

ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
ФГАОУ ВО НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

Факультет компьютерных наук
Образовательная программа «Прикладная математика и информатика»

УДК 336.765

Отчет о программном проекте на тему:
Математические методы для торговли на бирже

Выполнили:

студент группы БПМИ202

(подпись)

(дата)

студент группы БПМИ202

(подпись)

(дата)

студент группы БПМИ204

(подпись)

(дата)

студент группы БПМИ204

(подпись)

(дата)

Принял руководитель проекта:

Румянцев Владислав Дмитриевич
Приглашенный преподаватель
Факультета компьютерных наук НИУ ВШЭ

(подпись)

(дата)

Москва 2023

Содержание

Аннотация	4
Введение (Курилов Олег)	7
Обзор литературы (Куцаков Александр)	8
1 Предобработка данных для обучения (Казадаев Максим и Куцаков Александр)	9
1.1 Разметка по времени (Казадаев Максим)	9
1.1.1 Volume Bars и Dollar Bars	9
1.1.2 Tick Imbalance Bars и Volume Imbalance Bars	10
1.2 Инструменты технического анализа (Куцаков Александр)	12
1.2.1 Тренд (trend)	12
1.2.2 Волатильность (volatility)	14
1.2.3 Возврат к среднему (mean-reversion)	15
2 Оптимизация портфеля (Казадаев Максим и Куцаков Александр)	17
2.1 Оценка стратегии (Казадаев Максим)	17
2.2 Классический подход (Казадаев Максим)	19
2.2.1 Оптимизация по Марковицу	19
2.2.2 Удаление рыночной компоненты доходности	20
2.2.3 Удаление шума	22
2.2.4 Кластеризация матрицы корреляций	23
2.3 Предлагаемые методы обучения с подкреплением (Куцаков Александр)	25
2.3.1 Постановка задачи обучения с подкреплением	25
2.3.2 Терминология и выбор методов	25
2.3.3 REINFORCE	27
2.3.4 Advantage Actor-Critic (A2C)	28
2.3.5 Deep Deterministic Policy Gradient	29
3 Стохастические дифференциальные уравнения (Stochastic Differential Equations, SDE) (Курилов Олег)	31
3.1 Определение SDE	31
3.2 Примеры процессов и исследование их свойств	32

3.3	Оценка параметров	33
3.4	Применение SDE	33
3.4.1	Расчёт справедливой цены	33
3.4.2	Игра на волатильности	34
3.4.3	Применения для генерации данных	34
4	Модели на классификации свеч (Кузьмин Максим)	34
4.1	Выбор бумаги для построения стратегии	35
4.2	Построение стратегии и предобработка данных	35
4.3	Метрики оценки качества классификации	36
5	Коннектор к бирже	36
5.1	Задачи коннектора (Кузьмин Максим)	37
5.2	Архитектура (Казадаев Максим)	37
5.2.1	InfoManager	38
5.2.2	MarketManager	38
5.2.3	UserManager	39
5.2.4	Strategy	41
6	Выполненные задачи и план дальнейшей работы	43
	Список литературы	45

Аннотация

В данной работе разработана алгоритмическая стратегия для торговли на бирже рассматривается с разных сторон: правильная обработка финансовых данных, составление математических моделей и использование методов машинного обучения. Результатом проекта станут несколько стратегий, работающих на средней частоте торгов. Первая стратегия – оптимизация портфеля с помощью обучения с подкреплением, которую разработают Казадаев Максим и Куцаков Александр. Второй тип стратегий – стратегии, основанные на случайных процессах, заключающихся в оценке волатильности ряда и справедливой цены производных финансовых инструментов, которую разработает Курилов Олег. Третья – свечной анализ с использованием алгоритмов классификации, которую разработает Кузьмин Максим. Для увеличения обучающей выборки будет написан генератор биржевых данных. Дополнительно будет разработана программная реализация стратегий и инфраструктура для их использования через Tinkoff Invest API.

Ключевые слова

Оптимизация по Марковицу, технический анализ, возврат к среднему, обучение с подкреплением, policy gradient методы, actor-critic методы, REINFORCE, A2C, DDPG, Black-Scholes economy, SDE, моделирование временного ряда.

Введение (Курилов Олег)

В данной работе рассматриваются различные подходы к торговле на бирже, а также непосредственно тонкости работы с ними.

Почему именно финансовые рынки? Причиной являются две вещи:

1 Появление математического подхода.

Где-то 40 лет назад на биржу стали активно приходить игроки, имеющие математическое образование. Задача прогнозирования будущих цен естественным образом является известной математической задачей прогнозирования временного ряда. Из-за того, что на биржу приходило всё больше математиков, со временем появилась целая области в математике, называемая финансовой математикой. Сейчас она разрослась и использует подходы из многих областей, начиная с классической математики и теории вероятностей и заканчивая передовыми методами глубинного обучения.

2 Развитие вычислительных мощностей и появление алгоритмической торговли.

С появлением электронных машин люди получили доступ к автоматизации вычислений, а с появлением интернета через пару десятилетий появилась и электронная торговля, что позволило также автоматизировать процесс. Люди стали придумывать и реализовывать различные модели и методы, ускоряющие торги и увеличивающие их объем, что активно развивало рынок.

Благодаря причинам, описанным выше, появляется широкий спектр задач, где можно использовать как теорию вероятностей в виде регрессионного анализа, статистики и случайных процессов, так и машинного обучения в виде классических и уже крайне современных методов.

В отчёте рассматриваются следующие вопросы и методы решения задач из финансовой математики:

1 Предобработка данных.

Финансовые данные, как и большинство данных, имеют свои особенности, которые затрудняют их анализ и ухудшают возможности прогнозирования.

Проблема непосредственно финансовых данных в их объемности и шумности. Обычно данные торгов представлены в виде книги ордеров, где написано кто сколько активов за сколько выставил на покупку/продажи или купил/продал. Полный анализ всей этой информации крайне затруднителен в силу огромного шума, порожденного этими данными. Для решения этой проблемы используют подходы, описанные в первой части

главы.

Но даже после того, как данные приведены к виду, более удобного для анализа, оказывается, что они всё ещё слишком шумны и близки к случайным. Поэтому хочется научиться выделять какие-то основные критерии ряда, чтобы с ним можно работать. Поиску таких критериев и посвящена вторая часть первой главы.

2 Оптимизация портфеля

Для заработка на бирже используют разные модели, но как можно оценить какая модель лучше или хуже? Для этого нужно ввести какие-то метрики, которые описываются в первой главе.

Во второй главе описываются классические подходы, которые основаны на чистой математике. Также приведены методы, позволяющие улучшить их.

В третьей главе описываются методы, основанные на обучении с подкреплением (Reinforcement Learning, RL). Данные методы появились сравнительно недавно ввиду роста вычислительных мощностей и используют нейросети в основе своего подхода.

Новизна проекта заключается в генерации новых признаков и новых оптимизациях, так идея использования RL для трейдинга ещё мало изучена, потому что область молодая, а также в потенциально новых методах предобработки данных.

3 Стохастические дифференциальные уравнения

Порой, хочется уметь моделировать данные временного ряда или же просто приблизить временной ряд каким-то известным случайным процессом. Для всего этого используются стохастические дифференциальные уравнения. Определению этого понятия и посвящена первая часть главы.

В следующей части описываются основные примеры случайных процессов, используемых для моделирования распределения с описанием свойств, возникающих из-за вида модели, в частности можно отследить все критерии, введенные в главе 1.

Часть 3 описывает, как на практике можно оценить параметры распределения с помощью метода максимального правдоподобия.

Часть 4 посвящена приложению случайных процессов в финансах для расчёта справедливой цены и в машинном обучении для генерации данных.

Для генерации данных по временным рядам есть материал по классическим алгоритмам МСМС и методам глубинного обучения, но мало работ по современным методам взаимодействия частиц или диффузионным моделям, что является новизной

4 Модели для классификации свеч

Можно ставить не задачу поиска правильной цены, а ограничиться вопросом пойдёт ли цена вверх или вниз хотя бы на сколько-то процентов. Постановка данной задачи и выбор актива для его анализа рассмотрен в первой части главы.

Часть 2 описывает, как и почему выбрана модель, и как нужно преобразовывать данные конкретно для задачи классификации.

В последней части описано, как нужно правильно оценивать модели в задаче классификации и какие проблемы возникли из-за этого.

Новизна метода заключается в выборе рассматриваемого интервала, который является среднесрочным с точки зрения рынка. Обычно же рассматриваются срочные и мало-срочные методы.

5 **Коннектор к бирже** Мало придумать стратегии, ведь её нужно подключить для торгов на практике, ровно, как и мало оценивать данные, ведь их нужно быстро получать. Решением этой проблемы занимается коннектор к бирже. Подробно про задачу можно почитать в первой половине главы.

Последняя глава подробно рассказывает какая архитектура была использована для реализации коннектора, какие основные части в ней есть, и какие важные решения приняты.

6 **План** Последняя глава рассказывает про распределение ролей, про то, что уже сделано и про дальнейшие планы.

Обзор литературы (Куцаков Александр)

Задача ведения алгоритмической торговли возникла давно и претерпела несколько различных подходов. Постепенная автоматизация торговли привела к появлению технического анализа рынка и большого количества индикаторов [6]. В большинстве случаев значение индикатора в текущий момент времени определяются ценами актива за фиксированное число предыдущих “свечей”. Для их подсчёта часто используются несложные статистики – чаще всего среднее. Это позволяет применять одновременно только небольшое количество технических индикаторов [14] из-за их высокой коррелированности. Такая проблема возникает вполне обоснованно, поскольку каждый индикатор является самостоятельным и детерминированную стратегию можно построить по нему одному. Тем не менее подсчёт все-

возможных статистик по предыдущим периодам является обязательным и будет использован для наших стратегий.

Дальнейший технический прогресс достаточно быстро вызвал популярность высокочастотной торговли, в которой основное внимание уделяется мощности оборудования и скорости получения информации и отправки ответа на биржу. В нашем проекте мы не будем рассматривать эти методы, поскольку ориентируемся на среднюю частоту. Отсюда вытекает возможность использовать методы с большими вычислительными и временными затратами для обработки поступающей информации, например, нейронные сети или другие сложные алгоритмы машинного обучения.

Переходя к обучению с подкреплением (Reinforcement Learning, RL), существует несколько общих подходов в этом методе. Сразу стоит отметить, что использование RL показывает себя гораздо лучше, чем алгоритмические стратегии, рассмотренные ранее [21]. Первый из подходов RL, *critic-only*, учится оценивать полезность определённого действия в данном состоянии, применим для дискретного пространства действий. В нём часто используются [2] [10] Deep Q-Learning нейронные сети [13]. В нашем случае это пространство можно считать дискретным, но тогда оно будет иметь экспоненциально большой размер, который мы не сможем хранить в памяти. Второй подход, *actor-only*, позволяет использовать непрерывное пространство действий, поскольку агент учится сразу некоторой, необязательно дискретной, политике в каждом состоянии, а не паре состояние + действие, как это происходит в предыдущем методе. Несмотря на хорошие результаты и возможность вести торговлю онлайн [5], этот подход достаточно рискованный, если не учить агента на протяжении очень большого промежутка времени. Связано это с тем, что модель не отличает плохое действие от хорошего, если награда остаётся большой, поскольку, в отличие от *critic-only*, мы не учимся максимизировать её отдельно для каждого действия, а смотрим только на состояние в целом. Последний подход, *actor-critic*, является объединением первых двух, как можно догадаться из названия. В нём обучаются параллельно обе стороны – агент, который отвечает за политику, и критик, оптимизирующий значения Q -функции. Данный метод наиболее подходит для задачи, поскольку объединяет плюсы предыдущих, к минусам его можно отнести высокую чувствительность к изменению гиперпараметров, а с этой проблемой можно бороться, например, с помощью ансамблирования нескольких методов [20].

1 Предобработка данных для обучения (Казадаев Максим и Куцаков Александр)

1.1 Разметка по времени (Казадаев Максим)

Работать напрямую с книгой ордеров и сделками на бирже невозможно из-за огромного количества данных и высокого уровня шума в них. Поэтому, возникает идея агрегации рыночной информации за некоторые промежутки времени. Многие исследователи рассматривают статистики по временному ряду с фиксированной периодичностью, например, рассматривают минутные свечи. Однако такой подход не оптимальный, особенно в приложении к финансовым временным рядам.

Первая проблема заключается в том, что количество игроков на бирже меняется в течение дня. Бывает, что торговля идет менее активно, а иногда в дни высокой волатильности на рынке объем торгов увеличивается.

Другой проблемой является тот факт, что для некоторых моделей необходимо, чтобы между тиками, которые подаются им на вход, признаки менялись с одинаковой скоростью. Промежутки времени должны заканчиваться в те моменты, когда появилась новая информация, т. е. было замечено значительное изменение некоторых статистик.

Данные проблемы затрагивались в [4], где были предложены подходы Volume Bars и Dollar Bars для работы с разницей в торговой активности, подходы Tick Imbalance Bars и Volume Imbalance Bars для разделения ряда на отрезки с одинаковым количеством информации. Данные подходы мы применим при построении наших торговых стратегий, выберем среди них лучшие.

1.1.1 Volume Bars и Dollar Bars

Чтобы учесть периодические эффекты, рассмотрим статистики не за фиксированные промежутки времени, а за переменные. Будем использовать меньшие промежутки времени в периоды больших объемов торгов и более длинные при низкой торговой активности. Чтобы добиться этого эффекта, будем включать в каждый период фиксированный объем торгов. Т. е. если в какой-то момент совокупный объем торгов достигает фиксированной отметки, то мы считаем текущий отрезок (bar) завершенным и начинаем новый отрезок (bar).

Объем можно измерять как в единицах актива, так и в денежном выражении. Назовем подход с использованием единиц актива – Volume Bars, а с использованием денежных единиц – Dollar Bars. Часто объем торгов участников рынка в денежном выражении ведет себя более

стабильно, чем объем в единицах актива. Есть очень простое объяснение такого эффекта. Участники рынка в среднем получают доходность, близкую к безрисковой. Капитал, который используется для торговли, в денежных единицах остается постоянным. Объем в единицах актива при фиксированном объеме в денежном выражении обратно пропорционален его цене. После получения нами необходимых данных об объемах торгов, мы рассмотрим, какой вид объема ведет себя более стабильно. Исходя из этого, мы выберем Dollar Bars или Volume Bars.

1.1.2 Tick Imbalance Bars и Volume Imbalance Bars

Проблему нахождения моментов изменения характеристик временного ряда решим с помощью Tick Imbalance Bars и Volume Imbalance Bars. Дополнительно исследуем другие подходы, аналогичные им. Все они основаны на том, что мы будем учитывать не только объемы торгов, но и изменения цен. Конкретнее, будем учитывать отдельно объемы, которые привели к повышению и понижению цены. В тот момент, когда они будут вести себя не как раньше, мы будем считать промежуток завершенным.

Начнем с Tick Imbalance Bars. Пусть мы завершили формирование прошлого отрезка, и начали новый. С момента его начала имеем последовательность цен $\{p_t\}_{t=0, \dots, T-1}$. Это может быть либо последовательность цен сделок, либо последовательность цен закрытия с фиксированной периодичностью. Определим последовательность $\{b_t\}_{t=0, \dots, T-1}$, где $b_t \in \{-1, 1\}$:

$$b_t = \begin{cases} b_{t-1}, & \text{если } \Delta p_t = 0, \\ \text{sign}(\Delta p_t), & \text{если } \Delta p_t \neq 0 \end{cases}$$

где b_0 – начальное значение (отсчитывается после первого изменения цены), $\Delta p_t = p_t - p_{t-1}$

– изменение цены, $\text{sign}(x) = \begin{cases} +1, & x > 0 \\ -1, & x < 0 \end{cases}$

Определим значение асимметрии (imbalance) так:

$$\theta_T = \sum_{t=0}^{T-1} b_t$$

Асимметрия отражает, насколько сильно цена чаще менялась в одну сторону, чем в другую. По нему мы будем отмерять отрезки, заканчивая текущий там, где асимметрия превысила ее ожидаемое значение. Более формально, рассмотрим случайную величину T_{end} – время окончания текущего отрезка. Посчитаем математическое ожидание асимметрии для

момента времени T_{end} при условии, что мы находимся в $t = 0$:

$$\mathbb{E}_0[\theta_{T_{end}}] = \mathbb{E}_0 \left[\sum_{t=0}^{T_{end}-1} b_t \right] = \mathbb{E}_0[T_{end}] \cdot \mathbb{E}_0[b_t] = \mathbb{E}_0[T_{end}] \cdot (P_0[b_t = 1] - P_0[b_t = -1]),$$

где $\mathbb{E}_0[T_{end}]$ – ожидаемая продолжительность отрезка, $P_0[b_t = 1]$ – частота положительных b_t .

В моменте времени $t = 0$ мы можем оценить $\mathbb{E}_0[T_{end}]$ как экспоненциальное скользящее среднее по продолжительности прошлых периодов, а $(P_0[b_t = 1] - P_0[b_t = -1])$ как экспоненциальное скользящее среднее по b_t в этих периодах. Отметим, что продолжительность периода зависит от продолжительности прошлых периодов. Поэтому, в самом начале необходимо задать продолжительность первого периода и вероятность $P_0[b_t = 1] = \frac{1}{2}$. Заметим, что может возникнуть проблема сильного увеличения или уменьшения $\mathbb{E}_0[T_{end}]$, что не очень хорошо отразится на стратегии. Если такая проблема возникнет, будем сокращать разность между $\mathbb{E}_0[T_{end}]$ и длиной первого отрезка на фиксированное число процентов.

Будем считать, что отрезок закончился, когда асимметрия по модулю превысит ожидаемую асимметрию, т. е. первый момент T , в который:

$$|\theta_T| \geq \mathbb{E}_0[T_{end}] \cdot |P_0[b_t = 1] - P_0[b_t = -1]|$$

Таким образом, мы будем получать более короткие промежутки времени, когда цена будет долго меняться в одном направлении. На каждом отрезке будет содержаться новая информация. Например, статистики, оценивающие тренд, будут показывать разные значения.

Tick Imbalance Bars имеют существенный недостаток. Они не учитывают объемы торгов. Например, свечи с минимальным объемом будут иметь такое же влияние, как и свечи с большим объемом торгов. Чтобы учесть информацию об объеме, попробуем использовать Volume Imbalance Bars.

Аналогично, введем асимметрию (imbalance) как:

$$\theta_T = \sum_{t=0}^{T-1} b_t \cdot v_t,$$

где v_t – объем сделки, или в случае свечей, объем, проторгованный за t -ый период.

Ожидаемая асимметрия за следующий промежуток:

$$\mathbb{E}_0[\theta_{T_{end}}] = \mathbb{E}_0 \left[\sum_{t=0}^{T_{end}-1} b_t \cdot v_t \right] = \mathbb{E}_0[T_{end}] \cdot (P[b_t = 1]\mathbb{E}_0[v_t|b_t = 1] - P[b_t = -1]\mathbb{E}_0[v_t|b_t = -1]) =$$

$$= \mathbb{E}_0[T_{end}] \cdot (v_+ - v_-),$$

где $v_+ = P[b_t = 1]\mathbb{E}_0[v_t|b_t = 1]$ и $v_- = P[b_t = -1]\mathbb{E}_0[v_t|b_t = -1]$

$\mathbb{E}_0[T_{end}]$ аналогично оцениваем как скользящее среднее прошлых длин отрезков, $v_+ - v_-$ как скользящее среднее по предыдущим $b_t v_t$. Начальные значения и небольшой возврат к фиксированной продолжительности реализуем так же, как и для Tick Imbalance Bars.

Tick Imbalance Bars и Volume Imbalance Bars позволят нам расставить периоды так, чтобы между соседними точками распределение признаков менялось примерно в одинаковой степени. Модели машинного обучения смогут извлекать одинаковое количество полезной информации из каждого периода.

1.2 Инструменты технического анализа (Куцаков Александр)

Технический анализ в стандартном виде можно считать эквивалентным анализу графика изменения цены со временем без какой-либо дополнительной информации. Задача заключается в предсказании цены в текущий момент, имея информацию о ней за предыдущие. Ограничивать себя только ценой нельзя: как было сказано ранее, не только цена активов играет роль в движении рынка. Однако правильная работа с прошлыми значениями и извлечением из них признаков для обучения модели в любом случае необходима.

Несмотря на то, что технический анализ включает не так много некоррелированных инструментов, как уже было упомянуто выше, индикаторы имплементируют достаточно хорошие способы поиска полезных статистик, и нам потребуется использовать несколько приёмов оттуда. В первую очередь, индикаторы могут показывать несколько принципиально разных аспектов изменения цены со временем. К ним относятся, в частности, её среднее и дисперсия (волатильность). Используя детерминированные алгоритмы с проверенной эффективностью по сравнению с другими подобными, можно найти наиболее релевантные для практического применения статистики, чему и будет посвящён этот раздел.

1.2.1 Тренд (trend)

Наиболее логичным способом предсказания цены является наблюдение за её средним за последние интервалы, для этого используются скользящие средние [17]. Например, пусть взяты значения цены P_1, \dots, P_k для k моментов в прошлом (будем считать, что они упорядочены по времени). Это могут быть цены открытия или закрытия для свечей за k предыдущих

периодов. Более простым является обычное скользящее среднее

$$SMA = \frac{P_1 + \dots + P_k}{k}$$

которое считается в предположении, что все рассмотренные моменты времени находятся достаточно близко и в равной степени характеризуют цену в текущий момент. *Смысл:* достаточно точная оценка цены в текущий момент.

Однако более логично считать, что близкие периоды имеют более высокую корреляцию с текущим моментом времени, потому чаще используется экспоненциальное скользящее среднее, которое учитывает ранние периоды с меньшим весом

$$EMA_t = \begin{cases} P_1, & t = 1 \\ \alpha \cdot P_t + (1 - \alpha) \cdot EMA_{t-1}, & t > 1 \end{cases} \quad t \in [k], \alpha \in [0, 1]$$

Смысл: оценка текущей стоимости актива, коэффициент α показывает уровень доверия предыдущему состоянию по сравнению с более ранними.

Именно это среднее используется для вычисления *трендового* индикатора MACD (Moving Average Convergence/Divergence). Более строго, пусть $EMA_{t,i}$ — значение экспоненциального скользящего среднего, посчитанного за i предыдущих периодов, в момент времени t , тогда

$$MACD_t = EMA_{t,a} - EMA_{t,b}$$

Где $a < b$, и обычно используется $a = 12, b = 26$ для продаж, $a = 8, b = 17$ для покупок. Использование обычно включает подсчёт

$$SGNL_t = EMA_{t,c}(MACD) - \text{среднее считается не по цене, а по значениям } MACD$$

значения которого зачем сравниваются с самим индикатором. *Смысл:* сам $MACD$ принимает значения недалеко от нуля и показывает возрастающий или убывающий тренд. Сравнение с его же скользящим средним нужно, чтобы понять, продолжается ли рост (если он выше) или сходит на нет.

Использование среднего в оценке не только логично, но и имеет теоретическое обоснование в случае симуляции Броуновским процессом [16], однако стабильно работает только при достаточно медленных сменах тренда, что может встречаться в поведении рынка, можно считать его некоторой частью текущей траектории, но нельзя строить стратегию только на нём.

1.2.2 Волатильность (volatility)

Естественным продолжением среднего является добавление дисперсии, поскольку для стратегии важно понимать риск, связанный с каждым активом. Для этого значение индикатора (комбинации скользящих средних) нормируется на стандартное отклонение [16]

$$N_t = \frac{MACD_t}{sd_{t,m}}$$

где t – текущий момент времени, m – число периодов для подсчёта стандартного отклонения цены по ним. Такая нормировка повышает значение индикатора для менее волатильных периодов и понижает для более волатильных, что позволяет нам учитывать риск. *Смысл:* насколько тренд, который показывает индикатор, вызван ростом актива, а не его волатильностью.

Помимо учёта дисперсии, волатильность активов можно учитывать и другими способами. Для этого применяются осцилляторы, которые не напрямую учитывают дисперсию, а предсказывают поведение цены в ближайшем будущем, что позволяет их применять отдельно от трендовых индикаторов [3]. В качестве одного из лучших можно рассматривать следующий индикатор

$$RSI_t = \begin{cases} 100 - \frac{100}{1 + \frac{AG_t}{AL_t}} & t \leq N \\ 100 - \frac{100}{1 + \frac{13 \cdot AG_{t-1} + G_t}{13 \cdot AL_{t-1} + L_t}} & t > N \end{cases}$$

Здесь $N = 14$, G_t (*gain*) = $\max\{0, R_t\}$, L_t (*loss*) = $\max\{0, -R_t\}$ – зависят от return-а в момент времени t , который равен $P_t - P_{t-1} = R_t$. AG , AL усредняют значения G и L соответственно. RSI показывает перекупленность (обычно выше 70) или перепроданность (ниже 30) актива, с помощью него можно предсказывать изменения тренда, которые зачастую связаны с высоким разбросом цен и объёмом торгов.

Помимо стандартного анализа одной и той же статистики цены (например, средней между закрытием и открытием) за каждый период, мы можем также использовать максимальное и минимальное её значение, доступ к которым дают японские свечи¹. Это будет небольшим отступлением от концепции анализа одного графика, которое позволяет нам не только оценивать волатильность актива с течением времени, но и в данный период – визуально человек постоянно это делает по свечам.

Зафиксируем обозначения P_t^l, P_t^h, P_t^c – минимальная и максимальная цены, а также цена закрытия соответственно за период t . Длина скользящего окна по-прежнему равна k .

¹https://en.wikipedia.org/wiki/Candlestick_chart, дата обращения 12.02.2023

В качестве оценки дисперсии будем использовать CCI (Commodity Channel Index), который считается по формуле

$$CCI_t = \frac{TP_t - \overline{TP}_t}{\varepsilon \cdot D_t}$$

В этой формуле $TP_t = \frac{P_t^l + P_t^h + P_t^c}{3}$, $\overline{TP}_t = \frac{PT_t + \dots + PT_{t-k+1}}{k}$, $D_t = \frac{1}{k} \sum_{i=1}^k |TP_{t-i+1} - \overline{TP}_t|$, $\varepsilon = 0.015$. *Смысл*: достаточно близок к MACD, только вместо разности двух средних рассматривается более мгновенное свойство. Индекс показывает отношение текущей цены (TP) к своему среднему, используя при этом дополнительную информацию от свечей; \overline{TP} просто считает скользящее среднее; D_t показывает отклонение цены в предыдущих периода, что является аналогом дисперсии для L_1 -нормы; коэффициент ε просто с большой вероятностью нормирует результат в удобные границы $[-100, 100]$. Количество периодов k обычно полагается равным 20. Можно заметить, что индикатор не имеет строгих границ, поэтому сигналы от него будут разными для разных активов, откуда следует, что лучше использовать сигналы от других осцилляторов, чтобы подтвердить или опровергнуть эти. Однако использование других значений из данных одного периода поможет избежать коррелированности, потому у нас появляется возможность брать больше “разных” индикаторов.

1.2.3 Возврат к среднему (mean-reversion)

Помимо нахождения некоторых аналогов среднего и дисперсии, существует множество других индикаторов. Поскольку принцип работы и основные типы уже рассмотрены, то дальнейшая работа может включать их подсчёт и возможное использование в качестве признаков для обучения модели. Однако наиболее прикладные индикаторы уже были перечислены и большое пополнение заявленного списка не имеет особого смысла в силу большой коррелированности, что наблюдается и на практике [20]. Поэтому перейдём к конкретной статистической идее, которая имеет интуитивное обоснование.

Идея возврата к среднему (mean-reversion) заключается в том, что при нахождении цены выше среднего уровня имеет смысл ожидать движения вниз, иначе – движения вверх. Действительно, при, например, росте цены на актив будут приниматься заявки из стакана на продажу, создавая предложение, которое может вызвать отскок цены обратно к среднему значению. Более формально, используя обозначения предыдущих пунктов и аппроксимируя доходность (return) линейной моделью [1]

$$R_t = \lambda \cdot P_{t-1} + \mu + \beta \cdot t + \alpha_1 \cdot R_{t-1} + \dots + \alpha_k \cdot R_{t-k} + \varepsilon_t \quad (1)$$

Здесь $\lambda, \beta, \alpha_i, i \in [k]$ – некоторые параметры, ε_t – шум. Далее используется статистический тест Augmented Dickey-Fuller Test² для статистики

$$T = \frac{\lambda}{se(\lambda)}, \quad se(\cdot) \quad - \text{стандартное отклонение}$$

Проверяется гипотеза $H_0 : \lambda = 0$, против альтернативы $H_1 : \lambda < 0$. Заметим, что $\lambda = 0$ означает независимость return-а от значения цены на предыдущем шаге, то есть имеем некоторый аналог случайного блуждания. Если же $\lambda < 0$, то в нашей модели наблюдается эффект mean-reversion – с ростом цены происходит уменьшение return на следующем периоде.

Однако в реальности такая модель сталкивается с несколькими проблемами. Во-первых, ограничения для H_1 слишком строгие, тогда как в реальности mean-reversion всё равно может наблюдаться и достаточно его более слабых проявлений для получения прибыли. Во-вторых, сам коэффициент дрефта β также в реальности очень мал по сравнению с флуктуациями цены – её вызывает большая зашумленность рыночных данных. Поэтому модель (1) можно ослабить и упростить, убрав зависимость от $R_{t-i}, i \in [k]$ и от β . Упрощённый вид в непрерывной форме является уравнением для процесса Орнштейна-Уленбека³

$$dP_t = (\lambda P_{t-1} + \mu)dt + d\varepsilon \quad (2)$$

здесь $d\varepsilon$ будет уже гауссовым шумом, а остальные коэффициенты совпадают с одноимёнными из (1). Заметим, что при $\lambda = 0$ приращения станут независимыми, аналогично дискретному случаю, поэтому будем считать $\lambda \neq 0$. Избавившись от части переменных, можно из (2) найти аналитическое выражение для среднего цены

$$\mathbb{E}P_t = y_0 \exp(\lambda t) - \frac{\mu}{\lambda} \cdot [1 - \exp(\lambda t)]$$

Пусть $\lambda < 0$ и выполняется свойство mean-reversion. Тогда матожидание убывает экспоненциально со временем, поскольку $-\lambda t > 0$. Несложно видеть, что при $\lambda \rightarrow 0-0$ убывание замедляется, растягиваясь на большее время. Формально это описывается *периодом полураспада* процесса и он будет равен $t_{hf} = -\frac{\ln 2}{\lambda} \rightarrow +\infty$ при $\lambda \rightarrow 0-0$ (t_{hf} – такой промежуток времени, за который мы ожидаем, что расстояние до среднего уменьшится в два раза). Посчитать само λ можно с помощью приближения полученной непрерывной версии, подавая в линейную модель $R_t = P_t - P_{t-1}$ в качестве dP_t . Период полураспада не только помогает

²https://en.wikipedia.org/wiki/Augmented_Dickey%E2%80%93Fuller_test, дата обращения 12.02.2023

³https://en.wikipedia.org/wiki/Ornstein%E2%80%93Uhlenbeck_process, дата обращения 12.02.2023

визуально оценить время, за которое цена возвращается к среднему, но и является хорошим ориентиром для выбора длины скользящего окна (число периодов k , по которым считаются скользящие метрики). При неправильном выборе числа периодов модель может выдавать менее надёжные результаты из-за неполноты или избыточности информации (причём эти результаты не получится исправить в процессе или это будет сделать очень сложно), потому выбор этого гиперпараметра является ключевым. Чтобы сократить перебор по нему, в качестве начального приближения можно использовать период полураспада.

2 Оптимизация портфеля (Казадаев Максим и Куцаков Александр)

Допустим, имеется несколько различных активов, которые торгуются на бирже, например, наиболее ликвидные акции. Мы хотели бы оптимально распределить средства между ними. Оптимально означает, что мы хотим иметь высокую доходность и низкие риски. Данные понятия формализованы в разделе "Оценка стратегии". Со временем, получая новую рыночную информацию, мы можем перераспределять средства между активами. На суммарную стоимость портфеля влияют изменения цен приобретенных активов, а также издержки при перебалансировке в виде комиссий и потерь при исполнении, например, пересечение bid-ask спреда. Классическим подходом является оптимизация портфеля по Марковицу [19]. Однако он имеет несколько недостатков. В чем они заключаются, и как с ними бороться, описано в разделах "Удаление рыночной компоненты", "Удаление шума" и "Кластеризация матрицы корреляций". Тем не менее, даже если использовать модификации, транзакционные издержки и изменчивость среды в классическом методе не учитываются. Чтобы учесть их, естественно применить методы обучения с подкреплением. О предлагаемых подходах написано в разделе "Предлагаемые методы обучения с подкреплением". Именно комбинация классического подхода и обучения с подкреплением может дать хороший результат.

2.1 Оценка стратегии (Казадаев Максим)

Можно поставить цель для стратегии по-разному. Наивный подход заключается в максимизации математического ожидания годовой доходности. В реальности инвесторов волнует не только ожидаемая доходность, но еще и распределение этой доходности. Большинство инвесторов являются не склонными к риску. Они готовы иметь меньшую ожидаемую доходность, если ее разброс значительно снизится. Вследствие такого поведения, на рынке

складываются условия, в которых наиболее высокую доходность имеют более рискованные активы, потому что из-за их рискованности покупать их готовы только с дисконтом. Если максимизировать ожидаемую доходность, не обращая внимания на риск, то в портфеле с большой вероятностью окажется много рискованных инструментов. Т. к. инвесторы склонны избегать риска, то необходима корректировка цели стратегии.

Вместо ожидаемой доходности часто максимизируют коэффициент Шарпа (Sharpe ratio, SR), показывающий отношение доходности к риску:

$$SR = \frac{\mathbb{E}[R_{\text{strategy}} - R_f]}{\sqrt{\mathbb{D}[R_{\text{strategy}} - R_f]}} = \frac{\mathbb{E}[R_{\text{strategy}} - R_f]}{\sigma_{\text{strategy}}},$$

где R_{strategy} – годовая доходность стратегии, R_f – безрисковая ставка, например, по государственным облигациям, σ_{strategy} – стандартное отклонение годовой доходности стратегии.

Интерпретировать его можно как скорректированную на риск дополнительную доходность, которую мы получили по сравнению с альтернативой в виде безрисковых вложений. Данный коэффициент является безразмерной величиной в отличие от математического ожидания доходности.

У коэффициента Шарпа есть существенный недостаток. Коэффициент может значительно меняться в зависимости от макроэкономических условий. Существуют события, которые влияют на состояние практически всех компаний. Например, изменение ключевой ставки или геополитические события. Т. к. их нельзя предсказать на основе имеющихся рыночных данных, было бы хорошо исключить их влияние при оценке стратегии. Особенно это актуально для задачи оптимизации портфеля. Например, если разрешать стратегии иметь только длинные позиции, то существенной составляющей доходности будет общее движение рынка.

Чтобы исключить движение рынка, будем рассматривать Information Ratio (IR) [7]. Его можно получить, заменив в формуле коэффициента Шарпа безрисковую доходность R_f на доходность бенчмарка R_B :

$$IR = \frac{\mathbb{E}[R_{\text{strategy}} - R_B]}{\sqrt{\mathbb{D}[R_{\text{strategy}} - R_B]}} = \frac{\mathbb{E}[R_{\text{strategy}} - R_B]}{\sigma_{\text{strategy} - \text{benchmark}}},$$

где $\sigma_{\text{strategy} - \text{benchmark}}$ – стандартное отклонение разности доходности стратегии и бенчмарка.

Когда ставится задача балансировки портфеля, альтернативными вложениями являются не покупка государственных облигаций, а индексное инвестирование. Поэтому, в качестве бенчмарка для оптимизации портфеля удобно рассматривать рыночный индекс, в

который включены акции, которыми торгует стратегия. Например, в случае торговли акциями на Московской бирже, бенчмарком является индекс МОЕХ. Также, в качестве бенчмарка можно брать классические подходы к оптимизации портфеля. Information Ratio будет использоваться для оценки стратегии оптимизации портфеля в данном проекте.

2.2 Классический подход (Казадаев Максим)

Классический подход к оптимизации портфеля, который часто используют на практике, пригодится нам, во-первых, как baseline модель. Во-вторых, некоторые трюки могут быть использованы при генерации новых признаков для машинного обучения. Например, далее будут описаны техники, позволяющие убрать из матрицы корреляций общее движение рынка, убрать из нее шум и значительно снизить ее размерность. Дополнительно, полезно рассмотреть в качестве нового признака финальное распределение между активами, которое дает классическая стратегия. Обучаемой модели останется только правильно скорректировать его, чтобы получить результаты лучше.

2.2.1 Оптимизация по Марковицу

Классической теорией для решения поставленной задачи является оптимизация по Марковицу [19]. Допустим, имеется N активов, которые торгуются на бирже. Будем рассматривать состояние рынка в моменты времени $t \in 0, 1, \dots, T - 1$. Расстояние между этими моментами времени в классическом подходе считается одинаковым. К моменту времени t доступны прошлые цены активов. Стратегия принимает решение о распределении весов инструментов в своем портфеле, т. е. выбирает вектор $w \in \mathbb{R}^N$ такой, что сумма его координат равна 1. В классическом подходе не рассматривается проблема перебалансировки портфеля со временем.

Самая важная часть стратегии заключается в процедуре подбора оптимальных весов активов w_i . Пусть сейчас момент времени $T - 1$, и мы имеем достаточно большую историю цен. Обозначим вектор цен активов в момент времени t за $p^{(t)} \in \mathbb{R}^N$. Вектор доходностей активов между $t - 1$ и t равен $r^{(t)} = \frac{p^{(t)}}{p^{(t-1)}} - 1$. Положим доходности в каждый период по строкам в матрицу $R = (r^{(0)}, \dots, r^{(T-1)})^T$, $R \in \mathbb{R}^{T \times N}$. Найдем выборочное среднее доходностей $\mu = \widehat{\mathbb{E}}r = \frac{1}{T} \sum_{t=0}^{T-1} r^{(t)}$, а также их выборочную матрицу ковариаций $\Sigma \in \mathbb{R}^{N \times N}$, где $\Sigma_{ij} = \widehat{\text{cov}}(r_i^{(t)}, r_j^{(t)})$. Она равна:

$$\Sigma = \frac{1}{T} \sigma^T R_{\text{scaled}}^T R_{\text{scaled}} \sigma,$$

где R_{scaled} – матрица доходностей R , в которой отнормировали столбцы (в каждом столб-

це вычли среднее и разделили на стандартное отклонение), $\sigma \in \mathbb{R}^N$ – вектор стандартных отклонений по столбцам в R .

Если считать, что мы оценили верно вектор ожидаемых доходностей и матрицу их ковариаций, то можно записать формулу для ожидаемой доходности и ее дисперсии для всего портфеля:

$$\mathbb{E}[R_{\text{portfolio}}] = E \left[\sum_{i=1}^N r_i^{(t+1)} \cdot w_i \right] = \sum_{i=1}^N \mu_i w_i = \mu^T w$$

$$\text{Var}[R_{\text{portfolio}}] = \text{cov} \left(\sum_{i=1}^N r_i^{(t+1)} w_i, \sum_{i=1}^N r_i^{(t+1)} w_i \right) = \sum_{i=1}^N \sum_{j=1}^N w_i w_j \text{cov}(r_i^{(t+1)}, r_j^{(t+1)}) = w^T \Sigma w$$

Зная ожидаемую доходность портфеля, равную $\mu^T w$, и дисперсию этой доходности, равную $w^T \Sigma w$, можно поставить задачу оптимизации:

$$w^T \Sigma w - q \cdot \mu^T w \rightarrow \min_w \quad \text{s.t.} \quad 1^T w = 1,$$

где $q \in [0, +\infty)$ – параметр, отражающий насколько инвестор склонен к риску. Если $q = 0$, то инвестор абсолютно нетерпим к риску, в результате получается портфель с минимальной дисперсией. Если $q \rightarrow +\infty$, то инвестора не интересует дисперсия доходности, он вкладывает все средства в самые прибыльные активы.

Эквивалентно можно записать задачу в виде минимизации дисперсии доходности портфеля при его фиксированной ожидаемой доходности. Дополнительно можно накладывать ограничение, что все веса w_i должны быть неотрицательными, если стратегии не разрешены короткие позиции. Данный подход является основой для современных методов составления портфеля, но обладает недостатками, которые могут быть решены с помощью его модификаций.

2.2.2 Удаление рыночной компоненты доходности

Главная проблема оптимизации портфеля по Марковицу заключается в том, что распределение прошлых доходностей сильно отличаются от их распределения в будущем. Одним из главных неучтенных факторов является общее состояние рынка, о котором говорилось в разделе про методы оценки эффективности стратегии. В один год акции всех компаний могут сильно расти, в другой год сильно падать. Было бы хорошо исключить этот фактор при оптимизации. В частности, это можно сделать с помощью PCA (Principal Component analysis) [15].

Чтобы не учитывать дисперсию доходности каждого актива и сконцентрироваться

только на корреляциях между доходностями, далее будем считать, что доходности отнормированы. Т. е. рассмотрим матрицу доходностей R_{scaled} и матрицу корреляций

$$\Sigma_{\text{corr}} = \frac{1}{T} R_{\text{scaled}}^T R_{\text{scaled}}. \quad (3)$$

Если бы мы вложились в активы с весами $w \in \mathbb{R}^N$, то доходности портфеля были бы равны $R_{\text{scaled}} \cdot w \in \mathbb{R}^T$. Выделим портфель, который имеет наибольшую дисперсию, т. е. $w_{\text{market}} = \operatorname{argmax}_{\|w\|_2=1} \frac{1}{T} \|R_{\text{scaled}} \cdot w\|_2^2$. По определению w_{market} – первая компонента в PCA для матрицы корреляций (3). Стоит отметить, что это не совсем портфель, т. к. у него L_2 -норма равна 1, а не сумма координат 1. Мы используем матрицу корреляций, а не матрицу ковариаций, потому что в случае матрицы ковариаций PCA бы включал в первые компоненты с большим весом активы, которые имеют высокую дисперсию доходностей. А в случае матрицы корреляций в первой компоненте оказывается портфель из ”наиболее” коррелированных активов.

Можно найти и другие компоненты с помощью спектрального разложения матрицы корреляций:

$$\Sigma_{\text{corr}} = \frac{1}{T} R_{\text{scaled}}^T R_{\text{scaled}} = W \Lambda W^T \quad (4)$$

В матрице W будут идти по столбцам портфели в порядке убывания дисперсии. Чтобы исключить один или несколько портфелей с наибольшей дисперсией, достаточно занулить их собственные значения в Λ , что можно записать как:

$$C = W \Lambda W^T - W_M \Lambda_M W_M^T \quad (5)$$

$$\Sigma'_{\text{corr}} = \operatorname{diag}(C)^{-\frac{1}{2}} \cdot C \cdot \operatorname{diag}(C)^{-\frac{1}{2}} \quad (6)$$

где Σ'_{corr} – новая матрица корреляций, из которой исключили рыночную компоненту, W_M – первые несколько компонент в PCA (рыночные портфели), записанные по столбцам, Λ_M – диагональная матрица с собственными значениями при рыночных портфелях на диагонали. После удаления собственного значения в матрице C на диагонали окажутся значения, не равные 1. Чтобы получить из C новую матрицу корреляций Σ'_{corr} , нужен еще один шаг с нормировкой (6).

После удаления рыночной компоненты в матрице ковариаций другие зависимости будут лучше видны. Это положительно отразится на Information Ratio, т. к. мы будем сравнивать доходность стратегии с доходностью рыночного индекса, который очень похож на

рыночную компоненту. Будем использовать данный трюк как в классических методах, так и при построении признаков для машинного обучения.

2.2.3 Удаление шума

Еще одной проблемой оптимизации портфеля по Марковицу является наличие шума в ковариационной матрице. Это объясняется тем, что доходности в большой степени зависят от случайности. Эта случайность не несет никакой информации, она зашумляет результаты. Чтобы отделить случайность от той части ковариационной матрицы, в которой находится полезная информация, можно воспользоваться сравнением распределения собственных значений матрицы корреляций (4) с распределением Марченко-Пастура (Marchenko–Pastur) [12].

Распределение Марченко-Пастура. Пусть $X \in \mathbb{R}^{T \times N}$ – случайная матрица с независимыми одинаково распределенными значениями, которые имеют нулевое среднее и конечную дисперсию. Введем матрицу $Y = \frac{1}{T}X^T X$. Если $N \rightarrow +\infty$, $T \rightarrow +\infty$ с условием $\frac{N}{T} \rightarrow c \in (0, 1)$, то собственные значения матрицы Y сходятся к распределению Марченко-Пастура, которое имеет плотность:

$$p(\lambda) = \frac{\sqrt{(\lambda_+ - \lambda)(\lambda - \lambda_-)}}{2\pi\sigma^2 c \lambda} \cdot I\{\lambda_- < \lambda < \lambda_+\},$$

где $\lambda_{\pm} = \sigma^2(1 \pm \sqrt{c})^2$

С помощью данной теоремы мы определим, какие собственные значения (и соответственно собственные вектора) в матрице корреляций отражают шум, а какие несут полезную информацию. Опять будем использовать корреляционную матрицу, а не ковариационную матрицу, потому что для применимости распределения Марченко-Пастура доходности R_{scaled} должны быть отнормированы, т. е. иметь нулевое среднее и одинаковую дисперсию по каждому активу. Согласно распределению Марченко-Пастура, собственные значения $\lambda > \lambda_+$ при определенных условиях не должны встречаться в случайной матрице. Поэтому, можно неформально положить, что в (4) собственные значения $\lambda \in [0, \lambda_+]$ ассоциированы с шумом, а остальные несут неслучайную информацию.

После оценки параметров распределения удалить собственные значения можно аналогично удалению рыночной компоненты с помощью формул (5) и (6). Но можно сделать по-другому, не изменяя значения следа матрицы корреляций на шаге (5). Вместо того чтобы занулять собственные значения, которые меньше λ_+ , можно сделать их равными. Т. е. если j – первый индекс, т. ч. $\lambda_j < \lambda_+$, то положить $\forall s \geq j \lambda'_s = \frac{1}{N-j-1} \sum_{k=j}^N \lambda_k$. Далее получившуюся матрицу все равно надо отнормировать как в (6), но нормировка будет изменять матрицу не

так сильно, как если бы мы просто занулили собственные значения.

После удаления шума из матрицы корреляций, мы можем потерять часть полезной информации, но доля неслучайной информации в ней увеличится. Когда будем использовать матрицу корреляций в машинном обучении, будем не только удалять рыночную компоненту, но и шум.

2.2.4 Кластеризация матрицы корреляций

Матрица корреляций достаточно большая, нам хотелось бы уменьшить ее размер, сохранив важную информацию в ней. Основной причиной корреляции доходностей акций является рыночный фактор. После того как мы уберем его, корреляция данных активов будет связана с тем, насколько компании похожи друг на друга. Например, высокая корреляция будет у компаний в одной отрасли, и низкая у компаний в разных отраслях. В связи с этим возникает идея разбить все акции на кластеры.

Если у нас есть разбиение всех акций на кластеры, то в каждом кластере составим матрицу корреляций активов в нем (подматрица матрицы корреляций всех инструментов). Составим на ее основе портфель из акций только из этого кластера. Далее рассмотрим портфели из каждого кластера как новые активы. У этих "комбинированных" активов будет своя матрица корреляций (которая зависит от корреляционной матрицы всех инструментов и весов в кластерах). На основе корреляционной матрицы и доходности "комбинированных" активов составим портфель из кластеров. В финальный портфель каждая акция войдет с весом, равным произведению веса кластера на вес акции в кластере.

Такое пошаговое составление портфеля имеет ряд преимуществ. Во-первых, количество признаков снижается, т. к. теперь мы игнорируем попарные корреляции между всеми активами из разных кластеров, рассматривая только небольшую матрицу корреляций доходностей кластеров. Во-вторых, возникает возможность сначала обучить одну модель на каждом кластере отдельно, а затем другую модель на "комбинированных" активах. Однако в данном подходе важно следить за переобучением, поскольку в идеальном случае проводить кластеризацию, обучать модель на каждом кластере и обучать другую модель на инструментах, которые выдала предыдущая модель, надо на трёх разных выборках.

Для кластеризации можно использовать разные методы: k-means, DBSCAN, OPTICS и другие. Все эти методы требуют либо матрицы "похожести" объектов, либо матрицы "расстояния" между ними. Построить расстояние между активами естественно на основе матрицы корреляций, поскольку чем больше коррелированы активы, тем ближе они находятся. Лег-

ко показать, что если для любых двух последовательностей случайных величин $x_i, x_j \in \mathbb{R}^T$ посчитать их корреляции $\rho(x_i, x_j)$, то отображение

$$d: \mathbb{R}^T \times \mathbb{R}^T \rightarrow \mathbb{R}$$
$$d(x, y) = \sqrt{\frac{1}{2}(1 - \rho(x, y))} \quad (7)$$

будет являться метрикой. Заметим, что т. к. $\rho(x, y) \in [-1, 1]$, то $d(x, y) \in [0, 1]$.

Однако расстояние на основе корреляции между доходностями не учитывает нелинейные зависимости доходностей активов. Другой хорошей метрикой, которая учитывает любые зависимости, является метрика из теории информации, именуемая Variation of information⁴:

$$VI(X, Y) = H(X, Y) - I(X, Y),$$

где $H(X, Y)$ – совместная энтропия случайных величин X и Y , $H(X)$ – энтропия, $I(X, Y) = H(X) + H(Y) - H(X, Y)$ – их взаимная информация.

Отметим, что для использования этой метрики, когда мы не знаем реальное распределение, но имеем выборку, необходимо бинаризовать случайные величины. Т. е. нужно разбить диапазон значений на несколько отрезков, и считать значения в одном интервале за одно значение величины. Если взять слишком много отрезков, то пострадает оценка вероятности попадания в этот отрезок. Если взять слишком мало отрезков, то не будут учтены более сложные зависимости между величинами. Хорошее приближение для числа отрезков для оценки энтропии с точки зрения низкого смещения рассматривается в [9]. Оптимальное число отрезков для оценки совместной энтропии с точки зрения минимизации квадратичной ошибки описывается в [8].

Таким образом, используя метрики расстояния на основе корреляции или Variation of information, можно кластеризовать активы, и свести задачу к распределению весов внутри кластеров и между кластерами.

⁴https://en.wikipedia.org/wiki/Variation_of_information, дата обращения 10.02.2023

2.3 Предлагаемые методы обучения с подкреплением (Куцаков Александр)

2.3.1 Постановка задачи обучения с подкреплением

Отличие стандартных алгоритмов машинного обучения от обучения с подкреплением (Reinforcement Learning, RL) в том, что *агент* (роль которого исполняет модель) взаимодействует со *средой* – от неё агент получает результаты своих действий и награды. На Рисунке 2.1 показано⁵, как агент принимает ответы от среды, обрабатывает их, а затем действует в соответствии с заложенной в нём *политикой*. Одновременно с этим политика немного изменяется в соответствии с полученной наградой – так происходит обучение агента. В качестве среды может быть использована как реальная биржа, так и некоторый её симулятор.

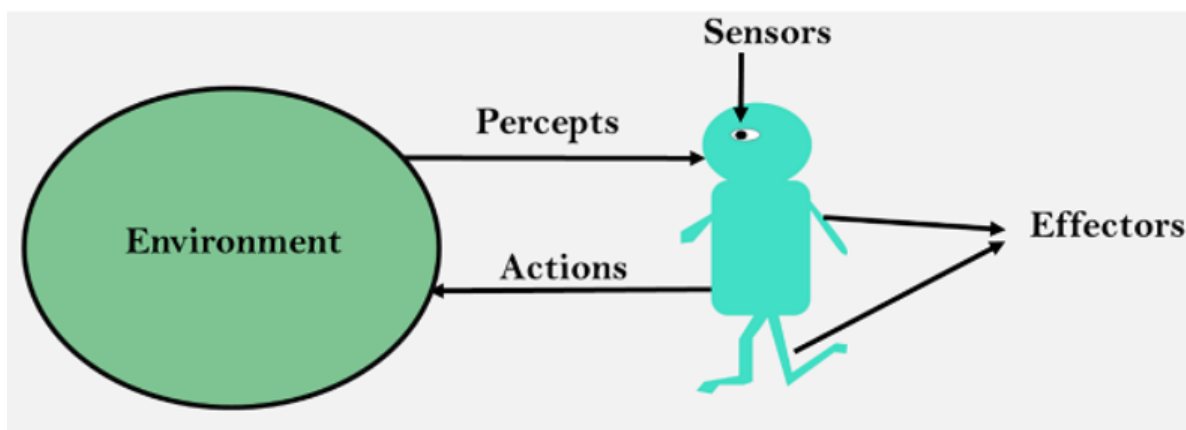


Рис. 2.1: Взаимодействие агента и среды.

Нашей задачей будет максимизация кумулятивной награды, полученной от среды. Для этого потребуется перейти к более формальной постановке.

2.3.2 Терминология и выбор методов

Итак, пусть \mathcal{A} – пространство действий, \mathcal{S} – пространство состояний. Они могут быть как дискретными (например, позиции и ходы в шахматах), так и непрерывными (в качестве примера можно рассмотреть бинарный поиск точки на прямой, $\mathcal{S} = \mathbb{R}$, $\mathcal{A} = \mathbb{R} \times \mathbb{R}$). Далее введём политику агента $\pi : \mathcal{A} \times \mathcal{S} \rightarrow [0, 1]$ – вероятностное распределение действий в каждом состоянии. Агент, находясь в состоянии $s_t = s \in \mathcal{S}$, выбирает действие $a_t = a \sim \pi(s)$, $a \in \mathcal{A}$. Далее среда семплирует следующее состояние s_{t+1} из $P_a(s, s') = P(s'|s, a)$ – распределение $s' \in \mathcal{S}$ при условии предыдущего состояния и действия, а также $r_{t+1} = r \sim Q(r|s, s', a)$ –

⁵<https://www.javatpoint.com/agents-in-ai>, дата обращения 12.02.2023

награду за переход. В итоге агент выбирает действие $a \in \mathcal{A}$, а получает в ответ (s', r) – следующее состояние и награду.

Далее введём кумулятивную награду, но не будем использовать просто сумму, поскольку для нас будущие награды являются менее ценными. Будем умножать их на коэффициент $\gamma \in [0, 1]$ (в нашем случае это соответствует инфляции, а теоретически улучшает сходимость – для $\gamma < 1 : \gamma^k \rightarrow 0, k \rightarrow +\infty$). Тогда ожидаемая кумулятивная награда в момент времени t

$$G_t = \mathbb{E} [r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \dots] = \mathbb{E} \left[\sum_{k=0}^{+\infty} \gamma^k r_{t+k} \right] \rightarrow \max_{\pi_\theta} \quad (8)$$

где θ – параметры модели, которые определяют политику. Максимизировать напрямую G_* неудобно, поэтому вводятся две вспомогательные функции $V_\pi(s)$ – value-функция, которая равна ожидаемому выигрышу, если мы начинаем в состоянии s и действуем согласно политике π , а также $Q_\pi(s, a)$ – то же самое, но из состояния s мы сначала совершаем действие a , а затем уже действуем по выбранной политике. Согласно уравнениям Беллмана⁶, в случае выбора *оптимальной* политики π^* для value-функции и Q-функции выполнено

$$V_*(s) = \max_{a \in \mathcal{A}} \sum_{s'} P(s'|a, s) (r(s, a) + \gamma V_*(s')) = \max_{a \in \mathcal{A}} Q_*(s, a) \quad (9)$$

$$Q_*(s, a) = \sum_{s'} P(s'|a, s) \left(r(s, a) + \gamma \max_{a'} Q_*(s', a') \right) \quad (10)$$

то есть наше действие в каждом состоянии не зависит от номера шага и от предыдущих состояний. Поэтому обучение с подкреплением реализует марковский процесс принятия решений⁷, что влечёт более простую работу с данными, но априори неполную информацию, которую можно компенсировать, если добавлять в состояние описанные выше технические индикаторы и прочие инструменты, которые добавляют информацию о предыдущих периодах.

Существует достаточно много алгоритмов обучения с подкреплением, которые основываются на оптимизации V, Q -функций (это логично, поскольку их вид очень похож на максимизируемую в (8) функцию кумулятивной награды). Их основная проблема заключается в том, что часто они не рассчитаны на большие пространства \mathcal{A} и \mathcal{S} . Если пространство состояний в играх (для которых в большинстве случаев применяется обучение с подкреплением) вполне может быть экспоненциально большим, то для каждого конкретного количество действий сильно ограничено (например, в шашках или шахматах ходов в данной позиции

⁶https://en.wikipedia.org/wiki/Bellman_equation, дата обращения 12.02.2023

⁷https://en.wikipedia.org/wiki/Markov_decision_process, дата обращения 12.02.2023

не так много, некоторые из них легко исключаются детерминированными алгоритмами). В случае же трейдинга одно действие может являться вектором целых чисел, показывающим, в каком количестве продавать/покупать каждый актив, поэтому действий в данном состоянии намного больше. Поэтому, предлагаемые методы будут направлены на оптимизацию политики напрямую, все они называются Policy Gradient методами.

2.3.3 REINFORCE

Пусть мы зафиксировали, что количество совершённых действий равно T , а их некоторая семплированная последовательность $[(A_1, R_1), \dots, (A_T, R_T)]$, R_i – полученные награды. Рассмотрим функционал

$$J(\pi_\theta) = \mathbb{E} \left[\sum_{t=1}^T R_t | \pi_\theta \right] \rightarrow \max_{\theta}$$

Можно заметить, что при $\gamma = 1$ это напоминает частичную сумму для куммулятивной награды. Чтобы максимизировать его, выпишем градиент

$$\nabla_{\theta} J(\pi_{\theta}) = \sum_{t=1}^T \nabla_{\theta} \log \pi_{\theta}(a_t | s_t) \cdot G_t \quad (11)$$

где G_t определено в (8). Поскольку матожидание мы не знаем, то предлагается следующий алгоритм для его оценки

- Фиксируем гиперпараметр N – размер выборки для семплирования и инициализируем θ (определяем как-то π_{θ}).
- Семплируем N траекторий, будем обозначать их τ_1, \dots, τ_N , используя политику π_{θ} .
- Для всех t для каждой τ считаем $R_t(\tau) = \sum_{\tilde{t} \geq t} \gamma^{\tilde{t}-t} r_{\tilde{t}}$. Это значение, которое мы используем для оценки Q_t , как увидим в дальнейшем.
- Используя полученные траектории, считаем оценку градиента

$$\nabla_{\theta} \tilde{J}(\pi_{\theta}) \stackrel{def}{=} \frac{1}{N} \sum_{\tau} \sum_{t \geq 0} \gamma^t \nabla_{\theta} \log \pi(a_t | s_t, \theta) \cdot R_t(\tau) \quad (12)$$

- Делаем шаг градиентного спуска, применяя полученную оценку.

Можно заметить схожесть оценки градиента (12) с (11), однако для самого градиента была фиксирована длина траектории T . На практике траектория, которую мы семплируем, может затянуться слишком сильно, поэтому длины траекторий из алгоритма имеет смысл

ограничить некоторой константой T и считать, что мы оцениваем именно (11). Если же длину не ограничивать, то истинная формула будет выглядеть так

$$\nabla_{\theta} J(\pi_{\theta}) = \mathbb{E}_{\tau \sim \pi_{\theta}} \sum_{t \geq 0} \gamma^t \nabla_{\theta} \log \pi_{\theta}(a_t | s_t, \theta) Q_{\pi_{\theta}}(s_t, a_t)$$

Формула выписана именно в таком виде, чтобы заметить, что в алгоритме оцениваются именно значения Q , которые мы не знаем в силу выбранного метода оптимизации политики. Несмотря на то, что данный метод достаточно простой и не требует сложного подбора гиперпараметров, у него есть достаточно понятный минус. Оценка неизвестной функции Q ($R_t(\tau)$ из (12)), считается по небольшому количеству траекторий, которых просто огромное количество из-за большого выбора действий в каждом состоянии. Потому она обладает большой дисперсией и сама оценка градиента наследует это свойство. Отсюда данный метод может крайне медленно сходиться и часто не доходит до оптимума. Чтобы уменьшить дисперсию, часто имеет смысл использовать в (12) вместо $R_t(\tau)$ выражение $R_t(\tau) - b(s_t)$ – если у нас уже имеется некоторый baseline в состоянии s_t , то, вычитая его, мы уменьшим разброс оценки.

2.3.4 Advantage Actor-Critic (A2C)

Для решения проблемы плохой оценки Q, V функций мы возвращаемся к идеям из Q-Learning [18], которые основываются на приведении функций V и Q к удовлетворяющим уравнениям Беллмана (9), (10). Однако всё это в предположении, что политика детерминированная, и все вероятности нулевые, а одна равна единице, то есть

$$V_*(s) = \max_{a \in \mathcal{A}} (r(s, a) + \gamma V_*(s')) = \max_a Q(s, a) \quad (13)$$

$$Q_*(s, a) = r(s, a) + \gamma \max_a Q(s', a) \quad (14)$$

здесь s' определяется однозначно действием в состоянии s , $a = \pi_*(s)$ для value-функции.

Идея actor-critic методов в использовании сразу двух нейронных сетей для симуляции политики (агент) и Q, V -функции (критик) соответственно. Пусть π_{θ} задаётся параметрически через θ , V_w через w . Тогда

$$J_{\pi}(\theta) = \log \pi_{\theta}(a | s, \theta) A(s, a) \rightarrow \max_{\theta} \quad (15)$$

где в качестве π_θ берётся агент, а также A – advantage-функция определена

$$A(s, a) = r(s, a) + \gamma V_w(s') - V_w(s) = Q(s, a) - V(s)$$

и равенство верно в предположении, что критик действует оптимально. Тогда в (15) для “хороших” действий вероятность будет выше, а для “плохих” – ниже. Чтобы добиться правильной оценки advantage-функции, для критика используется функция потерь, которая требует выполнение или хотя бы приближение (13)

$$J_V(w) = (r_t + \gamma V_w(s_{t+1}) - V_w(s_t))^2 \rightarrow \min_w$$

Чтобы обучение происходило на данных, близких к независимым и одинаково распределённым, используется специальный буффер R , в который добавляются наборы $(s_t, a_t, r_{t+1}, s_{t+1})$, где значение t не важно в силу того, как мы задаём модель. Из этого буффера семплируются наборы для дальнейших итераций обучения. Делается это для того, чтобы агент и критик не находились в одном месте среды, а учились одновременно на многих её частях, которые как-то перемешаны в буффере. Если же учиться последовательно и брать начальное состояние на следующей итерации такое же, как конечное на предыдущей, то мы можем оказаться в одном маленьком участке среды и переобучиться на него.

2.3.5 Deep Deterministic Policy Gradient

Другим методов с actor-critic подходом является DDPG (Deep Deterministic Policy Gradient) [11]. В нём предлагается использовать аналогичные функции, однако добавить ещё одну пару actor-critic, а также случайность при выборе действия. Моделируется она некоторым случайным процессом \mathcal{N} . Он может быть упомянутым выше процессом Орнштейна-Улебека, который хорошо описывает логику биржи, или любым другим шумом. Рассмотрим сначала сам алгоритм

- Фиксируем размер батча N и “скорость” обновления весов $\tau \in [0, 1]$.
- Инициализируем $\pi_\theta(s), Q_w(s, a)$ – первую пару actor-critic. Как можно видеть из обозначений, этот алгоритм концентрируется на Q -функции. Далее с теми же весами инициализируем $\pi_{\theta'}(s), Q_{w'}(s, a)$, то есть $\theta' = \theta, w' = w$. Затем инициализируем буффер R , описанный выше.
- Начнём внешний цикл по траекториям, на каждой итерации семплируем s_1 – начальное

состояние и обновляем \mathcal{N} – случайный процесс.

- Перейдём к внутреннему циклу по траектории $t \in \{1, \dots, T\}$, все дальнейшие пункты выполняются внутри него.
- Используя текущее состояние s_t , выбираем действие $a_t = \pi_\theta(s_t) + \mathcal{N}_t$, добавляя к нему шум из случайного процесса. Затем получаем от среды результат этого действия (r_{t+1}, s_{t+1}) , после чего сохраняем набор $(s_t, a_t, r_{t+1}, s_{t+1})$ в буффер R .
- Семплируем из буффера батч из наборов $(s_i, a_i, r_{i+1}, s_{i+1})$ размера N . Здесь i значит не подряд идущие числа, поскольку наборы между собой никак не связаны.
- Введём $y_i = r_i + \gamma Q_{w'}(s_{i+1}, \mu_{\theta'}(s_{i+1}))$ – даём действовать другой паре actor-critic.
- Обновляем критика Q_w , минимизируя MSE⁸

$$L(w) = \frac{1}{N} \sum_i (y_i - Q_w(s_i, a_i)) \rightarrow \min_w$$

- Обновляем агента π_θ , снова оценивая градиент средним

$$\nabla_\theta J(\theta) = \frac{1}{N} \sum_i \nabla_a Q_w(s, a) \Big|_{(s,a)=(s_i, \pi_\theta(s_i))} \cdot \nabla_\theta \pi_\theta(s) \Big|_{s_i}$$

- Мы обновили Q, π , осталось обновить веса другой пары

$$\theta' = \tau \cdot \theta + (1 - \tau) \cdot \theta', \quad w' = \tau \cdot w + (1 - \tau) \cdot w'$$

Теперь рассмотрим плюсы, которые есть в этом методе. Случайный процесс \mathcal{N} нужен для исследования среды: небольшая ошибка часто позволяет выйти из локального минимума, чтобы найти более оптимальное решение. Основным отличием является дополнение второй пары actor-critic $\pi_{\theta'}, Q_{w'}$, которые со “скоростью” τ двигаются за основной парой. Они являются аналогом ученика в этих двух парах, который при этом хранит больше информации о прошлом. Пока основная пара π_θ, Q_w движется по траектории и добавляет в буффер новые значения, другая пара благодаря подсчёту y_i привязывает её Q -функцию к своей через уравнение Беллмана (14), но с разными Q . Когда градиентный спуск начнёт замедляться, приближаясь к точке оптимума, параметры станут очень близки к друг другу и мы будем оптимизировать уже настоящее уравнение.

⁸https://en.wikipedia.org/wiki/Mean_squared_error, дата обращения 12.02.2023

3 Стохастические дифференциальные уравнения (Stochastic Differential Equations, SDE) (Курилов Олег)

3.1 Определение SDE

Рассмотрим задачу прогнозирования временного ряда. Порой может быть неинтересно точное значение процесса а лишь его распределение. Для анализа данной задачи удобно представлять временной ряд в виде непрерывного параметрически заданного случайного процесса S_t . Неформально говоря, случайный процесс это множество проиндексированных случайных величин, распределения которых могут зависеть друг от друга. Индекс t в дальнейшем будет обозначаться как время. Изменения этого процесса, обозначаемые как ΔS_t могут зависеть от следующих вещей:

- 1 Значения процесса в точке
- 2 Значения параметра t в момент прошлого и будущего
- 3 Какой-то стохастической составляющей. В дальнейшем в качестве данной составляющей будет использоваться Броуновское движение W_t , которое является естественным обобщением нормального распределения на случайный процесс.

Получаем в общем случае запись:

$$S_{t+\alpha} = S_t + \Delta(S_t, t, t + \alpha, W_{t+\alpha} - W_t)$$

Функция Δ может быть устроена очень сложно, поэтому естественно посмотреть на её поведение при $\alpha \rightarrow 0$.

Получается, что в некотором смысле хочется рассмотреть дифференциальное уравнение случайного процесса по времени. Понимать будем в том смысле, что в некоторых случаях можно взять интеграл от случайной функции по случайному процессу, именуемому обычно интегралом Ито. То есть имеет смысл такая запись:

$$G = \int S_t dW_t$$

Если определён интеграл, то можно записать стохастическое дифференциальное уравнение:

$$dS_t = a(S_t, t)dt + b(S_t, t)dW_t$$

То есть дифференциал случайного процесс S_t распался на детерминированную и стохастическую часть линейным образом. Теперь можно рассматривать конкретные примеры функций a и b и смотреть на эффект от них.

3.2 Примеры процессов и исследование их свойств

В данной секции рассматривается 2 примера с обоснованием их использования. Прежде всего стоит заметить, что процессы стохастическим образом зависят от Броуновского движения, так как в случайных процессах доказано, что все случайные процессы в среднем сходятся к нему, как и случайные функции в среднем сходятся к нормальному распределению.

Геометрическое Броуновское движение.

Данный процесс задаётся формулой:

$$dS_t = S_t(\mu dt + \sigma dW_t)$$

Если процесс задан в точке 0, то тогда его решением будет:

$$S_t = S_0 \exp\left(\left(\mu - \frac{\sigma^2}{2}\right)t + \sigma W_t\right)$$

Откуда видно, что процесс никогда не принимает значения меньше 0, что является несомненным плюсом для моделирования. Так же он отвечает за то, чтобы волатильность росла прямо пропорционально цене актива, что выполняется на практике.

Процесс Орнштейна-Уленбека.

Формула описания процесса следующая:

$$dS_t = \theta(\mu - S_t)dt + \sigma dW_t$$

Процесс Орнштейна-Уленбека моделирует возврат к среднему по уровню тренда, что также подмечено на практике. Решением данного процесса при $\mu = 0$ и заданной стартовой точки будет:

$$S_t = \frac{\sigma}{\sqrt{2\theta}} e^{-\theta t} W_{e^{2\theta t}}$$

Видно, что теперь процесс уже может принимать значения большие нуля, что решается с помощью правильного моделирования, описанного в статьях.

Также из интересного, что этот процесс получается за счёт ускорения Винервского движе-

ния. Как описано в учебниках, Винеровский процесс за бесконечное время с вероятностью 1 вернется в точку 0, а за счёт ускорения времени происходит как бы ускорение возврата в 0, что и позволяет моделировать реальное движение цен.

Также данный процесс был упомянут выше в контексте задач RL, что только подтверждает его актуальность и важность.

Если рассматривать какую-то взвешенную комбинацию этих процессов, то может получиться довольно хорошее приближение реальных цен.

3.3 Оценка параметров

Теперь будет рассматриваться вопрос, как же непосредственно оценить параметры заданного процесса. Различные методы описаны в различной классической литературе по случайным процессам, в данной главе рассматривается основной вариант, а именно метод максимального правдоподобия. Строится он из факта, что ΔW_t независим с W_t .

Тогда пусть дана выборка $S_{t_1}, S_{t_2}, \dots, S_{t_n}$.

Посмотрим на их функцию вида :

$$f(\Delta S_t) = g(\dots) \cdot \Delta W_t + h(\dots)$$

То есть смысл заключается в вынесении в представлении линейной комбинации с приращением. В примерах выше так сделать можно. Затем оценим параметры функции путем максимизации вероятности того, что произошло такое событие. В мягких предположениях данный ответ будет сходиться к верному по асимптотике.

Получили метод, которым можно оценивать параметры, чтобы точнее моделировать процессы, либо которые можно просто потом будет передать на вход какой-нибудь модели для более точного прогноза цены.

3.4 Применение SDE

3.4.1 Расчёт справедливой цены

Благодаря введённым выше процессам можно выводить справедливую цену для производных финансовых инструментов, считая их как функция от случайного процесса.

Справедливая цена это такая, когда игроки рынка в среднем заработают на инструменте столько, сколько бы они заработали на безрисковой кредитной ставке.

Классически динамика изменения цены берется из примеров, описанных выше, но можно

рассматривать и более сложные примеры. Например, в Формуле Блэка-Шоулза рассматривается решение процесса для опциона с некоторыми предпосылками, которые не выполняются на практике. Если менять предпосылки или в принципе рассматривать не один опцион, а несколько, то будут получаться разные решения, некоторые из которых на практике будут лучше оценивать цену.

3.4.2 Игра на волатильности

Также если положить найденную справедливую цену правильной, то получается, что остаётся играть лишь на ожиданиях об изменении цены, что называется игрой на волатильность.

Банальный пример заключается в том, что если рынок ожидает, что мы ожидаем сильного падения или взлета цены после их будущей презентации. Рынок может считать, что акция изменится не так сильно, так как показывать ничего не будут, тогда можно будет сыграть на этом с помощью финансовых инструментов. Оценка волатильности очень тонкая деталь, которую можно производить огромным числом способов с помощью оценок на случайные процессы.

3.4.3 Применения для генерации данных

Также случайные процессы нашли своё применение в методах генерации данных. Так передовые диффузионные модели работают на марковских процессах, ровно, как и методы взаимодействия частиц.

В рамках финансовых рынков методы ещё не были столь сильно исследованы, что порождает возможность применить их и получить новые результаты.

4 Модели на классификации свеч (Кузьмин Максим)

Людьми часто используется классический технический анализ, упомянутый в 1 части отчета. Поэтому естественным решением было рассмотреть стратегии, являющиеся автоматизацией технического анализа.

Технический анализ, в свою очередь, использует большое количество данных о торгах из прошлого. Отдельной компонентой технического анализа является свечной анализ.

Свечной анализ - предсказание изменения цены актива посредством рассмотрения свеч: агрегированных за какой-либо период данных о торгах определенным активом. Свечи

всегда содержат информацию о ценах открытия, закрытия, максимальных и минимальных ценах актива, и часто содержат также и информацию об объеме торгов.

4.1 Выбор бумаги для построения стратегии

Базовое предположение технического анализа - то, что на цену бумаги в настоящем оказывает влияние только цена бумаги в прошлом. Это предположение для разных бумаг верно в разной степени. Другие факторы, оказывающие влияние на цену актива - это, с одной стороны, события(новостной фон) и, с другой стороны, действия участников рынка(примером, когда этот фактор становится определяющим, может быть шорт-сквиз GameStop в январе 2021 года). Несложно понять, что новостной фон сильнее всего влияет на крупные компании, поскольку они чаще попадают в новости как минимум в силу размера и доли в экономике страны. Отдельные участники рынка, наоборот, сильнее влияют на компании с небольшим объемом торгов. Из этого следует, что необходимо брать акции компаний со средне-высоким объемом торгов: компания с большим объемом торгов может появляться в новостях достаточно редко(примером может быть Белуга АО), но котировками компаний с небольшим объемом торгов сможет управлять кто угодно с достаточными ресурсами. Поэтому для стратегии были выбраны акции "Магнита": компания входит в Индекс Мосбиржи, ее объем торгов - сотни миллионов рублей в день, и при этом компания редко появляется в новостях.

4.2 Построение стратегии и предобработка данных

Любая стратегия должна принимать решение продавать или покупать актив в том или ином количестве. Как известно из курса машинного обучения, задача машинного обучения с ответом в конечном множестве называется задачей классификации. Что же будет классифицироваться при построении стратегии, работающей со свечами?

Данные, то есть свечи, поступают к нам в неразмеченном виде. Логично было бы разметить их по решению, которое по ним должна принять модель(продавать или покупать бумаги). Легко понять, что модели выгодно покупать бумаги, если до конца дня и после покупки был момент времени, когда она могла бы продать их с прибылью с учетом комиссии выше заданного значения дневной доходности.

После разметки данных можно решать задачу классификации. Простейшим вариантом было бы использовать только последнюю свечу, однако даже классический теханализ, ограниченный возможностями человека в обработке информации, так обычно не делает и

рассматривает паттерны - несколько идущих друг за другом свеч. Таким образом, перед передачей в алгоритм признаки, соответствующие свечам, необходимо слепить в один. Оптимальное количество свеч в паттерне будет гиперпараметром стратегии.

После построения паттернов уже можно обучать любой желаемый алгоритм классификации для построения собственно стратегии. Мы обучим и будем сравнивать SVM, логистическую регрессию и градиентный бустинг.

4.3 Метрики оценки качества классификации

Метрика точности классификации (доли корректно предсказанных на тестовой выборке объектов) является самой простой метрикой классификации. Однако она неприменима для оценки качества классификации свечных паттернов из-за того, что эта классификация очень несбалансированная: при норме дневной доходности 0.5% к положительному классу относится около 20% объектов.

Метрика AUC-PR, обычно используемая в задачах несбалансированной классификации, также не подходит для оценки качества стратегий, классифицирующих свечные паттерны, из-за разной цены ошибок первого и второго рода при классификации паттернов: ошибка первого рода приведет к тому, что модель не совершит потенциально выгодную сделку, то есть доходность будет равна 0, а ошибка второго рода приведет к тому, что модель будет вынуждена закрывать сделку по стоп-лоссу или в конце дня, то есть доходность будет отрицательной.

Таким образом, лучшей метрикой является вычисленная по матрице частоты ошибок и верных предсказаний прибыль.

5 Коннектор к бирже

После получения данных и обучения модели возникает логичное желание эту модель испытать в реальной торговле. Для доступа к торговле на бирже, нужен брокер. Было рассмотрено большое количество брокеров, из которых наименьшие комиссии оказались у Tinkoff и ITInvest. API для работы с заявками у Tinkoff открыто и хорошо задокументировано, а у ITInvest – закрыто, и на момент написания данного отчета мы не смогли получить от них документацию по API. Поэтому, для реализации стратегий был выбран брокер Tinkoff.

5.1 Задачи коннектора (Кузьмин Максим)

Коннектор является посредником между биржей и стратегией. Из этого следуют его фундаментальные задачи: прием данных торгов с биржи и выставление заявок от стратегии. Прием данных с биржи для разных стратегий должен осуществляться по-разному: стратегии, основанные на оптимизации портфеля, вполне могут принимать данные с фиксированной небольшой частотой (порядка раза в несколько часов), а стратегии, ведущие активную торговлю, принимают данные о сделках в режиме реального времени, что обуславливает необходимость работы различных частей коннектора в асинхронном режиме. Выставление заявок, в свою очередь, требует также и отслеживания их состояния. Таким образом, коннектор должен:

- Загружать исторические рыночные данные
- Принимать данные торгов с биржи
- Выставлять заявки согласно сигналам стратегии
- Отслеживать статус исполнения заявок

5.2 Архитектура (Казадаев Максим)

Ссылка на текущую реализацию: <https://github.com/Maksim787/neurotrader>

Всю архитектуру коннектора можно разделить на 5 частей:

1 InfoManager

- Получает качественную информацию об инструментах: шаг цены, лотность, минимальные объемы и прочие постоянные в течение дня характеристики
- Получает информацию о времени торгов для каждого инструмента

2 MarketManager

- Загружает исторические рыночные данные: свечи, сделки и другие.
- Дополняет загруженные данные в реальном времени и получает новые данные, для которых нет истории, например, книги ордеров

3 UserManager

- Выставляет, изменяет и отменяет ордера, которые требует стратегия

- Отслеживает статусы исполнения ордеров
- Поддерживает текущие позиции и зарезервированные в заявках средства

4 Strategy

- Получает исторические данные от MarketManager
- Получает в реальном времени рыночные данные от MarketManager и изменения в позициях от UserManager
- Выставляет заявки через UserManager

5 Runner

- Инициализирует, запускает и поддерживает работу менеджеров и стратегии. Контролирует цикл событий.

5.2.1 InfoManager

InfoManager должен получать постоянные в течение дня общие характеристики инструментов: название, тикер (идентификатор инструмента), минимальный шаг цены и лотность. Т. к. режимы торгов для различных инструментов иногда меняются, он получает информацию о режиме торгов для всех активов. Есть период времени, когда биржа не работает (большинство инструментов не торгуется), поэтому перед каждым открытием торгов InfoManager должен получать заново всю информацию по инструментам и проверять, что ничего не изменилось. Если что-то поменялось, то человек, следящий за стратегией, должен принять решение о том, целесообразно ли запускать ее, или стоит что-то изменить. Также, он будет контролировать настройки аккаунта, т. е. комиссии и доступность торговли по выбранным инструментам.

Для того чтобы хранить инструменты, будет использоваться контейнер InstrumentInfo. Для каждого инструмента будет ровно один экземпляр этого класса, т. е. все его упоминания в коде будут ссылаться на один и тот же объект. Везде, где будет требоваться указать инструмент, будет передаваться именно ссылка на объект этого класса. Это позволит функциям, которые работают с инструментом, иметь полный доступ к информации о нем.

5.2.2 MarketManager

MarketManager должен уметь загружать и сохранять локально пропущенные рыночные данные. Сохранять необходимо, чтобы при следующих запусках не тратить время на

повторную загрузку истории. После подписки на рыночные данные MarketManager должен синхронизировать поступающие данные с историей. Т. е. исторические данные будут загружаться не ровно до текущего момента, а до момента, когда мы получили первую информацию из стрима.

MarketManager должен следить за тем, насколько велика задержка между получением данных, и тем временем, когда они были отправлены. Дополнительно, он должен сохранять все получаемые события вместе с временем получения. Т. е. записывать в базу данных, чтобы после работы стратегии можно было проанализировать ее действия в случае непредвиденных ситуаций или ошибок.

5.2.3 UserManager

UserManager является самой сложной частью инфраструктуры, поскольку он должен решать проблему отличия информации у нас и той, что есть на самом деле на бирже. Данная проблема возникает из-за задержек между реальными событиями в системе биржи и получением этих событий нами. Например, мы можем пытаться отменить заявку, которая уже была исполнена.

Существуют два типа ордеров, которые мы можем выставить через API. Первый – рыночный, мы покупаем или продаем фиксированный объем по лучшей цене, которую готовы предложить остальные участники торгов в стакане. Второй – лимитный, мы покупаем или продаем фиксированный объем по фиксированной цене. Если данная заявка может быть удовлетворена за счет противоположной заявки в стакане по более выгодной для нас цене, она исполняется как рыночная. Иначе, она попадает в стакан и ждет, пока не исполнится против рыночной заявки другого участника торгов.

Для хранения ордеров имеется специальный контейнер UserOrder. У него есть несколько постоянных полей: инструмент, тип ордера, направление (покупка или продажа), выставленный объем и цена. Также есть изменяющиеся со временем величины: исполненный объем и статус ордера. Непостоянные величины хранятся в списке из контейнеров UserOrderState внутри UserOrder. Т. е. при каждом их изменении, создается новый объект UserOrderState, который добавляется в историю. Благодаря этому, всегда можно проследить всю историю ордера.

Проблема различного состояния у нас и на бирже решается с помощью дополнительных статусов ордеров. Ордер может иметь один из следующих статусов: PENDING_OPEN, PENDING_CANCEL, PENDING_MODIFY, OPEN, PART_FILL, FILL, CANCEL, REJECT.

Статусы с префиксом PENDING означают, что мы отправили запрос за выставление, отмену или изменение ордера. Они позволяют стратегии понять, что данные ордера могли еще не дойти до биржи, и что работать с ними надо аккуратно. Для рыночных ордеров после статуса PENDING_OPEN идет либо статус FILL, означающий, что ордер исполнен, либо REJECT, означающий, что биржа по какой-то причине не смогла обработать заявку. Для лимитных ордеров доступны действия отмены и изменения ордера. Если стратегия принимает такое действие, ордеру дается соответствующий статус с префиксом PENDING. В случае отмены ордера статус сменится на CANCEL в случае успеха, либо на FILL в случае, если наше сообщение дошло позже, чем заявка исполнилась. Аналогично в случае изменения ордера. Подробная схема переходов статусов представлена на Рисунке 5.1.

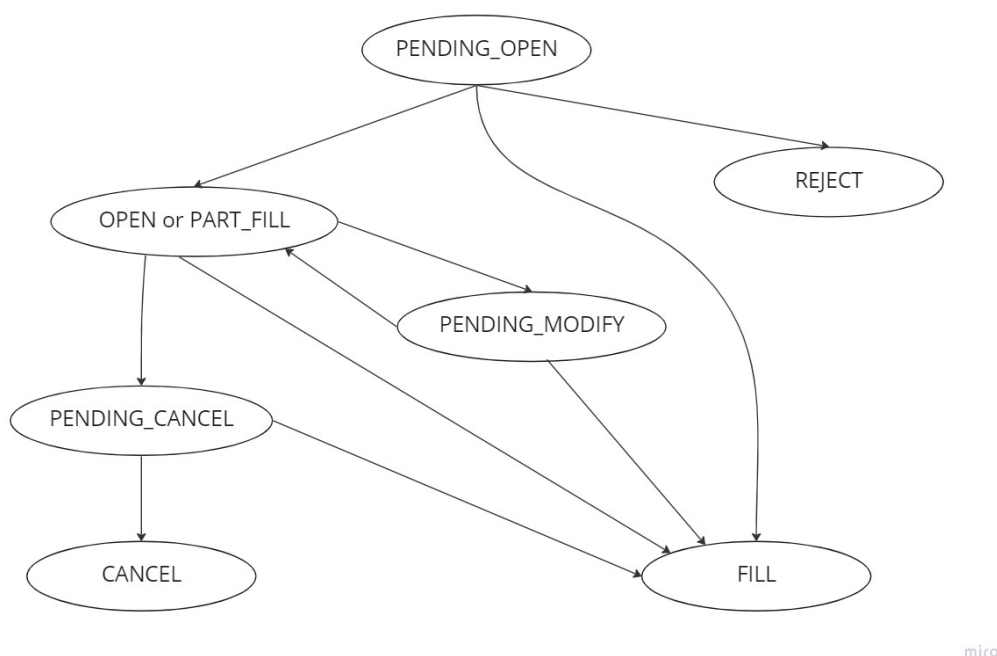


Рис. 5.1: Граф переходов статусов ордеров в UserOrder.

В задачи UserManager дополнительно входит поддержание текущих позиций. Во-первых, он считает количество валюты и разных активов на счете. Во-вторых, он считает количество заблокированных активов. Актив считается заблокированным, если он участвует в лимитном ордере. Например, если мы выставили лимитную заявку на некоторое число рублей, то мы не сможем использовать эти средства, пока не отменим заявку.

У UserManager будут неблокирующие методы `new_order(params)`, `cancel_order(order)` и `modify_order(order, new_params)`, о которых будет рассказано подробнее в следующих пунктах.

5.2.4 Strategy

Любая стратегия наследуется от абстрактного класса стратегии и реализует его группы методов:

1 Управление состоянием

- `subscribe()`. Вызывается в самом начале для того, чтобы попросить `MarketManager` загрузить необходимые данные, а также открыть все необходимые соединения для получения рыночной информации. В этом же методе можно указать, за сколько минут до окончания торгов необходимо вызывать метод `on_end()`.
- `on_start()`. Вызывается тогда, когда `MarketManager` загрузил нужные данные и получил хотя бы одно актуальное состояние данных из каждого стрима. В этом методе гарантируется, что `UserManager` получил наши позиции и открыл все необходимые соединения. В данном методе стратегия может сделать первичную инициализацию своих структур, а также выставить первые заявки.
- `on_end()`. Вызывается за некоторое время до окончания торгов, чтобы стратегия могла либо начать закрывать позиции, либо принять решение об их переносе на следующий день.

2 Получение рыночных данных

- `on_order_book_update(order_book)`, `on_market_trade(trade)` и другие. Используются для уведомления стратегии о поступлении новой книги ордеров, новых сделок, формировании новой свечи и других данных. Из переданной структуры стратегия может извлечь нужную информацию и принять какое-то решение.

3 Получение состояния ордеров

- `on_order_event(order)`. Используется для уведомления стратегии о том, что некоторый ордер реально выставлен, реально изменен, или он был частично или полностью исполнен.
- `on_order_error(order, error)`. Используется для уведомления стратегии о том, что произошла ошибка, связанная с ордером. Например, ошибки при модификации или отмене несуществующего ордера.

В любом методе, кроме `subscribe()`, стратегия имеет доступ к различным структурам из `MarketManager` и `UserManager`. Например, из `MarketManager` можно достать ссылку

на книгу ордеров, список исторических свечей, последовательность сделок и другие рыночные данные. Из `UserManager` можно достать позиции: баланс и заблокированный объем для каждого инструмента, а также список исполненных и открытых ордеров. Также, в любом методе, кроме `subscribe()`, можно вызвать соответствующий метод у `UserManager` и выставить новую заявку, изменить или отменить существующий ордер. Отметим, что эти методы не блокируют исполнение, запросы будут отправлены, когда стратегия выйдет из `callback`-а.

Предложенная система `callback`-ов позволяет немедленно уведомлять стратегию о всех нужных событиях. Структуры, поддерживаемые `MarketManager` и `UserManager`, предоставляют полную информацию о том, что происходит на рынке, и что происходит с нашими позициями и заявками. Неблокирующие методы управления ордерами и система статусов ордеров позволяет решать возникающие проблемы на уровне стратегии, давая возможность правильно обработать ошибку.

6 Выполненные задачи и план дальнейшей работы

Задача	Исполнитель	Срок
Предварительная обработка данных		
Найти источник данных по акциям с Московской биржи, выгрузить их: свечи, сделки, книги ордеров	Куцаков Александр	Выполнено
Найти источник данных по акциям других стран, выгрузить их: свечи, сделки, книги ордеров	Куцаков Александр	01.03.2023
Стандартизовать данные, создать API для их быстрого получения (использование базы данных)	Куцаков Александр	10.03.2023
Разметить данные: сгенерировать отрезки и признаки на этих отрезках	Куцаков Александр	15.03.2023
Классический подход к оптимизации портфеля		
Оптимизация по Марковицу	Казадаев Максим	01.03.2023
Удаление шума	Казадаев Максим	05.03.2023
Удаление рыночной компоненты	Казадаев Максим	05.03.2023
Кластеризация матрицы корреляций	Казадаев Максим	10.03.2023
Генерация признаков на основе матрицы корреляций	Казадаев Максим	15.03.2023
Обучение с подкреплением для оптимизации портфеля		
Написание среды	Казадаев Максим и Куцаков Александр	15.03.2023
Подбор архитектуры сети, оптимально учитывающей признаки	Казадаев Максим и Куцаков Александр	20.03.2023
Написание детерминированного baseline-метода для сравнения с более продвинутыми	Куцаков Александр	25.03.2023
Имплементация REINFORCE	Куцаков Александр	01.04.2023
Имплементация Advantage Actor-Critic метода	Казадаев Максим и Куцаков Александр	15.04.2023
Имплементация DDPG метода	Казадаев Максим и Куцаков Александр	30.04.2023
Поиск более новых и продвинутых подходов	Казадаев Максим и Куцаков Александр	30.05.2023

Задача	Исполнитель	Срок
Классификация свечей для торговли одним инструментом		
Собрать исторические данные	Куцаков Александр	Выполнено
Обучить базовые алгоритмы	Максим Кузьмин	Выполнено
Адаптировать стратегию под коннектор	Максим Кузьмин	15.03.2023
Провести сравнительное тестирование разных алгоритмов в реальных условиях	Максим Кузьмин	30.03.2023
Оптимизировать гиперпараметры	Максим Кузьмин	15.04.2023
Реализация стратегий, основанных на справедливой цене		
Изучить пререквизиты в виде экономики Блэка-Шоулза	Курилов Олег	Выполнено
Реализовать расчёт стратегий, основанных на этой модели	Курилов Олег	20.04.2023
Тестирование стратегий для выявления эффективности	Курилов Олег	10.05.2023
Генератор биржевых данных		
Изучить марковские процессы	Курилов Олег	Выполнено
Изучить основные методы моделирования, построенные с помощью Марковских цепей (МСМС алгоритмы и диффузионные модели)	Курилов Олег	Выполнено
Применить алгоритмы МСМС для генерации данных	Курилов Олег	01.03.2023
Применить диффузионные модели для генерации данных	Курилов Олег	15.03.2023
Тестирование алгоритмов для выявления эффективности	Курилов Олег	01.04.2023
Коннектор к Tinkoff Invest API		
Написать минимальную рабочую версию коннектора	Казадаев Максим	Выполнено
Тестирование коннектора на искусственных стратегиях	Казадаев Максим	15.03.2023
Оптимизация частых операций в стратегии, упрощение API стратегии, обработка ошибок	Казадаев Максим	30.03.2023
Написание и запуск простейших стратегий	Казадаев Максим	15.04.2023
Написание и запуск разработанных стратегий	Все	01.05.2023

Список литературы

- [1] Ernie Chan. *Algorithmic trading: winning strategies and their rationale*. Т. 625. John Wiley & Sons, 2013.
- [2] Lin Chen и Qiang Gao. “Application of deep reinforcement learning on automated stock trading”. В: *2019 IEEE 10th International Conference on Software Engineering and Service Science (ICSESS)*. IEEE. 2019, с. 29—33.
- [3] Gil Cohen и Elinor Cabiri. “Can technical oscillators outperform the buy and hold strategy?”. В: *Applied Economics* 47.30 (2015), с. 3189—3197.
- [4] Marcos Lopez De Prado. *Advances in financial machine learning*. John Wiley & Sons, 2018. Гл. 2.3.
- [5] Yue Deng и др. “Deep direct reinforcement learning for financial signal representation and trading”. В: *IEEE transactions on neural networks and learning systems* 28.3 (2016), с. 653—664.
- [6] Jiali Fang, Yafeng Qin и Ben Jacobsen. “Technical market indicators: An overview”. В: *Journal of Behavioral and Experimental Finance* 4 (2014), с. 25—56. ISSN: 2214-6350. DOI: <https://doi.org/10.1016/j.jbef.2014.09.001>. URL: <https://www.sciencedirect.com/science/article/pii/S2214635014000495>.
- [7] Xin Guo и др. “Quantitative trading: algorithms, analytics, data, models, optimization”. В: Chapman и Hall/CRC, 2017. Гл. 1.4.
- [8] Abdenour Hacine-Gharbi и Philippe Ravier. “A binning formula of bi-histogram for joint entropy estimation using mean square error minimization”. В: *Pattern Recognition Letters* 101 (2018), с. 21—28.
- [9] Abdenour Hacine-Gharbi и др. “Low bias histogram-based estimation of mutual information for feature selection”. В: *Pattern recognition letters* 33.10 (2012), с. 1302—1308.
- [10] Chien Yi Huang. “Financial trading as a game: A deep reinforcement learning approach”. В: *arXiv preprint arXiv:1807.02787* (2018).
- [11] Timothy P Lillicrap и др. “Continuous control with deep reinforcement learning”. В: *arXiv preprint arXiv:1509.02971* (2015).
- [12] Elias Little. “The Marchenko-Pastur Distribution: From Understanding to Application”. В: (2020).

- [13] Volodymyr Mnih и др. “Playing atari with deep reinforcement learning”. В: *arXiv preprint arXiv:1312.5602* (2013).
- [14] Darie Moldovan, Mircea Moca и Stefan Nitchi. “A stock trading algorithm model proposal, based on technical indicators signals”. В: *Informatica Economica* 15.1 (2011), с. 183.
- [15] Marcos M López de Prado. “Machine learning for asset managers”. В: Cambridge University Press, 2020. Гл. 2.6.
- [16] Janick Rohrbach, Silvan Suremann и Joerg Osterrieder. “Momentum and trend following trading strategies for currencies revisited-combining academia and industry”. В: *Available at SSRN 2949379* (2017).
- [17] Jack D Schwager. *A Complete Guide to the Futures Market: Technical Analysis, Trading Systems, Fundamental Analysis, Options, Spreads, and Trading Principles*. John Wiley & Sons, 2017.
- [18] Christopher JCH Watkins и Peter Dayan. “Q-learning”. В: *Machine learning* 8 (1992), с. 279—292.
- [19] Graeme West. “An introduction to Modern Portfolio Theory: Markowitz, CAP M, APT and Black-Litterman”. В: *Parktown North: Financial Modelling Agency* (2006).
- [20] Hongyang Yang и др. “Deep reinforcement learning for automated stock trading: An ensemble strategy”. В: *Proceedings of the first ACM international conference on AI in finance*. 2020, с. 1—8.
- [21] Zihao Zhang, Stefan Zohren и Stephen Roberts. “Deep reinforcement learning for trading”. В: *The Journal of Financial Data Science* 2.2 (2020), с. 25—40.