Федеральное государственное образовательное бюджетное учреждение  
 высшего профессионального образования  
 «Финансовый университет при Правительстве Российской Федерации»

Департамент анализа данных, принятия решений и финансовых технологий

Курсовая работа

на тему

**«Проверка гипотезы** **о равенстве дисперсий логарифмической доходности индекса фондового рынка и входящих в его состав акций»**

Вид данных для исследования:

«**Котировки акций компаний, входящих в индекс ММВБ финансового сектора**»

Выполнил:

студент группы ПМ20-4,

Есаков В.А.

Научный руководитель:

доцент, к.э.н.

Игудесман К.Б.

Москва

2022

**План работы**

[I. Введение 3](#_Toc101928955)

[II. Основные теоретически положения 5](#_Toc101928956)

[1) Математическая статистика 5](#_Toc101928957)

[2) Статистическая гипотеза 5](#_Toc101928958)

[3) Первый и второй род ошибок 6](#_Toc101928959)

[4) P-значения 6](#_Toc101928960)

[5) Логарифмическая доходность 7](#_Toc101928961)

[6) Критерий Колмогорова 8](#_Toc101928962)

[7) Критерий Фишера 10](#_Toc101928963)

[III. Практическая работа 12](#_Toc101928964)

[1) Предварительная обработка и аналитика данных 12](#_Toc101928965)

[2) Проверка гипотезы на сгенерированных данных 17](#_Toc101928966)

[3) Проверка правильности гипотезы на реальной статистике 18](#_Toc101928967)

[4) Другие гипотезы и оценка мощности критерия 21](#_Toc101928968)

[IV. Заключение 23](#_Toc101928969)

[V. Список использованных источников 24](#_Toc101928970)

[Приложения 25](#_Toc101928971)

[Приложение 1. Технические характеристики персонального ноутбука 25](#_Toc101928972)

[Приложение 2. Список файлов 26](#_Toc101928973)

[Приложение 3. 27](#_Toc101928974)

# **I. Введение**

В качестве задачи для исследования в курсовой работе я проведу аналитическую проверку правильности следующей гипотезы: дисперсия логарифмической доходности фондового рынка равна дисперсиям входящих в его состав акций. Для подтверждения справедливости гипотезы будет применяться простой и распространённый способ сравнения дисперсий – критерий Фишера.

Предметом исследования я выбрал данные котировок акций с фондовой биржи из одного индекса. В данном случае, индекс финансового сектора (MOEXFN), являющийся частью Московской биржи (ММВБ). Рассматриваемый в работе временной отрезок – с 1 января 2016 года по 31 декабря 2021 года.

Индекс ММВБ, экспертами указывается, как один из важнейших показателей экономики, так как он взвешен по рыночной капитализации. Является ценовым композитным фондовым индексом и включает в себя пакеты акций крупнейших российских эмитентов. Показателем их значимости является, главным образом, принадлежность к ликвидным и активно расширяющимся сферам жизни : финансовый сектор, сельское хозяйство, промышленность, транспортные сети, здравоохранение.

Выбранный для анализа индекс (MOEXFN – финансовый сектор) состоит из банковских компаний и других крупных организаций, заведующих финансами. Они предоставляют различные услуги по управлению, хранению деньгами, их инвестициями или займами. В данных на апрель 2022 года, частью индекса являются: Сбербанк (один из крупнейших банков России), имеет обычный и привилегированный пакет акций, QIWI (российская компания, специализирующаяся на виртуальных платёжных сервисах), ВТБ, Тинькофф, банк Санкт-Петербурга, Московский кредитный банк (крупные российские банки, предоставляющие различные услуги по хранению активов, депозитов, получению займов, инвестициям в различные ценные бумаги), RENI (компания, предоставляющая услуги страхования по любым имущественным, здравоохранительным и другим вопросам), SFI (публичное акционерное агентство, занимающееся инвестиционными управлением), индекс Московской биржи (Московская биржа занимается проведением операций по ценным бумагам, задаёт основной инвестиционный курс для всего российского рынка, является общим финансовым регулятором, поэтому является неотъемлемой частью финансового сектора экономики).

Отличительной особенностью данной работы с технической точки зрения будет проверка гипотезы с помощью языка программирования Python (версии 3.0 и новее). Это позволит действительно разумно использовать ресурсы для анализа большого объёма данных. Дополнительно будет использоваться среда разработки Jupiter Notebook, поддерживающая множество гибких способов визуализации данных в интерактивном формате, в частности : графики, гистограммы, формулы, табличные структуры.

Выбор язык программирования Python позволит использовать вычислительные мощности компьютера и современных программ для анализа данных, что является актуальным средством для исследования данной гипотезы. Выбор финансового сектора обусловлен его ключевым влиянием на всю экономику и авторитет всего государства в целом, который будет актуален всегда при существующей модели функционирования рынка.

В качестве итогов проведения данной работы получится установить наличие взаимосвязи между изменениями логарифмической доходности фондового рынка и изменениями логарифмической доходности входящих в состав индекса акций, а значит и их биржевых котировок, как следствие.

# **I. Основные теоретически положения**

## **1) Математическая статистика**

Предмет высшего образования – теория вероятностей и математическая статистика – включает в себя этот раздел математики и представляет собой совокупность множества более узких дисциплин, таких как финансовая, социальная, экономическая или любая другая статистическая аналитика сферы жизни общества. Основная задача дисциплины – обоснование теоретической базы всех этих конкретных дисциплин.

Создание единой методики и алгоритма исследования задачи путём фундаментальных и прикладных утверждений определяет цель математической статистики.

## **2) Статистическая гипотеза**

Такой гипотезой является любого рода утверждение, закрепляющее за ситуацией определённые свойства : параметры внутреннего распределения, о соотношениях между случайными величинами и так далее. При этом если достоверно известно, что предположение о характере генерального распределения рассчитано до конечного числа параметров, то такую гипотезу можно считать параметрической.

Введём обозначения - H0 (основная) и H1(дополнительная) – две статистические гипотезы, которые являются взаимоисключающими. Далее всегда будем считать, что у нас есть базисное утверждение о справедливости одной из гипотез. Соответственно, если гипотеза H0 подтвердилась, то H1 автоматически была опровергнута и наоборот.

Статистикой критерия называют правило, при котором отрицание гипотезы H0 и наличие некоторой выборки принадлежащей непустой области К, и таким же образом в обратную сторону гипотеза H0 подтверждается, а выборка не входит в некоторую область К.   
В этом случае К принято называть областью допустимых значений (альтернативное название – область принятия гипотезы), при обратной ситуации (когда в какой-то зоне отрицается принятие гипотезы H0) её называют критической областью, как правило задаваемую через неравенства:

Изображение выглядит как текст

Автоматически созданное описание

[6]

## **3) Первый и второй род ошибок**

Два различных вида ошибок могут возникнуть при статистических вычислениях. I род – гипотеза H0 отвергается, хотя в действительности является верной. Ошибкой же II-ого рода является ситуация, в которой отрицается гипотеза H1 , являющаяся истинной.

Ошибка первого рода является мерой значимости критерия (обозначается как α), а ошибка второго рода задаёт формулу для мощности критерия. Если принять за β вероятность ошибки, то искомое значение будет равно 1 – β.

Отклонение или принятие основной гипотезы опирается на оценку уровня значимости критерия. Так проверяется статистическая гипотеза для фиксированного значения уровня значимости. Если его изменить, значения придётся пересчитывать, в том числе критическую оценку [1]

## **4) P-значения**

Данная характеристика влияет на принятие основной гипотезы для любого уровня значимости, что позволяет не делать лишних операций по вычислению критических значений.

, где является фиксированным выбором из случайного набора данных, a – уровень значимости, для которых принята гипотеза H0

И напротив, , для всех а, при которых гипотеза отвергается.   
PV(x) является P-значением статистического критерия.

Теперь отдельно разберём случай, при котором .

Пусть - уравнение произвольной убывающей функции

 Также из этого равенства следует другое, утверждающее, что основная гипотеза принимается. Так получается вывод общепринятой формулы, которую чаще всего применяют при расчёте р-значений:  
Изображение выглядит как текст

Автоматически созданное описание

[1]

По таблице р-значений, зная число степеней свободы эксперимента в соответствующей строке находим большее значение, чем значений хи-квадрата. Уравнение для хи-квадрата следующее: , где «xн» — это наблюдаемое значение, а «xо» — это ожидаемое значение.Суммируем результаты данного уравнения для всех возможных результатов и после этого по данной таблице определяем соответствующее р-значение в заголовке столбца.

Изображение выглядит как стол

Автоматически созданное описание

## **5) Логарифмическая доходность**

Альтернативный способ измерения доходности. Может использоваться вместо процентной и имеет существенное преимущество в том, насколько большие данные можно сравнивать, используя его, при этом в итоге получая корректный результат.

(так как по свойству логарифм частного равен разности логарифмов)

k – текущий временной промежуток, Рk – цена акции в момент времени k,   
Рk-1 – цена акции за предшествующий период.

Помимо сказанной выше, логарифмическая доходность имеет другие ощутимые преимущества :

В первую очередь, имея низкую частоту выплат при такой оценке легко объединять доходы с имеющими высокую частоту. В итоге ежемесячная доходность будет складываться из ежедневных значений. Такой подход позволяет упростить вычисления по сравнению с процентной оценкой. Кроме того, при таком методе можно вычислить начальное значение стоимости ценной бумаги, которая сначала выросла в цене на x%, а после снизилась на равную величину процентов x. [3]

## **6) Критерий Колмогорова**

Один из основных критериев, определяющих оценку достоверности гипотезы. С помощью него можно проверить, сходно ли утверждение с каким-то из классических законов распределения.

Так и считается наибольшая по модулю разница между фактическим значением заданной эмпирической функцией Fn(x) и теоретически ожидаемым значением функции распределения F(x).

Введём обозначение – количество составляющих вектора , где xi < x для любого x . В случае случайного вектора, формула не изменится, но принимать она будет значения дискретной случайной величины от 0 до n . Эмпирическая функция распределения, полученная из выборки X объёма n и сопоставленная с некоторой функцией F(x), будет определяться по следующей формуле:

Изображение выглядит как текст

Автоматически созданное описаниеИзображение выглядит как текст

Автоматически созданное описание

Вторая формула применяется для оценки функции F(x) по случайной выборке Х.

Примечание: – функция случайного процесса, поскольку x принимает различные случайные значения, и в то же время, как является числовой функцией.

Формулу расстояния между 2 функциями можно задать следующим соотношением :

. (6.3)

Как и раньше, в зависимости от аргумента, d может быть числом (при функции , а может быть случайной величиной, принимающей значения от 0 до 1, поскольку будет зависеть от при наличии функции

Таким образом, опираясь на теоремы Колмогорова, существует предел функции F(х), если она непрерывна и выбрано неотрицательно значение u:



По утверждению этой теоремы также устанавливается согласованность с критической областью , где - корень уравнения K(u) = 1-а, уровень значимости задаётся через предел стремления к при бесконечно возрастающем n. То есть можно утверждать, что появляется понятие асимптотического уровня значимости, которым является . При таких условиях, а также при n > 20, имеет смысл применять данный критерий, получивший наименование Колмогорова. Если n < 20, настоящий уровень значимости будет существенно отличаться от номинального значения.

Если принять F(х), как некоторую теоретическую функцию, а , как фактическую эмпирическую функцию распределения, то для вычисления расстояния между ними используется данная формула:

Изображение выглядит как текст

Автоматически созданное описание

Таблица. “Граничные значения для статистического критерия Смирнова-Колмогорова ”

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 0,15 | 0,10 | 0,05 | 0,03 | 0,01 |
| DnH | 0,775 | 0,819 | 0,895 | 0,995 | 1,035 |

## **7) Критерий Фишера**

Использование критерия Фишера обычно, когда требуется проверить соотношение двух дисперсий выборок нормального распределения. Для этого необходимо вычислить дисперсии случайных процессов, после чего по значению соотношения можно будет понять, выполняется критерий Фишера или нет.

По центральной предельной теореме, принято считать, что закон нормального распределения задаёт функции распределения котировок доходностей акций компаний. Иными словами, распределение суммарной величины всех логарифмов близко к нормальному, при условии, что количество дней достаточно велико. Чтобы убедиться в этом, смоделируем ситуацию, воспользовавшись критерием Колмогорова

В качестве начальных данных смоделируем по функциям нормального распределения две выборки случайных значений :

Изображение выглядит как текст

Автоматически созданное описание

Допустим, что значения параметров нам известны. Тогда основной гипотезой будет являться утверждение, что (H0 ). В качестве дополнительной гипотезы выберем один из 3 вариантов:

Изображение выглядит как текст

Автоматически созданное описание

Значение критерия Фишера для двух выборок данных будет вычисляться по формуле:

- дисперсии некоторой случайной выборки

# **II. Практическая работа**

## **1) Предварительная обработка и аналитика данных**

Для начала работы необходимо подготовить данные. Источником послужит веб ресурс, содержащий глобальную финансовую аналитику, а также вся статистика котировок акций компаний [7] за прошлые годы. Данные о том, акции каких компаний входят в этот индекс, а также о других секторах рынка можно использовать с сайта Московской биржи [8], а также воспользоваться дополнительно данными Московского финансового дома [9] для дополнительной проверки статистики.

Сопоставим тикеры акций, входящих в состав Финансового сектора ММВБ, с названиями компаний в таблице:

Таблица 1.   
Состав тикеров индекса MOEXFN, включающего в себя акции финансового сектора Мосбиржи

|  |  |
| --- | --- |
| Тикер акции | Название компании |
| SBER | ПАО «Сбербанк России» |
| SBER\_p | ПАО «Сбербанк России»  (привилегированные акции) |
| CBOM | ПАО «Московский кредитный банк» |
| MOEX | ПАО «Московская биржа» |
| RENI | ПАО «Группа Ренессанс Страхование» |
| QIWIDR | АО «Киви» |
| BSPB | ПАО «Банк Санкт-Петербург» |
| SFIN | ПАО «ЭсЭфАй» |
| TCSGDR | TCS Group Holding PLC |
| VTBR | ПАО «Банк ВТБ» |

Чтобы вычислить количество дней, в которые проводились торги акциями этой компании, используем столбец ‘Дата’, записанную в формате ddMMyyyy. Для этого сгруппируем имеющиеся данные этого столбца по годам и приведём их к таблице в удобном виде. Записи строк будут перечислять данные по одному тикеру, а столбцы будут содержать информацию по одному году. Всё сохраняем, как файл с расширением csv с разделителем между записями «;». Результаты данного преобразования сохранены и их можно посмотреть в таблице 2.

Таблица 2. Суммарное количество торговых дней для всех компаний.

Изображение выглядит как стол

Автоматически созданное описание

Как можно заметить, среди данных компаний есть котировки акций, которые начали продаваться позже 2016 года. Так у RENI статистика начинается только в конце 2021 года, также и у TCSGDR только с конца 2019 года. Их данные не будут полезны для анализа общего положения индекса фондового рынка, поэтому далее в статистике они рассматриваться не будут.

Сформируем таблицу, в которой будут присутствовать только компании, которые стабильно продавались большую часть года в рассматриваемый временной промежуток – с 2016 по 2021 годы. Результат просеивания данных можно наблюдать в таблице 3.

Таблица 3. Суммарное количество торговых дней для отредактированного списка компаний

Изображение выглядит как стол

Автоматически созданное описание

Список, который можно увидеть выше, содержит 8 компаний, каждая из которых на протяжении последних 6 лет имела более 229 торговых, поэтому на основании этих данных можно продолжить анализ гипотезы и преобразование данных к нужному виду.

Теперь выделим для каждой акции максимальный по возрастанию и по убыванию относительный скачок цены. Для этого используем данные, которые содержаться в столбце “Цена” – стоимость акции, сформированная в момент закрытия. После применения функции, рассчитывающей рост значения, относительно предыдущего, мы получаем 2 таблицы, которые сохранены также в формате csv файла. Чтобы цифры выглядели более понятными, откроем файл в Microsoft Excel и применим к данным условное форматирование.

В таблице 4 приведены максимальные относительные скачки цен вниз. Вариация идёт от меньшего по модулю значения (Зелёный) к большему (Красный)

Таблица 4. Максимальный относительный рост цены вверх

Изображение выглядит как стол

Автоматически созданное описание

Далее рассмотрим обратную ситуацию. Посчитаем максимальные относительные скачки цен вниз. Результаты занесём в таблицу 5, где также применим условное форматирование Excel таблиц. Вариация идёт от меньшего по модулю значения (Зелёный) к большему (Красный)

Таблица 5. Максимальный относительный рост цены вниз

Изображение выглядит как стол

Автоматически созданное описание

В итоге теперь мы располагаем всей статистикой значительных изменений цен выбранных акций .По данным таблиц можно выделить 2 котировки, стабильность которых стоит проверить : SFIN и QIWIDR. Для этого откроем файлы соответствующих акций и по данным в столбце “Цена” построим график изменения цены с течением времени.

Рисунок 1. График изменения цены акций компании SFIN.



Наибольшие скачки цен вниз были заметны в 2018 году (-26,1 %), однако данные случаи не выбиваются за общие границы значений выборки, а также не превышают 40 % в относительном изменении цены. Поэтому данные этих акций можно использовать для дальнейшего анализа.

Рисунок 2. График изменения цены акций компании QIWIDR.



Максимальный рост цены акции зафиксирован в 2019 году (24,2 %). Постоянные рост и падение цены акции в течение дня по своей величине не имеют слишком большое значение, поэтому данную выборку также можно считать репрезентативной.

Таким образом, после предварительной обработки данных, у нас остались котировки 8 акций, которые подходят для дальнейшего анализа и исследования гипотезы, то есть имеют большое число (> 229) дней, в которые их акции участвовали в торгах на бирже и не имеют существенный разброс в стоимости. Выборка реальных данных, на которых будет проверяться исследуемая гипотеза, подготовлена и приведена к нужному виду.

## **2) Проверка гипотезы на сгенерированных данных**

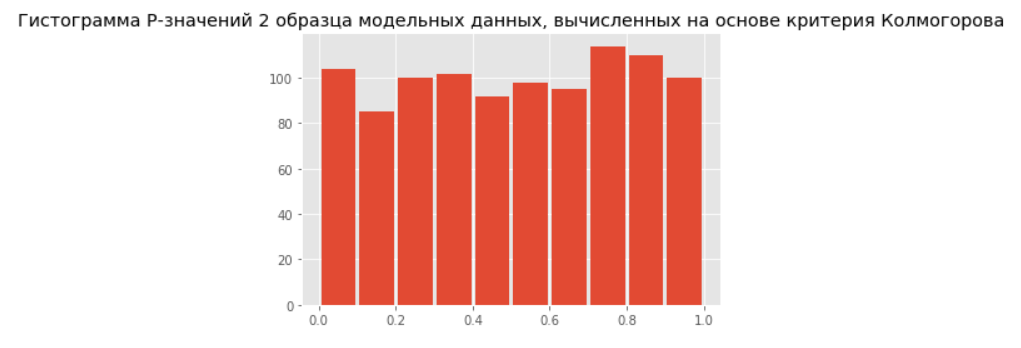
Проверим нашу гипотезу о равенстве дисперсий логарифмических доходностей фондового рынка и входящих в его состав акций для данных, которые будут сгенерированы по определённому образцу, в соответствии с критерием Колмогорова. Для этого используем метод Монте-Карло, создадим 2 выборки из 252 элементов каждая, после чего на их основе выясним: верна ли гипотеза, и правильно ли написана программа для вычисления критерия Фишера и Р-значения.

Для наглядности выведем гистограмму Р-значений, отдельно продемонстрировав каждое из распределений на графике. Так как мы проводим несколько экспериментов, и каждый раз получаем различные р-значения, стоит проверить, как часто встречаются определённые значения:

Рисунок 3. Первое распределение сгенерированных данных. Р-значения.



Рисунок 4. Второе распределение сгенерированных данных. Р-значения.



После соответствующих вычислений в среде разработки, получаем Р-значения, равные 0.41963114811849644 и 0.36498239642797903 для первой и второй выборки соответственно. Это показатель того, что Р-значения распределены по равномерному закону для таких модельных данных, так как р-значение по своей величине больше общепринятого критерия значимости (5%).

Теперь проверим эти данные критерием Фишера. Полученное значение: 1.0321616259272595. Результат свидетельствует о том, что дисперсии практически равные, а значит вывод о том, что гипотеза о равенстве дисперсий логарифмических доходностей фондового рынка и входящих в его состав акций полностью выполняется в идеальных случаях, когда используются модельные данные одинакового образца.

## **3) Проверка правильности гипотезы на реальной статистике**

После проверки, что программа работает корректно, все данные обрабатываются в нужном формате, и котировки проверены и приведены к нужному виду, можно приступить к моделированию алгоритма на реальных данных. Итак, используем критерий Фишера, чтобы проверить верна ли гипотеза о равенстве дисперсий логарифмических доходностей фондового рынка и входящих в его состав акций на примере акций индекса MOEXFN – индекса Московской биржи, включающий в себя финансовый сектор. Для вычислений, как и при предварительном анализе котировок, нам потребуются столбцы с данными “Дата” (день совершения сделок на бирже) и “Цена” (стоимость закрытия последней сделки с этой акцией в определённый день). Для того, чтобы проверить гипотезу, нам необходима логарифмическая доходность, данными о которой мы пока не располагаем в наших таблицах с данными. Вычислим эти значения и сохраним в списке с другими данными. Для этого используем уже существующие функции языка Python.

После этого необходимо вычислить Р-значения (I раздел. Пункт 4) функции на основе реальных данных. Для этого мы можем использовать написанную ранее функцию, подготовить выборку данных за 6 лет, удовлетворяющую критерию Колмогорова. Итоговые значения вынесем для наглядности в таблицу.

Таблица 6. Р-значения для настоящих данных за 6 лет в течение каждого года

Изображение выглядит как стол

Автоматически созданное описание

Также, предоставим гистограмму, отражающую полученные данные для наглядного отображения:

Рисунок 5. Распределение Р-значений



Из-за того, что Р-значения приближены к 0, нет оснований для выводов о том, какое распределение задаёт эти функции цены реальных акций компаний, в то числе исключается и равномерное распределение. Следовательно, для каждой из котировок акций компаний вычисляется отдельно значение дисперсии, чтобы потом провести все финальные расчёты.

Таблица 7. Данные о дисперсии логарифмических доходностей за 6 лет по каждому году

Изображение выглядит как стол

Автоматически созданное описание

Так как мы не можем использовать прямое сравнение дисперсий при анализе данных, поэтому переходим сразу же к подсчёту значений критерия Фишера для выбранного набора акций. Сравниваться каждая из них будет со значениями индекса Мосбиржи Финансового сектора с помощью уже написанной ранее функции.

Значение критерия Фишера для двух выборок данных будет вычисляться по формуле:

- дисперсии некоторой случайной выборки

Таблица 8. Значения критериев Фишера для каждой акции

Изображение выглядит как стол

Автоматически созданное описание

Полученные значения для данного эксперимента свидетельствуют о том, что выбранная для проверки гипотеза выполняется для индекса Финансового сектора Московской биржи. Значения критерия Фишера отражают соотношение близкое к нормальному.

Наибольшее значение дисперсии логарифмической доходности у привилегированных акций Сбербанка стало причиной наибольшего значения критерия Фишера при проверке вычислений в данных. Далеко не все реальные данные могут проявляться таким же образом, так как ситуация в разных секторах экономики принципиально отличается. Финансовый сектор очень сильно централизован и следует общей политике государства, более того, он во многом зависим от её состояния, вследствие чего напрашивается вывод о том, что предложенная для анализа гипотеза выполняется на этом наборе реальных данных.

# **IV. Заключение**

При написании данной курсовой работы была поставлена цель – проверить гипотезу о равенстве дисперсий логарифмических доходностей индекса фондового рынка и входящих в его состав акций, с использованием методов сравнения и анализа, такие как критерий Фишера и критерий Колмогорова. За основу брались данные индекса Финансового сектора Московской биржи (MOEXFN), а проверка проводилась как на случайно сгенерированных модельных данных, так и на реальной выборке котировок акций компаний.

Результатом анализа выполнимости выбранной гипотезы, мною были получены следующие результаты : гипотеза подтвердилась на модельных данных, сгенерированных согласно определённому принципу, и также подтвердилась на выбранном наборе реальных данных, предварительно прошедших просеивание и анализ репрезентативности их, как достоверных.

По данным анализа лишь одного сектора бизнеса не следует обобщать вывод о справедливости гипотезы, однако можно сказать, что она выполнима для финансового сектора, который представляет интересы общего рынка, а также основных его регуляторов.

Искажения в результатах работы и входе анализа данных могли быть вызваны различными недостатками в критериях, выбранных для проверки гипотезы. Так критерий Фишера очень чувствителен к отклонениям от нормального распределения в исследуемой выборке, а также к выборкам разного размера, которые гораздо чаще встречаются в реальных данных, в отличие от настроенных моделей.

Однако, фактическое выполнение критериев на модельных данных в диапазоне от 0 до 1 подтверждено мною в ходе проведения работы, а это означает, что проведённые вычисления не являются ошибочными и выбранные критерии обеспечивают достоверную проверку.

# **V. Список использованных источников**

1. – М.: Финансы и статистика, 2013
2. Браилов А.В. Лекции по математической статистике. – М.: Финакадемия, 2008.
3. Браилов А.В. Лекции по теории вероятности. – М.: Финакадемия, 2008.
4. Глебов Криволапов Практикум по математической статистике. Проверка гипотез с использованием Excel, MatCale, R и Python. М.: Прометей, 2019.
5. Теория вероятностей и математическая статистика (для бакалавров): учебное пособие / Кацко И.А. ред. – Москва: КноРус, 2019. – 389с. – URL: https://[www.book.ru/book/930219](http://www.book.ru/book/930219) (Дата обращения: 25.04.2021)
6. Смирнова З.М., Крейнина М.В. ПРОВЕРКА СТАТИСТИЧЕСКИХ ГИПОТЕЗ.
7. <https://ru.investing.com/equities/>
8. <https://mfd.ru/>
9. <https://smart-lab.ru/q/index_stocks/MOEXFN/>

# **VI. Приложения**

## Приложение 1. Технические характеристики персонального ноутбука

Изображение выглядит как стол

Автоматически созданное описание

## Приложение 2. Список файлов

|  |  |
| --- | --- |
| Имя файла | Содержание |
| Таблица 1 | Состав тикеров индекса MOEXFN, включающего в себя акции финансового сектора Мосбиржи |
| Таблица 2 | Суммарное количество торговых дней для всех компаний. |
| Таблица 3 | Суммарное количество торговых дней для отредактированного списка компаний |
| Таблица 4 | Максимальный относительный рост цены вверх |
| Таблица 5 | Максимальный относительный рост цены вниз |
| Таблица 6 | Р-значения для настоящих данных за 6 лет в течение каждого года |
| Таблица 7 | Данные о дисперсии логарифмических доходностей за 6 лет по каждому году |
| Таблица 8 | Значения критериев Фишера для каждой акции |
| Таблица 9 | Мощность критерия Фишера для различных вычислений |
| Рисунок 1 | График изменения цены акций компании SFIN. |
| Рисунок 2 | График изменения цены акций компании QIWIDR. |
| Рисунок 3 | Первое распределение сгенерированных данных. Р-значения. |
| Рисунок 4 | Второе распределение сгенерированных данных. Р-значения. |
| Рисунок 5 | Распределение Р-значений |

## Приложение 3. Программный код на языке python, реализованный в среде Jupyter notebook

import time

start\_time = time.time() #Переменная для подсчёта времени работы программы

#Импорт задействованных библиотек(технические)

import numpy as np

import pandas as pd

import csv

#Библиотеки для анализа различных статистических показателей

import matplotlib.pyplot as plt

import matplotlib

import statistics as st

import scipy.stats as stats

#Задаём стиль графиков

matplotlib.style.use('ggplot')

**Подготовка данных для анализа**

#Подсчёт общего числа рабочих дней в компаниях

ticker\_names = ['BSPB','CBOM', 'QIWIDR', 'RENI', 'SBER', 'SBER\_p', 'SFIN', 'TCSGDR', 'VTBR', 'MOEX']

all\_days\_amount = pd.DataFrame()

all\_days\_amount['Ticker'] = ticker\_names

years = [i for i in range(2016, 2022)]

for i in years:

work\_days\_amount = []

for cur\_ticker in ticker\_names:

file = pd.read\_csv(cur\_ticker + '.csv', sep = ';')

dlit = (file['Дата'] % 10000 >= i)&(file['Дата'] % 10000 < (i+1))

amount = len(file[dlit])

work\_days\_amount.append(amount)

all\_days\_amount[str(i)] = work\_days\_amount

all\_days\_amount.to\_csv('Общее количество торговых дней в году для всех компаний.csv', sep = ';')

all\_days\_amount

#Подсчёт общего числа рабочих дней в компаниях, у которых сохраняется стабильное число рабочих дней за весь период измерения

ticker\_names = ['BSPB', 'CBOM', 'QIWIDR', 'SBER', 'SBER\_p', 'SFIN', 'VTBR', 'MOEX']

all\_days\_amount = pd.DataFrame()

all\_days\_amount['Ticker'] = ticker\_names

years = [i for i in range(2016, 2022)]

for i in years:

work\_days\_amount = []

for cur\_ticker in ticker\_names:

file = pd.read\_csv(cur\_ticker + '.csv', sep = ';')

dlit = (file['Дата'] % 10000 >= i)&(file['Дата'] % 10000 < (i+1))

amount = len(file[dlit])

work\_days\_amount.append(amount)

all\_days\_amount[str(i)] = work\_days\_amount

all\_days\_amount.to\_csv('Общее количество торговых дней в году для выбранных компаний.csv', sep = ';')

all\_days\_amount

#Подсчёт относительных скачков цен вниз

ticker\_names = ['BSPB', 'CBOM', 'QIWIDR', 'SBER', 'SBER\_p', 'SFIN', 'VTBR', 'MOEX']

years = [i for i in range(2016, 2022)]

lower\_cost = pd.DataFrame() #Создаём Data Frame для сохранения данных о котировках по годам

lower\_cost['Тикер'] = ticker\_names

for i in years:

decreas\_vol = []

for cur\_name in ticker\_names:

company\_data = pd.read\_csv(cur\_name + '.csv', sep = ';')#Считываем данные котировки

company\_data['Delta'] = company\_data['Цена'].pct\_change().round(3)#Считаем разницу цены в соседние дни

dlit = (company\_data['Дата'] % 10000 >= i)&(company\_data['Дата'] % 10000 < (i+1)) #ограничиваем данные только текущим годом

cur\_year = company\_data[dlit]['Delta'].min()

decreas\_vol.append(cur\_year)

lower\_cost[str(i)] = decreas\_vol

lower\_cost.to\_csv('Относительные изменения цен вниз.csv', sep = ';', decimal=',') # Записываем полученные данные в csv файл. Чтобы потом использовать их в работе

lower\_cost

#Подсчёт относительных скачков цен вверх

ticker\_names = ['BSPB', 'CBOM', 'QIWIDR', 'SBER', 'SBER\_p', 'SFIN', 'VTBR', 'MOEX']

years = [i for i in range(2016, 2022)]

upper\_cost = pd.DataFrame() #Создаём Data Frame дял сохранения данных о котировках по годам

upper\_cost['Тикер'] = ticker\_names

for i in years:

incr\_vol = []

for cur\_name in ticker\_names:

company\_data = pd.read\_csv(cur\_name + '.csv', sep = ';')#Считываем данные котировки

company\_data['Delta'] = company\_data['Цена'].pct\_change().round(3)#Считаем разницу цены в соседние дни

dlit = (company\_data['Дата'] % 10000 >= i)&(company\_data['Дата'] % 10000 < (i+1)) #ограничиваем данные только текущим годом

cur\_year = company\_data[dlit]['Delta'].max()

incr\_vol.append(cur\_year)

upper\_cost[str(i)] = incr\_vol

upper\_cost.to\_csv('Относительные изменения цен вниз.csv', sep = ';', decimal=',')

upper\_cost

**Создание модели данных и проверка гипотезы на них**

#Функция проверки критерия Колмогорова

def kolm\_krit(n): #Функция вычисления критерия Колмогорова. Параметр n задаёт размер запрашиваемой выборки

data = np.random.normal(loc = 0, scale = 1, size = n) #Задаём параметры генерации нормального распределения

result = stats.kstest(data,'norm') #Проверка правильности критерия Колмогорова с помощью существующей функции

return data, round(result[1],3) #Возвращаем полученное распределение и Р-значение, являющееся вторым выходным результатом, сохранённым в массив result

#Размер выборки - среднее число торговых дней в год

n = 252

#Далее сгенерируем 2 выборки данных, сравнив дисперсии которых по критерию Фишера, мы сможем проверить достоверность гипотезы

#построение гистограммы Р-значений, вычисленных с помощью критерия Колмогорова. (1 выборка)

P\_values\_1 = [] # Пустой массив для сохранения Р-значений выборки

tetst\_data\_1 = kolm\_krit(n)[0] # Генерация данных на основании критерия Колмогорова

print(stats.kstest(tetst\_data\_1, 'norm'))

for i in range(10 \*\* 3):

P\_values\_1.append(kolm\_krit(n)[1])#Получаем Р-значения с помощью написанного модуля и сохраняем их в массив

#Отображение данных (из предыдущей ячейки) в формате гистограммы

plt.hist(P\_values\_1,rwidth = 0.9) #Создаём макет гистограммы и передаём ей данные и параметры оформления

plt.title('Гистограмма Р-значений 1 образца модельных данных, вычисленных на основе критерия Колмогорова) #Заголовок графика

#построение гистограммы Р-значений, вычисленных с помощью критерия Колмогорова. (2 выборка)

P\_values\_2 = [] # Пустой массив для сохранения Р-значений выборки

tetst\_data\_2 = kolm\_krit(n)[0] # Генерация данных на основании критерия Колмогорова

print(stats.kstest(tetst\_data\_2, 'norm'))

for i in range(10 \*\* 3):

P\_values\_2.append(kolm\_krit(n)[1])#Получаем Р-значения с помощью написанного модуля и сохраняем их в массив

#Отображение данных (из предыдущей ячейки) в формате гистограммы

plt.hist(P\_values\_2,rwidth = 0.9) #Создаём макет гистограммы и передаём ей данные и параметры оформления

plt.title('Гистограмма Р-значений 2 образца модельных данных, вычисленных на основе критерия Колмогорова) #Заголовок графика# Проверим, имеют ли сгенерированных 2 набора данных общий вид

compar1 = stats.ks\_2samp(data\_1, data\_2)

print(compar1[1])

#Функция проверки критерия Фишера

def fisher\_crit(var1, var2):

return ((max(var1, var2))\*\*2/(min(var1, var2))\*\*2)

#Вычислим дисперсию 2 выборок данных

var\_data\_1 = np.var(data\_1)

var\_data\_2 = np.var(data\_2)

print("Дисперсия первой выборки = " + str(var\_data\_1), '\nДисперсия второй выборки = ' + str(var\_data\_2))

res\_fisher = fisher\_crit(var\_data\_1, var\_data\_2)

print("Критерий Фишера = ", res\_fisher)

#Проверка мощности критерия

m = 10000 #количество экспериментов

res1 = []

res2 = []

for i in range(m):

data\_11 = (stats.t.rvs(df = 3, loc = 0, scale = 1,size = 252))

data\_12 = (stats.t.rvs(df = 3, loc = 0, scale = 1,size = 252))

data\_21 = (stats.lognorm.rvs(s = 1/4 ,loc = 0, scale = 1,size = 252))

data\_22 = (stats.lognorm.rvs(s = 1/4 ,loc = 0, scale = 1,size = 252))

var\_data\_11 = np.var(data\_11)

var\_data\_12 = np.var(data\_12)

var\_data\_21 = np.var(data\_21)

var\_data\_22 = np.var(data\_22)

res1.append(fisher\_crit(var\_data\_11, var\_data\_12))

res2.append(fisher\_crit(var\_data\_21, var\_data\_22))

power1 = 0 #Подсчёт числа малых значений критерия Фишера для первой выборки

power2 = 0 #Подсчёт числа малых значений критерия Фишера для второй выборки

## Вычисление мощности критерия

for counter in res1:

if counter < 0.05: #Если значение достаточно мало

power1 += 1

for counter in res2:

if counter < 0.05: #Если значение достаточно мало

power2 += 1

#Далее вычислим значения в среднем, разделив число удачных испытаний на число экспериментов

print(power1 / m)

print(power2 / m)

**Проверка гипотезы на подготовленных реальных данных фондового рынка**

#импорт таблиц

def to\_table(ticker): # Функция, выполняющая считывание файла(входной параметр ticker) и преобразование даты в Data Frame к нужному формату

company = pd.read\_csv(ticker+'.csv', sep=';') #Открываем файл с котировками акции определённой компании

company['Дата'] = company['Дата'].astype('object')

for i in range(len(company['Дата'])):

company['Дата'][i] = str(company['Дата'][i])

if len(company['Дата'][i]) < 8:

company['Дата'][i] = '0' + company['Дата'][i]

company['Дата'] = pd.to\_datetime(company['Дата'], format = '%d%m%Y') #преобразование даты

return company

def logTransf(ticker, num):

LD = []

all\_years = ['2016', '2017', '2018', '2019', '2020', '2021']

company = to\_table(ticker)

company['Логарифм доходности'] = np.log(company['Цена'])

all\_days\_table = pd.read\_csv('Общее количество торговых дней в году для выбранных компаний.csv', sep=';')

all\_days\_table = all\_days\_table.drop(columns = all\_days\_table.iloc[:, range(1)]) #Убираем лишний столбец с нумерацией

for year in all\_years:

n = (all\_days\_table[str(year)][num]) #Количество дней в текущем проверяемом году

first\_index = company['Логарифм доходности'][company['Дата'] <= str(year)+'-12-31'].size - n #Находим индекс первого дня в году

#Незабываем сохранить данные по текущему году, чтобы потом целиком скинуть всю инофрмацию по тикеру

LD.append((company['Логарифм доходности'][company['Дата'] <= year + '-12-31'])[first\_index:].kurtosis()) #вычисляем несмещённый эксцесс от выборки данных

return LD

ticker\_names = ['BSPB', 'CBOM', 'QIWIDR', 'SBER', 'SBER\_p', 'SFIN', 'VTBR', 'MOEX']

PV = []

for cur\_ticker in ticker\_names:

num = ticker\_names.index(cur\_ticker)

log = logTransf(cur\_ticker, num)

def calc\_log\_max(file, year = 0):

#Функция для расчёта логарифмической доходности.

# Параметры - название файла (file) и год, за который требуется посчитать доходность (year)

#при значении 0 считает за все годы

company = pd.read\_csv(file, sep=';') #считывание файла csv по определённому тикеру

company['Дата'] = [int(str(company['Дата'][i])[4:]) for i in range(len(company['Дата']))] #из даты извлекается год

#Далее через встроенную функцию логарифма, нормируем значения выборки, используя точность вычисления до сотых

if year > 0: #Расчёт логарифмической доходности при заданном параметре год

log\_max\_res = np.log(company['Цена'][company['Дата'] <= int(year)][company['Дата'] > int(year) - 1])

else:

log\_max\_res = np.log(company['Цена'])

return [\*stats.kstest(log\_max\_res, 'norm'), np.var(log\_max\_res)] #возвращаем 3 параметра под разные запросы программы

#Список всех рассматриваемых тикеров компаний

all\_suitable\_tickers = ['BSPB', 'CBOM', 'QIWIDR', 'SBER', 'SBER\_p', 'SFIN', 'VTBR', 'MOEX', 'MOEXFN']

#Временной промежуток, на котором у нас есть данные: с 2016 по 2021

years = [i for i in range(2016,2022)]

#Создание дата фрейма для дальнейших вычислений

table\_log\_max = pd.DataFrame()

#Столбец с наименованиями тикеров

table\_log\_max['Тикер'] = all\_suitable\_tickers

#Вычисляем, какой был наибольший скачок стоимостей акций при группировке по годам

all\_data = [] #Те же данные в формате массива, чтобы их можно было показать на гистограмме

for year in years:

log\_cur\_max = [] #Список для хранения значения по всем тикерам в течение одного года

for ticker in all\_suitable\_tickers:

#Добавление в список значение макс.логарифмических доходностей

#Подсчёт идёт по каждому тикеру за все дни в течение одного года, после чего число записывается в ячейку памяти

log\_cur\_max.append(round(calc\_log\_max(ticker+'.csv', year)[1], 10))

all\_data.append(round(log\_cur\_max[-1], 10))

#После вычисления всех данных по тикерам в течение одного года (log\_number\_max)

#Сохраняем в таблицу (table\_log\_max) все данные для общей статистики

table\_log\_max[str(year)] = log\_cur\_max #группировка в данной таблице идёт по годам

#Заголовок перед таблицей с данными (выделяем жирным шрифтом)

print('\033[1m' + 'Таблица. Р-значения по критерию Колмогорова')

table\_log\_max

#Отображение данных (из предыдущей ячейки) в формате гистограммы

plt.hist(all\_data,rwidth = 0.9) #Создаём макет гистограммы и передаём ей данные и параметры оформления

plt.title('Гистограмма Р-значений реальных данных, вычисленных с помощью критерия Колмогорова') #Заголовок графика

plt.show()

#Расчёт максимального скачка цены (по годам)

for year in years:

log\_cur\_max = [] #Список для хранения значения по всем тикерам в течение одного года

for ticker in all\_suitable\_tickers:

#Добавление в список значение макс.логарифмических доходностей

#Подсчёт идёт по каждому тикеру за все дни в течение одного года, после чего число записывается в ячейку памяти

log\_cur\_max.append(round(calc\_log\_max(ticker + '.csv', year)[2], 10))

#После вычисления всех данных по тикерам в течение одного года (log\_number\_max)

#Сохраняем в таблицу (table\_log\_max) все данные для общей статистики

table\_log\_max[str(year)] = log\_cur\_max

#Заголовок перед таблицей с данными (выделяем жирным шрифтом)

print('\033[1m' + 'Таблица. Дисперсии логарифмической доходности по годам')

table\_log\_max

#Расчёт значений дисперсии по каждому тикеру за всё время

list\_var = []

for ticker in all\_suitable\_tickers:

#Добавление в список значение макс.логарифмических доходностей

#Подсчёт идёт по каждому тикеру за все дни измеряемого промежутка, после чего число записывается в ячейку памяти

list\_var.append(round(calc\_log\_max(ticker + '.csv')[2], 10))

#После вычисления всех данных по тикеру

#Сохраняем в список (table\_log\_max) все данные для общей статистики

#Заголовок перед таблицей с данными (выделяем жирным шрифтом)

print('Таблица. Дисперсии логарифмической доходности\n')

for i in range(len(all\_suitable\_tickers)):

print(all\_suitable\_tickers[i], ':', list\_var[i]) #Выводим дисперсию вместе с названием тикера

#Создаю Дата фрейм для сохранения значений критерия Фишера

result = pd.DataFrame()

result['Tiker'] = all\_suitable\_tickers[:-1] #Переносим список всех тикеров кроме индекса MOEXFN(с ним всё сравниваем)

result["Fishre's criterion"] = all\_suitable\_tickers[:-1] #Столбец для значений критерия Фишера

for i in list\_var[:-1]:

y = fisher\_crit(i, list\_var[-1]) #По уже написанной функции считаем значение критерия

result["Fishre's criterion"][list\_var.index(i)] = y #Сохраняем в строке соответствующего тикера

result.to\_csv('Значения критерия Фишера для всех выбранных акций.csv', sep = ';')

print('\033[1m' + 'Таблица. Значения критерия Фишера')

result

finish\_time = time.time()

#Общее время работы программы, включая все операции анализа, импорта библиотек и генерации данных

print("--- %s seconds ---" % (finish\_time - start\_time))