражения дифференциала функции логистической, гиперболического тангенса через саму функцию:

$$\frac{\partial \sigma(x)}{\partial x} = \sigma(x) \cdot (1 - \sigma(x)), \text{ где } \sigma(x) = \frac{1}{1 + e^{-x}} \text{ – логистическая функция;}$$

$$\frac{\partial \sigma(x)}{\partial x} = (1 + \sigma(x)) \cdot (1 - \sigma(x)), \text{ где } \sigma(x) = th \frac{x}{\alpha} \text{ – функция гиперболического тангенса.}$$

### 2.3.2.2 Альтернативные методы обучения многослойного перцептрона

Задача минимизации функции эмпирической ошибки нейронной сети топологии многослойного перцептрона может быть решена и с помощью других алгоритмов обучения.

Одним из подходов к рассмотрению задачи обучения является математическая постановка задачи обучения многослойного перцептрона с точки зрения классической оптимизации.

Поставим задачу рассмотрения альтернативного варианта отыскания весов нейронов одной из популярных архитектур нейросетей для распознавания образов и классификации – многослойного перцептрона. Будем считать, что исходные данные для обучения уже прошли процедуру нормировки.

Для этого вернемся к математическим основам нейросетевой теории. Математическая модель нейронной сети основана на теореме Колмогорова (Колмогоров, 1957) которая утверждает,

что любая непрерывная функция  $n$  аргументов на единичном кубе  $[0,1]^n$  представима в виде суперпозиции непрерывных функций одного аргумента и операции сложения.

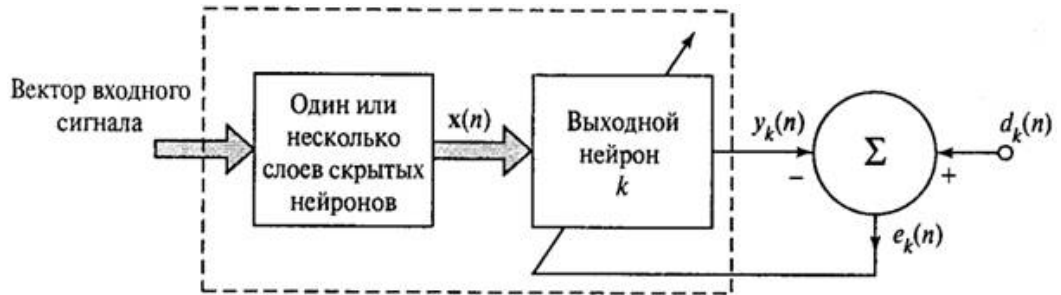


Рисунок 2.5 – Многослойная нейронная сеть прямого распространения

Выходы двухслойной нейросети (рис. 2.5) представляют собой суперпозицию функций одного аргумента с операциями сложения:

$$f(x_1, x_2, \dots, x_n) = \sum_{k=1}^{2n+1} h_k \left( \sum_{i=1}^n \varphi_{ik}(x_i) \right) \quad (2.52),$$

где  $h_k$  – функция активации нейронов скрытого слоя,  $x_i$  – входные нейроны с весами  $\varphi_{ik}$ .

Цель функционирования сети найти оптимальные веса, с которыми функция наилучшим образом была приближена к реальным выходам нейронной сети из обучающей выборки:

$$e_k = d_k(n) - y_k(n) \quad (2.53),$$

$$E(n) = \frac{1}{2} e_k^2(n) \rightarrow \min \quad (2.54),$$

то есть в минимизации суммарной ошибки  $E(n)$  выходов сети (где  $e_k$  – ошибка одного выхода сети).

Функция выходов нейронной сети, является универсальным механизмом моделирования широкого спектра задач. Так даже двухслойная нейронная сеть может описать подавляющее большинство функций [3].

Для нейросетей с одним выходным нейроном задача сводится к экстраполяции функции выхода сети. Рассмотрим простую структуру многослойного перцептрона с одним выходом (рис. 2.6).

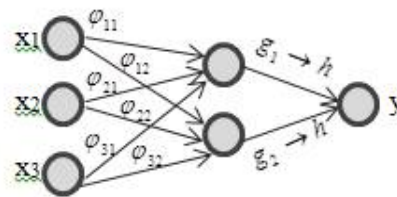


Рисунок 2.6 – Перцептрон с одним скрытым слоем

Функция значения выходного нейрона перцептрона с одним скрытым слоем (рис. 2.6) принимает вид:



$$y = h \sum_{j=1}^2 g_j \left( \sum_{i=1}^3 \varphi_{ij} \cdot x_i \right) \quad (2.55),$$

где  $h$  – функция активации выходного нейрона,  $g_j$  – функции активации нейронов скрытого слоя.

Процесс обучения нейронной сети с одним выходом сводится к нахождению таких весов, которые бы максимально приближали функцию значений выходного нейрона к реальному значению выхода из обучающей выборки:

$$E(n) = e^2 = (d - y)^2 = \left( d - h \sum_{j=1}^2 g_j \left( \sum_{i=1}^3 \varphi_{ij} \cdot x_i \right) \right)^2 \rightarrow \min \quad (2.56).$$

В качестве обучающей выборки возьмем выборку для задачи классификации по 3 классам:

$$d = \begin{cases} 0 \\ 0.5 \\ 1 \end{cases}$$

Рассмотрим процесс обучения для первого класса ( $d=0$ ):  $\left( h \sum_{j=1}^2 g_j \left( \sum_{i=1}^3 \varphi_{ij} \cdot x_i \right) \right)^2 \rightarrow \min$ . В данном случае, задача обучения сводится к минимизации функции выхода нейросети. Значения  $x_i$  нормированные в интервал  $[0,1]$  величины входов из обучающей выборки. Для рассмотрения задачи минимизации ошибки с точки зрения классической оптимизации, необходимо определиться с функциями активации нейронов скрытого и выходного слоев.

В большинстве случаев функциями активации нейронов рекомендуется выбирать функции, которые нормируют значения суммы произведений весов предыдущего слоя в интервалы  $[0,1]$  или  $[-1,1]$ . По сравнению с пороговыми, знаковыми или функциями насыщения непрерывные функции, типа логистической (сигмоиды), гиперболического тангенса, являются более гибкими для задач оптимизации, так как они равномерно сглаживают средневзвешанное значение нейронов предыдущего слоя. Чтобы исключить различия влияний на значение нейрона выходного слоя отдельных нейронов, функции активации всех нейронов должны совпадать.

Анализ начнем с выходного нейрона, для этого введем обозначения:  $t_j = \sum_{i=1}^3 \varphi_{ij} \cdot x_i$ , тогда

$$y^2 = \left( h \sum_{j=1}^2 g_j \left( \sum_{i=1}^3 \varphi_{ij} \cdot x_i \right) \right)^2 = \left( h \sum_{j=1}^2 g_j(t_j) \right)^2, \text{ и если } K = \sum_{j=1}^2 g_j(t_j), \text{ то } y^2 = h^2(K).$$

Возьмем в качестве функции активации выходного нейрона логистическую функцию:

$h(K) = \frac{1}{1 + e^{-K}}$ . Тогда задача оптимизации сводится к отысканию минимума функции

$$\frac{1}{\left(1 + e^{-\sum_{j=1}^2 g_j(t_j)}\right)^2} \rightarrow \min .$$

Функция вида  $\frac{1}{(1 + L)^2}$  принимает наименьшие значения при максимизации аргумента  $L$  (см. рис. 2.7), при положительных значениях аргумента (т.к. в нашем случае аргументом является экспонента, которая принимает только значения больше 0).

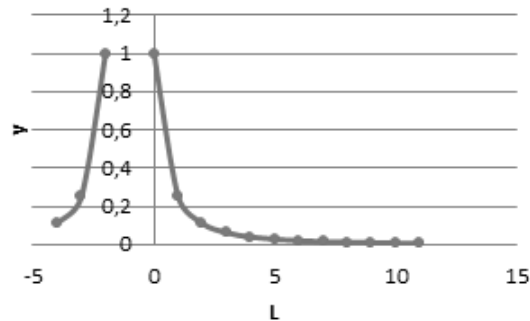


Рисунок 2.7 – График логистической функций активации

Переходим к задаче максимизации экспоненциальной функции от аргумента функции ак-

тивации скрытых нейронов:  $e^{-\sum_{j=1}^2 g_j(t_j)} \rightarrow \max .$

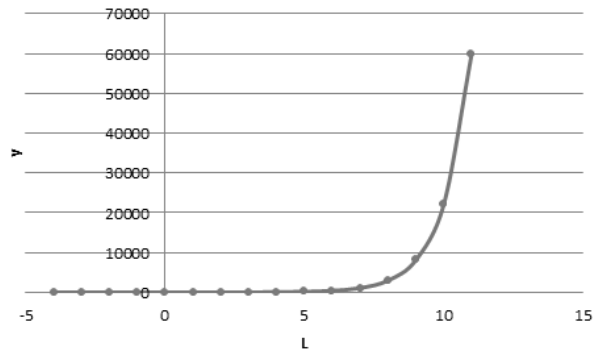


Рисунок 2.8 – График экспоненциальной функций активации

График экспоненциальной функции активации представлен на рисунке 2.8. Функция монотонно возрастающая. Переходим к задаче максимизации ее аргумента:  $-\sum_{j=1}^2 g_j(t_j) \rightarrow \max ,$

$-g_1(t_1) - g_2(t_2) \rightarrow \max \Rightarrow g_1(t_1) + g_2(t_2) \rightarrow \min .$

Для скрытого слоя берем такую же функцию активации, что и для выходного нейрона, логистическую, тогда  $g_1(t_1) + g_2(t_2) \rightarrow \min \Rightarrow \frac{1}{1 + e^{-t_1}} + \frac{1}{1 + e^{-t_2}} \rightarrow \min$ .

За  $t_1$  было взято средневзвешанное значение, попадающее на вход первого нейрона скрытого слоя, за  $t_2$ , средневзвешанное, попадающее на вход второго скрытого нейрона:

$$t_1 = \varphi_{11}x_1 + \varphi_{21}x_2 + \varphi_{31}x_3, \quad t_2 = \varphi_{12}x_1 + \varphi_{22}x_2 + \varphi_{32}x_3.$$

Задача обучения трехслойного перцептрона с тремя входными, двумя промежуточными и одним выходным слоем сводится к задаче безусловной минимизации непрерывной функции 9 переменных (6 весов и 3 значений входных нейронов):

$$\frac{1}{1 + e^{-\varphi_{11}x_1 + \varphi_{21}x_2 + \varphi_{31}x_3}} + \frac{1}{1 + e^{-\varphi_{12}x_1 + \varphi_{22}x_2 + \varphi_{32}x_3}} \rightarrow \min.$$

Задачи безусловной оптимизации состоит в отыскании оптимального плана  $(\varphi_{ij}, x_i)^*$ , и может быть решена с помощью классических методов, таких как различные модификации метода градиентного спуска. Но, если включать переменные  $x_i$  в базис, то будет найдена один из допустимых планов, в котором значения входных нейронов будет фиксированным. А процесс обучения предусматривает нахождение таких весов, которые будут подстраиваться под значения входов нейронов. То есть значения входов сети являются управлениями над системой весов.

Таким образом, можно перейти к задаче оптимального управления, где вектор входных значений нейронной сети является вектором управления, вектор весов является вектором состояния, а минимизируемое выражение критерием качества. На каждом последующем промежутке времени, значения входов нейронов меняются, и полученные на предыдущем шаге веса влияют на веса нового шага.

$$\varphi(\tau + 1) = A\varphi(\tau) + Bx(\tau) \quad (2.57),$$

где  $\tau$  – переменная, характеризующая динамику процесса обучения, момент, за который подаются значения входов сети,  $A$  – матрица влияния весов предыдущего шага,  $B$  – вектор корректировки предыдущего управления.

$$\varphi(\tau) – \text{вектор состояния} \quad (2.58),$$

$$x(\tau) – \text{вектор управления, } x(\tau) \in X : \{x_i \in [0,1], i = 1..3\} \quad (2.59),$$

где  $X$  – множество допустимых управлений (множество отнормированных значений входов нейронной сети).

При рассмотрении процесса управления в динамике, соответственно, критерий качества выражается как сумма минимизируемых выражений:

$$J(\varphi, x) = \sum_{\tau=0}^T \left( \frac{1}{1 + e^{-\sum_{i=1}^3 \varphi_{i1} x_i(t)}} + \frac{1}{1 + e^{-\sum_{i=1}^3 \varphi_{i2} x_i(t)}} \right) \rightarrow \min \quad (2.60).$$

Получена линейная дискретная задача оптимального управления с суммарным функционалом [84], которая является хорошо исследованной и имеет специальные методы оптимизации: принцип максимума Понтрягина и методы динамического программирования (метод Беллмана).

Описанным методом может быть проведено обучение многослойного перцептрона для задач классификации. Структура перцептрона и количество классов будут влиять на размерность конечной задачи, но алгоритм проведенных действий останется неизменным. Основным отличием от стандартного метода обучения алгоритма распространения ошибки, является отсутствие риска переобучения сети, которое часто возникает при обучении в несколько эпох на одном и том же входе сети. Переобучение сети означает, что веса максимально подстраиваются под конкретные значения входных значений сети, и в рабочем режиме такая сеть дает неверные выходы даже при подаче незначительно отличающихся от обучаемой выборки входных данных. В методе, основанном на классической оптимизации и сведению процесса обучения к дискретной задаче оптимального управления, сеть не может переобучиться, поскольку на каждом новом шаге обучения подается новое управление (входной набор значений сети), и на каждом шаге предыдущее управление вносит корректировки в веса. Соответственно состояния (веса) подстраиваются под динамику управлений, а не под конкретные значения входов сети [71].

### 2.3.3 Распознавание кризисных паттернов с использованием нейронных сетей

При анализе финансовых рынков существует возможность широкого применения нейронных сетей не только для классификации ситуаций конкретного финансового инструмента для принятия решений о направлении открытия сделок, но и для других задач, например, задачи распознавания кризисных паттернов. Вопрос о влиянии различных кризисных ситуаций на динамику временного ряда цен финансового инструмента является особенно важным при нестабильной экономической ситуации в стране и в мире. Рассмотрим на примере изменения котировок фьючерсов на нефть сорта Brent [77].

За 2014-2015 годы котировки фьючерсов на нефть сорта Brent претерпели сильные изменения под давлением мощного нисходящего тренда, сформированного нестабильной геополитической и экономической ситуацией в России и в мире в целом.

Каждое внешнеполитическое событие или обострение, экономический коллапс сказывается на котировках Brent падениями цен, прогнозирование которых было невозможно даже с помощью современных адаптивных моделей.

Основными причинами резких спадов цен на нефть с марта 2014 года являются:

- война на юго-востоке Украины;
- противостояние США и России;
- введение и ужесточение экономических санкций ЕС и США против России;
- события, связанные с крупными нефтяными компаниями;
- волнения в мире (террористические акты, народные бунты в различных областях);
- прочие манипуляции по поддержанию экономики России или направленные на недопущение экономического кризиса в Украине.

Исследования цен закрытия дневных котировок фьючерсов Brent в период с 3 марта 2014 по 4 марта 2015 года на ММВБ показывают, что снижения более чем 0,75 рублей за баррель являются реакцией рынка на кризисные условия. Такие снижения цен характеризуются не прогнозируемостью их появления, не зависимостью от трендовой составляющей временного ряда, от индексов технического анализа, а также скорым восстановлением уровня цен и приближением к трендовой линии. Среднее падение цены под психологическим влиянием происходящих в мире событий за один торговый день по сравнению с предыдущим располагается в районе 1,5-2,5 рублей за баррель. Максимальное однодневное падение зафиксировано 27.11.2014 года в 4,71 рубль.

Первый такой спад за анализируемый период произошел 4.03.2014, цена уменьшилась на 2,16 рубля за баррель, основной причиной которого стало введение российских войск на территорию Украины:

«Ведущие мировые издания буквально по минутам информируют о ходе военного вмешательства армии России на территории Украины в Крыму.» (источник: <http://pressorg24.com/politics/novosti-kryma-segodnya-v-zapadnykh-smi-4032014>).

Восстановление в сторону подъема уровня цены произошло уже 6.03.2014.

Аналогичные изменения временного ряда котировок от неожиданного падения до относительного восстановления цены образуют определенную фигуру, представляющую собой последовательность точек, соединенных отрезками.

Выделяются 3 основных вида фигур резких психологических «выпадов» (рисунок 2.9):

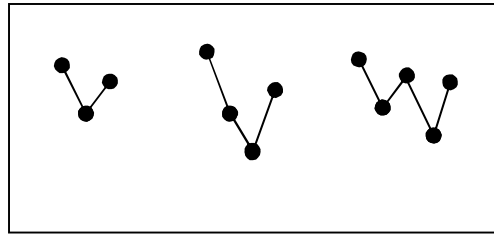


Рисунок 2.9. Фигуры «выпадов» котировок

Фигура а) с рисунка 2.9 является самой простой и характеризуется высокой скоростью восстановления уровня котировок после падения. Фигуры б) и в), как и простая фигура а), начинается со спада цены, но восстановление не происходит столь быстро, в данных случаях сильное негативное влияние продолжает сохраняться на протяжении нескольких дней\периодов в общем случае.

Фигуры типа а) встречаются в исследуемом временном ряду, например, 28.04.2014, 4.06.2014, 2.09.2014, 18.09.2014; б): 1.04.2014, 30.04.2014, 12.11.2014; в): 22.07.2014, 12.08.2014, 20.10.2014 (датой фигур считается первая точка спада). Всего за год выявлено 25 фигур, из них: к типу а) относятся 11, б) – 9, в) – 5.

Все эти фигуры объединяют предпосылки их образования, они являются кризисными образами или паттернами, отражением психологической реакции на внешнеполитические влияния. Таким образом, кризисный паттерн – резкое непредсказуемое снижение уровня котировки, обусловленное неуправляемой психологической реакцией рынка на негативные внешние условия.

«Выпады» в уровне котировок, образующие подобные фигуры, при моделировании временного ряда, говорим мы о задачах прогнозирования или классификации, увеличивают среднюю ошибку по модели. Это ведет к образованию так называемых «тяжелых хвостов», что в свою очередь требует от исследователя дополнительных инструментов по учету или устранению их при моделировании. При элиминации кризисных паттернов временной ряд очищается от излишних шумов, уменьшается дисперсия, временной ряд естественным образом сглаживается, что ведет к упрощению его анализа и моделирования.

Для исключения кризисных паттернов данные образы необходимо распознавать во временном ряду котировок.

Распознавание образов представляет собой информационный процесс, реализуемый некоторым преобразователем информации, имеющим вход и выход. На вход системы подается информация о том, какими признаками обладают предъявляемые объекты. На выходе системы отображается информация о том, к каким классам (образам) отнесены распознаваемые объекты [92].

Целью распознавания является построение алгоритмов отнесения формализованных описаний ситуаций и объектов к соответствующим классам на основе систематических теоретических и экспериментальных исследований.

Выделяют два основных способа представления знаний (Д.А.Поспелов, 1990):

- интенциональное, в виде схемы связей между признаками – фиксируют закономерности и связи, которыми объясняется структура данных.
- экстенциональное, с помощью конкретных объектов или примеров – описывают и фиксируют конкретные объекты из предметной области, реализуются в операциях над объектами, как целостными системами [85].

К интенциональным методам могут быть отнесены методы, основанные на теории статистики, в которой объекты рассматриваются как распределение многомерной случайной величины в пространстве признаков по определенному закону. К таким методам относят методы, основанные на оценке плотностей распределения признаков (Байсовский подход [9]), при этом строится предположение о соответствии нормальному закону распределения и в оценке нуждаются только ограниченное число признаков (средние значения, ковариационные матрицы).

Другим способом выражение закономерностей и связей в структуре данных является предположение о классе решающих функций. При таком подходе задача распознавания сводится к четкой математической постановке, как задача отыскания локального экстремума. Общим для алгоритмов оптимизации функций является метод стохастической аппроксимации [59, 51]. К математической задаче нелинейной оптимизации могут быть отнесены нейросетевые модели, такие как многослойный персептрон. Недостатком градиентных алгоритмов является доказательность их сходимости только при представлении признаков в компактных пространствах. К этой же группе относятся методы эволюционного программирования (генетические алгоритмы, основанные на аналогии с процессами селекции) [64]. К методам распознавания, основанных на работе с признаками и свойствами, могут быть отнесены логические и лингвистические методы [102].

В методах экстенциональной группы каждому объекту придается собственное полноценное диагностическое значение. В отличие от интенциональных методов, где объект раскладывается на составляющие, образующие собой некоторые закономерности, основными операциями данных методов являются определения сходства и различия объектов при классификации и распознавании образов. Самым простым экстенциональным методом является метод сравнения с прототипом. В качестве усредненного значения объекта берется центр геометрической группировки отдельного класса, и задача отнесения нового объекта к какому-либо классу сводится к

вычислению расстояния до прототипа каждого из классов и определению минимального из расстояний (например, метод «ближайшего соседа», рассматриваемый в главе 1).

Основные проблемы экстенциональных методов связаны с трудностями поиска свойств сходства объектов и с анализом структуры экспериментальной информации. Переход от экстенциональных к интенциональным методам может быть произведен, когда в предоставленной информации выявлена определенная структура, определены сходства и различия объектов, тогда можно выявлять закономерности и исследовать эффективность примененных алгоритмов.

При построении алгоритмов распознавания образов обобщенные классы могут задаваться экспертом на основе собственных представлений и дополнительной внешней информации о критериях сходства и различия объектов в контексте конкретной решаемой задачи. В случае распознавания кризисных паттернов в котировках нефтяных фьючерсов обобщенными классами и соответствующими критериями сходства служат представленные фигуры трех типов (рисунки 2.9).

При выявлении определенных фигур в динамике временных рядов ставится задача распознавания образов «с учителем». То есть входные образы, отысканию которых необходимо обучить алгоритм, подаются на входы готовыми в качестве экспертной оценки и не подлежат дополнительному выделению самим обучаемым алгоритмом.

Для задачи распознавания образов «с учителем» могут быть использованы нейронные сети Хопфилда и многослойный персептрон, которые являются наиболее распространенными алгоритмами для классификации и распознавания паттернов среди нейросетевой методологии.

В качестве предобработки и нормировки данных для работы с нейронными сетями временной ряд с ценами рекомендуется перевести в ряд изменений цен (разницей между предыдущим и текущим значением). При такой предобработке обученные сети можно будет использовать при различных уровнях цен на нефть, будь то 60-70 или 110-120 рублей за баррель.

Рассмотрим применимость нейронной сети Хопфилда для задачи распознавания кризисных паттернов в котировках, так как он объединяет достоинства как экстенциональных, так и интенциональных методов распознавания образов.

При использовании сети Хопфилда могут быть заданы исходные образцы в виде векторов изменений за 6-8 дней, так как наибольшее количество точек в выявленных фигурах достигает 5, плюс необходимо обозначить входную, выходную или обе точки в образ. Например, это могут быть вектора:



$$\begin{aligned}
x_1 &= [0,23;-1,27;0,72;-0,34;1,61;-0,79], \\
x_2 &= [-0,28;0,05;0,22;2,39;0,61;-1,44], \\
x_3 &= [0,17;-0,71;0,79;-1,29;1,05;-1], \\
x_4 &= [0,39;0,22;-0,34;-0,07;-0,69;-0,28], \\
x_5 &= [-0,16;0,14;0,64;0,55;0,85;0,11]
\end{aligned}
\tag{2.61}$$

где  $x_1, x_2, x_3$  являются примерами векторов для кризисных паттернов фигур с рисунка 2.9 типов а), б) и в), соответственно, вектор  $x_4$  является примером для нормальной трендовой фигуры восходящего типа, а  $x_5$  фигуры нисходящего тренда. Таким образом, на вход сети может быть подано по несколько векторов-примеров исходных образцов данных пяти типов.

Далее из векторов-примеров составляется симметричная матрица связей для сети Хопфилда:

$$W = \sum_{k=1}^5 \bar{x}_k^T \cdot \bar{x}_k \tag{2.62}.$$

При работе с матрицей весовых коэффициентов сети большой размерности может быть применено условное обнуление диагонали.

Далее новый вектор  $y$  для проверки («испорченный образ»), принадлежность которого к одному из классов необходимо определить, подается на вход сети. «Испорченный образ» преобразуется с помощью матрицы связей:

$$\bar{y}^* = W \cdot \bar{y} \tag{2.63}.$$

Взвешенный вектор  $\bar{y}^*$  подставляется в функцию активации (например, в функцию гиперболического тангенса) и получается выходной вектор:

$$\bar{y}' = f(\bar{y}^*) \tag{2.64}.$$

Выходной вектор сравнивается с входными векторами-образами и в зависимости от необходимой точности, если ошибка между векторами еще велика, то выходной вектор подается снова на вход в качестве «испорченного образца» и процесс (2.63)-(2.64) итерационно повторяется. Сходимость метода определяется размерностью векторов, допустимой максимальной ошибкой, количеством запомненных сетью образцов и емкостью сети.

Выявив с помощью нейронных сетей кризисные паттерны реагирования на внешние пагубные условия, далее для моделирования временного ряда можно основываться на самих изменениях котировок без анализа внешнеполитических и экономических условий, влияющих на направления котировок. Данный подход упростит анализ и моделирование временного ряда котировок и приблизит нас к принципу самоописания временного ряда – динамика временного

ряда любого финансового инструмента полностью включает в себя все условия и причины понижения и повышения котировок.

## 2.4 Алгоритмы с использованием элементов фундаментального анализа

Как уже упоминалось выше (см. глава 1) современные трейдеры и аналитики придерживаются довольно спорной позиции по отношению необходимости и адекватности применения фундаментального анализа, особенно для средне- и краткосрочного анализа финансовых рынков. В конце предыдущего раздела автором работы приведены результаты исследования влияния кризисных событий на дневные котировки фьючерсного рынка и сделана попытка обобщить возникаемые реакции рынка и свести их к ограниченному количеству паттернов, для распознавания которых могут служить инструменты нейросетевой методологии. К тому же аналитики сходятся во мнении, что важные экономические события действительно оказывают влияние на котировки финансовых инструментов, по крайней мере влияют на увеличение волатильности в момент их публикации. Поэтому в общую методику анализа финансового рынка принято решение включить алгоритмы с применением элементов фундаментального анализа, а именно анализа событий экономических календарей и степени их влияния на волатильность котировок.

### 2.4.1 Исследование влияния экономического календаря на волатильность котировок

В современной торговле на финансовых рынках одним из важных инструментов трейдеров являются экономические календари, в которых приведены списки событий, связанных с публикацией различных экономических, финансовых показателей, и выступлений представителей основных финансовых институтов различных стран. В настоящее время сайты брокеров рынка Forex предоставляют в бесплатном доступе экономические календари, примерами таких сайтов могут послужить [https://www.instaforex.com/ru/forex\\_calendar](https://www.instaforex.com/ru/forex_calendar), [http://www.alpari.ru/ru/analytics/calendar\\_fxstreet](http://www.alpari.ru/ru/analytics/calendar_fxstreet), <http://www.fxstreet.ru.com/economic-calendar> [101, 107, 109]. Использование экономического календаря для трейдера является неотъемлемой частью фундаментального анализа рынка. Напомним, что фундаментальный анализ основывается на изучении различных макроэкономических показателей и типичные реакции рынка на публикацию тех или иных данных.

Для каждого финансового инструмента определяется свой список событий из экономического календаря, который оказывают влияние на изменения его цены. В рамках данной работы мы не ставим перед собой цель проанализировать направления и динамику изменений цен. Важным для построения интеллектуальных систем поддержки принятия решений является

учесть влияние данных событий на увеличение волатильности, так как этот критерий влияет на большинство инструментов технического, в том числе эконометрического, анализа рынков.

Рассмотрим влияние событий экономического календаря на изменения волатильности котировок на примере финансового инструмента – валютной пары EURUSD.

Для целей выявления зависимостей между событиями экономического календаря и повышением волатильности котировок проведен анализ изменчивости котировок пары EURUSD на Forex в период с 1.07.2016 по 14.10.2016 на часовом периоде изменений. Данные для исследования отбирались из экономического календаря сайта investing.com [108].

Мерой волатильности в данном исследовании процент от средней дисперсии суммарных средних отклонений максимальной и минимальной цены в час (high и low) от двенадцатичасового экспоненциального скользящего среднего.

Алгоритм получения значения меры волатильности:

1. Экспоненциальное скользящее среднее рассчитывалось по формуле:

$$EMA_t = k \cdot P_t + (1 - k) \cdot EMA_{t-1} \quad (2.65),$$

где  $P_t$  – средняя цена текущего торгового периода (часа),  $EMA_{t-1}$  – значение экспоненциального скользящего среднего, рассчитанного для предыдущего торгового периода и  $k$  – регулирующий коэффициент. Регулирующий коэффициент применялся в размере 0,4.

2. Вычислить отклонения максимальной и минимальной цены в час от экспоненциального скользящего среднего:

$$e_{high_t} = P_{high_t} - EMA_t \quad (2.66),$$

$$e_{low_t} = P_{low_t} - EMA_t \quad (2.67).$$

3. Вычислить дисперсию максимальных и минимальных цен:

$$\sigma_{high}^2 = \frac{\sum e_{high_t}^2}{n} \quad (2.68),$$

$$\sigma_{low}^2 = \frac{\sum e_{low_t}^2}{n} \quad (2.69).$$

7. Вычислить отклонения квадратов отклонений максимальных и минимальных цен от дисперсии и сумму квадратов данных значений.

$$e_{\sigma_{high}^2} = e_{high_t}^2 - \sigma_{high}^2 \quad (2.70),$$

$$e_{\sigma_{low}^2} = e_{low_t}^2 - \sigma_{low}^2 \quad (2.71),$$

$$S_{\sigma^2_t} = e_{\sigma_{high}^2} + e_{\sigma_{low}^2} \quad (2.72).$$

8. Вычислить среднее сумм квадратов отклонений из формулы (2.72) и процент текущей суммы от ее среднего значения.

$$\overline{S_{\sigma^2}} = \frac{\sum S_{\sigma^2_t}}{n} \quad (2.73),$$

$$Pr_{S_{\sigma^2 t}} = \frac{S_{\sigma^2 t} * 100}{S_{\sigma^2}} \quad (2.74).$$

Пример вычисления предложенной меры волатильности для часовых изменений цен в течении дня 01.07.2016 приведен в таблице 2.1.

За период с 1.07.2016 по 14.10.2016 на часовом периоде с 8:00 до 00:00 изменений цен пары EURUSD было выявлено 133 часа с повышенной волатильностью. Под повышенной волатильностью понимаются значения, вычисленная мера волатильности которых принимает значение больше среднего значения данной меры. Средним значением меры волатильности для исследуемого периода было 100.

Из 133 случаев выявленной повышенной волатильности в 50 % (66 случаев) на текущий час приходилось событие из экономического календаря. В подавляющем большинстве остальных 50% случаев событие состоялось в предыдущем или в последующем часе.

Распространение действия события на волатильность на последующий час объясняется реакцией на публикацию показателей или выступлений представителей центральных банков и других главных финансовых учреждений стран. В 16 случаях (в 12%) повышенная волатильность наблюдалась в час публикации экономических показателей и сохранялась через 1 или 2 часа после совершения событий (через 2 часа выявлено 4 случая). Анализ подобных случаев показал, что сохранение высокой волатильности на 2 часа наблюдались после публикации нескольких экономических показателей (от 3 до 7 показателей, среди которых максимум одно важное событие или несколько событий средней важности, например USD: Число выданных разрешений на строительство (июнь), USD: Индекс ожиданий потребителей от Мичиганского университета (июль)) в один час или после средней важности выступлений экономических представителей (например, EUR: Ежемесячный отчет ЕЦБ, USD: Речь члена FOMC Фишера). Сохранение высокой волатильности на 1 час наблюдалось и при публикации показателей низкой важности, например, USD: Аукцион по размещению 3-месячных казначейских векселей, USD: Число буровых установок от Baker Hughes, USD: Объем товарных запасов на оптовых складах (м/м).

Примеры событий с сохранением волатильности на несколько часов со значениями меры волатильности в час совершения события и последующие часы приведены в таблице 2.2.

Таблица 2.1 Вычисление меры волатильности

Час	open	High	Low	close	$P_t$	$EMA_{t-1}$	$e_{high}^2$	$e_{low}^2$	$e_{\sigma_{high}}^2$	$e_{\sigma_{low}}^2$	$S_{\sigma^2}$	$Pr_{S_{\sigma^2}}$
1:00	1,1107	1,1109	1,10889	1,10889	1,109845	1,109845	0,0000011130	0,0000009120	3,08167E-14	1,2667E-13	1,57487E-13	0,824007
2:00	1,11023	1,11085	1,10924	1,1099	1,110055	1,109929	0,0000008482	0,0000004747	1,93891E-13	6,29184E-13	8,23075E-13	4,306524
3:00	1,10996	1,11081	1,10953	1,1102	1,110125	1,1100074	0,0000006442	0,0000002279	4,15258E-13	1,08164E-12	1,4969E-12	7,832147
4:00	1,1103	1,11165	1,11	1,11125	1,1108	1,11032444	0,0000017571	0,0000001053	2,19527E-13	1,3518E-12	1,57133E-12	8,221572
5:00	1,11125	1,11167	1,11013	1,11036	1,1108525	1,11053566	0,0000012867	0,0000001646	3,43671E-18	1,21742E-12	1,21743E-12	6,369861
6:00	1,11039	1,11089	1,10981	1,11033	1,110355	1,1104634	0,0000001820	0,0000004269	1,22453E-12	7,07285E-13	1,93181E-12	10,107693
7:00	1,11034	1,11054	1,10879	1,109	1,1096675	1,11014504	0,0000001560	0,0000018361	1,28273E-12	3,2285E-13	1,60558E-12	8,400785
8:00	1,10902	1,11051	1,10854	1,10996	1,1095075	1,10989002	0,0000003844	0,0000018226	8,1758E-13	3,07616E-13	1,1252E-12	5,887286
9:00	1,10998	1,11028	1,10776	1,10802	1,10901	1,10953801	0,0000005505	0,0000031613	5,44687E-13	3,58497E-12	4,12966E-12	21,607338
10:00	1,10799	1,10949	1,107	1,10877	1,1083125	1,10904781	0,0000001955	0,0000041935	1,19473E-12	8,55906E-12	9,75379E-12	51,034147
11:00	1,10885	1,11169	1,10836	1,11071	1,1099025	1,10938969	0,0000052914	0,0000010603	1,6023E-11	4,31313E-14	1,60662E-11	84,061902
12:00	1,11062	1,11149	1,10925	1,11055	1,1104775	1,10982481	0,0000027729	0,0000003304	2,20309E-12	8,78952E-13	3,08205E-12	16,125990
13:00	1,11054	1,11139	1,1099	1,11062	1,1106125	1,11013989	0,0000015628	0,0000000575	7,51919E-14	1,46504E-12	1,54023E-12	8,058833
14:00	1,11062	1,11244	1,11	1,1123	1,11134	1,11061993	0,0000033126	0,0000003843	4,09688E-12	7,80778E-13	4,87766E-12	25,521067
15:00	1,11228	1,11375	1,11134	1,11194	1,1123275	1,11130296	0,0000059880	0,0000000014	2,20847E-11	1,60417E-12	2,36889E-11	123,945800
16:00	1,11194	1,11442	1,1117	1,11321	1,1128175	1,11190878	0,0000063062	0,0000000436	2,51771E-11	1,49902E-12	2,66761E-11	139,575622
17:00	1,1132	1,11699	1,1129	1,11527	1,11459	1,11298127	0,0000160700	0,0000000066	2,18489E-10	1,59095E-12	2,2008E-10	1151,511413

Примечание: Пример вычисления приведены с учетом средних значений за весь период с 1.07.2016 по 14.10.2016:

$$\sigma_{high}^2 = 0,0000012886,$$

$$\sigma_{low}^2 = 0,0000012679,$$

$$S_{\sigma^2} = 0,000000000019112.$$

Таблица 2.2 Примеры событий с последующим сохранением повышенной волатильности на несколько часов

Дата	Время	През	События
19.07.2016	15:00	559,6281858	USD: Число выданных разрешений на строительство (июнь) USD: Число выданных разрешений на строительство (м/м) (июнь) USD: Объем строительства новых домов (м/м) (июнь) USD: Объем строительства новых домов (июнь) USD: Индекс розничных продаж Redbook (г/г) USD: Индекс розничных продаж Redbook (м/м)
19.07.2016	16:00	398,4246936	
19.07.2016	17:00	160,3258686	
29.07.2016	17:00	766,281372	USD: Индекс потребительских инфляционных ожиданий в США на 5 лет вперед от Мичиганского университета (июль) USD: Индекс ожиданий потребителей от Мичиганского университета (июль) USD: Индекс настроения потребителей от Мичиганского университета (июль) USD: Индекс настроения потребителей от Мичиганского университета (июль) USD: Ожидаемая инфляция от Мичиганского университета (июль) USD: Недельный индекс ECRI (аннуализированный) (еженедельный) USD: Недельный индекс ECRI
29.07.2016	18:00	639,7748744	
29.07.2016	19:00	909,6845453	
08.09.2016	18:00	174,7730719	USD: Запасы сырой нефти USD: Импорт нефти USD: Данные по избыточным запасам нефти в Кушинге (штат Оклахома) USD: Объем производства дистиллятного топлива USD: Недельные запасы дистиллятов по данным EIA USD: Объем производства бензина USD: Данные по запасам мазута USD: Запасы бензина
08.09.2016	19:00	184,447402	
08.09.2016	20:00	251,8038197	

Примеры событий с сохранением волатильности на час приведены в таблице 2.3.

Таблица 2.3 Примеры событий с последующим сохранением повышенной волатильности

час

Дата	Время	$\overline{Pr_{\frac{1}{2} \sigma^2}}$	События
21.07.2016	17:00	478,0456661	USD: Индекс опережающих экономических индикаторов от СВ (м/м) (июнь) USD: Продажи на вторичном рынке жилья (м/м) (июнь) USD: Продажи на вторичном рынке жилья (июнь) USD: Объем запасов природного газа
21.07.2016	18:00	794,8995569	
26.08.2016	22:00	1791,315986	USD: Количество чистых спекулятивных позиций по меди от CFTC USD: Количество чистых спекулятивных позиций по сырой нефти от CFTC USD: Количество чистых спекулятивных позиций по золоту от CFTC USD: Количество чистых спекулятивных позиций по S&P 500 от CFTC USD: Количество чистых спекулятивных позиций по серебру от CFTC EUR: Количество чистых спекулятивных позиций по EUR от CFTC
26.08.2016	23:00	500,7156233	
20.09.2016	9:00	14,782898	EUR: Индекс цен производителей (PPI) Германии (г/г) (авг) EUR: Индекс цен производителей (PPI) Германии (м/м) (авг)
20.09.2016	10:00	109,2196811	

Если в один час совершается сразу большое количество событий (в пределах от 5 до 16, среди которых одно или несколько важных событий), или результаты данных отличаются от ожиданий рынка, или это слабо предсказуемые события (например, если это речь представителей или решения по процентной ставке), в таких ситуациях рынок замирает в момент совершения событий и ему требуется некоторое время, чтобы обработать результаты опубликованных данных. Соответственно, такие события дают о себе знать повышением волатильности в последующие часы. Количество выявленных в ходе анализа подобных паттернов равно 26. Из них только 6 случаев с событиями низкой важности, такими как USD: Индекс делового оптимизма малого бизнеса NFIB, EUR: Аукцион по размещению 12-месячных векселей Италии (BOT), USD: Недельный индекс ECRI, после которых эффект повышения волатильности распространяется на 1 час. Остальные случаи с повышением волатильности наблюдаются на фоне событий средней и высокой важности, например, USD: Публикация протоколов FOMC, USD: Выступление председателя ФРС, USD: Базовый индекс потребительских цен (ИПЦ) (м/м), эффект после

которых сохраняется 2 или, в некоторых случаях, даже 3 часа. Особое место среди важных экономических событий, публикация результатов которых способствует последующей высокой волатильности в течении 2 и более часов, занимают показатели, связанные с запасами нефти в США, такие как USD: Запасы сырой нефти, USD: Импорт нефти, USD: Данные по избыточным запасам нефти в Кушинге (штат Оклахома) и т.д., которые обычно публикуются совместно.

Примеры событий с последующим пролонгированным повышением волатильности со значениями меры волатильности в час совершения события и последующие часы приведены в таблице 2.4.

Таблица 2.4 Примеры событий с последующим повышении волатильности

Дата	Время	$\overline{Pr_{USD}}$	События
03.08.2016	17:00	50,34892363	USD: Число выданных разрешений на строительство (июнь) USD: Число выданных разрешений на строительство (м/м) (июнь) USD: Объем строительства новых домов (м/м) (июнь) USD: Объем строительства новых домов (июнь) USD: Индекс розничных продаж Redbook (г/г) USD: Индекс розничных продаж Redbook (м/м)
03.08.2016	18:00	164,8399572	
03.08.2016	21:00	164,8399572	
27.07.2016	21:00	7,297410493	USD: Заявление FOMC USD: Решение по процентной ставке ФРС
27.07.2016	22:00	818,1026849	
27.07.2016	23:00	3649,152716	
05.08.2016	15:00	11,50764732	USD: Средняя почасовая заработная плата (м/м) (июль) USD: Средняя продолжительность рабочей недели (июль) USD: Количество служащих государственного сектора (июль) USD: Количество рабочих мест в обрабатывающей промышленности (июль) USD: Изменение числа занятых в несельскохозяйственном секторе (июль) USD: Доля экономически активного населения (июль) USD: Изменение числа занятых в частном несельскохозяйственном секторе (июль) USD: Сальдо торгового баланса (июнь) USD: Уровень безработицы U6 (июль)
05.08.2016	16:00	4679,937818	
05.08.2016	17:00	8051,041375	



14.09.2016	17:00	3,56244751	USD: Запасы сырой нефти USD: Импорт нефти USD: Данные по избыточным запасам нефти в Кушинге (штат Оклахома) USD: Объем производства дистиллятного топлива USD: Недельные запасы дистиллятов по данным EIA USD: Объем производства бензина USD: Данные по запасам мазута USD: Запасы бензина
14.09.2016	18:00	221,2162589	
14.09.2016	19:00	569,9534042	
14.09.2016	20:00	152,2660759	

В некоторых случаях (всего 2 случая, 1,5%) повышенная волатильность наблюдалась перед публикацией данных по экономическим показателям (примеры приведены в таблице 2.5). И всего в 7 случаях (в 5%) повышенная волатильность не могла быть объяснена исследуемыми событиями из экономического календаря. Соответственно, данные повышения волатильности могли быть связаны с экономическими событиями взаимосвязанных финансовых инструментов (например, GBPUSD), экономические события которых не анализировались в рамках данного исследования.

Таблица 2.5 Примеры повышения волатильности перед событиями

Дата	Время	$Pr_{USD}$	События
02.08.2016	11:00	208,6411287	
02.08.2016	12:00	9,83631483	EUR: Индекс цен производителей (PPI) (м/м) (июнь)
16.08.2016	10:00	1154,669555	
16.08.2016	11:00	1970,780786	EUR: Аукцион по размещению 12-месячных казначейских векселей Испании

Наблюдались случаи, когда повышение волатильности проявлялось в час, когда не было событий, но они совершались в предыдущий и последующий час. Такие случаи не включены в выборки описанных выше событий, так как взаимовлияние предшествующих и последующих событий не позволяют проанализировать степень их влияния на волатильность по отдельности. Примеры таких ситуаций приведены в таблице 2.6.

Таблица 2.6 Примеры повышения волатильности между событиями

Дата	Время	$Pr_{USD}$	События
15.07.2016	18:00	749,5139734	USD: Индекс потребительских цен (ИПЦ) от ФРБ Кливленда)
15.07.2016	19:00	377,1337408	

15.07.2016	20:00	110,9923372	USD: Число буровых установок от Baker Hughes
13.09.2016	21:00	23,40195101	USD: Отчет об исполнении федерального бюджета США (авг)
13.09.2016	22:00	150,3712216	
13.09.2016	23:00	26,1085906	USD: Недельные запасы сырой нефти по данным Американского института нефти (API)

Качественный анализ событий из экономического календаря, особо сильно повлиявшие на повышение волатильности пары EURUSD показал, что к таким событиям относятся индексы США по запасам нефти, продажам, строительства, по странам Евросоюза и США индексы цен, доверия потребителей и уровню безработицы, то есть ключевые индексы, показывающие экономическое и финансовое состояние страны, а также выступления представителей ФРС, FOMC и ЕЦБ. Если сравнивать влияние показателей стран Евросоюза и США, то, данные США более значительно влияют на повышение и сохранение повышенной волатильности. Разбивку более распространенных событий, оказавших влияние на повышенную волатильность валютной пары EURUSD за исследуемый период приведена в таблице 2.7.

Таблица 2.7. Распространенные события, влияющие на повышение волатильности EURUSD

Основное событие	Сопровождающие события	Кол-во	Примечание
Речь членов FOMC		13	Члены FOMC: Тарулло, Джорджа, Уильямса, Местера, Лакера, Каплана, Кашкари, Брейнарда, Фишера, Булларда, Местера
USD: Запасы сырой нефти	USD: Импорт нефти, USD: Данные по избыточным запасам нефти в Кушинге (штат Оклахома)	7	
USD: Индекс ожиданий потребителей от Мичиганского университета	USD: Индекс настроения потребителей от Мичиганского университета	6	
USD: Уровень безработицы	USD: Средняя почасовая заработная плата (м/м), USD: Средняя продолжительность рабочей недели	4	
USD: Индекс цен производителей (PPI) (м/м)	USD: Индекс цен производителей (PPI) (г/г), USD: Базовый индекс цен производителей (PPI) (м/м), USD: Базовый индекс цен производителей (PPI) (г/г)	3	

USD: Число первичных заявок на получение пособий по безработице	USD: Общее число лиц, получающих пособия по безработице, USD: Среднее число заявок на пособие по безработице за 4 недели	3	
EUR: Индекс цен производителей (PPI) (м/м)	EUR: Индекс цен производителей (PPI) (г/г)	3	Встречается 1 случай с повышенной волатильностью в предыдущий час, связанный с ожиданием публикации данных
EUR: Индекс цен производителей (PPI) Германии (м/м)	EUR: Индекс цен производителей (PPI) Германии (г/г)	2	
USD: Число выданных разрешений на строительство	Объём строительства новых домов (м/м)	2	
USD: Индекс деловой активности в производственном секторе (PMI) от ISM	USD: Индекс цен в производственном секторе от ISM	2	
USD: Базовый индекс розничных продаж	USD: Объём розничных продаж (м/м)	2	
USD: Индекс розничных продаж Redbook (м/м)	Индекс розничных продаж Redbook (г/г)	2	
USD: Решение по процентной ставке ФРС	USD: Заявление FOMC	2	

Показатели и выступления представителей США оказывают большее влияние, но и среди событий Евросоюза встречаются хоть и не столь важные, но события, оказывающие влияние на повышение волатильности данной валютной пары. Так, например, в выборке встречаются, события, связанные с ЕЦБ, такие как EUR: Речь представителя ЕЦБ Новотны, EUR: Выступление главы ЕЦБ Драги, Ежемесячный отчет ЕЦБ.

Довольно распространенным показателем, встречающийся в выборке событий, способствующих повышению волатильности пары EURUSD, является USD: Недельный индекс ECR1 (вместе с USD: Недельный индекс ECR1 (аннуализированный) (еженедельный)). Количество случаев повышения волатильности, связанных с присутствием среди событий данного индекса равно 8. Но поскольку данный индекс в подавляющем большинстве (7 из 8) встречается с сильно влияющими на волатильность индексами, такими как USD: Индекс ожиданий потребителей от Мичиганского университета, USD: Индекс цен в производственном секторе от ISM, USD: Речь члена FOMC, сложно выделить степень его единичного влияния.

Дополнительно стоит отметить, что среди случаев повышения волатильности, связанной с событием USD: Речь членов FOMC встречается 4 с предварительным повышением волатильно-

сти, но они не включены в выборку с повышением волатильности перед публикацией, так как каждый из этих случаев сопровождался публикацией за 2-3 часа других важных показателей, соответственно, повышение волатильности могло быть связано как с предыдущими событиями, так и с ожиданием последующих событий.

Как уже упоминалось выше, происходящие события оказывают различное влияние: некоторые события вызывают рост волатильности в час их публикации, некоторые в следующий час, некоторые оказывают пролонгированное действие на повышение волатильности. Диаграмма самых распространенных событий за исследуемый период и их распределение по влиянию на повышение волатильности в текущий и последующие часы представлена на рисунке 2.10.

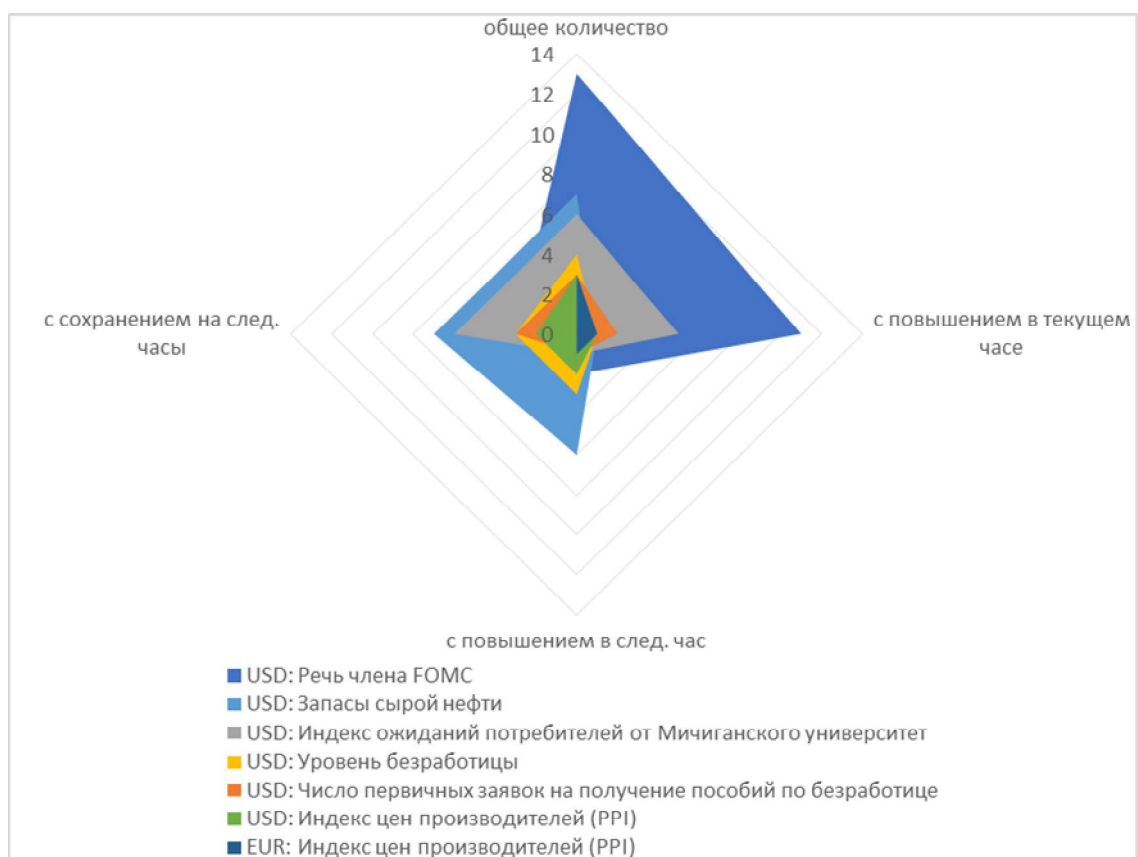


Рисунок 2.10. Диаграмма распределения влияния на волатильность основных событий

Обобщая результаты проведенного исследования, можно утверждать, что события из экономического календаря оказывают значительное влияние на повышение волатильность финансового инструмента. Более 94% случаев повышения волатильности вызваны реакцией на публикацию экономических показателей, индексов или выступлений представителей основных национальных финансовых институтов [69]. При совпадении времени совершения большого количества событий, среди которых встречаются важные, волатильность увеличивается с небольшой задержкой и имеет тенденцию сохраняться на некоторое количество периодов вперед. При публикации показателей средней важности реакция на них следует в час обнаружения

информации и может сохраняться в следующий час. Естественно, для каждого финансового инструмента необходимо определить перечень событий, влияющих на изменения его котировок. По крайней мере, список событий стоит ограничить странами, от которых зависит данный финансовый инструмент. В целом для валютных пар – это страны выпускающие базовую и котируемую валюты, для акций – это страны, в которых находятся эмитенты данных ценных бумаг, для сырьевых фьючерсов – страны, разрабатывающие сырьевые месторождения, основные страны экспортеры и импортеры данного сырья.

#### 2.4.2 Алгоритм анализа повышений волатильности при совершении экономических событий

Как показано в предыдущем разделе, события экономического календаря оказывают влияние на повышение волатильности в временных рядах котировок финансовых инструментов. Для включения в общую систему поддержки принятия решений данной зависимости необходимо разработать алгоритм данного анализа.

Для этого рассмотрим возможные ситуации с изменением волатильности при отсутствии или присутствии событий в экономическом календаре.

Предположим, мы анализируем временной ряд котировок финансового инструмента  $X_t$ . Задана желаемая прибыль  $Pr$ , относительно которой мы обучаем некоторую систему, когда нужно покупать финансовый инструмент, а когда продавать.

Обучение системы происходит на основе исторических данных, если в предыдущие 10 периодов процент максимального повышения от  $Pr$  был больше процента минимального понижения от  $Pr$ , то системе дается обучающий ответ, что финансовый инструмент необходимо продавать, иначе покупать. Но в некоторых случаях, если системе будет дан ответ, но он был достигнут в период, на который приходится экономическое событие, а значит максимальное или минимальное значение (в зависимости от ответа) обусловлено повышением волатильности, и оно может не повториться в выборке, предоставленной системе при ее функционировании в случае отсутствия аналогичной важности и степени влияния экономического события. Таким образом, качество обучение системы может быть пониженным.

При обучении такой системы необходимо учитывать, что ожидаемая прибыль может быть получена в большей или меньшей степени в зависимости от условий отсутствия или п экономических событий в выборке, используемой при обучении, и в выборке, используемой системой при ее функционировании для выдачи необходимой рекомендации.

Возможны 3 случая зависимости присутствия или отсутствия экономических событий в обучающей и реальной выборке (под обычной волатильностью будем понимать изменения вре-

менного ряда в отсутствии событий экономического календаря, при их присутствии будем говорить о повышенной волатильности):

1. В обучающем примере обычная волатильность и в анализируемом системой примере сохранена обычная волатильность. В данном случае система, обученная на примере с обычной волатильностью будет давать адекватный ответ на реальный пример в режиме своего функционирования без влияния экономических событий. Корректировка процесса обучения при таких условиях не требуется.

2. В обучающем примере повышенная волатильность, в анализируемом системой примере обычная волатильность. В данном случае система, обученная на примере с повышенной волатильностью, будет являться «сверхпозитивной» и ее ожидания повышенных изменений котировок, на которых она обучалась, будут распространяться и на реальные примеры в режиме ее функционирования. Тогда система при анализе примера без повышения волатильности будет давать неадекватный ответ: если она обучилась, что достигается прибыль  $Pr$ , то при функционировании достижения данной прибыли может не быть, хотя система выдает рекомендацию на основе своего обучения. В такой ситуации требуется либо при обучении системы увеличивать ожидаемую прибыль, тогда она будет составлять  $kPr$ , где  $k$  – коэффициент увеличения ожидаемой прибыли при обучении в условиях повышенной волатильности, либо в ответе системы в режиме функционирования предупреждать, что  $Pr$  может быть недостижима.

3. В обучающем примере обычная волатильность, в анализируемом системой примере повышенная волатильность. Получается, при обучении система правильным образом научится распознавать паттерны достижения ожидаемой прибыли и давать на их основе рекомендации по покупке или продаже финансового инструмента. Анализируя реальный пример с повышенной волатильностью система выдаст верный ответ, но здесь можно также дать рекомендацию о возможности увеличения прибыли от ожидаемой, так как при функционировании данной системы в условиях совершения событий экономического календаря при повышенной волатильности мы можем получить прибыль большую, чем была ожидаемая прибыль  $Pr$ .

Таким образом самой рискованной ситуацией из перечисленных является случай 2, когда обучение проводится при повышенной волатильности. Из исследования, приведенного в предыдущем разделе, мы убедились, что совершение событий экономического календаря влияют на повышение волатильности (приведенной в разделе меры волатильности). Получается, при обучении системы на финансовом временном ряду для учета повышений волатильности, необходимо рассчитать меру волатильности по формулам (2.65)-(2.74). Для этого необходимы значения цен открытия, закрытия, минимальные и максимальные цены за период финансового инструмента.

При обучении системы если значение меры волатильности получается выше обычного:  $\frac{Pr_{t+1}}{Pr_t} > 100$ , то обучение проводим не по обычной ожидаемой прибыли, а по скорректированной:  $kPr$ , где  $k$  – коэффициент повышения волатильности. Приведем алгоритм определения коэффициента повышения волатильности [73].

Вычислить среднее значение квадрата отклонений от максимальных цен периода (*high*) и минимальных цен периода (*low*) для всех случаев в обучающей выборке с повышенной волатильностью.

$$\sigma_{highVolat}^2 = \frac{\sum e_{highVolat}^2}{n_{Volat}} \quad (2.75),$$

$$\sigma_{lowVolat}^2 = \frac{\sum e_{lowVolat}^2}{n_{Volat}} \quad (2.76),$$

где  $n_{Volat}$  – количество примеров в обучающей выборке с повышенной волатильностью.

Вычислить среднее значение квадрата отклонений от *high* и *low* для всех случаев в обучающей выборке с нормальной волатильностью.

$$\sigma_{highNorm}^2 = \frac{\sum e_{highNorm}^2}{n_{Norm}} \quad (2.77),$$

$$\sigma_{lowNorm}^2 = \frac{\sum e_{lowNorm}^2}{n_{Norm}} \quad (2.78),$$

где  $n_{Norm}$  – количество примеров в обучающей выборке с обычной волатильностью.

Вычислить процент среднего значения квадрата отклонений от *high* для случаев с обычной волатильностью от среднего значения квадрата отклонений от *high* для случаев с повышенной волатильностью.

$$k_{high} = \frac{\sigma_{highNorm}^2 \cdot 100}{\sigma_{highVolat}^2} \quad (2.79),$$

Вычислить процент среднего значения квадрата отклонений от *low* для случаев с обычной волатильностью от среднего значения квадрата отклонений от *low* для случаев с повышенной волатильностью.

$$k_{low} = \frac{\sigma_{lowNorm}^2 \cdot 100}{\sigma_{lowVolat}^2} \quad (2.80),$$

Вычислить среднее значение между процентами отклонений обычной и повышенной волатильности для *high* и *low*.

$$k = 1 + \frac{k_{high} + k_{low}}{2} / 100 \quad (2.81).$$

Таким образом  $k$  коэффициент повышения волатильности в рассматриваемом алгоритме представляет собой процент квадратов отклонений цен от средних значений (ЭСС12) при обычной волатильности от квадратов отклонений цен от средних значений при повышенной волатильности.

## 2.5 Методы и алгоритмы нейросетевой классификации рыночных ситуаций

В связи с высокой актуальностью инвестирования денежных средств в торговлю на рынках акций, фьючерсов, опционов, валютных рынках, исследования в сфере анализа и прогнозирования динамики цен финансовых инструментов с применением нейронных сетей вызывают живой интерес. В последние годы в университетах по всей России появляются все новые работы по данной тематике [2, 1, 5, 12, 48, 49, 63, 91]. Данные исследования направлены на разработку информационных систем на основе нейронных сетей для автоматизации принятия инвестиционных решений, но большинство из них имеет экспериментальный характер. Нейронная сеть используется в них в качестве «черного ящика», на вход которого подаются имеющиеся данные, а на выходе получается требуемый ответ. Авторы работ не интересуют принципы построения и моделирования нейронной сети, а чаще они экспериментальным путем исследуют эффективность случайно подобранных, необоснованных архитектур нейросетей [5, 1, 91]. Так, например, в работе Мицеля и Ефремова [63] количество входов и слоев нейросети предоставляется задавать самому пользователю, что, бесспорно, не имеет никакого отношения к научному обоснованию используемой для прогнозирования архитектуры нейронной сети. Другие исследования ориентированы более на разработку экономико-математического аппарата, увлекаясь математическими методами, авторы теряют фокус практической полезности подобных разработок [5, 12, 48]. Более подробное описание выбора критериев нейронной сети см. в [70].

Автором данного диссертационного исследования сделана попытка обоснования выбора архитектуры нейронной сети на основе качественного и корреляционного анализа факторов модели и теории нейронных сетей, основные положения которой структурированы и обобщены Галушкиным Александром Иванович [115, 21].

Структура нейронной сети в качестве ядра интеллектуальной системы поддержки принятия решений выбрана по следующим причинам:

1. Она представляет возможность массового параллельного синхронного выполнения большого числа операций, состоящих в свою очередь из простейших операций.
2. Реализует достаточно сложное и гибкое функциональное преобразование входного пространства состояний в выходное.



3. Допускает аналитическое описание преобразования входного пространства состояний в выходное.

4. Позволяет организовать контролируемый процесс настройки коэффициентов сети в адаптивном режиме.

Пусть имеется  $\Psi$  – формальная постановка задачи принятия инвестиционного решения.  $\Psi$  включает множество исходных данных  $I$ , в том числе предыдущие изменения цен финансового инструмента, значений технических индикаторов и прогнозной модели и множество объектов, подлежащих определению  $O$ . Основой разработки нейронного алгоритма решения задачи является системный подход, при котором процесс решения задачи представляется как функционирование во времени динамической системы. На вход которой подается множество  $I$ , а на выходе снимается множество  $O$  (объекты, подлежащие определению и получившие свои значения – решение о покупке-продаже финансового инструмента в текущей ситуации).

Для получения нейросетевой динамической системы, решающей исходную задачу, необходимо:

- 1) Определить объект, выступающий в роли входного сигнала нейросети. В данном исследовании таким объектом выступают последние изменения цен финансового инструмента, значений по прогнозной модели и технических индикаторов.
- 2) Определить объект, выступающий в роли выходного сигнала нейросети. Это может быть само решение или некоторая его характеристика. В нашем случае это будет само управленческое решение в виде рекомендации для трейдера (в общем случае пользователя системы) на покупку или продажу конкретного финансового инструмента в текущей рыночной ситуации.
- 3) Определить требуемый выходной сигнал сети. Им будет дискретное величина, принимающая значение: от 1 до 0,45 – покупка, от 0,45 до 0,55 – ожидание, от 0 до 0,45 – продажа.
- 4) Определить структуру нейросети:
  - а) число слоев;
  - б) связи между слоями;
  - в) объекты, являющиеся весовыми коэффициентами.
- 5) Определить функцию ошибки системы, то есть функцию, характеризующую отклонение желаемого выходного сигнала от реального выходного сигнала нейросети.
- 6) Определить критерии качества системы и функционал ее оптимизации, зависящий от ошибки.
- 7) Определить значения весовых коэффициентов – в зависимости от задач это можно сделать различными способами:

- а) аналитически, непосредственно из постановки задачи;
- б) с помощью некоторых численных методов;
- в) применив процедуру настройки коэффициентов нейронной сети.

Решение задачи с помощью нейросетевого алгоритма заключается в применении построенной вычислительной процедуры с конкретными значениями числовых данных.

Процесс решения включает [115, стр.26]:

1. Получение конкретной структуры нейронной сети, соответствующей применяемому алгоритму;
2. Нахождение значений весовых коэффициентов;
3. Генерация начальных приближений параметров, если необходимо;
4. Передачу всех численных значений в нейросеть и ее запуск;
5. Функционирование сети в соответствии с режимом: за один шаг или фиксированное число шагов, либо за переменное число, зависящее от требуемой точности.
6. Получение решения.

Практически все известные подходы к проектированию нейронных сетей связаны в основном с выбором и анализом некоторых частных видов структур с известными свойствами. Например, сети Хопфилда, многослойный перцептрон, самоорганизующиеся карты Кохонена, и некоторых режимов их работы. Использование нейросетей сводится к применению этих структур для решения классов адекватных им задач, при изменении или выборе параметров структур. Так для задачи классификации рыночных ситуаций наиболее подходящей является структура многослойного перцептрона, поскольку такая структура в большей степени пригодна для задач классификации с учителем и довольно широко применяется специалистами нейросетевого анализа для распознавания образов, классификации и кластеризации различных по виду и сложности объектов.

Важным критерием эффективности работы нейросети является выбор информативных признаков. К сожалению, к настоящему времени не существует единых методов выбора данных признаков и любой метод является субъективным и прямо относится к конкретной нейронной сети. Выбор признаков следует проводить после окончания процедуры синтеза и исследования динамики нейросети [115, стр.78].

### 2.5.1 Проектирование архитектуры нейронной сети

Задача синтеза трехслойной нейронной сети, какой является многослойный перцептрон Розенблатта, выбранный в качестве основного при моделировании нейронной сети, при заданном числе нейронов в первом и третьем слое сводится к минимизации числа нейронов второго слоя и настройке коэффициентов нейронной сети.

Задачи оптимизации нейросетевой структуры:

- Задано общее число нейронов сети. Необходимо найти число слоев и распределение нейронов между слоями, максимизирующее  $\Psi$  (максимальное число областей, на которое можно разбить пространство признаков размерности  $NH$  гиперплоскостями).
- Найти количество слоев и распределение нейронов по слоям при заданном  $\Psi$  (максимальном числе областей, на которое можно разбить  $H$  гиперплоскостями пространство признаков размерности  $N$ ), минимизирующую количество нейронов в сети.

Для заранее заданного числа слоев  $W$  (равного 3: входной, промежуточный, выходной) и при ограничениях на количество нейронов  $H$  (не более 50) в сети найдем оптимальную по верхней оценке структуру. Это можно записать следующим образом:

$$\left. \begin{aligned} \max_{W \leq H} \Psi_{1[W]}^{opt} \\ \Psi_{1[W]}^{opt} = \sum_{j=1}^W H_j = H \end{aligned} \right\} \quad (2.82).$$

Из очевидного неравенства следует:

$$\left( \frac{H}{W} + 1 \right)^W < \left( \frac{H}{W+1} + 1 \right)^{W+1} \quad (2.83).$$

$$\Psi_{1[W]}^{opt} = \left( \frac{H}{W} + 1 \right)^W \quad (2.84).$$

Следует, что число областей с ростом числа слоев монотонно возрастает. Отсюда получается, что оптимальной в данном случае является  $H$ -слойная ( $H=3$  – для трехслойного многослойного перцептрона) нейронная сеть с одним элементом в каждом слое, для которой

$$\Psi_{1[W]}^{opt} = \left( \frac{H}{H} + 1 \right)^H = 2^H \text{ есть точная верхняя оценка.}$$

Для трехслойной сети  $\Psi=2,27^3=11,697$ .

Для многомерного варианта сети, оптимального по верхней оценке, на основании

$$\Psi_{N[j]} = \prod_{i=1}^j \Psi_{NH_j} \quad (2.85),$$

так же, как в одномерном случае, можно записать:

$$\left. \begin{aligned} \psi_{N[W]}^{opt} = & \left. \begin{aligned} \max_{W \leq H} \max_{H_1 \dots H_H} \prod_{j=1}^W \psi_{NH_j} \\ \sum_{j=1}^W H_j = H \end{aligned} \right\} \end{aligned} \quad (2.86).$$

Учитывая

$$\psi_{NH_1} = 2^{H_1} \text{ при } H_1 \leq N \quad (2.87),$$

$$\psi_{NH_1} < 2^{H_1} \text{ при } H_1 > N \quad (2.88)$$

Следует, что условиям оптимальности отвечает целый класс структур, именно все структуры, для которых  $H_j=N$  выполняется:

$$\psi_N^{opt} = 2^{\sum_{j=1}^W H_j} = 2^H \quad (2.89).$$

Для структур, у которых  $H_j > N$ , для любого  $j=1, \dots, W$   $\psi_N^{opt} < 2^H$ .

При программной реализации нейронной сети возникает необходимость ограничить суммарное число входов в сети, вызванное тем, что число входов есть число технически трудно реализуемых блоков умножения. На этапе настройки выбранной структуры число входов равно размерности пространства настраиваемых коэффициентов, в котором производится поиск экстремума функционала качества модели. Поэтому уменьшение числа входов нейронов многослойной сети облегчает как реализацию, так и настройку.

Для нейронной сети с полными перекрестными связями суммарное количество входов нейронов в  $i$ -ом слое равно  $\gamma_N = (L_j + N)H_{ij}, i_j = 1, \dots, W$ .

Отсюда для суммарного количества входов нейронов в  $W$ -слойной сети:

$$\gamma_W = \sum_{j=1}^W \gamma_j = \sum_{j=1}^W \left[ \sum_{j=1}^i H_j + N \right] H_j = N \sum_{j=1}^W H_j = \frac{1}{2} H^2 - \frac{1}{2} \sum_{j=1}^W H_j \quad (2.90).$$

На основании (2.88) для задачи многослойной нейронной сети, оптимальной по верхней или нижней оценке количества областей при ограничении на суммарное число входов в сети, формулируется следующим образом:

$$\left. \begin{aligned} \psi_{N[W]}^* \max_W \max_{H_1 \dots H_H} \psi_{N[W]j} \\ \gamma \geq N \sum_{j=1}^W H_j + \frac{1}{2} \left[ \sum_{j=1}^W H_j \right]^2 - \frac{1}{2} \sum_{j=1}^W H_j^2 \end{aligned} \right\} \quad (2.91),$$

где \* – экстремальное значение. Обратная задача синтеза структуры многослойной нейронной сети, минимальной по общему числу входов, при ограничении на количество областей  $\Psi$ , реализуемых нейронной сетью, имеет следующий вид:

$$\left. \begin{aligned} \gamma_W = \min_W \min_{H_1, \dots, H_N} \left[ N \sum_{j=1}^W H_j + \frac{1}{2} \left[ \sum_{j=1}^W H_j \right]^2 - \frac{1}{2} \sum_{j=1}^W H_j^2 \right] \\ \Psi_{N[W]} \geq \Psi \end{aligned} \right\} \quad (2.92).$$

В формулах (2.91) и (2.92)  $\Psi_{N[W]}$  определяется выражением (2.86). [115, стр. 50-70]

Используя формулы (2.82)-(2.92), учитывая, что ограничение на максимальное количество нейронов равно 50 и количество выходных нейронов равно 1 получаем для первого слоя  $\gamma_1=22$  – оптимальное количество нейронов.

В качестве экспериментальной выборки было взято множество значений ежедневных цен одного из финансовых инструментов, анализируемых интеллектуальной системой поддержки принятия решений, акций ОАО «Газпром» за период с 11.01.2010 по 13.05.2010.

Данный метод оптимизации выдает оптимальное количество нейронов, но не указывает на распределение этого количества по входным признакам. Так как на вход нейронной сети будут подаваться 4 вида данных (цены, RSI, MACD, GARCH), необходимо оптимально распределить между ними общее количество входных нейронов.

Главным входным определяющим признаком нейронной сети для задачи классификации рыночных ситуаций являются цены финансового инструмента. Следовательно, все остальные признаки необходимо выбирать по степени связанности и взаимному влиянию признаков на цены. В качестве критериев оптимизации выбора количества входных признаков каждого вида выбраны коэффициент корреляции и коэффициент детерминации, отражающие соответственно тесноту связи и степень взаимообусловленности.

В первую очередь необходимо определить, существует ли в ряде значений исследуемого главного признака автокорреляционная зависимость. При обнаружении присутствия автокорреляции данные ряды значений необходимо так же включить в множество входных значений проектируемой модели нейронной сети. Результаты исследования автокорреляционных зависимостей в рядах цен представлены в таблице 2.8.

Таблица 2.8 Автокорреляция цен

Задержка	Коэффициент корреляции	Коэффициент детерминации
r1	0,92219	0,850435
r2	0,822593	0,67666
r3	0,746068	0,556617

Продолжение таблицы 2.8

r4	0,640258	0,409931
r5	0,5234	0,273947
r6	0,459903	0,211511
r7	0,411539	0,169364
r8	0,357022	0,127465
r9	0,346096	0,119782
r10	0,344172	0,118454

В таблице 2.8 r1, r2, ..., r10 – обозначают задержки в датах. Так в строке r1 представлены коэффициенты для рядов цен с 12.01.2010 по 13.05.2010 и цен с 11.01.2010 по 12.05.2010, то есть данные коэффициенты отражают влияние предыдущей цены на текущую. В строке r2 коэффициенты отражают влияние цены за позапрошлую дату на текущую. В строке r10 – влияние цены за дату, предшествующую на 9 дней текущей, и цены на текущую дату.

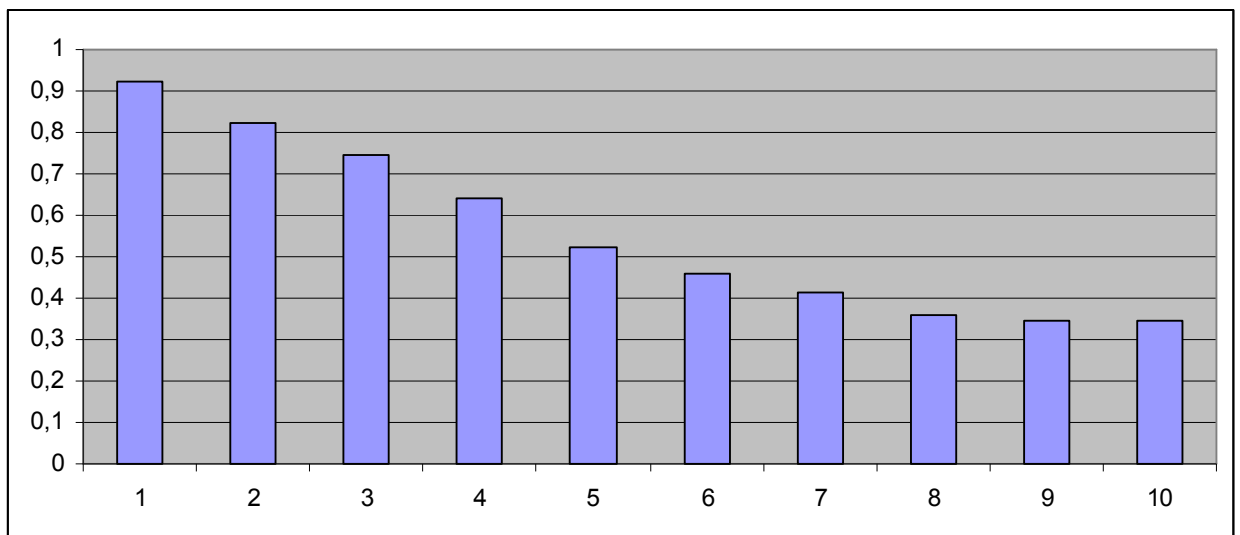


Рисунок 2.11 – Автокоррелограмма цен

На рисунке 2.11 представлено графическое отображение присутствия автокорреляционной зависимости в рядах цен финансового инструмента.

Значения коэффициента корреляции определяет степень статистической связи между двумя рядами данных. Так как значимыми считаются связи, оцениваемые модулем коэффициента корреляции в размере более 0,15, то есть все признаки, имеющие коэффициент корреляции с главным определяющим признаком менее данного значения не должны включаться в модель.

Значения коэффициента детерминации определяет какая доля отклонений определяющего признака обусловлена долей отклонения данного признака. Значимыми являются значения коэффициентов детерминации более 0,1, то есть если более 10 % отклонений определяющего при-

знака могут быть объяснены отклонениями данного признака. Все ряды данных, значения которых более 0,1 могут быть включены в множество входных данных модели.

Из таблицы 2.8 видно, что все 10 рядов данных могут быть включены в входное множество.

В таблице 2.9 приведены результаты экспериментального исследования взаимовлияния цен и значений технического индикатора RSI.

Таблица 2.9 Взаимовлияние цен и значений технического индикатора RSI

Задержка	Коэффициент корреляции	Коэффициент детерминации
r0	0,460427	0,211993
r1	0,368239	0,1356
r2	0,273773	0,074952
r3	0,180238	0,032486
r4	0,120303	0,014473
r5	0,11061	0,012235
r6	0,098803	0,009762
r7	0,074281	0,005518
r8	0,073051	0,005336
r9	0,05745	0,0033

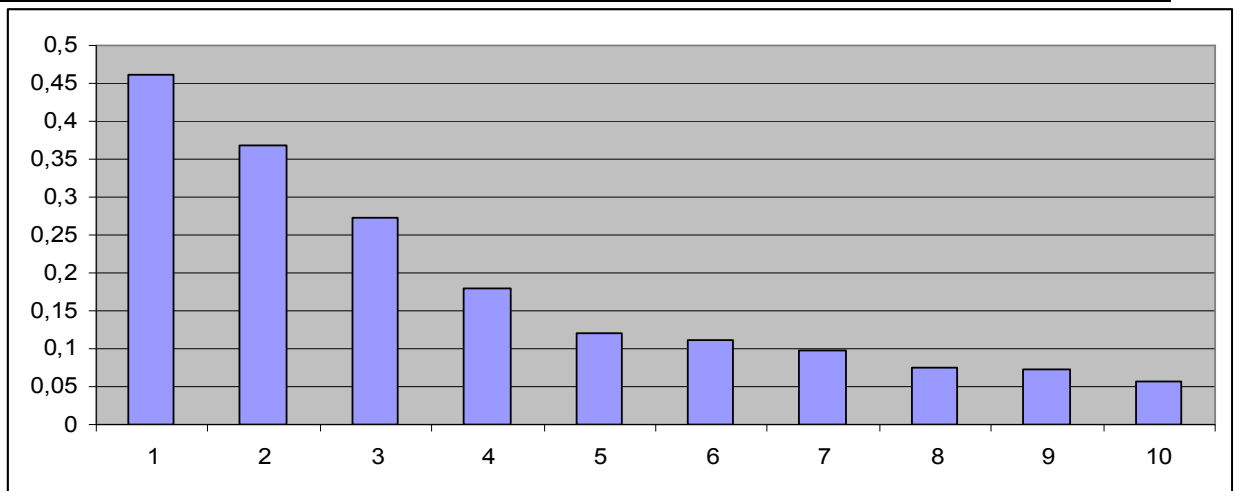


Рисунок 2.12 – Коррелограмма цен и RSI

В таблице 2.9  $r_0, r_2, \dots, r_9$  – обозначают задержки в датах. Например, в строке  $r_1$  представлены коэффициенты для ряда цен с 11.01.2010 по 13.05.2010 и значений RSI для этого же периода. А в строке  $r_2$  – цены за период с 12.01.2010 по 13.05.2010, а RSI – с 12.01.2010 по 12.05.2010, то есть исследуется зависимость цен от значений RSI за предыдущую дату. В строке  $r_3$  исследуются зависимость цен от значений RSI с задержкой 2 дня.

На рисунке 2.12 изображена коррелограмма цен и RSI, отражающая корреляцию между данными признаками.

Из таблицы 2.9 и рисунка 2.12 видно, что в множество входных значений могут быть включены ряды данных значений RSI с задержкой до 3 дней.

В таблице 2.10 приведены результаты экспериментального исследования взаимовлияния цен и значений технического индикатора MACD (иллюстрация результатов на рисунке 2.13).

Таблица 2.10. Взаимовлияние цен и значений технического индикатора MACD

Задержка	Коэффициент корреляции	Коэффициент детерминации
r0	-0,51207	0,262216
r1	-0,46388	0,215189
r2	-0,4048	0,163864
r3	-0,34743	0,120704
r4	-0,27798	0,077275
r5	-0,22808	0,052021
r6	-0,20902	0,043688
r7	-0,20094	0,040378
r8	-0,2017	0,040682
r9	-0,21925	0,048072

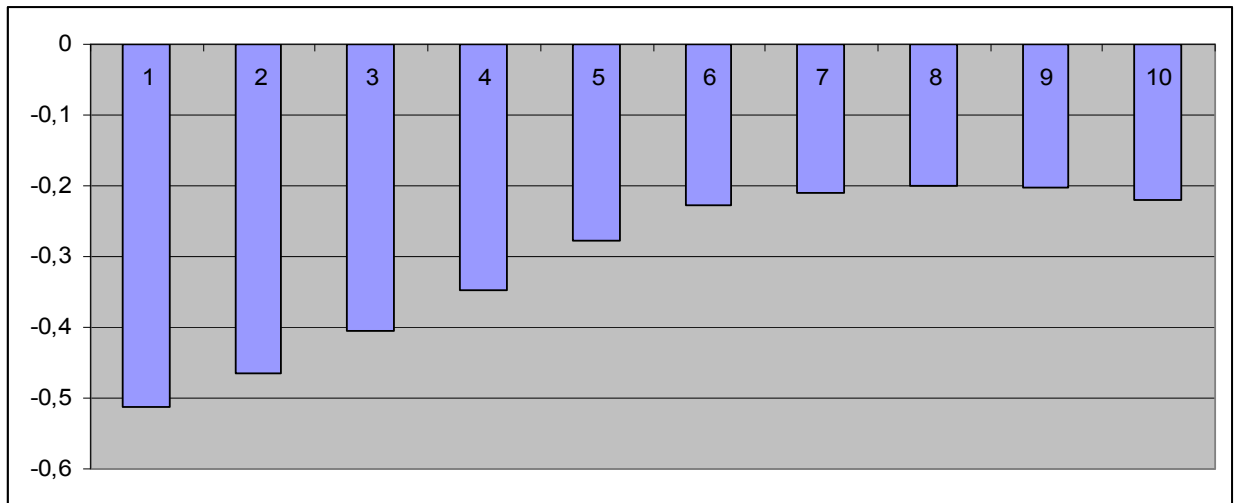


Рисунок 2.13 – Коррелограмма цен и MACD

По данным из таблицы 2.10 можно сделать вывод, что в множество входных значений могут быть включены ряды данных значений MACD с задержкой до 3 дней (с задержкой 4 дня при необходимости тоже может быть включена в модель, так как значение коэффициента корреляции достаточно велико).



В таблице 2.11 приведены результаты экспериментального исследования взаимовлияния цен и значений технического индикатора GARCH (иллюстрация результатов на рисунке 2.14).

Таблица 2.11 Взаимовлияние цен и значений по модели GARCH

Задержка	Коэффициент корреляции	Коэффициент детерминации
r0	0,385516	0,148623
r1	0,403354	0,162695
r2	0,39518	0,156167
r3	0,340645	0,116039
r4	0,279248	0,077979
r5	0,223215	0,049825
r6	0,173277	0,030025
r7	0,158457	0,025109
r8	0,138432	0,019164
r9	0,138697	0,019237

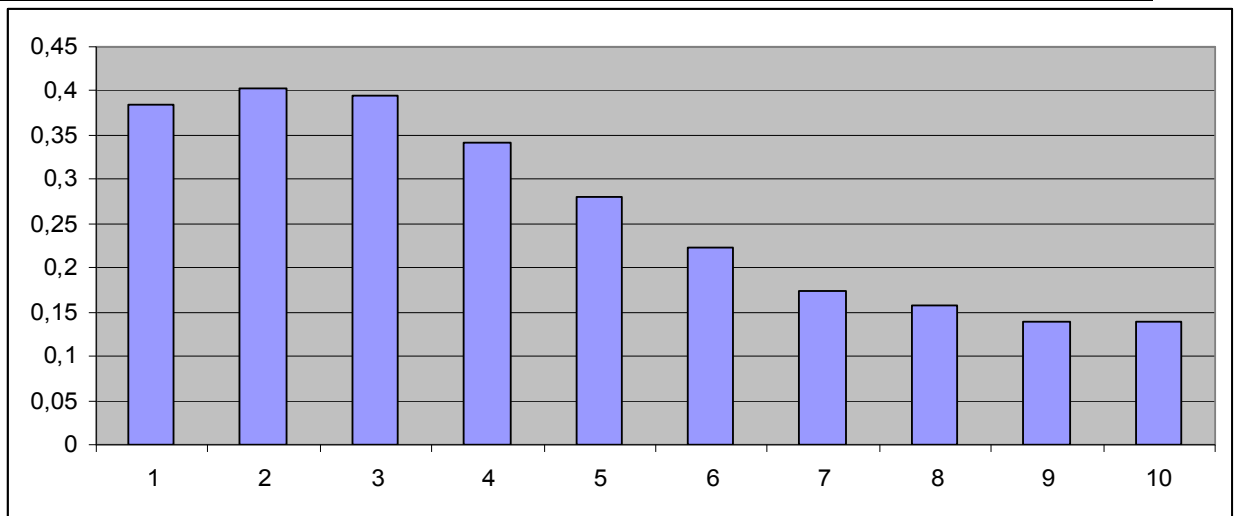


Рисунок 2.14 – Коррелограмма цен и GARCH

По значениям коэффициентов из таблицы 2.11 можно сделать вывод, что в множество входных значений проектируемой модели нейросети могут быть включены ряды данных значений GARCH с задержкой до 4 дней.

В результате экспериментов было выяснено, что оптимальным количеством нейронов входного слоя 22. Методами корреляционного анализа было доказано обоснованность включения в входное множество значений:

- цен с задержкой до 10 дней;
- RSI с задержкой до 3 дней;
- MACD с задержкой до 4 дней;

– GARCH с задержкой до 4 дней.

Для баланса влияния технических индикаторов и значений по прогнозной модели необходимо включить во входной слой по одинаковому количеству нейронов, на которые поступают значения по данным признакам. Таким образом в модель включаем 22 входных нейрона, из которых 4 – это значения цен, RSI, MACD, GARCH за текущий период, 9 являются значениями цен за предыдущие 9 периодов, 3 – значения по 3 предыдущим периодам технического индикатора RSI, 3 – значения по 3 предыдущим периодам технического индикатора MACD, 3 – значения по 3 предыдущим периодам технического индикатора по прогнозной модели.

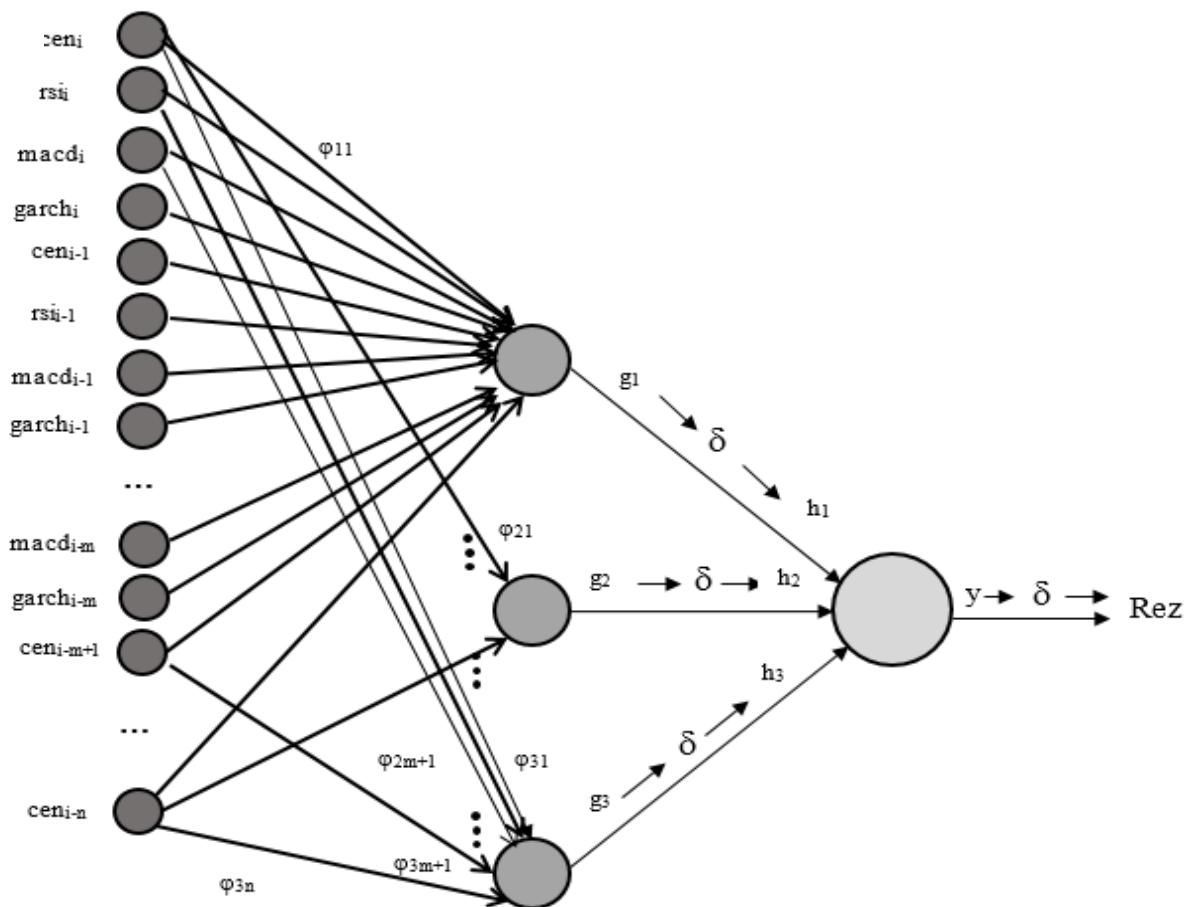


Рисунок 2.15 – Архитектура спроектированной нейросети

После того, как архитектура нейронной сети стала ясна (см. рисунок 2.15), рассмотрим процесс формирования выходной выборки для задачи классификации рыночных ситуаций с учетом повышения волатильности.

### 2.5.2 Метод генерации выходной выборки для задачи нейросетевой классификации рыночных ситуаций

Классификация рыночных ситуаций заключается в определении паттерна «покупки», «продажи» или «ожидания», которому принадлежит множество значений котировок, их технических индикаторов и значений по прогнозной модели.

Используется полносвязанный многослойный перцептрон, на входы которого поступают значения предыдущих цен финансового инструмента, соответствующие им значения индикаторов RSI и MACD (исходные данные являются разно размерными, необходимо произвести предварительную обработку данных, подробнее о нормировке данных см. раздел 2.3), а выходами которой является число в интервале от 0 до 1, означающие рекомендацию продавать  $[0;0,45)$ , ожидать  $0,5\pm 0,05$ , покупать  $(0,55;1]$ .

В процессе обучения нейронной сети необходимо распознавать образы (классы) рыночных ситуаций, подходящих для покупки-продажи финансового инструмента, и сравнивать их с подаваемыми на входы образами.

Возникает вопрос, каким образом при обучении формировать выходную выборку, то есть множество значений, принадлежащих к одному из классов: покупка, продажа, ожидание.

Первым интуитивно понятным ответом возникает экспертная оценка. При классификации экспертом выступает опытный трейдер-аналитик, который относит текущую ситуацию к одному из классов. То есть на вход сети подаются сигналы, эксперт анализирует происходящие изменения после последнего периода из входного множества, и дает свою оценку, к какому классу соответствовало входное множество.

Были проведены экспериментальные исследования данного подхода к обучению. В обучающую выборку входили 100 пар входов и выходов. Вход – вектор из 18 отнормированных значений цен и технических индикаторов. Выход – экспертная оценка для текущей ситуации на рынке финансового инструмента в диапазоне от 0 до 1. Тестовая выборка состояла из 100 пар входов и выходов. Было произведено 2 эпохи обучения со скоростью 0,1 и с критерием остановки 0,05.

Результаты экспериментов приведены на данных котировок фьючерсов нефти Brent на Московской Бирже с периодичностью 1 час за период с 01.01.2016 до 05.03.2016 г (график изменений котировок за данный период представлен на рисунке 2.16).

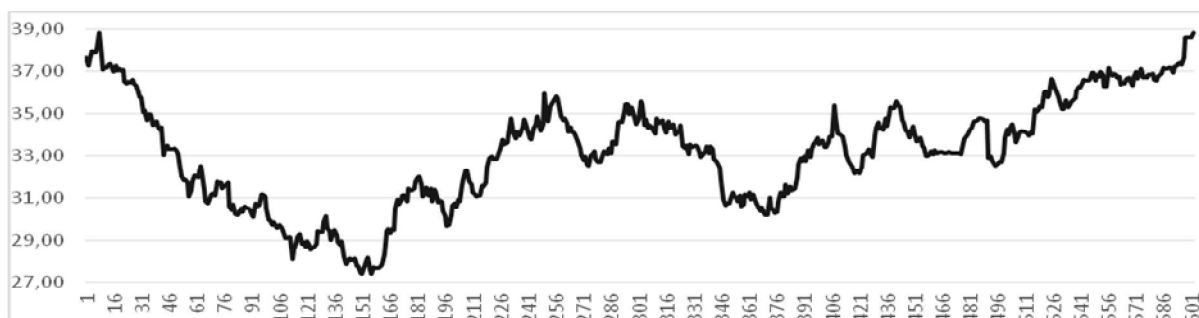


Рисунок 2.16 – Часовые котировки фьючерсов нефти Brent на Московской Бирже с 01.01.2016 до 05.03.2016 г

Таблица 2.12 Результаты тестирования нейросети для классификации рыночных ситуаций

Рекомендация	Процент верных рекомендаций	$\sigma$
Покупать	52,2%	0,32
Ожидать	30%	0,37
Продавать	72,2%	0,21
Итого	62,2%	0,34

По результатам тестирования (приведены в таблице 2.12) обученного на основе экспертных оценок нейросетевого классификатора рыночных ситуаций, можно заметить, что в данном случае, нейронная сеть эмулирует стратегию данного конкретного эксперта, который опирался на свой опыт и интуицию в выставлении оценок. По отсутствию верных рекомендаций в классе ожидания можно судить о более агрессивной позиции трейдера-эксперта, характеризующейся частым открытием сделок даже в неблагоприятный момент. Большое число правильно распознанных позиций на продажу свидетельствует о большей предрасположенности данного трейдера к продаже.

Нивелировать воздействие личной стратегии эксперта, устранить вклад эмоционального, интуитивного фактора, позволит метод, учитывающий ожидаемую прибыль трейдера. В данном способе оценки становятся точнее и последовательнее, что улучшает качество обучения нейронной сети.

Для эксперимента оценка (значение выходной выборки) давалась на основе 20 значений котировок: 10 текущих котировок (в 1 входной вектор входили по 10 последовательных значений котировок) и 10 последующих. Уровень прибыли 0,9, означал, что в следующие 10 часов от рассматриваемого периода трейдер желает получить прибыль со сделки по одному фьючерсу в размере 0,9 руб. Если за последующий период прибыль достигалась полностью ставилась оценка 1 или 0, соответственно для продажи и покупки. Если прибыль была меньше, то оценка уменьшалась пропорционально. Если прибыль была менее 0,09, выдавался вывод о ситуации ожидания.

Тестирование нейросетевого классификатора с оценками на основе учета ожидаемой прибыли, производилось с архитектурой и параметрами обучения, совпадающими с экспериментом первого метода. Результаты эксперимента второго подхода к обучению приведены в таблице 2.13. В сравнении с первым методом результаты выдаваемых рекомендаций всех классов примерно выровнялись. Уменьшилась среднеквадратичное отклонение. Увеличилась общее число верно выдаваемых классификатором рекомендаций.

Таблица 2.13 Результаты тестирования нейросети для классификации рыночных ситуаций с учетом ожидаемой прибыли

Рекомендация	Процент верных рекомендаций	$\sigma$
Покупать	69,7%	0,17
Ожидать	58,8%	0,31
Продавать	78,6%	0,19
Итого	73,2%	0,21

Таким образом, нейросетевой классификатор на основе оценки с учетом ожидаемой прибыли позволит более эффективно работать системе поддержки принятия решений трейдера финансового рынка. При формировании обучающей и тестовой выборки в случае применения данного подхода также появляется возможность автоматизации процесса выработки выходных значений, что упростит обработку исходных данных и позволит использовать систему поддержки принятия решений без обращения за экспертными оценками [75].

Нейросетевая классификация заключается в распознавании на основе входных данных закономерностей и отнесении к одному из образов {покупка, продажа, ожидание}, где  $I_t$  – вектор входных значений из входной выборки,  $O_t$  – выходной вектор, состоящий из одного значения в диапазоне от 0 до 1, обозначающее степень близости к классу продажи (значения из множества  $\{0;0,45\}$ ), классу ожидания (значения из множества  $\{0,45;0,55\}$ ), к классу покупки (значения из множества  $\{0,55;1\}$ ). Вектор  $I_t$  состоит из входных значений, в которые включены в том числе и 10 предыдущих цен финансового инструмента, он является входом в обучающей паре. Обозначим часть входного вектора, необходимого для формирования выхода сети, за  $C_t$ . Метод генерации выходного значения на основе входного вектора  $C_t$  заключается в сравнении процента достижения ожидаемой прибыли при покупке и продаже на указанном временном промежутке.

Рассмотрим алгоритм автоматизации процесса генерации выходного множества для обучения нейронной сети с учетом ожидаемой прибыли. В данном случае система будет выдавать оценки (выходные значение) согласно вышеописанному подходу. На вход каждого обучающего примера подается вектор из 10 цен финансового инструмента. Визуальное пояснение приведено на рисунке 2.17.

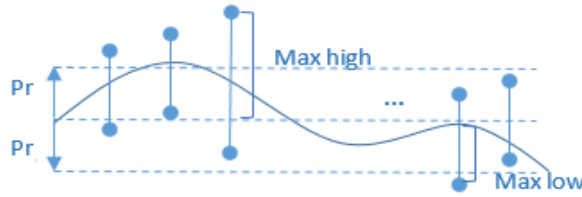


Рисунок 2.17 – Иллюстрация алгоритма обучения сети с учетом ожидаемой прибыли

1. Задаем ожидаемый уровень прибыли  $Pr$ , который планируется достигнуть трейдером при заключении сделки по конкретному финансовому инструменту в следующие  $n$  периодов.
2. Для каждого входного вектора  $I_t$  среди значений в  $C_t$  вычисляем максимальное повышение от первой цены закрытия в выборке *Max high*.
3. Для каждого входного вектора вычисляем максимальное понижение от первой цены закрытия в выборке *Max low*.
4. Вычисляем процент максимального повышения от ожидаемой прибыли *Proc Max high*. Если получаем значение больше 1, округляем его до 1.
5. Вычисляем процент максимального понижения от ожидаемой прибыли *Proc Max low*. Если получаем значение больше 1, округляем его до 1.
6. Если  $Proc\ Max\ high > Proc\ Max\ low$ , возвращает результат покупки:
 
$$R = 0.5 + 0.05 * ProcMaxHigh * 100$$
7. Если  $Proc\ Max\ high < Proc\ Max\ low$ , возвращает результат продажи:
 
$$R = 0.5 - 0.05 * ProcMaxLow * 100$$
8. Если  $Proc\ Max\ high = Proc\ Max\ low$ , возвращает результат ожидания:  $R=0.5$ , если процент достижения прибыли менее 50, иначе анализируем направление покупки или продажи по тренду.

Таким образом, для каждого обучающего входного набора  $I_t$  формируется обучающий выход нейронной сети  $O_t$  в диапазоне от 0 до 1 по приведенному алгоритму. И как уже говорилось в разделе 2.4.2 при обучении стоит учитывать коэффициент повышения волатильности (алгоритм описан там же). При резком повышении цены в условиях повышенной волатильности уменьшаем процент максимального повышения на коэффициент:  $ProcMaxHigh/k$ . При понижении цены в условиях повышенной волатильности, увеличиваем процент максимального понижения:  $ProcMaxLow*k$ .

2.5.3 Методика поддержки принятия решений трейдера и ее алгоритм классификации рыночных ситуаций с использованием многослойного перцептрона и методов анализа финансовых рынков

Суть методики нейросетевой классификации рыночных ситуаций [74] сводится к построению нейросетевой динамической системы, обрабатывающей данные из трех основных подходов к анализу рынков: технический анализ, прогнозные модели, фундаментальный анализ, сличающей образы с 3 классами рыночных ситуаций, и предоставляющей на выходе ответ, к какому классу относится текущая рыночная ситуация выбранного финансового инструмента. Методика нейросетевой классификации сводится к совокупности шагов, комбинации внутри них различных методов и алгоритмов, направленных на выработку рекомендации о направлении открытия сделки для трейдера по заявленным им параметрам.

В качестве ядра динамической системы, функционирующей во времени и устанавливающей соответствие между входными сигналами и выходами, выбрана нейронная сеть типа многослойный перцептрон. Обучение многослойного перцептрона производится алгоритмом обратного распространения ошибки на выборке из 200 предыдущих значений цен, соответствующих им значениям технических индикаторов RSI и MACD (об обосновании выбора индикаторов см. [68]) и значений по прогнозной модели GARCH.

В обучающем множестве входами являются описанные выше значения, а выходами – значения:

- $[0;0,45)$  – рекомендация со значением *Prod*;
- $0,5\pm 0,05$  – рекомендация со значением *Ozh*;
- $(0,45;1]$  – рекомендация со значением *Pok* (см. рисунок 2.18).

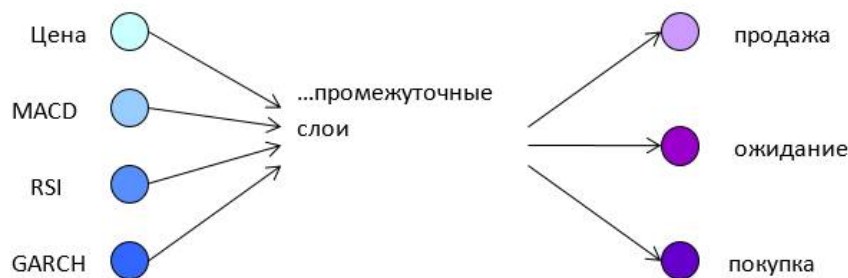


Рисунок 2.18. Общая модель спроектированной нейронной сети

Таким образом, сеть учиться устанавливать соответствие между предыдущими значениями цены, технических индикаторов, прогнозными значениями и будущими повышениями или понижениями цены в соответствии с оценками (см. рис. 2.18), полученными по методу генерации выходной выборки. Многослойный перцептрон будет обучаться с помощью алгоритма обратного распространения ошибки.

Обученная нейронная сеть будет выдавать значение в интервале от 0 до 1:

- если выходное значение будет находиться в интервале  $[0;0,45)$ , то СППР будет рекомендовать продажу финансового инструмента;
- если выходное значение будет равно  $0,5 \pm 0,05$ , то рекомендуется ожидание;
- если выходное значение будет находиться в интервале  $(0,55;1]$ , то система будет рекомендовать покупку финансового инструмента.

В процессе обучения сеть должна будет научиться на заданных примерах определять ситуацию на рынке. С помощью использования передаточной функции и метода обратного распространения ошибок, обучающего правила проектируемый многослойный перцептрон обучится сам выявлять такого рода ситуации для любых входных значений.

Идея классификации рыночных ситуаций состоит в том, чтобы обучить многослойный перцептрон с помощью алгоритма обратного распространения так, что на входной образ (несколько предыдущих цен и значений технических индикаторов, значений по прогнозной модели) он будет реагировать сигналом: «покупать», «продавать» или «ожидать».

Таким образом модель принятия решения о направлении заключения сделки трейдером финансового рынка состоит из информации, информационного управления, рекомендации к действию, действия и полученного результата от сделки (см. рисунок 2.19).

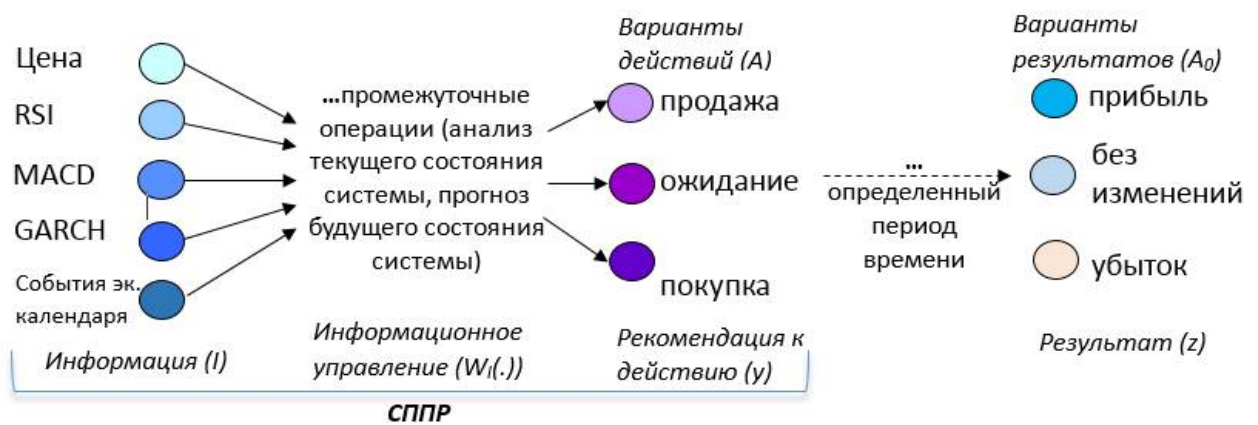


Рисунок 2.19 – Модель принятия решения трейдером о направлении сделки

Информационное управление в данном случае отводится системе поддержки принятия решений, которая анализирует полученную многокритериальную информацию и выносит рекомендацию о дальнейшем состоянии системы, которое может быть воспринято ЛППР в качестве рекомендации к действию по покупке или продаже финансового инструмента.

Алгоритм классификации рыночных ситуаций с использованием многослойного перцептрона и методов анализа финансовых рынков, который составит основу интеллектуальной системы поддержки принятия решений на финансовых рынках (см. рисунок 2.20):

1. Определить финансовый инструмент, для которого необходимо произвести анализ.



2. Определить временной интервал (15 минут, час, день и т.д.), цены закрытия по которым будут анализироваться.

3. Определить ожидаемую прибыль от сделки по финансовому инструменту.

4. Загрузить последние двести тридцать четыре (первые 34 значений необходимы для расчета значений технических индикаторов, соответствующих двадцать седьмому периоду) цен выбранного финансового инструмента. Разработка методов извлечения данных из интернета выходит за пределы проводимого исследования, но ознакомиться с разработанным автором работы методом можно в Приложении В к данной диссертации.

5. Вычислить соответствующие ценам технические индикаторы RSI и MACD, значения по GARCH.

6. Загрузить экономический календарь по странам, соответствующим выбранному финансовому инструменту (о методах загрузки см. Приложение В). Проанализировать наличие важных событий экономического календаря.

6. Нормировать цены и значения индикаторов в интервал  $[-1,1]$  с помощью функции гиперболического тангенса.

7. Нормированные данные разделить на две части:

- обучающее множество (за 120 периодов);
- тестовое множество (оставшиеся 80).

8. Обучать многослойный перцептрон алгоритмом обратного распространения ошибки (математическую постановку см. в [72]), применяя метод генерации выходной выборки.

На входы подаются вектора из обучающего множества (цена, значение RSI, значение MACD, значение по модели GARCH). На выходы – полученное с помощью метода генерации выходной выборки значение из множества:  $\{0;0,5;1\}$ , как оценка с учетом ожидаемой прибыли и повышением волатильности.

9. Тестировать многослойный перцептрон.

На входы подаются вектора из тестового множества. Выходом перцептрона является величина в пределах от 0 до 1, сравнить этот выход с реальным (оценкой по методу). Если ошибка тестирования достаточно мала, то можно использовать обученный многослойный перцептрон для получения рекомендации о ситуации на рынке данного финансового инструмента.

10. Использовать многослойный перцептрон для получения рекомендации по покупке-продаже данного финансового инструмента.

На вход нейросети подать последние нормированные данные о финансовом инструменте. Выходом нейросети будет являться значение из множества от 0 до 1, отражающее ситуацию на

финансовом рынке, которое система интерпретирует в рекомендацию {продавать, ожидать, покупать}.

11. Уведомить пользователя о наличии важных событий экономического календаря в период, близкий к текущему.

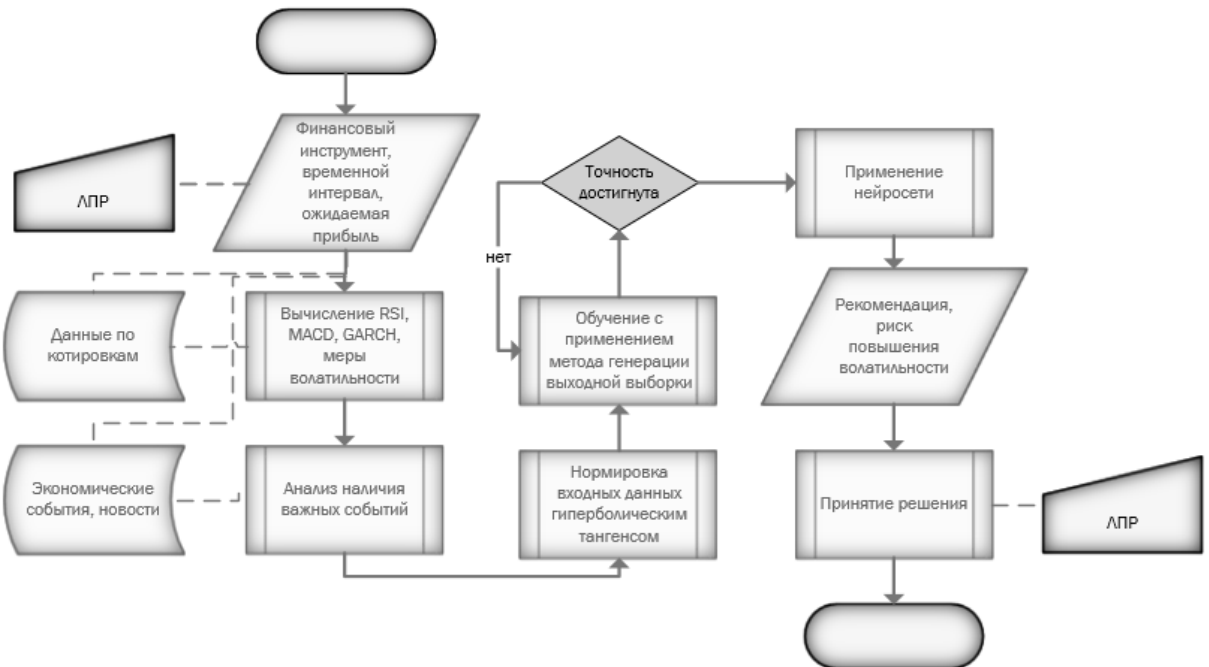


Рисунок 2.20 – Схема алгоритма нейросетевой классификации рыночных ситуаций

Представленная схема (рис. 2.20) алгоритма разработанной методики нейросетевой классификации рыночных ситуаций делает возможным пошаговую реализацию данного алгоритма в информационной системе.

Таким образом, многослойный перцептрон будет работать как классификатор рыночных ситуаций. И тот факт, что при обучении используется оценки с учетом ожидаемой прибыли и повышения волатильности, дает возможность получения хороших результатов по предсказанию будущего поведения финансовых рынков, что положительным образом скажется на доходности от применения системы поддержки принятия решений, основанной на данной архитектуре нейронной сети.

### Выводы по главе

В главе рассмотрены методы и алгоритмы, направленные на разработку методики интеллектуальной классификации рыночных ситуаций.

1. Представлены используемые алгоритмы технического анализа, приведены алгоритмы составления прогнозной модели GARCH, и методы оценки ее параметров.

2. Подробно описана выбранная топология нейронной сети для задачи классификации – многослойный перцептрон. Приведены результаты исследования по синтезу архитектуры

нейросети, подбору функций нормировки данных перед обучением сети. Описаны математическая модель задачи обучения нейронной сети, алгоритмы обучения многослойного перцептрона, приведено обоснование эффективности алгоритма обратного распространения в качестве алгоритма обучения, а также приведен пример рассмотрения задачи обучения с точки зрения классической оптимизации.

3. В дополнение к разделу о возможностях нейронных сетей при анализе финансовых рынков приведены результаты исследования по распознаванию кризисных паттернов в финансовых временных рядах, и дана математическая постановка распознавания кризисных паттернов с использованием нейросети Хопфилда.

4. Приведены результаты исследований влияния событий экономического календаря на повышение волатильности в временных рядах цен финансовых инструментов. Установлена исследуемая взаимосвязь и разработан алгоритм анализа повышенных волатильности при совершении экономических событий для обучающейся системы.

5. Дана формальная постановка задачи классификации рыночных ситуаций для решения системой поддержки принятия решений. Приведена модель функционирования нейронной сети и шаги для ее составления.

6. Приведены сравнительные результаты обучения нейросети с помощью экспертных оценок и на основе анализа ожидаемой прибыли. Описан метод генерации выходной выборки с учетом ожидаемой прибыли и повышения волатильности.

7. В заключении главы описана разработанный алгоритм интеллектуальной классификации рыночных ситуаций с использованием многослойного перцептрона и методов анализа финансовых рынков.

По результатам исследований и наработок, представленных в данной главе, доказана возможность создания интеллектуальной системы поддержки принятия решений трейдеров финансовых рынков на основе разработанной методики классификации рыночных ситуаций, поскольку все методы и алгоритмы имеют под собой строгое научное обоснование.

Проектные решения по разработке системы поддержки принятия решений «НейроПрофит», основанной на разработанной методике интеллектуальной классификации рыночных ситуаций, проверка работоспособности системы будет рассмотрена в следующей главе.

### **3 ПРОЕКТНЫЕ РЕШЕНИЯ ПО РАЗРАБОТКЕ ИНТЕЛЛЕКТУАЛЬНОЙ СИСТЕМЫ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ**

Моделирование работы информационной системы особенно важно на первых этапах её создания. Так как исправление допущенных на этом этапе ошибок обходится наиболее дорого, то и польза на этапе анализа задачи и разработки логической модели её решения значительна.

Если сравнить усилия на непосредственное кодирование с затратами труда на остальные составляющие разработки системы реального времени, то непосредственно кодирование занимает очень малое время по сравнению со всем остальным: анализом, проектированием, обработкой ошибок, тестированием, отладкой, документированием, совещаниями, проектным менеджментом и т.п.

Этап анализа определяет не только технические требования, но и даёт базу для разработки необходимых структур данных. По завершении анализа можно оценить количество рабочих дней, необходимых для реализации проекта. Для этого просчитывалось время на обработку информации на самом низком уровне. На этой стадии важно не забыть учесть время на оформление документации, контроль качества, совещания, интеграцию решения и т.п.

#### **3.1 Проектирование архитектуры системы**

Проектирование программного обеспечения является этапом жизненного цикла программного обеспечения, на котором исследуют структуру и взаимосвязи элементов разрабатываемой системы. В результате получают технический проект, содержащий описание для реализации системы. Различают функциональное проектирование, проектирование архитектуры системы и детальное проектирование программных модулей.

Перед проектированием системы важно определить, с помощью каких инструментов систему планируется реализовывать. Платформа Microsoft Visual Studio является одним из наиболее популярных программных продуктов для создания информационных систем. Visual Studio с точки зрения разработчика – это, прежде всего, великолепная среда визуального программирования, понятная, простая для изучения. В среде Visual Studio можно создавать достаточно сложные программные системы практически с нуля, написав минимум программного кода. Также платформа позволяет писать код на различных языках программирования. При разработке ИСППР «НейроПрофит» использовался распространённый на данный момент язык программирования C#.

Однако если для создания небольших программных систем для собственных нужд можно ограничиться средой визуального программирования, то создание любого более или менее критичного к качеству программного обеспечения (ПО) требует принципиально другого подхода,

как минимум, использования при разработке модели жизненного цикла ПО (со всеми следующими отсюда требованиями к документированию процесса).

Принципиальным для сложных систем является наличие базовой архитектуры (см. рис. 3.1), основные элементы которой закладываются на ранних стадиях проектирования.

Система состоит из 3 модулей, каждый из которых соответствует одному из подходов для анализа рынков. Для получения исходных данных используются 2 веб-сервиса для получения данных из интернета (подробнее см. Приложение В). Полученные данные хранятся в кэше, который позволит многократно к ним обращаться без необходимости повторной загрузки. Информация, преобразованная в этих трех модулях, передается в ядро системы, которым является непосредственно нейросеть и прочие механизмы анализа данных.

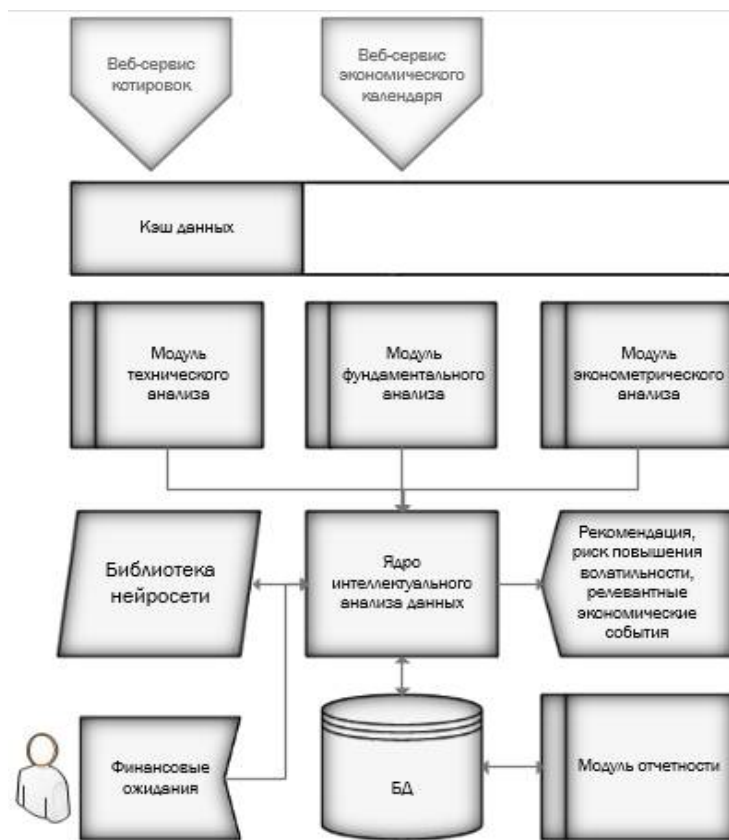


Рисунок 3.1 – Архитектура ИСППР

Базовая архитектура поясняется, как правило, с использованием ряда моделей, отражающих основные моменты, связанные со структурой системы и ее функциями.

### 3.2 Проектирование системы с использованием case-средств

CASE-средства позволяют получить описание будущей функциональности создаваемой системы до ее реализации. Также с их помощью можно анализировать работу системы и оптимизировать подготавливаемые решения.

В настоящее время все case-средства (специальные программные продукты, предназначенные для моделирования и проектирования информационных систем) делятся на две большие группы. Такое разделение связано с применяемой в них методологией проектирования, основанные на двух парадигмах: структурное и объектно-ориентированное проектирование. Для проектирования ИСППР воспользуемся возможностями, которые предоставляет средства ER-Win и BP-Win с используемой в них нотацией IDEF, в качестве представителя парадигмы структурного проектирования, и Rational Rose, проектирование в котором происходит на основе UML – объектно-ориентированной парадигмы.

### 3.2.1 Функциональное проектирование

Функциональная модель информационной системы отражает ее будущий функционал, отражающий ее назначение и способы использования.

Название информационно-управляющей системы:

Интеллектуальная система поддержки принятия решений трейдеров финансовых рынков «НейроПрофит»

Цель системы:

На основе нейросетевого анализа прошлых цен финансового инструмента (классификации рыночных ситуаций многослойным перцептроном) выдавать рекомендации по покупке-продаже в режиме реального времени (см. рисунок 3.2).

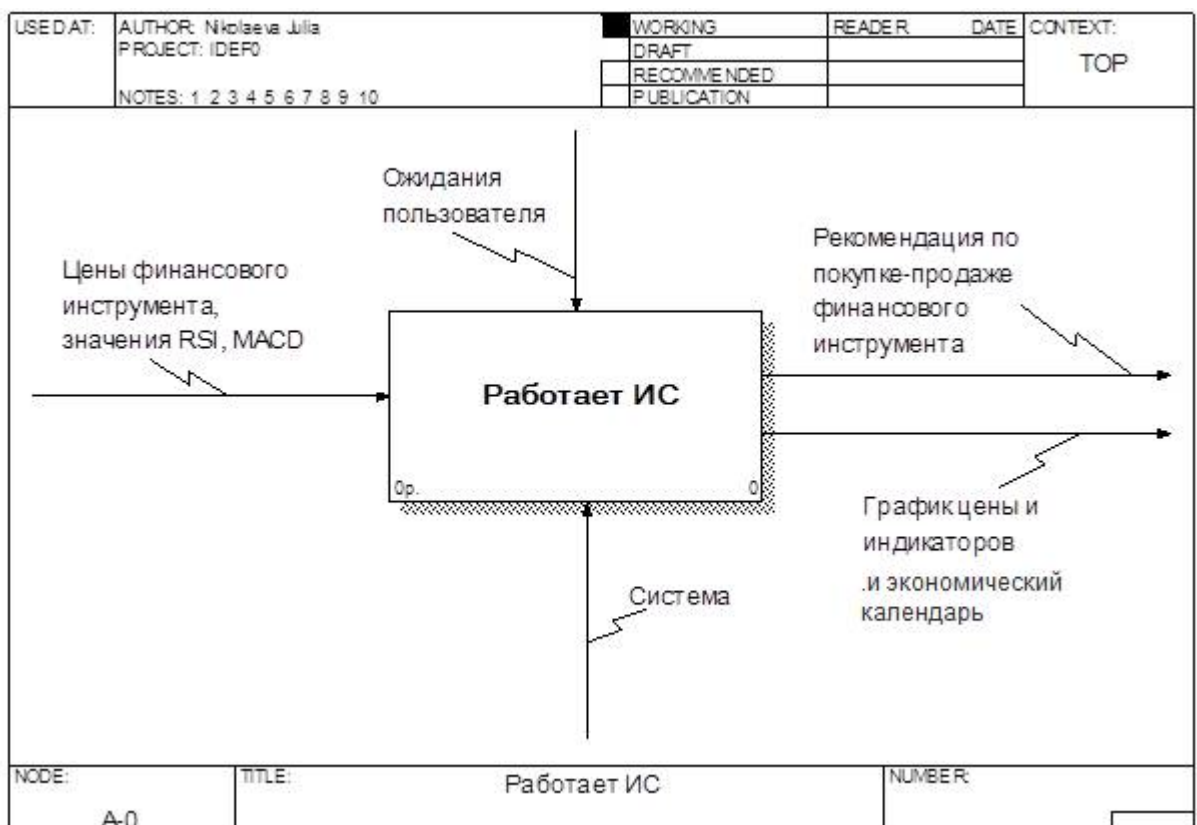


Рисунок 3.2 – Контекстная диаграмма IDEF0

На вход системы подаются предыдущие за несколько периодов цены финансового инструмента и значения его индикаторов технического анализа RSI (Relative Strength Index) и MACD (Moving Average Convergence/Divergence). Под финансовым инструментом понимается любой объект, торговля которым ведется на финансовых рынках. Это могут быть акции, опционы, фьючерсы, валюты и другие.

Управление включает в себя ожидание пользователя от системы, то есть ограничение по времени работы системы (ее быстродействие) и выделение под нее памяти, а также требования к точности выдаваемых рекомендаций.

Механизмом выполнения системы является сама информационная система и ее программный код, включающий в себя математические и логические алгоритмы (в том числе методы искусственного интеллекта – нейронные сети).

На выходе система выдает рекомендацию пользователю по покупке-продаже финансового инструмента, графики цены и технических индикаторов, экономический календарь.

Контекстная диаграмма (рисунок 3.2) позволяет увидеть общую схему действия системы, но, чтобы подробно разобраться, какие функции она осуществляет, нужно выполнить декомпозицию функции верхнего уровня.

Можно выделить пять подфункций системы (рисунок 3.3):

1. Создать базу данных. Под базой данных понимается таблица, в которой будет находиться данные о ценах и значениях индикаторов за последние 200 периодов (дней, часов...). Эти данные будут скачиваться в режиме реального времени из интернета, поэтому этот процесс управляется скоростью подключения к интернету, а механизмом его выполнения является программный код, относящийся к соединению с интернетом и скачиванию информации. На выходе этого процесса получаем 3 множества данных: обучающее (значения за 120 периодов), тестовое (значения за 70 периодов) и рабочее (значения за 10 периодов). Эти данные подаются на входы соответственно блока А2, А3, А4, поэтому с этими блоками А1 имеет отношения «выход-вход».

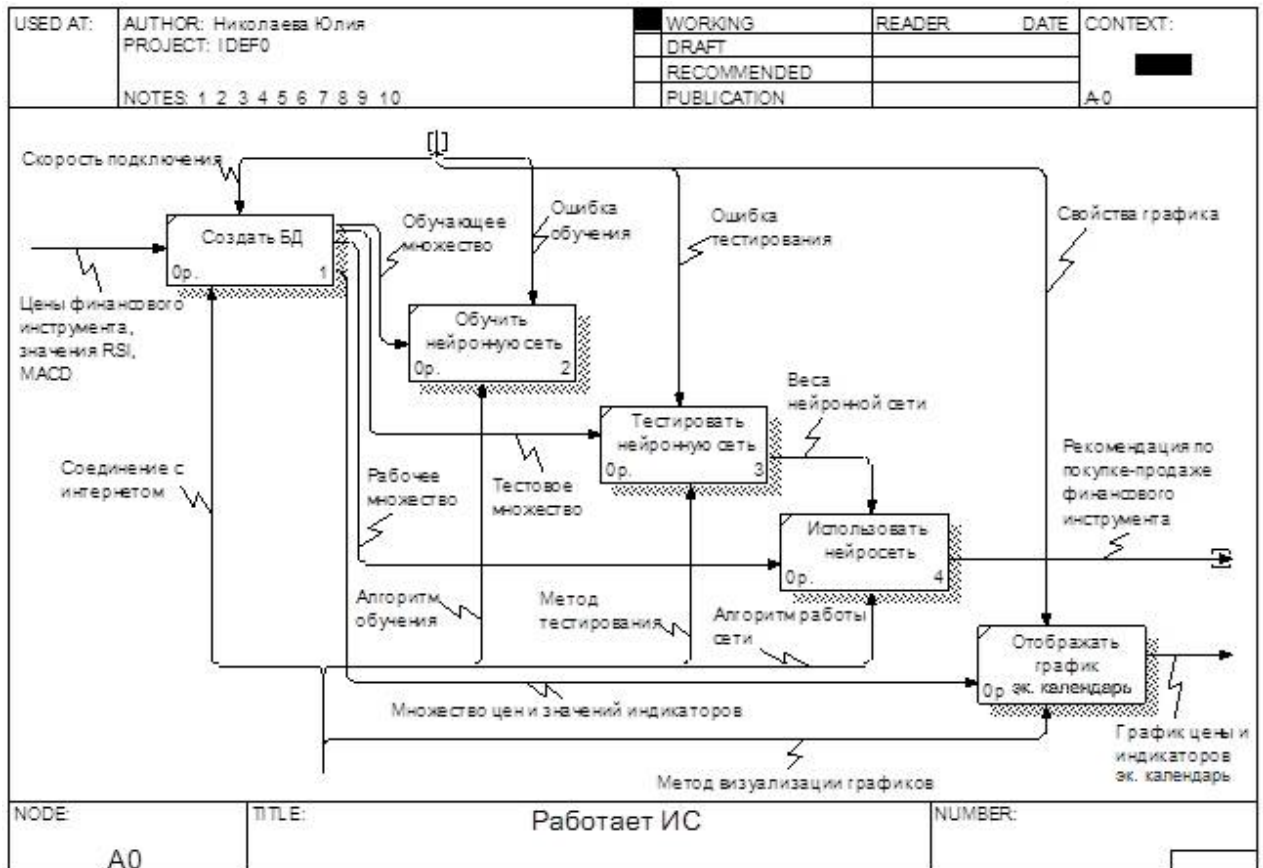


Рисунок 3.3 – Декомпозиция контекстной диаграммы A0

2. Обучить нейронную сеть. Вход этого блока – обучающее множество, выдающееся на выходе блока A1. Обучение нейронной сети происходит по алгоритму обучения, который прописан в программном коде системы под управлением ошибки обучения. В результате выполнения этого процесса получаем набор настроенных весов сети, которые подаются на входы блока A3. Отсюда следует, что отношение между блоками A2 и A3 – «выход - вход».

3. Тестировать нейронную сеть. На входы этого блока подаются тестовое множество и настроенные веса. Тестирование происходит под управлением ошибки тестирования, и выполняются по методу тестирования, описанного в программном коде. На выходе блока – проверенные веса нейронной сети, которые являются управлением для блока A4, что говорит об отношении управления между этими блоками.

4. Использовать нейронную сеть. Входом является рабочее множество, по которому трейдер хочет получить рекомендацию. Работа сети осуществляется по определенному алгоритму классификации рыночных ситуаций, описанному в программном коде, управляясь весами сети, полученными из предыдущего блока. И на выходе выдается рекомендация по покупке-продаже финансового инструмента, которую трейдер может использовать на практике.

5. Отобразить график и экономические календарь. Входом будет являться все множество значений цен и технических индикаторов, хранимое в базе данных, отсюда следует, что отно-



шения между этими блоками «выход-вход». Управлением данного блока являются свойства графиков, задаваемые пользователем, а механизмом – метод визуализации графиков, свойственный конкретной среде разработки (в данном случае среде Visual Studio). На выходе данного блока получаем готовые графики цен и значений индикаторов.

В качестве акторов (действующих персон) спроектированной ИССПР в диаграмме вариантов использования можно выделить пользователя, разработчика системы и саму систему (см. рисунок 3.4).

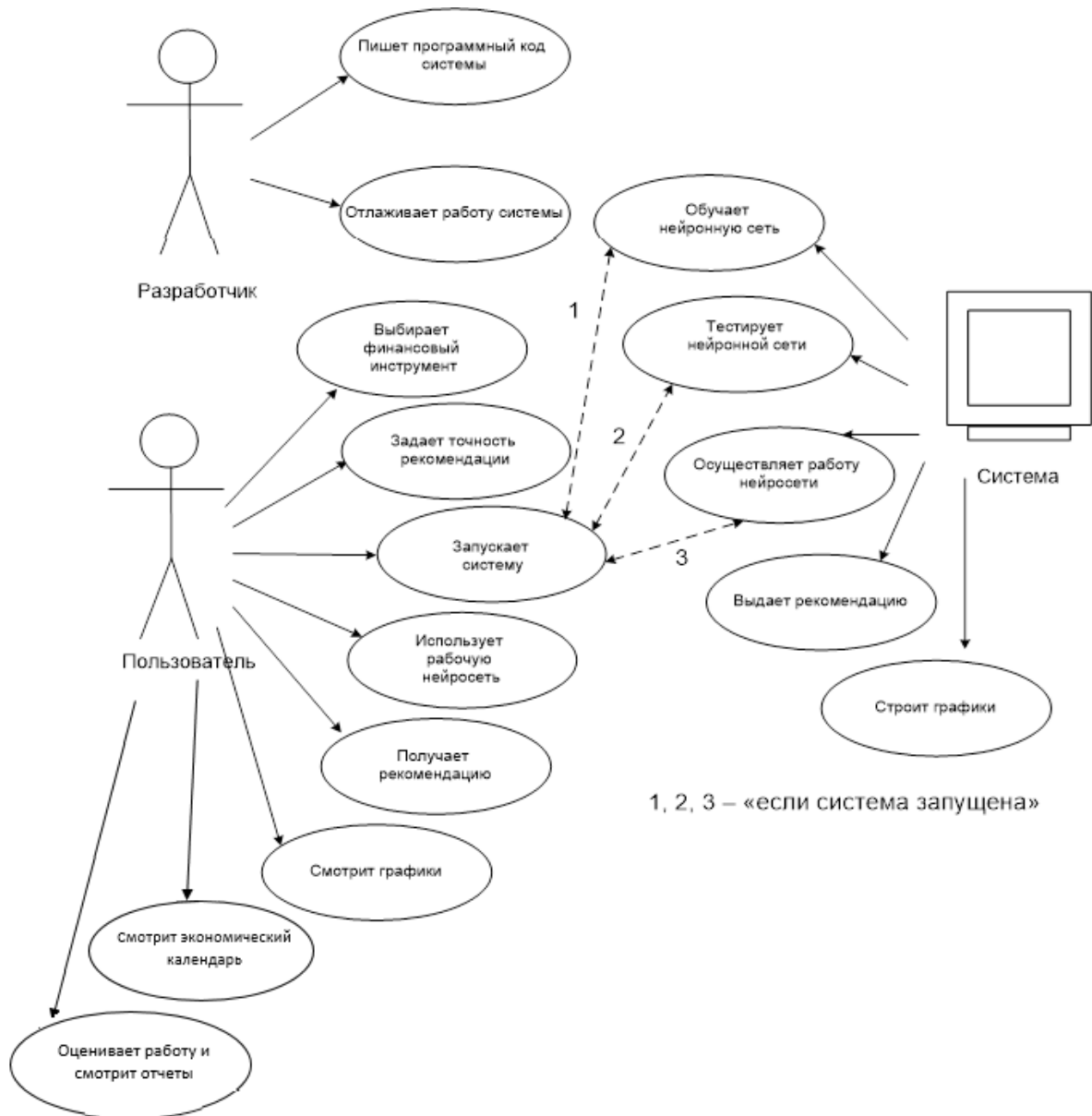


Рисунок 3.4 – Диаграмма прецедентов

Разработчик отвечает за создание системы: он пишет программный код интеллектуальной системы поддержки принятия решения и отлаживает ее. В программный код входят все необ-

ходимые алгоритмы, включая алгоритмы обучения, тестирования, работы нейронной сети, так же все компоненты интерфейса системы.

Пользователь имеет дело с интерфейсом системы. У него есть возможность выбирать из списка необходимый в данный момент финансовый инструмент, задавать ожидаемую прибыль, оценивать работу системы, пользоваться отчетами о работе системы. Так же он непосредственно запускает работу самой системы, использует рабочую нейросеть, смотрит графики цен и индикаторов, смотрит экономический календарь и получает рекомендацию по покупке-продаже выбранного финансового инструмента.

Система после ее запуска пользователем выполняет следующие функции: обучает нейронную сеть, тестирует нейронную сеть, осуществляет работу обученной и проверенной нейронной сети. В итоге, она выдает рекомендацию по покупке-продаже.

### 3.2.2 Детальное проектирование системы

Чтобы проникнуть в суть протекаемых в информационной системе процессов недостаточно только функциональных моделей, необходимо построить другие модели нотаций, отражающие динамические свойства системы.

На диаграммах IDEF0 и use-case явно не указываются ни последовательность выполнения действий, ни время. Этот недостаток в известной степени восполняет использование еще одной распространенной нотации – IDEF3 [16]. Нотация IDEF3 ориентирована на описание бизнес-процесса на основе сбора данных, необходимых для проведения структурного анализа системы. Она дополняет технологию IDEF0 [60].

С помощью IDEF3 имеется возможность уточнить картину процесса, привлекая внимание аналитика к очередности выполнения функций и бизнес-процесса в целом, выстраивая альтернативные сценарии развития изучаемых процессов (модели типа «что, если?») [19].

Диаграмма IDEF3 (рисунок 3.5) содержит 9 блоков работ, 13 связей и три перекрестка.

Начинается процесс работы информационной системы с запуска самой программы (первый блок работ). Второй блок – выбор финансового инструмента, периода, ввод ожидаемой прибыли. Следующий за ним блок отражает заполнение базы данных предыдущими ценами финансового инструмента, полученными из интернета, и выполняется строго после завершения второй работы.

Четвертый блок отражает процесс задания обучающей и тестовой ошибки. Эта работа может выполняться одновременно или даже может начаться раньше, чем заполнение БД, поэтому между ним и первым перекрестком связь отношения. Можно было отразить те же отношения с помощью распараллеливания процессов, но это означало бы введение еще одного дополнительного перекрестка, что усложнило бы визуальное восприятие диаграммы. Выходом

данного блока будут ошибки, которые подаются на вход блока обучения, поэтому между ним и следующим перекрестком связь потоков данных.

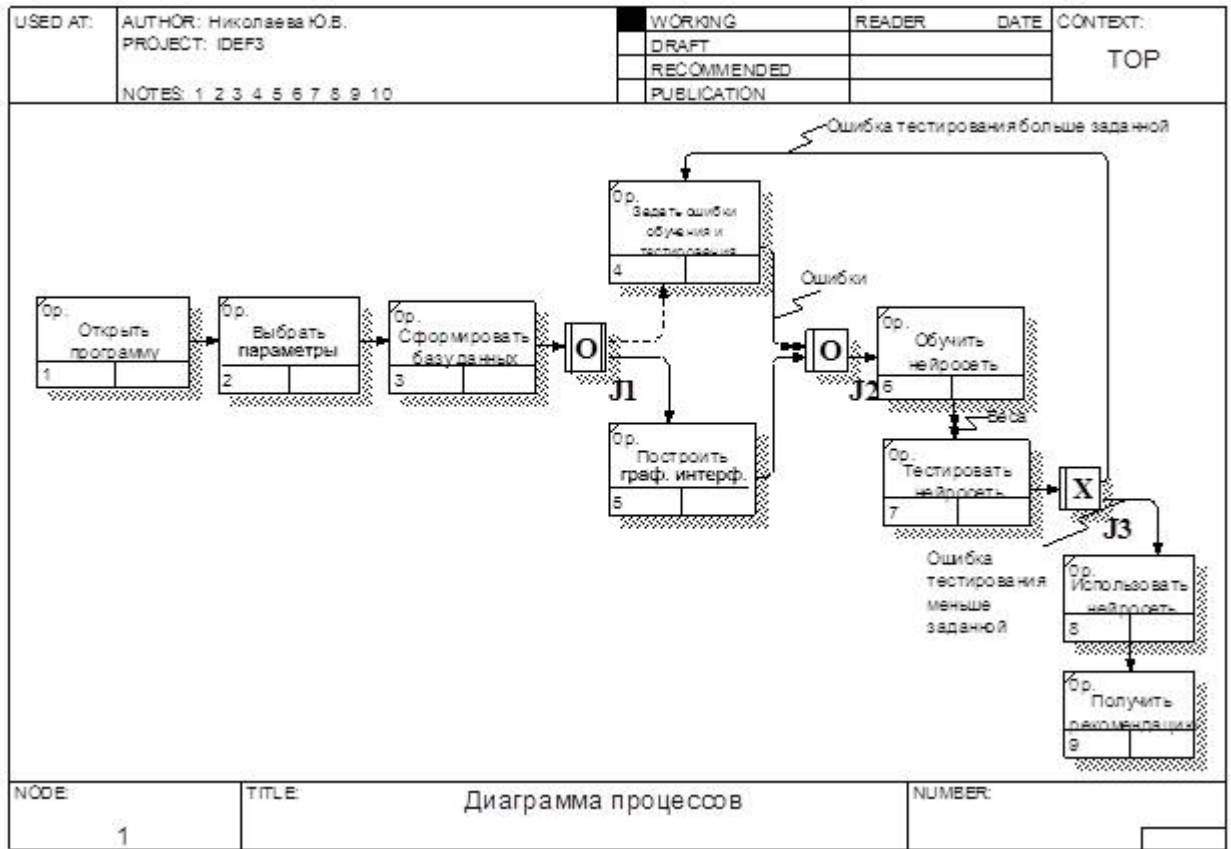


Рисунок 3.5 – Диаграмма IDEF3

Пятый блок отражает построение графических интерфейсов, интерпретаций полученных данных, в том числе графиков цен, индикаторов технического анализа и экономического календаря. Четвертый и пятый блок могут выполняться в любой последовательности.

Следующий блок обучения нейронной сети (блок работ 6). Между 6 и 7 блоками наблюдается связь потоков данных, так как выходом блока обучения являются подобранные веса, которые используются при тестировании нейросети.

После блока тестирования сети идет перекресток «исключающее или». Он означает выбрать одно из следующих действий:

- использовать нейронную сеть, если ошибка тестирования меньше заданной, то есть сеть обучена правильно и рекомендации, выдаваемые сетью, могут быть достоверны;
- перейти к блоку задания ошибок и откорректировать их, если ошибка тестирования больше заданной пользователем.

Последний блок – это получение рекомендации от нейронной сети по покупке-продаже финансового инструмента, выдаваемой обученной и протестированной нейронной сетью.

Диаграмма деятельности (*activity diagram*) – диаграмма, на которой показано разложение некоторой деятельности на её составные части [30].

Диаграммы деятельности можно использовать для моделирования динамических аспектов поведения системы. Как правило, они применяются, чтобы промоделировать последовательные (а иногда и параллельные) шаги процесса. С помощью диаграмм деятельности можно также моделировать жизнь объекта, когда он переходит из одного состояния в другое в разных точках потока управления [20]. Диаграммы деятельности могут использоваться самостоятельно для визуализации, специфицирования, конструирования и документирования динамики совокупности объектов, но они пригодны также и для моделирования потока управления при выполнении некоторой операции.

На диаграмме деятельности показывается последовательность шагов, возникающая при использовании информационно-управляющей системы.

Первым шагом является открытие программы. Второй – выбор параметров. На следующем шаге формируется база данных, этот шаг включает в себя скачивание из интернета последних цен финансового инструмента, значений технических индикаторов, экономического календаря и сохранение этих данных в определенных документах – электронной таблице и документе формата html (подробнее о вопросах разработки структуры хранения данных для ИС-ППР см. в [96]). Выходом данного шага является обучающее, тестовое и рабочее множества, данные о событиях экономического календаря, которые потом поставляются на шаг построения графических интерфейсов, поэтому между ними отношения потоков объектов. Далее идут два параллельных шага:

- построение графических интерфейсов: графиков цен и технических индикаторов, отображения событий экономического календаря;
- задание ошибок обучения и тестирования, выходом которого являются объекты (ошибки), поэтому отношения со следующим шагом обозначены потоком объектов.

Оба этих шага могут выполняться в любой последовательности. Между шагами обучения и тестирования нейросети отношение показаны тоже как поток объектов, так как выходом шага обучения являются объекты – синаптические веса нейронной сети.

После тестирования нейронной сети происходит проверка ошибки тестирования:

- если она меньше заданной величины, то переходим к следующему шагу использования нейросети;
- иначе возвращаемся к шагу задания ошибок тестирования и обучения.

И последним шагом является получение рекомендации от системы по покупке-продаже финансового инструмента и рекомендации по анализу событий экономического календаря.

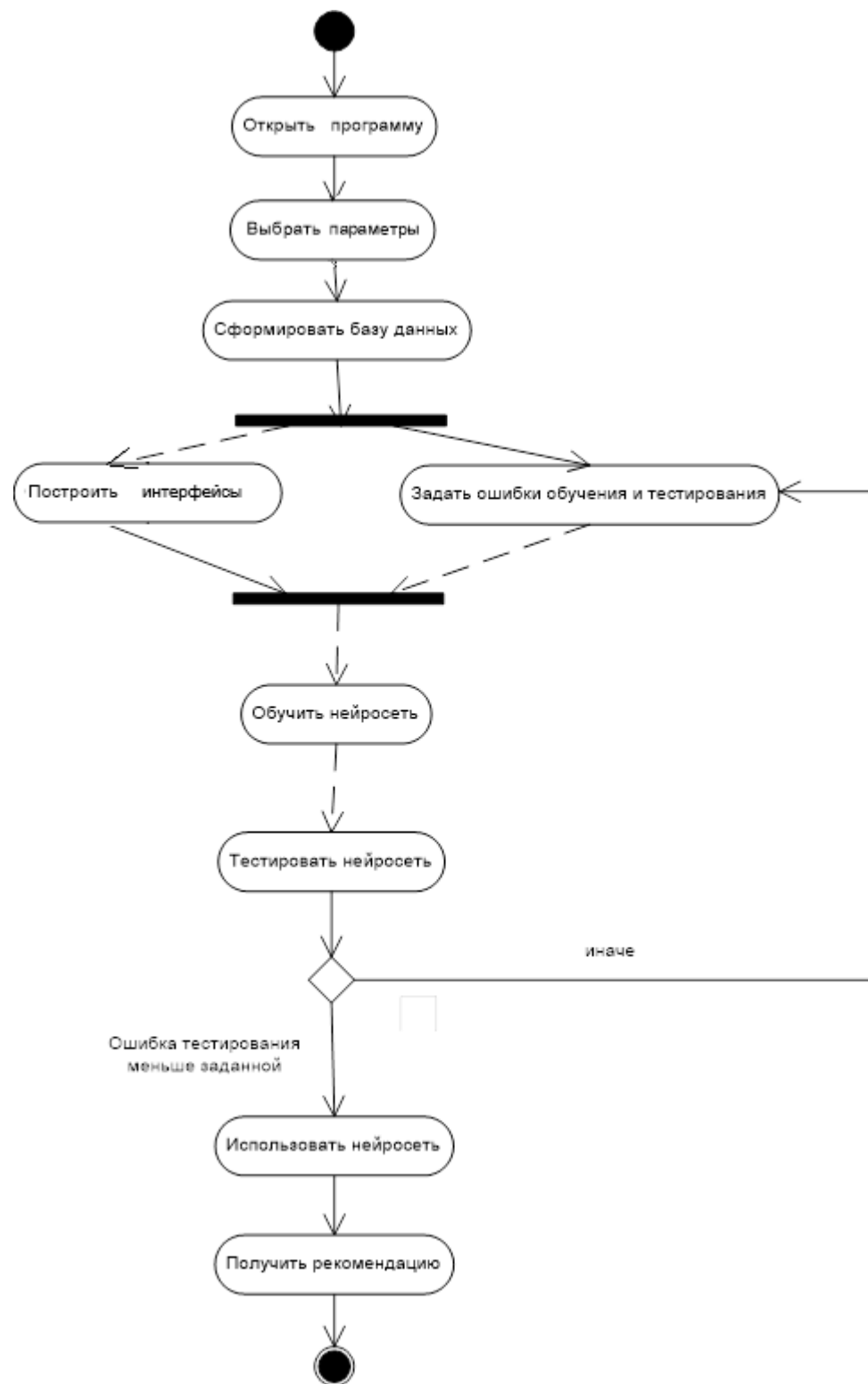


Рисунок 3.6 – Диаграмма деятельности (Microsoft Visio)



Рисунок 3.7 – Диаграмма деятельности (Rational Rose)

Выше представлены два варианта диаграммы деятельности (см. рисунки 3.6 и 3.7). Обе эти диаграммы составлены с помощью Case-средств: первая диаграмма составлена в Microsoft Visio, вторая в Rational Rose.

На первый взгляд они идентичны. Но если рассмотреть подробно, то видно, что в первой диаграмме некоторые действия соединены пунктирными стрелками, что соответствует потокам объектов. Потоки данных наблюдаются между действиями «Сформировать базу данных» и «Построить интерфейсы», «Задать ошибки обучения и тестирования» и «Обучить нейросеть», «Обучить нейросеть» и «Тестировать нейросеть».

На второй диаграмме данные связи не отображены, так как Rational Rose не дает такой возможности. Поэтому для наглядности лучше использовать для построения диаграммы действия Microsoft Visio.

Диаграмма взаимодействия – это диаграмма, на которой представлено взаимодействие, состоящее из множества объектов и отношений между ними, включая и сообщения, которыми они обмениваются [29].

Диаграммы взаимодействий используются для моделирования динамических аспектов системы. Сюда входит моделирование конкретных и прототипических экземпляров классов, интерфейсов, компонентов и узлов, а также сообщений, которыми они обмениваются, - и все это в контексте сценария, иллюстрирующего данное поведение. Диаграммы взаимодействий могут существовать автономно и служить для визуализации, специфицирования, конструирования и документирования динамики конкретного сообщества объектов, а могут использоваться для моделирования отдельного потока управления в составе прецедента [16].

Диаграммы взаимодействий важны не только для моделирования динамических аспектов системы, но и для создания исполняемых систем посредством прямого и обратного проектирования. Диаграммы взаимодействий в объектно-ориентированном подходе являются, по сути, проекцией участвующих во взаимодействии элементов.

Диаграмма последовательности – диаграмма, на которой показаны взаимодействия объектов, упорядоченные по времени их проявления. Временной аспект поведения может иметь существенное значение при моделировании синхронных процессов, описывающих взаимодействия объектов. Именно для этой цели в языке UML используются диаграммы последовательности.

На диаграмме последовательностей (рисунок 3.8) можно выделить двух актеров: система и пользователь, и она показывает их взаимодействие друг с другом.

Видно, что инициатором всего процесса является пользователь, а затем они поочередно получают фокус контроля. Пользователь выбирает финансовый инструмент, период, ожидаемую прибыль, которые будут анализироваться системой. Пользователь нажимает кнопку для запуска процесса анализа и обучения сети. Система в свою очередь загружает последние цены, вычисляет значения индикаторов, прогнозной модели, меры волатильности для этого финансового инструмента, сообщает о завершении загрузки и обработки данных пользователю, затем строит графики этих показателей. Система сообщает пользователю о наличии важных для выбранного инструмента событий в экономическом календаре. Пользователь сообщает системе о своем решении просмотреть календарь. Система отображает графический интерфейс экономических событий.

Далее система занимается обучением, тестированием и применением нейронной сети.

Во время обучения система сама себе сообщает о величине ошибки обучения и продолжает обучение до тех пор, пока эта ошибка не станет меньше заданной, после чего переходит к тестированию сети. Если ошибка тестирования приемлема, то переходит к применению нейросети и выдает рекомендацию по покупке-продаже. Но если ошибка тестирования больше заданной, то система сообщает об этом пользователю.

Завершает процесс работы информационной системы пользователь, закрывая ее.

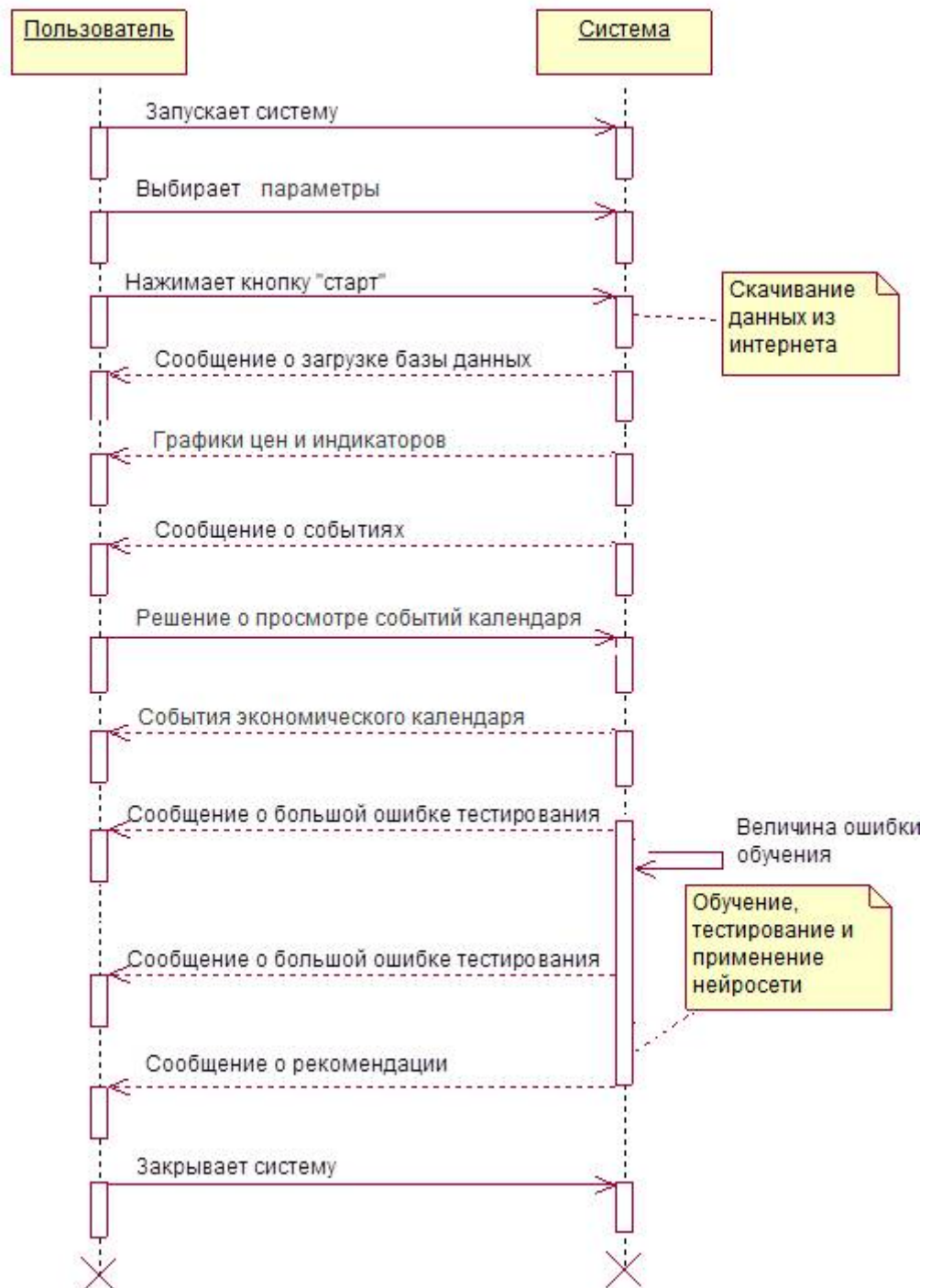


Рисунок 3.8 – Диаграмма последовательностей



В итоге моделирования системы с помощью различных средств приходим к окончательной блок-схеме функционирования системы поддержки принятия решений трейдеров на финансовых рынках (рисунок 3.9).

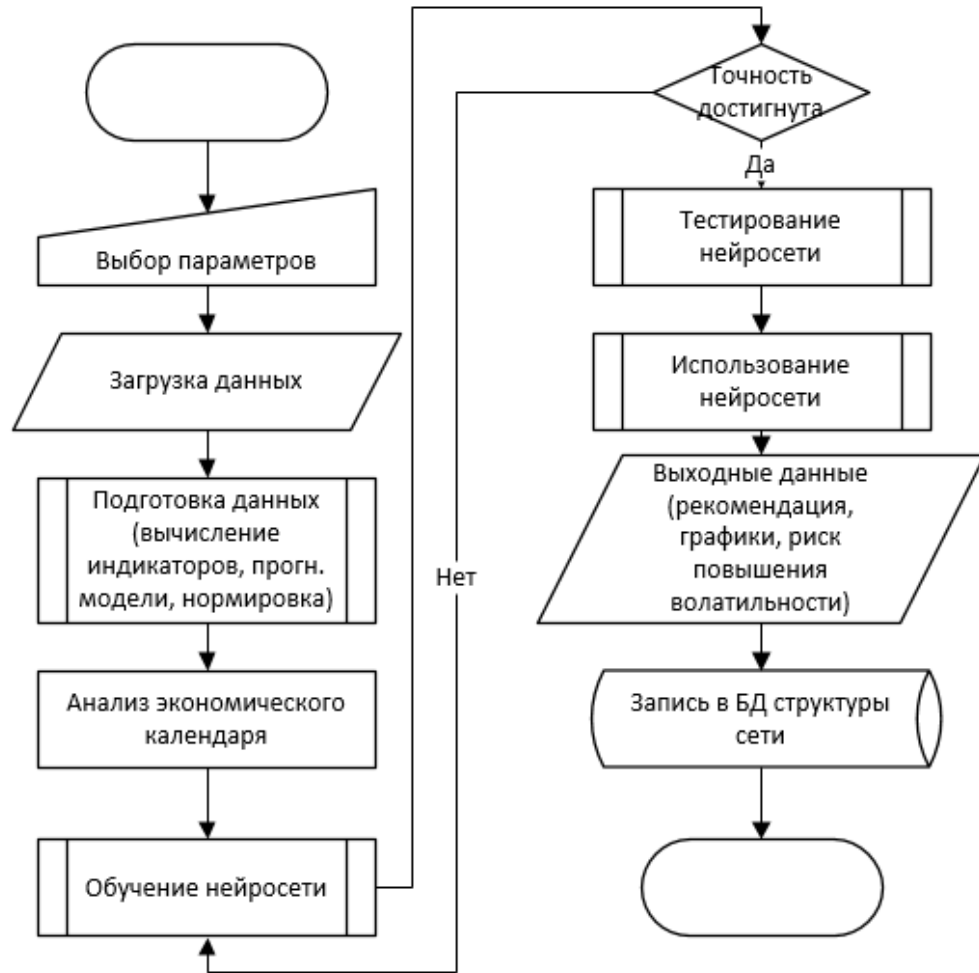


Рисунок 3.9 – Блок-схема функционирования системы

С помощью структурных case-средств, таких как VPwin и ERwin, лучше проектировать системы, в которых много вычислительных модулей и объектами являются в основном только компоненты интерфейса. Так же удобно проектирование баз данных в ErWin, если в нем поддерживается СУБД, которая будет применяться в будущем для организации запросов, отчетов и администрирования базы данных.

Объектно-ориентированные case-средства, такие как Rational Rose, удобны для проектирования систем, которые будут программироваться на объектно-ориентированных языках, например, Java или C#. В этих случаях case-средства дают возможность не только смоделировать систему, но и они генерируют ее код, что облегчает задачу программисту.

При проектировании системы «Интеллектуальная система поддержки принятия решений трейдеров на финансовых рынках» удобнее оказалось применение структурного подхода, так как в системе много вычислительных алгоритмов.

### 3.3 Проектирование базы данных

Проектирование баз данных, как правило, играет одну из ключевых ролей в большинстве проектов. Грамотно спроектированная база позволяет без особых проблем вносить изменения в структуру системы.

В базе данных отражается информация об определенной предметной области. В автоматизированных информационных системах отражение предметной области обеспечивается посредством информационной модели.

Предметная область – часть реального мира, подлежащая изучению с целью организации управления и, в конечном счете, автоматизации.

Базы данных – это специальным образом организованные данные. Эти особенности в организации данных заключаются, прежде всего, в том, что БД представляют собой системы взаимосвязанных данных, единство и целостность которых поддерживается специальными программными средствами [31].

Выделяют три уровня представления предметной области: реальность, описание (представление) реальности и данные, которые отражают это представление.

Проектирование баз данных – процесс создания схемы базы данных и определения необходимых ограничений целостности.

Основные задачи:

- Обеспечение хранения в БД всей необходимой информации.
- Обеспечение возможности получения данных по всем необходимым запросам.
- Сокращение избыточности и дублирования данных.
- Обеспечение целостности данных (правильности их содержания): исключение противоречий в содержании данных, исключение их потери и т.д.

Широкое распространение реляционных СУБД и их использование в самых разнообразных приложениях показывает, что реляционная модель данных достаточна для моделирования предметных областей.

При этом проявляется ограниченность реляционной модели данных в следующих аспектах:

- Модель не предоставляет достаточных средств для представления смысла данных. Семантика реальной предметной области должна независимым от модели способом представляться в голове проектировщика. В частности, это относится к упоминавшийся нами проблеме представления ограничений целостности.
- Для многих приложений трудно моделировать предметную область на основе плоских таблиц. В ряде случаев на самой начальной стадии проектирования проектировщику приходится

ся производить насилие над собой, чтобы описать предметную область в виде одной (возможно, даже ненормализованной) таблицы.

– Хотя весь процесс проектирования происходит на основе учета зависимостей, реляционная модель не предоставляет каких-либо средств для представления этих зависимостей.

– Несмотря на то, что процесс проектирования начинается с выделения некоторых существенных для приложения объектов предметной области (сущностей) и выявления связей между этими сущностями, реляционная модель данных не предлагает какого-либо аппарата для разделения сущностей и связей.

Основные этапы проектирования:

1. Концептуальное (инфологическое) проектирование – построение семантической модели предметной области, то есть информационной модели наиболее высокого уровня абстракции. Такая модель создаётся без ориентации на какую-либо конкретную СУБД и модель данных.

2. Логическое (даталогическое) проектирование – создание схемы базы данных на основе конкретной модели данных, например, реляционной модели данных. Для реляционной модели данных даталогическая модель – набор схем отношений, обычно с указанием первичных ключей, а также «связей» между отношениями, представляющих собой внешние ключи.

На этапе логического проектирования учитывается специфика конкретной модели данных, но может не учитываться специфика конкретной СУБД.

3. Физическое проектирование – создание схемы базы данных для конкретной СУБД. Специфика конкретной СУБД может включать в себя ограничения на именование объектов базы данных, ограничения на поддерживаемые типы данных и т.п.

На концептуальном уровне разрабатываются сущности, атрибуты, связи, этот уровень соответствует представлению аналитика.

На логическом уровне разрабатываются записи, элементы данных, связи между записями, этот уровень соответствует представлению программиста.

На физическом уровне разрабатываются группирование данных, индексы, методы доступа, этот уровень соответствует представлению администратора базы данных.

На концептуальном уровне выполняется структуризация знаний о предметной области, стандартизируется терминология. Затем строится концептуальная модель, описываемая на естественном языке. Для описания свойств и связей объектов применяют различные диаграммы [111].

Чаще всего концептуальная модель представляется в виде диаграммы сущностей – связей (entity – relationship) или ER-диаграммы.

Основными понятиями ER-модели являются сущность, связь и атрибут.

К типу методологий «Сущность-взаимосвязь» (ER – Entity-Relationship) относится методология IDEF1X построения реляционных структур (баз данных), она, как правило, используется для моделирования реляционных баз данных, имеющих отношение к рассматриваемой информационной системе [50].

Методологию IDEF1X наиболее целесообразно использовать при построении логической структуры базы данных уже после проведения исследования всех информационных ресурсов (например, используя IDEF1) и принятия решения о необходимости реляционной базы данных в информационной системе. Но не стоит забывать, что средства моделирования IDEF1X специально разработаны для построения реляционных информационных систем [26], и если существует необходимость проектирования другой системы, скажем объектно-ориентированной, то лучше избрать другие методы моделирования.

### 3.3.1 Концептуальная модель базы данных

База данных предназначена для хранения истории работы пользователя с приложением интеллектуальной системой поддержки принятия решений трейдеров финансового рынка. В таблицах хранятся статически данные о финансовых инструментах, периодах и оценках, с которыми работает пользователь в приложении, и динамически формируемые данные по операциям, выполняемым в приложении: по какому инструменту, за какой период, с какой ожидаемой прибылью строилась нейронная сеть, и какую рекомендацию выдала сеть в процессе работы. В отдельные таблицы записывается история работы нейронной сети для каждой операции: подобранные веса каждого слоя. При работе с приложением у пользователя есть возможность восстановить последние подходящие параметры нейронной сети из базы данных.

Концептуальная модель:

1. Финансовый инструмент. В этой сущности описывается используемый в приложении финансовый инструмент, его название и описание.

Ключ сущности: ID финансового инструмента.

Атрибуты сущности: название и описание.

2. Период. В этой сущности описывается используемый в приложении временной период, его название.

Ключ сущности: ID периода.

Атрибуты сущности: название.

3. Оценка. В этой сущности описывается используемая в приложении оценка, ее название.

Ключ сущности: ID оценки.

Атрибуты сущности: название.

4. Рекомендация. В этой сущности описывается используемая в приложении рекомендация, ее название.

Ключ сущности: ID рекомендации.

Атрибуты сущности: название.

5. Операция. В этой сущности хранятся данные о осуществляемой в приложении операции: по какому инструменту, с какими периодом, ожидаемой прибылью, рекомендацией и оценкой.

Ключ сущности: ID операции.

Атрибуты сущности: дата операции, ID финансового инструмента, ID периода, ID оценки, ожидаемая прибыль, ID рекомендации.

6. Нейронная сеть. В этой сущности хранятся данные о формируемой приложением по каждой операции нейронной сети.

Ключ сущности: ID нейронной сети.

Атрибуты сущности: ID операции.

7. Слой 1. В этой сущности хранятся параметры первого слоя по каждой нейронной сети.

Ключ сущности: ID слоя 1.

Атрибуты сущности: ID нейронной сети, номер нейрона второго слоя, веса каждого из 18 нейронов первого слоя, относящиеся к соответствующему нейрону второго слоя.

8. Слой 2. В этой сущности хранятся параметры второго слоя по каждой нейронной сети.

Ключ сущности: ID слоя 2.

Атрибуты сущности: ID нейронной сети, веса каждого из 3 нейронов второго слоя.

9. Слой 3. В этой сущности хранятся параметры выходного слоя по каждой нейронной сети.

Ключ сущности: ID слоя 3.

Атрибуты сущности: ID нейронной сети, значение выходного нейрона

Связи сущностей:

1. Сущности Финансовый инструмент, Период, Оценка, Рекомендация и Операция связываются по типу один ко многим. Для одного финансового инструмента, периода, оценки, рекомендации может быть проведено много операций, но каждая операция проводится только по одному финансовому инструменту, периоду, оценке, рекомендации.

2. Сущности Нейронная сеть и Операция связываются по типу один к одному. По каждой операции создается отдельная нейронная сеть.

3. Сущности Слой 1, Слой 2, Слой 3 связаны с Нейронной сетью по типу один к одному. Для каждой нейронной сети записываются значения их весовых коэффициентов и выхода в соответствующие таблицы.

Концептуальный уровень базы данных представлен на рисунке 3.10.

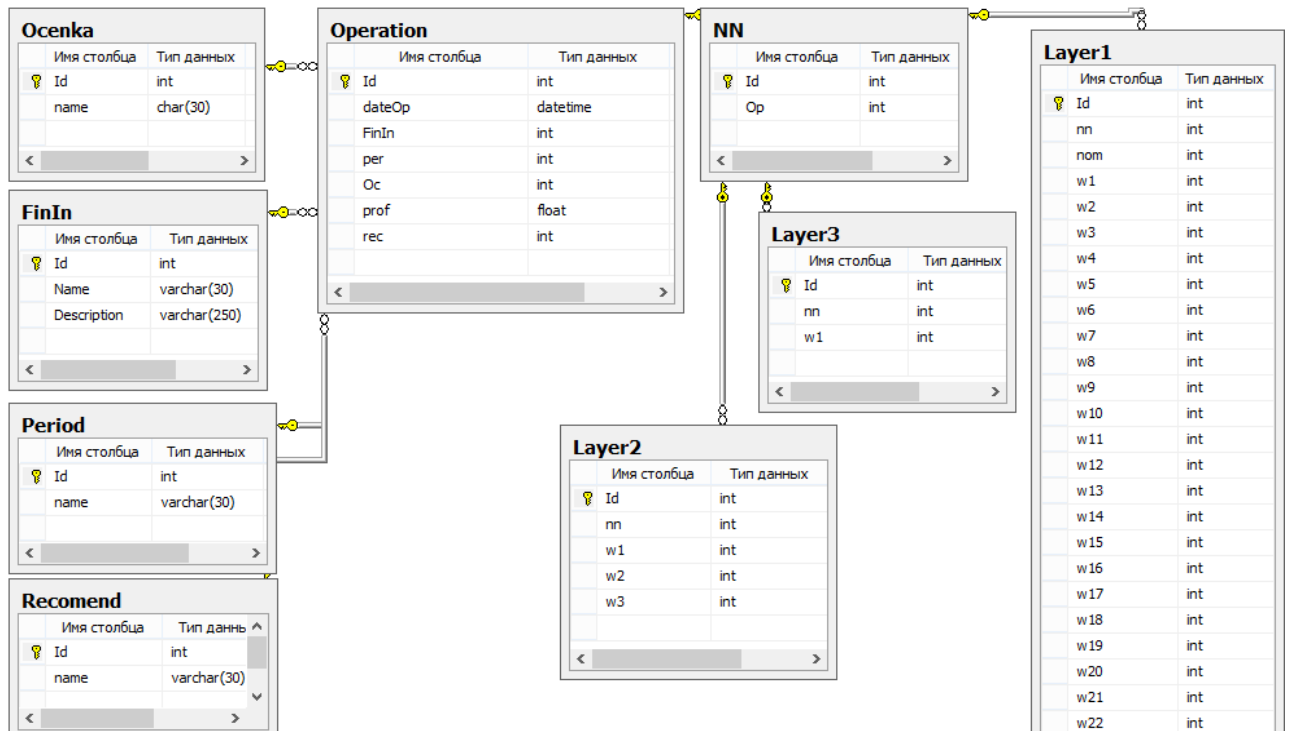


Рисунок 3.10 – Концептуальная модель базы данных

Ниже представлены примеры заполнения каждой из таблиц (таблицы 3.1 – 3.9).

Таблица 3.1. Пример заполнения таблицы Финансовый инструмент

ID	Название	Описание
1	EURUSD	Курс евро к американскому доллару
2	GBPUSD	Курс британского фунта к американскому доллару
3	USDRUR	Курс рубля к американскому доллару
4	Gazprom	Обыкновенные акции ОАО «Газпром»
5	Rosneft	Обыкновенные акции ОАО «Роснефти»
6	Brent	Курс фьючерсов нефти сорта Brent

Таблица 3.2. Пример заполнения таблицы Период

ID	Название
1	15 min

Продолжение таблицы 3.2

2	30 min
3	1 hour
4	1 day
5	1 week

Таблица 3.3. Пример заполнения таблицы Оценка

ID	Название
1	2
2	3
3	4
4	5

Таблица 3.4. Пример заполнения таблицы Рекомендация

ID	Название
1	Buy
2	Sell
3	Wait

Таблица 3.5. Пример заполнения таблицы Операции

ID	Дата операции	ID фин. инструм.	ID периода	ID оценки	Ожид. прибыль	Рекомендация
123	04.06.2016 18:05:34	EURUSD	1	2	0.001	1
124	05.06.2016 12:08:37	Газпром	2	3	4.1	2
125	12.06.2011	EURUSD	2	4	0.0009	1

Таблица 3.6. Пример заполнения таблицы Нейронная сеть

ID	ID операции
123	123
124	124
125	125

Таблица 3.7. Пример заполнения таблицы Слой 1

ID	ID нейронной сети	Номер нейрона слоя 2	вес 1	вес2.. вес21	вес 22
369	123	0	4.46197	...	-2.15476
370	123	1	5.46216	...	-8.12456
371	123	2	-9.45143	...	1.45671

Таблица 3.8. Пример заполнения таблицы Слой 2

ID	ID нейронной сети	вес 1	вес 2	вес 3
123	123	1.46197	4.44497	-4.46197
124	124	-20.46197	1.46178	-14.46166
125	125	-17.46197	2.46197	-50.46194

Таблица 3.9. Пример заполнения таблицы Слой 3

ID	ID нейронной сети	выход
123	123	0.99987
124	124	0.12465
125	125	0.44564

На основе концептуальной модели создается логическая. Логическая модель данных является визуальным представлением структур данных, их атрибутов и бизнес-правил. Разработчик модели использует требования к данным и результаты анализа для формирования логической модели данных.

### 3.3.2 Логическая модель базы данных

Логическая модель данных является источником информации для физического проектирования. Физическое проектирование базы данных предусматривает принятие решения о способах реализации модели на основе конкретной СУБД.

Система управления базами данных (СУБД) – совокупность программных и лингвистических средств общего или специального назначения, обеспечивающих управление созданием и использованием баз данных [94].

Обычно современная СУБД содержит следующие компоненты:

– ядро, которое отвечает за управление данными во внешней и оперативной памяти, и журнализацию,



- процессор языка базы данных, обеспечивающий оптимизацию запросов на извлечение и изменение данных и создание, как правило, машинно-независимого исполняемого внутреннего кода,
- подсистему поддержки времени исполнения, которая интерпретирует программы манипуляции данными, создающие пользовательский интерфейс с СУБД
- сервисные программы (внешние утилиты), обеспечивающие ряд дополнительных возможностей по обслуживанию информационной системы.

По способу доступа к базам данных все СУБД делятся на три категории:

1. **Файл-серверные.** В файл-серверных СУБД файлы данных располагаются централизованно на файл-сервере. СУБД располагается на каждом клиентском компьютере. Доступ СУБД к данным осуществляется через локальную сеть. Преимуществом этой архитектуры является низкая нагрузка на ЦП сервера. Недостатки: потенциально высокая нагрузка локальной сети; затруднённая централизованного управления; затруднённая обеспечения таких важных характеристик как высокая надёжность, высокая доступность и высокая безопасность. Применяются чаще всего в локальных приложениях, которые используют функции управления БД.

На данный момент файл-серверная технология считается устаревшей.

Примеры: Microsoft Access, Paradox, dBase, FoxPro, Visual FoxPro.

2. **Клиент-серверные.** Клиент-серверная СУБД располагается на сервере вместе с БД и осуществляет доступ к БД непосредственно, в монопольном режиме. Все клиентские запросы на обработку данных обрабатываются клиент-серверной СУБД централизованно.

Недостатком клиент-серверных СУБД являются повышенные требования к серверу. Достоинства: потенциально более низкая нагрузка локальной сети; удобство централизованного управления; удобство обеспечения таких важных характеристик как высокая надёжность, высокая доступность и высокая безопасность.

Примеры: Oracle, Firebird, Interbase, IBM DB2, Informix, MS SQL Server, Sybase Adaptive Server Enterprise, PostgreSQL, MySQL, Caché, ЛИНТЕР.

3. **Встраиваемые.** СУБД, поставляемая как составная часть некоторого программного обеспечения, не требующая самостоятельной установки. Встраиваемая СУБД предназначена для локального хранения данных своего приложения и не рассчитана на массовое использование. Реализована физически как подключаемая библиотека. Доступ к данным предоставляется приложением через программные интерфейсы, например, через sql.

Примеры: OpenEdge, SQLite, BerkeleyDB, Firebird Embedded, Sav Zigzag, Microsoft SQL Server Compact, ЛИНТЕР.

Выбор СУБД представляет собой сложную многопараметрическую задачу и является одним из важных этапов при разработке приложений баз данных. Выбранный программный продукт должен удовлетворять как текущим, так и будущим потребностям системы, разработчика и пользователя.

Критерии отбора СУБД [6]:

- Моделирование данных
- Особенности архитектуры и функциональные возможности
- Контроль работы системы
- Особенности разработки приложений
- Производительность
- Надежность
- Требования к рабочей среде
- Смешанные критерии

Рассмотрим подробнее основные критерии.

#### 1. Моделирование данных

Используемая модель данных. Существует множество моделей данных; самые распространенные: иерархическая, сетевая, реляционная, объектно-реляционная и объектная. Вопрос об использовании той или иной модели должен решаться на начальном этапе проектирования информационной системы.

При проектировании системы будет использоваться реляционная модель данных.

Триггеры и хранимые процедуры. Триггеры обеспечивают проверку любых изменений на корректность, прежде чем эти изменения будут приняты. Хранимая процедура – программа, которая хранится на сервере и может вызываться клиентом. Поскольку хранимые процедуры выполняются непосредственно на сервере базы данных, обеспечивается более высокое быстродействие, нежели при выполнении тех же операций средствами клиента БД.

В проектируемой системе не будет необходимости в триггерах.

Средства поиска. Некоторые современные системы имеют встроенные дополнительные средства контекстного поиска.

Поиск не важен в данной информационной системе.

Предусмотренные типы данных. Здесь следует учесть два фактически независимых критерия: базовые или основные типы данных, заложенные в систему, и наличие возможности расширения типов. В то время как отклонения базовых наборов типов данных у современных систем от некоего стандартного, обычно, невелики, механизмы расширения типов данных в системах того или иного производителя существенно различаются.

Типы данных, используемые в системе: темпоральный (дата-время) и вещественный (число с несколькими знаками после запятой), строковый поддерживаются большинством СУБД.

Реализация языка запросов. Все современные системы совместимы со стандартным языком доступа к данным SQL-92, однако многие из них реализуют те или иные расширения данного стандарта.

В системе будут осуществляться единичные запросы на чтение, изменение, запись данных, но они будут не столь масштабны.

## 2. Особенности архитектуры и функциональные возможности

Мобильность. Мобильность – это независимость системы от среды, в которой она работает. Средой в данном случае является как аппаратура, так и программное обеспечение (операционная система).

Мобильность является достаточно важным фактором, так как база данных будет привязана к среде разработки Visual Studio, которая имеет возможности применения на разных платформах.

Масштабируемость. При выборе СУБД необходимо учитывать, сможет ли данная система соответствовать росту информационной системы, причем рост может проявляться в увеличении числа пользователей, объема хранимых данных и объеме обрабатываемой информации.

Число пользователей, объем хранимых данных строго определены, поэтому масштабируемость не играет роли в выборе СУБД для конкретной информационной системы.

Распределенность. Распределение баз данных на различные сервера, распределение данных между базами. Различные системы имеют разные возможности управления распределенными базами данных.

Вся база данных будет централизованно храниться на компьютере пользователя.

Сетевые возможности. Многие системы позволяют использовать широкий диапазон сетевых протоколов и служб для работы и администрирования.

Важна возможность поддержания работы с сетевыми сервисами, так как база данных будет скачиваться из интернета с сайтов, на которых возможен экспорт котировок финансовых инструментов и обращаться по средствам веб-запросов за данными экономического календаря.

## 3. Контроль работы системы

Контроль использования памяти компьютера. Система может иметь возможность управления использованием как оперативной памяти, так и дискового пространства. Во втором случае это может выражаться, например, в сжатии баз данных, или удалении избыточных файлов.

В проектируемой базе данных изначально будет фиксированное использование памяти компьютера и будет отсутствовать избыточность.

#### 4. Особенности разработки приложений

Многие производители СУБД выпускают также средства разработки приложений для своих систем. Как правило, эти средства позволяют наилучшим образом реализовать все возможности сервера, поэтому при анализе СУБД стоит рассмотреть также и возможности средств разработки приложений.

Средство разработки системы уже выбрано, им является среда разработки Visual Studio.

Средства проектирования. Некоторые системы имеют средства автоматического проектирования, как баз данных, так и прикладных программ. Средства проектирования различных производителей могут существенно различаться.

При проектировании системы были использованы такие средства, как ERWin, BPwin, RationRose. Эти средства поддерживают кодогенерацию на различные языки программирования и различные СУБД.

Многоязыковая поддержка. Поддержка большого количества национальных языков расширяет область применения системы и приложений, построенных на ее основе.

Языком, используемым в системе, является только русский.

Возможности разработки Web-приложений. При разработке различных приложений зачастую возникает необходимость использовать возможности среды Internet. Средства разработки некоторых производителей имеют большой набор инструментов для построения приложений под Web.

Это свойство СУБД было бы полезно для обеспечения возможности расширения системы в сторону веб-сервиса.

Поддерживаемые языки программирования. Широкий спектр используемых языков программирования повышает доступность системы для разработчиков, а также может существенно повлиять на быстродействие и функциональность создаваемых приложений.

С языком программирования C# совместима работа с большим количеством различных СУБД, такими как Microsoft Access, Microsoft SQL Server, MySQL, Oracle.

#### 5. Производительность

Возможности параллельной архитектуры. Для обеспечения параллельной обработки данных существует, как минимум, два подхода: распараллеливание обработки последовательности запросов на несколько процессоров, либо использование нескольких компьютеров-клиентов, работающих с одной БД, которые объединяют в так называемый параллельный сервер.

В системе не предусматривается параллельная обработка данных.

Возможности оптимизирования запросов. При использовании непроецедурных языков запросов их выполнение может быть неоптимальным. Поэтому необходимо произвести процесс

оптимизации запросов, т.е. выбрать такой способ выполнения, когда по начальному представлению запроса путем его синтаксических и семантических преобразований вырабатывается процедурный план выполнения запроса, наиболее оптимальный при существующих в базе данных управляющих структурах.

После рассмотрения всех основных факторов подбора системы управления базами данных становится ясным, что большинство функций по управлению базой данных являются не столь сложными и для их координации не требуется полномасштабной системы. Вполне достаточной для такого случая может быть СУБД MySQL, так как у нее простая интеграция с Visual Studio по средствам самой платформы и настраиванию необходимых динамических библиотек. К тому же данная СУБД является свободно распространяемой, что положительно сказывается на стоимости проектируемой системы поддержки принятия решений. Так же MySQL в отличие от других клиент-серверных СУБД имеет не большой объем, что уменьшает требования к рабочим станциям, на которых в дальнейшем будет использоваться ИСППР. Поэтому, для исключения усложнения взаимодействия систем и потребления лишней памяти, решено применять СУБД MySQL 5.1.

### **Выводы по главе**

В данной главе рассмотрены вопросы проектирования системы поддержки принятия решений трейдеров финансовых рынков.

1. Приведено обоснование и критерии выбора платформы разработки, языка программирования, системы управления базами данных. Описана архитектура системы, шаги функционального и детального проектирования информационной системы типа системы поддержки принятия решений. При функциональном проектировании систем использованы нотации IDEF0 контекстная диаграмма и ее декомпозиция, диаграммы прецедентов. Для детального проектирования использованы нотации IDEF3 и UML, приведено описание бизнес-процесса на диаграмме процессов, показано разложение деятельности, производимой с ИСППР, на ее составные элементы на диаграмме деятельности. На диаграмме последовательностей показано упорядоченное по времени взаимодействие объектов с системой.

2. Проектирование СППР удобно вести с помощью специализированных программных средств, таких как Microsoft Visio, Rational Rose, BPwin, ERwin, Microsoft Visual Studio. Структурные case-средства, такие как BPwin и ERwin, используются для проектирования систем с большим количеством вычислительных элементов. Объектно-ориентированные case-средства, например, Rational Rose, средства, предоставляемые платформой Microsoft Visual Studio, позво-

ляют не только спроектировать систему, но и могут сгенерировать код для ее реализации на объектно-ориентированных языках, например, C#.

3. Рассмотрены достоинства и недостатки различных систем управления базами данных. Приведены концептуальная и логическая базы данных системы, структура и примеры заполнения таблиц баз данных для хранения информации по финансовым инструментам, операциям в системе, сохраняемым весам нейронной сети.

4. Даны практические рекомендации по проектированию системы поддержки принятия решений, разработке базы данных, обеспечивающих их работу. Для реализации системы выбрана платформа Microsoft Visual Studio, язык программирования C#, СУБД MySQL.

## 4 РЕАЛИЗАЦИЯ ИСППР И РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТОВ

### 4.1 Сравнение с аналогами ИСППР «НейроПрофит»

Распространенными СППР для анализа данных финансовых рынков и поддержки принятия решений на финансовых рынках являются торговые платформы, торговые советники: MetaTrader (Кипр), MetaStock (США), QUIK (Россия, Новосибирск), TSLab (Россия, Москва), Wall Street Forex Robot Evolution (США). Характеристик и возможности систем представлены в таблице 4.1.

Таблица 4.1 Сравнительные характеристики СППР для финансовых рынков

Возможность	MetaTrader	MetaStock	QUIK	TSLab	Wall Street Forex Robot Evolution
Технический анализ	Есть	Есть	Есть	Есть	Есть
Просмотр экономических событий	Есть	Нет	Нет	Есть	Нет
Учет влияния экономических событий	Ограничено	Нет	Нет	Нет	Нет
Указание ожидаемой прибыли	Ограничено	Ограничено	Нет	Есть	Нет
Ориентированность на конкретный финансовый инструмент\таймфрейм	Нет	Частично	Частично	Нет	Есть
Написание собственных скриптов	Ограничено	Есть	Есть	Есть	Нет
Ручное заключение сделки	Есть	Есть	Есть	Нет	Есть

Большинство брокерских систем представляют возможности по анализу различных технических индикаторов, при этом их выбор и соотношение между собой остается на усмотрение самого пользователя. Практически все системы предлагают работу на различных временных интервалах (таймфреймах) и с различными финансовыми инструментами, такими как валютные пары, акции, фьючерсы, но также есть и системы, ориентированные на работу с конкретными группами финансовых инструментов, например, финансовые советники.

В некоторых существующих системах отображаются поступающие новости экономического календаря, но нет возможности просмотра планируемых к публикации в ближайшее время экономических событий и нет никакого анализа рисков, связанных с выходом экономических новостей и событий.

Некоторые системы предоставляют инструменты для написания пользователем собственных аналитических инструментов, торговых стратегий и даже торговых роботов, но в данном случае вся ответственность за выдачу рекомендаций такими системами остается на самом пользователе. Группа систем, такие как механические торговые системы (например, TSLab), наоборот предлагают уже готовые торговые стратегии и сами заключают сделки по заранее заданным алгоритмам, полностью исключая пользователя (трейдера) из процесса принятия решения о необходимости заключения сделки, ее направления и анализа рисков, связанных с ее заключением.

Довольно серьезным недостатком текущих систем является отсутствие возможности учета ожидаемой пользователем прибыли от заключаемой сделки. В торговых платформах есть возможность указания уровня прибыли (тейк-профит), по наступлению указанного уровня цены сделка будет автоматически закрыта системой, но в данном случае уровень прибыли указывается самим пользователем на основе интуитивной торговли, не учитывается при определении момента открытия сделки и не влияет на анализ направления открытия сделки. В большинстве случаев системы предлагают только сигналы на вход в сделку (например, различные комбинации сигналов технических индикаторов), но пользователю остается самому решать, как долго ему можно оставаться в открытой сделке и по достижению какой прибыли необходимо закрывать сделки. Это также приводит к увеличению количества убыточных сделок, поскольку правильный момент выхода из сделки может быть пропущен пользователем.

Также нейронные сети довольно часто используются аналитиками финансовых рынков при реализации узкоспециализированных частных систем поддержки принятия решений. Зачастую нейронные сети в таких системах используются в качестве прогнозной модели, которая выдает в качестве ответа одно или несколько прогнозных будущих значений котировок анализируемого финансового инструмента. Распространенное наименование данного подхода к прогнозированию рынка – «метод окна» [63, 7]. В такой трактовке подбор входной и выходной выборки для обучения нейронной сети не является затруднительным: входами сети в данном случае будут цены (или приращения цен) финансового инструмента, а выходами одно или несколько следующих значений во временном ряду котировок финансового инструмента. Данный подход к нейросетевому анализу финансовых рынков позволяет трейдеру получить знания о последующих ценах конкретного финансового инструмента и на этой основе скорректировать



свою торговую стратегию, то есть принять решение по открытию сделки на покупку или продажу финансового инструмента. Трейдер ориентируется только на предполагаемые будущие значения цен, которые могут быть изменчивы под влиянием различных сиюминутных или значимых событий в макроэкономике, мировой политике. Также трейдеру по полученным данным необходимо провести дальнейший анализ, чтобы выяснить, каково направление тренда и насколько выгодно в данной ситуации заключить сделку в том или ином направлении. Таким образом, системы, основанные на нейросетевом прогнозировании временного ряда, являясь системой анализа данных, выдают в итоге пользователю в качестве ответа данные, которые трейдер вынужден самостоятельно повторно анализировать. Помимо недостаточно достоверных прогнозных данных (процент верно спрогнозированных значений цен колеблется от 58 до 77) [22], в данном случае трейдер не получает дополнительно важных критериев: как долго ему оставаться в открытой сделке или при получении какой прибыли сделку необходимо закрывать.

Некоторые аналитики финансовых рынков реализуют системы для прогнозирования временных рядов на основе нейронных сетей, используя программные продукты для их построения. Можно выделить класс не узко специализированных программ для построения нейронных сетей: NeuroSolutions, Neuroph, OpenCV, STATISTICA – позволяют в визуальном режиме проектировать нейросети.

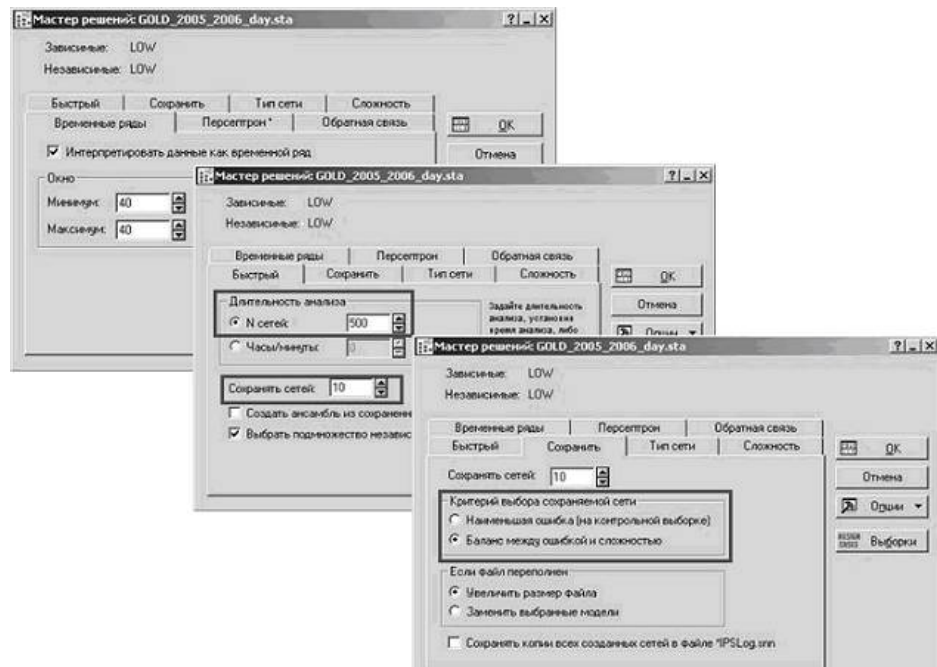


Рисунок 4.1 – Этапы работы мастера решений STATISTICA

Например, при проектировании нейронной сети для прогнозирования цен на золото в распространенном в России статистическом пакете STATISTICA [86], параметры нейронной сети (количество слоев, количество нейронов) задаются пользователем вручную. Пользователь тако-

го нейросетевого приложения становится экспериментатором и, основываясь только на эмпирических результатах работы нейронных сетей с теми или иными параметрами, выбирает модель для прогнозирования. Позиции динамического определения пользователем структуры весов или выбора наилучшей сети из комитета нейронных сетей придерживаются и прочие авторы работ по тематике прогнозирования временных рядов на основе нейронных сетей [63, 7, 1]. Автору данной работы видится наиболее логичным нахождение параметров нейронной сети по определенным научно обоснованным правилам до ее построения в момент использования системы, данный подход реализован в ИСППР «НейроПрофит».

У рассмотренных СППР в большинстве случаев отсутствует возможность учета влияния экономических событий и указания ожидаемой прибыли. Также в большинстве данных систем отсутствует возможность агрегирования данных и обобщенного анализа разнородных показателей. Все перечисленные характеристики закрываются использованием методов и алгоритмов внутри комбинированной методики нейросетевой классификации, послужившей основой для проектирования ИСППР, и включены в разрабатываемую систему ИСППР «НейроПрофит».

На основе разработанного алгоритма реализации методики классификации рыночных ситуаций в СППР и используя проектные решения (диаграммы функционирования и использования, модель базы данных) была разработана интеллектуальная система поддержки принятия решений трейдеров финансовых рынков «НейроПрофит».

## **4.2 Разработка ИСППР «НейроПрофит»**

Разработка программного обеспечения проводилась на основе спроектированных моделей, представленных в предыдущей главе. Система «НейроПрофит» разработана на платформе Microsoft Visual Studio, на языке программирования C#. При реализации программы для взаимодействия с базой данных использовалась СУБД MySQL. С примерами исходных кодов вычислительных алгоритмов системы можно ознакомиться в Приложении Г.

### **4.2.1 Пользовательские интерфейсы**

Интерфейс – совокупность средств, методов и правил взаимодействия (управления, контроля и т. д.) между элементами системы [37].

Интерфейс может состоять из компонентов аппаратного и программного обеспечения, посредством которых ведется управление системой, и с помощью которых пользователи получают необходимую информацию о ее состоянии. Компоненты аппаратного обеспечения включают устройства ввода данных и печатающие устройства (например, клавиатура, тачпад, мышь), информационно-отображающие устройства (например, экраны, динамики, наушники), а также документацию по системе и инструкции по эксплуатации для пользователей. Компоненты про-

граммного обеспечения включают командное меню, пиктограммы, окна, сообщения, предупреждения, системы передвижения, информационную обратную связь и т.п. Компоненты программного и аппаратного обеспечения могут быть настолько тесно связаны, что являются неразделимыми (например, функциональные клавиши на клавиатурах).

Разработка пользовательского интерфейса занимает более 40% времени, требуемого на разработку самой программы. А код, связанный с описанием интерфейса, занимает около 35-50% всего программного кода.

При хорошо разработанном пользовательском интерфейсе возникает доступность возможностей системы для широкого круга пользователей, уменьшение затрат на обучение пользователя, как следствие уменьшение стоимости самого продукта, уменьшение продуктивности использования системы, а также вероятность ошибок системы из-за неверного понимания пользователем [90].

После создания ассоциативной модели или метафоры интерфейса необходимо тщательно продумать и разработать концептуальный дизайн. Нужно создать систему интерфейсных элементов, алфавит взаимодействия программы и пользователя. Каждый из этих элементов должен обладать определенными цветом, формой, надписью и т. п., и все вместе они должны объединяться в единую систему, вызывающую стойкую систему ассоциаций у пользователей [46].

Немаловажную роль в восприятии интерфейса играет цвет его элементов. Привычным по ОС Windows для пользователя являются приглушенные серые оттенки, которые применяются в большинстве приложений.

С точки зрения психологии, светло-серый действует очень легко, выражает свободу и возвышенное психоэнергетическое состояние, готовность к контактам, хорошо успокаивает, умиротворяет (стремится к белому).

Присутствие серого цвета в элементах интерфейса дают пользователю уверенность в надежности и стабильности данного продукта, настраивают его на активное взаимодействие и изучение функциональности и возможностей системы.

Самым спокойным является белый цвет, он вызывает собой положительные ассоциации, т.к. представляется малое количество цвета. Белый цвет характеризуется совершенством и завершенностью, демонстрирует абсолютное и окончательное решение, полная свобода для возможностей и снятие препятствий.

Избыток белого цвета в интерфейсе вызывает у пользователя отрешенные чувства, не вызывающие у него желания взаимодействовать с системой.

Черный всегда скрывает в себе все, что несет, то есть он «загадочен». Черный цвет связан с любопытством, он притягивает к себе и вызывает интерес пользователя. Поэтому информа-

тивная составляющая автоматизированных систем обычно представляется именно этим цветом, все текстовые составляющие бывают черными [87].

Наряду с базовыми цветами: белым, черным и серым – в интерфейсе должны применяться яркие оттенки в качестве акцентов. Важные элементы интерфейса должны выделяться яркими рамками, подчеркиваниями, элементы, на которые пользователю необходимо обратить особое внимание могут сами иметь яркую окраску.

Красный цвет заставляет насторожиться при опасности, символизирует страсть, возбуждает страстность, ему свойственна максимальность в чувствах. Этот цвет заставляет быть активным во всем, воодушевляет; олицетворяет победу. Поэтому элементы, такие как предупреждения, могут быть выделены красным цветом.

Синий цвет – это постоянство, упорство, настойчивость, преданность, самоотверженность, серьезность, строгость. Под воздействием этого цвета у человека уменьшается уровень тревожности, снижается напряжение и кровяное давление. При слишком долгом воздействии возникают утомление, усталость, успокаивающее действие переходит в угнетающее, способствует затормаживанию функций физиологических систем человека. Насыщенный синий – корпоративный цвет. Он профессиональный, достойный, официальный, уважаемый, интеллигентный, дорогой и сложный [89].

Основное окно разработанной интеллектуальной системы поддержки принятия решений выполнено в базовых цветах, основой которых является умеренно серый, представляет собой разделенное на три логических блока (см. рисунок 4.2):

- 1 - блок общего взаимодействия пользователя с системой;
- 2 - блок ввода информации, запрашиваемой системой у пользователя;
- 3 - блок вывода информации системой.

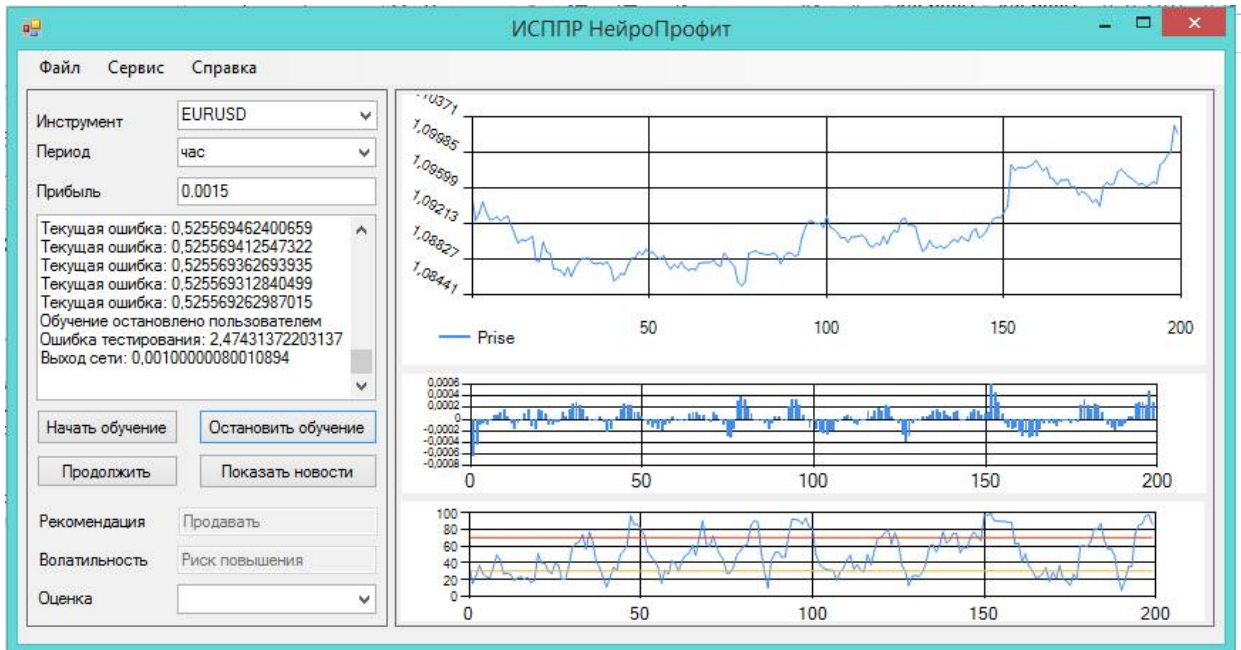


Рисунок 4.2 – Главное окно пользовательского интерфейса ИСППР

Блок общего взаимодействия представляет основное меню (рисунок 4.3), где пользователь может выбрать действия по сохранению и загрузки нейросети, также получить отчетность по работе системе и необходимую справочную информацию информации.

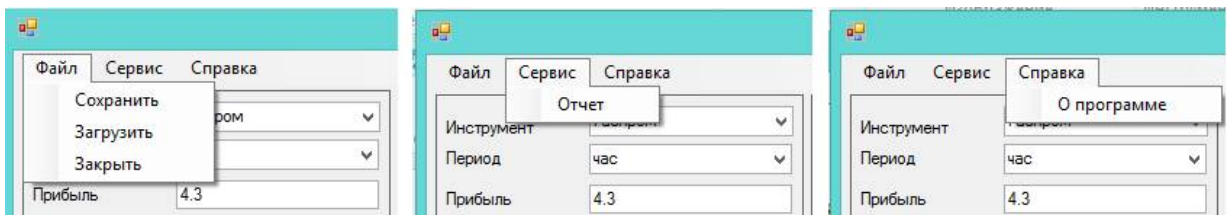


Рисунок 4.3 – Основное меню

Блок ввода информации представляет собой панель, на которой пользователь может выбрать параметры, с которыми система будет работать (рисунок 4.4).

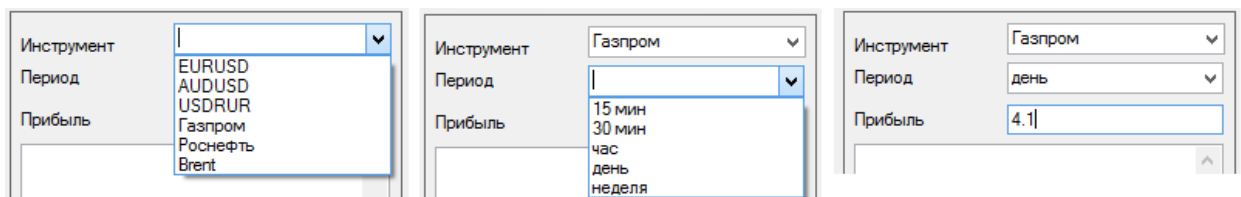


Рисунок 4.4 – Элементы ввода пользователем информации

Третий блок интерфейса – это блок вывода системой проанализированной информации (рисунок 4.5), в том числе рекомендации, как главной информационной составляющей работы системы в целом.

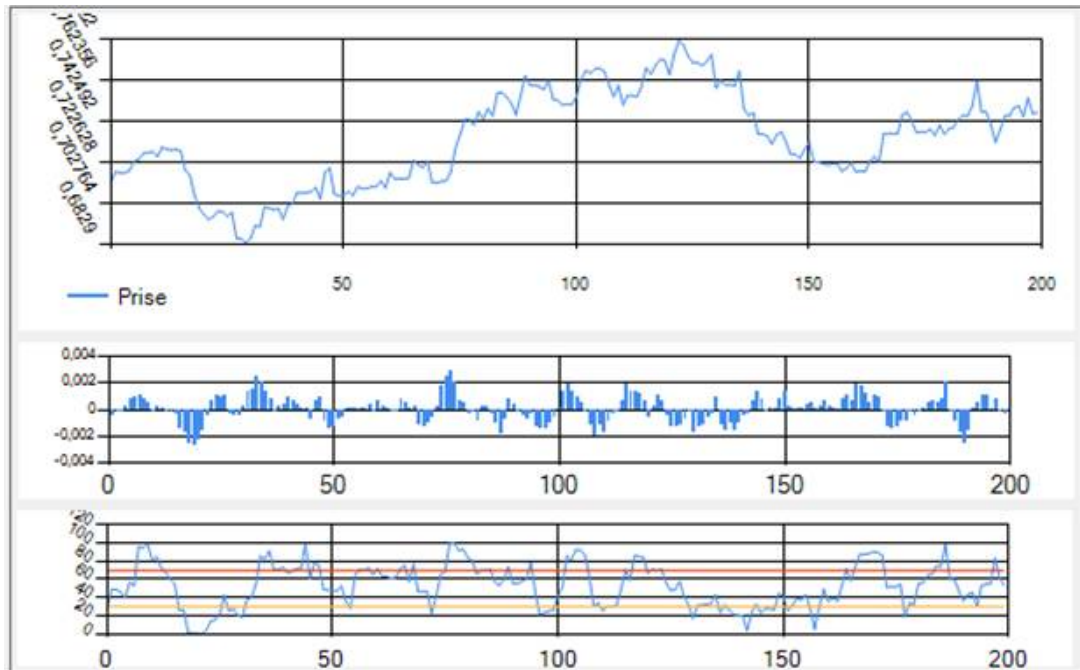


Рисунок 4.5 – Графики цен и индикаторов

На рисунке 4.5 видно, при отрисовке графиков используются синий цвет, который характеризуется постоянством, и снимает напряжение от ожидания ответа системы пользователя. А линия уровней одного из индикаторов выделены более яркими цветами, так как они дают ориентир для трейдера.

Также интерфейс содержит панель с логированием действий системы: загрузки данных, обучения и тестирования нейронной сети (рисунок 4.6).

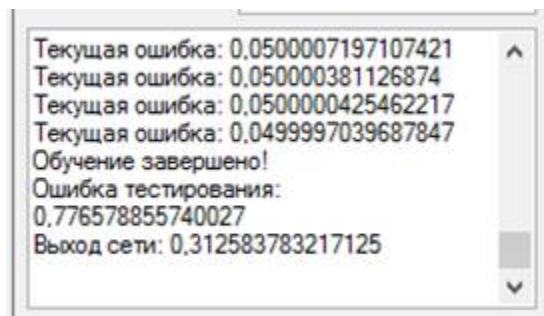


Рисунок 4.6 – Графики цен и индикаторов

Главным результатом работы ИСППР является выдача рекомендаций по покупке-продаже финансового инструмента (рисунок 4.7).

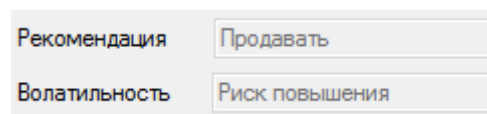
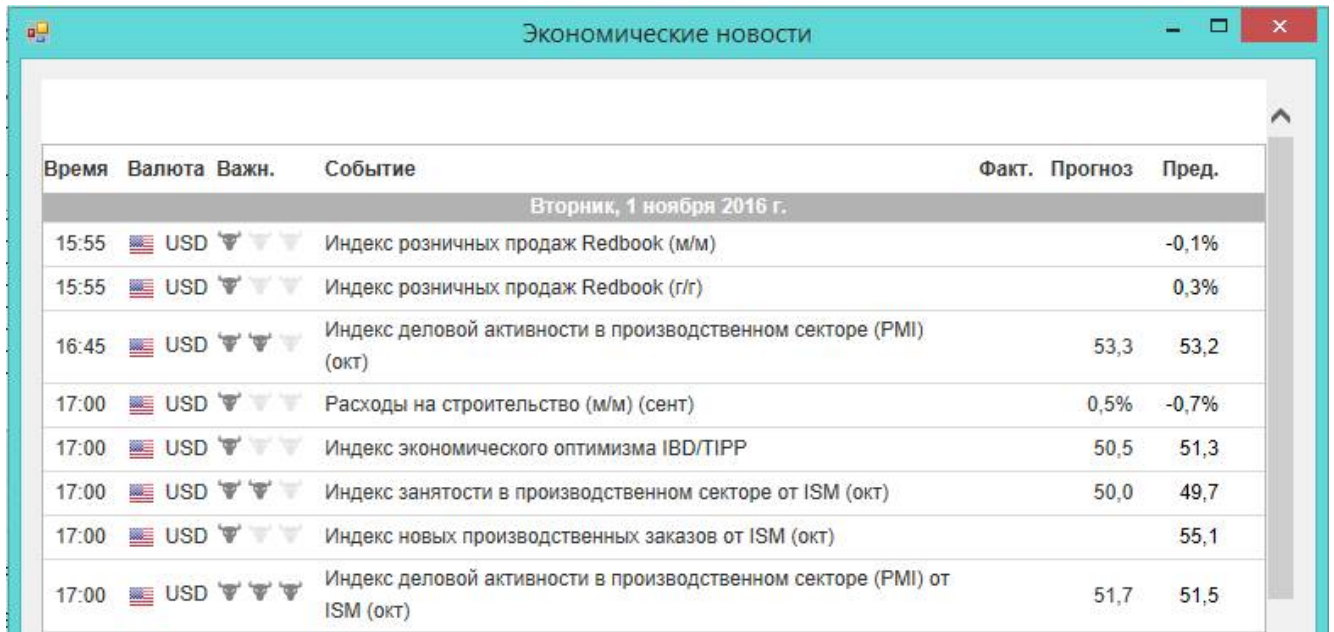


Рисунок 4.7 – Панель рекомендация

Во время обучения нейронной сети у пользователя есть возможность ознакомиться с экономическими событиями, рекомендованными системой как важными для выбранного финансового инструмента и периода (см. рисунок 4.8).



Время	Валюта	Важн.	Событие	Факт.	Прогноз	Пред.
Вторник, 1 ноября 2016 г.						
15:55	USD	▼▼▼	Индекс розничных продаж Redbook (м/м)			-0,1%
15:55	USD	▼▼▼	Индекс розничных продаж Redbook (г/г)			0,3%
16:45	USD	▼▼▼	Индекс деловой активности в производственном секторе (PMI) (окт)		53,3	53,2
17:00	USD	▼▼▼	Расходы на строительство (м/м) (сент)		0,5%	-0,7%
17:00	USD	▼▼▼	Индекс экономического оптимизма IBD/TIPP		50,5	51,3
17:00	USD	▼▼▼	Индекс занятости в производственном секторе от ISM (окт)		50,0	49,7
17:00	USD	▼▼▼	Индекс новых производственных заказов от ISM (окт)			55,1
17:00	USD	▼▼▼	Индекс деловой активности в производственном секторе (PMI) от ISM (окт)		51,7	51,5

Рисунок 4.8 – Окно с событиями экономического календаря

После обучения сети и выдачи рекомендации работу системы можно оценить (рисунок 4.9), данная оценка сохраниться в историю, которую можно в дальнейшем анализировать.

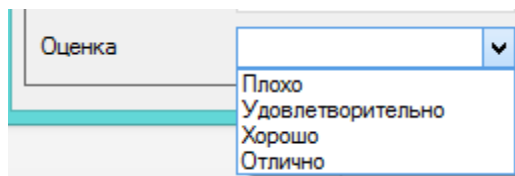


Рисунок 4.9 – Панель оценки работы системы

В разделе «Сервис» есть возможность анализа отчетов по работе системы.

В окне «Формирования отчета» (рисунок 4.10) предоставлена возможность производить анализ работы системы.



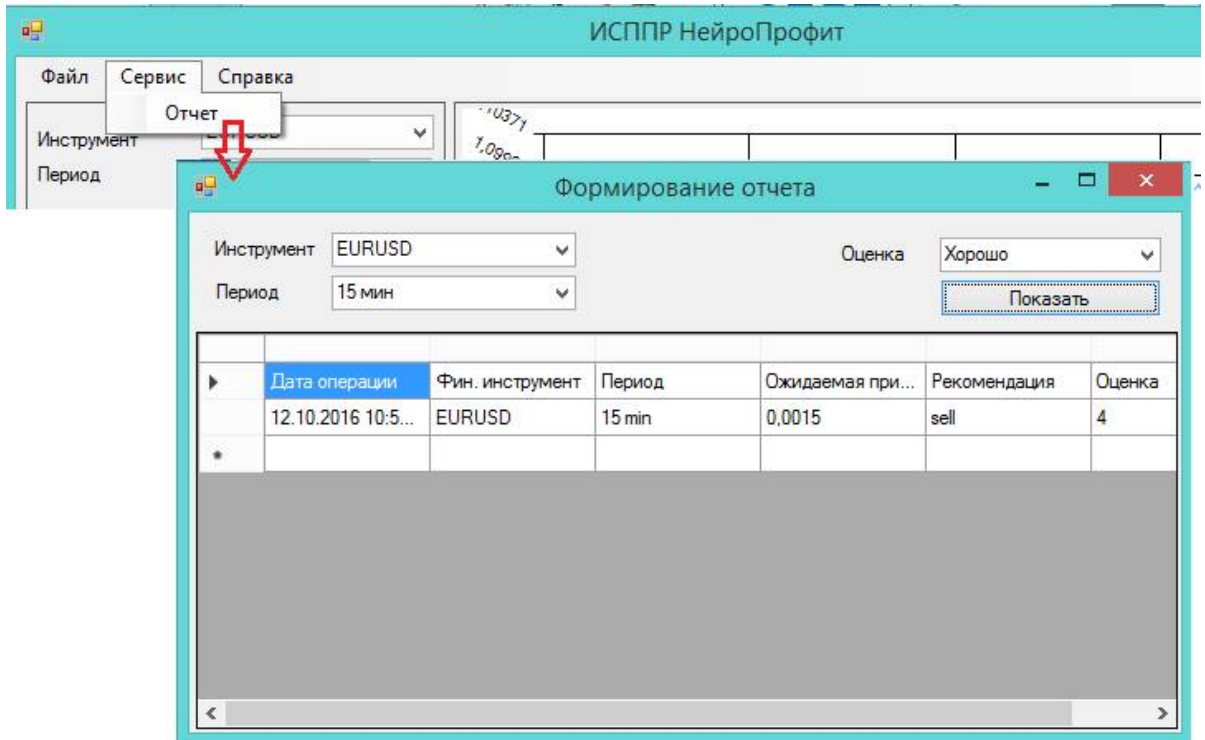


Рисунок 4.10 – Окно работы с отчетами

Прочие дочерние окна при работе с системой приведены на рисунке 4.11.

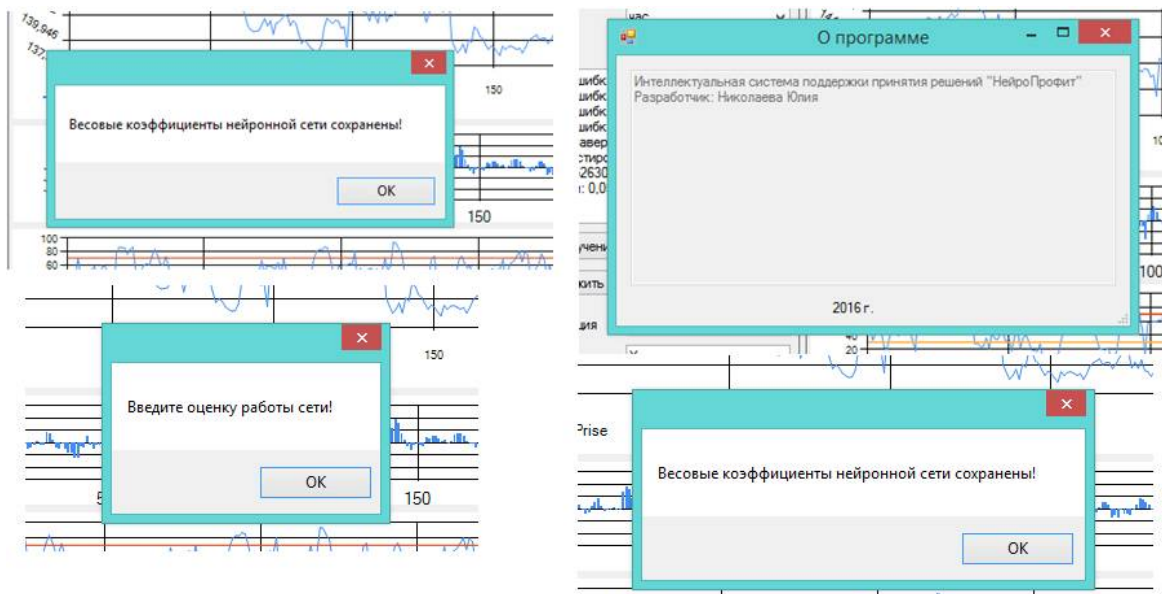


Рисунок 4.11 – Примеры дополнительных дочерних форм

Разработанный интерфейс интеллектуальной системы поддержки принятия решений на финансовых рынках является интуитивно понятным пользователю, основан на привычном ему по многим другим приложениям и по операционной системе Windows оконному графическому принципу. Данный интерфейс выполнен с учетом требований и пожеланий будущих пользователей и предусматривает их уровень образования и владения компьютером, а также психологическое влияние различных цветов на состояние человека. С таким интерфейсом легко и приятно



взаимодействовать, он не вызывает затруднений и лишних вопросов в процессе эксплуатации интеллектуальной системы.

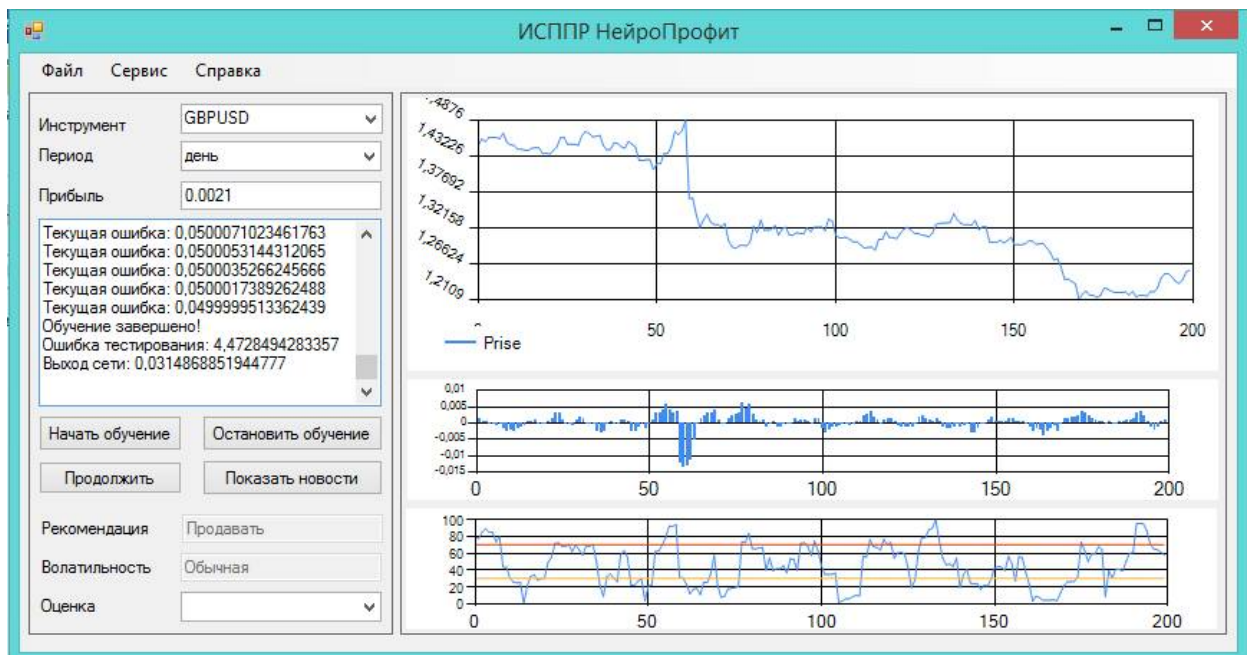
#### 4.2.2 Примеры функционирования системы

Интеллектуальная система поддержки принятия решений «НейроПрофит», основанная на разработанной методике классификаций рыночных ситуаций, предназначена для выдачи рекомендаций по совершению сделок на финансовых рынках. Она выдает одну из трех рекомендаций: покупка, продажа, ожидание, – в зависимости от выбранных начальных параметров: финансового инструмента, временного отрезка и ожидаемой прибыли, анализируя ситуацию на рынке в недавнем прошлом, изменения цен выбранного финансового инструмента.

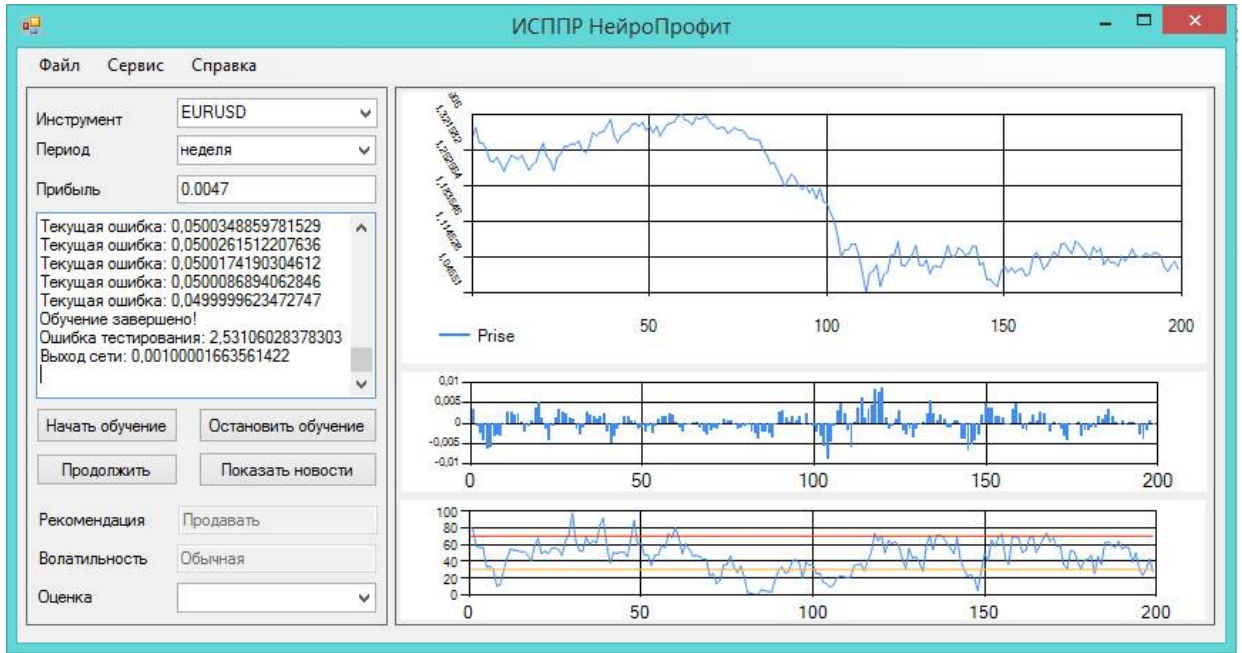
Некоторые примеры исходного кода вычислительных алгоритмов системы представлены в Приложении Г.

Ниже приведены примеры использования интеллектуальной системы поддержки принятия решений «НейроПрофит».

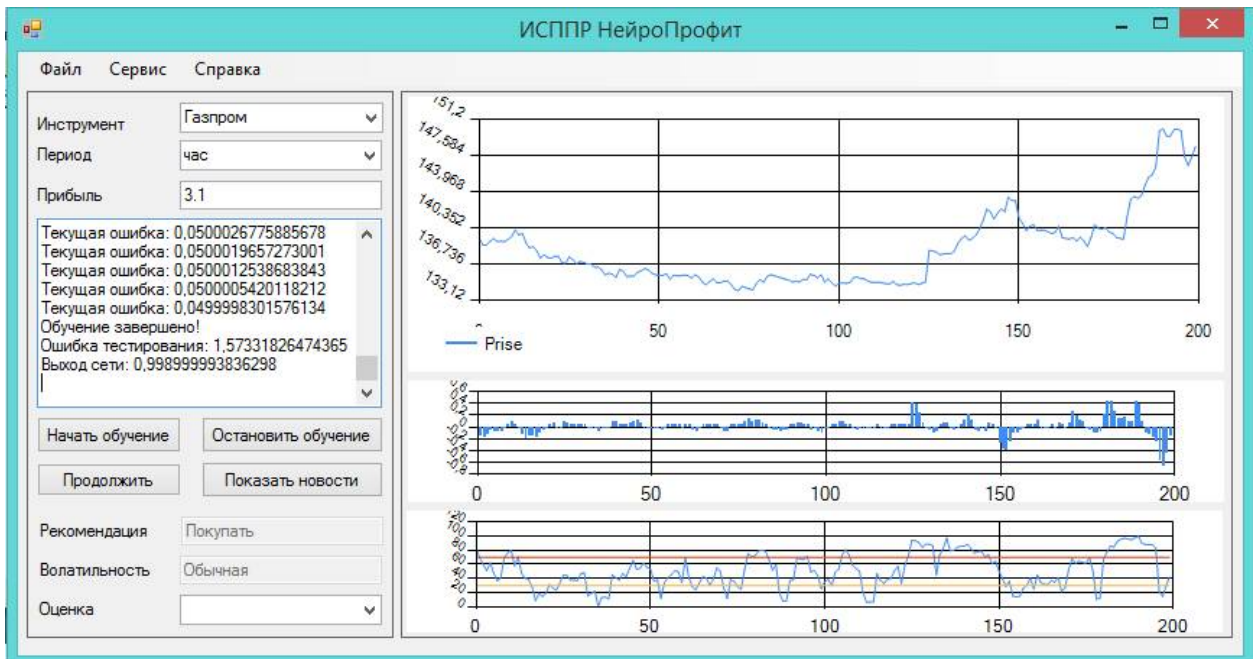
На рисунке 4.12 представлены пример разнообразных выданных рекомендаций по акциям ОАО «Роснефть», ОАО «Газпром», валютным парам USDRUR, GBPUSD, EURUSD, фьючерсу на нефть сорта Brent по графикам различной периодичности:



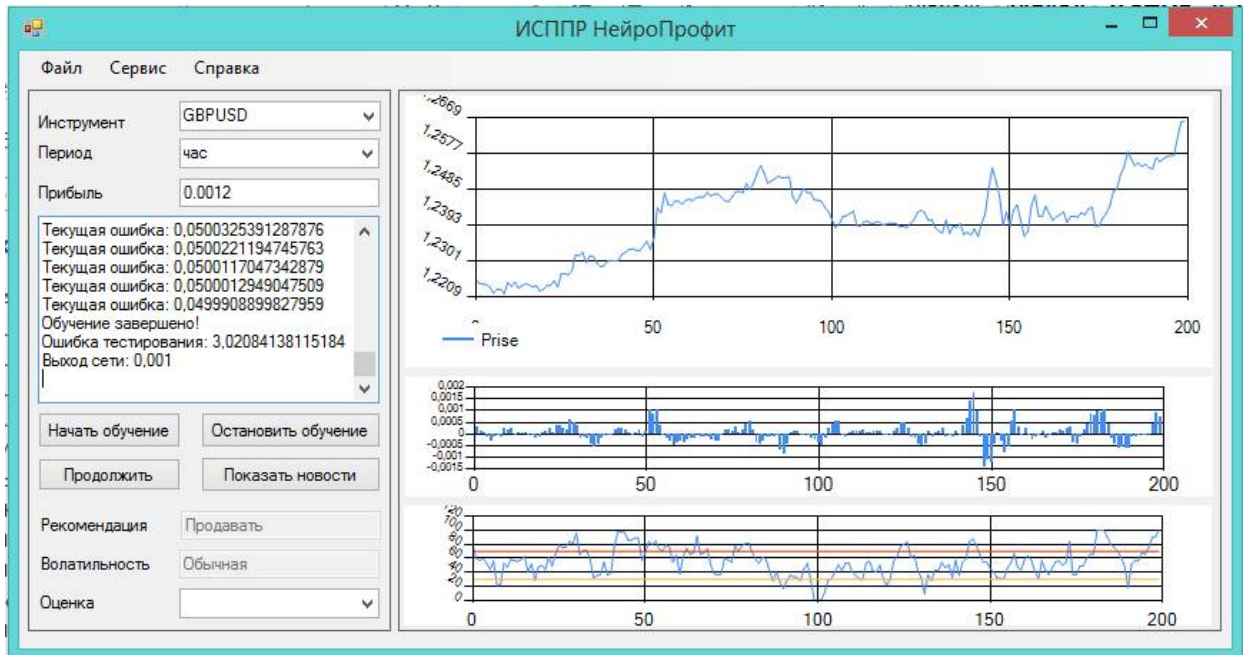
а)



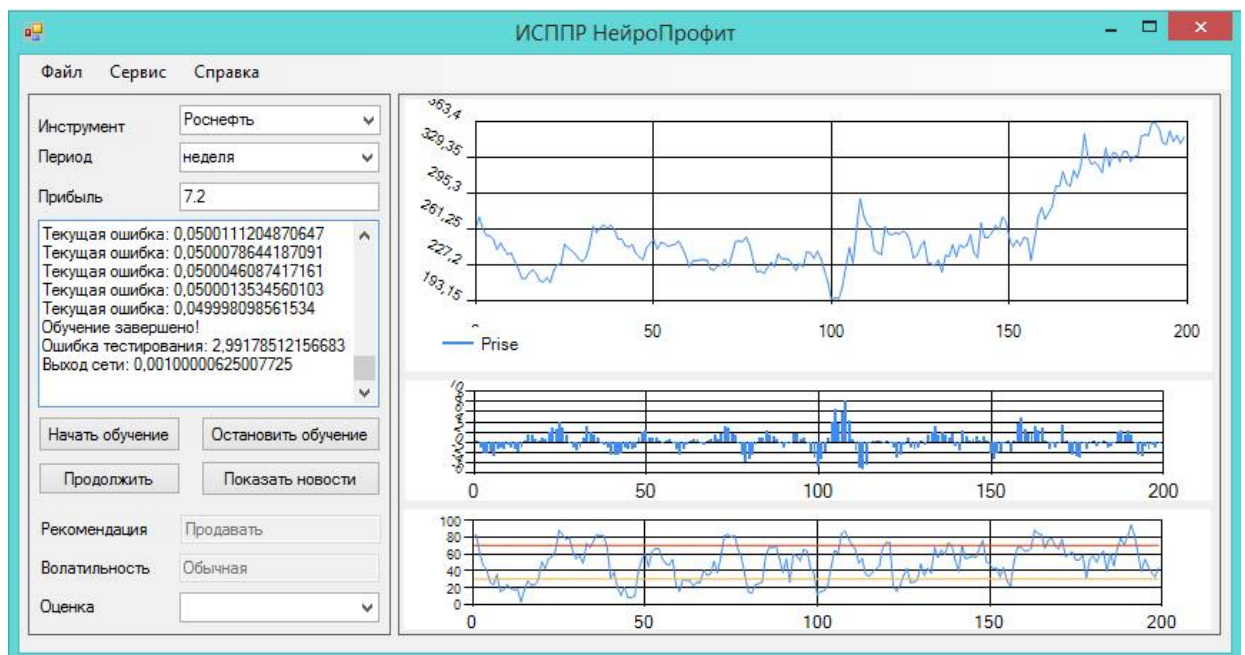
б)



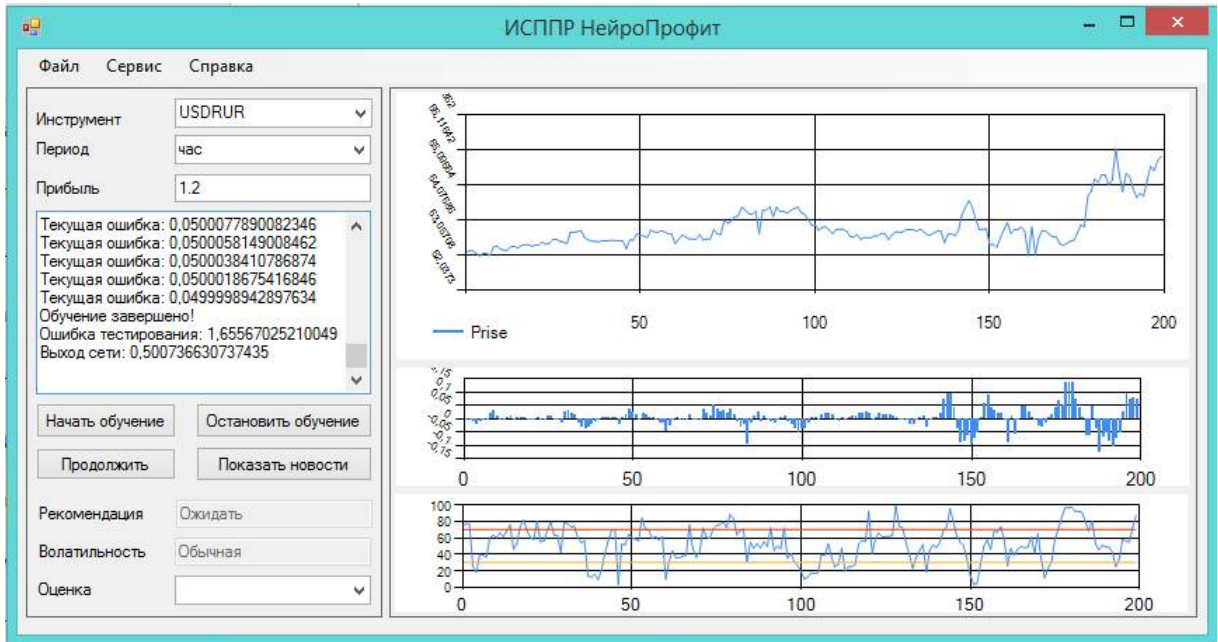
в)



г)



д)



е)

Рисунок 4.12 (а-е) – Примеры функционирования ИССПР «НейроПрофит»

### 4.3 Исследование работоспособности системы

Подобные системы разрабатывались в рамках многих научных исследований по данной тематике, например, [2, 1, 49, 91], но в большинстве случаев не были проведены эксперименты для анализа их эффективности. В некоторых же работах говорить об объективном исследовании эффективности подобных систем не представляется возможным, в связи с переменчивостью моделей системы, поскольку критерии архитектуры нейронной сети определяется пользователем программы либо выбираются из коллекции параметров автоматически или используются комитеты нейронных сетей [63, 7, 35]. В тех работах, где все же приведены результаты эффективности применения нейросетевой методологии для прогнозирования финансовых рынков, предоставляемая эффективность (соотношение количества верно выдаваемых сигналов по отношению к общему количеству) не многим выше выбора по случайному процессу [17 гл. 10, 88 гл. 2].

Так, например, в работе [22, стр. 61] показано, что при использовании нейронной сети топологии многослойный перцептрон для прогнозирования знака доходности в следующем периоде процент правильного прогноза не поднимается выше 58%.

Довольно близкий по концепции использования нейронных сетей в анализе финансовых рынков и используемым инструментам алгоритмом (аналогом) является метод «окон». Суть метода заключается в формировании прогнозной функции на основе многослойного перцептрона, на входы которого подаются  $N-1$  предыдущих значений временного ряда цен финансового инструмента, на выходе выдается прогноз цены на период  $N+M$ . Система, построенная на исполь-

зовании данного метода описана в работе [7], где многослойный перцептрон используется для прогнозирования временного ряда финансового инструмента. В работе видно, что максимальный процент верно спрогнозированных цен финансового инструмента, даже с учетом выбора победителя нейросети из коллекции обученных нейронных сетей, составляет 74,5%.

В процессе проверки работоспособности системы, были выявлены следующие сложности:

1. Продолжительность обучения по стандартному алгоритму обратного распространения ошибки была не удовлетворительной для реальной работы пользователей с системой.
2. При обучении возникали ситуации, когда ошибка обучения долгое время оставалась практически на одинаковом уровне, то есть изменение весов при такой ситуации было минимальным, что значительно замедляло обучением нейронной сети.

Для решения первой проблемы была проанализирована зависимость между шагом коррекции весов, продолжительностью обучения и ошибки тестирования (см. таблицу 4.2).

Под шагом коррекции весов ( $g$ ) понимается дополнительный коэффициент для ускорения нахождения траектории минимизации в алгоритме обратного распространения ошибки при итерационном изменении весов:

$$w_{ij}(t+1) = w_{ij}(t) - g \cdot h \cdot \frac{\partial E}{\partial w_{ij}} \quad (4.1).$$

Таблица 4.2 Результаты исследования зависимости шага коррекции весов и продолжительности обучения и ошибки тестирования

Шаг коррекции весов	Средняя ошибка	Средняя продолжительность обучения*
1	3,1	2,6 мин
1,6	3,0	1,8 мин
1,8	3,0	1,6 мин
4	3,1	1,3 мин
6	3,2	1,2 мин
10	3,2	1,0 мин
16	3,3	0,9 мин
40	4,1	2,1 мин
50	4,2	2,9 мин

\* – из результатов эксперимента убраны случаи продолжительного заикливания алгоритма вблизи локального минимума.

Из таблицы 4.2 видно, что по мере увеличения шага коррекции весов, уменьшается продолжительность обучения и ошибка тестирования, но до определенных пределов. Если шаг коррекции весов сильно увеличивать (более 30), то нейронная сеть снова характеризуется про-

должительным и ошибочным функционированием. Это связано с тем, что при поиске оптимума, двигаясь в направлении минимизации ошибки, алгоритм делает слишком большие шаги и пропускает действительные точки минимума.

Экспериментальным путем, было установлено, что при повышенном шаге коррекции весов сеть довольно быстро находит верное направление минимизации до ошибки обучения величиной 1,02. Далее для снижения ошибки сети необходимо более детальное рассмотрение обучающего пространства. Если при этом не снижать скорость обучения, то, как было сказано выше, алгоритм проскакивает точки минимума и результатом является не самый оптимальный подбор весов, что влияет на ошибку тестирования.

Таким образом, было принято решение усовершенствовать алгоритм обратного распространения ошибки для обучения нейронной сети путем применения гибкой настройки скорости обучения. Суть модификации в следующем: начинаем обучение с довольно большой скорости обучения, что позволяет быстро привести процесс обучения к приемлемо малой ошибке обучения. Далее по мере снижения ошибки обучения, в виду необходимости более досконального исследования алгоритмом поверхности обучающего множества, скорость обучения снижается, и при достижении ошибки обучения 1,02 шагу коррекции присваивается значение 1, что позволяет алгоритму двигаться по рассчитанным по алгоритму дельтам без потери направления и величины шага.

Вторая проблема, периодически возникающая в процессе обучения, связана с заикливанием алгоритма обучения при довольно малых изменениях ошибки обучения менее чем 0,00001. Если рассматривать математическую трактовку алгоритма обратного распространения с точки зрения теории минимизации, такие ситуации возникают при нахождении алгоритмом локальных минимумов поверхности обучающего множества и блуждании алгоритма в близи данного локального минимума. При попадании в точки локальных минимумов алгоритм двигается в его направлении, но не может снизить ошибку обучения до предельного уровня, так как она может быть достигнута в другом локальном или глобальном минимуме.

Для решения данной проблемы, решено внести модификации в алгоритм и добавить элементы стохастического обучения. При достижении большого количества (более 200) шагов алгоритма с изменениями ошибки обучения менее 0,00001, поочередно вносим правки в каждый вес первого слоя перед обучением на очередной паре входных и выходных данных.

$$w_{cr}(t) = w_{cr}(t) + 0,1 * h \quad (4.2),$$

где  $c, r$  – номера координат веса, сдвигаемого перед одной итерацией обучения.

Таким образом, мы не сбиваем алгоритм с изначального направления минимизации, но сдвигаем по одной координате на поверхности обучения, для нахождения выхода из области



локального минимума. Результаты ускорения процесса обучения при помощи модификации алгоритма с элементами стохастического обучения приведены в таблице 4.3.

Таблица 4.3 Сравнение обучения алгоритмом обратного распространения ошибок без и с применением стохастического сдвига по одному из весов

Коэффициент сдвига веса	Средняя продолжительность обучения*	Средняя ошибка тестирования*
-	4,6	3,7
0,1	3,1	3,5

\* – в случае попадания при обучении алгоритма в описанную ситуацию.

Таким образом после проведенных исследований по работоспособности системы, модифицирован стандартный алгоритм обучения нейронной сети – алгоритм обратного распространения ошибки – в него включена гибкая скорость обучения и элементы стохастического обучения, что привело к уменьшению продолжительности обучения и сделало возможным применять систему в режиме тестовой эксплуатации.

Тестовая эксплуатация системы заключалась в ее использовании в качестве системы поддержки принятия решений трейдеров на реальных исторических данных по котировкам финансовых инструментов, результаты которого стали объектом экспериментального исследования. Для оценки качества выдаваемых разработанной автором данного исследования системы поддержки принятия решений рекомендаций было проведен эксперимент на выборке из 90 различных примеров, подаваемых на вход функционирования системы. Параметры выборки являлись период и финансовый инструмент. Для исключения влияния параметров на результаты эксперимента, была произведена нормировка выборки по параметрам (см. в таблице 4.4).

Таблица 4.4 Параметры экспериментальной выборки

Период \ Фин. инстр.	15 мин	30 мин	1 ч	1 д	1 н
EURUSD	3	3	3	3	3
Газпром	3	3	3	3	3
GBPUSD	3	3	3	3	3
Роснефть	3	3	3	3	3
Brent	3	3	3	3	3
USDRUR	3	3	3	3	3

Оценивалась точность и достоверность классификации рыночных ситуаций. Выданная системой рекомендация сравнивалась с реальной ситуацией на финансовом рынке в конкретный момент. Реальная ситуация на рынке оценивалась по аналогичному обучающему правилу алго-

ритму: выявлялась максимальные повышение и понижение цены от цены закрытия в ценах за следующие 10 периодов.

Ситуация продажи считалась, если абсолютное значение максимального понижения больше максимального повышения. Ситуацией покупки, если максимальное повышение, наоборот, превышало максимальное понижение (по модулю). Ситуацией ожидания признавались случаи совпадения абсолютных значений максимальных повышения и понижения.

В выборку вошли примеры котировок цен все перечисленных финансовых инструментов за все возможные при работе в системе периоды методом случайного подбора в промежутке с 01.03.2008 г. по 4.11.2016 г.

С полученными результатами по экспериментальной выборке можно ознакомиться в Приложении Д.

По результатам эксперимента количество верно распознанных рыночных ситуаций равняется 75, что составляет 83% от общего количества проведенных экспериментов.

Общие результаты экспериментального исследования представлены в таблице 4.5:

Таблица 4.5 Количество выданных рекомендации

Рекомендация	Количество
Покупать	40
Продавать	47
Ожидать	3

Количество различных комбинаций рекомендации системы и мнения эксперта, полученные в ходе эксперимента представлены в таблице 4.6:

Таблица 4.6 Комбинации рекомендаций и оценок эксперта

Рекомендация \ Реальная Ситуация	Покупать	Продавать	Ожидать
Покупать	36	8	1
Продавать	4	39	2
Ожидать	0	0	0

Важными для реальной стратегии торговли профессионального трейдера являются правильными выданные рекомендации по покупке или продаже финансового инструмента, количество таковых по результатам эксперимента составляет 75 рекомендации. Получается, в 83% случаев система правильно выдает рекомендацию по активному действию.



Процент верных рекомендаций по покупке составляет 90% от количества рекомендаций в данном направлении, 48% от общего числа верных рекомендации, по продаже – 83% от количества рекомендаций в данном направлении, 52% от общего числа верных рекомендации. Также ошибочными являлись 3 рекомендации по ожидательной ситуации на рынке, что в целом не привело бы к убыткам трейдера, если не рассматривать упущенную выгоду (см. рисунок 4.13).

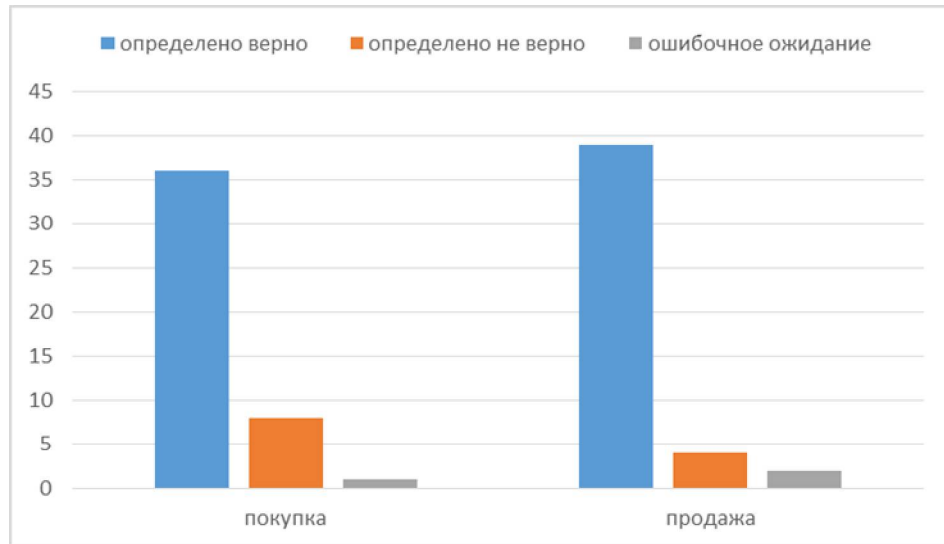


Рисунок 4.13 – Распределение распознанных рыночных ситуаций

Убыточными для пользователя системы являются рекомендации системы, противоположены реальному положению на рынке, то есть совет покупать, когда цена финансового инструмента падает, или продавать, когда рынок на подъеме. Ошибочных рекомендаций по покупке из общего количества рекомендаций по покупке составляет 4,4%, а по отношению к верным рекомендациям данного направления 10%. Ошибочных рекомендаций по продаже финансового инструмента из общего количества рекомендаций по продаже составляет 8,8%, а по отношению к верным рекомендациям данного направления 17 %.

Общее количество рекомендаций, гипотетически способных привести пользователя к неверным действиям на рынке и соответственно потерпеть материальные потери равно 12 из 90, то есть 13% рекомендаций интеллектуальной СППР «НейроПрофит» способны привести к убытку.

Стоит отметить, что при подробном анализе рыночных ситуаций, при которых системой были выданы ошибочные рекомендации обнаружилось 4 случая, при которых даже по неверно выданной рекомендации пользователь получил бы от 90 до 112% ожидаемой прибыли. Эти случаи связаны с примерно одинаковым максимальным повышением и максимальным понижением цен финансового инструмента за рассматриваемые 10 следующих от анализируемого периода. Также анализ ошибочно выданных рекомендаций показал, что их возникновение может быть связано с слишком завышенными или заниженными ожиданиями пользователя по поводу

изменения цен. Так было выявлено 4 случая, в которых ожидаемая прибыль была указана выше действительно достигнутой, для них процент достижения прибыли составлял от 27 до 91%.

Среди неверно выданных рекомендаций было выдано 2 при слишком заниженных ожиданиях пользователя в прибыли, для них процент достижения прибыли составил 225% и 448%. Но поскольку количество верно выданных рекомендаций, при которых процент достижения прибыли составлял бы более 200% насчитывает 11 случаев, то следует скорее говорить, что заниженные ожидания пользователя в целом положительно сказываются на обучении нейронной сети.

Так как в результате тестирования было выявлено, что задаваемая пользователем ожидаемая прибыль сказывается на качестве функционирования системы, решено в систему добавить инструмент-советник, анализирующий средние, максимальные и минимальные изменения значений за 10 периодов в просматриваемой выборке цен финансового инструмента. И чтобы исключить ситуации не верного функционирования системы из-за неквалифицированности пользователя, данный инструмент предупреждает пользователя о значительных (более чем в 4 раза) отклонениях указанной ожидаемой прибыли от средней в исследуемой выборке. Данный инструмент помогает избежать ситуаций задания заранее не верной ожидаемой прибыли, на основе которой будет произведено некорректное обучение нейронной сети по разработанному автором работы обучающему правилу, зависящему от ожидаемой прибыли.

По результатам принятого решения на основе рекомендаций, выданных системой поддержки принятия решений, трейдер может оценить совершенное действие, то есть заключенную сделку, насколько результат (полученная или нет прибыль) соответствует его ожиданиям. Что в дальнейшем учитывается в системе как критерий выбора наиболее подходящей изначальной структуры нейронной сети для начала обучения на новых данных в новый момент времени при последующем использовании системы. В системе доступна возможность оценки и сохранения параметров нейронной сети, которые могут использоваться при последующей работе системы, что положительным образом сказывается на повышении эффективности работы системы и уменьшает время обучения сети (см. таблицу 4.7).

Таблица 4.7 Влияние сохранения весов нейронной сети на продолжительность обучения

Параметры нейронной сети	Оценка пользователя для сохраненной структуры	Средняя продолжительность обучения, мин.
Первоначальное обучение	-	1,2
Повторное обучение на сохраненных параметрах	Удовлетворительно	1,6
	Хорошо	1,1
	Отлично	0,9

Продолжение таблицы 4.7

Третье обучение на со- храненных параметрах	Удовлетворительно	1,6
	Хорошо	0,75
	Отлично	0,6

Таким образом в результате проведенных экспериментальных исследований, можно сделать вывод, что система выдает примерно одинаковое количество верных рекомендаций по покупке и продаже финансовых инструментов с небольшим перевесом в 6% в сторону верных рекомендаций по продаже.

При тестировании системы средняя продолжительность обучения нейронной сети составила 0,63 мин, что значительно превышает показатели работы системы, до усовершенствования алгоритма обучения. Средняя ошибка тестирования также снизилась и по результатам тестирования системы на экспериментальной выборке составила 2,54.

Средний процент превышения указанной пользователем ожидаемой прибыли при в случае правильной рекомендации составил 37,7%, то есть при верно выданной системой рекомендации пользователь получает ожидаемую прибыль и может рассчитывать на ее увеличение.

Общая эффективность использования интеллектуальной системы «НейроПрофит» составляет 83%, что позволяет применять ее для реальной торговли на финансовых рынках.

#### Выводы по главе

1. В данной главе рассмотрены вопросы функционирования и тестирования системы поддержки принятия решений трейдеров финансовых рынков.
2. Показано сравнение с аналогичными системами поддержки принятия решений на финансовых рынках.
3. Рассмотрена программная реализация системы на основе разработанной методики анализа финансовых рынков. Приведены примеры функционирования системы на реальных примерах.
4. Разработанная система позволяет трейдеру получать рекомендации по направлению заключения сделки (покупка или продажа) в рыночной ситуации, соответствующей выбранному финансовому инструменту, периоду, и учитывая ожидаемую прибыль от сделки.
5. В заключении главы приведены результаты тестирования системы: проверки ее работоспособности и эффективности.

При проверке работоспособности разрешены 2 проблемы, влияющие на замедление процесса обучения нейронной сети. В ходе решения данных проблем предложена и реализована модификация алгоритма обратного распространения ошибок. В алгоритм добавлены гибкая

настройка скорости обучения: скорость обучения уменьшается по мере уменьшения ошибки обучения и приближения алгоритма к искомому минимуму. Для исключения ситуаций продолжительного блуждания в области локальных минимумов, в алгоритм добавлены элементы стохастического обучения: последовательные сдвиги по каждой из координат поверхности обучения при застревании процесса обучения.

После модификации алгоритма обучения средняя продолжительность обучения оставила 0,63 мин, что значительно превышает показатели работы системы, до усовершенствования алгоритма обучения, при уменьшении также и ошибки тестирования на экспериментальной выборке до 2,54.

6. Выявлены практические рекомендации по использованию системы. Важным при использовании системы является указание ожидаемой прибыли релевантной рассматриваемой выборке цен финансового инструмента. Выбор ожидаемой прибыли должен осуществляться в пределах средних значений изменений цен за 10 периодов, для анализа которых в систему добавлен инструмент-советник. Также установлено, что немного заниженные ожидания трейдера по поводу ожидаемой прибыли, положительно влияют на результаты обучения нейронной сети.

По результатам экспериментального исследования результатов тестирования системы процент верно выданных рекомендаций по продаже на 6% превышает процент верно выданных рекомендаций по покупке. Общая эффективность системы, представляющая собой соотношение количества верно выданных рекомендаций к ошибочным, составляет 83%.

Надежность и достоверность полученных результатов подтверждаются тестированием системы на реальных исторических данных по котировкам финансовых инструментов и их соответствием результатам ручной проверки по проценту достижения прибыли в рекомендуемом системой направлении.

## ЗАКЛЮЧЕНИЕ

Успешность деятельности на финансовом рынке зависит от умения правильно прогнозировать изменение цен, от своевременности принятия решений и от личности самого трейдера. Зачастую причиной разорения является именно психологический фактор. Разработанная интеллектуальная система поддержки принятия решений на финансовых рынках, основанная на методике классификации рыночных ситуаций, позволяет четко оценивать ситуацию на рынке конкретного финансового инструмента, и исключает негативное влияние психологического фактора на успешность торговой деятельности трейдера.

Разработанная интеллектуальная система поддержки принятия решений на финансовых рынках будет хорошим подспорьем в реальной работе по покупке и продаже акций, фьючерсных контрактов, валют спекулянтами, биржевыми трейдерами, крупными инвестиционными компаниями, заинтересованными в исследуемом эмитенте.

Результатом проведенных исследований является разработанная интегрированная методика классификации рыночных ситуаций с применением искусственной нейронной сети, методов технического, фундаментального, эконометрического анализа, и реализованная на его основе интеллектуальная система поддержки принятия решений. Проведенные в рамках диссертационного исследования работы позволили ответить на поставленные вопросы, соответствуют цели и задачам исследования. Экспериментальные результаты исследования показывают высокую эффективность предложенных и разработанных в диссертации методики, методов, модификаций, алгоритмов и программного обеспечения для решения задачи классификации рыночных ситуаций. Выполненные научные изыскания позволили получить следующие результаты и выводы:

1. Финансовые рынки являются примером социально-экономических систем, к которым может быть применена теория управления организационными системами. Анализ рынков требует многостороннего подхода, в котором должны быть задействованы как инструменты технического, фундаментального анализа, экономико-математического подхода, так и инструменты современного интеллектуального анализа данных. Разработанные методика и алгоритм анализа финансовых рынков, отвечает данным требованиям.
2. Одним из эффективных методов, на котором строятся современные интеллектуальные системы поддержки принятия решений являются нейронные сети. Для целей классификации рыночных ситуаций подходящей является топология нейронной сети, такая как многослойный перцептрон. При разработке архитектуры нейронной сети необходимо учитывать взаимную

обусловленность функций распределения отбираемых критериев, и минимизировать количество нейронов входного слоя с учетом особенностей теории нейронных сетей.

3. Разработан метод генерации выходной выборки для задачи нейросетевой классификации, учитывающий ожидаемую прибыль и повышение волатильности. Метод позволяет исключить влияние на эффективность обучения нейросети квалификации эксперта и учесть риск повышения волатильности, связанный с влиянием макроэкономических показателей и новостей.

4. В результате проделанных исследований разработана комбинированная методика поддержки принятия решений трейдера по направлению сделки с финансовым инструментом и алгоритм ее реализации в системах поддержки принятия решений. Методика заключается в нейросетевой классификации ситуаций на финансовых рынках, в основу которой входит использование модели GARCH, индикаторов технического анализа RSI, MACD, интеграция полученных данных и обучение нейросети с учетом повышения волатильности за счет реакции на экономические события. Сформированная методика и ее алгоритм позволяют проводить анализ многокритериальной информации о текущем состоянии рынка выбранного финансового инструмента, и повышает точность классификации ситуаций на его рынке на 8,5% по сравнению с близким аналогичным алгоритмом нейросетевого анализа финансовых временных рядов.

5. Разработанный алгоритм нейросетевой классификации рыночных ситуаций стал основой проектирования и реализации интеллектуальной системы поддержки принятия решений, которая может использоваться при реальной торговле на финансовых рынках профессиональными трейдерами. Разработанный программный продукт «НейроПрофит» обеспечивает трейдера рекомендацией по покупке-продаже финансовых инструментов с достоверностью в 83%. При использовании ИСППР «НейроПрофит» трейдер может получать до 137% ожидаемой от сделки, открытой по рекомендации системы, прибыли.

Таким образом, работа полностью отвечает поставленным целям и задачам. Разработанная по результатам исследования система доказала достоверность выдаваемых рекомендаций (83%), и может применяться в реальной торговле на финансовых рынках, что сокращает субъективность принятия решений по совершению торговых операций и ведет к достижению трейдером прибыли от совершения торговых операций.

**СПИСОК СОКРАЩЕНИЙ**

ARCH (AutoRegressive Conditional Heteroscedasticity) – модель авторегрессионной условной гетероскедестичности.

Business Intelligence (BI) – методы, инструменты и программное обеспечение для бизнес-аналитики.

Data Mining (ИАД) – методы и средства извлечения знаний и закономерностей из «сырых» данных, средства интеллектуального анализа данных.

GARCH (Generalized AutoRegressive Conditional Heteroscedasticity) – обобщенная модель авторегрессионной условной гетероскедестичности.

AP( $k$ ) – модель авторегрессии  $k$ -го порядка.

APCC( $k, m$ ) – модель авторегрессии-скользящего среднего.

АРПСС ( $k, d, m$ ) (ARIMA) - проинтегрированная модель авторегрессии-скользящего среднего.

ИНС – искусственные нейронные сети.

ИСППР – интеллектуальная система поддержки принятия решений.

ММП (MLE) – метод максимального правдоподобия (Maximum Likelihood Estimation).

ПО – программное обеспечение.

СППР (Decision Support System, DSS) – система поддержки принятия решений.

СС( $m$ ) – модель скользящего среднего  $m$ -го порядка.

СУБД – система управления базами данных.

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. Авдеев А.С. Разработка адаптивных моделей и программного комплекса прогнозирования экономических временных рядов [Текст]/ Авдеев А.С. // дис. канд. техн. наук. – Барнаул, 2010. – 196 с.
2. Авдеев А.С. Разработка программного комплекса нейропрогнозирования [Текст]/ А.С. Авдеев, О.И. Пятковский // Программные продукты и системы. – Тверь, 2010. – №1. С. 106-109.
3. Автоматизированные нейронные сети [Электронный ресурс]: информационная статья/ Statsoft.ru. – Режим доступа: [http://www.statsoft.ru/products/STATISTICA\\_Neural\\_Networks/](http://www.statsoft.ru/products/STATISTICA_Neural_Networks/) (Дата обращения: 04.11.2016).
4. Адаптивная резонансная теория [Электронный ресурс]: информационная статья / wikipedia.org. – Режим доступа: [http://ru.wikipedia.org/wiki/Адаптивная\\_резонансная\\_теория](http://ru.wikipedia.org/wiki/Адаптивная_резонансная_теория) (Дата обращения: 27.12.2017).
5. Анненков А.П. Разработка модели эволюции валютных котировок [Текст]/ Анненков А.П.// дис. канд. экон. наук. – Москва, 2011. – 182 с.
6. Аносо А. Критерии выбора СУБД при создании информационных систем [Электронный ресурс]: информационная статья/ Аносо А.. — Режим доступа: <http://www.interface.ru/home.asp?artId=2147> (Дата обращения: 04.11.2016).
7. Арсланов А. Прогнозирование финансовых временных рядов [Электронный ресурс]: информационная статья/ Азат Арсланов. – Режим доступа: <https://geektimes.ru/post/144405/> (Дата обращения: 25.09.2017).
8. Афанасьев В.Н. Анализ временных рядов и прогнозирование [Текст]/ В.Н. Афанасьев. – М.: Финансы и статистика, Инфра-М, 2012. – 320 с.
9. Байесовских подход и его применение в задачах страхования, гарантийного обслуживания и принятия решений с проведением экспериментов [Электронный ресурс]: информационная статья/ levvu.narod.ru. – Режим доступа: <http://www.levvu.narod.ru/Papers/Bayes.pdf> (Дата обращения: 12.03.2015).
10. Богатырев В.Д., Герасимов Б.Н. Основы теории управления экономическими системами: учеб. пособие [Текст]/ В.Д. Богатырев, Б.Н. Герасимов. – Самара: Изд-во Самар. гос. аэрокосм. ун-та, 2008. – 260 с.
11. Бокс, Дж. Анализ временных рядов прогноз и управление [Текст]/ Дж. Бокс, Г. Дженкинс // пер. с англ. под ред. В.Ф. Писаренко. – М.: Мир, 1974, кн. 1. – 406 с.



12. Борздов, С. Ю. Развитие методов оценки и прогнозирования волатильности курсов акций на фондовых рынках [Текст]/ С.Ю. Борздов// дис. канд. эк. наук. – Ставрополь, 2010. – 168 с.
13. Боровиков В.П. Прогнозирование в системе Statistica в среде Windows. Основы теории и интенсивная практика на компьютере [Текст]/ В.П. Боровиков, Г.И. Ивченко – М.: Финансы и Статистика, 2000. – 380 с.
14. Бугорский В.Н., Сергиенко А.Г. Использование нейронных сетей в работе трейдера [Текст]/ В.Н. Бугорский, А.Г. Сергиенко // Прикладная информатика №1. 2011. №31. С.17-27.
15. Бурков В.Н., Новиков Д.А. Теория адаптивных систем: состояние и перспективы [Текст]/ В.Н. Бурков, Д.А. Новиков – М.: Синтег, 1999. – 128 с.
16. Буч Г. UML. Классика CS [Текст]/ Г. Буч, А. Якобсон, Дж. Рамбо// 2-е изд. – Спб.: Питер, 2006. –736 с.
17. Бэстенс Д.-Э., ван ден Берг В.-М. Нейронные сети и финансовые рынки: принятие решений в торговых операциях [Текст]/ Д.-Э. Бэстенс, В.-М. ван ден Берг, Д. Вуд – Москва: ТВП, 1997. – 236 с.
18. Васильев В.А. Математические модели оценки и управления финансовыми рисками хозяйствующих субъектов [Текст]/ Васильев В.А., Летчиков А.В., Лялин В.Е. // Аудит и финансовый анализ, 2006. №4. С. 200-237.
19. Верников Г. Основы IDEF3 [Электронный ресурс]: информационная статья/ Верников Г. – Режим доступа: <http://www.cfin.ru/vernikov/idef/idef3.shtml> (Дата обращения: 04.11.2016).
20. Виды диаграмм UML [Электронный ресурс]: электронные лекции/ ИНТУИТ // Национальный открытый университет. – Режим доступа: <http://www.intuit.ru/department/se/intuml/2/2.html> (Дата обращения: 04.11.2016).
21. Галушкин А.И. Нейронные сети, как последовательностные машины [Текст]/ А.И. Галушкин, Ю.И. Фомин. – М.: МАИ. 1991. – 157 с.
22. Головачев С.С. Прогнозирование доходности на фондовом и валютном рынках на основе моделей искусственных нейронных сетей [Текст]/ Головачев С.С. // дис. канд. экон. наук. – Москва, 2014. – 181 с.
23. Горбань А.Н. Нейроинформатика [Текст]/ А.Н. Горбань, В.Л. Дунин-Барковский, А.Н.Кирдин и др. – Новосибирск: Наука. Сибирское предприятие РАН, 1998. – 296 с.

24. Горбань А.Н. Нейронные сети на персональном компьютере [Текст]/ А.Н. Горбань, Д.А. Россиев. – Новосибирск: Наука, 1996. – 276 с.
25. Горбань А.Н. Обучение нейронных сетей [Текст]/ А.Н. Горбань. – М.: СССР-США СП "ПараГраф", 1990. – 160 с.
26. Грошев А.С. Основы работы с базами данных/ Грошев А.С. [Электронный ресурс]: электронные лекции/ ИНТУИТ // Национальный открытый университет. – Режим доступа: <http://www.intuit.ru/department/database/basedbw/2/3.html> (Дата обращения: 04.11.2016).
27. Деревья принятия решений [Электронный ресурс]: информационная статья/ wikipedia.org. – Режим доступа: [http:// wikipedia.org/wiki/Дерево\\_принятия\\_решений](http://wikipedia.org/wiki/Дерево_принятия_решений) (Дата обращения: 04.11.2016).
28. Деревья решений – общие принципы работы [Электронный ресурс]: информационная статья/ BaseGroup Labs. – Режим доступа: <http://www.basegroup.ru/library/analysis/tree/description/> (Дата обращения: 21.05.2011).
29. Диаграммы взаимодействия: крупным планом [Электронный ресурс]: электронные лекции/ ИНТУИТ // Национальный открытый университет. – Режим доступа: <http://www.intuit.ru/department/se/intuml/5/> (Дата обращения: 04.11.2016).
30. Диаграмма деятельности (activity diagram) IDEF3 [Электронный ресурс]: информационная статья/ info-system.ru. – Режим доступа: [http://www.info-system.ru/designing/methodology/uml/theory/activity\\_diagramm t heory. html](http://www.info-system.ru/designing/methodology/uml/theory/activity_diagramm_t heory. html) (Дата обращения: 04.11.2016).
31. Диго С.М. Базы данных. Проектирование и создание: Учебно-методический комплекс [Текст]/ С.М. Диго. – М.: Изд. центр ЕАОИ, 2008. – 171 с.
32. Дунин-Барковский В.Л. Нейронные сети и нейрокомпьютеры: тенденции развития исследований и разработок [Текст]/ В.Л. Дунин-Барковский, А.Т. Терехин // Микропроцессорные средства и системы. 1990. №2. С. 12-14.
33. Ежов А.А., Шумский С.А. Нейрокомпьютинг и его применения в экономике и бизнесе [Текст]/ А.А. Ежов, С.А. Шумский // серия "Учебники экономико-аналитического института МИФИ" под ред. проф. В.В. Харитонов. – МИФИ, 1998. – 224 с.
34. Елисеева И.И. Эконометрика [Текст]/ И.И. Елисеева – М.: Финансы и статистика, 2003. – 344 с.
35. Ефремова Е.А. Применение нейронных сетей для прогнозирования финансовых временных рядов [Текст]/ Ефремова Е.А., Е.В. Дунаев// Доклады ТУСУРа, Автоматизированные си-

стемы обработки информации, управления и проектирования. 2004 г. С. 192 – 196.

36. Задачи Data Mining. Классификация и кластеризация [Электронный ресурс]: информационная статья. – Режим доступа: <http://www.intuit.ru/department/database/datamining/5/> (Дата обращения: 04.11.2016).

37. Интерфейс [Электронный ресурс]: информационная статья/ wikipedia.org. – Режим доступа: <http://ru.wikipedia.org/wiki/Интерфейс> (Дата обращения: 04.11.2016).

38. Искусственный интеллект [Электронный ресурс]: сайт/ www.AIportal.ru// Портал искусственного интеллекта – Режим доступа: <http://www.aiportal.ru/> (Дата обращения: 04.11.2016).

39. История развития методов интеллектуального анализа данных [Электронный ресурс]: информационная статья/ azfor.ucoz.ru. – Режим доступа: <http://azfor.ucoz.ru/publ/3-1-0-3> (Дата обращения: 04.11.2016).

40. Канторович Г.Г. Анализ временных рядов: Лекционные и методические материалы. Лекции № 1-4 [Текст]/ Г.Г. Канторович // Экономический журнал ВШЭ, 2002. – №1. С. 85-115.

41. Канторович Г.Г. Анализ временных рядов: Лекционные и методические материалы. Лекции № 5-7 [Текст]/ Г.Г. Канторович // Экономический журнал ВШЭ, 2002. – №2. С. 251-273.

42. Канторович Г.Г. Анализ временных рядов: Лекционные и методические материалы. Лекции № 8-9 [Текст]/ Г.Г. Канторович // Экономический журнал ВШЭ, 2002. – №3. С. 379-401.

43. Канторович Г.Г. Анализ временных рядов: Лекционные и методические материалы. Лекции № 10-13 [Текст]/ Г.Г. Канторович // Экономический журнал ВШЭ, 2002. – №4. С. 498-523.

44. Канторович Г.Г. Анализ временных рядов: Лекционные и методические материалы. Лекции № 14-16 [Текст]/ Г.Г. Канторович // Экономический журнал ВШЭ, 2003. – №1. С. 79-103.

45. Кияница А. С. Фундаментальный анализ финансовых рынков [Текст] / А. С. Кияница. – Спб: Питер, 2005. – 288 с.

46. Коломиец С. Психология и эргономика интерфейса [Электронный ресурс]: электронная статья/ Сергей Коломиец. – Режим доступа: <http://www.xserver.ru/computer/langprogr/razn/71/> (Дата обращения: 04.11.2016).

47. Корнеев С.В. Системы поддержки принятия решений в бизнесе [Текст]/ Корнеев С.В. – Сети и бизнес, 2005. – №6. С. 57-75.

48. Кратович П.В. Нейронные сети и модели ARIMA для прогнозирования котировок [Текст]/ П.В. Каратович// Программные продукты и системы, 2011, № 1. С. 95-98.

49. Кратович П.В. Нейросетевые модели для управления инвестициями в финансовые инструменты фондового рынка [Текст]/ П.В. Кратович// дис. канд. техн. наук. – Тверь, 2011. – 165 с.

50. Кривошеин М. ER: диаграммы сущность-связь/ Кривошеин Михаил [Электронный ресурс]: электронные статья/ mikkri.narod.ru. – Режим доступа: [http://mikkri.narod.ru/library/pdf/ER\\_Modeling.pdf](http://mikkri.narod.ru/library/pdf/ER_Modeling.pdf) (Дата обращения: 04.11.2016).

51. Логинов Н. В. Методы стохастической аппроксимации [Текст]/ Н. В. Логинов. – Автомат. и телемех. 1966, № 4. С. 185–204.

52. Лукашин Ю.П., Адаптивные методы краткосрочного прогнозирования временных рядов [Текст]/ Ю.П. Лукашин. – М.: Статистика, 2003. – 415 с.

53. Меладзе В. Курс технического анализа [Текст]/ Владимир Меладзе. – Днепропетровск: SAXON Finance Group, 2005. – 108 с.

54. Метод ближайшего соседа [Электронный ресурс]: информационная статья/ www.AIportal.ru// Портал искусственного интеллекта. – Режим доступа: <http://www.aiportal.ru/articles/autoclassification/single-link.html> (Дата обращения: 04.11.2016).

55. Метод ближайших соседей [Электронный ресурс]: информационная статья/ MachineLearning.ru. – Режим доступа: [http://www.machinelearning.ru/wiki/index.php?title=Метод\\_ближайшего\\_соседа](http://www.machinelearning.ru/wiki/index.php?title=Метод_ближайшего_соседа) (Дата обращения: 04.11.2016).

56. Метод максимального правдоподобия [Электронный ресурс]: информационная статья/ wikipedia.org. – Режим доступа: [http://ru.wikipedia.org/wiki/Метод\\_максимального\\_правдоподобия](http://ru.wikipedia.org/wiki/Метод_максимального_правдоподобия) (Дата обращения: 04.11.2016).

57. Метод наименьших квадратов [Электронный ресурс]: информационная статья/ wikipedia.org. – Режим доступа: [http://ru.wikipedia.org/wiki/Метод\\_наименьших\\_квадратов](http://ru.wikipedia.org/wiki/Метод_наименьших_квадратов) (Дата обращения: 04.11.2016).

58. Метод опорных векторов [Электронный ресурс]: информационная статья/ wikipedia.org. – Режим доступа: [http://ru.wikipedia.org/wiki/Метод\\_опорных\\_векторов](http://ru.wikipedia.org/wiki/Метод_опорных_векторов) (Дата обращения: 04.11.2016).

59. Метод стохастической аппроксимации для адаптации эконометрических моделей [Элек-

тронный ресурс]: информационная статья/ sergey.svetunkov.ru – Режим доступа: <http://sergey.svetunkov.ru/study/forecasting/files/223.pdf> (Дата обращения: 12.03.2015).

60. Методология IDEF0 [Электронный ресурс]: информационная статья/ itteach.ru. – Режим доступа: <http://itteach.ru/bpwin/metodologiya-idef0> (Дата обращения: 04.11.2016).

61. Миркес Е.А. Нейрокомпьютер. Проект стандарта [Текст]/ Е.А. Миркес. – Новосибирск: Наука, Сибирская издательская фирма РАН, 1998. – 190 с.

62. Миркес Е. М. Нейрокомпьютер. Проект стандарта. [Текст]/ Е. М. Миркес. – Новосибирск: Наука, 1999. – 337 с.

63. Мицель А.А. Прогнозирование динамики цен на фондовом рынке [Текст]/ Мицель, А.А., Ефремова Е.А.// Известия ТПУ, 2006, т. 309, №8. С. 197-201.

64. Мусман А.Э. Синтез алгоритмов на основе эволюционного программирования [Электронный ресурс]: информационная статья/ jurnal.org: – Режим доступа: <http://jurnal.org/articles/2010/infl1.html> (Дата обращения: 13.03.2015).

65. Нейрокомпьютерная техника - Перцептронная представляемость [Электронный ресурс]: информационная статья/ Cod:Net. – Режим доступа: <http://www.codenet.ru/progr/alg/nks/19.php> (Дата обращения: 04.11.2016).

66. Нейронные сети [Электронный ресурс]: информационная статья/ Neuroproject// Аналитические технологии для прогнозирования и анализа данных. – Режим доступа: <http://www.neuroproject.ru/neuro.php> (Дата обращения: 04.11.2016).

67. Нейронные сети адаптивного резонанса [Электронный ресурс]: информационная статья/ Портал искусственного интеллекта. – Режим доступа: <http://neuronus.com/theory/962-nejronnye-seti-adaptivnogo-rezonansa.html> (Дата обращения: 27.12.2017).

68. Николаева Ю. В. Индекс относительной силы для прогнозирования смены тренда [Текст] / Ю. В. Николаева, В. Е. Лялин // Математические модели и информационные технологии в организации производства. 2010. №2. С. 87-106.

69. Николаева Ю.В. Исследование влияния событий экономического календаря на волатильность котировок EURUSD [Текст] / Ю. В. Николаева// Международное научное периодическое издание «Новая наука: от идеи к результату» по итогам международной научно-практической конференции (Сургут, 22.10.2016 г.)/ в 3 ч. Ч.3. – Стерлитамак: АМИ, 2016. – С. 76-80.

70. Николаева Ю.В. Критерии выбора архитектуры нейронной сети для прогнозирования

финансовых рынков [Текст] / Ю.В. Николаева // Вестник Ижевского государственного технического университета. 2015. № 1. С. 96-97.

71. Николаева Ю.В. Математическая постановка задачи обучения многослойного перцептрона с точки зрения классической оптимизации [Текст] / Ю.В. Николаева // Вестник Ижевского Государственного технического университета. – 2016. № 1. С. 60-63.

72. Николаева Ю.В. Математическая постановка задачи обучения многослойного перцептрона для классификации рыночных ситуаций [Текст] / Ю.В. Николаева // Интеллектуальные системы в производстве. - 2016. - № 3. - С. 10-12.

73. Николаева Ю.В. Метод генерации выходной выборки для задачи нейросетевой классификации рыночных ситуаций [Текст] / Ю.В. Николаева // Вестник Иркутского государственного технического университета. - 2017. Т. 21. - №12. – С. 86-92. DOI: 10.21285/1814-3520-2017-12-86-92.

74. Николаева Ю.В. Методика нейросетевой классификации ситуаций на финансовых рынках и ее реализация в ИСППР [Текст] / Ю.В. Николаева // Интеллектуальные системы в производстве. - 2017. - №2. – С. 113-116.

75. Николаева Ю. В. Методы нейросетевой классификации ситуаций на финансовых рынках относительно ожидаемой прибыли [Электронный ресурс] / Ю. В. Николаева // Молодые ученые – ускорению научно-технического прогресса в XXI веке: сборник материалов IV Всероссийской научно-технической конференции аспирантов, магистрантов и молодых ученых с международным участием, Ижевск, 20-21 апреля 2016 года. – Ижевск: ИННОВА, 2016. – С. 562-566. – 1 электрон. опт. диск (CD-ROM).

76. Николаева Ю.В. Нормировка данных для нейронных сетей [Текст]/ Николаева Ю.В.// Молодые ученые – ускорению научно-технического прогресса в XXI веке: сб. тр. науч.- техн. конф. аспирантов, магистрантов и молодых ученых (Ижевск, 15-18 марта 2011 года). В 3 т. Т.1. – Ижевск: Изд-во ИжГТУ, 2011. С. 210-217.

77. Николаева Ю.В. Распознавание кризисных паттернов в котировках фьючерсов на нефть сорта Brent [Электронный ресурс] / Ю.В. Николаева // Молодые ученые – ускорению научно-технического прогресса в XXI веке: сборник материалов III Всероссийской научно-технической конференции аспирантов, магистрантов и молодых ученых с международным участием, Ижевск, 22-23 апреля 2015 года / ИжГТУ имени М.Т. Калашникова. 2015. С. 673-679.

78. Новиков Д.А. Структура теории управления социально-экономическими системами [Текст]/ Д.А. Новиков // УБС. 2009. №24. С. 216-257.

79. Новиков Д.А. Теория управления организационными системами [Текст]/ Д.А. Новиков. – М.: МПСИ, 2005. – 584 с.
80. Новиков Д.А., Чхартишвили А.Г. Активный прогноз [Текст]/ Д.А. Новиков, А.Г. Чхартишвили. – М.: ИПУ РАН, 2002. – 101 с.
81. Осминин К.П. Алгоритмы построения статистик для анализа и прогнозирования нестационарных временных рядов [Текст]/ К.П. Осминин // Информационные технологии вычислительные системы, 2009. № 1. С. 3 – 15.
82. Осминин К.П. Нестационарные временные ряды: Методы прогнозирования с примерами анализа финансовых и сырьевых рынков [Текст]/ Ю.Н. Орлов, К.П. Осминин. – М.: Книжный дом «ЛИБРОКОМ», 2011. – 384 с.
83. Орлов Ю.Н. Методика определения оптимального объема выборки для прогнозирования нестационарного временного ряда [Текст]/ Ю.Н. Орлов, К.П. Осминин // Информационные технологии вычислительные системы, 2008. № 3. С. 3 – 13.
84. Понтрягин Л.С. Математическая теория оптимальных процессов [Текст]/ Л.С. Понтрягин// 3-е изд. – М.: Наука, 1976. – 392 с.
85. Поспелов Д.А. Искусственный интеллект. Справочник. Книга 2. Модели и методы [Текст]/ Д.А. Поспелов. – М.: Радио и связь, 1990. – 304 с.
86. Прогнозирование цен на золото с помощью нейронных сетей STATISTICA SNN [Электронный ресурс]: информационная статья/ <https://www.statistica.ru>. – Режим доступа: [http://www.statistica.ru/statportal/tabID\\_32/MId\\_141/ModelID\\_0/PageID\\_354/DesktopDefault.aspx](http://www.statistica.ru/statportal/tabID_32/MId_141/ModelID_0/PageID_354/DesktopDefault.aspx) (Дата обращения: 04.11.2015).
87. Психология цвета, значение цвета [Электронный ресурс]: информационная статья/ [yugzone.ru](http://yugzone.ru). – Режим доступа: <http://www.yugzone.ru/psy/colors.htm> (Дата обращения: 04.11.2016).
88. Прогнозирование финансовых рынков с использованием искусственных нейронных сетей [Электронный ресурс]: электронные лекции/ [bookwu.net](http://bookwu.net). – Режим доступа: [http://bookwu.net/book\\_prognozirovanie-finansovyh-rynkov-s-ispolzovaniem-iskusstvennyh-nejronnyh-setej\\_1187/24\\_2.4.2.-klassifikaciya-rynochnyh-situacij](http://bookwu.net/book_prognozirovanie-finansovyh-rynkov-s-ispolzovaniem-iskusstvennyh-nejronnyh-setej_1187/24_2.4.2.-klassifikaciya-rynochnyh-situacij) (Дата обращения: 04.11.2016).
89. Психологическое восприятие и эмоциональное значение цвета [Электронный ресурс]: электронная статья/ [rosdesign.com](http://rosdesign.com). – Режим доступа: [http://rosdesign.com/design\\_materials2/cvet\\_3.htm](http://rosdesign.com/design_materials2/cvet_3.htm) (Дата обращения: 04.11.2016).

90. Раскин Джеф. Интерфейс: новые направления в проектировании информационных систем [Электронный ресурс]: электронный учебник/ Джеф Раскин. – Режим доступа: <http://raskin-interface.narod.ru/interface/index.htm> (Дата обращения: 04.11.2016).
91. Сергиенко А.Г. Моделирование прогноза значений котировок ценных бумаг с использованием нейронных сетей [Текст]/ Сергиенко А.Г.// дис. канд. техн. наук. – Санкт-Петербург, 2011. – 182 с.
92. Симанков В.С. Адаптивное управление сложными системами на основе теории распознавания образов [Электронный ресурс]: информационная статья/ В. С. Симанков, Е. В. Луценко – Режим доступа: <http://victor-safronov.ru/systems-analysis/books/simankov-lucenko/14.html> (дата обращения: 11.03.2015).
93. Система поддержки принятия решений [Электронный ресурс]: информационная статья/ wikipedia.org. – Режим доступа: [http://ru.wikipedia.org/wiki/Система\\_поддержки\\_принятия\\_решений](http://ru.wikipedia.org/wiki/Система_поддержки_принятия_решений) (Дата обращения: 04.11.2016).
94. Система управления базами данных [Электронный ресурс]: информационная статья/ wikipedia.org. – Режим доступа: [http://ru.wikipedia.org/wiki/Система\\_управления\\_базами\\_данных](http://ru.wikipedia.org/wiki/Система_управления_базами_данных) (Дата обращения: 04.11.2016).
95. Скользящая средняя [Электронный ресурс]: информационная статья/ wikipedia.org. – Режим доступа: <http://ru.wikipedia.org/wiki/Скользящая> (Дата обращения: 04.11.2016).
96. Сучкова Е.А., Николаева Ю.В. Разработка оптимальной структуры хранения данных для систем поддержки принятия решений [Электронный ресурс] / Е.А. Сучкова, Ю.В. Николаева // Кибернетика и программирование. – 2016. № 4. С. 58-64. DOI: 10.7256/2306-4196.2016.4.18281. – Режим доступа: [http://e-notabene.ru/kp/article\\_18281.html](http://e-notabene.ru/kp/article_18281.html) (Дата обращения: 04.11.2016).
97. Терехов С.А. Лекции по теории и приложениям искусственных нейронных сетей. Лекция 4. Персептрон Розенблатта [Электронный ресурс]: информационная статья/ Сергей А. Терехов// Лаборатория Искусственных Нейронных Сетей НТО-2, ВНИИТФ, Снежинск. – Режим доступа: [http://alife.narod.ru/lectures/neural/Neu\\_ch04.htm](http://alife.narod.ru/lectures/neural/Neu_ch04.htm) (Дата обращения: 04.11.2016).
98. Технический анализ: история появления и развития [Электронный ресурс]: информационная статья/ Форум E-ProfiCom.ru. – Режим доступа: <http://forum.e-proficom.ru/viewtopic.php?f=86&t=959> (Дата обращения: 04.11.2016).
99. Тихомиров Н.П. Эконометрика [Текст]/ Н.П. Тихомиров, Е.Ю. Дорохина. – М.: Изд-во Рос. экон. акад., 2002. – 640 с.



100. Финансовый рынок [Электронный ресурс]: информационная статья/ wikipedia.org. – Режим доступа: [http://ru.wikipedia.org/wiki/Финансовый\\_рынок](http://ru.wikipedia.org/wiki/Финансовый_рынок) (Дата обращения: 04.11.2016).
101. Форекс-календарь [Электронный ресурс] / [www.instaforex.com/](http://www.instaforex.com/) Форекс-аналитика / Форекс-календарь – 2016. Режим доступа: [https://www.instaforex.com/ru/forex\\_calendar](https://www.instaforex.com/ru/forex_calendar) (Дата обращения: 04.11.2016).
102. Фу К. Структурные методы в распознавании образов [Текст]/ К. Фу; пер. с англ. Н.В. Завалишина; под ред. М.А. Айзермана. – М.: Мир, 1977. – 319 с.
103. Фундаментальный анализ [Электронный ресурс]: информационная статья / wikipedia.org. – Режим доступа: [http://ru.wikipedia.org/wiki/Фундаментальный\\_анализ](http://ru.wikipedia.org/wiki/Фундаментальный_анализ) (Дата обращения: 04.11.2016).
104. Функции активации в нейронных сетях [Электронный ресурс]: информационная статья/ [www.AIportal.ru/](http://www.AIportal.ru/) Портал искусственного интеллекта – Режим доступа: <http://www.aiportal.ru/articles/neural-networks/activation-function.html> (Дата обращения: 04.11.2016).
105. Цыплаков А.А. Некоторые эконометрические методы. Метод максимального правдоподобия в эконометрике: Методическое пособие. [Текст]/ А.А. Цыплаков. – Новосибирск: НГУ, 1997. – 129 с.
106. Ширяев В. И. Финансовые рынки. Нейронные сети, хаос и нелинейная динамика. [Текст]/ В. И. Ширяев. – М.: Либроком, 2009. – 232 с.
107. Экономический календарь от FXStreet [Электронный ресурс] / [www.alpari.ru](http://www.alpari.ru/) // Аналитика Фогех от Альпари – 2016. Режим доступа: [http://www.alpari.ru/ru/analytics/calendar\\_fxstreet](http://www.alpari.ru/ru/analytics/calendar_fxstreet) (Дата обращения: 04.11.2016).
108. Экономический календарь [Электронный ресурс] / <http://ru.investing.com/> Экономический календарь – 2016. Режим доступа: <http://ru.investing.com/economic-calendar> (Дата обращения: 04.11.2016).
109. Экономический календарь [Электронный ресурс] / [www.fxstreet.ru.com](http://www.fxstreet.ru.com/) // Экономический календарь от FXStreet – 2016. Режим доступа: <http://www.fxstreet.ru.com/economic-calendar> (Дата обращения: 04.11.2016).
110. Энциклопедия трейдера. Технический анализ. История развития [Электронный ресурс]: информационная статья / Консультант online assessor.ru. – Режим доступа: <http://www.assessor.ru/forum/index.php?t=525> (Дата обращения: 04.11.2016).

111. Этапы проектирования баз данных [Электронный ресурс]: электронные лекции/ mstu.edu.ru. – Режим доступа: [http://www.mstu.edu.ru/study/materials/zelenkov/ch\\_5\\_1.html](http://www.mstu.edu.ru/study/materials/zelenkov/ch_5_1.html) (Дата обращения: 04.11.2016).
112. Artificial neuron [Электронный ресурс]: информационная статья/ wikipedia.org. – Режим доступа: [http://en.wikipedia.org/wiki/Artificial\\_neuron](http://en.wikipedia.org/wiki/Artificial_neuron) (Дата обращения: 04.11.2016).
113. Box, G. E. P. Time Series Analysis, Forecasting and Control [Текст]/ George E. P. Box and Gwilym M. Jenkins.// rev. Ed. San Francis-co: Holden-Day, 1976. – 575 pp.
114. Brillinger, David R. Time Series: Data Analysis and Theory [Текст]/ R. David Brillinger. – SIAM, 1975. – 504 pp.
115. Galushkin I.A. Neural Networks Theory [Текст]/ Alexander I. Galushkin // Springer-Verlag Berlin Heidelberg, 2007. – 396 pp.
116. Genevieve, Orr. Neural [Электронный ресурс]: информационная статья / Orr Genevieve, Schraudolph Nici, Cummins Fred. – Режим доступа: <http://www.willamette.edu/~gorr/classes/cs449/intro.html> (Дата обращения: 04.11.2016).
117. Grossberg, Stephen. Adaptive pattern classification and universal recoding, 1: Parallel development and coding of neural feature detectors [Text]/ Stephen Grossberg: – Biological Cybernetics, 23. 1976. – P.187-202.
118. Data mining [Электронный ресурс]/ информационная статья / wikipedia.org. – Режим доступа: [http://wikipedia.org/wiki/Data\\_mining](http://wikipedia.org/wiki/Data_mining) (Дата обращения: 04.11.2016).
119. Engle, R. F. Autoregressive Conditional Heteroscedasticity with Estimates of the Variance of United Kingdom Inflation [Text]/ Robert F. Engle// Econometrica, Vol. 50, No. 4, 1982. – P. 987-1007.
120. Hinton, G. E. Parallel computations for controlling an arm [Текст]/ G. E. Hinton// The Journal of Motor Behavior, 1984, 16. – P. 171-194.
121. Holt, C.C. Forecasting Trends and Seasonals by Exponentially Weighted Averages [Текст]/ Charles C. Holt // Carnegie Institute of Technology: Pittsburgh Office of Naval Research memorandum no. 52, 1957.
122. Kohonen, T. *Self-Organization and Associative Memory* [Текст]/ Teuvo Kohonen// Springer, second edition. 1988. – 312 pp.
123. Kros, B. An Introduction to neural networks [Text]/ Ben Kros, Patrick van der Smagt. – University of Amsterdam, 1996. – 137 pp.

124. Minsky M. Perceptrons: An Introduction to Computational Geometry [Text]/ Marvin Minsky, Seymour Papert. – The MIT Press, Cambridge MA, 1972. – 308 pp.
125. Multilayer Perceptron Neural Networks [Электронный ресурс]: информационная статья/ DTREG – Режим доступа: <https://www.dtreg.com/solution/view/21> (Дата обращения: 04.11.2016).
126. Nelson, D. B. Conditional heteroskedasticity in asset returns: A new approach [Text]/ Nelson D. B.// *Econometrica* № 59, 1991. – P. 347-370.
127. Neural Networks [Электронный ресурс]: информационная статья/ STATISTICA. – Режим доступа: <http://www.statsoft.com/textbook/neural-networks/> (Дата обращения: 04.11.2016).
128. *Nikolaeva J.V.* The neural networks. The multilayer perceptron [Текст] / *J.V. Nikolaeva* // Communication of Students, Master Students and Post-Graduates in Academic, Professional and Scientific Fields (Ижевск, 27 апреля-3 мая 2011 г.): материалы межвуз. студ. науч. конф. 2011. – P. 38-41.
129. Rosenblatt, Frank. The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain [Текст]/ Frank Rosenblatt // Cornell Aeronautical Laboratory, Psychological Review: v65. 1958. No. 6. – P. 386–408.
130. Rumelhart, D. E. Learning internal representations by error propagation [Текст]/ Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Volume 1 D. E. Rumelhart and J. L. McClelland, editors. – Foundations, MIT Press, Cambridge, MA. 1986. – P. 318-362.
131. Rumelhart, D. E. Learning internal representations by error propagation [Текст]/ Rumelhart David E., Hinton Geoffrey E., Williams Ronald J.// *Nature*, № 323. 1986. – P. 533–536.
132. Stock market [Электронный ресурс]: информационная статья/ Neuro AI. – Режим доступа: <http://www.learnartificialneuralnetworks.com/stockmarketprediction.html> (Дата обращения: 04.11.2016).
133. Turban, E. Decision support and expert systems: management support systems [Text]/ E. Turban. – Englewood Cliffs, N.J.: Prentice Hall, 1995. – 887 pp.
134. Winters, P.R. Forecasting sales by exponentially weighted moving averages [Текст]/ P.R. Winters. – *Management Science*, 1960, 6. – P. 324 - 342.
135. Yann LeCun, Leon Bottou, Y. Bengio. Gradient-based learning applied to document recognition / Yoshua Bengio, Yann LeCun, Leon Bottou, Patrick Haffner // *PIEEE*, 86(11):2278–2324, 1998.

136. Y. LeCun and Y. Bengio: Convolutional Networks for Images, Speech, and Time-Series, in Arbib, M. A. (Eds), *The Handbook of Brain Theory and Neural Networks*, MIT Press, 1995.



ПРИЛОЖЕНИЕ А. СВИДЕТЕЛЬСТВА О РЕГИСТРАЦИИ ПРОГРАММЫ ДЛЯ  
ЭВМ

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2016661369

Система поддержки принятия решений «НейроПрофит»

Правообладатель: *федеральное государственное бюджетное образовательное учреждение высшего образования «Ижевский государственный технический университет имени М.Т. Калашникова» (RU)*

Автор: *Николаева Юлия Викторовна (RU)*



Заявка № 2016618968

Дата поступления 22 августа 2016 г.

Дата государственной регистрации

в Реестре программ для ЭВМ 07 октября 2016 г.

Руководитель Федеральной службы  
по интеллектуальной собственности

 Г.П. Ислев



РОССИЙСКАЯ ФЕДЕРАЦИЯ



## СВИДЕТЕЛЬСТВО

о государственной регистрации базы данных

**№ 2016621361**

**База данных системы поддержки принятия решений  
«НейроПрофит»**

Правообладатель: *федеральное государственное бюджетное образовательное учреждение высшего образования «Ижевский государственный технический университет имени М.Т. Калашникова» (RU)*

Автор: *Николаева Юлия Викторовна (RU)*

Заявка № **2016621147**

Дата поступления **22 августа 2016 г.**

Дата государственной регистрации

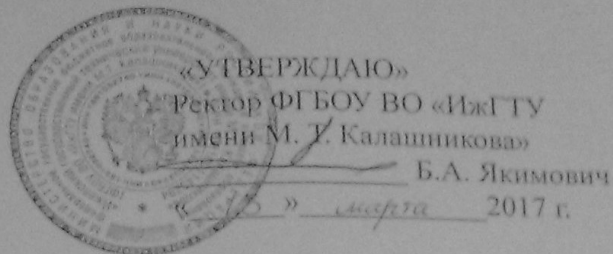
в Реестре баз данных **07 октября 2016 г.**

*Руководитель Федеральной службы  
по интеллектуальной собственности*

*Г.П. Излиев*



**ПРИЛОЖЕНИЕ Б. АКТ ВНЕДРЕНИЯ РЕЗУЛЬТАТОВ ДИССЕРТАЦИОННОЙ РА-  
БОТЫ**

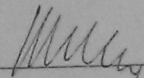


АКТ

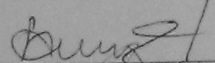
о внедрении (использовании) результатов  
диссертационной работы  
Николаевой Юлии Викторовны


Комиссия в составе: председатель – декан факультета ИВТ, д.т.н., профессор, Лялин В. Е., члены комиссии: д.т.н., профессор кафедры ЗИКС Сметанин А. М.; к.т.н., доцент кафедры ЗИКС Файзуллин Р. В.; составила настоящий акт о том, что отдельные результаты диссертационного исследования Николаевой Ю. В. использованы на кафедре ЗИКС ФГБОУ ВО «Ижевский государственный технический университет имени М. Т. Калашникова» для проведения лекционных, лабораторных и практических занятий студентов по дисциплинам «Методы оптимизации», «Базы данных», «Интеллектуальные информационные системы».

Председатель комиссии:

 (Лялин В. Е.)

Члены комиссии:

 (Сметанин А. М.)

 (Файзуллин Р. В.)



## ПРИЛОЖЕНИЕ В. ИСПОЛЬЗОВАНИЕ ВЕБ-ТЕХНОЛОГИЙ В СИСТЕМЕ

При работе системы поддержки принятия решений трейдеров финансовых рынков первоочередной задачей стоит получение котировок финансовых инструментов. Для решения данной задачи без непосредственного участия пользователя есть возможность использования сервисов, предоставляемых сетью интернет.

Сайт, который предоставляет подобный сервис, – это официальный сайт инвестиционной компании «Финам» [www.finam.ru](http://www.finam.ru). Для того, чтобы воспользоваться данным сервисом необходимо на сайте перейти на вкладку «Экспорт». На данной вкладке указываются параметры экспортируемых данных, и выполняется непосредственно экспорт по средствам нажатия кнопки «Получить файл». На рисунке ПВ.1 представлен скриншот данной вкладки сайта.

The screenshot shows the 'Export' page for GAZPROM AO on the Finam website. The page is titled 'МосБиржа акции' and 'GAZПРОМ оо'. The main content area contains the following settings:

- Интервал и периодичность:** 01.08.2016 to 07.09.2016, 1 день
- Имя выходного файла:** GAZP\_160801\_160907, .csv
- Имя контракта:** GAZP
- Формат:** даты: гggmmdd, времени: ччmmcc
- Выдавать время:**  начала свечи,  окончания свечи,  московское
- Разделитель:** полей: запятая (,), разрядов: нет
- Формат записи в файл:** TICKER, PER, DATE, TIME, OPEN, HIGH, LOW, CLOSE
- Добавить заголовок файла:**
- Заполнять периоды без сделок:**

A red box highlights the 'Получить файл' button. Below the form, there is a note: 'На выходе вы получите текстовый файл, который через DownLoader Вы сможете легко конвертировать в формат MetaStock.' and a footer: 'Данные предоставлены ПАО Московская Биржа. Перераспространение информации только с разрешения'.

Рисунок ПВ.1 –Вкладка «Экспорт»

Рассмотрим подробнее форму, в которой выбираются параметры экспорта данных (смотреть рисунок ПВ.1).

На форме можно изменять следующие параметры: секцию рынка, контракт, дата начала и конца, периодичность, имя выходного файла, имя контракта, формат даты и времени и т.п. После заполнения формы нажимается кнопка «Получить файл» и на компьютер пользователя загружается соответствующий файл с сервера, содержащий котировки выбранного инструмента с выбранными параметрами.



Для получения данных в автоматическом режиме можно воспользоваться веб-запросами к указанному сайту. В данном случае возможно отправление get-запроса на сервер <http://export.finam.ru> с нужными параметрами, например, следующими:

[http://export.finam.ru/export9.out?d=d&market=5&em=901&df=17&mf=10&yf=2015&dt=1&mt=10&yt=2016&p=8&f=USDRUR\\_151117\\_161101&e=.csv&cn=USDRUR&dtf=3&tmf=4&MSOR=1&mstime=on&mstimever=1&sep=3&sep2=1&datf=2&at=1&fsp=0](http://export.finam.ru/export9.out?d=d&market=5&em=901&df=17&mf=10&yf=2015&dt=1&mt=10&yt=2016&p=8&f=USDRUR_151117_161101&e=.csv&cn=USDRUR&dtf=3&tmf=4&MSOR=1&mstime=on&mstimever=1&sep=3&sep2=1&datf=2&at=1&fsp=0)

В запросе необходимо указать следующие параметры:

<http://export.finam.ru/export9.out> ?d=d& market=секция рынка& em=контракт&df=число начала &mf=месяц начала & yf=год начала & dt=число конца & mt= месяц конца & yt=год конца & p=периодичность & f=названия файла & e=.формат & sp=названия финансового инструмента & dtf=формат даты & tmf= формат времени & MSOR=отображаемое время &mstime = использовать московское время &mstimever = 1 & sep=разделитель полей & sep2= разделитель разрядов & datf =формат записи в файл & at=присутствие заголовка файла & fsp= заполнение периодов без сделок

Таким образом запрос состоит из множества параметров:

{Название файла. расширение файла; секция рынка; контракт; число начала; месяц начала; год начала; число конца; месяц конца; год конца; периодичность; названия файла; формат; названия финансового инструмента; формат даты; формат времени; отображаемое время; разделитель полей; разделитель разрядов; формат записи в файл; присутствие заголовка; заполнение периодов без сделок}.

Некоторые параметры одинаковы для всех запросов, отправляемых проектируемой системой поддержки принятия решений:

- e=.csv – скачивать файл в формате .csv;
- dtf=3 – формат даты «ддммгг»;
- tmf=4 – формат времени «чч:мм»;
- MSOR=0 – выдавать время начала свечи;
- mstime = on – использовать рабочее время;
- sep=3 – разделитель полей «;»;
- sep2=1 – нет разделителей разрядов;
- datf=2 – формат записи в файл «TICKER, PER, DATE, TIME, OPEN, HIGH, LOW, CLOSE», то есть сокращенное название финансового инструмента, номер, дата, время, цена открытия, самая высокая цена за период, самая низкая цена за период, цена закрытия.
- at=1 – заголовок файла присутствует;
- fsp=0 – не заполнять периоды без сделок.

Остальные параметры меняются динамически, в зависимости от выбранных пользователем параметров, по которым необходимо в данный момент получить рекомендацию.

Названия файла формируется из названия финансового инструмента, даты начала: год, месяц, число и даты конца: год, месяц, и расширения. Пример: GAZP\_160601\_160609.csv.

Секция рынка: market = {1, 5, 24},

где 1 – МосБиржа; 5 – рынок валют; 24 – рынок сырья.

Контракт: em = {16842, 17273, 83, 666999, 901, 19473},

где 16842 – номер контракта для акций ОАО «Газпром»;

17273 – номер контракта для акций ОАО «Роснефть»;

83 – номер контракта для валютной пары EURUSD;

86 – номер контракта для валютной пары GBPUSD;

901 – номер контракта для валютной пары USDRUB;

19473 – номер контракта для нефти сорта Brent.

По желанию можно увеличить множество секций рынка и контрактов в пределах, представленных на сайте.

Число начала/конца: df = dt = {1...31}.

Месяц начала/конца: mf = mt = {0...11},

где 0 – январь, 1 – февраль, ..., 11 – декабрь.

Год начала/конца: yf = yt = {2001...2011}.

Периодичность: p = {5, 6, 7, 8, 9},

где 5 – периодичность цен 15 мин, 6 – 30 мин, 7 – 1 час, 8 – 1 день, 9 – неделя.

С помощью одного из описанных ранее инструментальных средств извлечения данных из интернета, используемых в среде разработки Visual Studio, можно отправлять запросы на сервер с заданными пользователем на форме параметрами. Пользователь задает название инструмента и периодичность, с которыми он желает поработать. Параметры даты вычисляются автоматически: датой окончания является текущая дата, а дата начала вычисляется по следующей формуле: Дата начала = Датой окончания – 200 периодов.

Примеры таких запросов представлены ниже:

Для валютной пары EURUSD с 25 октября по 1 ноября 2016, периодичность 15 мин:

[http://export.finam.ru/export9.out?d=d&market=5&em=83&df=25&mf=9&yf=2016&dt=1&mt=10&yt=2016&p=5&f=EURUSD\\_161025\\_161101&e=.csv&cn=EURUSD&dtf=3&tmf=4&MSOR=1&mstime=on&mstimever=1&sep=3&sep2=1&datf=2&at=1&fsp=0](http://export.finam.ru/export9.out?d=d&market=5&em=83&df=25&mf=9&yf=2016&dt=1&mt=10&yt=2016&p=5&f=EURUSD_161025_161101&e=.csv&cn=EURUSD&dtf=3&tmf=4&MSOR=1&mstime=on&mstimever=1&sep=3&sep2=1&datf=2&at=1&fsp=0).

Для акций ОАО «Газпром» с 1 мая по 9 июня 2016, периодичность 1 час:

[http://export.finam.ru/export9.out?d=d&market=1&em=16842&df=1&mf=4&yf=2016&dt=9&mt=5&yt=2016&p=7&f=GAZP\\_160501\\_160609&e=.csv&cn=GAZP&dtf=3&tmf=4&MSOR=1&sep=3&sep2=1&datf=4&at=1](http://export.finam.ru/export9.out?d=d&market=1&em=16842&df=1&mf=4&yf=2016&dt=9&mt=5&yt=2016&p=7&f=GAZP_160501_160609&e=.csv&cn=GAZP&dtf=3&tmf=4&MSOR=1&sep=3&sep2=1&datf=4&at=1).

В итоге выполнения запроса (response запроса) получаем файл в формате csv, дальнейшая обработка которого позволяет загрузить и обрабатывать данные по котировкам в системе.

Следующим важным этапом в анализе финансового рынка является анализ экономического календаря, соответствующего финансовому инструменту. Сервисы с финансовыми календарями также предоставляются множеством сайтов. В разработанной системе используется сервис, предоставляемый сайтом <http://ru.investing.com> (сайт аналитики для инвесторов).

Для получения актуальных событий из экономического календаря воспользуемся аналогичным образом веб-запросами. В данном случае это будет get-запрос для выгрузки данных событий в формате Html. Запрос отправляется на сервер поставщика данных для экономического календаря, которым пользуются многие аналитические сайты по финансовым рынкам.

Пример запроса:

[http://ec.ru.forexprostools.com/?columns=exc\\_flags,exc\\_currency,exc\\_importance,exc\\_actual,exc\\_forecast,exc\\_previous&features=datepicker,timezone&countries=4,5,17,25&calType=day&timeZone=18&lang=1](http://ec.ru.forexprostools.com/?columns=exc_flags,exc_currency,exc_importance,exc_actual,exc_forecast,exc_previous&features=datepicker,timezone&countries=4,5,17,25&calType=day&timeZone=18&lang=1).

В данном запросе множество параметров оставляем неизменными для анализа по различным финансовым инструментам, в том числе columns==exc\_flags, exc\_currency, exc\_importance, exc\_actual, exc\_forecast, exc\_previous – перечисление колонок для вывода данных (время, валюта, важность, текущее, прогнозное и предыдущее значение), calType= day – получение данных за день, timeZone=1 – временная зона (московское время).

Изменяемым параметром является список стран, по которым необходимо получать события – countries. Список стран задается в формате списка номеров, соответствующих странам countries = {4,5,17,26,10,22,56, 72}, где:

4 – Великобритания; 5 – Америка; 17 – Германия; 26 – Испания; 10 – Италия; 22 – Франция; 56 – Россия; 72 – Еврозона.

Список стран может быть расширен в зависимости от анализируемых системой финансовых инструментов. Для каждого финансового инструмента в параметре countries запроса может передаваться сразу несколько номеров стран, например, countries =4,5,72.

В итоге выполнения запроса (response запроса) мы получаем файл в формате html, дальнейшая обработка которого позволяет загрузить и обрабатывать данные по событиям экономического календаря в системе.

Программная реализация описанных методов:

```

public string LoadData(string FinIn, string timeOt)
{ // метод получения данных из интернета:
  // на вход: FinIn - наименование фин. INSTR., timeOt - период
  // выход: наименование скаченного файла
  string datEndForLoad = DateTime.Now.ToShortDateString();// дата конца
  string Market, EM, df, mf, yf, dt, mt, yt;
  string nameFileCSV = "";
  // определить market для запроса
  Market = "5";
  if (FinIn == "GAZP" || FinIn == "ROSN")
    Market = "1";
  if (FinIn == "EURUSD" || FinIn == "GBPUSD" || FinIn == "USDRUR")
    Market = "5";
  if (FinIn == "ICE.BRN")
    Market = "24";
  // определить em для запроса
  EM = "16842";
  if (FinIn == "GAZP")
    EM = "16842";
  if (FinIn == "ROSN")
    EM = "17273";
  if (FinIn == "EURUSD")
    EM = "83";
  if (FinIn == "GBPUSD")
    EM = "86";
  if (FinIn == "USDRUR")
    EM = "901";
  if (FinIn == "ICE.BRN")
    EM = "19473";
  /* временной период timeOt:
    5 - 15 мин
    6 - 30 мин
    7 - час
    8 - ежедневные

```

```

    9 - неделя */
// определить дату начала (как дата конца - 250 периодов)
int shPer = 0;
int shPerDop = 0;
dt = datEndForLoad.Substring(0, 2); // дата конца
mt = datEndForLoad.Substring(3, 2); //месяц конца
yt = datEndForLoad.Substring(6, 4); //год конца
int dEndInt = 1;
int mEndInt = 1;
int yEndInt = 1;
DateTime dateEnd;
DateTime dateStart;
try
{ // собрать дату конца
    dEndInt = Convert.ToInt32(dt, 10);
    mEndInt = Convert.ToInt32(mt, 10);
    yEndInt = Convert.ToInt32(yt, 10);
    dateEnd = new DateTime(yEndInt, mEndInt, dEndInt);

    // если дата с 1 по 10 января, сдвинуть ее на 30.12 прошлого года
    if (mEndInt == 1 && dEndInt >= 1 && dEndInt <= 10)
        dateEnd = new DateTime(yEndInt - 1, 12, 30);
    // если период неделя, найти предыдущий понедельник
    if (timeOt == "9")
    {
        while (dateEnd.DayOfWeek == DayOfWeek.Monday)
        {
            dateEnd = dateEnd.AddDays(-1);
        }
    }
}
catch (Exception)
{
    return nameFileCSV;
}

```

```

        throw;
    }
    df = "13"; // день начала
    mf = "03"; // месяц начала
    yf = "2016"; // год начала
    DateTime newdateEnd = dateEnd;
    int countForDop = 24;
    bool weekend = false;
    if (DateTime.Now.AddDays(-1).DayOfWeek == DayOfWeek.Saturday ||
    DateTime.Now.AddDays(-1).DayOfWeek == DayOfWeek.Sunday ||
        DateTime.Now.AddDays(-2).DayOfWeek == DayOfWeek.Saturday ||
    DateTime.Now.AddDays(-2).DayOfWeek == DayOfWeek.Sunday)
        weekend = true;
    if (Market == "1")
        countForDop = 9;
    if (timeOt == "6")
    {
        if (!weekend)
            countForDop = countForDop * 2;
    }
    if (timeOt == "5")
    {
        if (weekend)
            countForDop = countForDop * 2;
        else
            countForDop = countForDop * 4;
    }
    int count = 250; // Общее количество
    if (timeOt == "7" || timeOt == "6" || timeOt == "5")
        count = 360; // 280
    if (finin == "USDRUR")
        count = 350;
    MessageBox.Show(countForDop.ToString());
    while (shPer < count)

```

```

{ // 250 - на случай праздников берем больше, далее из файла будут считаны 234
значения

// уменьшаем дату на соответствующий период
if(timeOt == "8")
{ // ежедневные
    newdateEnd = newdateEnd.AddDays(-1);

    // если дата не выходной, увеличиваем счетчик
    if(Market == "1" || Market == "24")
    {
        if (newdateEnd.DayOfWeek != DayOfWeek.Saturday && newdateEnd.DayOfWeek != DayOfWeek.Sunday &&
!(newdateEnd.Month == 1 && newdateEnd.Day >= 1 && newdateEnd.Day
<= 10))

            shPer++;
    }
    else
        shPer++;
}
if(timeOt == "9")
{ // еженедельные
    newdateEnd = newdateEnd.AddDays(-7);

    // если дата не выходной, увеличиваем счетчик
    if(Market == "1" || Market == "24")
    {
        if (!(newdateEnd.Month == 1 && newdateEnd.Day >= 1 && newdateEnd.Day
<= 10))

            shPer++;
    }
    else
        shPer++;
}
}

```

```

if (timeOt == "7")
{ // ЧАСОВЫЕ
    if (shPerDop >= countForDop)
    {
        newdateEnd = newdateEnd.AddDays(-1);
        shPerDop = 0;
    }
    shPerDop++;
    if (Market == "1" || Market == "24")
    {
        if (newdateEnd.DayOfWeek != DayOfWeek.Saturday && newdateEnd.DayOfWeek != DayOfWeek.Sunday &&
            !(newdateEnd.Month == 1 && newdateEnd.Day >= 1 && newdateEnd.Day <= 10))
            shPer++;
    }
    else
        shPer++;
}

if (timeOt == "6")
{ // ПОЛУЧАСОВЫЕ
    if (shPerDop >= countForDop)
    {
        newdateEnd = newdateEnd.AddDays(-1);
        shPerDop = 0;
    }
    shPerDop++;
    if (Market == "1" || Market == "24")
    {
        if (newdateEnd.DayOfWeek != DayOfWeek.Saturday && newdateEnd.DayOfWeek != DayOfWeek.Sunday &&
            !(newdateEnd.Month == 1 && newdateEnd.Day >= 1 && newdateEnd.Day <= 10))

```



```

        shPer++;
    }
    else
        shPer++;
    }
    if (timeOt == "5")
    { // 15-ти минутные
        if (shPerDop >= countForDop)
        {
            newdateEnd = newdateEnd.AddDays(-1);
            shPerDop = 0;
        }
        shPerDop++;
        if (Market == "1")
        {
            if (newdateEnd.DayOfWeek != DayOfWeek.Saturday && newdateEnd.DayOfWeek != DayOfWeek.Sunday &&
                !(newdateEnd.Month == 1 && newdateEnd.Day >= 1 && newdateEnd.Day <= 10))
                shPer++;
        }
    }
    else
        shPer++;
    }
} // пока (shPer < 260 )

dateStart = newdateEnd;
string dateStartStr = dateStart.ToShortDateString();
df = dateStartStr.Substring(0, 2); // дата начала
mf = dateStartStr.Substring(3, 2); //месяц начала
yf = dateStartStr.Substring(6, 4); //год начала
// Console.WriteLine(newdateEnd.ToShortDateString());

// сформировать наименование файла: FinIn_дата нач_дат конц

```

```

string dateStartFormatForName = String.Format("{0}{1}{2}", yf.Substring(2, 2), mf,
df);

string dateEndFormatForName = String.Format("{0}{1}{2}", yt.Substring(2, 2), mt, dt);
nameFileCSV = String.Format("{0}_{1}_{2}", FinIn, dateStartFormatForName, da-
teEndFormatForName);

// сдвинуть месяцы для запроса
int mtIntSdv = dateEnd.Month - 1;
mt = mtIntSdv.ToString();
int mfIntSdv = dateStart.Month - 1;
mf = mfIntSdv.ToString();

// обрезать дату дня для запроса
if (dt == "01" || dt == "02" || dt == "03" || dt == "04" || dt == "05" || dt == "06" || dt ==
"07" || dt == "08" || dt == "09")
    dt = dt.Substring(1, 1);
if (df == "01" || df == "02" || df == "03" || df == "04" || df == "05" || df == "06" || df ==
"07" || df == "08" || df == "09")
    df.Substring(1, 1);

// сформировать строку get-запроса к серверу финама
string query = "http://export.finam.ru/export9.out" +
    "?d=d&market=" + Market + "&em=" + EM + "&df=" + df + "&mf=" + mf + "&yf="
+ yf + "&dt=" + dt + "&mt=" + mt + "&yt=" + yt
    + "&p=" + timeOt + "&f=" + nameFileCSV + "&e=.csv&cn=" + FinIn +
"&dtf=3&tmf=4&MSOR=1" +
    "&mstime=on&mstimever=1&sep=3&sep2=1&datf=2&at=1&fsp=0";
MessageBox.Show(query);

// выполнить запрос
HttpWebRequest req = (HttpWebRequest)HttpWebRequest.Create(query);
HttpWebResponse resp = (HttpWebResponse)req.GetResponse();
Stream str = resp.GetResponseStream();
nameFileCSV = nameFileCSV + ".csv";
string path = @"c:\ispprNP\" + nameFileCSV;
if (File.Exists(path))
{
    File.Delete(path);
}

```

```

    }
    FileStream fs = File.Create(path);
    str.CopyTo(fs);
    fs.Close();
    resp.Close();
    str.Close();
    return nameFileCSV;
}
public void GetEconomicCalendar(bool flagAll)
{
    // сформировать строку get-запроса к серверу экономических новостей
    bool need = false;
    if (htmlEcNews == "")
    {
        string numbCountryList = "4,5,17,25,26,10,22,56, 72";
        string finInSelected = comboBoxFinIn.SelectedItem.ToString();
        string[] keyWord = new string[6];
        switch (finInSelected)
        {
            case "EURUSD":
                numbCountryList = "5,72,17,26,10,22";
                break;
            case "GBPUSD":
                numbCountryList = "4,5,72";
                break;
            case "USDRUR":
                numbCountryList = "5,56";
                break;
            case "Газпром":
                numbCountryList = "5,56,72,17,26,10,22";
                keyWord = new string[6] { "нефт", "топлив", "дистиллят", "бензин", "мазут",
"газ" };
                break;
            case "Роснефть":

```

```

        numbCountryList = "5,56,72,17,26,10,22";
        keyWord = new string[5] { "нефт", "топлив", "дистиллят", "бензин", "мазут" };
        break;
    case "Brent":
        numbCountryList = "5,56,72,17,26,10,22";
        keyWord = new string[6] { "нефт", "топлив", "дистиллят", "бензин", "мазут",
"опек" };

        break;
    default:
        numbCountryList = "4,5,17,25,26,10,22,56,72";
        break;
    }
    string query =
"http://ec.ru.forexprostools.com/?columns=exc_flags,exc_currency,exc_importance,exc_actual,exc_fo
recast,exc_previous&features=datepicker,timezone&countries=" + numbCountryList +
"&calType=day&timeZone=18&lang=1";
    HttpWebRequest req = (HttpWebRequest)HttpWebRequest.Create(query);
    HttpResponseMessage resp = (HttpResponseMessage)req.GetResponse();

    Stream str = resp.GetResponseStream();
    StreamReader strRead = new StreamReader(str, Encoding.GetEncoding("utf-8"));
    string stline = "";
    int i = 1;
    string valut = "";
    string volat = "";
    string newsStr = "";
    string country = "";
    string eventN = "";
    string eventType = "";
    string eventTime = "";
    string periodType = "";
    bool highVolat = false;
    int counterMiddle = 0;
    periodType = comboBoxPer.SelectedItem.ToString();

```

```

int maxTime = 1;
switch (periodType)
{
    case "15 мин":
        maxTime = 3;
        break;
    case "30 мин":
        maxTime = 5;
        break;
    default:
        maxTime = 24;
        break;
}

string allStr = "";
while ((stline = strRead.ReadLine()) != null)
{
    allStr += stline;
}
resp.Close();
str.Close();
HtmlAgilityPack.HtmlDocument HD = new HtmlAgilityPack.HtmlDocument();
HD.LoadHtml(allStr);
string htmlItog = "";
var str2 = HD.DocumentNode.SelectNodes("//tbody");
DateTime now = DateTime.Now;
now = now.AddHours(-1); // сдвинуть на московское время
foreach (HtmlNode node in str2)
{
    int k = 0;
    foreach (HtmlNode nodeCh in node.ChildNodes)
    {
        bool needWriteToItog = true;
        if (k > 0 && nodeCh.InnerHtml.ToString().Contains("first left time"))

```

```

{
    bool needWrite = false;
    foreach (HtmlNode nodeChCh in nodeCh.ChildNodes)
    {
        if (!String.IsNullOrEmpty(nodeChCh.InnerHtml.ToString()) &&
!String.IsNullOrEmpty(nodeChCh.InnerHtml.ToString()))
        {
            var classCh = nodeChCh.GetAttributeValue("class", null);
            if (classCh.ToString() == "first left time")
            {
                eventTime = nodeChCh.InnerText.ToString();
                int hourEv = Convert.ToInt32(nodeChCh.InnerText.ToString().Substring(0, 2));
                int minEv = Convert.ToInt32(nodeChCh.InnerText.ToString().Substring(3, 2));

                DateTime dateEvent = new DateTime(now.Year, now.Month,
now.Day, hourEv, minEv, now.Second);

                TimeSpan ts = dateEvent - now; // если ts < 0, значить событие уже
было

                if (ts.Hours >= -maxTime & ts.Hours <= maxTime)
                    needWrite = true;
                else
                    needWrite = false;
            }
            if (needWrite)
            {
                if (classCh.ToString() == "flagCur")
                {
                    HtmlNode nodeSpan = nodeChCh.FirstChild;
                    country = nodeSpan.GetAttributeValue("title", null);
                }
            }
        }
    }
}

```

```

if(classCh.ToString() == "sentiment")
{
    volat = nodeChCh.GetAttributeValue("title", null);
    if(!highVolat && volat == "Высокая волатильность")
        highVolat = true;
    if(!highVolat && volat == "Средняя волатильность")
        counterMiddle++;
}
if(classCh.ToString() == "left event")
{
    eventN = nodeChCh.InnerText.ToString().Replace("&nbsp;",
"").TrimEnd();

    // проверить событие на вхождение ключевых слов, если они
есть

    bool keyExists = false;
    if(!String.IsNullOrEmpty(keyWord[0]))
    {
        int counterKW = 0;
        while (!keyExists && counterKW <= 5)
        {
            if(eventN.Contains(keyWord[counterKW]))
                keyExists = true;
            counterKW++;
        }
    }
    else
        keyExists = true;
    if(!keyExists)
        needWrite = false;
}
}
}
}
}

```

```
        if (needWrite)
        {
            needWriteToItog = true;
            newsStr += "Время: " + eventTime + "Страна: " + country + ", Событие: "
+ eventN + " (" + volat + ")" + "\r\n";
        }
        else
            needWriteToItog = false;
    }
    if (needWriteToItog)
    {
        string htmlStr = nodeCh.WriteTo();
        htmlItog += htmlStr;
    }
    k++;
}
if (highVolat || counterMiddle > 0)
    textBoxVolatil.Text = "Риск повышения";
else
    textBoxVolatil.Text = "Обычная";
}
}
```



**ПРИЛОЖЕНИЕ Г. ПРИМЕРЫ ИСХОДНЫХ КОДОВ АЛГОРИТМОВ РАБОТЫ  
ИСППР «НЕЙРОПРОФИТ»**

```

public void CalcRSI()
{ // RSI на 7 меньше, чем цен

    double[] x = new double[233]; // [7]; // массивы повышения цен
    double[] y = new double[233]; // [7]; // массивы понижения цен
    double[] xsev = new double[227]; // [1]; // массивы сумм повышений за 7 дней
    double[] ysev = new double[227]; // [1]; // массивы сумм понижения за 7 дней

    for (var i = 0; i < 233; i++)
    { // цикл вычисления разницы между
        if (arrCen[i + 1] > arrCen[i])
        {
            x[i] = arrCen[i + 1] - arrCen[i];
            y[i] = 0;
        }
        else
        {
            y[i] = arrCen[i] - arrCen[i + 1];
            x[i] = 0;
        }
    }

    for (var i = 0; i < 227; i++)
    { // цикл вычисления сумм повышений и понижений цен за семь дней
        xsev[i] = x[i + 6];
        ysev[i] = y[i + 6];

        for (var j = 1; j < 7; j++)
        {
            double a = i + 6 - j;
            xsev[i] = xsev[i] + x[i + 6 - j];
            ysev[i] = ysev[i] + y[i + 6 - j];
        }
        arrRSI[i] = 100 * (xsev[i] / (xsev[i] + ysev[i]));
    }
}

public void CalcMACD() // на вход массив цен
{ // MACD на 33 меньше, чем цен (всего цен 234, macd 200, rsi 227)
    // double[] cen = new double[35]; // массив цен - глобальная переменная 234

    double[] macdgis = new double[201]; // [2]; // массив значений КДСС-гистограммы
    double[] ema12 = new double[223]; // [24]; // массив значений ЭСС12
    double[] ema26 = new double[209]; // [10]; // массив значений ЭСС26
    double[] macd1 = new double[209]; // [10]; // массив значений КДСС-линии
    double[] macds = new double[201]; // [2]; // массив значений Сигнальной линии
    ema26[0] = arrCen[25]; // присваиваем начальные значения ЭСС12 и ЭСС26 как соот-
    ветствующее значение цены
    ema12[0] = arrCen[11];
    for (var i = 1; i < 209; i++)
    { // рекуррентно вычисляем остальные значения ЭСС26
        ema26[i] = fema(ema26[i - 1], arrCen[i + 25], 26);
    }

    for (var i = 1; i < 223; i++)
    { // рекуррентно вычисляем остальные значения ЭСС12
        ema12[i] = fema(ema12[i - 1], arrCen[i + 11], 12);
    }
}

```

```

for (var i = 0; i < 209; i++)
{
    // вычисление КДСС-линии
    macdl[i] = ema12[i + 14] - ema26[i];
}

macds[0] = macdl[8]; // вычисляем остальные значения сигнальной линии
for (var i = 1; i < 201; i++)
{
    macds[i] = fema(macds[i - 1], macdl[i + 8], 9);
}
for (var i = 1; i < 201; i++)
{
    macdgis[i] = macdl[i + 8] - macds[i]; // вычисляем КДСС-гистограмму
    arrMACD[i - 1] = macdgis[i];
}
}
public void CalcGARCH()
{
    //arrGARCH // массив значений по GARCH - глобальная переменная
    int n = 1;
    int m = 1;
    double[,] mastr = new double[n, m];
    double[,] c = new double[n, m];

    double[] doh = new double[201]; // массив доходов
    double[] arpsc = new double[201]; // массив значений по АРПС
    double[] osharpssc = new double[201]; // массив ошибок по модели АРПС
    double[] disp = new double[201]; // массив дисперсий
    double[,] gradosh = new double[200, 3]; // массив градиент ошибок
    double[,] q = new double[200, 3]; // матрица Q
    double[,] graddisp = new double[200, 7]; // градиент дисперсий
    double[,] s = new double[200, 7]; // матрица S
    double[,] estr = new double[200, 1]; // вектор Estr
    double[,] vi = new double[200, 1]; // вектор V
    double[,] qtr = new double[3, 200]; // матрица QT
    double[,] str = new double[7, 200]; // матрица ST
    double[,] qtrq = new double[3, 3]; // матрица QT* Q
    double[,] strsq = new double[7, 7]; // матрица ST* S
    double[,] qtrest = new double[3, 1]; // вектор QT* Estr
    double[,] strvi = new double[7, 1]; // вектор ST* V
    double[,] delt = new double[7, 1]; // вектор D
    double[] q0 = new double[7]; // вектор Q0 // или 7
    double[] q1 = new double[7]; // вектор Q1
    double sumdoh = 0;
    double srsumdoh;
    for (int i = 0; i < 201; i++)
    {
        doh[i] = arrCen[i + 33] - 2 * arrCen[i + 32] + arrCen[i + 31]; // вычисление
        // доходов
        sumdoh += doh[i];
    }
    srsumdoh = sumdoh / 201;
    for (int i = 0; i < 201; i++)
    {
        // вычисление значений по АРПС
        arpsc[i] = -0.2796 - 0.4657 * doh[i] - 0.326 * (doh[i] - srsumdoh);
        osharpssc[i] = doh[i] - arpsc[i]; // вычисление ошибок по АРПС
        disp[i] = sumsqr(osharpssc, i) / (i + 1); // вычисление дисперсий
    }

    for (int i = 0; i < 200; i++)
    {
        gradosh[i, 0] = -1; // вычисление матриц градиента ошибок
    }
}

```

```

gradosh[i, 1] = -doh[i]; //arpss [i];
gradosh[i, 2] = -osharpss[i];
graddisp[i, 0] = 0;
graddisp[i, 1] = 0;
graddisp[i, 2] = 0;
graddisp[i, 3] = 1;
graddisp[i, 4] = disp[i];
graddisp[i, 5] = Math.Pow(osharpss[i], 2);
graddisp[i, 6] = 0;
}
for (int i = 0; i < 200; i++)
{ // вычисление Q
    for (int j = 0; j < 3; j++)
    {
        q[i, j] = (1 / Math.Sqrt(disp[i + 1]) * gradosh[i, j]);
    }
}
for (int i = 0; i < 200; i++)
{ // вычисление S
    for (int j = 0; j < 7; j++)
    {
        s[i, j] = (1 / disp[i + 1]) * graddisp[i, j];
    }
}

for (int i = 0; i < 200; i++)
{ // вычисление векторов Estr, V
    estr[i, 0] = osharpss[i + 1] / Math.Sqrt(disp[i + 1]);
    vi[i, 0] = Math.Pow(estr[i, 0], 2) - 1;
}
masTrans(q, 200, 3, out qtr); // вычисление QT
masMultip(qtr, q, 3, 3, 200, out qtrq); // вычисление QT*Q

masTrans(s, 200, 7, out str); // вычисление ST
masMultip(str, s, 7, 7, 200, out strq); // вычисление ST*S

for (int i = 0; i < 7; i++)
{
    for (int j = 0; j < 7; j++)
    { // вычисление QTQ+0,5* STS
        if (i < 3 && j < 3)
        {
            strq[i, j] = qtrq[i, j] + 0.5 * strq[i, j];
        }
        else
        {
            strq[i, j] = 0.5 * strq[i, j];
        }
    }
}

masMultip(qtr, estr, 3, 1, 200, out qtrest); // вычисление QT*Estr
masMultip(str, vi, 7, 1, 200, out strvi); // вычисление ST*V
{// вычисление QT*Estr +0.5 ST*V
    if (i < 3)
    {
        strvi[i, 0] = qtrest[i, 0] + 0.5 * strvi[i, 0];
    }
    else
    {
        strvi[i, 0] = 0.5 * strvi[i, 0];
    }
}
}
}

```

```

masMultip(strs, strvi, 7, 1, 7, out delt); // вычисление delt
q0[0] = -0.279;
q0[1] = -0.465;
q0[2] = -0.0326;
for (int i = 0; i < 3; i++)
{ // вычисление новых коэффициентов
    q1[i] = q0[i] + 0.003 * delt[i, 0];
}
for (int i = 0; i < 200; i++)
{ // вычисление garch
    arrGARCH[i] = q1[0] + q1[1] * doh[i] + q1[2] * osharpss[i];
}
}
public void CalcIndicators()
{
    // рассчитать значения индикаторов
    CalcRSI();
    CalcMACD();
    CalcGARCH();

    // заполнить массивы для нн
    for (int i = 0; i < 200; i++)
    {
        arrCenForNN[i] = arrCen[i + 34];
    }
    for (int i = 0; i < 200; i++)
    {
        arrRSIForNN[i] = arrRSI[i + 27];
    }

    textBox1.AppendText("Значения индикаторов рассчитаны \r\n");
    // отнормировать все значения
    normirovka(arrCenForNN, out arrNormCenForNN);
    normirovka(arrRSIForNN, out arrNormRSIForNN);
    normirovka(arrMACD, out arrNormMACDForNN);
    normirovka(arrGARCH, out arrNormGARCHForNN); // с GARCH
    textBox1.AppendText("Нормировка данных завершена \r\n");
}
public void normirovka(double[] arr, out double[] arrNorm)
{
    double sr;
    double sum = 0;
    double[] otkl = new double[200];
    double[] otklsqr = new double[200];
    double[] disp = new double[200];
    double[] sko = new double[200];
    arrNorm = new double[200];
    for (int i = 0; i < 200; i++)
    {
        sum += arr[i];
    }
    sr = sum / 200;
    for (int i = 0; i < 200; i++)
    {
        otkl[i] = arr[i] - sr;
        otklsqr[i] = Math.Pow(otkl[i], 2);
        disp[i] = sumsqr(otkl, (i + 1)) / (i + 1);
        sko[i] = Math.Sqrt(disp[i]);
    }
    // нормируем гиперболическим тангенсом с коэф 1
    for (int i = 0; i < 200; i++)
    {
        arrNorm[i] = Math.Tan(otkl[i] / sko[i]);
    }
}

```

```

}
// устраняем всплески с коррекцией коэффициента нормировки
double ymax = 0;
for (int i = 0; i < 200; i++)
{
    double a = Math.Abs(arrNorm[i]);
    if (a > ymax)
        ymax = a;
}
double alph = (ymax * 4) / Math.PI;
for (int i = 0; i < 200; i++)
{
    double d = (otkl[i] / sko[i]) / alph;
    arrNorm[i] = Math.Tan(d);
}
}
public void CalcVolat()
{
    double[] arrEMA12 = new double[200];
    double[] arrE2HighCen = new double[200];
    double[] arrE2LowCen = new double[200];
    double DHighCen = 0;
    double DLowCen = 0;
    double[] arrE2DHighCen = new double[200];
    double[] arrE2DLowCen = new double[200];
    double[] SDCen = new double[200];
    double DDCen = 0;
    double[] ema12 = new double[200]; // массив значений ЭСС12
    ema12[0] = arrMCen[0];
    for (var i = 1; i < 200; i++)
    {
        // рекуррентно вычисляем остальные значения ЭСС12
        ema12[i] = fema(ema12[i - 1], arrMCen[i], 12);
    }
    // вычислить квадрат отклонений high от эсс12, low от эсс12
    for (var i = 0; i < 200; i++)
    {
        arrE2HighCen[i] = Math.Pow(arrHighCen[i] - ema12[i], 2);
        arrE2LowCen[i] = Math.Pow(arrLowCen[i] - ema12[i], 2);
        DHighCen += arrE2HighCen[i];
        DLowCen += arrE2LowCen[i];
    }
    DHighCen = DHighCen / 200; // среднеквадратичное отклонение high
    DLowCen = DLowCen / 200; // среднеквадратичное отклонение low
    // вычислить квадрат отклонений e2high от dhigh, e2low от dlow
    DDCen = 0;
    for (var i = 0; i < 200; i++)
    {
        arrE2DHighCen[i] = Math.Pow(arrE2HighCen[i] - DHighCen, 2);
        arrE2DLowCen[i] = Math.Pow(arrE2LowCen[i] - DLowCen, 2);
        SDCen[i] = arrE2DHighCen[i] + arrE2DLowCen[i];
        DDCen += SDCen[i];
    }
    DDCen = DDCen/200;
    for (var i = 0; i < 200; i++)
    {
        arrPrVoldCen[i] = SDCen[i] * 100 / DDCen;
    }

    // вычислить среднее значение квадрата отклонения high от эсс12, low от эсс12
    double ME2HighNorm = 0;
    double ME2HighVolat = 0;
    double ME2LowNorm = 0;
    double ME2LowVolat = 0;

```

```

double shNorm = 0;
double shVolat = 0;
for (var i = 0; i < 200; i++)
{
    if (arrPrVoldCen[i] >= 100) // ПОВЫШЕННАЯ ВОЛАТИЛЬНОСТЬ
    {
        ME2HighVolat += arrE2HighCen[i];
        ME2LowVolat += arrE2LowCen[i];
        shVolat++;
    }
    else
    {
        ME2HighNorm += arrE2HighCen[i];
        ME2LowNorm += arrE2LowCen[i];
        shNorm++;
    }
}
if (shVolat != 0)
{
    ME2HighVolat = ME2HighVolat / shVolat;
    ME2LowVolat = ME2LowVolat / shVolat;
}
if (shNorm != 0)
{
    ME2HighNorm = ME2HighNorm / shNorm;
    ME2LowNorm = ME2LowNorm / shNorm;
}
// % норм от волат = норм * 100/волат
double prHighNormVolat = ME2HighNorm * 100 / ME2HighVolat;
double prLowNormVolat = ME2LowNorm * 100 / ME2LowVolat;
prNormVolat = (prHighNormVolat + prLowNormVolat) / 2;
}
public void WriteWeightsNNSQL()
{
    // веса по каждому слову записать в БД
    InitConnectSQL();
    conn.Open();
    string perId;
    string fininId;
    string ocId;
    string recId;
    string rec = textBoxRec.Text;
    GetParamId(periodAll, fininAll, oцenka, rec, out perId, out fininId, out ocId, out
recId);

    // запись операции
    string query = String.Format("insert into Operation (DateOp, finIn, per, oc, prof,
rec) " +
        " values (NOW(), {0}, {1}, {2}, {3}, {4})", fininId, perId, ocId,
        textBoxProfit.Text.ToString().Replace(',', '.'), recId);

    MySqlCommand cmd = new MySqlCommand(query, conn);
    cmd.ExecuteNonQuery();

    string opId = "";
    string sql = "select Id from Operation order by Id desc Limit 1";
    MySqlCommand select = new MySqlCommand(sql, conn);
    MySqlDataReader dataReader = select.ExecuteReader();
    while (dataReader.Read())// Читаем
    {
        opId = dataReader.GetValue(0).ToString();
    }
    dataReader.Close();
}

```

```

// запись нейронной сети
query = "insert into NN (Op) values (" + opId + ") ";

MySQLCommand cmd2 = new MySQLCommand(query, conn);
cmd2.ExecuteNonQuery();

string NNId = "";
sql = "select Id from NN order by Id desc Limit 1";
MySQLCommand select2 = new MySQLCommand(sql, conn);
MySQLDataReader dataReader2 = select2.ExecuteReader();
while (dataReader2.Read())// Читаем
{
    NNId = dataReader2.GetValue(0).ToString();
}
dataReader2.Close();

// запись весов нейронной сети
LayerNW l1 = NET.Layer(0); // первый слой
double d;
query = "insert into Layer1 (nn, nom, w1, w2, w3, w4, w5, w6, w7, w8, w9, w10,
w11, w12," +
        " w13, w14, w15, w16, w17, w18, w19, w20, w21, w22 ) values (" + NNId
+ ", "; //
string queryDop = "";
for (int j = 0; j < 3; j++)
{
    queryDop = query + j.ToString() + ", ";
    for (int i = 0; i < 22; i++) // 18; i++)
    {
        d = Math.Round(l1[i, j], 5);

        queryDop += d.ToString().Replace(',', '.');
        if (i < 21)
            queryDop += ",";
    }
    queryDop += ")";
    MySQLCommand cmd3 = new MySQLCommand(queryDop, conn);
    cmd3.ExecuteNonQuery();
}
LayerNW l2 = NET.Layer(1); // второй слой
double w1 = Math.Round(l2[0, 0], 5);
double w2 = Math.Round(l2[1, 0], 5);
double w3 = Math.Round(l2[2, 0], 5);
query = String.Format("insert into Layer2 (nn, w1, w2, w3) values ({0}, {1}, {2},
{3}) ",
        NNId, w1.ToString().Replace(',', '.'),
w2.ToString().Replace(',', '.'),
        w3.ToString().Replace(',', '.'));
MySQLCommand cmd4 = new MySQLCommand(query, conn);
cmd4.ExecuteNonQuery();

LayerNW l3 = NET.Layer(1); // выходной слой
query = String.Format("insert into Layer3 (nn, w1) values ({0}, {1}) ",
        NNId, Math.Round(output, 5).ToString().Replace(',', '.'));
MySQLCommand cmd5 = new MySQLCommand(query, conn);
cmd5.ExecuteNonQuery();

conn.Close();
}

```

## ПРИЛОЖЕНИЕ Д. ЭКСПЕРИМЕНТАЛЬНЫЕ ДАННЫЕ

Таблица Д.1 Результаты экспериментов

№	Фин. ин-стр.	Период	Ожидаемая прибыль	Выданная рекомендация	Реальная ситуация на рынке (достижение макс. прибыли)	Процент достижения прибыли	Время обучения*	Ошибка тестирования
1	EURUSD	15 мин	0,0012	Покупать	Покупать	130	2	2,34
2	EURUSD	15 мин	0,0012	Покупать	Покупать	12	1,9	2,31
3	EURUSD	15 мин	0,0015	Покупать	Покупать	173	4	3,1
4	EURUSD	30 мин	0,0025	Продавать	Продавать	32	0,1	3,3
5	EURUSD	30 мин	0,001	Продавать	Продавать	80	1	4,7
6	EURUSD	30 мин	0,002	Продавать	Покупать	112	0,9	3,1
7	EURUSD	1 ч	0,0025	Покупать	Покупать	256	0,08	2,1
8	EURUSD	1 ч	0,0037	Покупать	Покупать	182	0,08	2
9	EURUSD	1 ч	0,0028	Продавать	Продавать	100	0,2	2,25
10	EURUSD	1 д	0,015	Покупать	Покупать	149	0,2	2,87
11	EURUSD	1 д	0,018	Продавать	Продавать	77	0,2	4,1
12	EURUSD	1 д	0,01	Продавать	Продавать	120	1,2	2,7
13	EURUSD	1 н	0,08	Покупать	Покупать	50	0,05	1,4
14	EURUSD	1 н	0,04	Покупать	Покупать	100	0,05	0,93
15	EURUSD	1 н	0,019	Покупать	Покупать	89	1,1	2,7
16	Газпром	15 мин	0,9	Продавать	Покупать	80	0,8	1,05
17	Газпром	15 мин	1,1	Покупать	Покупать	81	0,2	4,06
18	Газпром	15 мин	0,8	Покупать	Покупать	68	0,2	2,5
19	Газпром	30 мин	2,3	Покупать	Покупать	88	0,4	3,5
20	Газпром	30 мин	0,7	Покупать	Покупать	120	0,4	2,97
21	Газпром	30 мин	1,5	Продавать	Продавать	106	0,3	4,33
22	Газпром	1 ч	1	Продавать	Продавать	150	2,5	2,2
23	Газпром	1 ч	2,1	Покупать	Продавать	71	0,2	1,2
24	Газпром	1 ч	0,8	Продавать	Продавать	107	0,1	3,6
25	Газпром	1 д	4,1	Продавать	Продавать	114	0,2	3,3
26	Газпром	1 д	2,6	Покупать	Покупать	184	0,8	3
27	Газпром	1 д	4,2	Покупать	Покупать	114	0,4	3,21
28	Газпром	1 н	4,6	Продавать	Продавать	240	1,2	3,57
29	Газпром	1 н	6,9	Покупать	Продавать	137	0,05	3,4
30	Газпром	1 н	12	Покупать	Покупать	152	0,05	2,02
31	GBPUSD	15 мин	0,013	Покупать	Покупать	65	0,15	3,89
32	GBPUSD	15 мин	0,007	Ожидать	Покупать	115	0,1	1,2
33	GBPUSD	15 мин	0,0082	Покупать	Покупать	99	0,1	1,78
34	GBPUSD	30 мин	0,005	Продавать	Продавать	90	1,1	2,79
35	GBPUSD	30 мин	0,006	Продавать	Продавать	125	0,1	1,99
36	GBPUSD	30 мин	0,0056	Покупать	Продавать	132	0,1	3,2



## Продолжение таблицы Д.1

37	GBPUSD	1 ч	0,009	Покупать	Покупать	192	0,3	2,03
38	GBPUSD	1 ч	0,003	Продавать	Продавать	160	0,6	3
39	GBPUSD	1 ч	0,006	Продавать	Продавать	105	0,1	2,39
40	GBPUSD	1 д	0,017	Продавать	Продавать	607	2,5	3,76
41	GBPUSD	1 д	0,01	Продавать	Продавать	248	0,2	3,4
42	GBPUSD	1 д	0,02	Покупать	Продавать	448	3	4,5
43	GBPUSD	1 н	0,04	Покупать	Покупать	131	0,2	3,88
44	GBPUSD	1 н	0,09	Продавать	Продавать	126	0,5	2,18
45	GBPUSD	1 н	0,065	Продавать	Продавать	69	0,2	1,5
46	Роснефть	15 мин	1,2	Продавать	Покупать	125	0,3	3,15
47	Роснефть	15 мин	1,6	Продавать	Продавать	112	1,6	3,04
48	Роснефть	15 мин	1,9	Покупать	Покупать	131	0,2	3,68
49	Роснефть	30 мин	0,9	Продавать	Продавать	422	0,1	4,4
50	Роснефть	30 мин	2,2	Продавать	Продавать	180	0,1	4,1
51	Роснефть	30 мин	7,5	Покупать	Покупать	93	0,15	2,05
52	Роснефть	1 ч	2,6	Продавать	Покупать	167	2,1	2,54
53	Роснефть	1 ч	3	Покупать	Покупать	101	3	2,1
54	Роснефть	1 ч	5,4	Покупать	Покупать	106	2	3,29
55	Роснефть	1 д	6,5	Покупать	Покупать	282	1	2,89
56	Роснефть	1 д	8,9	Покупать	Покупать	221	3,5	2,86
57	Роснефть	1 д	9,5	Продавать	Продавать	119	3,9	5,9
58	Роснефть	1 н	20,1	Продавать	Покупать	225	1,2	2,66
59	Роснефть	1 н	26,7	Покупать	Покупать	131	0,4	2,25
60	Роснефть	1 н	24,5	Продавать	Продавать	90	0,2	3,34
61	Brent	15 мин	0,15	Продавать	Продавать	273	0,2	2,5
62	Brent	15 мин	0,5	Ожидать	Продавать	91	0,2	2,7
63	Brent	15 мин	0,21	Продавать	Продавать	85	0,25	2,4
64	Brent	30 мин	1,6	Продавать	Продавать	104	0,05	0,29
65	Brent	30 мин	0,4	Покупать	Покупать	65	0,05	2,72
66	Brent	30 мин	1,2	Продавать	Продавать	78,5	0,2	0,73
67	Brent	1 ч	0,6	Продавать	Покупать	105	0,1	2,8
68	Brent	1 ч	1,2	Продавать	Продавать	115	0,2	0,8
69	Brent	1 ч	1,1	Продавать	Продавать	170	0,2	1,98
70	Brent	1 д	2,4	Продавать	Покупать	101	0,1	3,88
71	Brent	1 д	7,2	Продавать	Продавать	77	0,05	0,79
72	Brent	1 д	3,4	Покупать	Покупать	76	0,2	2,33
73	Brent	1 н	6,9	Продавать	Продавать	89	0,2	1,15
74	Brent	1 н	5,6	Продавать	Продавать	110	0,1	3,45
75	Brent	1 н	7,9	Продавать	Продавать	286	0,1	2,5
76	USDRUR	15 мин	0,4	Покупать	Покупать	118	0,2	0,99
77	USDRUR	15 мин	0,6	Продавать	Покупать	123	0,1	1,6
78	USDRUR	15 мин	0,78	Покупать	Покупать	105	0,1	1,4

Продолжение таблицы Д.1

79	USDRUR	30 мин	0,3	Покупать	Покупать	69	0,05	1,03
80	USDRUR	30 мин	0,23	Продавать	Продавать	63	0,1	1,94
81	USDRUR	30 мин	0,4	Продавать	Продавать	99	0,2	0,97
82	USDRUR	1 ч	1,2	Покупать	Покупать	163	0,05	1,66
83	USDRUR	1 ч	0,5	Продавать	Продавать	117	0,5	3,53
84	USDRUR	1 ч	1,9	Продавать	Продавать	60	0,05	1,45
85	USDRUR	1 д	4,1	Продавать	Продавать	85	0,4	3,3
86	USDRUR	1 д	0,95	Покупать	Покупать	112	0,25	2,7
87	USDRUR	1 д	2,1	Покупать	Покупать	335	0,2	1,19
88	USDRUR	1 н	5,1	Ожидать	Продавать	27	0,05	0,34
89	USDRUR	1 н	1,2	Продавать	Продавать	115	2,2	1,25
90	USDRUR	1 н	4,1	Покупать	Покупать	282	0,3	1,4

\* – время обучения приведено при использовании следующего аппаратного и программного обеспечения:

Процессор: Intel(R) Core(TM) i5-4210U

CPU: 1.70 GHz 2.40 GHz

ОЗУ: 8,00 ГБ

Емкость диска: 518 ГБ (при занятых 410 ГБ)

Тип системы: 64-разрядная

ОС: Windows 8.1