

**Правительство Российской Федерации**  
**Федеральное государственное автономное образовательное**  
**учреждение высшего образования**  
**«Национальный исследовательский университет «Высшая школа экономики»**

**Факультет компьютерных наук**  
**Основная образовательная программа**  
**Прикладная математика и информатика**

## **ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА**

**на тему**

**Применение алгоритмов машинного обучения для моделирования биржевого агента**

**Выполнил студент группы БПМИ-123, 4 курса,**  
**Душацкий Аркадий Владиславович**

**Научный руководитель:**  
**к.ф-м.н, доцент, Вознесенская Тамара Васильевна**

## Содержание

|  |    |
|--|----|
| Введение.....  | 2  |
| Обзорно-постановочная часть.....   | 4  |
| Предметная область.....  | 4  |
| Объект и предмет исследования.....   | 5  |
| Обзор работ по теме исследования.....  | 6  |
| Цель и задачи работы.....  | 7  |
| Используемые методы.....   | 9  |
| Актуальность работы.....   | 10 |
| Новизна и практическая ценность работы.....                                  | 11 |
| Теоретическая часть.....   | 12 |
| Обучение с подкреплением.....  | 12 |
| Деревья решений.....   | 13 |
| Генетические алгоритмы.....  | 15 |
| Построение деревьев решений с помощью генетических алгоритмов.....           | 19 |
| Ансамбли деревьев решений.....   | 23 |
| Построение ансамблей деревьев решений с помощью генетических алгоритмов..... | 24 |
| Модель биржевого агента.....   | 24 |
| Симуляция агента.....  | 24 |
| Ограничения модели.....  | 26 |
| Технические индикаторы.....  | 27 |
| Эксперименты.....  | 32 |
| Описание данных.....   | 32 |
| Параметры алгоритмов.....  | 32 |
| Результаты.....  | 33 |
| Выводы.....  | 34 |
| Заключение.....  | 34 |

## **1. Введение**

Работа посвящена применению алгоритмов машинного обучения к области алгоритмической торговли. Область алгоритмической торговли является высокотехнологичной быстроразвивающейся благодаря заинтересованности крупных участников в постоянном поиске более эффективных алгоритмов алгоритмической торговли и улучшении существующих решений.

В работе рассматривается алгоритмическая торговля с точки зрения моделирования биржевого агента: некоторого алгоритма, который совершает действия на бирже (покупка и продажа финансовых инструментов ) и целью которого является получение прибыли в результате совершения операций. Объектом исследования является биржа, предметом исследования (то есть свойствами рассматриваемой системы, которые представляют интерес в работе) - скрытые закономерности рынка, используя их, агент может получать прибыль.

Моделируемый в работе агент использует алгоритмы машинного обучения для совершения действий. Рассматриваемыми в работе алгоритмами машинного обучения являются генетические алгоритмы, деревья решений, ансамбли деревьев решений. Генетические алгоритмы применяются для построения деревьев решений, и для построения ансамблей решений. Деревья решений используются как классификаторы, по набору входных признаков они предсказывают действие, которое необходимо совершить агенту в каждый момент времени. Входными признаками являются рассчитываемые по данным рынка технические индикаторы. На обучающей выборке (некоторый период времени) проводится процесс обучения, который заключается в построении генетическим алгоритмом деревьев решений, показывающих наилучший результат за этот период. Затем проводится проверка качества работы

полученного агента на тестовой выборке: симуляция его работы на некотором периоде времени, следующем за обучающим. Результаты агента на тестовой выборке сравниваются с торговыми бенчмарк-стратегиями агентов, распространёнными в области. Все эксперименты проводятся на реальных исторических биржевых данных, которые получены из открытых источников.

Традиционным подходом в алгоритмической торговле с использованием машинного обучения является обучение с учителем, в данной работе используется обучение с подкреплением. В работах в этой области, рассматривающих подход применения генетических алгоритмов для создания деревьев решений строятся сложные деревья с различными операторами в вершинах, что приводит к переобучению. Идеей, предложенной в работе, является создание простых неглубоких деревьев решений и объединение их в ансамбли, что позволяет снизить переобучение и, соответственно, агент способен показывать стабильно высокие результаты на тестовой выборке. Практической ценностью работы является возможность создания автоматизированных торговых систем, основанных на алгоритмах, рассматриваемых в работе.

В теоретической части работы содержатся сведения об алгоритмах, применяемых в исследовании: деревьях решений, ансамблях деревьев решений, генетических алгоритмах в обобщённой формулировке, генетических алгоритмах для построения деревьев решений. Затем в работе содержится описание модели: описываются рассматриваемые технические индикаторы, которые используются как входные признаки для деревьев решений. В практической части работы приводятся описания данных каждого из проведённых экспериментов, их результаты с приведением различных построенных графиков для более наглядной демонстрации работы алгоритмов, а также сравнение результатов работы алгоритмов в разных экспериментах и выводы по ним. В заключении работы содержится перечисление решённых в

ней задач и выводы о возможности успешного применения рассмотренных алгоритмов к задачам алгоритмической торговли.

## **2. Обзорно-постановочная часть**

### **2.1 Предметная область**

Алгоритмическая торговля (трейдинг) является высокотехнологичной областью, находящейся на стыке компьютерных наук (таких направлений, как анализ данных, машинное обучение, программная инженерия), эконометрики и математики (статистики, методов оптимизации и др.). Благодаря заинтересованности крупнейших финансовых компаний и хедж-фондов в постоянном совершенствовании собственных торговых систем и алгоритмов, область непрерывно развивается. Кроме того, стремительно развивается такая область алгоритмического трейдинга, как высокочастотная торговля (High Frequency Trading), которую отличает стремление участников торгов зарабатывать на микродвижениях рынка, соответственно, в этой области алгоритмической торговли все торговые системы основаны на высокоэффективных алгоритмах, в том числе и алгоритмах машинного обучения.

Особенностью области алгоритмической торговли является закрытость информации, поскольку компании, занимающиеся алгоритмическим трейдингом, являются конкурирующими друг с другом, из-за чего публикуемые в открытых источниках исследования дают неполную картину всех исследований, проводимых в области. Несмотря на это, крупнейшие участники рынка являются спонсорами таких известных конференций по машинному обучению, как, например, «Conference on Neural Information Processing Systems» («NIPS»), что подтверждает их заинтересованность в научных работах, проводимых в области.

## 2.2 Объект и предмет исследования

Системами, рассматриваемыми в работе являются финансовые рынки. Биржа представляет собой систему, в которой участники рынка отправляют заявки (ордера) на покупку или продажу на разные ценовые уровни. Простейшая заявка состоит из желаемого действия: покупка (bid), продажа (ask); желаемой цены покупки или продажи; желаемого количества единиц финансового инструмента для покупки или продажи. Также в заявке могут присутствовать дополнительные атрибуты в зависимости от особенности конкретной биржи, например, сведения о времени отзыва заявки в случае неисполнения и др. Таким образом участники рынка могут дожидаться пока их заявка будет удовлетворена или же совершать действие покупки или продажи немедленно, удовлетворяя заявки других участников. В данной работе рассматриваются только действия агента по удовлетворению заявок других участников, то есть действия немедленной покупки или продажи.

Существует несколько уровней подробности рассмотрения биржи, наиболее подробным является биржевой стакан - структура, описывающая выставленные участниками рынка ордера. Более общим способом описания рынков являются рыночные свечи: при этом методе описания время торгов разбивается на отдельные периоды, в базовом варианте каждый из периодов характеризуется четвёркой чисел: цена актива в начале периода (open), цена актива в конце периода (close), минимальная цена за этот период (low, максимальная цена период (high). В более расширенном варианте к этим данным добавляется также величины объёма сделок по покупке и продаже инструмента за данный период. Название свеча объясняется графическим представлением данного способа описания рынка. В данной работе рассматривается описательный подход с помощью расширенного варианта свечей (каждый период времени характеризуется пятью величинами), поскольку алгоритмам, применяемым в работе, достаточен уровень описания рынка, который обеспечивают рыночные

свечи. Для численного описания рынка в каждый момент времени используется набор технических индикаторов — рассчитываемых на основе данных свечей. В работе технические индикаторы выполняют функцию входных признаков для алгоритмов машинного обучения.

Предметом исследования являются скрытые закономерности в работе рынков, выявление паттернов в событиях на финансовых рынках позволяют получать прибыль от совершения торговых операций. При этом в работе используется метод, который не требует в явном виде описания существующих или предполагаемых закономерностей и не выявляет их в явном виде, однако он использует их неявно, что демонстрируется успешной работой алгоритма.

## **2.3 Обзор работ по теме исследования**

Наиболее традиционным способом применения машинного обучения к задаче алгоритмической торговли является обучение с учителем (supervised learning). В качестве целевой переменной, то есть переменной, значение которой предсказывается алгоритмом, выступает изменение цены актива за некоторый период времени [1] или, альтернативно, волатильность рынка [2].

В начале исследований по применению машинного обучения к прогнозированию событий на финансовых биржах наиболее часто используемыми алгоритмами были более простые и хорошо изученные алгоритмы: деревья решений и машины опорных векторов. При таких методах входными признаками алгоритмов машинного обучения обычно являются технические индикаторы рынка [3]. С началом успешного применения нейронных сетей к другим задачам, таким, как задачи обработки естественного языка (natural language processing), нейронные сети стали применяться и в области алгоритмической торговли, например, для предсказания движений цен на рынке под влиянием появляющихся политических или экономических новостей [4], то есть появилась возможность извлекать сигналы рынка из

неструктурированной информации.

Обучение с подкреплением (reinforcement learning) в последние годы является одним из наиболее быстро развивающихся разделов машинного обучения, с помощью этого метода достигнуты такие результаты, как, например, способность компьютера играть в различные игры на симуляторе Atari на уровне, превосходящем уровень человека, ничего изначально не зная о правилах игры [5]. Применение обучения с подкреплением к задачам алгоритмической торговли имело место и ранее [6], но в последнее время в связи с успехами, достигнутыми с помощью обучения с подкреплением в других предметных областях, возрос интерес к исследованиям возможности применения этого класса алгоритмов к задачам алгоритмической торговли [7] .

Генетические алгоритмы являются общим классом алгоритмов, способным решать задачи в разных областях, так, например, они успешно применяются в робототехнике [8] . Существуют и применения генетических алгоритмов к обобщённым задачам, например таким, как построение ансамблей алгоритмов машинного обучения для повышения точности работы отдельных алгоритмов [9] и построение деревьев решений [10], в таких задачах генетические алгоритмы также показывают свою эффективность. В задачах алгоритмической торговли изначально они применялись в основном для генерации так называемых торговых правил [11]. Этот метод заключается в том, что с помощью генетических алгоритмов создаются функции от технических индикаторов рынка, которые наиболее точно способны предсказывать изменение цены актива на некотором промежутке времени. Затем генетические алгоритмы стали применяться на практике для построения автоматизированных торговых систем [12].

## **2.4 Цель и задачи работы**

Целью данной работы является применение к области алгоритмического



трейдинга подхода, основанного на комбинации генетических алгоритмов и деревьев решений. Более подробно цель работы может быть сформулирована так: моделирование работы торгового агента (агент работает без вмешательства человека) и создание алгоритма на основе методов машинного обучения, который позволяет агенту показывать стабильно высокие результаты торговли на разных периодах времени. В данном случае под высокими результатами понимается сравнение результатов алгоритма с максимальным результатом, полученным при многократном запуске симуляции торговли, в которой агент выбирает действие случайно и равновероятно (симуляция Монте-Карло [13]), а также с традиционной бенчмарк-стратегией «buy-and-hold» [14] (покупка инструмента в начале периода, продажа в конце).

Допустимыми действиями агента являются покупка единицы некоторого финансового инструмента, продажа единицу или отсутствие одного из этих действий. Целью работы агента является получение максимальной прибыли за некоторый заранее фиксированный период времени. Данная задача может рассматриваться как задача обучения с подкреплением: в течение некоторого периода времени алгоритм совершает действия в зависимости от состояния среды в которой он находится (среда — это финансовый рынок), агент узнает о результате своей работы — выраженные в условных единицах прибыль или убыток от совершения торговых операций. Для алгоритмов, применяемых в работе, достаточно узнавать результат торговых операций только в конце периода, в этом состоит разница с традиционной формулировкой задачи обучения с подкреплением, в которой агент получает вознаграждение по мере выполнения своих действий.

Задачами, решаемыми в данной работе являются:

- Знакомство с предметной областью алгоритмической торговли
- Исследование алгоритмов машинного обучения, таких как: деревья решений, генетические алгоритмы, их комбинация, ансамбли

деревьев решений, обучение с подкреплением

- Рассмотрение задачи алгоритмической торговли в терминах задачи обучения с подкреплением и моделирование работы автоматизированного торгового агента
- Исследование способов применения рассмотренных алгоритмов к задаче алгоритмической торговли
- Проведение экспериментов на реальных исторических биржевых данных, сравнение результатов алгоритмов на различных данных и выводы по ним

## **2.5 Используемые методы**

Работа торгового агента моделируется следующим образом: в течение некоторого заранее определённого периода времени агент на основании состояния финансового рынка совершает действия (в качестве действий агента рассматриваются покупка, продажа или отсутствие действия с финансовым инструментом) и узнаёт результат своих действий (результат торговых операций, он может быть положительным, отрицательным или нулевым). Цель агента состоит в максимизации своей прибыли. Используемые в работе алгоритмы не используют промежуточные результаты, полученные агентом: в них учитывается только финальный результат работы агента за период времени. Такая модель может быть отнесена к модели обучения с подкреплением: так же, как в классической задаче обучения с подкреплением алгоритмом решается задача поиска оптимальной стратегии (в данном случае стратегией является дерево решений, которое на основании данных рынка определяет действие агента) для максимизации получаемого результата.

С помощью первого рассматриваемого в работе алгоритма эволюционным методом строятся деревья решений: выбираются те деревья, которые дают наилучший результат на обучающей выборке, то есть генетическим алгоритмом

решается задача максимизации прибыли, получаемой деревом решений на обучающей выборке. В качестве входных признаков для деревьев решений используются технические индикаторы, рассчитанные на основе данных рыночных свечей.

Для улучшения результатов работы первого алгоритма, применяется второй алгоритм: с его помощью также генетическим образом деревья объединяются в ансамбли, в данном случае генетически подбираются веса деревьев, с которыми они входят в ансамбль так, чтобы ансамбль показывал наилучший результат на обучающей выборке. Второй метод позволяет повысить обобщающую способность алгоритма, то есть снизить переобучение и повысить результат на тестовой выборке.

В качестве обучающих и тестовых выборок рассматриваются реальные исторические биржевые данные из открытых источников. В проводимых экспериментах используются данные по разным финансовым инструментам и разным периодам времени.

## **2.6 Актуальность работы**

Алгоритмическая торговля в последние годы активно развивается благодаря совокупности факторов: стремительному развитию методов машинного обучения, развитию технологий работы с данными и их анализа, росту возможностей хранения и обработки большего объёма данных. Кроме того, сложность применяемых участниками рынка алгоритмов торговых систем растёт, поскольку участники конкурируют не только с участниками, не использующими автоматизированные системы, но и друг с другом. В связи с этими тенденциями, исследование возможности применимости нестандартных (то есть ещё не до конца изученных) алгоритмов машинного обучения к задачам алгоритмической торговли является актуальной задачей. В подобных исследованиях есть интерес не только со стороны компаний, занимающихся

алгоритмической торговли, таких как хедж-фонды, но и со стороны научного сообщества, поскольку развитие направления машинного обучения в этой области способно привнести новые ценные знания в развитие машинного обучения, как раздела компьютерных наук, которые могут быть применены к другим предметным областям.

## **2.7 Новизна и практическая ценность работы**

В данной работе применяется нестандартный подход к применению машинного обучения к области финансовых рынков: традиционным подходом является обучение с учителем, в данной работе рассматривается модель обучения, которая может быть с некоторыми ограничениями отнесена к модели обучения с подкреплением. Также нестандартным является подход к построению деревьев решений с помощью генетического алгоритма. Кроме того, в тех работах, в которых применяется аналогичный подход ([11] [15] [16]), рассматриваются сложные деревья: в их вершинах могут находиться логические операторы и даже вещественные функции. В данной же работе намеренно рассматриваются простейшая версия деревьев решений (бинарные деревья, в вершинах могут быть только операторы сравнения, глубина деревьев ограничивается) для того, чтобы снизить переобучение и повысить их обобщающую способность (при использовании сложных деревьев переобучение является серьёзной проблемой [11]). Использование самой базовой разновидности деревьев (с использованием технических индикаторов как признаков) позволяет достигать на тестовых выборках более высоких результатов, чем сложные модификации (благодаря существенно более слабому переобучению), и может считаться новой идеей, предложенной в работе.

Рассмотренный в работе подход к задаче алгоритмической торговли как к частному случаю задачи обучения с подкреплением (в котором вознаграждение за действия сообщается агенту в конце периода его работы) и, более конкретно,

подход к построению деревьев решений, является обобщённым и не использует особенности предметной области. Использование в качестве входных признаков технических индикаторов рынка не является фундаментальным в данном подходе, в других предметных областях входными признаками могут служить любые признаки, описывающие состояние среды, в которой находится агент. Демонстрация возможности успешного применения к рассматриваемой задаче такого общего подхода является теоретической ценностью работы, поскольку в настоящее время активно ведутся исследования о возможности применения к данной задаче алгоритмов, не использующих обучение с учителем и особенности предметной области, ранее практически все работы по применению машинного обучения к области алгоритмической торговле были узко направленными именно на эту область. Кроме того, подобный метод, может быть перенесён на другие предметные области, задачи в которых могут быть рассмотрены как задачи поиска стратегии поведения агента, максимизирующей его результат работы. В качестве практической ценности работы можно рассматривать возможность создания автоматизированных торговых систем на основании алгоритмов, исследуемых в работе.

### **3. Теоретическая часть**

В теоретической части приводятся сведения об основных используемых в работе алгоритмах машинного обучения.

#### **3.1 Обучение с подкреплением**

Алгоритмы обучения с подкреплением составляют отдельный обширный класс алгоритмов машинного обучения (наряду с такими классами как обучение с учителем и обучение без учителя). Задачи, решаемые с помощью методов обучения с подкреплением формулируются в общем виде следующим образом: некоторый агент находится в среде, каким-либо образом формально описываемой (например, вектором действительных чисел), и имеет

возможность совершать различные действия. В зависимости от совершенных действий и состояния среды, агент получает некоторое вознаграждение, которое может быть как положительным, так и отрицательным. Задача обучения агента состоит в том, чтобы совершать действия так, чтобы максимизировать полученную награду.

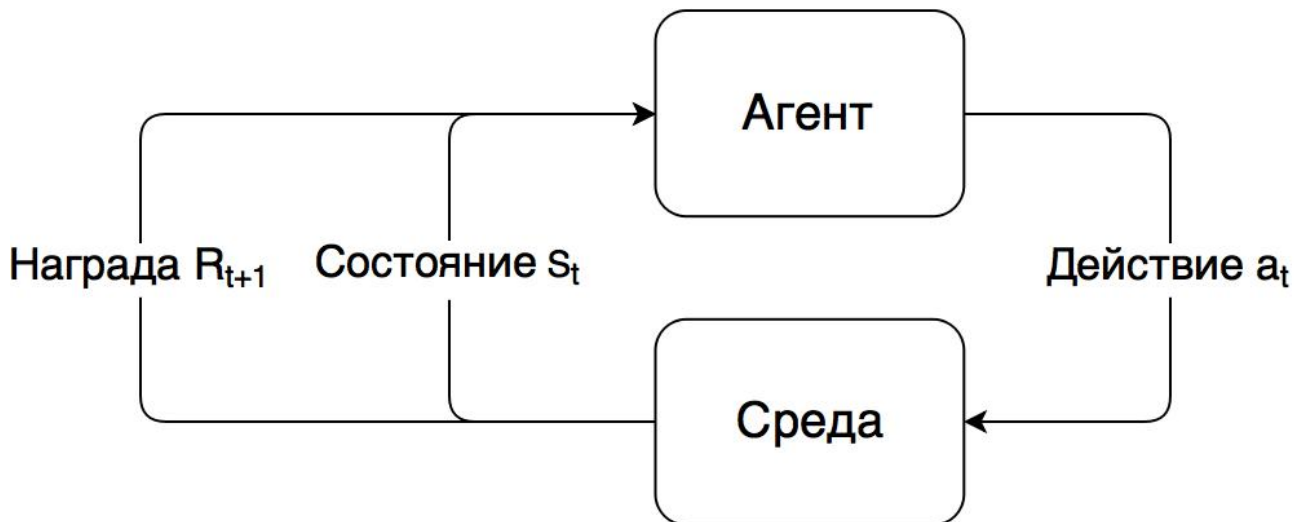


Иллюстрация 3.1: Схема модели обучения с подкреплением: в момент времени  $t$  агент наблюдает состояние среды, совершает действие и получает награду в следующий момент времени

В данной работе особенностью является то, что агент узнаёт о своей награде не сразу после совершения действий, а только в конце периода своей работы, однако в остальном постановка задачи соответствует традиционной постановке задачи обучения с подкреплением: состоянием среды является набор рассчитанных по состоянию рынка технических индикаторов, действиями агента являются действия покупки, продажи или ни одно из них.

### 3.2 Деревья решений

Деревья решений являются одним из классических алгоритмов машинного обучения, существует большое количество их разновидностей [17], они могут применяться как для задач классификации, так и для задач регрессии, однако в

данной работе деревья решений будут рассматриваться только как классификаторы, причём для численных признаков.

Дерево решений представляет собой разновидность дерева как объекта из теории графов: это связный граф без петель и кратных рёбер. Вершины, являющиеся листовыми и не являющиеся таковыми, выполняют различные функции. Так, в вершинах — листах записаны метки, которые означают один из возможных классов, на которые ведётся классификация.

Будут рассматриваться так называемые «Axis-Parallel Decision Trees» [10], деревья решений, в вершинах которых допускаются только простые условия вида  $feature < operator > threshold$ , где  $operator$  это один из операторов сравнения, который применяется к сравнению некоторого признака  $feature$  с некоторым значением  $threshold$ .

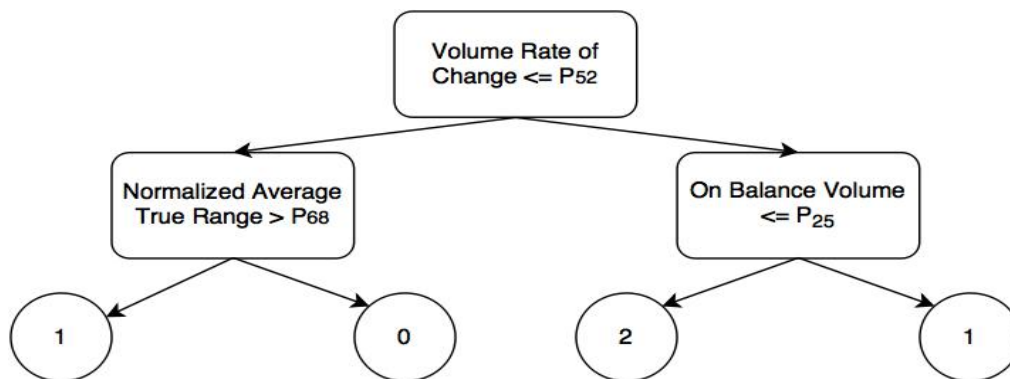
Характеристики деревьев, используемых в работе:

1. Глубина деревьев строго задаётся.
2. Деревья бинарные и полные. Это означает что количество вершин в дереве равно  $2^D - 1$ , где  $D$  — глубина дерева.
3. В каждой нетерминальной вершине хранится  $(indicator, percentile, sign)$  тройка:  $indicator$  — номер технического индикатора;  
 $percentile$  — значение перцентиля;  
 $sign$  — знак, записываемый бинарно: 0 или 1, 0 означает знак «меньше или равно», 1 означает знак «больше».

Например, тройка  $(PVT, 72, 0)$  означает следующее: при попадании в данную вершину значение индикатора «Price-Volume Trend» сравнивается с перцентилем 72 этого индикатора для обучающей выборке и если оно не больше, то переход осуществляется по левому ребру, выходящему из вершины, иначе — по правому.

4. В терминальных вершинах записаны метки, характеризующие действие агента: 0 — ничего не делать, 1 — покупка единицы инструмента, 2 —

продажа.



Пример 3.1: Дерево решений глубины 3, классифицирующее на 3 класса

### 3.3 Генетические алгоритмы

Поскольку идея генетических алгоритмов заимствована из процесса эволюции, происходящего в природе, в них используются термины, используемые и в описании процессе эволюции с точки зрения биологии. Основные понятия, используемые в генетических алгоритмах:

- *Объект (индивидуум)* — сущность, рассматриваемая в задаче. Например, в качестве сущностей могут выступать вектора вещественных чисел:  
$$x = (x_1, \dots, x_N), \quad x_i \in \mathbb{R} \text{ для } i=1, \dots, N \quad (x \in \mathbb{R}^n)$$
- *Популяция* — мультимножество объектов (индивидуумов).
- *Фитнес-функция* — некоторая функция от объекта, значение которой необходимо максимизировать или минимизировать алгоритмом. Для использования функции как фитнес-функции необходимо задание нестрогого полного порядка на множестве её значений. Примеры фитнес-функций для векторов  $x \in \mathbb{R}^n$  :



$$1. f_1: \mathbb{R}^N \rightarrow \mathbb{R}, f_1(x) = \sum_{i=1}^N \sin(x_i)$$

$$2. f_2: \mathbb{R}^N \rightarrow \mathbb{N}, f_2(x) = \sum_{i=1}^N \text{sign}(x_i)$$

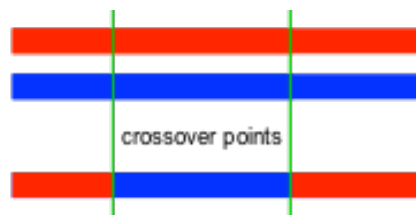
- *Кроссовер (скрещивание)* — операция получения нового объекта из двух родительских. В зависимости от задачи она может определяться по-разному, для векторов  $x \in \mathbb{R}^n$  она определяется следующим образом:

$$x, y \in \mathbb{R}^N$$

Случайно выбираются точки разделения  $i_1 < \dots < i_K, 1 \leq i_j < N$  для  $j \in \overline{1, K}$

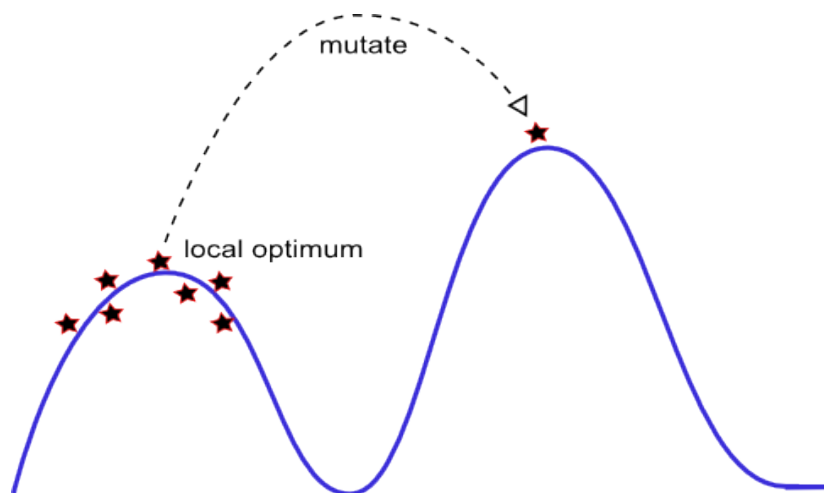
$$I = \{0, i_1, \dots, i_K, N\}$$

$$\text{Новый объект } Z = (z_1, \dots, z_N), z_i = \begin{cases} x_i & \text{if } i_k < i \leq i_{k+1}, k \bmod 2 \equiv 0 \\ y_i & \text{if } i_k < i \leq i_{k+1}, k \bmod 2 \equiv 1 \end{cases}$$



Пример 2.2: Кроссовер с 2 разделяющими точками (отмечены зелёными линиями)

- *Мутация* — операция изменения нового объекта. Практически для любых типов объектов мутация происходит одинаковым образом: в новом созданном (в результате кроссовера) объекте случайным образом выбираются части для изменения и изменяются случайным образом. Для векторов  $x \in \mathbb{R}^n$  это означает случайный выбор некоторого количества координат и изменение их на сгенерированные случайные числа. Интуитивный смысл мутации состоит в том, чтобы при попадании фитнес-функции объектов популяции в локальный экстремум, выйти из него.



*Иллюстрация 3.2: Демонстрация смысла операции мутации: при примерном равенстве значений фитнес-функций популяции локальному экстремуму возможность выйти из него*

Псевдокод обобщённого генетического алгоритма:

### *Алгоритм 3.1: Обобщённый генетический алгоритм*

#### *Входные данные:*

Размер поколения  $N$ ;

Значение  $T$  – количество наилучших объектов, переходящих в следующее поколение;

В случае, если условием остановки является созданное количество поколений, количество поколений  $R$ .

#### *Выходные данные:*

Объект с максимальным значением фитнес-функции.

- (1) Случайная инициализация начального поколения
- (2) Пока не достигнуто условие остановки:
- (3)       Отсортировать объекты по убыванию значения фитнес-функции
- (4)       Сохранить в популяции первые  $T$  объектов
- (5)       Создать новые объекты:
- (6)             Выбрать случайно 2 объекта из оставшихся в популяции
- (7)             С помощью операции кроссовер создать новый объект
- (8)             С помощью операции мутации преобразовать созданный объект
- (9)             Добавить новый объект в поколение
- (10)       Вычислить значения фитнес-функций новых объектов
- (11) После условия остановки:
- (12)       Выбрать объект с максимальным значением фитнес-функции как ответ алгоритма

### 3.4 Построение деревьев решений с помощью генетических алгоритмов

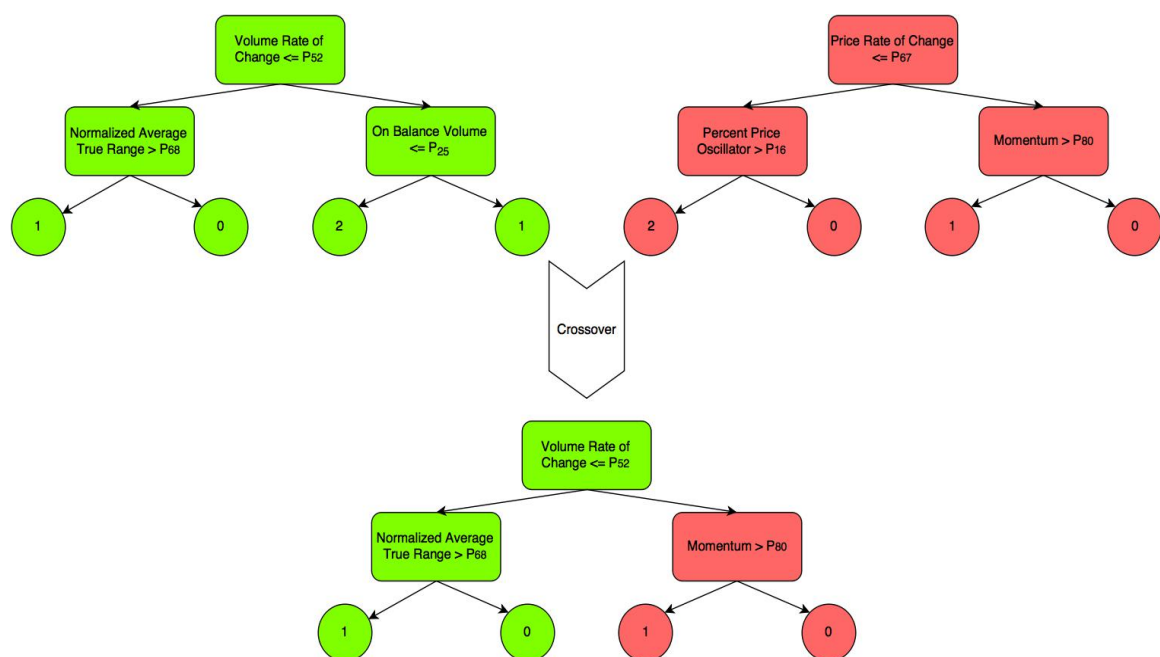
Для использования деревьев решений с помощью генетических алгоритмов необходимо определить операции кроссовера и мутации для деревьев, а также фитнес-функцию.

#### 1. Кроссовер

Операция кроссовер может быть реализована двумя способами: с использованием структуры дерева и без её использования. В рамках построения деревьев решений в рассматриваемой задаче рассматривается только применение кроссовера к двум деревьям, которые полностью идентичны по структуре вершин: они отличаются только значениями в вершинах, а их глубина и расположение вершин одинаковы.

##### 1.1 С использованием структуры дерева:

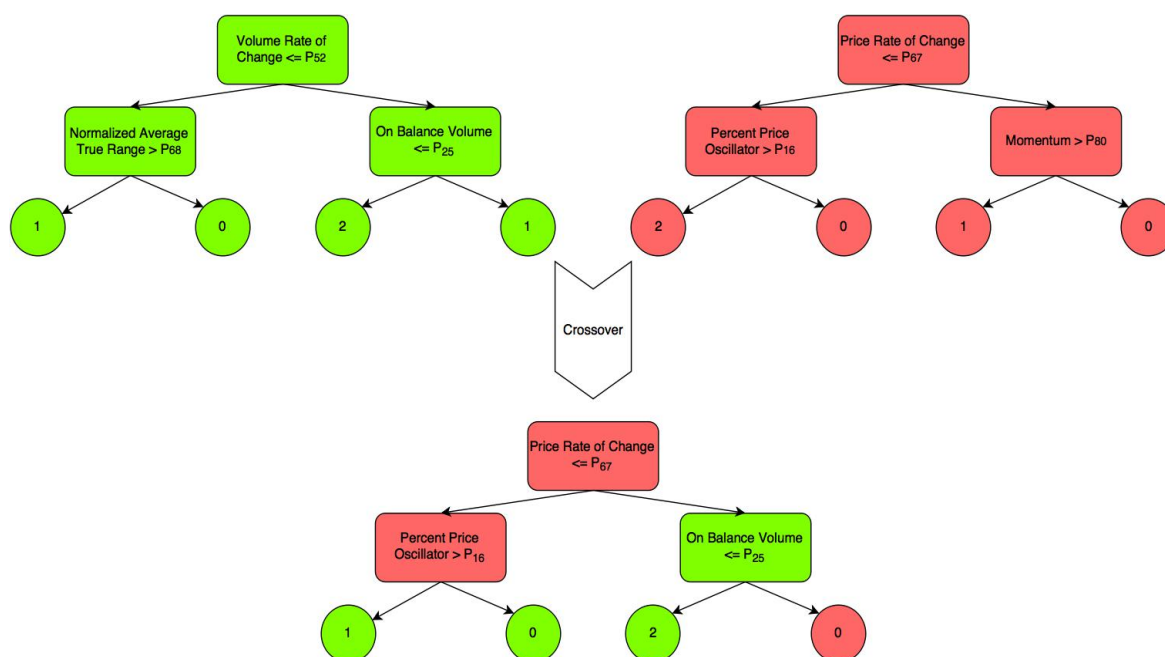
Выбирается случайным образом некоторая вершина  $v$  в дереве  $A$ , новое дерево получается заменой поддерева дерева  $A$  с корнем в  $v$  поддеревом дерева  $B$  с корнем в вершине, соответствующей  $v$ .



Пример 2.3: Кроссовер двух деревьев с использованием их структуры: в качестве вершины разделения выбрана правая вершина на глубине 2

## 1.2 Без использования структуры дерева:

В этом случае новое дерево получается заменой случайно выбранных вершин дерева  $A$  на соответствующие вершины дерева  $B$ .



Пример 2.4: Кроссовер двух деревьев без учёта их структуры: некоторые вершины первого дерева заменены на соответствующие вершины второго

## 2. Мутация

Рассматривается только простая реализация операции мутации: в случайно выбранных вершинах дерева значения заменяются на новые случайно сгенерированные. Таким образом в нетерминальных вершинах заменяются: технический индикатор, величина перцентиля, знак сравнения. В терминальных вершинах заменяются метки классов.

В реализованном в работе алгоритме в процессе мутации могут заменяться не все значения в вершине, а, например, только значение перцентиля, то есть каждое из значений в выбранной для мутации вершине может быть с некоторой вероятностью заменено. Причём технический индикатор, знак сравнения и метки классов заменяются на случайные, а перцентиль сдвигается на случайную целую величину, не превосходящую максимальный размер сдвига, который является параметром алгоритма.

*Алгоритм 2.2: Алгоритм мутации при использовании генетического алгоритма для построения деревьев решений*

*Входные данные:*

Дерево  $A$ ;

Вероятность изменения метки в терминальной вершине:

$p_1$

Вероятность изменения признака в вершине:  $p_2$

Вероятность изменения знака сравнения в вершине:  $p_3$

Максимальный размер смещения перцентили:  $b$

*Выходные данные:*

Дерево  $A_{new}$ .

(1) Если вершина терминальная:

(2)       Сгенерировать  $\alpha \sim R(0,1)$

(3)       Если  $\alpha < p_1$ :

(4)       Заменить метки классов в вершине на  
             случайные

(5) Иначе:

(6)       Сгенерировать  $\beta \sim R(0,1)$

(7)       Если  $\beta < p_2$ :

(8)       Заменить признак в вершине на случайный

(9)       Иначе:

(10)       Сгенерировать  $\gamma \sim R(0,1)$

(11)       Если  $\gamma < p_3$ :

(12)       Изменить знак сравнения в вершине на  
             противоположный (  $\leq$  на  $>$  ,  $>$  на  $\leq$  )

(13)       Сгенерировать  $\epsilon \sim R(0,b)$

(14)       Заменить перцентиль:

(15)       percentile  $\leftarrow \lfloor \epsilon \rfloor$

(16)       percentile  $\leftarrow \min(\max(\text{percentile}, 0), 100)$

### 3. Фитнес-функция

При построении деревьев решений с помощью генетического алгоритма в качестве фитнес-функции выступает некоторая метрика качества построенного дерева. В задачах обучения с учителем в качестве фитнес-функций могут выступать стандартные метрики качества классификации: точность (Accuracy), F-мера, площадь под кривой (ROC — Curve) и др. В рассматриваемой в работе задаче обучения с подкреплением метрикой качества построенного дерева является его результат на обучающей выборке, то есть сколько смог бы заработать торговый агент (в условных единицах) за период времени обучающей выборки, принимая решения о действии, которое необходимо совершить в каждый момент времени в соответствии с предсказаниями дерева решений.

#### 3.5 Ансамбли деревьев решений

В общем случае, *ансамбль*— это мета-алгоритм машинного обучения, который позволяет путём объединения простых классификаторов или регрессоров получать более высокую точность, чем входящие в ансамбль алгоритмы по отдельности [9]. Поскольку в данной работе рассматриваются деревья решений для классификации, то ансамбли будут так же рассматриваться для задачи классификации, более детально, рассматривается модель ансамбля с взвешенным голосованием: ансамбль из  $N$  классификаторов задаётся вектором весов  $(w_1, \dots, w_N)$ .

В случае алгоритмов для классификации с предсказанием ансамбля путём взвешенного голосования предсказание строится следующим образом:



Ансамбль с весами  $(w_1, \dots, w_N)$ ; классы:  $1, \dots, K$ ;

Предсказания классификаторов:  $p_1, \dots, p_N$

Количества голосов классов:  $g_1, \dots, g_K$ ,  $g_j = \sum_{i=1}^N w_i I(i)$ ,  $I(i) = \begin{cases} 1, & p_i = j \\ -1, & p_i \neq j \end{cases}$

Предсказание ансамбля  $P_{ensemble} = \underset{j=\overline{1, m}}{\operatorname{argmax}} g_j$

### 3.6 Построение ансамблей деревьев решений с помощью генетических алгоритмов

При построении ансамблей с помощью генетических алгоритмов ансамбль рассматривается как вектор весов, таким образом становится возможным применять операции кроссовера и мутации, определённые выше для векторов  $x \in \mathbb{R}^n$ . В качестве фитнес-функций используются функции, аналогичные функциями, используемым при построении деревьев решений: в случае задачи обучения с учителем это метрики качества классификации; в случае задачи обучения с подкреплением в качестве фитнес-функции выступает результат ансамбля на обучающей выборке: сколько смог бы заработать торговый агент (в условных единицах) за период времени обучающей выборки, принимая решения о действиях в соответствии с предсказаниями ансамбля.

3.7

## 4. Модель биржевого агента

### 4.1 Симуляция агента

Перед описанием процесса симуляции торгового агента введём дополнительные обозначения.

*Позиция* — количество единиц инструмента, которым владеет агент. При этом

возможна отрицательная позиция — ситуация, при которой агент продаёт инструмент, не имея его в наличии, то есть, фактически, это означает, что агент обязуется выкупить его в конце торговой сессии.

*Результат агента* — его заработок или убыток, полученный после проведения торговых операций.

*Количество сделок* — количество покупок или продажи инструмента.

Результат считается за некоторый период, который разбит на промежутки времени — биржевые свечи. Результат агента за некоторый период времени является суммой изменений результатов между каждым из промежутков и предыдущим перед ним. Изменение результата торгового агента между двумя промежутками времени является заработком на изменении цены инструмента между этими двумя промежутками. Результат агента определяется таким образом, чтобы стимулировать его совершать большое количество сделок и зарабатывать на изменении цены на коротких промежутках времени вместо того, чтобы пытаться заработать на изменении цены на длительных периодах времени: агенту выгодно многократно поменять знак своей позиции, чтобы заработать на многократных движениях цены в обе стороны. Подобные методы используются в области алгоритмической торговли, которая называется высокочастотной (High Frequency Trading или HFT).

Такое определение результата следует из предположения, что агент может сделать в текущий промежуток времени действие, противоположное действию в предыдущий (покупка, если была продажа; продажа, если была покупка) и заработать на изменении цены. Например, в некоторый промежуток времени средняя цена свечи (среднее арифметическое наибольшей и наименьшей цен свечи) равна 8124 у. е., в следующий — 8129 у. е., позиция агента равна -3. К следующему периоду результат агента увеличился на  $5 \cdot (-3) = -15$  у. е. . Если позиция агента была равна 3, то его результат увеличился бы на  $5 \cdot 3 = 15$  у. е. .

Более формально, результат агента рассчитывается следующим образом:

Обозначения :

$T$  – количество периодов времени ( количество свечей)

$r_t$  – прибыль или убыток агента за период  $t$

$pos_t$  – позиция агента в период  $t$

$price_t$  – среднее цены в свече в период  $t$

$$\begin{aligned} \text{Результат агента: } Result_{total} &= \sum_{t=2}^T r_t = \sum_{t=2}^T pos_{t-1} \Delta price_t = \\ &= \sum_{t=2}^T pos_{t-1} (price_t - price_{t-1}) = \\ &= \sum_{t=2}^T pos_{t-1} \left( \frac{low_t + high_t}{2} - \frac{low_{t-1} + high_{t-1}}{2} \right) \end{aligned}$$

## 4.2 Ограничения модели

В данной модели сделаны следующие предположения:

Агент может покупать и продавать инструмент по средней цене свечи.

Данное предположение не является серьёзным ограничением для ликвидных инструментов на современных биржах, но может быть достаточно большим допущением на менее ликвидных инструментах или менее популярных биржах.

Действие агента происходит мгновенно.

В реальности существует некоторая задержка (round-trip) между принятием агентом решением о покупке или продаже и исполнением его заявки на бирже. Однако поскольку рассматриваемые периоды являются значительными (минимальный из них равен 10 минут), при том, что задержка на современных биржах составляет микросекунды, данной задержкой можно пренебречь.

Комиссия за совершение операций составляет 0.035% от каждой сделки. Размер всех сделок составляет 1 единицу финансового инструмента.

В соответствии с реальными условиями торгов на биржах комиссия составляет

обычно некоторую долю от размера сделки. В данном случае комиссия принимается равной 0.0035%, поскольку это реальное значение комиссии на фондовой бирже ММВБ по состоянию на апрель 2016 года [18]

Позиция агента равняется 0 на начало периода торговли, но не обязательно равняется 0 по его окончанию. Максимальная позиция агента ограничивается по модулю константой для того, чтобы снизить риск большой потери результата при резком изменении цены инструмента.

### 4.3 Технические индикаторы

Состояние рынка в каждый момент времени описывается техническими индикаторами — специальными величинами, рассчитанными по данным рыночных свечей, помогающими прогнозировать изменения на рынке.

Основные величины:

- $close_t, open_t$  — цены открытия и закрытия свечи в период времени  $t$  соответственно
- $low_t, high_t$  — минимальные и максимальные цены свечи в период времени  $t$  соответственно
- $volume_t$  — объём сделок по покупке и продаже финансового инструмента в период времени  $t$  соответственно

Вспомогательные величины:

Скользящее среднее за  $N$  периодов (Simple Moving Average):

$$SMA_t(N) = \frac{\sum_{p=t-N}^t close_p}{N}$$

Стандартное отклонение за  $N$  периодов (Standard Deviation):

$$StdDev_t(N) = \sqrt{\frac{\sum_{p=t-N}^t (close_p - SMA_t(N))^2}{N}}$$

$$HighestHigh_t(N) = \max_{t-N \leq p \leq t} high_p; \quad LowestLow_t(N) = \min_{t-N \leq p \leq t} low_p$$

$$TrueRange_t = \max\{high_t - low_t, high_t - close_{t-1}, close_{t-1} - low_t\}$$

Индикаторы можно разделить на 2 группы: осцилляторы и индикаторы тенденций. Осцилляторы позволяют предсказывать развороты трендов (смену текущего тренда движения цены на противоположный), в то время как индикаторы тенденций позволяют предсказывать продолжение текущего тренда.

## Индикаторы тенденций

Price Volume Trend (PVT)

$$PVT_t = PVT_{t-1} + \frac{close_t - close_{t-1}}{close_{t-1}} volume_t$$

Accumulation / Distribution (AD)

$$ADL_t = \frac{(close_t - low_t) - (high_t - close_t)}{high_t - low_t} volume_t$$

Moving Average Convergence Divergence (MACD)

$$MACD_t = fastMA_t - slowMA_t;$$

$$slowMA = SMA_t(N), \quad fastMA = SMA_t(M), \quad N > M$$

Price Rate of Change (PROC)

$$PROC_t = \frac{close_t - close_{t-N}}{close_{t-N}}$$

Volume Rate of Change (VROC)

$$VROC_t = \frac{volume_t - volume_{t-N}}{volume_{t-N}}$$

Momentum (MOM)

$$MOM_t = close_{t-N} - close_t$$

Price Acceleration Between Times (PABT)

$$PABT_t(N) = MOM_{t-N} - MOM_t$$

On-Balance Volume (OBV)

$$OBV_t = \begin{cases} OBV_{t-1} + volume_t & \text{if } close_t > close_{t-1} \\ OBV_{t-1} - volume_t & \text{if } close_t < close_{t-1} \\ OBV_{t-1} & \text{if } close_t = close_{t-1} \end{cases}$$

Normalized Average True Range (NATR)

$$NATR_t(N) = \frac{\sum_{p=t-N}^t TrueRange_p}{N * close_t}$$

## Осцилляторы

Volatility (Vol)

$$Vol_t(N) = \frac{StdDev_t(N)}{\sqrt{(N)}}$$

Accumulation Distribution Oscillator (ADO)

$$ADL_t = \frac{(high_t - close_t) + (close_t - low_t)}{high_t - low_t} volume_t$$

Percent Price Oscillator (PPO)

$$PPO_t = 100 \frac{fastMA_t - slowMA_t}{slowMA_t};$$

$$slowMA = SMA_t(N), \quad fastMA = SMA_t(M), \quad N > M$$

Fast Stochastics %K (F%K)

$$F\%K_t(N) = 100 \frac{close_t - LowestLow_t(N)}{HighestHigh_t(N) - LowestLow_t(N)}$$

Fast Stochastics %D (F%D)

$$F\%D_t(N) = \frac{\sum_{p=t-N}^t F\%K_t(N)}{N}$$

Slow Stochastics %K (S%K)

$$S\%K_t(M, N) = \frac{\sum_{p=t-M}^t F\%K_t(N)}{M}, \quad M \leq N$$

Slow Stochastics %D (S%D)

$$S\%D_t(M, N) = \frac{\sum_{p=t-M}^t S\%D_t(M, N)}{M}, \quad M \leq N$$

Williams % R (WillR)

$$WillR_t(N) = -100 \frac{HighestHigh_t(N) - close_t}{HighestHigh_t(N) - LowestLow_t(N)}$$

Bollinger Bands — семейство индикаторов для оценки волатильности цены:

Middle Band

$$MBand_t = SMA_t(N)$$

Upper Band

$$UBand_t = MBand_t + 2StdDev_t(N)$$

Lower Band

$$LBand_t = MBand_t - 2 StdDev_t(N)$$

Band Width (Bwidth)

$$BWidth = UBand - LBand$$

Percent B (%B):

$$\%B = \frac{close_t - LBand_t}{UBand_t - LBand_t}$$

Во всех проведённых экспериментах входными признаками деревьев решений и ансамблей деревьев решений являются 17 технических индикаторов, описанных в главе 2 в разделе «Технические индикаторы» с различными параметрами.

Полный список:

1. Price Volume Trend.
2. Accumulation / Distribution.
3. Accumulation Distribution Oscillator.
4. Percent Price Oscillator; N = 10, M = 240.
5. Moving Average Convergence Divergence; N = 10, M = 240.
- 6-8. Price Rate of Change; N = 10, N = 60, N = 240.
- 9-11. Volume Rate of Change; N = 10, N = 60, N = 240.
- 12-14. Momentum; N = 10, N = 60, N = 240.
- 15-17. Price Acceleration Between Times; N = 10, N = 60, N = 240.
- 18-20. On-Balance Volume; N = 10, N = 60, N = 240.
- 21-23. Normalized Average True Range; N = 10, N = 60, N = 240.
- 24-26. Fast Stochastics %K; N = 10, N = 60, N = 240.
- 27-29. Fast Stochastics %D; N = 10, N = 60, N = 240.
- 30-32. Slow Stochastics %K; N = 10, N = 60, N = 240.
- 33-35. Slow Stochastics %D; N = 10, N = 60, N = 240.
- 36-38. Band Width; N = 10, N = 60, N = 240.



39-41. Percent B; N = 10, N = 60, N = 240.

42-44. Williams % R; N = 10, N = 60, N = 240.

45-47. Volatility ; N = 10, N = 60, N = 240.

Таким образом, каждый период времени (рыночная свеча) описывается вектором действительных чисел размерности 47.

#### 4.4 а

## 5. Эксперименты

### 5.1 Описание данных

В экспериментах, проведённых в работе, рассматривается два типа финансовых инструментов: акции крупных компаний и фьючерсы на биржевые индексы. Все данные по рассматриваемым инструментам взяты из открытого доступа: с сайта [finam.ru](http://finam.ru).

|                            | Российские   | США                                      |
|----------------------------|--|--|
| <b>Фьючерсы на индексы</b> | PTC (SPFB.RTS)<br>MMBB (SPFB.MIX)                  | Dow Jones (DSX)<br>NASDAQ (NQ-100-FUT)   |
| <b>Акции</b>               | Сбербанк (SBER)<br>Газпром (GAZP)<br>Лукойл (LKOH) | GE (GE)<br>Apple (AAPL)<br>Yahoo (YAHOO) |

*Таблица 6.1: Рассматриваемые финансовые инструменты. В скобках указаны коды инструментов на сайте [finam.ru](http://finam.ru) по состоянию на апрель 2016 г.*

### 5.2 Параметры алгоритмов

Построение деревьев решений генетическим алгоритмом

- Глубина деревьев 3, 6.

- Размер поколения: 100
- Доля индивидуумов популяции, переходящих в следующее поколение: 0.2
- Количество поколений: 10
- Вероятности мутаций: признака: 0.3, знака оператора сравнения: 0.3, класса в терминальных вершинах: 0.2, максимальный размер сдвига перцентиля: 30%
- Вероятности мутации с каждым поколением увеличиваются (указанные изменения считаются от текущего значения на каждой итерации): признака - на 1%, знака оператора сравнения - на 1%, класса в терминальных вершинах - на 1% максимальный размер сдвига перцентиля - на 1%.

### Построение ансамблей деревьев решений генетическим алгоритмом

- Используются деревья, построенные генетическим алгоритмом (то есть деревья) глубины 3, 6
- Размер поколения: 100
- Доля индивидуумов популяции, переходящих в следующее поколение: 0.2
- Количество поколений: 10
- Вероятность мутации: 0.05
- Вероятности мутации с каждым поколением увеличиваются на 1% от текущего значения
- Допустимые значения весов деревьев: 0-9 (целые числа)

### 5.3 Результаты

|  |         |  |  |  |  |
|--|---------|--|--|--|--|
|  | Глубина |  |  |  |  |
|--|---------|--|--|--|--|

|      |          |  |  |  |  |
|------|----------|--|--|--|--|
|      | деревьев |  |  |  |  |
| LKOH |          |  |  |  |  |
| GAZP |          |  |  |  |  |
| SBER |          |  |  |  |  |
|      |          |  |  |  |  |
|      |          |  |  |  |  |
|      |          |  |  |  |  |
|      |          |  |  |  |  |
|      |          |  |  |  |  |
|      |          |  |  |  |  |
|      |          |  |  |  |  |
|      |          |  |  |  |  |
|      |          |  |  |  |  |
|      |          |  |  |  |  |
|      |          |  |  |  |  |
|      |          |  |  |  |  |
|      |          |  |  |  |  |
|      |          |  |  |  |  |
|      |          |  |  |  |  |
|      |          |  |  |  |  |
|      |          |  |  |  |  |
|      |          |  |  |  |  |
|      |          |  |  |  |  |

### 5.4 Выводы

### 6. Заключение

### Список литературы

1: Li Tun, Liu Gong Shen, Stock Price's Prediction with Decision Tree, 2011

2: R. Seethalakshmi , V. Saavithri, V. Badrinath , C. Vijayabanu, PCA BASED SUPPORT VECTOR MACHINE TECHNIQUE FORVOLATILITY FORECASTING,

3: S. Lahmiri , A Comparison of PNN and SVM for Stock Market TrendPrediction using Economic and Technical Information, 2011

4: X. Ding, Y. Zhang, T. Liu, J. Duan, Deep Learning for Event-Driven Stock Prediction,

5: Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, Martin Riedmiller, Playing Atari with Deep Reinforcement Learning, 2013

6: Moody J.; Saffell M., Reinforcement Learning for Trading,

- 7: Fernandez-Tapia J., High-Frequency Trading meets Reinforcement Learning,
- 8: Floreano, D., S. Nolfi, F. Mondada, Competitive Co-Evolutionary Robotics: From Theory to Practice, 1998
- 9: J. Sylvester, N. V. Chawla, Evolutionary Ensembles: Combining Learning Agents using Genetic Algorithms, 2005
- 10: R. C. Barros, M. P. Basgalupp, A. C. P. L. F. de Carvalho, A. A. Freitas, A Survey of Evolutionary Algorithms for DecisionTree Induction, 2012
- 11: J. - Y. Potvin, P. Soriano, M. Vallée, Generating trading rules on the stock markets withgenetic programming, 2004
- 12: H. Jiang, L. Kang, Building Trade System by Genetic Algorithm,
- 13: J. D. Whiteside, A Practical Applicationof Monte Carlo Simulation in Forecasting, 2008
- 14: E. Tower, O. Gokcekus, Evaluating A Buy and Hold Strategy for the S&P 500 Index, 2001
- 15: F. Allen , R. Karjalainen, Using genetic algorithms to find technical trading rules, 1999
- 16: Floreano, D., S. Nolfi, F. Mondada, Competitive Co-Evolutionary Robotics: From Theory to Practice, 1998
- 17: Mitchell, Tom M., Machine Learning, 1997
- 18: , Комиссия Фондовой биржи ММББ, ,