

Министерство образования и науки Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
“САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ,
МЕХАНИКИ И ОПТИКИ”

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

**Разработка системы поддержки принятия решений при
заключении контрактов на поставку электроэнергии с
использованием методов оценки стоимости опционов**

Автор: Коршунов Андрей Андреевич _____
(Подпись)

Направление подготовки (специальность): 09.04.01
Информатика и вычислительная техника

Квалификация: магистр

Руководитель: Радченко И.А., К.Т.Н., Доцент _____
(Подпись)

К защите допустить

Зав. Кафедрой: Муромцев Д.И., К.Т.Н., Доцент _____
(Подпись)

“ _____ ” _____ 20 ____ г.

Санкт-Петербург, 2018 г.

Студент Коршунов А.А., Группа Р4215, Кафедра ИПМ, Факультет ПИиКТ

Направленность (профиль), специализация 09.04.01 Математические модели и компьютерное моделирование

Консультант (ы):

а) _____
(Фамилия, И., О., ученое звание, степень) (Подпись)

ВКР принята “ ____ ” _____ 20 ____ г.

Оригинальность ВКР _____ %

ВКР выполнена с оценкой _____

Дата защиты “ ____ ” _____ 20 ____ г.

Секретарь ГЭК _____
(ФИО) (подпись)

Листов хранения _____

Демонстрационных материалов/Чертежей хранения _____

Министерство образования и науки Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
“САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ,
МЕХАНИКИ И ОПТИКИ”

УТВЕРЖДАЮ

Зав. кафедрой _____

(ФИО)

(подпись)

« ____ » « ____ » 2018 г.

ЗАДАНИЕ
НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ

Студенту Коршунову А.А., Группа Р4215, Кафедра ИПМ, Факультет ПИиКТ

Руководитель: Радченко Ирина Алексеевна, кандидат технических наук, доцент

1 Наименование темы: Разработка системы поддержки принятия решений при заключении контрактов на поставку электроэнергии с использованием методов оценки стоимости опционов

Направление подготовки (специальность): 09.04.01 Информатика и вычислительная техника

Направленность (профиль): Математические модели и компьютерное моделирование

Квалификация: Магистр

2 Срок сдачи студентом законченной работы « ____ » « ____ » 20 ____ г.

3 Техническое задание и исходные данные к работе

Необходимо разработать систему поддержки принятия решений для менеджера компании, занимающейся оптовыми и розничными поставками электроэнергии в Скандинавии.

Система должна быть реализована в виде веб-сайта с возможностью синхронизации цен с биржей Nasdaq, синхронизации заключенных сделок с существующей серверной системой, механизмом расчёта надбавленной цены опциона, а также интерактивным отображением рассчитанных данных.

4 Содержание выпускной квалификационной работы (перечень подлежащих разработке вопросов)

Во-первых, необходимо исследовать существующие модели и методы ценообразования опционов, выявить плюсы и минусы каждой, а также понять, какие части этих моделей и методов могут быть изменены, чтобы приблизить вычисления к «реальной жизни».

Во-вторых, для оценки моделей и методов и их возможных будущих доработок, необходимо создать систему, позволяющую проводить имитационное моделирование портфеля опционных позиций на предыдущие года, где цена базового актива в момент исполнения заранее известна.

В-третьих, проанализировав все модели и методы, нужно сделать предположение об внедрении таких изменений в них, которые бы приблизили их к «реальной жизни», запрограммировать эти изменения и провести имитационное моделирование, сделав вывод о жизнеспособности данных оптимизаций.

В результате нужно реализовать систему поддержки принятия решений для менеджера, осуществляющего прямую продажу опционных контрактов на электроэнергию.

5 Перечень графического материала (с указанием обязательного материала)

В обязательном порядке необходимо предоставить графики реальных распределений приращений биржевых цен. Также, все проводимые исследования должны быть по возможности проиллюстрированы с помощью графиков и диаграмм.

6 Исходные материалы и пособия

Black F. The pricing of commodity contracts // Journal of Financial Economics. – Vol.3 – 1976

Шеннон Р. Имитационное моделирование – искусство и наука. / Р. Шеннон // Мир. – 1978

Глухов М. Оценка опционов методом Монте Карло / М. Глухов // Futures & Options. – 2009

7 Дата выдачи задания « ____ » « _____ » 20 ____ г.

Руководитель ВКР

(подпись)

Задание принял к исполнению

(подпись)

« ____ » « _____ » 2018 г.

Министерство образования и науки Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
“САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ,
МЕХАНИКИ И ОПТИКИ”

АННОТАЦИЯ

ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЫ

Студент Коршунов Андрей Андреевич
(ФИО)

Наименование темы ВКР: Разработка системы поддержки принятия решений при заключении контрактов на поставку электроэнергии с использованием методов оценки стоимости опционов

Наименование организации, где выполнена ВКР Университет «ИТМО»

ХАРАКТЕРИСТИКА ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЫ

1 Цель исследования Нахождение оптимального метода расчёта цены опциона на электричество, устойчивого к резким скачкам цены базового актива с последующим возвратом, для дальнейшего его использования в разрабатываемой системе поддержки принятия решений

2 Задачи, решаемые в ВКР 1. Провести обзор существующих методов ценообразования опционов; 2. Создать систему, позволяющую проводить имитационное моделирование портфеля опционных позиций; 3. Сделать предположение о способах оптимизации методов ценообразования, приближающих их к «реальной жизни»; 4. Провести ряд имитационных испытаний, сделать выводы; 5. Реализовать систему с пользовательским интерфейсом для менеджера, осуществляющего продажу опционных контрактов на поставку электроэнергии.

3 Число источников, использованных при составлении обзора 20

4 Полное число источников, использованных в работе 20

5 В том числе источников по годам

Отечественных			Иностранных		
Последние 5 лет	От 5 до 10 лет	Более 10 лет	Последние 5 лет	От 5 до 10 лет	Более 10 лет
1	2	2	4	1	10

6 Использование информационных ресурсов Internet Да, 2
(Да, нет, число ссылок в списке литературы)

7 Использование современных пакетов компьютерных программ и технологий (Указать, какие именно, и в каком разделе работы)

Пакеты компьютерных программ и технологий	Параграф работы
Microsoft .NET Framework	2.10
Microsoft Windows Communication Foundation	2.10
Microsoft ASP.NET MVC Framework	2.10
HTML, CSS, Javascript	2.10
jQuery	2.10

8 Краткая характеристика полученных результатов В рамках данной работы были рассмотрены такие понятия как волатильность, опцион и методы его ценообразования

В результате был предложен новый подход к выбору распределения отклонений цен при их генерации для метода Монте-Карло, который позволил немного улучшить показатели, минимизировав цену опциона для высоковолатильных месяцев.

Также в рамках данной работы было проведено имитационное моделирование портфеля, сделаны выводы об оптимальном подходе вычисления цены опциона, а также на основе этого подхода была реализована информационная система

9 Полученные гранты, при выполнении работы _____
(Название гранта)

10 Наличие публикаций и выступлений на конференциях по теме выпускной работы Нет _____
(Да, нет)

а) 1 _____
(Библиографическое описание публикаций)
2 _____
3 _____

б) 1 _____
(Библиографическое описание выступлений на конференциях)
2 _____
3 _____

Студент Коршунов Андрей Андреевич _____
(ФИО) (подпись)

Руководитель Радченко Ирина Алексеевна _____
(ФИО) (подпись)

“ _____ ” _____ 20 ____ г.

СОДЕРЖАНИЕ

ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ.....	5
1. ВВЕДЕНИЕ	6
1.1. Постановка цели и задач	9
1.2. Введение в предметную область	10
1.3. Существующие методы ценообразования опционов	16
1.4. Модель Блэка-Шоулза.....	18
1.5. Метод Монте-Карло	22
1.6. Имитационное моделирование	25
2. ОСНОВНАЯ ЧАСТЬ.....	29
2.1. Имитационное моделирование портфеля опционных позиций.....	29
2.2. Выбор распределения изменений цен	32
2.3. Индекс толщины «хвоста» распределения	34
2.4. Зависимость индекса MAD/SD от волатильности.....	36
2.5. Использование реального распределения цен актива при расчёте цены опциона методом Монте-Карло.....	38
2.6. «Индекс хвоста» ограниченного распределения Коши	42
2.7. Имитационное моделирование портфеля с использованием ограниченного распределения Коши	44
2.8. Заказчик	45
2.9. Внедрение в СППР	46
2.10. Выбор средств и технологий	49
2.11. Проектирование интерфейса	54
2.12. Работа с библиотеками.....	56
3. ЗАКЛЮЧЕНИЕ	59

4. СПИСОК ЛИТЕРАТУРЫ.....	60
---------------------------	----

Приложение А. Графики распределения изменения цен по месяцам. Сравнение 2016 и 2017 годов.	63
--	----

Приложение Б. Исходные коды алгоритмов и имитационного моделирования.	66
---	----

ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ

1. Опцион - финансовый инструмент, за счёт которого покупатель опциона получает право, но не обязательство, совершить покупку или продажу данного актива по заранее оговорённой цене в определённый договором момент в будущем.
2. Спот-рынок – или наличный рынок – это рынок, где товары, акции или другие финансовые инструменты, например, валюты, покупаются за наличные с немедленной поставкой.
3. Страйк цена – цена, установленная в опционе, по которой владелец опциона имеет право купить актив, а владелец актива обязан его продать.
4. Волатильность – показатель изменчивости цены базового актива за заданный промежуток времени.
5. Исторические цены – подневный массив цен на актив за заданный промежуток времени в прошлом.
6. Ценообразование — установление цены на товар или услугу.
7. Биржа – юридическое лицо, обеспечивающее регулярное функционирование организованного рынка биржевых товаров, валют, ценных бумаг и производных финансовых инструментов.
8. Кроссплатформенность – способность программного обеспечения функционировать в нескольких различных операционных системах или на разных аппаратных платформах без необходимости модифицирования программного кода.

1. ВВЕДЕНИЕ

Опцион — важнейший инструмент, за счёт которого покупатель опциона (потенциальный покупатель или потенциальный продавец базового актива) получает право, но не обязательство, совершить покупку или продажу данного актива по заранее оговорённой цене в определённый договором момент в будущем («европейский» тип опционов) или на протяжении определённого отрезка времени («американский» тип опционов).

Рынок электроэнергии Скандинавии сильно отличается от Российского, единая энергосистема стран Северной Европы стала своего рода плацдармом рыночных реформ, и сейчас здесь один из самых либеральных рынков в мире торговли электроэнергией. Маленький розничный потребитель в Швеции может заключить контракт с большим производителем в Норвегии, или он может купить его электроэнергию на рынке деривативов и застраховаться от скачков цен, купив фьючерсный контракт или опцион.

Возможность купить опцион очень важна, так как предоставляет покупателям гибкость в планировании затрат и возможность застраховывать свои риски от скачков цен (если цена актива резко вырастет — у них уже есть право купить по заранее оговорённой цене) и знать точную сумму, которую они могут потерять в худшем для них случае (если цена актива резко упадёт — они просто не будут исполнять опцион, а купят актив на рынке, таким образом потеряв сумму, равную цене опциона). Основная привлекательность опционов для покупателя объясняется тем, что ему заранее известен максимально возможный размер убытков — это величина, уплаченная за опцион, тогда как потенциальная прибыль

теоретически неограниченна – в случае значительного роста цены базовых акций в период действия опциона, покупатель может рассчитывать на высокую прибыль.

С другой стороны, при продаже опциона риски продавца возрастают. За счёт того, что рынок электроэнергии в Скандинавии является либеральным (не имеет строгого регулирования цен со стороны государства), цены на этот ресурс устанавливаются самим рынком на бирже (в результате уравнивания цен спроса и предложения). Для того, чтобы увеличить конкурентную привлекательность, разные компании, занимающиеся производством и/или продажей электроэнергии, предоставляют дополнительные услуги, одной и очень важной из которых является продажа опционов, другими словами, предоставление покупателю возможности застраховаться от резких скачков цены. Способность продавать опционы приносит компании, торгующей электроэнергией, не только конкурентную привлекательность, но и дополнительную прибыль, так как в большинстве случаев проданные опционы не исполняются (другими словами, страховой случай не наступает). В этой ситуации удовлетворёнными остаются обе стороны контракта, так как продавец опциона заработал сумму, равную цене проданного опциона, а покупатель всё это время был спокоен, будучи застрахованным от скачков цены, даже несмотря на то, что «худший сценарий» не наступил.

Опционы можно сравнить с любой страховкой, например, автомобильной. Самой сложной задачей является рассчитать справедливую цену (комиссию) этой страховки, при которой покупатель согласится застраховать свои риски, а продавец, даже в случае страховой выплаты, не понесёт значительных потерь и

сможет восполнить затраты за счёт комиссии с опционов, проданных на другие даты. Сравнивая с автомобильной страховкой – всегда можно продавать «КАСКО» в три раза дороже, чем это делают конкуренты, но, разумеется, в этом случае такую страховку никто не купит, а если продавать её дешевле, нужно понять, насколько именно дешевле, чтобы при наступлении страхового случая было возможно нивелировать затраты за счёт других полисов.

Таким образом, цена на опцион должна быть минимально возможной, чтобы одновременно обеспечивать конкурентоспособность и покрывать возможные риски. Справедливая цена опциона зависит от множества факторов. В первую очередь это:

1. Цена базового актива в данное время в данном месте (спот цена).
2. Цена, установленная в опционе, по которой владелец опциона имеет право купить актив, а владелец актива обязан его продать (страйк цена).
3. Показатель изменчивости цены базового актива за заданный промежуток времени (волатильность), как правило рассчитываемый статистическими методами на основе исторических показаний цены.
4. Время до срока истечения опциона (чем больше, тем неопределённость выше, соответственно, опцион дороже).

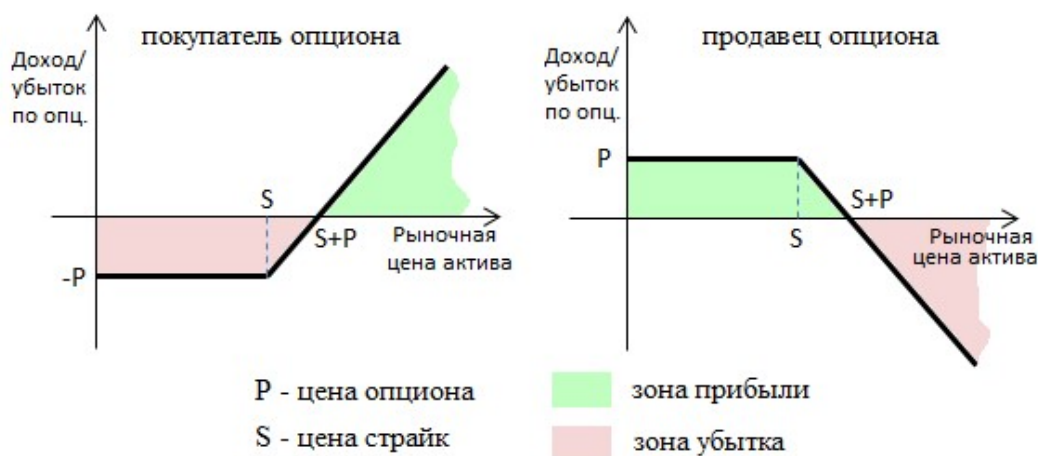


Рисунок 1 Зоны прибыли и убытка по опционам для покупателя и продавца в зависимости от цены базового актива на день исполнения

1.1. Постановка цели и задач

На данный момент мировым научным сообществом было предпринято множество попыток отыскать минимальную справедливую цену опциона, которая бы в то же время покрывала затраты, связанные со страховыми выплатами. Однако, данная задача, очевидно, невыполнима на сто процентов, так как её выполнение подразумевает возможность гарантированного предсказания будущей цены рынка. Если эта задача однажды будет решена, это уже не будет иметь никакого смысла, весь рынок ценных бумаг строится именно на наличии неопределённости и, если она исчезнет, все ценные бумаги мгновенно обесценятся. Таким образом, невозможно (и не нужно) предсказать будущее со сто процентной вероятностью, однако, можно немного приблизить методы к реальной жизни, тем самым увеличив точность прогноза на 1-2%.

В связи с этим был поставлен ряд задач, выполнение которых должно привести к достижению поставленной выше цели:

1. Во-первых, необходимо исследовать существующие модели и методы ценообразования опционов, выявить плюсы и минусы каждой, а также понять, какие части этих моделей и методов могут быть изменены, чтобы приблизить вычисления к «реальной жизни».
2. Во-вторых, для оценки моделей и методов и их возможных будущих доработок, необходимо создать систему, позволяющую проводить имитационное моделирование портфеля опционных позиций на предыдущие года, где цена базового актива в момент исполнения заранее известна.
3. В-третьих, проанализировав все модели и методы, нужно сделать предположение об внедрении таких изменений в них, которые бы приблизили их к «реальной жизни», запрограммировать эти изменения и провести имитационное моделирование, сделав вывод о жизнеспособности данных оптимизаций.
4. Реализовать систему поддержки принятия решений для менеджера, осуществляющего прямую продажу опционных контрактов на электроэнергию.

1.2. Введение в предметную область

Замечание: в рамках данной работы все термины рассматриваются только применительно к «колл» опционам (опционам на покупку) «европейского» типа, так как опционы на продажу («пут») не имеют отношения к розничной торговле электроэнергией.

В зависимости от отношения цены исполнения к текущей рыночной цене опционы делятся на три типа (Рисунок 2, Тип опциона в зависимости от отношения его цены к цене базового актива):

1. Опцион называется «вне денег», когда его цена страйк выше текущей рыночной цены базового актива, то есть покупатель опциона ожидает повышения этой цены. Данный тип является самым популярным, так как для этого и был формализован опцион.
2. Опцион является «ванильным» в том случае, если покупатель вовсе не ждёт изменения цены актива, однако, хочет застраховать себя от возможных рисков.
3. Последний тип опциона «в деньгах» появляется тогда, когда рынок ждёт снижения цены базового актива. Однако, на случай если этого не произойдёт, цена такого опциона всегда как минимум включает в себя разницу между ценой базового актива на дату заключения контракта и ценой исполнения опциона.

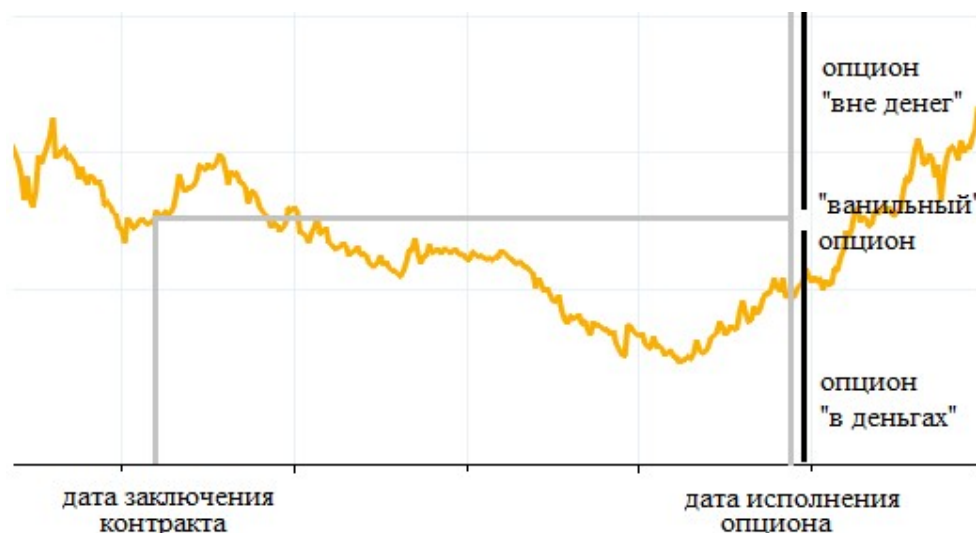


Рисунок 2 Тип опциона в зависимости от отношения его цены к цене базового актива

Цена (комиссия, премия) любого опциона складывается из двух составляющих – внутренней и временной стоимостей.

Внутреннюю стоимость имеет только опцион «в деньгах» и она равняется разнице между ценой базового актива и ценой страйк. Таким образом, данная величина может быть либо положительной, либо равной нулю. Опцион может менять свой тип после его заключения и до наступления даты его исполнения за счёт движения цены базового актива. Например, опцион, который изначально был «вне денег», за счёт падения цены актива может стать в деньгах, однако, премия, уплаченная за него, не изменяется, но покупатель может перепродать его дороже.

Второй составляющей цены опциона является его временная стоимость. Чем дольше времени до даты исполнения, тем больше шансов, что на цену базового актива повлияют какие-либо факторы, соответственно, опцион будет стоить дороже. Именно эта составляющая играет самую главную роль, а именно, покрывает риск продавца в случае наступления «страхового случая» (исполнения опциона). На рисунке 3 представлена схематичная зависимость временной стоимости опциона от времени до даты его исполнения.



Рисунок 3 График зависимости временной стоимости опциона от времени до даты его исполнения

Определение временной стоимости опциона является очень сложной и важной задачей. Кроме, непосредственно, времени, на временную стоимость влияет множество разных факторов:

1. Цена базового актива. Как и с любой страховкой, чем дороже объект страхования, тем дороже обойдётся страховой полис.
2. Цена страйк (желаемая цена исполнения опциона). Чем дальше опцион «вне денег», тем ниже его цена. Например, опцион на актив, торгующийся в данный момент по 30 у.е., с исторической волатильностью 20%, датой истечения через 30 дней и ценой страйк в 100 у.е. будет и вовсе бесплатным, так как шансы того, что цена вырастет до такого значения минимальны, и брать такой опцион не имеет смысла (конечно, если покупатель не обладает инсайдерской информацией, но использовать её при заключении биржевых операций – уголовное преступление).

3. Процентная ставка является малозначительным фактором, который, однако, нельзя убирать из рассмотрения. Под этим термином подразумевается безрисковая процентная ставка, то есть доходность, которую покупатель опциона мог бы занять, просто положив деньги в банк, то есть временная стоимость денег. При рассмотрении данного параметра, как правило, упускаются нюансы банковского обслуживания, такие как минимальный срок вклада, статус покупателя и прочие – подразумевается, что любой человек может положить деньги в банк на любой срок и приобрести выгоду, равную безрисковой процентной ставке. Данный параметр является константой (может зависеть от страны) и обычно принимается за 6%.
4. Волатильность базового актива – мера быстроты изменения цен на рынке. Этот фактор является наиболее важным при расчёте теоретической цены опциона, так как именно он отражает возможные риски изменения кривой цены. Волатильность отражает только изменчивость цены, но не направления этих изменений.



Рисунок 4 График цен на Bitcoin (левая ось) с годовыми волатильностями (правая ось)

Существует два вида волатильности – историческая и подразумеваемая. Под исторической волатильностью понимают статистический показатель среднего квадратичного отклонения цены актива за определённый промежуток времени в прошлом.

$$\delta = \sqrt{\frac{1}{n-1} \sum_{i=2}^n (u_i - \bar{u})^2} \quad (1)$$

$$\bar{u} = \frac{1}{n} \sum_{j=2}^n u_j \quad (2)$$

$$u_j = \frac{p_j - p_{j-1}}{p_{j-1}} \quad (3)$$

Где P_j – цена базового актива, u_0 не определено.

На рисунке 4 представлен график исторических цен на Bitcoin по левой оси (синим цветом) и годовые волатильности по правой (красным цветом), из которого видно, что волатильность была высокой во время развития и бурного роста этого актива и заметно уменьшилась к 2014 году.

Подразумеваемая волатильность не является детерминированной величиной и зависит от текущего положения дел. Для учёта подразумеваемой волатильности можно применять различные методики искусственного интеллекта, нейросетей, например, для анализа новостных сводок либо текущих цен на другие, смежные активы. Рассмотрение способов вычисления подразумеваемой волатильности выходит за рамки данной работы.

1.3. Существующие методы ценообразования опционов

Методы ценообразования опционов – это методы, объясняющие случайный порядок формирования стоимости опциона. Инвесторы и трейдеры активно используют такие методы для расчета потенциальной прибыли от сделок с опционом. На данный момент существует несколько универсальных моделей, наиболее популярные из которых представлены на рисунке 5.

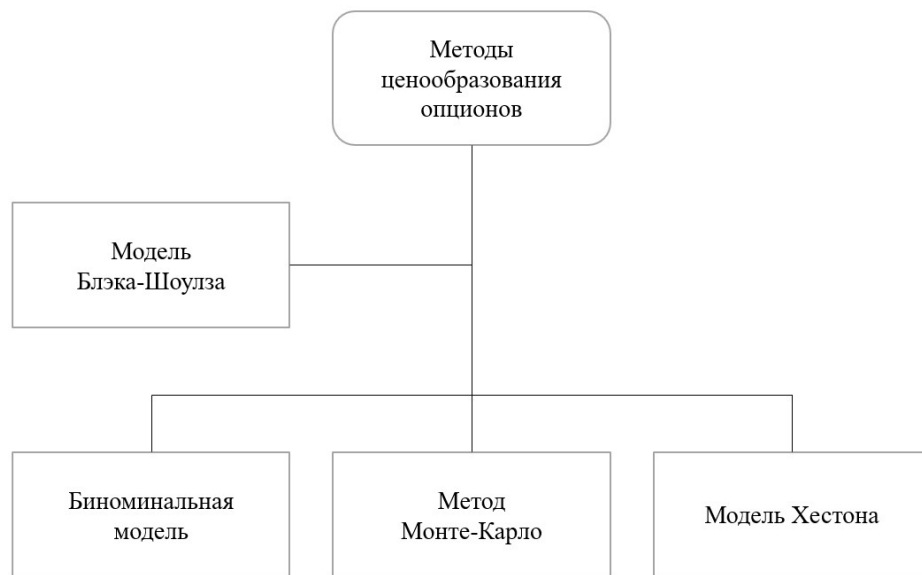


Рисунок 5 Методы ценообразования опционов

Существующие методы оценки цены опционов можно разделить на две основные группы – аналитические и численные. В аналитических методах оценка происходит с помощью математически выведенных формул, которые позволяют мгновенно получить теоретическую стоимость опциона. Один из примеров – модель Блэка-Шоулза. Численные методы включают в себя биномиальную модель и метод Монте-Карло. Их достоинство в том, что, применяя их, есть возможность влиять на их поведение изменяя те или иные параметры оценки, тогда как в аналитических методах изменять можно лишь параметры функции, заложенные в неё создателем.

«Современное управление рисками, применяемое в страховании, торговле на фондовом рынке и инвестировании, основывается на возможности использовать математические методы для предсказания будущего. Конечно, не со 100%-ной вероятностью, но достаточно точно для того, чтобы принять взвешенное инвестиционное решение.

Основополагающий принцип работы на финансовых рынках состоит в

следующем: чем больший риск вы готовы на себя принять, тем на большее вознаграждение вы вправе рассчитывать. Использование математики никогда не сможет полностью элиминировать риск, но может помочь правильно оценить степень принимаемого на себя риска и решить вопрос о справедливом вознаграждении.» [6, с. 5].

1.4. Модель Блэка-Шоулза

В 1973 году Ф. Блэк, М. Шоулз и Р. Мертон вывели универсальную математическую формулу, позволяющую вычислять стоимость опционов и других производных инструментов, впоследствии оказавшую огромное влияние на развитие мирового финансового рынка. Создание этой формулы привело к стремительному увеличению интереса к опционной торговле, позволив отойти от оценки их стоимости, полагаясь лишь на интуицию и субъективные ощущения. Сама идея использования математического аппарата для оценки стоимости опционов была на тот момент революционна. На рисунке 5 эта модель неслучайно стоит отдельно от остальных – она не просто была первой и послужила началом применения математического аппарата в этой сфере, но и явилась основой для остальных методов. Базой для модели Блэк и Шоулз приняли тот факт, что цена опциона увеличивается и уменьшается прямо пропорционально стоимости базового актива, а основным фактором является волатильность его цены.

Формула вычисления стоимости опциона для модели Блэка-Шоулза:

$$C = S N(d_1) - N(d_2)Ke^{-rt} \quad (4)$$

$$d_1 = \frac{\ln\left(\frac{S}{K}\right) + \left(r + \frac{\delta^2}{2}\right)t}{\delta\sqrt{t}} \quad (5)$$

$$d_2 = d_1 - \delta\sqrt{t} \quad (6)$$

- C – Стоимость опциона;
- S – Текущая рыночная цена соответствующего опциону актива;
- K – Цена, установленная в опционе (страйк);
- t – Время до окончания существования опциона, выраженное в годовой пропорции;
- r – Безрисковая процентная ставка;
- N – Стандартное нормальное интегральное распределение;
- δ – Волатильность цены актива.

Стандартное нормальное интегральное распределение вычисляется по следующей формуле:

$$N(x) = \frac{1}{2} \left(1 + \operatorname{erf}\left(\frac{x}{\sqrt{2}}\right) \right) \quad (7)$$

где erf – функция ошибок (функция Лапласа или интеграл вероятности) — это неэлементарная функция, возникающая в теории вероятностей, статистике и

теории дифференциальных уравнений в частных производных. Она определяется следующим образом:

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_x^{\infty} e^{-t^2} dt \quad (8)$$

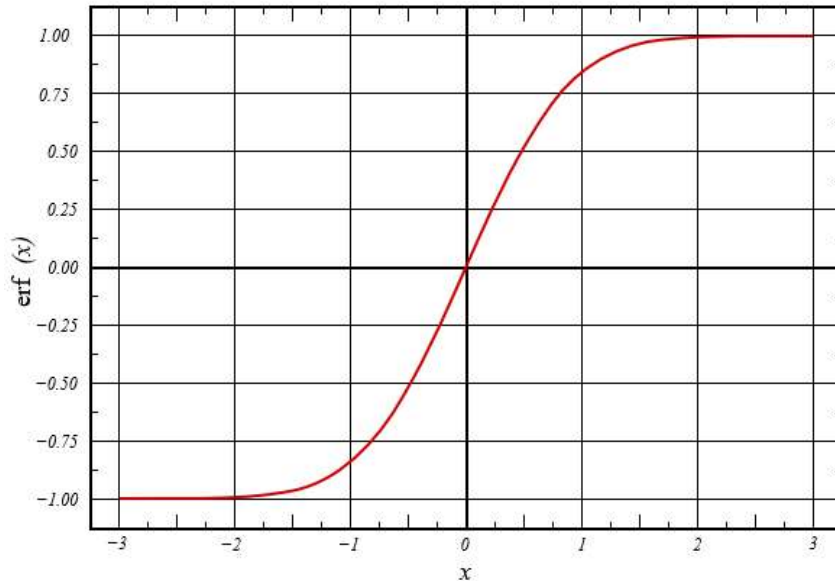


Рисунок 6 График функции ошибок erf

Функция ошибок не может быть представлена через элементарные функции, но, разлагая интегрируемое выражение в ряд Тейлора и интегрируя почленно, мы можем получить её представление в виде ряда:

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} e^{-x^2} \sum_{n=0}^{\infty} \frac{2^n x^{2n+1}}{(2n+1)!!} \quad (9)$$

Это равенство выполняется (и ряд сходится) как для любого вещественного x , так и на всей комплексной плоскости. Для практического вычисления значения необходимо определиться с числом n – количеством итераций вычисления

функции ошибки. Так как ряд сходящийся, чтобы гарантированно достичь максимально возможной точности вычисления, необходимо итерироваться до тех пор, пока слагаемое не будет равно нулю (в зависимости от используемого типа данных). Как только цикл достигает такой ситуации – точнее вычислить значение работая с этим типом данных невозможно.

Как и у любой модели, у формулы Блэка-Шоулза есть несколько различных по степени важности допущений.

1. По базовому активу не выплачиваются дивиденды, другими словами, обладание активом не приносит владельцу никакой прибыли, кроме непосредственно изменения рыночной цены на этот актив.
2. Модель не учитывает транзакционные затраты на покупку или продажу актива и опциона, такие как различные комиссии и прочие.
3. Безрисковая процентная ставка известна и не изменяется на протяжении действия опциона.
4. Любой покупатель опциона может получать ссуды по безрисковой процентной ставке независимо от срока действия опциона.
5. Торговля базовым активом ведётся непрерывно и движения его цен распределены по нормальному закону.

С точки зрения продажи опционов на фьючерсные контракты на электроэнергию, последнее, пятое допущение имеет наибольшую значимость. Возникает вопрос, а что произойдёт, если движения цен не будут распределены по нормальному закону? Этот вопрос будет рассмотрен в следующей главе.

1.5. Метод Монте-Карло

Выкладки, произведённые Блэком и Шоулзом привели учёных экономистов к возможности использования численных методов Монте-Карло для оценки опционных контрактов. Этот метод заключается в оценке математического ожидания страховой выплаты по опциону путём многократного генерирования возможных путей движения цены акции.

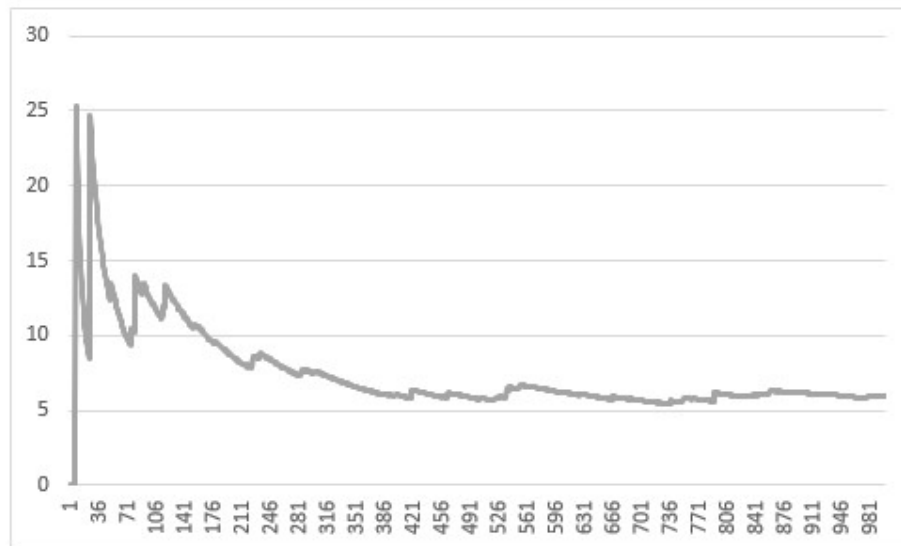


Рисунок 7 Увеличение точности оценки стоимости опциона методом Монте-Карло с ростом количества итераций

Генерация случайного значения цены акции на момент исполнения опциона в рамках одной итерации в симуляции Монте-Карло происходит по следующей формуле:

$$C = S k^e \quad (10)$$

$$k = \left(r - \frac{\delta^2}{2} \right) t + N \delta \sqrt{t} \quad (11)$$

- C – Стоимость опциона;
- S – Текущая рыночная цена соответствующего опциону актива;
- t – Время до окончания существования опциона, выраженное в годовой пропорции;
- r – Безрисковая процентная ставка;
- δ – Волатильность цены актива;
- N – Случайная величина, имеющая стандартное нормальное распределение.

«Данная формула соответствует общепринятой модели движения цены акции, которая лежит в основе модели Блэка-Шоулза и многих других моделей оценки деривативов, и является одним из «столпов» современной деривативной теории. Математически данная формула является решением стохастического дифференциального уравнения, описывающего движение цены акции.» [8]

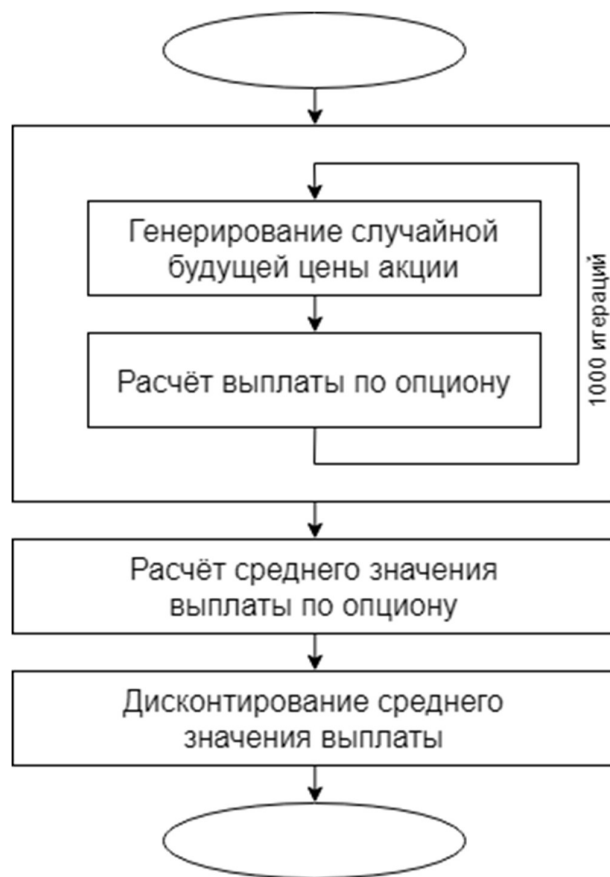


Рисунок 8 План оценки опциона методом Монте-Карло

В качестве плюсов этого метода можно отметить:

1. Интуитивную понятность – в качестве стоимости опциона выступает сумма, которую покупатель заработает, купив его
2. Универсальность и гибкость – в отличие от формулы Блэка-Шоулза, метод Монте-Карло может быть изменён под любую конкретную задачу с учётом специфики рассматриваемого рынка, типа опциона и пр.

Из минусов можно отметить ресурсоёмкость – чтобы посчитать цену достаточно точно, необходимо сделать порядка ста тысяч итераций.

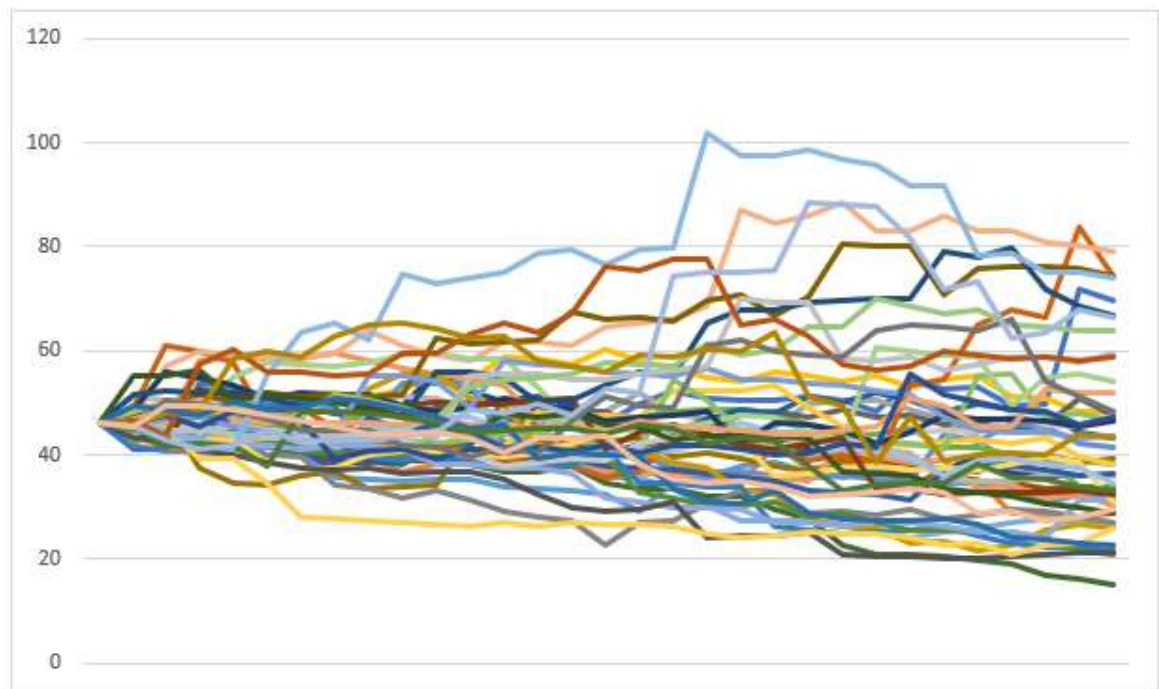


Рисунок 9 Изображение 50-ти итераций генерации случайной цены актива на дату исполнения методом Монте-Карло

1.6. Имитационное моделирование

Имитационное моделирование (simulation) является одним из мощнейших методов анализа экономических систем.

В общем случае, под имитацией понимают процесс проведения на ЭВМ экспериментов с математическими моделями сложных систем реального мира.

Цели проведения подобных экспериментов могут быть самыми различными – от выявления свойств и закономерностей исследуемой системы, до решения конкретных практических задач. С развитием средств вычислительной техники и программного обеспечения, спектр применения имитации в сфере экономики

существенно расширился. В настоящее время ее используют как для решения задач внутрифирменного управления, так и для моделирования управления на макроэкономическом уровне. Рассмотрим основные преимущества применения имитационного моделирования в процессе решения задач финансового анализа.

Как следует из определения, имитация – это компьютерный эксперимент. Единственное отличие подобного эксперимента от реального состоит в том, что он проводится с моделью системы, а не с самой системой. Однако проведение реальных экспериментов с экономическими системами, по крайней мере, неразумно, требует значительных затрат и вряд ли осуществимо на практике. Таким образом, имитация является единственным способом исследования систем без осуществления реальных экспериментов.

Часто практически невыполним или требует значительных затрат сбор необходимой информации для принятия решений. Например, при оценке риска инвестиционных проектов, как правило, используют прогнозные данные об объемах продаж, затратах, ценах и т.д.

Однако, чтобы адекватно оценить риск необходимо иметь достаточное количество информации для формулировки правдоподобных гипотез о вероятностных распределениях ключевых параметров проекта. В подобных случаях отсутствующие фактические данные заменяются величинами, полученными в процессе имитационного эксперимента (т.е. сгенерированными компьютером).

При решении многих задач финансового анализа используются модели, содержащие случайные величины, поведение которых не поддается управлению со стороны лиц, принимающих решения. Такие модели называют стохастическими. Применение имитации позволяет сделать выводы о возможных результатах, основанные на вероятностных распределениях случайных факторов (величин). Стохастическую имитацию часто называют методом Монте-Карло.

Имитационное моделирование представляет собой серию численных экспериментов призванных получить эмпирические оценки степени влияния различных факторов (исходных величин) на некоторые зависящие от них результаты (показатели).

В общем случае, проведение имитационного эксперимента можно разбить на следующие этапы.

1. Установить взаимосвязи между исходными и выходными показателями в виде математического уравнения или неравенства.
2. Задать законы распределения вероятностей для ключевых параметров модели.
3. Провести компьютерную имитацию значений ключевых параметров модели.
4. Рассчитать основные характеристики распределений исходных и выходных показателей.
5. Провести анализ полученных результатов и принять решение.

Результаты имитационного эксперимента могут быть дополнены статистическим анализом, а также использоваться для построения прогнозных моделей и сценариев.

«В экономике метод имитационного моделирования можно рассматривать как определяющий и наиболее эффективный подход к разработке обоснованных и оптимальных по совокупности критериев финансового, технологического, экологического и прочего характера бизнес-планов и инвестиционных проектов» [3].

2. ОСНОВНАЯ ЧАСТЬ

2.1. Имитационное моделирование портфеля опционных позиций

Для того, чтобы сравнить результаты рассмотренных моделей в «реальной жизни», а также иметь возможность оценить производимые доработки моделей, необходимо прибегнуть к так называемому имитационному моделированию. Сейчас 2018 год, а это значит, что все цены на фьючерсные контракты на 2017 год уже известны и могут быть скачаны через API с биржи NASDAQ OMX. Таким образом мы можем представить, что мы находимся в любой точке 2017 года, например, в марте, и продаём опцион на фьючерсный контракт на, например, июнь. За счёт того, что на данный момент у нас есть абсолютно все цены на 2017 год, мы можем посчитать цену опциона на основе волатильности по ценам, которые были известны в марте, а затем выяснить, наступил ли страховой случай по этому контракту, так как мы, находясь в 2018 году, уже знаем реальную цену на июнь 2017. Собрав случайный портфель из трёхсот проданных опционов на каждый месяц 2017 года можно собрать статистику выигрышных и проигрышных контрактов для продавца в зависимости от используемого метода оценки опциона.



Рисунок 10 Пример построения портфеля опционов на 2017 год (красные точки - даты опционного контракта)

Среди потребителей электроэнергии в Скандинавии есть как промышленные объекты (различные заводы, производства, торговые центры, морские порты, коттеджные посёлки), так и розничные потребители (частные дома, квартиры). Однако, как показывает практика, в большинстве своём к страхованию рисков прибегают те потребители, чей месячный объём потребления превосходит 30-50 МВт*ч, так как иначе прибыли и риски настолько малы, что не имеет большого смысла прибегать к их регулированию. Таким образом, для имитационного моделирования можно взять, например, 25 различных промышленных предприятий со случайным месячным объёмом потребления от 50 до 250 МВт*ч, которые хотят застраховать свои риски от 5 до 30 процентов повышения цены (цена Страйк) и заключают контракты за 20-80 дней до начала месяца поставки.

При сравнении модели Блэка-Шоулза и стандартной реализации метода Монте-Карло различий в цене опциона, прибыли и убытках выявлено не было. Оба метода проявляют себя хорошо, давая в среднем около 8% убыточных

контрактов, убытки по которым с лихвой покрываются премией по неисполненному опциону (рисунок 11).



Рисунок 11 Соотношение прибыльности опционов в моделируемом портфеле

Таблица 1 Анализ результатов имитационного моделирования

Месяц	Количество выплат	Средняя волатильность	Средняя цена опциона	Средняя продолжит. прибыльного опциона	Средняя продолжит. убыточного опциона
1	0	32.73%	0.53	48	0
2	0	39.76%	0.77	51	0
3	2	55.40%	1.43	52	49
4	4	63.88%	1.61	51	61
5	11	63.32%	1.27	56	40
6	0	59.21%	0.80	44	0
7	16	49.76%	0.79	50	51
8	12	35.92%	0.48	46	52
9	12	28.21%	0.35	51	49
10	1	25.92%	0.24	48	45
11	4	23.24%	0.25	51	22
12	0	22.19%	0.06	53	0
Общее	62		0.71	50	46

В таблице 1 представлен анализ результатов имитационного моделирования. Отсюда видно, что прибыльность или убыточность не коррелирует ни с продолжительностью опциона ни с волатильностью. Однако можно заметить, что подавляющее количество страховых выплат приходится на май, июль, август и сентябрь.

2.2. Выбор распределения изменений цен

Одним из допущений обоих рассмотренных методов оценки цены опциона является предположение о том, что изменения цены базового актива распределены в соответствии с нормальным законом. Выбор распределения – одна из главных проблем при определении справедливой цены опциона. Нормальное распределение было выбрано Блэком и Шоулзом за счёт того, что оно наиболее точно описывает поведение цены на среднестатистический актив с учётом его броуновского движения. Но для того, чтобы понять, подчиняется ли этому закону цена на электроэнергию в странах Скандинавии, необходимо провести исследование.

Для начала можно рассмотреть графики цены и распределения её дневных изменений для нескольких месяцев 2017 года (рисунок 12).

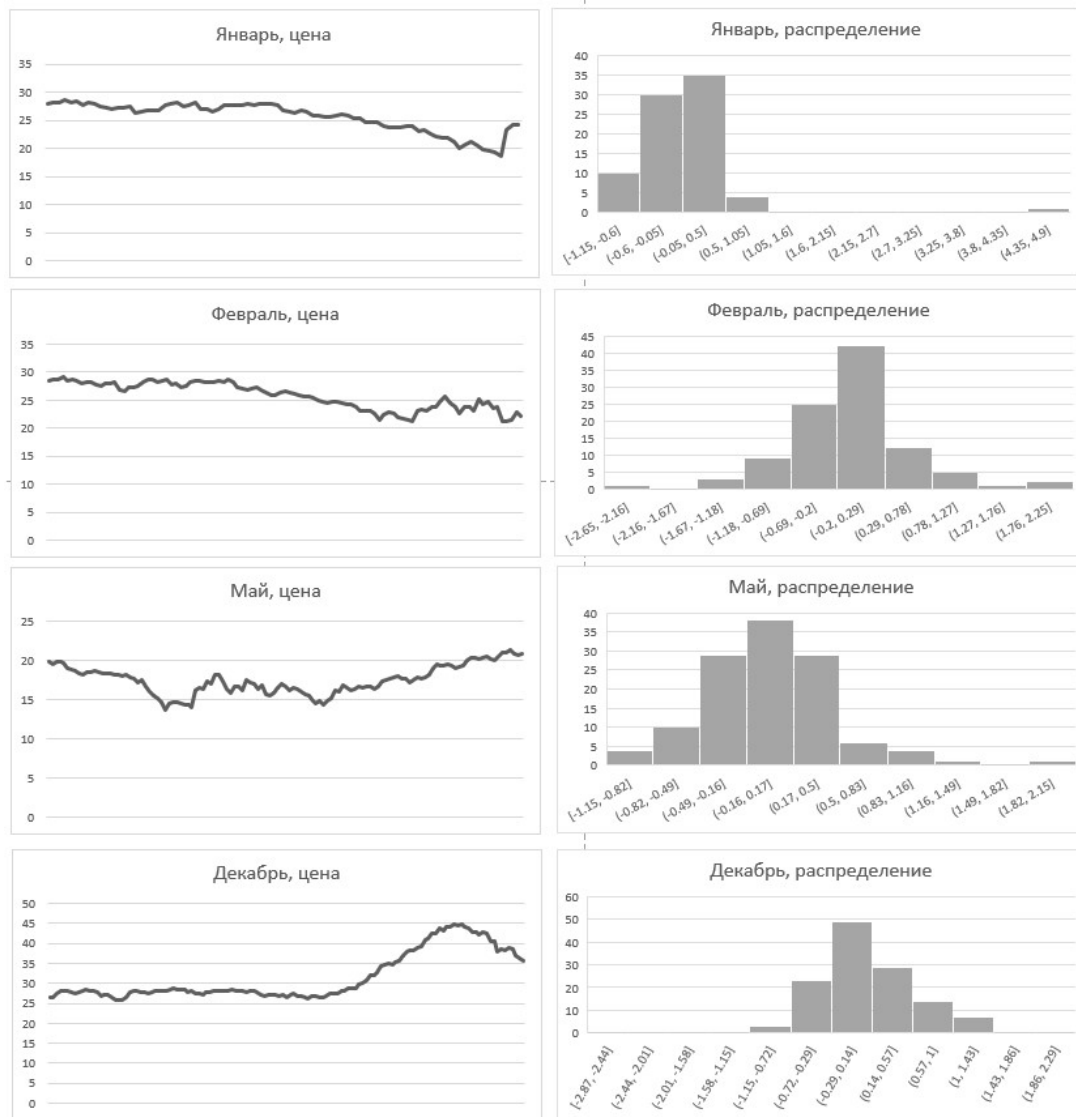


Рисунок 12 Графики цены и распределения её дневных изменений за несколько месяцев 2017 года

Как можно заметить, распределение отличается от месяца к месяцу и является либо более высоким, либо более пологим, чем нормальное распределение. Рассмотрев распределения дневных изменений по месяцам 2016 и 2017 года можно увидеть, что зависимость действительно имеет место быть (графики распределений для каждого месяца представлены в приложении А). Но для того, чтобы более детально проанализировать эту зависимость, необходимо

каким-либо образом формализовать «пологость» этих распределений. В идеале необходимо получить одно число, которое бы отражало эту характеристику.

2.3. Индекс толщины «хвоста» распределения

В статье «How fat are your tails?» (рус. – «Как толсты твои хвосты?») М. Ашер предлагает использовать отношение среднеарифметического абсолютного отклонения (Mean Absolute Deviation, MAD) к среднеквадратичному (Standard Deviation, SD) в качестве индекса толщины «хвоста» любого распределения [19]. По его словам, MAD измеряет среднее абсолютное расстояние между случайной величиной и её средним значением. В отличие от SD, MAD не содержит квадратов, что делает его более устойчивым к резким скачкам, в результате чего мы можем использовать это соотношение в качестве индекса «толщины хвоста». Для нормального распределения индекс MAD/SD равняется 0.7970, что является квадратным корнем из $2/\pi$.

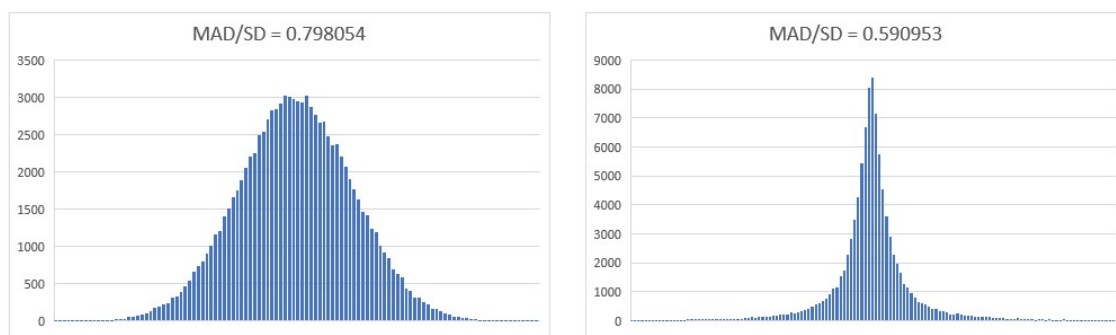


Рисунок 13 Сравнение отношения MAD/SD для двух случайных распределений

Воспользовавшись этим соотношением, мы можем составить таблицу индексов «хвоста» для цен на электричество по месяцам 2016 и 2017 годов.

Таблица 2 Индексы "хвоста" распределения цен по месяцам 2016 и 17 годов

	2016	2017
Январь	0.5846	0.6936
Февраль	0.6997	0.6787
Март	0.7300	0.7289
Апрель	0.7690	0.7231
Май	0.7523	0.7290
Июнь	0.7641	0.7084
Июль	0.7657	0.7697
Август	0.7523	0.7757
Сентябрь	0.7108	0.7617
Октябрь	0.7658	0.7548
Ноябрь	0.7370	0.7545
Декабрь	0.7051	0.7252

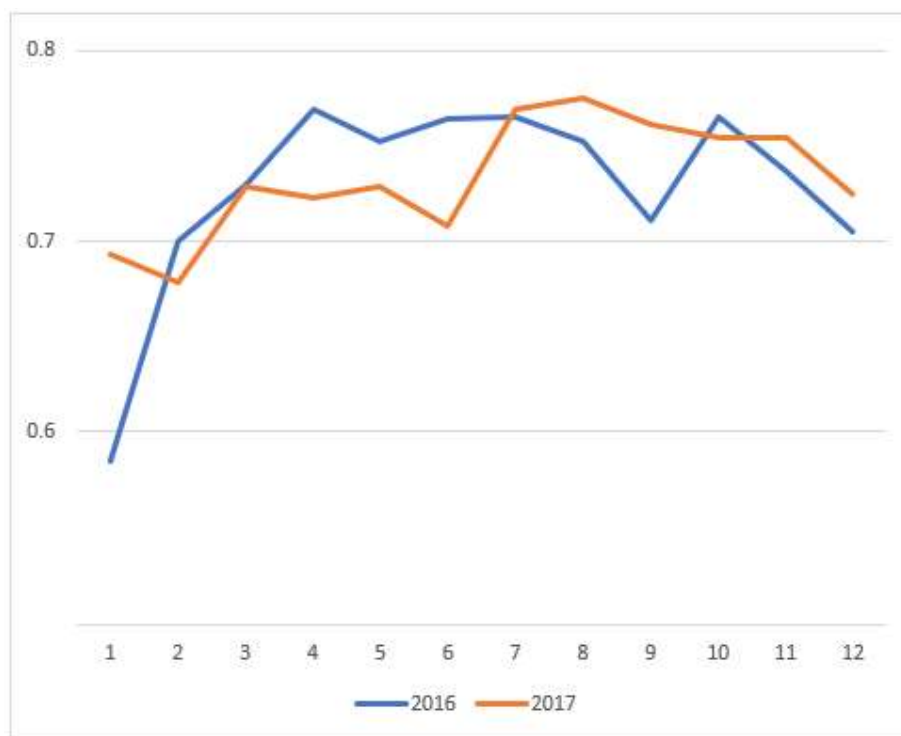


Рисунок 14 График индекса "хвоста" распределения цен по месяцам 2016 и 17 годов

При рассмотрении большего количества лет прослеживается всё большая зависимость – индекс больше в летние месяцы, что говорит о том, что летом прыжки цен имеют наибольший вес. Однако, чтобы сделать какие-либо выводы, необходимо рассмотреть, как ведёт себя волатильность от месяца к месяцу.

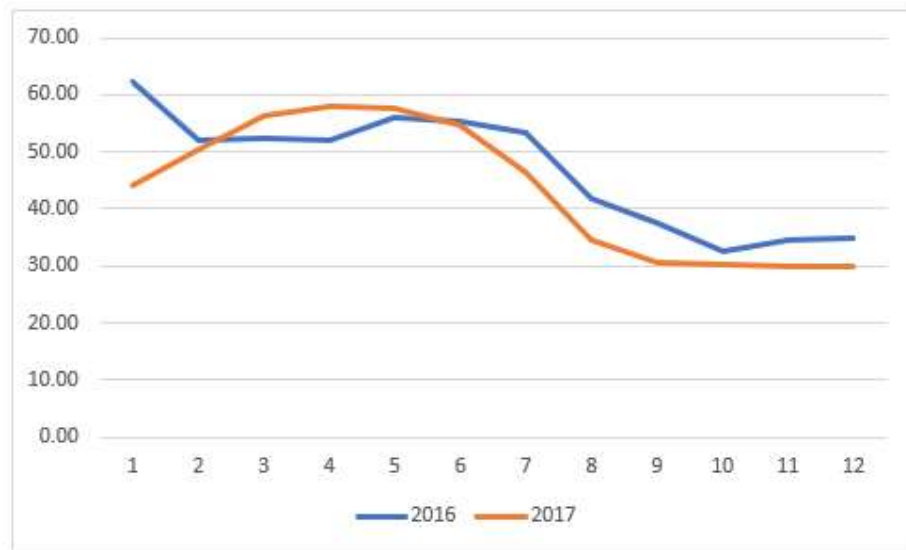


Рисунок 15 График изменения волатильности по месяцам 2016 и 17 годов

Как можно заметить, волатильность, как и индекс «хвоста», имеет определённую зависимость от года к году, но эта зависимость мало перекликается с зависимостью индекса MAD/SD.

2.4. Зависимость индекса MAD/SD от волатильности

Для того, чтобы понять, как зависит индекс MAD/SD от волатильности, необходимо рассмотреть несколько абстрактных примеров.

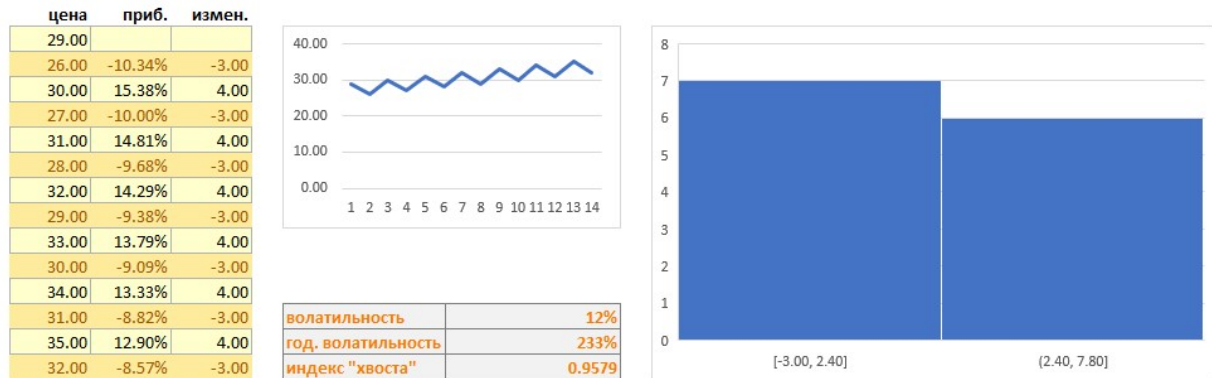


Рисунок 16 Индекс MAD/SD и волатильность для равномерно растущей цены

На рисунке 16 представлен график изменения цены на абстрактный актив, цена которого колеблется вверх/вниз каждый день, но при этом стабильно растёт. Волатильность такого поведения невелика – всего 12%, тогда как индекс «хвоста» почти максимальный – 0.96. Связано это с тем, что хвоста как такового и нет – всё распределение представляет из себя две точки с огромным количеством элементов в них.

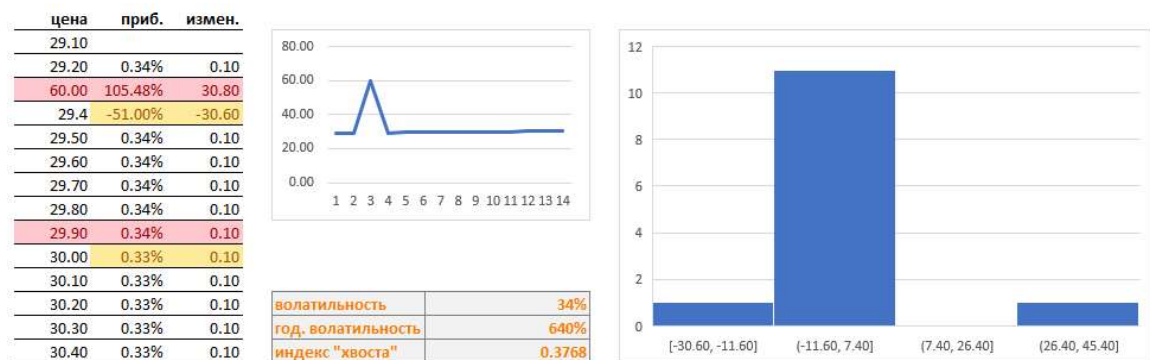


Рисунок 17 Индекс MAD/SD и волатильность для очень плавно растущей цены, но с резким скачком в один из дней

На рисунке 17, в отличие от предыдущего примера, цена растёт очень плавно и равномерно, кажется, что мы знаем, что завтра тоже ничего не изменится. Но

есть одно «но» – в один из дней рассматриваемого периода произошёл неожиданный скачок, который на следующий же день нейтрализовался. На рынке такое может быть связано с, например, выходом в свет какой-либо новости о неблагоприятном изменении состояния для актива (для электроэнергии это может быть новость о том, что зима будет очень холодной), а на следующий день может прийти опровержение этой информации. К сожалению, такое случается достаточно часто, но проблема заключается в том, что волатильность такого не прощает. Как видно из примера, волатильность для этого абстрактного актива составила аж 34%, что почти в три раза выше предыдущего примера, который постоянно находится в состоянии стресса. Однако, индекс MAD/SD крайне низок – 0.3768, что говорит об очень тонких «хвостах» распределения.

Из рассмотренного примера можно сделать вывод, что волатильность не является «серебряной пулей» при анализе текущего положения дел на рынке.

2.5. Использование реального распределения цен актива при расчёте цены опциона методом Монте-Карло

Как уже было сказано, одним из плюсов метода Монте-Карло перед моделью Блэка-Шоулза является возможность его адаптации к конкретному типу актива. Например, в генераторе случайных цен можно заменить стандартное их распределение на любое другое, более подходящее для конкретного актива и даже времени года.

Если сделать предположение, что опционы на некоторые месяцы, используя нормальное распределение, переоценены за счёт резких дневных скачков и

возвращения цены обратно через некоторое время (увеличенная волатильность), но при этом имеют низкий MAD/SD индекс распределения, можно попробовать использовать распределение, более точно описывающее текущую ситуацию на рынке., после чего провести ряд испытаний и сравнить результаты с моделью Блэка-Шоулза, предполагающую нормальное распределение.

Таким образом, нужно найти такое распределение, которое бы имело параметр, позволяющий регулировать «толщину хвоста», но в то же время, как и нормальное распределение, было симметрично относительно нуля (несимметричные распределения, например, гамма, тоже имеют место существовать в финансовом мире, но их исследование выходит за рамки данной работы).

Одним из таких распределений является распределение Коши (частный случай Т-распределения Стьюдента с одной степенью свободы), однако, его проблема заключается в том, что его математическое ожидание не определено, а дисперсия бесконечна.

$$(\pi\gamma(1 + (\frac{x-m}{\gamma})^2))^{-1} \quad (12)$$

Где m – коэффициент сдвига, а $\gamma > 0$ – коэффициент масштаба. Для нашего исследования сдвиг не нужен, поэтому во всех дальнейших формулах он будет равен 0.

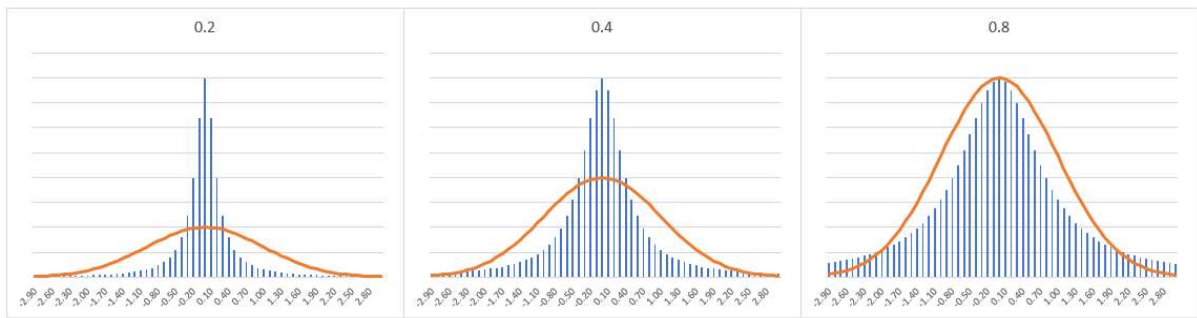


Рисунок 18 Сравнение распределения Коши с нормальным распределением при разных коэффициентах масштаба. Коши – столбчатая диаграмма, Нормальное распределение – линия.

За счёт того, что дисперсия распределения Коши бесконечна, в процессе генерации случайной последовательности, подчиняющейся закону распределения Коши (преобразования равномерно распределённой последовательности в распределение Коши), с небольшой вероятностью появляются очень большие и очень малые числа, которые делают весь алгоритм непригодным, так как бесконечно большое число в стоимости актива, даже если это одна итерация из миллиона, делает опцион бесконечно дорогим.

«Поскольку на практике это распределение всегда ограничено, не следует беспокоиться о длинных хвостах, которые приводят к бесконечной дисперсии, а достаточно проверить, верно ли поведение $q(\gamma)$ при малых значениях γ .» [20].

В работе «Making the Cauchy work» (рус. «Делаем так, чтобы распределение Коши работало» [12], С. Надараджа описывает поведение ограниченного распределения Коши. Это распределение имеет следующую функцию распределения вероятности:

$$f(x; A, B) = \frac{1}{\gamma D} \left(1 + \left(\frac{x}{\gamma} \right)^2 \right)^{-1} \quad (13)$$

$$\text{где } D(A, B) = \arctan\left(\frac{B}{\gamma}\right) - \arctan\left(\frac{A}{\gamma}\right) \quad (14)$$

Это распределение наиболее часто используется на практике, так как имеет конечную дисперсию и мат ожидание, при этом сохраняя возможность управлять толщиной его «хвоста» за счёт изменения параметра γ . Для быстрой генерации этого распределения можно использовать метод «отклонения неподходящих» величин, генерируя стандартное распределение Коши и отклоняя значения, не попадающие в $[-A; B]$. Например, ограничивая распределение Коши с $\gamma = 0.5$ для промежутка $[-3; 3]$, вероятность выпадения числа из интервала менее 1.72%, при этом всё, что нужно сделать – сгенерировать новое число – это будет быстрее, чем вычислять число по формуле, предложенной С. Надараджа, за счёт того, что она требует вычисления тангенса:

$$X = \gamma * \tan(DU + \arctan(1)) \quad (15)$$

где U – равномерно распределённая случайная величина на отрезке $[0; 1]$.

Генерация стандартного (не ограниченного) распределения Коши в общем случае представляет собой деление двух нормально распределённых чисел друг на друга и умножение на γ , что является более быстрым алгоритмом, если использовать полярный алгоритм Марсальи для генерации нормально распределённой величины.

Таким образом, можно использовать распределение Коши, ограниченное до «трёх сигм», в качестве распределения прибыльностей в Монте-Карло. Однако, необходимо выбрать параметр масштаба для каждого конкретного случая в зависимости от «индекса хвоста» реального распределения цен на рынке.

2.6. «Индекс хвоста» ограниченного распределения Коши

Как уже было сказано, «индекс хвоста» рассчитывается с помощью деления среднеарифметического абсолютного отклонения (MAD) к среднеквадратичному ($SD = \sqrt{\delta}$).

Для вычисления этих параметров из имеющейся функции распределения вероятности для ограниченного распределения Коши можно использовать следующие формулы:

$$\delta = \int_A^B x^2 * f(x) dx \quad (16)$$

$$MAD = \int_A^B |x| * f(x) dx \quad (17)$$

За счёт симметричности нашего отрезка, интегралы можно решать в интервале от 0 до B и умножать на 2, после чего получаем следующую формулу для «индекса хвоста» ограниченного распределения Коши:

$$\frac{MAD}{SD} = \frac{\ln\left(\frac{\gamma^2+B^2}{\gamma^2}\right) \sqrt{\frac{B\gamma-\gamma^2 \arctan\left(\frac{B}{\gamma}\right)}{\arctan\left(\frac{B}{\gamma}\right)}}}{2B-2\gamma \arctan\left(\frac{B}{\gamma}\right)} \quad (18)$$

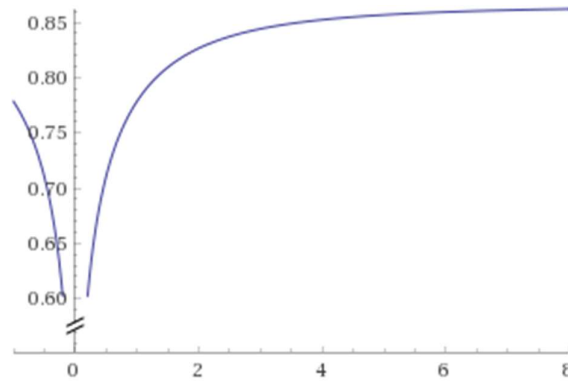


Рисунок 19 График зависимости "индекса хвоста" от коэффициента масштаба для распределения Коши, при $B = 3$

Таким образом, посчитав индекс MAD/SD для реального распределения цен на рынке, мы можем с высокой точностью определить необходимый коэффициент масштаба для ограниченного распределения Коши.

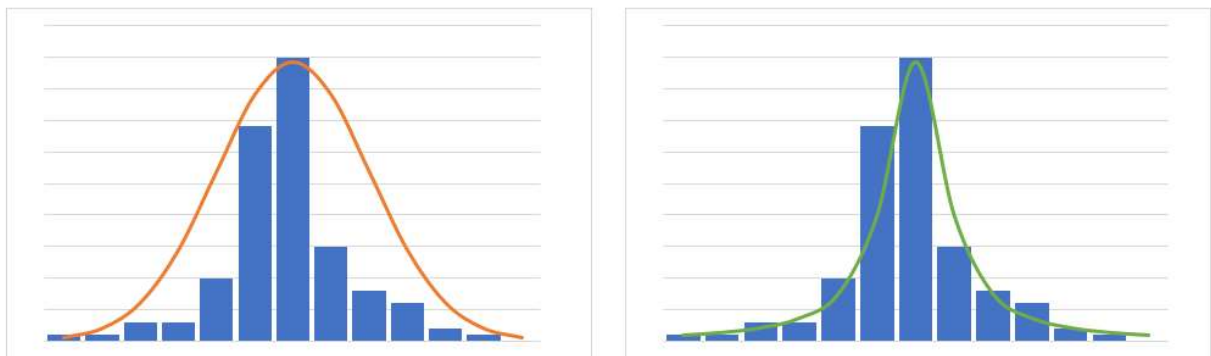


Рисунок 20 Сравнение реального распределения цен на январь 2017 года (столбчатая диаграмма) с нормальным распределением (линия слева) и с

ограниченным распределением Коши (линия справа) с подсчитанным коэффициентом масштаба

2.7. Имитационное моделирование портфеля с использованием ограниченного распределения Коши

Применив распределение Коши в методе Монте-Карло, предварительно рассчитав не только волатильность по ценам, известным на момент заключения сделки, но и «индекс хвоста», можно снова провести имитационное моделирование и посмотреть, как это распределение влияет на окончательную цену опциона.

После ряда испытаний был сделан вывод, что использование реального соотношения MAD/SD действительно уменьшает цену опциона для месяцев с длинными, но тонкими «хвостами» и при этом же относительно большой волатильностью. Однако, для месяцев с толстыми «хвостами» цена опциона растёт, но недостаточно, чтобы избежать всех страховых выплат. Таким образом, можно применять комбинированное решение, а именно, брать наименьшую цену между расчётами Монте-Карло и Блэка-Шоулза, что создаст большую конкурентную привлекательность, при этом практически не изменив показатели убыточности.

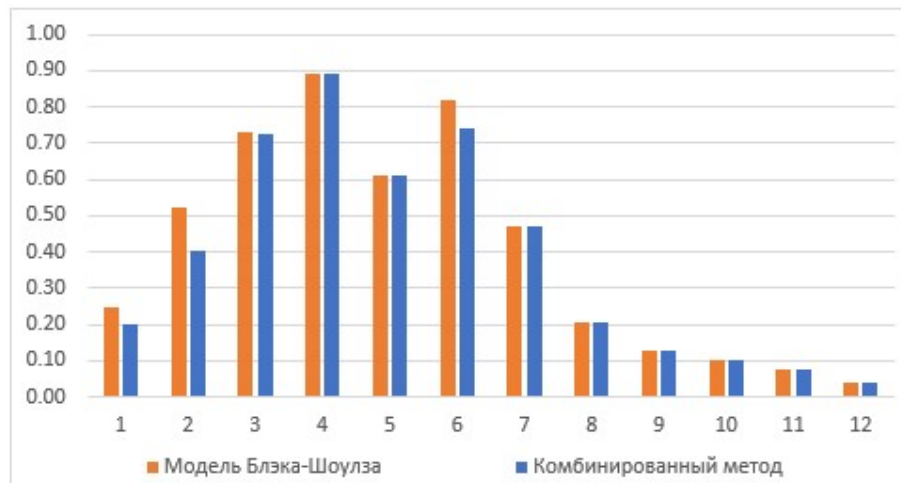


Рисунок 21 Сравнение средняя цена опциона по месяцам 2017 года (оранжевый столбец - модель Блэка-Шоулза, синий столбец - комбинированный метод с ограниченным распределением Коши)

На рисунке 21 представлено сравнение средних цен опциона при имитационном моделировании портфеля на 2017 год. Как можно заметить, на некоторые, наиболее волатильные месяцы (январь, февраль, июнь) действительно за счёт применения более подходящего распределения удаётся снизить цену опциона, при этом не изменив количества убыточных контрактов.

Таким образом можно сделать вывод, что использования рассчитанного распределения пусть и не сильно, но помогает снизить цену опциона, незначительно увеличив риск, что делает его достаточно безопасным методом для применения на практике.

2.8. Заказчик

В рамках данной работы информационная система создаётся по заказу норвежского разработчика программных решений для продажи электроэнергии.

Эта компания занимается разработкой и поддержкой ПО для большого числа продавцов электроэнергии в Скандинавии.

2.9. Внедрение в СППР

Система поддержки принятия решений в компании представляет из себя веб-сайт, позволяющий менеджерам компании регистрировать как простые сделки на поставку электроэнергии, так и с возможностью фиксирования максимальной цены, то есть предоставляя возможность заключения опционных контрактов.

Основные функции информационной системы – вычисление параметров сделки, отображение вычисленной информации в графическом виде, отправка деталей зарегистрированной сделки.

Так как вся программная экосистема компании представлена программными продуктами Microsoft, СППР выполнена с применением технологии ASP.NET MVC.

В простейшем случае вариант использования подсистемы выглядит так: менеджер, в обязанности которого входит обслуживание оптовых клиентов по вопросам продажи электроэнергии, использует ПК. К нему обращается клиент, желающий заключить контракт на поставку электроэнергии. Они вместе взаимодействуют с графическим интерфейсом, устанавливая необходимые параметры сделки, такие как количество электроэнергии по месяцам, дату начала, дату окончания, необходимую зону поставки (в зависимости от страны и региона) и валюту сделки. После того, как все параметры сделки оговорены, подсистема

вычисляет детали сделки, например, примерную цену, посчитанную на основании колебания цен в прошлых годах, и выводит эту информацию в доступном виде (с помощью таблиц и графиков) на экран. Если в итоге клиент удовлетворён деталями сделки, подсистема регистрирует сделку в остальной системе.

Исходя из этого, можно выделить следующие процессы, происходящие в рассматриваемой подсистеме.



Рисунок 22 Процессы при использовании СППР

Таким образом, входами подсистемы являются:

- Параметры сделки (а именно, количество электроэнергии по месяцам, дата начала, дата окончания, необходимая зона поставки (в зависимости от страны и региона) и валюта сделки);
- Цена актива для заданных параметров сделки в данное время;
- Почасовой прогноз цен актива для заданного периода сделки.

Выходами подсистемы являются:

- Детали ещё не заключенной сделки в виде таблиц и графиков, которые могут быть использованы как для анализа эффективности, так и для закрепления деталей сделки;
- Зарегистрированная сделка в информационной системе.

Таблица 3 Входы подсистемы

ДАННЫЕ	ИСТОЧНИК	СПОСОБ ВВОДА	ТИП ДАННЫХ
Кол-во электроэнергии по месяцам	Графический интерфейс	Гистограмма	double[12]
Дата начала	Графический интерфейс	Календарь	DateTime
Дата окончания	Графический интерфейс	Календарь	DateTime
Зона поставки	Графический интерфейс	Текстовое поле	String
Валюта сделки	Графический интерфейс	Текстовое поле	String
Исторические цены на каждый месяц сделки	Биржевые данные	HTTP	double[]

Таблица 4 Выходы подсистемы

ДАННЫЕ	НАЗНАЧЕНИЕ	СПОСОБ ВЫВОДА	ТИП ДАННЫХ
Детали сделки	Графический интерфейс	График, таблица, текстовое поле	struct

Зарегистрированная сделка	Регистратор сделок	HTTP	Struct
------------------------------	-----------------------	------	--------

2.10. Выбор средств и технологий

В первую очередь выбор средств для реализации подсистемы зависит от средств, использованных при реализации остальных подсистем (а именно тех, с которыми рассматриваемая подсистема непосредственно контактирует).

Средства, использованные при реализации этих подсистем:

- Microsoft Visual C# – язык программирования основной логики процессов;
- Microsoft SQL Server 2012 – хранилище данных;
- .NET Framework – программная платформа;
- Windows Communication Foundation (WCF) – расширение .NET Framework, позволяющее без лишних трудозатрат создавать и использовать API для обмена данными между приложениями.

Таким образом, принимая во внимание стек технологий, используемых в информационной системе, с которой предстоит взаимодействовать, можно сделать вывод, что для разработки подсистемы необходимо использовать:

1. Microsoft Visual C#;
2. .NET Framework;
3. WCF для коммуникации с остальными подсистемами.

Пользовательский интерфейс подсистемы будет выполнен в виде веб-сайта и функционировать через веб-браузер. Для создания веб-сайта для платформы .NET Framework воспользуемся технологией

4. Microsoft ASP.NET и ASP.NET MVC Framework.

Для создания вёрстки и дизайна главного представления необходимо использовать

5. Гипертекстовый язык разметки HTML 5;

6. Каскадные таблицы стилей CSS 3.0.

Для того, чтобы сделать интерфейс интерактивным, также потребуется использовать технологии

7. Javascript;

8. JQuery для удобства манипулирования объектной моделью документа.

Также для реализации графического интерфейса пользователя потребуется использование некоторых библиотек как для языка Visual C#, так и для Javascript.

Для загрузки главного представления в виде HTML страницы используется обычный HTTP-Get запрос. Для того, чтобы обновление данных происходило незаметно для пользователя и без сброса и повторного заполнения формы, было

решено использовать технологию AJAX Get и Post для загрузки рассчитанных волатильностей цен с сервера и отправки транзакций на сервер.

С помощью концепции AJAX пользователь может работать с таблицами так, как если бы он работал в обычном приложении у себя на компьютере.

Преимущества использования технологии AJAX:

1. Экономия трафика. Использование AJAX позволяет значительно сократить трафик при работе с веб-приложением благодаря тому, что вместо загрузки всей страницы достаточно только получить/передать набор данных в формате JSON, а затем изменить содержимое страницы с помощью JavaScript.
2. Ускорение реакции интерфейса. Поскольку загрузка изменившейся части значительно быстрее, то пользователь видит результат своих действий быстрее и без мерцания страницы (возникающего при полной перезагрузке).

Самый простой способ выполнить AJAX-запрос – использовать функции \$.get и \$.post библиотеки jQuery. Именно этот способ было решено использовать.

1. Создание шаблона html-страницы, изменяющего свою компоновку на маленьких экранах планшетов и телефонов.

- а. Заявленным требованиям полностью удовлетворяет фреймворк bootstrap, принося легкое и эффективное масштабирование проекта с одной базой кода, от телефонов и планшетов до настольных компьютеров. Также bootstrap существенно ускоряет и упрощает разработку вёрстки html-страницы веб-приложения.

2. Отображение рассчитанной информации в виде таблицы с динамическим количеством строк, возможностью сортировки и форматирования данных.

a. Jqwidgets – в результате изучения было обнаружено, что эта библиотека имеет сложный api и чересчур перегружена для наших небольших задач.

b. Wijmo 5 – подходящая библиотека, очень простая в настройке, имеет хорошую производительность, за счёт чего и была выбрана как полностью устраивающая.

3. Отображение рассчитанной информации в виде интерактивного графика.

a. Jqwidgets – так же имеет возможности отрисовывать интерактивные графики. Минусы те же, что и в пункте про таблицы.

b. Wijmo 5 – снова нам подходит (лёгковесный, производительный, простой).

4. Задание количество электроэнергии, потребляемой покупателем в каждом месяце года.

a. Wijmo таблица для ввода информации и график для анализа введённой информации – решение действенное, но не совсем удобное, за счёт разных элементов для ввода и анализа введённой информации. Гораздо удобнее было бы иметь график с возможностью изменять высоту столбцов с помощью мыши.

b. Fusioncharts – библиотека, отображающая графики с возможностью изменять высоту столбцов с помощью мыши. Сложная в настройке, поэтому не годится для вывода рассчитанной информации в виде графика, но единственная, позволяющая динамически изменять высоту

столбцов графика. Именно за счёт этого было решено реализовать эту задачу с помощью этой библиотеки.

Таким образом, итоговый список технологий и библиотек, необходимых для разработки Веб-приложения, можно составить следующим образом:

Таблица 5 Список используемых технологий и библиотек

Разработчик	Наименование	Версия	Назначение
Microsoft	Visual C#	5.0	Язык программирования серверной части
Microsoft	.NET Framework	4.5	Платформа для C#
Microsoft	Windows Communication Foundation	4	Коммуникация с остальными подсистемами
Microsoft	ASP.NET MVC Framework	5.2.3	Фреймворк для создания веб-приложений
W3C	HTML	5	Язык разметки страниц
W3C	CSS	3	Таблицы стилей

	Javascript	1.8.1	Интерактивность UI
jQuery	jQuery	1.10.2	Манипуляция DOM
Component 1	Wijmo	5	Таблицы и графики
Twitter inc.	Bootstrap	3.0.0	HTML и CSS шаблоны

2.11. Проектирование интерфейса

На главном представлении информационной системы должны находиться следующие основные элементы:

1. Поля ввода данных сделки (6 шт).
2. График с 12-ю изменяемыми столбцами для ввода информации о ежемесячном потреблении электроэнергии покупателем.
3. Кнопки основных действий (применить, сохранить, очистить).
4. График, отображающий рассчитанную информацию о сделке.
5. Таблица, отображающая рассчитанную информацию о сделке.

Примерное расположение элементов графического интерфейса выглядит следующим образом:

Таблица 6 Расположение элементов графического интерфейса пользователя

График с 12-ю изменяемыми столбцами для ввода информации о ежемесячном потреблении электроэнергии покупателем.	Первые три поля ввода параметров сделки.	Вторые три поля ввода параметров сделки.
	Кнопки основных действий (применить, очистить, сохранить).	
График, отображающий рассчитанную информацию о сделке.		
Таблица, отображающая рассчитанную информацию о сделке.		

На графике должна отображаться следующая информация:

1. Посчитанная в зависимости от месячного объёма ожидаемая цена с учётом её волатильности (должна отображаться сама цена и её лучший и худший для покупателя случаи). Лучший способ отобразить эту информацию – диаграммы с областями, частично перекрывающимися друг друга, так как серия «лучшая цена» всегда меньше серии «цена», которая, в свою очередь, всегда меньше серии «худшая цена».
2. Посчитанная в зависимости от месячного объёма цена страйк без добавленной цены опциона и с ней. Лучший способ отобразить эту информацию – две линии разных цветов, разрезающие диаграммы из п.1.

Таблица должна содержать все данные, отображенные на графике, а также объем электричества по месяцам и цену опциона отдельно от общей цены.

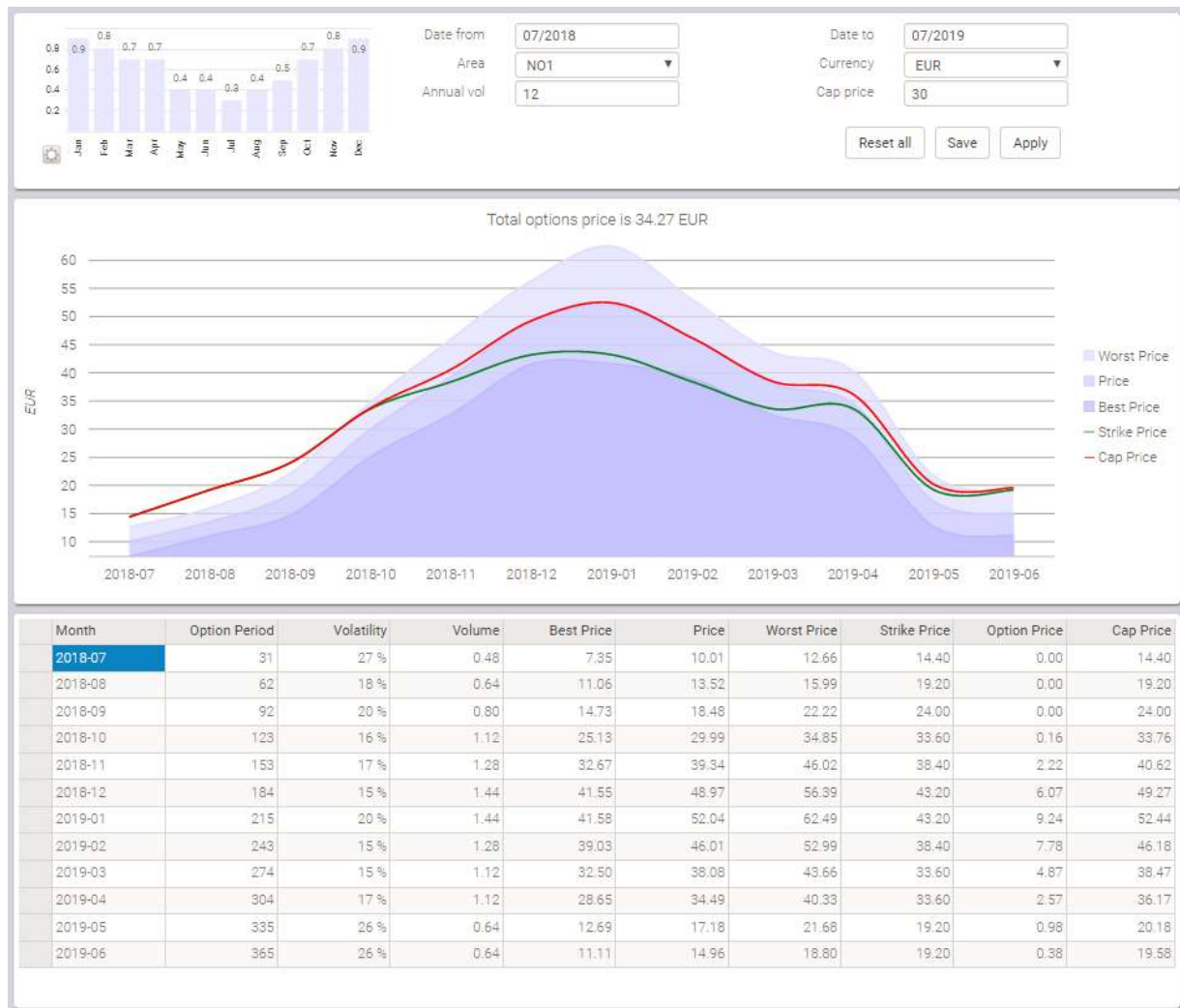


Рисунок 23 Интерфейс реализованной СППР

2.12. Работа с библиотеками

Для настройки javascript-библиотек используется текстовый формат JSON. Отображение графиков и таблиц библиотеками Wijmo и FusionCharts имеет

множество параметров, некоторые из которых необходимо использовать для правильного отображения интерфейса.

Для создания графика `wijmo` необходимо вызвать нужный конструктор, передав ему параметром идентификатор `html`-элемента, который будет контейнером для создаваемого графика.

```
_mainChart = new wijmo.chart.FlexChart('#main-chart');
```

После этого график нужно инициализировать параметрами, передаваемыми в формате `JSON`.

```
_mainChart.initialize({
    bindingX: 'Month',
    palette: ['#ddf', '#ccf', '#bbf', 'green', 'red'],
    series: [
        { name: "Worst Price", binding: "WorstPrice", chartType: SplineArea },
        ...
        { name: "Cap Price", binding: "CapPrice", chartType: Spline },
    ]
});
```

Параметр `bindingX` указывает, какие значения будут отложены по оси `X`. В нашем случае это месяца рассчитанного контракта. Параметр `palette` является не обязательным и определяет цвета отображения каждой серии данных. Параметр `series` определяет серии данных, отображаемых по оси `Y`, их название и тип их представления.

Настройка отображения таблицы для библиотеки `wijmo` выглядит похожим образом, за исключением параметров, передаваемых при инициализации.

```

_mainGrid = new wijmo.grid.FlexGrid("#main-grid");
_mainGrid.initialize({
    autoGenerateColumns: false,
    isReadOnly: true,
    columns: [
        { header: "Month", binding: "Month", width: "*" },
        { header: "Option Period", binding: "DaysDiff", width: "*" },
        .....
        { header: "Cap Price", binding: "CapPrice", width: "*", format: "n2" },
    ],
});

```

Параметр `autoGenerateColumns` необходим для возможности указания колонок вручную. Параметр `isReadOnly` говорит о том, что строки таблицы невозможно редактировать непосредственно (редактирование возможно только путём изменения входных данных для расчёта). Параметр `columns` содержит описания всех колонок, отображаемых в таблице (их ширины, заголовки и форматы).

3. ЗАКЛЮЧЕНИЕ

В рамках данной работы были исследованы такие понятия как волатильность, опцион, методы и модели его ценообразования. Результаты исследования показали, что существующие методы не в полной мере соответствуют реальному положению дел на рынке электроэнергии, хоть и, безусловно, обладают очень мощным инструментарием.

В результате был предложен новый подход к выбору распределения отклонений цен при их генерации для метода Монте-Карло, который позволил немного улучшить показатели, минимизировав цену опциона для высоковолатильных месяцев. Также в рамках данной работы на основе этого подхода была реализована система поддержки принятия решений.

В качестве дальнейших доработок можно рассмотреть использование более сложных, ассиметричных распределений, не только в целях уменьшения цены опциона, но и для минимизации страховых выплат в месяцы, когда цена опциона более склонна расти, чем падать.

4. СПИСОК ЛИТЕРАТУРЫ

Книги

1. Johnson N.L., Kotz S., Balakrishnan N. Continuous Univariate Distributions / N.L. Johnson, S. Kotz, N. Balakrishnan. – Vol.1. – 1994. – 754 p. - ISBN: 978-0-471-58495-7.
2. Лукасевич И.Я. Анализ финансовых операций. Методы, модели, техника вычислений / И.Я. Лукасевич // М.: Финансы, ЮНИТИ. - 1998. - 400 с.
3. Шеннон Р. Имитационное моделирование – искусство и наука. / Р. Шеннон // М.: Мир. -1978. - 418 с.
4. Wilkens S. Option Pricing Based on Mixtures of Distributions: Evidence from the Eurex Index and Interest Rate Futures Options Market / S. Wilkens // 2016. – 390p.
5. Alexander C. O. A practitioners guide to financial data analysis // New York: Wiley. – 2001. – 514p. - ISBN: 978-0-471-89975-4.

Статьи

6. Ворошилова Н.А. Сравнительный анализ методов моделирования стоимости опционов / Н.А. Ворошилова // Научный журнал КубГАУ, №26(2). – 2007.
7. Black F., Scholes M. The pricing of options and corporate liabilities // Journal of Political Economy Vol.81. – 1973. - pp.637-659.
8. Глухов М. Оценка опционов методом Монте Карло / М. Глухов // Futures & Options. – 2009. – с.38-43
9. Klebanov L.V. No Stable Distributions in Finance, please! / Klebanov L.V. // Cornell University Library. – 2016.

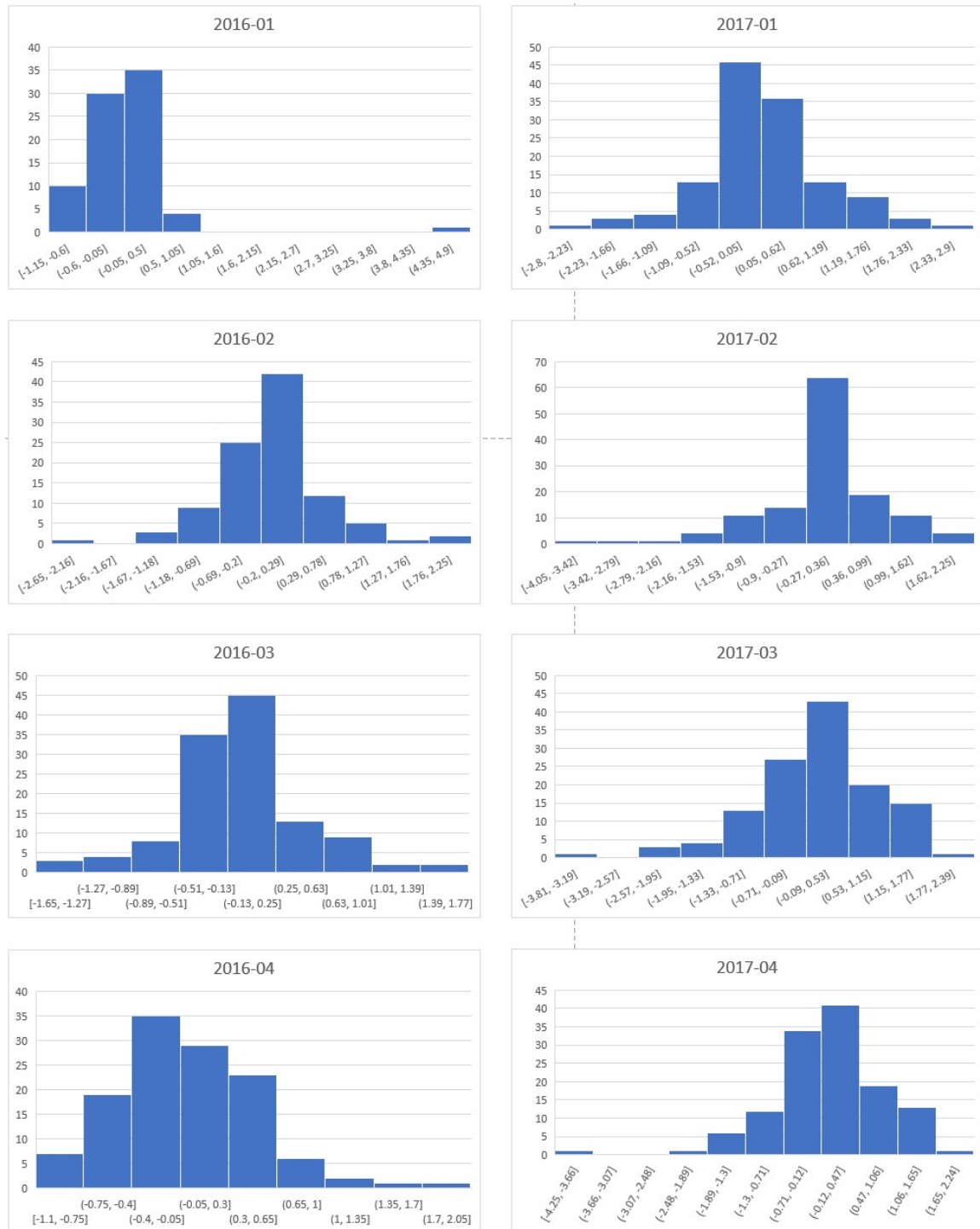
10. Klebanov L.V., Volchenkova I. Heavy Tailed Distributions in Finance: Reality or Myth? Amateurs Viewpoint. / L.V. Klebanov, I. Volchenkova // Cornell University Library. – 2015.
11. Балаев А.И. Многомерное скошенное t-распределение с вектором степеней свободы и его применение в моделях финансовых рынков / А.И. Балаев // Прикладная эконометрика. – 2011.
12. Nadarajah S. Making the Cauchy work / S. Nadarajah // Brazilian Journal of Probability and Statistics. – 2011. – pp.99-120.
13. Jones M. C. Marginal replacement in multivariate densities, with application to skewing spherically symmetric distributions // Journal of Multivariate Analysis. – Vol.81. – 2002. - pp. 85–99.
14. Black F. The pricing of commodity contracts // Journal of Financial Economics. – Vol.3 – 1976. – pp.167-179.
15. Carmona, R. and Durrleman V. Pricing and Hedging spread options in a log-normal model / Department of Operations Research and Financial Engineering, Princeton University, Princeton, NJ. - 2003.
16. Cox, J. and Ross S. The valuation of options for alternative stochastic processes / Journal of Financial Economics. – Vol.3. – 1976. – pp.145-166.
17. Harrison, J.M. and Kreps D. Martingales and arbitrage in multiperiod security markets / Journal of Economic Theory. – 1979.
18. Harrison, J.M. and Pliska S. Martingales and stochastic integrals in the theory of continuous trading / Stochastic Processes and Their Applications. – Vol.11. – 1981.

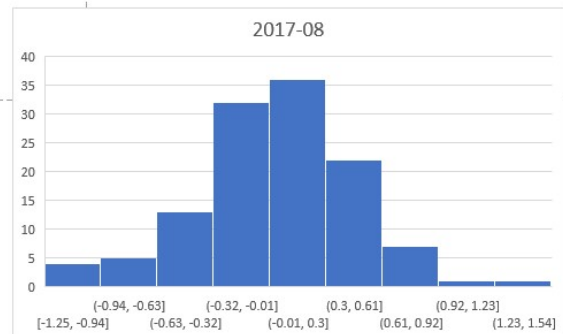
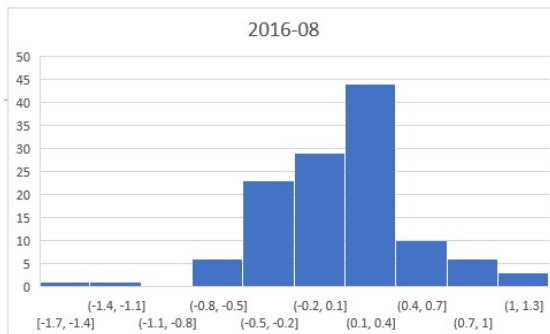
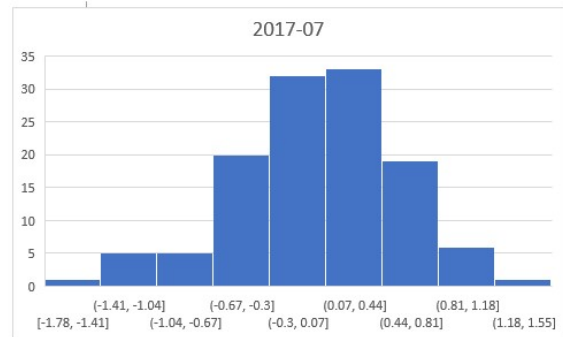
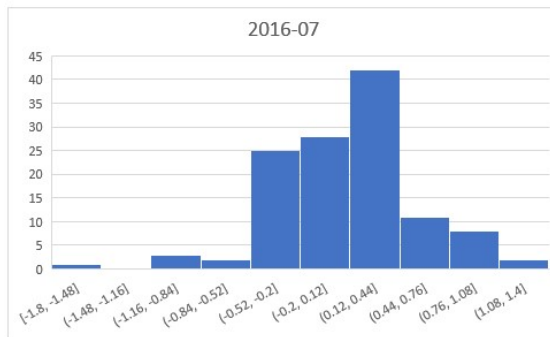
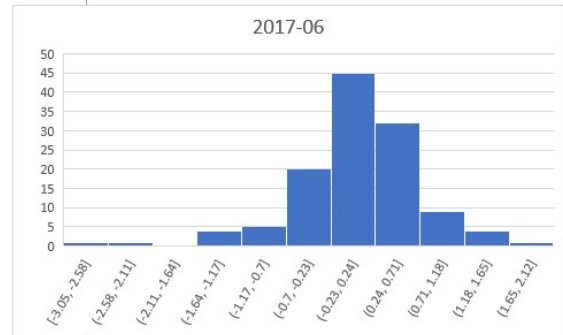
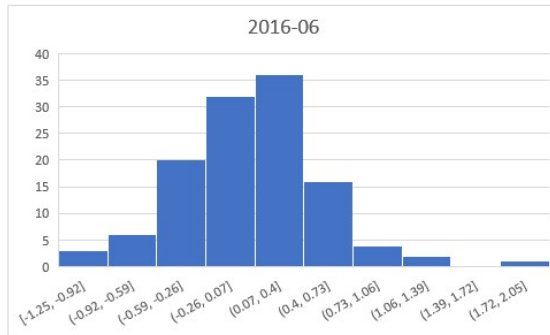
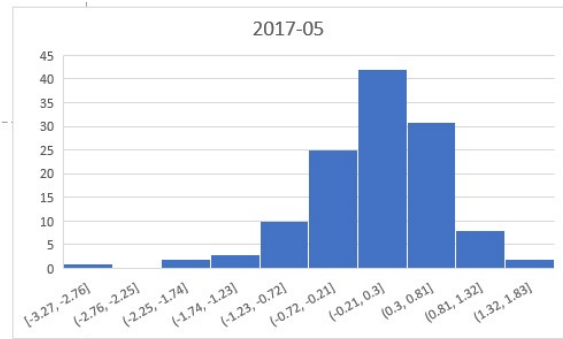
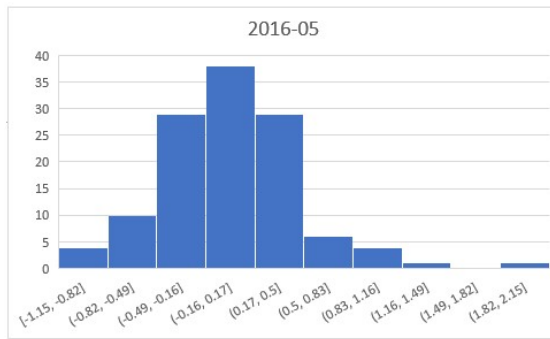
Электронные ресурсы

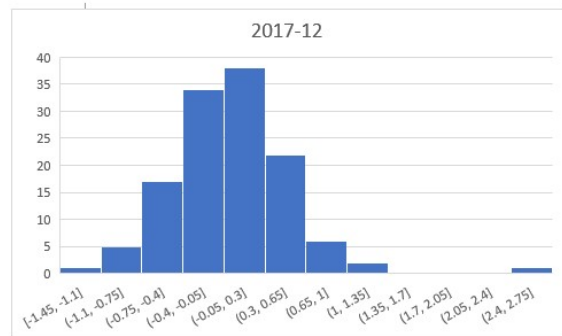
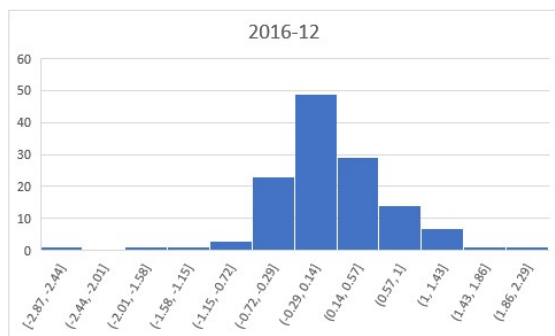
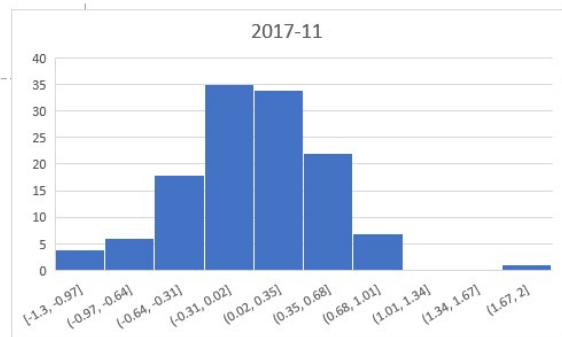
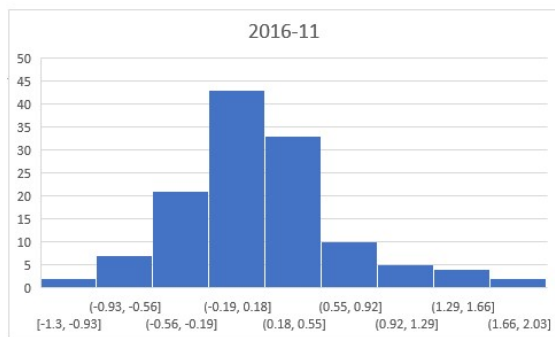
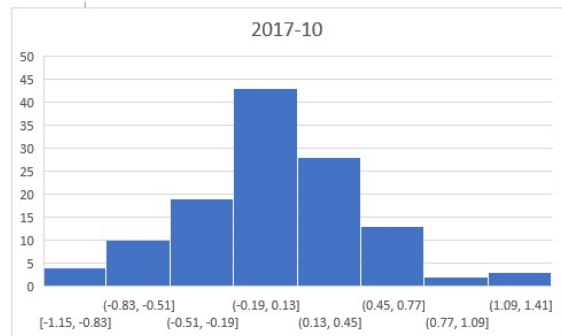
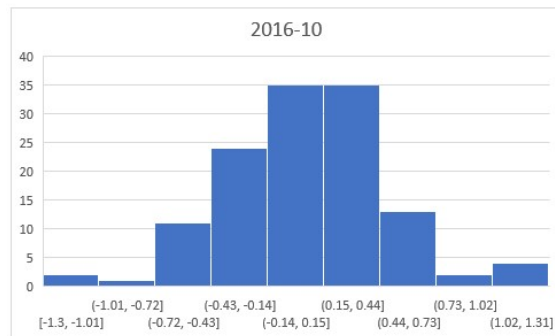
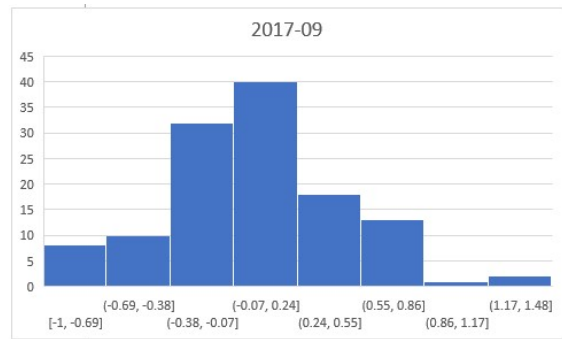
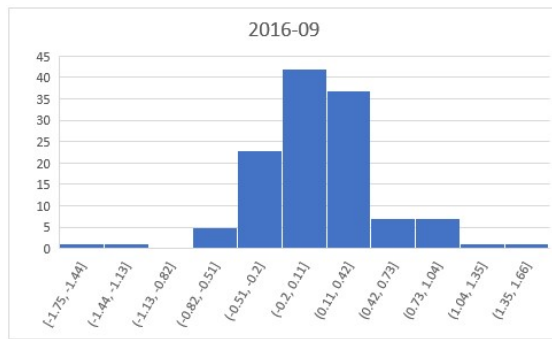
19. Asher M. How fat are your tails? [Электронный ресурс] / Ahser M. // R-bloggers. – 2012. – Режим доступа: <https://www.r-bloggers.com/how-fat-are-your-tails/>.
20. Распределение вероятностей [Электронный ресурс] // DocPlayer. – Режим доступа: <http://docplayer.ru/60911418-Raspredelenie-veroyatnostey.html>

Приложение А. Графики распределения изменения цен по месяцам.

Сравнение 2016 и 2017 годов.







Приложение Б. Исходные коды алгоритмов и имитационного моделирования.

```

class Program
{
    static Dictionary<DateTime, List<XlmNasdaqPriceParser.Price>> FuturePricesByStartDate;

    static void Main(string[] args)
    {
        FuturePricesByStartDate = new Dictionary<DateTime,
List<XlmNasdaqPriceParser.Price>>();

        for (int i = 1; i <= 12; i++)
        {
            var month = new DateTime(2000, i, 1);
            var m2016 = month.AddYears(16);
            var m2017 = month.AddYears(17);

            FuturePricesByStartDate.Add(m2016,
XlmNasdaqPriceParser.Parse($"futureprices/monthly/NOPEOAFUTBLM{month:MMM}-16.xml", m2016));
            FuturePricesByStartDate.Add(m2017,
XlmNasdaqPriceParser.Parse($"futureprices/monthly/NOPEOAFUTBLM{month:MMM}-17.xml", m2017));
        }

        PortfolioSimulator.SimulatePortfolio(FuturePricesByStartDate);

        Console.Read();
    }
}

class XlmNasdaqPriceParser
{
    public class Price
    {
        public DateTime Date;
        public double ClosePrice;

        public override string ToString()
        {
            return $"{Date:yyyy-MM-dd} :: {ClosePrice:0.00}";
        }
    }

    public static List<Price> Parse(string fileName, DateTime expirationDate)
    {
        XmlDocument doc = new XmlDocument();
        doc.Load(fileName);

        var result = new List<Price>();

        foreach (XmlNode node in doc.DocumentElement.ChildNodes)
        {
            var dateString = node.Attributes["dt"]?.Value;
            var priceString = node.Attributes["sp"]?.Value;

            if (String.IsNullOrEmpty(dateString) || String.IsNullOrEmpty(priceString))
            {
                continue;
            }
        }
    }
}

```

```

    }

    result.Add(new Price
    {
        Date = DateTime.Parse(dateString),
        ClosePrice = Double.Parse(priceString)
    });
}

result = result.Where(p => p.Date <= expirationDate).OrderBy(p => p.Date).ToList();

return result;
}
}

```

```

class VolatilityCalculator
{
    public static double CalcHistoricalVolatility(List<XlmNasdaqPriceParser.Price> prices)
    {
        var returns = new List<double>();

        for (var i = 1; i < prices.Count; i++)
        {
            returns.Add((prices[i].ClosePrice - prices[i - 1].ClosePrice) / prices[i - 1].ClosePrice);
        }

        return CalcHistoricalFolatilityByReturns(returns);
    }

    private static double CalcHistoricalFolatilityByReturns(List<double> deltas)
    {
        var avg = deltas.Average();
        var sum = 0.0;

        foreach (var delta in deltas)
        {
            var d2 = delta - avg;
            sum += d2 * d2;
        }

        sum /= deltas.Count - 1;

        return Math.Sqrt(sum) * Math.Sqrt(251.0);
    }

    public static double CalcTailness(List<XlmNasdaqPriceParser.Price> prices)
    {
        var diffs = new List<double>();

        for (int i = 1; i < prices.Count; i++)
        {
            diffs.Add(prices[i].ClosePrice - prices[i - 1].ClosePrice);
        }

        return AveDev(diffs) / StDev(diffs);
    }

    private static double StDev(List<double> prices)
    {
        var avg = prices.Average(p => p);
        var sum = prices.Sum(p => Math.Pow(p - avg, 2));
    }
}

```

```

        return Math.Sqrt(sum / (prices.Count - 1));
    }

    private static double AveDev(List<double> prices)
    {
        var avg = prices.Average(p => p);
        return prices.Average(p => Math.Abs(p - avg));
    }
}

class PortfolioSimulator
{
    private class OptionPosition
    {
        public DateTime ExpirationDate;
        public DateTime ContractDate;
        public double MonthlyConsumption;
        public double Strike;

        public double SpotPrice;
        public double Volatility;
        public double DaysToExpiry;
        public double OptionPrice;
        public double PnL;
        public double RealPrice;
        public double RealVolatility;
        public bool Executed;
        public double Tailness;

        public void CalcOptionPrice(Dictionary<DateTime, List<XlmNasdaqPriceParser.Price>>
futurePricesByStartDate, IOptionCalculator calculator)
        {
            var prices = futurePricesByStartDate[ExpirationDate];
            var pricesAvailable = prices.Where(p => p.Date <= ContractDate).ToList();
            var volatility = VolatilityCalculator.CalcHistoricalVolatility(pricesAvailable);
            var tailness = VolatilityCalculator.CalcTailness(pricesAvailable);
            var currentPrice = pricesAvailable.Last().ClosePrice;

            var daysToExpiry = prices.Count - pricesAvailable.Count;
            var yearsToExpiry = daysToExpiry / 251.0;

            OptionPrice = calculator.CalcPremium(currentPrice, Strike, yearsToExpiry,
volatility, 0.06, tailness);

            SpotPrice = currentPrice;
            Volatility = volatility;
            DaysToExpiry = daysToExpiry;
            Tailness = tailness;
        }

        public void CalcProfitOrLoss(Dictionary<DateTime, List<XlmNasdaqPriceParser.Price>>
futurePricesByStartDate)
        {
            var realPrice = futurePricesByStartDate[ExpirationDate].Last().ClosePrice;

            if (realPrice >= Strike)
            {
                PnL = Strike - realPrice + OptionPrice;
                Executed = true;
            }
            else
            {

```



```

        PnL = OptionPrice;
        Executed = false;
    }

    PnL *= MonthlyConsumption;

    RealPrice = realPrice;
    RealVolatility =
VolatilityCalculator.CalcHistoricalVolatility(futurePricesByStartDate[ExpirationDate]);
    }
}

public static void SimulatePortfolio(Dictionary<DateTime,
List<XlmNasdaqPriceParser.Price>> futurePricesByStartDate)
{
    Console.WriteLine("\n--- Simulate Portfolio ---");
    SimulatePortfolio(futurePricesByStartDate,
GetExamplePortfolio(futurePricesByStartDate));
}

private static void SimulatePortfolio(Dictionary<DateTime,
List<XlmNasdaqPriceParser.Price>> futurePricesByStartDate, List<OptionPosition> portfolio)
{
    SimulatePortfolio(futurePricesByStartDate, portfolio, BlackSholes.Instance);
    SimulatePortfolio(futurePricesByStartDate, portfolio, Combined.Instance);
    SimulatePortfolio(futurePricesByStartDate, portfolio, MonteCarlo.Instance);
    SimulatePortfolio(futurePricesByStartDate, portfolio, Binominal.Instance);
}

private static void SimulatePortfolio(Dictionary<DateTime,
List<XlmNasdaqPriceParser.Price>> futurePricesByStartDate, List<OptionPosition> portfolio,
IOptionCalculator calculator)
{
    int totalNumber = portfolio.Count;
    int executedNumber = 0;
    int totalInProfit = 0;
    double totalProfit = 0, totalLoss = 0;

    foreach (var contract in portfolio)
    {
        contract.CalcOptionPrice(futurePricesByStartDate, calculator);
        contract.CalcProfitOrLoss(futurePricesByStartDate);

        if (contract.Executed) executedNumber++;

        if (contract.PnL >= 0) { totalProfit += contract.PnL; totalInProfit++; }
        else totalLoss += contract.PnL;
    }

    StringBuilder sb = new StringBuilder();

    sb.AppendLine(
        "Date," +
        "Target," +
        "Days," +
        "Volat," +
        "Real V," +
        "Consum," +
        "Profit," +
        "Loss," +
        "Spot," +
        "Strike," +
        "Real," +
        "Premium," +

```

```

        "Tailness," +
        "Exec");

foreach (var contract in portfolio)
{
    sb.AppendLine(
        $"{contract.ContractDate:yyyy-MM-dd}," +
        $"{contract.ExpirationDate:MM}," +
        $"{contract.DaysToExpiry}," +
        $"{contract.Volatility}," +
        $"{contract.RealVolatility}," +
        $"{contract.MonthlyConsumption}," +
        $"{(contract.PnL > 0 ? contract.PnL : 0)}," +
        $"{(contract.PnL < 0 ? contract.PnL : 0)}," +
        $"{contract.SpotPrice}," +
        $"{contract.Strike}," +
        $"{contract.RealPrice}," +
        $"{contract.OptionPrice}," +
        $"{contract.Tailness}," +
        $"{(contract.Executed ? 1 : 0)}");
}

File.WriteAllText("out/" + calculator.GetType().Name + ".csv", sb.ToString());

var avgPrice = portfolio.Sum(p => p.OptionPrice) / portfolio.Count;

Console.WriteLine($"{calculator.GetType().Name}:\\t " +
    $"Exec: {executedNumber}/{totalNumber}, " +
    $"InProfit: {totalInProfit}/{totalNumber}, " +
    $"P: {totalProfit:0.00}, " +
    $"L: {totalLoss:0.00}, " +
    $"PnL: {(totalProfit + totalLoss):0.00}, " +
    $"AvgPrice: {avgPrice:0.0000}");
}

private static List<OptionPosition> GetExamplePortfolio(Dictionary<DateTime,
List<XlmNasdaqPriceParser.Price>> futurePricesByStartDate)
{
    var random = new Random(100000);
    random.Next();
    random.Next();

    var result = new List<OptionPosition>();

    for (int k = 0; k < 300; k++)
    {
        var monthlyConsumption = random.Next(50, 250);

        for (int i = 1; i <= 12; i++)
        {
            var expirationDate = new DateTime(2017, i, 1);

            var prices = futurePricesByStartDate[expirationDate];

            var contractDate = expirationDate.AddDays(-random.Next(20, 90));
            var currentPrice = prices.Last(p => p.Date <= contractDate).ClosePrice;

            result.Add(new OptionPosition
            {
                ExpirationDate = expirationDate,
                ContractDate = contractDate,
                MonthlyConsumption = monthlyConsumption,
                Strike = currentPrice + currentPrice * (random.Next(5, 30) / 100.0)
            });
        }
    }
}

```

```

    }
}

return result;
}
}

interface IOptionCalculator
{
    double CalcPremium(double price, double strike, double yearsToExpiry, double volatility,
double riskFreeRate, double tailness = 0);
}

class BlackSholes : IOptionCalculator
{
    public static BlackSholes Instance = new BlackSholes();

    /// <summary>
    /// https://support.office.com/en-us/article/normsdist-function-463369ea-0345-445d-802a-4ff0d6ce7cac
    /// </summary>
    private static double NormSDist(double x)
    {
        const double b1 = 0.319381530;
        const double b2 = -0.356563782;
        const double b3 = 1.781477937;
        const double b4 = -1.821255978;
        const double b5 = 1.330274429;
        const double p = 0.2316419;
        const double c = 0.39894228;

        if (x >= 0.0)
        {
            double t = 1.0 / (1.0 + p * x);
            return (1.0 - c * Math.Exp(-x * x / 2.0) * t * (t * (t * (t * (t * b5 + b4) + b3)
+ b2) + b1));
        }
        else
        {
            double t = 1.0 / (1.0 - p * x);
            return (c * Math.Exp(-x * x / 2.0) * t * (t * (t * (t * (t * b5 + b4) + b3) + b2)
+ b1));
        }
    }

    public double CalcPremium(double price, double strike, double yearsToExpiry, double
volatility, double riskFreeRate, double tailness = 0)
    {
        var d1 = (Math.Log(price / strike) + (riskFreeRate + 0.5 * volatility * volatility) *
yearsToExpiry) / (volatility * Math.Sqrt(yearsToExpiry));
        var d2 = d1 - volatility * Math.Sqrt(yearsToExpiry);
        var nd1 = NormSDist(d1);
        var nd2 = NormSDist(d2);

        return price * nd1 - strike * Math.Exp(-riskFreeRate * yearsToExpiry) * nd2;
    }
}

class Binominal : IOptionCalculator
{

```

```

public static Binominal Instance = new Binominal();

public double CalcPremium(double price, double strike, double yearsToExpiry, double
volatility, double riskFreeRate, double tailness = 0)
{
    var steps = (int)(yearsToExpiry * 365);

    double totalValue = 0.0;
    double optionUp = OptionUp(yearsToExpiry, volatility, steps);
    double optionDown = OptionDown(yearsToExpiry, volatility, steps);
    double probability = Probability(yearsToExpiry, volatility, steps, riskFreeRate,
optionDown, optionUp);

    for (int step = 0; step <= steps; step++)
    {
        var finalPrice = price * Math.Pow(optionUp, step) * Math.Pow(optionDown, steps -
step);
        var payoff = Math.Max(finalPrice - strike, 0);
        var nodeValue = BinomialNodeValue(step, steps, probability);
        totalValue += nodeValue * payoff;
    }

    return totalValue / Math.Exp(riskFreeRate * yearsToExpiry);
}

private static double BinomialCoefficient(int step, int steps)
{
    return Factorial(steps) / (Factorial(step) * Factorial(steps - step));
}

private static double BinomialNodeValue(int step, int steps, double probability)
{
    return BinomialCoefficient(step, steps) *
        Math.Pow(probability, step) *
        Math.Pow(1.0 - probability, steps - step);
}

private static double OptionUp(double yearsToExpiry, double volatility, int steps)
{
    return Math.Exp(volatility * Math.Sqrt(yearsToExpiry / steps));
}

private static double OptionDown(double yearsToExpiry, double volatility, int steps)
{
    return Math.Exp(-volatility * Math.Sqrt(yearsToExpiry / steps));
}

private static double Probability(double yearsToExpiry, double volatility, int steps,
double riskFreeRate, double optionDown, double optionUp)
{
    double discountRate = Math.Pow(1.0 + riskFreeRate, yearsToExpiry / steps);
    return (discountRate - optionDown) / (optionUp - optionDown);
}

private static double Factorial(int n)
{
    double d = 1.0;
    for (int j = 1; j <= n; j++)
    {
        d *= j;
    }
    return d;
}
}

```

```

class MonteCarlo : IOptionCalculator
{
    private abstract class DistributedRandom : Random
    {
        public abstract double NextSample();
    }

    private class NormalRandom : DistributedRandom
    {
        public override double NextSample()
        {
            return Normal.Sample(this, 0, 1);
        }
    }

    private class CauchyRandom : DistributedRandom
    {
        private double Scale;

        public CauchyRandom(double coef)
        {
            Scale = coef;
        }

        public override double NextSample()
        {
            return Cauchy.Sample(this, 0, Scale);
        }
    }

    public static MonteCarlo Instance = new MonteCarlo();

    private static double TruncatedCauchyLimit = Math.PI;
    private static double MaxTailnessForCauchy = Math.Sqrt(2 / Math.PI);
    private static double MinScaleForCauchy = 0.2;

    private static double GenerateRandomValue(DistributedRandom random)
    {
        double randomValue;

        do
        {
            randomValue = random.NextSample();
        }
        while (Math.Abs(randomValue) > TruncatedCauchyLimit && random is CauchyRandom);

        return randomValue;
    }

    private static double GenerateRandomReturn(DistributedRandom random, double
expectedReturn, double volatility, double peroid)
    {
        double randomValue = GenerateRandomValue(random);
        return (expectedReturn - 0.5 * volatility * volatility) * peroid + volatility *
Math.Sqrt(peroid) * randomValue;
    }

    private static double GenerateFinalPrice(DistributedRandom random, double price, double
yearsToExpiry, double volatility, double riskFreeRate)
    {
        var ror = GenerateRandomReturn(random, riskFreeRate, volatility, yearsToExpiry);
        return price * Math.Exp(ror);
    }
}

```

```

    private static double GeneratePayoff(DistributedRandom random, double price, double
strike, double yearsToExpiry, double volatility, double riskFreeRate)
    {
        var finalPrice = GenerateFinalPrice(random, price, yearsToExpiry, volatility,
riskFreeRate);
        var payoff = finalPrice - strike;
        return Math.Max(payoff, 0);
    }

    // https://stats.stackexchange.com/questions/343093/
    private static double TruncatedCauchyTailness(double s, double A)
    {
        double atan = Math.Atan(A / s);
        return Math.Log((s * s + A * A) / (s * s)) * Math.Sqrt((A * s - s * s * atan) / atan)
/ (-2 * s * atan + 2 * A);
    }

    private static List<Tuple<double, double>> ScaleByTailness = new List<Tuple<double,
double>>();

    static MonteCarlo()
    {
        for (double s = MinScaleForCauchy; ; s += 0.0001)
        {
            var tailness = TruncatedCauchyTailness(s, TruncatedCauchyLimit);
            ScaleByTailness.Add(new Tuple<double, double>(tailness, s));
            if (tailness > MaxTailnessForCauchy) break;
        }
    }

    public double CalcPremium(double price, double strike, double yearsToExpiry, double
volatility, double riskFreeRate, double tailness = 0)
    {
        DistributedRandom random;

        if (tailness <= MaxTailnessForCauchy)
        {
            var scale = ScaleByTailness.First(s => s.Item1 >= tailness).Item2;
            random = new CauchyRandom(scale);
        }
        else
        {
            random = new NormalRandom();
        }

        int iterations = 10000;
        double payoff = 0;

        for (int i = 0; i < iterations; i++)
        {
            payoff += GeneratePayoff(random, price, strike, yearsToExpiry, volatility,
riskFreeRate);
        }

        var avgPayoff = payoff / iterations;

        var discountedPayoff = avgPayoff / Math.Pow(1 + riskFreeRate, yearsToExpiry);

        return discountedPayoff;
    }
}

```