

Міністерство освіти і науки України
Національний університет «Кієво-Могилянська академія»
Факультет інформатики
Кафедра математики

Кваліфікаційна робота
освітній ступінь – бакалавр

на тему: «Дослідження якості моделі з ринковим часом»

Виконала
студентка 4-го року навчання,
освітньої програми «Прикладна математика», 113
Дубницька Марія Вікторівна

Керівник
Щестюк Наталія Юріївна
Доцент, кандидат ф.-м. наук
Рецензент

Кваліфікаційна робота захищена з оцінкою

Секретар ЕК

«_____» _____ 20____ р.

Міністерство освіти і науки України
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «КИЄВО-МОГИЛЯНСЬКА

АКАДЕМІЯ»

Факультет інформатики
Кафедра математики

ЗАТВЕРДЖУЮ
Зав. кафедри математики,
Доцент, кандидат ф.-м. наук
Р.К. Чорней

“ _____ ” _____ 2023

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ
на кваліфікаційну роботу студентці 4-го курсу факультету інформатики Дубницькій
Марії Вікторівні

Тема: Дослідження якості моделі з ринковим часом.

Зміст ТЧ до кваліфікаційної роботи:

- 1 Анотація
- 2 Вступ
- 3 Модель з активним ринковим часом
- 4 Аналіз якості моделі на основі реальних фінансових даних
- 5 Висновки
- Література
- 6 Додатки

Дата видачі “ _____ ” _____ 2023 р.

Керівник _____

Завдання отримано _____

Календарний план виконання кваліфікаційної роботи

Тема: Дослідження якості моделі з ринковим часом.

Календарний план виконання роботи:

№ п/п	Назва етапу дипломного проекту (роботи)	Термін виконання етапу	Примітка
1.	Отримання завдання на кваліфікаційну роботу	03.11.2022	
2.	Огляд технічної літератури за темою роботи	11.2022-12.2022	
3.	Аналіз предметної області	11.2022-12.2022	
4.	Написання програмного коду	02.02-02.05	
5.	Збір даних	09.02-17.02	
6.	Написання тексту	20.03-09.05	
7.	Створення таблиць, аналіз результатів	06.04-09.05	
8.	Попередній захист роботи	11.05.2023	
9.	Проходження перевірки на плагіат	28.05.2023	
10.	Захист роботи	05.06.2023-06.06.2023	

Студент Дубницька М.В. _____

Керівник Щестюк Н.Ю. _____

“ _____ ” _____ 2023 р.

ЗМІСТ

Анотація	5
ВСТУП.....	6
РОЗДІЛ 1. Модель з активним ринковим часом	8
1.1 Основні поняття та означення.....	8
1.2 Модель Блека-Шоулза як класична модель для ціноутворення	9
1.3 Недоліки моделі Блека-Шоулза.....	11
1.4 Модель з активним ринковим часом.....	13
1.5 Визначення параметрів моделі.....	15
1.6 Ціноутворення в моделі з ринковим часом.....	16
1.7 Статистичні похибки та регресійний аналіз.....	17
РОЗДІЛ 2. Аналіз якості моделі на основі реальних фінансових даних	19
2.1 Ринкові дані	19
2.2 Бібліотеки та змінні	20
2.3 Знаходження параметрів моделі.....	21
2.4 Розрахунок преміумів для опціонів	22
2.5 Розрахунок статистичних похибок, їх характеристики	23
2.6 Обрахунок грошовості	25
2.7 Регресійний аналіз	25
ВИСНОВКИ	28
Список використаної літератури	30
Додаток А. Програмний код	31
Додаток В. Зібрані дані щодо кол опціонів	45
Додаток С. Історичні дані.....	48

Анотація

Кваліфікаційна робота присвячена дослідженню якості моделі з ринковим часом з точки зору мінімізації похибок, які залежать від грошовості (moneyness) та часу до закінчення дії опціону (time to maturity). Для дослідження залежності похибок запропоновані рівняння регресії. Дослідження проведене на основі даних компанії Netflix. Для цього використовувались англомовні джерела та офіційні дані про ціни на акції та преміуми опціонів. Програмування моделі, аналіз її статистичних похибок та візуалізація отриманих даних відбувались за допомогою мови програмування Python. Її використовують для статистичних обчислень, аналізу та представлення даних у графічному вигляді. Також використовувалось відкрите інтегроване середовище розробки (IDE) для Python – Jupyter Notebook.

Ключові слова: математична модель, ринковий час, якість моделі, статистичні похибки RPE та ARPE, регресійний аналіз.

ВСТУП

Математичне моделювання фінансових ринків почалося в 1900-х роках, коли французький математик Луї Башельє розробив модель Броунівського руху для опису цінової динаміки акцій. Однак, більш широке застосування математичних методів в фінансах почалося в 1950-х і 1960-х роках, коли низка дослідників, включаючи Гаррі Марковіца, Уільяма Шарпа та Мертона Міллера, розробили моделі портфеля та теорію ціноутворення на фінансових ринках.

Пізніше, в 1973 році, Фішер Блек та Майрон Шоулз представили свою модель оцінки опціонів, тепер відому як модель Блека-Шоулза. Вона стала першим прикладом успішного використання математичного моделювання в фінансах.

З того часу математичні моделі відіграють вирішальну роль у сфері фінансової математики, забезпечуючи систематичний і кількісний підхід до розуміння, прогнозування та управління фінансовими ринками. Ці моделі використовуються для аналізу та інтерпретації поведінки фінансових ринків, що дозволяє інвесторам, фінансовим установам і політикам приймати обґрунтовані рішення.

Математичні моделі мають дуже широке застосування в фінансах. Прикладами можуть бути управління ризиками, оптимізація портфеля інвестора, фінансовий інжиніринг. Однак, одним із найбільш значущих застосувань математичних моделей у фінансах є ціноутворення похідних цінних паперів. Модель Блека-Шоулза, наприклад, є математичною моделлю, яка використовується для розрахунку теоретичної ціни опціонів. Крім моделі Блека-Шоулза існує також багато альтернативних моделей, які мають свої переваги та недоліки. Важливо також зазначити, що хоча математичні моделі є потужними інструментами, вони не є безпомилковими.

Метою даної роботи є дослідження якості моделі з ринковим часом за допомогою статистичних похибок RPE та ARPE, які залежать від грошовості (moneyness) та часу до закінчення дії опціону (time to maturity), з точки зору їх мінімізації. Для дослідження залежності похибок запропонований регресійний аналіз. Також запропонована програмна реалізація оцінки похибок та регресії на основі реальних даних за допомогою мови програмування Python.

Для досягнення поставленої мети необхідно виконати такі задачі:

1. Визначити особливості моделі з ринковим часом, її основні відмінності від класичної моделі Блека-Шоулза
2. Запрограмувати справедливу ціну кол опціону для моделі з ринковим часом
3. Розглянути застосування статистичних похибок RPE та ARPE, їх статистичні характеристики
4. Зробити регресійний аналіз статистичних похибок RPE та ARPE

Практичне дослідження в даній роботі проведене на основі даних про ціну акцій та кол опціонів за період з 9 лютого 2023 по 17 лютого 2023 компанії Netflix. Як середовище для розробки використовується Jupyter Notebook.

Дана робота складається зі вступу, двох розділів, висновків та додатків.

Перший розділ присвячений огляду предметної області, теоретичним поняттям та роз'ясненням.

В другому розділі описане практичне дослідження, наведені отримані результати.

Ця робота може бути корисною як для теоретичних досліджень, пов'язаних зі статистичним аналізом, так і для практичного застосування, для оцінки якості математичних моделей фінансових ринків, визначення чутливостей, головних тенденцій моделей.

РОЗДІЛ 1. Модель з активним ринковим часом

В цьому розділі спершу введемо необхідні поняття та розглянемо класичну модель Блека-Шоулза, її недоліки. Далі подивимося на модель з ринковим часом, її основні особливості, знаходження параметрів для неї. Розглянемо також інструменти для дослідження якості моделі – статистичні похибки, регресійний аналіз.

1.1 Основні поняття та означення

Введемо кілька означень:

- **Дериватив** - це фінансовий інструмент, контракт, який фіксує ціну на інший актив, відповідно вартість його залежить від ціни цього іншого активу, який називається базовим активом. Базовий актив може бути, акцією, валютою, сировиною або облігацією [1].
- **Кол- і пут-опціони** – це деривативи (договори), які надають право на купівлю або продаж активу за певною ціною до певної дати в майбутньому. Втім, цей фінансовий інструмент не зобов'язує купити або продати базовий актив за визначеною ціною в майбутньому.
- **Страйкова ціна** – це сума, за яку можна купити базовий актив (кол опціон) або продати (пут опціон) [2].
- **Преміум** - ціна, яку платить покупець за опціон (спочатку) [2].

Деривативи використовують для захисту від ризиків, пов'язаних з коливаннями цін на базові активи, або для спекуляції на зміну цін на базові активи. У минулому оператори визначали ці інструменти як інструменти з

«опціональністю», де одна сторона мала право вибору, а інша мала зобов'язання [3]. Опціони відрізняються від інших цінних паперів тим, що є потенційним активом для однієї сторони та потенційною відповідальністю для іншої. Наприклад, для валютної пари пут на гривня/долар (право продавати гривні та купувати долари) – це кол долар/гривня [3].

На ціноутворення опціонів впливає ряд чинників. Перш за все це вартість базового активу S_t – страхування активу більшої вартості відповідно коштуватиме інвестору більшу, через більший ризик. Також на величину преміуму впливає страйкова ціна K – за скільки гарантовано можна продати актив (також пов'язано з ризиком). Також серед важливих чинників волатильність ринку σ , рівень процентних ставок r , час до закінчення терміну дії опціону Y .

- **Внутрішня вартість кол або грошовість (moneyness) опціону** – це різниця між вартістю активів і ціною виконання (strike price) [3].

Грошовість опціону можна знайти за формулою:

$$moneyness = \frac{S_t - K}{100} \quad (1)$$

де S_t – поточна ціна акції, K – страйкова ціна, ділимо на 100 для представлення у відсотках. Якщо грошовість позитивна кажуть, що опціон в грошах (in the money), якщо негативна – без грошей (out of the money).

1.2 Модель Блека-Шоулза як класична модель для ціноутворення

Модель Блека-Шоулза, також відома як модель Блека-Шоулза-Мертонна, є математичною моделлю ціноутворення похідних інвестиційних інструментів, таких як опціони. Ця модель була розроблена економістами Фішером Блеком і Майроном Шоулзом за участі Роберта Мертонна у 1973 році. Вона вважається класичною моделлю для ціноутворення опціонів. Розглянемо її для порівняння досліджуваної моделі з ринковим часом.

Модель Блека-Шоулза можна подати у вигляді диференціального рівняння:

$$\frac{dS}{S} = \mu dt + \sigma dW, \quad (2)$$

де:

S – ціна базового активу,

μ – очікувана віддача або дрейф,

σ – стандартне відхилення прибутковості активу (волатильності),

W – процес Вінера або броунівський рух.

Розв'язок цього рівняння дає формулу Блека-Шоулза для визначення ціни європейських опціонів кол і пут:

$$C_{BS} = Pe^{-\delta(Y-t)}N(d_1) - Ke^{-\bar{r}(Y-t)}N(d_2). \quad (3)$$

Тут $N(*)$ – стандартна кумулятивна функція нормального розподілу,

$$d_1 = \frac{\log \frac{S}{K} + (\bar{r} - \delta + \frac{1}{2} \sigma^2)(Y-t)}{\sigma \sqrt{Y-t}},$$

$$d_2 = d_1 - \sigma \sqrt{Y-t} \text{ [4]}.$$

Для того, щоб отримати рівняння (3) як розв'язок диференціального рівняння (2), робимо заміну параметра μ на різницю $(\bar{r} - \delta)$, де \bar{r} – безризикова процентна ставка, а δ – дивідендна дохідність [4].

Дана математична модель розраховує теоретичну ціну європейських кол і пут опціонів, які можна реалізувати лише після закінчення терміну дії договору. Формула для розрахунку ціни кол опціону включає кілька змінних – поточну ціну акцій S , страйкову ціну K , час до закінчення терміну дії договору Y , безризикову процентну ставку r та волатильність σ [5]. Волатильність найкраще визначити як величину мінливості прибутку на конкретний актив. Фактична волатильність – це фактичний рух ціни, який відбувається на ринку. Його часто називають історичним. Передбачувана волатильність – це параметр волатильності, отриманий від преміуму на певний термін погашення [3].

Для моделі Блека-Шоулза припускаємо, що фінансові ринки є ефективними і що ціна базового активу (наприклад акцій) відповідає геометричному броунівському руху з постійною волатильністю. Крім того, безризикова ставка та волатильність базового активу відомі та є константами, а доходи від базового активу розподілені за нормальним розподілом [5].

1.3 Недоліки моделі Блека-Шоулза

Незважаючи на те, що модель Блека-Шоулза широко використовується, вона має обмеження та припущення, які не справедливі в усіх ринкових умовах. Наприклад, передбачається постійна волатильність і логарифмічний нормальний розподіл цін на активи, що не завжди справджується на реальних ринках.

Розглянемо недоліки моделі Блека-Шоулза більш детально:

- **Припущення про постійну волатильність:** модель Блека-Шоулза припускає, що волатильність базового активу є постійною (або константою) з часом. Однак насправді волатильність має тенденцію до коливань і може різко змінюватися у відповідь на події, що відбуваються на ринку. Це

явище відоме як «усмішка волатильності» або «перекіс волатильності» [3].

- **Припущення логарифмічного нормального розподілу:** модель припускає, що доходи від базового активу розподілені за нормальним розподілом. Насправді прибутковість активів часто демонструє асиметрію та ексцес, які відхиляються від нормального розподілу. Вони також можуть мати товсті хвости, тобто з більшою ймовірністю чутливі до екстремальних значень, ніж можна було б передбачити за нормальним розподілом [3].
- **Припущення про безризикову ставку:** модель припускає, що безризикова процентна ставка відома та постійна протягом терміну дії опціону. Насправді процентні ставки можуть змінюватися з часом [3].
- **Припущення про відсутність дивідендів:** класична модель Блека-Шоулза припускає, що базовий актив не виплачує дивідендів. Хоча існують модифікації моделі, які можуть включати дивіденди [6].
- **Припущення щодо ефективних ринків:** модель припускає, що ринки є ефективними, тобто вся інформація, яка може вплинути на ціну, відображається одразу й повному обсязі в цінах активів. Однак насправді ринки можуть бути неефективними через різні фактори, такі як асиметрія інформації, транзакційні витрати, поведінкові упередження інвесторів тощо [3].
- **Припущення про відсутність транзакційних витрат:** модель припускає, що немає транзакційних витрат при купівлі чи продажу опціону або базового активу. Насправді ж

транзакційні витрати можуть бути значними та впливати на прибутковість опціонної стратегії [6].

1.4 Модель з активним ринковим часом

При ціноутворенні виникає ще одна проблема: точність моделі Блека-Шоулза при передбаченні зміни ціни. Тож, в роботі з моделями ціноутворення деривативів, з'являються модифікації моделей, які варіюються від простих практичних правил до більш повних багатофакторних моделей.

Розглянемо модель з активним ринковим часом для ціни акції P в момент часу t

$$S_t = S_0 e^{\mu t + \theta T_t + \sigma W_{T_t}}, \quad t \geq 0, \quad (4)$$

з параметрами $\mu \in R$ – відхилення, $\sigma > 0$ – дифузія та $\theta \in R$, що визначає асиметрію. $W_t, t \geq 0$, – стандартний броунівський рух, незалежний від процесу T_t [7]. $T_t, t \geq 0$, – випадкова зміна часу або час фрактальної активності. Іншими словами, T_t – позитивний неспадний стохастичний процес, такий що $T_0 = 0$. Цей процес має стаціонарні, але необов'язково незалежні прирости $\tau_t = T_t - T_{t-1}$. Час T_t – це час коли ринок еволюціонує, інтерпретуємо його як «ринковий» або «активний» час. Він відображає обсяг або потік інформації, яка впливає на ціни акцій. Час T_t можна виразити наступною формулою:

$$T_0 = 0, \quad T_t = \sum_{i=1}^{[t]} \tau_i + \tau_{[t]+1}(t - [t]).$$

Прирости часу для T_t τ_t розподілені за оберненим гамма розподілом [7].

Рівняння для прогнозування ціни акції (4), при виконанні певних умов, є сильним розв'язком стохастичного диференціального рівняння [8]

$$\frac{dS_t}{S_t} = \mu dt + \left(\theta + \frac{\sigma^2}{2} \right) dT_t + \sigma dW_{T_t}, \quad t \geq 0. \quad (5)$$

Це означає, що за допомогою рівняння (4) можна точно відтворити траєкторії ціни акції S_t , що неможливо зробити за допомогою слабких розв'язків стохастичного диференціального рівняння (5).

Для неризикових активів з відсотковою ставкою r рівняння для визначення ціни акції матиме вигляд [7]:

$$B_t = B_0 e^{rt}$$

Модель, задана рівнянням (4) та диференціальним рівнянням (5), є досить поширеною моделлю фінансового ринку, особливо, якщо говоримо про «активний» час. У даній моделі, параметр μ відповідає за детерміновану складову, відображає загальну тенденцію зростання або спадання ціни акції, $\frac{\sigma^2}{2}$ відображає волатильність ціни, а параметр θT_t представляє фрактальну складову. Крім того, рівняння (4) можемо інтерпретувати як зміну ціни акції відносно початкової ціни S_0 на момент часу t . Це є експоненційною функцією від детермінованої складової, фрактальної складової та випадкової складової. Випадкова складова моделюється за допомогою броунівського руху W_t – стандартний процес Броунівського руху [9].

Застосування моделі з ринковим часом дозволяє враховувати різні особливості цінових траєкторій та волатильність фінансових активів. Це важливо для аналізу ринків та прийняття рішень в інвестиційній діяльності.

Оскільки час T_t має фрактальну природу, рівняння (4) є прикладом моделювання «активний» руху ціни активу. «Активний» рух відрізняється від класичного випадкового блукання тим, що він має більш складну

структуру, характерну для систем, які мають самоподібність (або автомодельність) на різних масштабах.

«Активний» час T_t є головною відмінністю моделі з ринковим часом від класичної моделі Блека-Шоулза.

1.5 Визначення параметрів моделі

Для знаходження параметрів моделі розглянемо логарифмічні дохідності

$$X_t = \log\left(\frac{P_t}{P_{t-1}}\right). \quad (6)$$

Логарифм (натуральний) двох цін відповідатиме відсотковому прибутку [3].

Параметр σ – волатильність зазвичай оцінюється як

$$\sigma_x = \sqrt{\frac{1}{n-1} \sum_{t=1}^n (x_t - \bar{x})^2},$$

де

$$\bar{x} = \sum_{t=1}^n \frac{x_t}{n}.$$

$(n - 1)$ використовується замість n через втрату одного ступеня свободи при оцінці середнього x . Щоб перевести на річний показник, потрібно помножити на коефіцієнт річного перерахування. Якщо доходи від обчислення волатильності є щоденними, без вихідних рік вважається 252 дні [3].

З формули (4) отримаємо, що логарифмічні дохідності X_t з формули (6) будуть рівні за розподілом [7]

$$\mu + \theta\tau_t + \sigma\tau_t^{\frac{1}{2}}W_1.$$

Якщо τ_t підкорюється оберненому гамма розподілу $R\Gamma\left(\frac{\nu}{2}, \frac{\delta^2}{2}\right)$ [10], де $\nu > 2$, $\delta > 0$, з щільністю:

$$f_{R\Gamma}(x, \alpha, \beta) = \frac{(\alpha)^\beta}{\Gamma(\beta)} x^{-\beta-1} e^{-\alpha x} I_{x>0},$$

де $\alpha = \frac{\nu}{2}$, $\beta = \frac{\delta^2}{2}$, I – індикатор, лог-дохідності матимуть розподіл Стюдента $T(\nu, \mu, \delta)$ з параметрами ν, μ та δ [7]. Таким чином, параметри моделі (4) знаходимо як параметри розподілу Стюдента для лог-дохідностей X_t .

Тоді параметр σ для нашої моделі визначається як

$$\sigma = \sqrt{252\delta}$$

1.6 Ціноутворення в моделі з ринковим часом

Нехай $C(Y, K)$ – справедлива ціна європейського кол опціону, K – страйкова ціна і Y – час до використання. Для моделі з ринковим часом $C(Y, K)$ визначається за формулою [7]

$$C(Y, K) = \int_0^\infty (P_0 N(d_1) - K e^{-rY} N(d_2)) f_{T_Y}(t) dt. \quad (7)$$

Тут, як і в моделі Блека-Шоулза, $N(*)$ – стандартна кумулятивна функція нормального розподілу для d_1 і d_2 , які є функціями від t і задані наступними формулами:

$$d_1 = \frac{\log \frac{P_0}{K} + rY + \frac{1}{2} \sigma^2 t}{\sigma \sqrt{t}}, \quad d_2 = \frac{\log \frac{P_0}{K} + rY - \frac{1}{2} \sigma^2 t}{\sigma \sqrt{t}}.$$

$f_{T_Y}(t)$ – це щільністю «активного» часу T_Y . Вона визначена як

$\frac{1}{\sqrt{Y}} f_{R\Gamma}\left(\frac{u - E(\sqrt{Y} - Y)}{\sqrt{Y}}, \frac{\nu}{2}, \frac{\delta^2}{2}\right)$, де $f_{R\Gamma}$ – функція щільності ймовірності для

оберненого гамма-розподілу $R\Gamma(\beta, \alpha)$ задана формулою

$$f_{RG}(x, \alpha, \beta) = \frac{(\alpha)^\beta}{\Gamma(\beta)} x^{-\beta-1} e^{-\alpha*x} I_{x>0},$$

і $E[\tau_1^{(1)}]$ для $RG\left(\frac{\nu}{2}, \frac{\delta^2}{2}\right)$ визначене як $E[\tau_1^{(1)}] = \frac{\delta^2}{\nu-2}$ [7].

1.7 Статистичні похибки та регресійний аналіз

Для оцінка якості моделі обраховуємо два показники статистичної похибки:

$$RPE = \frac{\hat{p}_i - p_i}{p_i} \quad (8)$$

$$ARPE = \left| \frac{\hat{p}_i - p_i}{p_i} \right| \quad (9)$$

де \hat{p}_i - теоретична ціна і p_i - реальна (ринкова).

Відносна похибка ціноутворення (relative pricing errors) RPE є мірою упередженості моделі ціноутворення. Таким чином, ненульовий RPE може вказувати на наявність систематичних помилок.

Абсолютна відносна похибка ціноутворення (absolute relative pricing errors) – ARPE вимірює як упередженість, так і ефективність ціноутворення. Перевага ARPE порівняно з середньою квадратичною помилкою полягає в тому, що ця похибка дає відсоткове значення відхилення змодельованого ціноутворення від реальних значень. Таким чином, його можна легше інтерпретувати [4].

Щоб вивчити помилки моделей ціноутворення опціонів, розглянемо регресію для ARPE на час до завершення дії опціону (Time to maturity) в роках та грошовість (Moneyness) опціону .

$$ARPE = a_0 + a_1 \text{Time to maturity} + a_2 \text{Moneyness} + \epsilon, \quad (10)$$

де $\epsilon \sim N(0, \sigma^2)$. Для даної регресії необхідно визначити параметри a_0, a_1, a_2 .

За допомогою рівняння регресії можемо дослідити, як зміна терміну придатності чи грошовості опціону буде впливати на точність моделі – до чого модель більш чутлива; можемо зробити висновок щодо теоретичних цін моделі, близькості реальних значень, які спостерігаються на ринку; також, чи є помилки систематичними [4].

РОЗДІЛ 2. Аналіз якості моделі на основі реальних фінансових даних

Раніше були розглянуті дві математичні моделі для ціноутворення – модель Блека-Шоулза та модель з ринковим часом. В цьому розділі дані моделі застосовуємо для обрахунку теоретичних цін опціонів, порівнюємо результати для двох моделей. Отримані преміуми порівнюємо з ринковими цінами на опціони та досліджуємо статистичні похибки для обох моделей, проведемо регресійний аналіз.

2.1 Ринкові дані

Для дослідження була обрана компанія Netflix (Netflix, Inc. Common Stock (NFLX)). Протягом обраного періоду часу – з 9 лютого 2023 року до 17 лютого 2023 року – 7 робочих днів – ринкові ціни опціонів компанії спостерігалися на сайті Nasdaq (<https://www.nasdaq.com>). Даний ресурс містить дані про проточну ціну акцій, ціни кол та пут опціонів для різних страйкових цін, історичні дані щодо цін акцій, зміна ціни з часом – графіки та діаграми тощо. Кожного дня був зроблений скріншот цін на опціони, як приклад Рис. 2.1.1, отримані дані зібрані в таблицю (див. додаток В).

Exp. Date	Calls						Puts						
	Last	Change	Bid	Ask	Volume	Open Int.	Strike	Last	Change	Bid	Ask	Volume	Open Int.
February 17, 2023													
Feb 17	38.20	+0.20 ▲	37.10	37.90	8	836	330.00	0.75	-0.10 ▼	0.72	0.74	390	3085
Feb 17	40.00	+8.60 ▲	34.80	35.45	5	81	332.50	0.89	-0.14 ▼	0.87	0.90	82	630
Feb 17	33.00	-1.41 ▼	32.55	33.15	415	5629	335.00	1.10	-0.13 ▼	1.05	1.08	347	6916
Feb 17	33.80	+1.80 ▲	30.25	30.75	6	136	337.50	1.31	-0.13 ▼	1.27	1.30	89	538
Feb 17	34.00	+4.15 ▲	28.05	28.55	14	1240	340.00	1.53	-0.22 ▼	1.53	1.57	579	1891
Feb 17	26.35	+5.05 ▲	25.80	26.50	29	82	342.50	1.86	-0.19 ▼	1.85	1.89	218	585
Feb 17	24.55	+1.65 ▲	23.75	24.40	33	1520	345.00	2.30	-0.14 ▼	2.22	2.28	398	866
Feb 17	23.60	+3.10 ▲	21.70	22.30	7	535	347.50	2.71	-0.19 ▼	2.66	2.73	299	478
Feb 17	19.85	-0.95 ▼	19.80	20.15	213	1826	350.00	3.20	-0.25 ▼	3.15	3.25	1022	1382

Рис. 2.1.1 Скріншот з ринковими цінами

Крім того для дослідження взяли історичні дані про ціни акцій для отримання параметрів (див. додаток С), а також окремо були виписані поточні ціни акцій для кожного дня. Отримала вибірка містить дані про

європейські опціони. Сформовані файли з розширенням .xlsx були використані безпосередньо в кодї (див. додаток А).

Після видалення рядків зі значеннями NAN отримали 22 релевантні набори цін опціонів на 7 днів для 22 різних страйкових цін (див. Рис. 2.1.2). З них для дослідження обрали 10.

Date	2023-02-09	2023-02-10	2023-02-13	2023-02-14	2023-02-15	2023-02-16	2023-02-17
Strike							
330.0	38.2	33.95	29.58	29.0	28.86	28.75	17.49
332.5	40.0	40.0	27.75	27.68	27.99	28.85	14.94
335.0	33.0	26.35	25.75	25.46	24.79	24.47	12.74
337.5	33.8	33.8	23.92	23.13	22.75	18.2	11.0
340.0	34.0	22.28	21.25	21.17	19.16	18.25	8.2
342.5	26.35	23.15	18.0	18.5	17.4	16.3	5.65
345.0	24.55	19.25	17.7	16.5	15.03	13.46	3.0
347.5	23.6	15.85	15.4	14.5	12.98	11.04	0.51
350.0	19.85	14.1	13.5	12.97	11.34	9.4	0.01
352.5	18.7	12.5	11.8	11.2	8.55	7.0	0.01
355.0	16.88	10.89	10.4	9.8	6.33	5.5	0.03
357.5	15.4	9.65	8.98	8.34	5.02	3.92	0.01
360.0	12.82	8.3	7.55	7.21	3.98	2.55	0.01
362.5	11.45	7.2	6.46	6.0	4.15	1.72	0.01
365.0	9.91	6.3	5.4	5.03	2.78	1.01	0.01
367.5	8.85	5.5	4.5	4.1	1.65	0.64	0.01
370.0	7.65	4.5	3.7	3.34	1.43	0.39	0.01
372.5	6.65	3.95	3.03	2.72	1.02	0.24	0.01
375.0	5.81	3.22	2.45	2.16	0.99	0.14	0.01
377.5	4.95	2.71	1.91	1.77	0.56	0.09	0.01
380.0	4.24	2.26	1.53	1.34	0.43	0.05	0.01
382.5	3.6	1.91	1.21	1.06	0.35	0.03	0.02

Рис. 2.1.2 Спостережувані ціни опціонів.

2.2 Бібліотеки та змінні

Для написання практичної частини цієї роботи була використана мова програмування Python – вона має зручний лаконічний синтаксис і багато інструментів для статистичного аналізу та візуалізації даних.

В кодї практичної частини використовуються наступні бібліотеки:

- numpy (np) – бібліотека Python для обробки масивів
- math – стандартна бібліотека Python, яка надає математичні функції
- statistics – бібліотека Python для виконання статистичних операцій

- `scipy.stats` – модуль бібліотеки SciPy, який надає функції для статистичного аналізу
- `scipy.integrate` – модуль бібліотеки SciPy, який надає функції для чисельного інтегрування
- `sympy (sp)` – це бібліотека Python для символьних обчислень
- `matplotlib` – бібліотека Python для створення статичних візуалізацій
- `pandas (pd)` – бібліотека Python для обробки і аналізу даних
- `mibian` – бібліотека Python для оцінки опціонів за моделлю Black-Scholes
- `sklearn.linear_model` – модуль бібліотеки scikit-learn, який надає функції для лінійної регресії
- `mpl_toolkits.mplot3d` – модуль бібліотеки matplotlib, який надає функції для тривимірної візуалізації

Крім того, напочатку задаємо значення деяких змінних (Рис. 2.2.1).

```

Y = 7/252
P0 = 362.5
r = 0.05
delta = 2.5
dt = 1/252
T = 7

```

Рис. 2.2.1. Задані змінні

2.3 Знаходження параметрів моделі

Для знаходження параметрів моделі завантажуюємо історичні дані за останній рік та застосовуємо до них формулу (6) для обрахунку лог-дохідностей (Рис. 2.3.1).

```

def get_params(data):
    log_returns = np.log([data/data.shift(1)])[0][1:]
    nu, mu, delta = t.fit(log_returns)
    #x, loc, scale
    sigma = math.sqrt(delta*252)
    return nu, mu, sigma

```

Рис. 2.3.1 Функція для знаходження параметрів моделі з ринковим часом.

Отримуємо параметри ν , μ та σ .

2.4 Розрахунок преміумів для опціонів

За початкову ціну P_0 беремо ціну акції 9 лютого 2023 року. На основі цього значення розраховуємо теоретичні ціни опціонів на всі 7 днів для моделі з ринковим часом та для моделі Блека-Шоулза.

Кол опціони для моделі Блека-Шоулза були розраховані за допомогою бібліотеки `mibian`.

Для функції `optionpricing` (Рис. 2.4.1) для знаходження справедливої ціни опціону для моделі з ринковим часом за формулою (7) було написано кілька допоміжних функцій – функція `fty` для знаходження щільності розподілу випадкової величини за оберненим гамма-розподілом та функції

```
def optionpricing(Y, K, P0, r, nu, delta, sigma):
    def integrand(t):
        fty_t = fty(Y, nu, delta, t)
        d1_t = d1(Y, K, P0, sigma, r, t)
        d2_t = d2(Y, K, P0, sigma, r, t)
        return (((P0 * norm.cdf(d1_t)) - (K * math.exp(-(r*Y)) * norm.cdf(d2_t))) * fty_t)
    # The [0] index is used to extract the value of 'y' from the tuple and return it
    # as the output of the optionpricing function.
    return integrate.quad(integrand, 0, np.inf)[0]
```

Рис. 2.4.1 Знаходження справедливої ціни опціону для моделі з ринковим часом

```
def fty(Y, nu, delta, t):
    x = ((t - ((delta*delta)/(nu-2)) * (math.sqrt(Y) - Y)) / math.sqrt(Y))
    if x <= 0:
        return 0
    else:
        return ( (math.pow(((delta*delta)/2), (nu/2)) / math.gamma(nu/2)) * (math.pow(x, -(nu/2)-1)) *
                (math.exp(-((delta*delta)/2/x))) )
```

Рис. 2.4.2 Знаходження щільності розподілу випадкової величини

```
def d1(Y, K, P0, sigma, r, t):
    return ((math.log(P0/K) + r*Y + ((sigma*sigma)*t)/2) / (sigma * math.sqrt(t)))

def d2(Y, K, P0, sigma, r, t):
    return ((math.log(P0/K) + r*Y - ((sigma*sigma)*t)/2) / (sigma * math.sqrt(t)))
```

Рис. 2.4.3 Обрахунок функцій $d1$ і $d2$ для формули (7)

$d1$ і $d2$ для обрахунку відповідних функцій в формулі справедливої ціни опціону (Рис. 2.4.2, Рис. 2.4.3).

Отримані значення для кол опціонів обраховані за обома моделями, а також ринкові ціни опціонів для порівняння зображаємо на одному графіку для кожної страйкової ціни (Рис. 2.4.4). З графіків бачимо, що

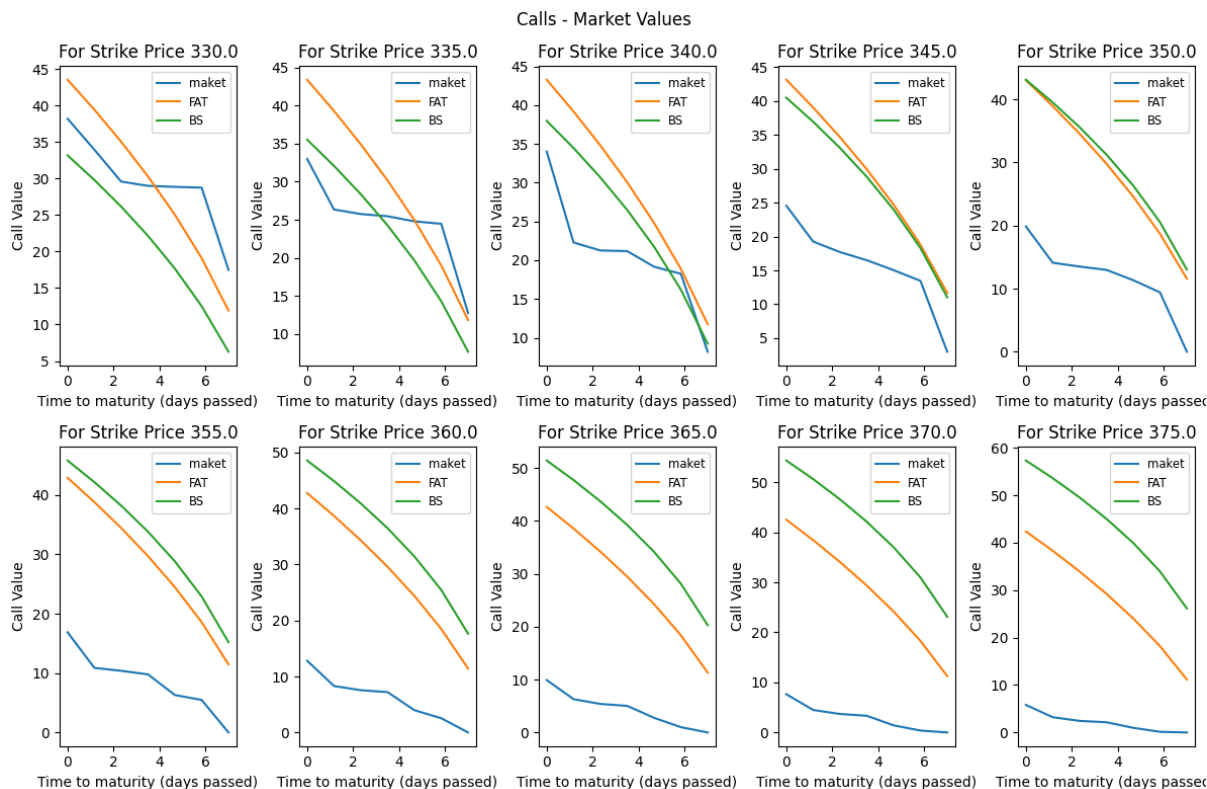


Рис. 2.4.4 Значення кол опціонів

існують суттєві розбіжності між реальними цінами та спрогнозованими. Більше того навіть значення обраховані за різними моделями не для всіх страйкових цін близькі.

2.5 Розрахунок статистичних похибок, їх характеристики

Для масивів преміумів кол опціонів, розрахованих за обома моделями, знаходимо статистичні похибки RPE та $APRE$ за формулами (8) і (9) (Рис. 2.5.1, Рис. 2.5.2). Для отриманих значень знаходимо статистичні характеристики – середнє, медіану, стандартне квадратичне відхилення. В таблиці 2.5.1 та таблиці 2.5.1 наведені результати. Бачимо, що середні значення дуже великі – 91.96 і 159.07 для RPE для моделей з ринковим часом та Блека-

```

def RPE(p_hat, p_i):
    return (p_hat - p_i)/p_i

def calculating_RPE(pred_calls, real_calls):
    rpes = []
    for j in range(len(pred_calls)):
        rpes.append([RPE(pred_calls[j][i], real_calls[j][i]) for i in range(len(real_calls[j]))])
    return rpes

```

Рис. 2.5.1 Функції для знаходження похибки *RPE*.

```

def ARPE(p_hat, p_i):
    return abs((p_hat - p_i)/p_i)

def calculating_ARPE(pred_calls, real_calls):
    arpes = []
    for j in range(len(pred_calls)):
        arpes.append([ARPE(pred_calls[j][i], real_calls[j][i]) for i in range(len(real_calls[j]))])
    return arpes

```

Рис. 2.5.2 Функції для знаходження похибки *ARPE*.

Шоулза відповідно; 91.99 та 159.16 для *ARPE* для моделей з ринковим часом та Блека-Шоулза відповідно. Це вказує на те, що присутні нереалістично великі значення в теоретичних цінах опціонів. Скоріш за все ці значення відповідають останнім дням, адже медіани відносно маленькі – 1.84 та 2.14 для обох похибок. Крім того, значення для *RPE* і *ARPE* дуже близькі – розподіли похибок симетричні. Оскільки, більшість

<i>RPE</i>	Mean	Median	Standard Deviation
FAT model	91.96	1.84	186.99
B-S model	159.07	2.14	350.45

таблиця 2.5.1 Статистичні характеристики похибки *RPE*

<i>ARPE</i>	Mean	Median	Standard Deviation
FAT model	91.99	1.84	187.01
B-S model	159.16	2.14	350.47

таблиця 2.5.2 Статистичні характеристики похибки *ARPE*

RPE додатні – обидві моделі завищують справедливую ціну опціонів (також спостерігаємо це на Рис. 2.4.4).

2.6 Обрахунок грошовості

Грошовість опціонів обраховуємо за формулою (1) використовуючи дані про реальні ціни акції для кожного дня (Рис. 2.6.1).

```
def calculate_moneyness(P0, K):
    return (P0 - K)/100

def moneyness_for_all_days(Ps, Ks):
    moneyness = np.zeros((len(Ks), len(Ps)))
    for j in range(len(Ks)):
        moneyness[j] = [calculate_moneyness(i, Ks[j]) for i in Ps]
    return moneyness
```

Рис. 2.6.1 Функції для обрахунку грошовості опціонів.

2.7 Регресійний аналіз

Для дослідження чутливості моделі, а також основних тенденцій, що яких схильна модель проведемо регресійний аналіз. Статистичні похибки регресуємо окремо на грошовість та на час до закінчення дії опціону, а також одночасно на обидва показники для кожної моделі. Для регресії застосовуємо бібліотеку sklearn.

Для кожної регресії необхідно знайти параметри a_0, a_1, a_2 для рівняння (10). Можемо записати рівняння прямих регресії:

- *ARPE* на *moneyness*
 - $ARPE = -694.34 \text{ moneyness} + 112.8$ – модель з ринковим часом, коефіцієнт регресії – 0.14
 - $ARPE = -1334.83 \text{ moneyness} + 199.17$ – модель Блека-Шоулза, коефіцієнт регресії – 0.15
- *ARPE* на *Time to maturity*
 - $ARPE = -65.94 \text{ Time to maturity} + 289.81$ – модель з ринковим часом, , коефіцієнт регресії – 0.2

- $ARPE = -114.93 \text{ Time to maturity} + 503.94$ – модель
Блека-Шоулза, коефіцієнт регресії – 0.19

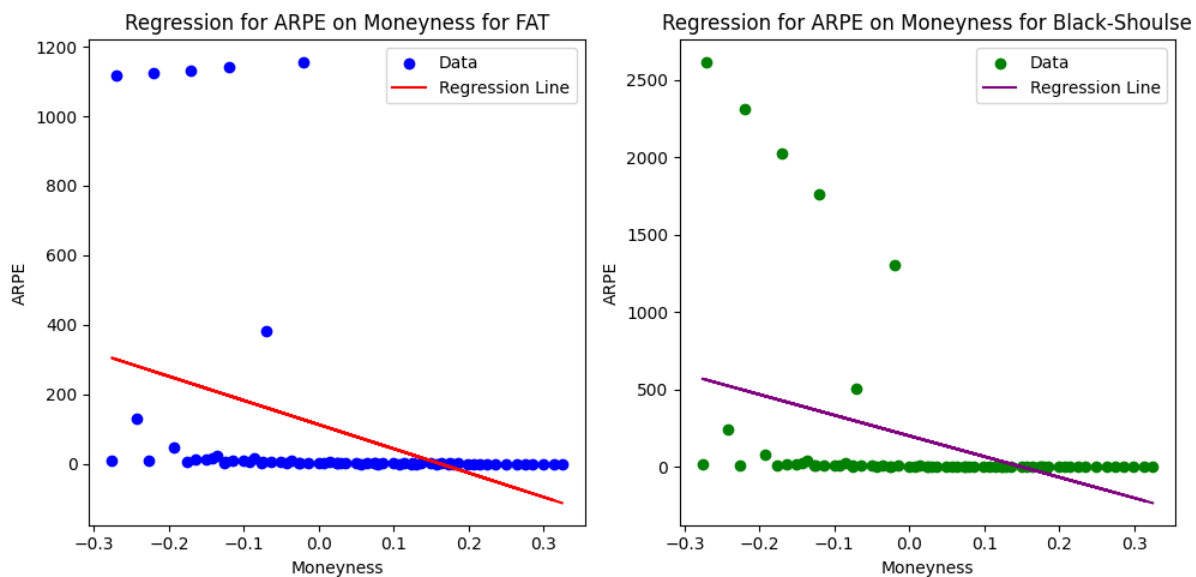


Рис. 2.7.1 Регресія $ARPE$ на $moneyness$.

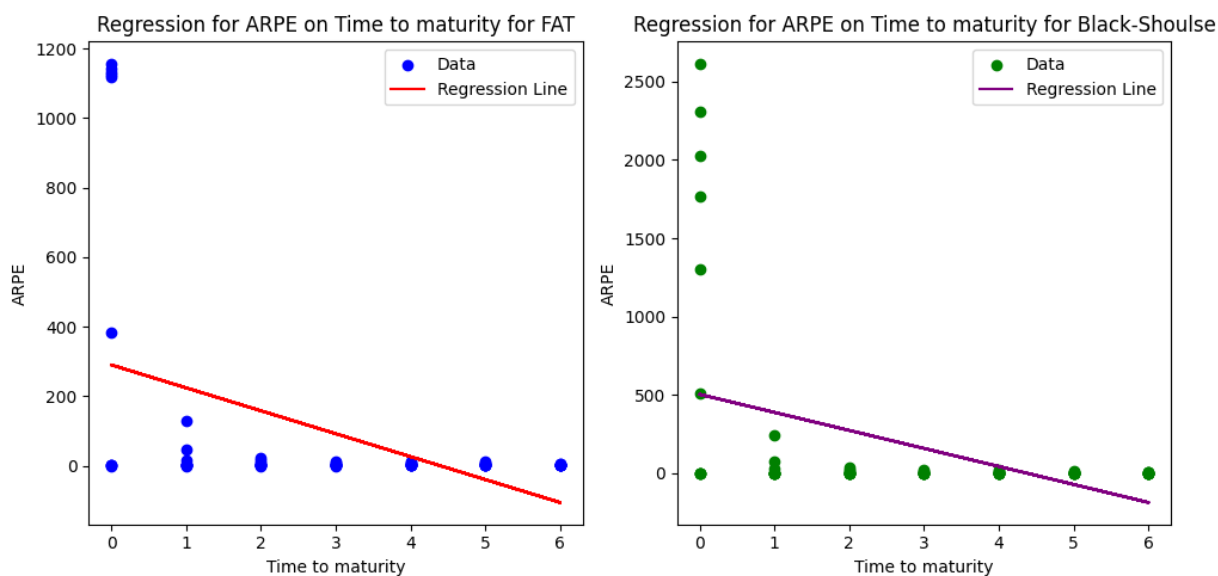


Рис. 2.7.2 Регресія $ARPE$ на $Time\ to\ maturity$.

Дані візуалізовані на Рис. 2.7.1 та Рис. 2.7.2. Бачимо, що коефіцієнти регресії дуже малі – залежність незначна. Це може бути зумовлено кількома нереалістично великими значеннями справедливої ціни кол опціонів,

отриманих за математичними моделями. Великі значення вкінці також вказують на те, що волатильність була визначена не зовсім вірно.

Регресія на обидва показники також не показала високих результатів (Рис. 2.7.3).

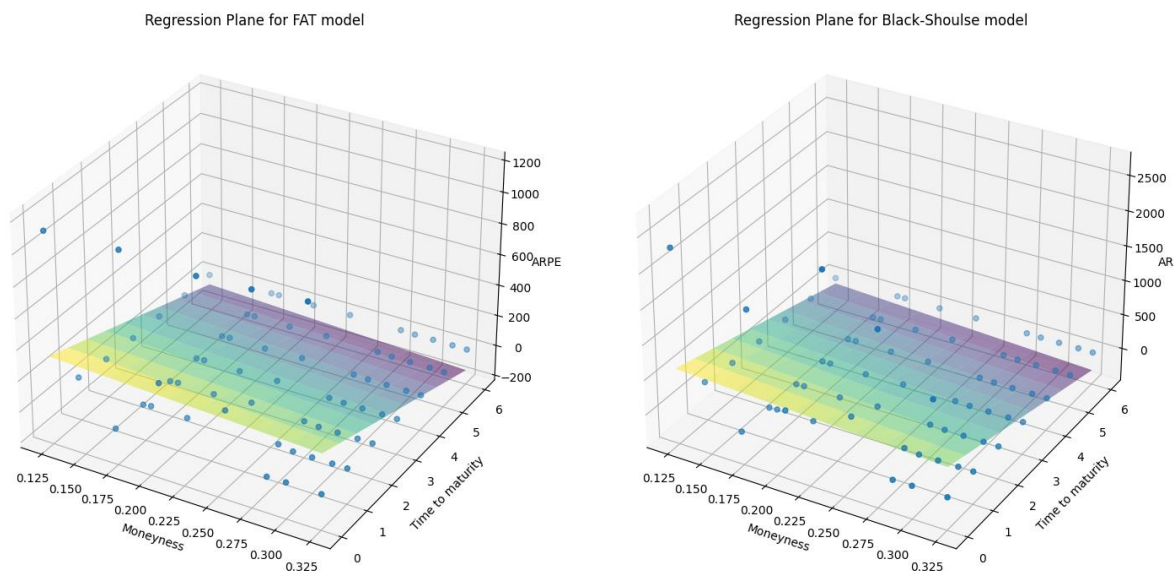


Рис. 2.7.3 Регресія $ARPE$ на $moneyness$ та $Time\ to\ maturity$.

Отримали рівняння для $ARPE$ обох моделей:

- $ARPE = -386.89\ moneyness - 65.94\ Time\ to\ maturity + 383.41$
– модель з ринковим часом, коефіцієнт регресії – 0.21
- $ARPE = -927.32\ moneyness - 114.93\ Time\ to\ maturity + 728.27$ – модель Блека-Шоулза, коефіцієнт регресії – 0.2

Коефіцієнти регресії трішки вищі, але не набагато.

Результати для моделі з ринковим часом та моделі Блека-Шоулза виявилися дуже схожими, відмінності несуттєві.

Можемо зробити висновки, щодо тенденцій притаманних обох моделям. Оскільки, коефіцієнти при на $moneyness$ та $Time\ to\ maturity$ від'ємні – це означає, що похибки зменшуються для обох моделей при зростанні грошовості та при зменшенні кількості днів до завершення дії опціону. Втім, оскільки коефіцієнти регресії не високі, можемо сказати, що моделі не є дуже чутливими до таких даних.

ВИСНОВКИ

Підсумовуючи зазначимо:

- В даній роботі було описано особливості моделі з ринковим часом, розглянуті особливості статистичних похибок RPE і $ARPE$, а також значення регресії в контексті ціноутворення.
- На основі реальних даних кол опціонів та цін акцій компанії Netflix, були застосовані теоретичні концепції для оцінки якості моделі з ринковим часом.
- В процесі були зібрані ринкові дані, визначені параметри моделі на основі історичних цін акцій.
- Було запрограмовано справедливу ціну для кол опціонів даної моделі.
- Справедлива ціна кол опціонів була розрахована за двома моделями – моделлю з ринковим часом та моделлю Блека-Шоулза. Отримані дані було проаналізовано та порівняно між собою та з ринковими реальними цінами кол опціонів.
- Для аналізу застосовувалися похибки RPE і $ARPE$, досліджувалися їх статистичні характеристики.
- Були створені необхідні графіки та таблиці.
- Був проведений регресійний аналіз, зроблені висновки щодо чутливості моделі.

На основі отриманих результатів були зроблені висновки про те, що обидві моделі, як модель з ринковим часом, так і модель Блека-Шоулза, схильні завишувати ціни на опціони. Не було виявлено значної залежності між статистичними похибками та грошовістю опціонів чи часом до закінчення строку дії опціону. Було зроблено висновок щодо неправдоподібності обрахованої волатильності. Було визначено, що обидві моделі показують кращий результат при збільшенні грошовості опціонів та

при зменшенні часу до закінчення строку дії опціону. Втім, не було виявлено високої чутливості в жодній з моделей.

Можна зробити загальний висновок, що за багатьма показниками модель з ринковим часом, як і модель Блека-Шоулза, показала результати далекі від реальності, хоча, значення обраховані за моделлю з ринковим часом були зазвичай ближчими до спостережуваних даних, ніж результати отримані за моделлю Блека-Шоулза. При ретельному підборі волатильності модель з ринковим часом прогнозує дані ближчі до реальних.

Для подальших досліджень залишається проблема визначення параметрів моделі, додавання нових параметрів для вдосконалення правдоподібності, адже при правильному підборі параметрів модель з ринковим часом (яка має більше параметрів) показує кращі результати ніж модель Блека-Шоулза.

Список використаної літератури

1. R. A. Brealey, S. C. Myers, F. Allen "Principles of Corporate Finance", McGraw-Hill Education (2016)
2. P. Wilmott "Derivatives: The Theory and Practice of Financial Engineering", Wiley (1998)
3. N. T. Nassim "Dynamic Hedging: Managing Vanilla and Exotic Options", Wiley (2007)
4. C. Schittenkopf, A. Lehar, M. Scheicher "GARCH vs Stochastic Volatility: Option Pricing and Risk Management. ", Journal of Banking & Finance, Volume 26, Issues 2–3, 323–345 p. (2002)
5. F. Black and M. Scholes "The Pricing of Options and Corporate Liabilities", The Journal of Political Economy, Vol. 81, No. 3, pp. 637-654, The University of Chicago Press (1973)
6. J. C. Hull "Options, Futures, and Other Derivatives" (10th ed.). Pearson (2017)
7. F. Castella, N. N. Leonenkob, and N. Shchestyukc "Student-like models for risky asset with dependence.", Stochastic Analysis and Applications, Volume 35, Issue 3, 452–464 p. (2017)
8. K. Kobayashi "Stochastic calculus for a time-changed semimartingale and the associated stochastic differential equations.", J. Theoret. Probab. 24(3):789–820, (2011)
9. I. Karatzas, S. E. Shreve "Brownian Motion and Stochastic Calculus", Springer (1987)
10. F.J. Giron, C. Castillo "A note on the convolution of inverted-gamma distributions with applications to the B-Fisher distribution", Statistic and Operations Research, 95(1), p.39-44. (2001)

Додаток А. Програмний код

Evaluation of the Fractal Activity Time Model

Libraries

```
##### from option_pricing
```

```
import numpy as np import math
```

```
import statistics
```

```
#from scipy.stats import gamma
```

```
from scipy.stats import norm
```

```
import scipy.integrate as integrate
```

```
import sympy as sp
```

```
import matplotlib
```

```
from matplotlib import pyplot as plt
```

```
from scipy.stats import invgamma from scipy.stats import norm from scipy.stats import t
```

```
import pandas as pd
```

```
from mibian import BS
```

```
from sklearn.linear_model import LinearRegression
```

```
import matplotlib.pyplot as plt
```

```
from mpl_toolkits.mplot3d import Axes3D
```

Defining variables

```
Y = 7/252
```

```
P0 = 362.5
```

```
r = 0.05
```

```
delta = 2.5
```

```
dt = 1/252
```

```
T=7
```

Functions

1. Functions for option pricing

```
def toOneDimension(arrey, rows, columns):
```

```
    res = np.zeros((rows*columns))
```

```
    i=0
```

```
    j=0
```

```
    for k in range(len(res)):
```

```
        res[k] = arrey[i][j]
```

```
        if j < (columns - 1):
```

```
            j += 1
```

```
        else:
```

```
            j=0
```

```
            i += 1
```

```
    return res
```

1.1 Density of the random variable

```
def fty(Y, nu, delta, t):
```

```
    x = ((t - ((delta*delta)/(nu-2)) * (math.sqrt(Y) - Y)) / math.sqrt(Y))
```

```
    if x <= 0:
```

```
        return 0
```

```
    else:
```

```
        return ( (math.pow(((delta*delta)/2), (nu/2)) / math.gamma(nu/2)) *  

        (math.pow(x, -(nu/2)-1)) * (math.exp(-(((delta*delta)/2)/x))) )
```

1.2 d-functions for FAT model

```
def d1(Y, K, P0, sigma, r, t):
```

```
    return ((math.log(P0/K) + r*Y + ((sigma*sigma)*t)/2) / (sigma * math.sqrt(t)))
```

```
def d2(Y, K, P0, sigma, r, t):
```

```
    return ((math.log(P0/K) + r*Y - ((sigma*sigma)*t)/2) / (sigma * math.sqrt(t)))
```

1.3 Call option price

```
def optionpricing(Y, K, P0, r, nu, delta, sigma):
```

```
    def integrand(t):
```

```
        fty_t = fty(Y, nu, delta, t)
        d1_t = d1(Y, K, P0, sigma, r, t)
        d2_t = d2(Y, K, P0, sigma, r, t)
        return (((P0 * norm.cdf(d1_t)) - (K * math.exp(-(r*Y)) * norm.cdf(d2_t))) * fty_t)
        # The [0] index is used to extract the value of 'y' from the tuple and return it
        # as the output of the optionpricing function.
```

```
    return integrate.quad(integrand, 0, np.inf)[0]
```

2. Getting parameters

```
def get_params(data):
```

```
    log_returns = np.log([data/data.shift(1)])[0][1:]
```

```
    nu, mu, delta = t.fit(log_returns) #x, loc, scale
    sigma = math.sqrt(delta*252)
```

```
    return nu, mu, sigma
```

3. Calculating theoretical calls (without generating stock prices) 3.1 Calculating theoretical calls for every strike price for FAT model

```
def calls_mt_for_one_K(P0, K, Y, r, nu, delta, sigma, T):
```

```
    calls_mt = np.zeros((T))
    dt = 0
```

```
    for i in range(T):
```

```
        calls_mt[i] = optionpricing((Y - dt), K, P0, r, nu, delta, sigma)
```

```
        dt += 1/252
```

```
    return calls_mt
```

```
def calls_mt_for_all_Ks(P0, Ks, Y, r, nu, delta, sigma):
```

```
    calls_table = np.zeros((len(Ks), T)) s = sigma
    #calls_table = []
```

```
    for i in range(len(Ks)):
```

```
        calls_table[i] = calls_mt_for_one_K(P0, Ks[i], Y, r, nu, delta, s, T)
```

```
    return calls_table
```

3.2 Calculating theoretical calls for every strike price for Black-Shoulse model

```

def calls_BS_for_all_Ks(Ks, P0, r, sigma, T):
    calls_BS_table = np.zeros((len(Ks), T))

    days = 0

    s = sigma*100

    for i in range(len(calls_BS_table)):
        for j in range(len(calls_BS_table[i])):
            if Y - days == 0:
                break

            c = BS([Ks[i], P0, (r*100), ((T - days) + 0.0001)], volatility = s)

            calls_BS_table[i][j] = c.callPrice

            days += 1

            days = 0

        return calls_BS_table

def calls_BS_for_all_KsAndPs(Ks, Ps, r, sigma, T):
    calls_BS_table = np.zeros((len(Ks), T))

    days = 0

    for i in range(len(calls_BS_table)):
        for j in range(len(calls_BS_table[i])):
            if Y - days == 0:
                break

            c = BS([Ks[i], Ps[j], (r*100), ((T - days) + 0.0001)], volatility = sigma)

            calls_BS_table[i][j] = c.callPrice

            days += 1

            days = 0

        return calls_BS_table

```

4. Functions for calculating statistical errors

4.1 Functions for calculating RPE

```

def RPE(p_hat, p_i):
    return (p_hat - p_i)/p_i

def calculating_RPE(pred_calls, real_calls):
    rpes = np.zeros((len(pred_calls), len(pred_calls[0])))
    for i in range(len(pred_calls)):
        for j in range(len(pred_calls[0])):
            rpes[i][j] = RPE(pred_calls[i][j], real_calls[i][j])
    return rpes

```

4.2 Functions for calculating ARPE

```

def ARPE(p_hat, p_i):
    return abs((p_hat - p_i)/p_i)

def calculating_ARPE(pred_calls, real_calls):
    arpes = np.zeros((len(pred_calls), len(pred_calls[0])))
    for i in range(len(pred_calls)):
        for j in range(len(pred_calls[0])):
            arpes[i][j] = ARPE(pred_calls[i][j], real_calls[i][j])
    return arpes

```

4.3 Functions for calculating moneyness

```

def calculate_moneyness(P0, K):
    return (P0 - K)/100

def moneyness_for_all_days(Ps, Ks):
    moneyness = np.zeros((len(Ks), len(Ps)))
    for j in range(len(Ks)):
        moneyness[j] = [calculate_moneyness(i, Ks[j]) for i in Ps]

```

```
return moneyness
```

Model evaluation

1. Importing data

1.1 Getting data from excel file

```
data = pd.read_excel(r'/Users/mariiadubnytska/Desktop/Diploma/Data_Call.xlsx')
data_table = pd.pivot_table(data = data, values = 'Last', index = 'Strike', columns = 'Date',
fill_value="NaN")
#data_table
```

1.2 Cleaning data

```
clear_data_table = data_table.loc[~(data_table['2023-02-09'] == 'NaN')].loc[~(data_table['2023-02-17'] == 'NaN')] #clear_data_table
```

1.3 Choosing 10 rows

```
arrey_data = clear_data_table.to_numpy()
chosen_data_calls = [arrey_data[i] for i in range(0, 20, 2)]
```

```
chosen_calls = []
for i in range(len(chosen_data_calls)):
```

```
    chosen_calls.append([chosen_data_calls[i][j] for j in range(len(chosen_data_calls[i]))])
```

```
Ks = [330.0, 335.0, 340.0, 345.0, 350.0, 355.0, 360.0, 365.0, 370.0, 375.0]
Ps = [362.50, 347.36, 358.57, 359.96, 361.42, 350.71, 347.96]
```

2. Getting parameters from historical data

```
data = pd.read_excel(r'/Users/mariiadubnytska/Desktop/Diploma/HistoricalData.xlsx')
nu, mu, sigma = get_params(data["Last"])
print(nu, mu, sigma)
```

3. Calculatin theoretical calls

3.1 Theoretical calls for FAT model

```
calls_mt_table = calls_mt_for_all_Ks(P0, Ks, Y, r, nu, delta, sigma) #calls_mt_table
```

3.2 Theoretical calls for Black-Shoulse model

```
calls_BS_table = calls_BS_for_all_Ks(Ks, P0, r, sigma, T) #calls_BS_table
```

4. Visualisation of the calls

```
# x - days to expiration
x = np.linspace(0, T, T)
```

```
# market calls for each strike price
```

```
y1 = chosen_calls[0]
y2 = chosen_calls[1]
y3 = chosen_calls[2]
y4 = chosen_calls[3]
y5 = chosen_calls[4]
y6 = chosen_calls[5]
y7 = chosen_calls[6]
y8 = chosen_calls[7]
y9 = chosen_calls[8]
y10 = chosen_calls[9]
```

```
# FAT calls for each strike price
```

```
y_mt1 = calls_mt_table[0]
y_mt2 = calls_mt_table[1]
y_mt3 = calls_mt_table[2]
y_mt4 = calls_mt_table[3]
y_mt5 = calls_mt_table[4]
y_mt6 = calls_mt_table[5]
y_mt7 = calls_mt_table[6]
y_mt8 = calls_mt_table[7]
y_mt9 = calls_mt_table[8]
y_mt10 = calls_mt_table[9]
```

```
# BS calls for each strike price
```

```
y_BS1 = calls_BS_table[0]
y_BS2 = calls_BS_table[1]
y_BS3 = calls_BS_table[2]
y_BS4 = calls_BS_table[3]
y_BS5 = calls_BS_table[4]
y_BS6 = calls_BS_table[5]
y_BS7 = calls_BS_table[6]
y_BS8 = calls_BS_table[7]
y_BS9 = calls_BS_table[8]
y_BS10 = calls_BS_table[9]
```

```
# create a figure and a set of subplots
```

```
fig, axs = plt.subplots(nrows=2, ncols=5, figsize=(12, 8)) # 2 columns for 2 plots
```

```
# add labels
```

```
fig.suptitle('Calls - Market Values')
fig.subplots_adjust(top=0.88)
```

```
for i in range(2):
```

```
    for j in range(5):
```

```

        axs[i, j].set_title(f'For Strike Price {Ks[i*5 + j]}')

for i in range(2):

    for j in range(5):

        axs[i, j].set_xlabel('Time to maturity (days passed)')
        axs[i, j].set_ylabel('Call Value')

# plot data on each subplot
axs[0, 0].plot(x, y1, label='maket')
axs[0, 0].plot(x, y_mt1, label='FAT')
axs[0, 0].plot(x, y_BS1, label='BS')
axs[0, 1].plot(x, y2, label='maket')
axs[0, 1].plot(x, y_mt2, label='FAT')
axs[0, 1].plot(x, y_BS2, label='BS')
axs[0, 2].plot(x, y3, label='maket')
axs[0, 2].plot(x, y_mt3, label='FAT')
axs[0, 2].plot(x, y_BS3, label='BS')
axs[0, 3].plot(x, y4, label='maket')
axs[0, 3].plot(x, y_mt4, label='FAT')
axs[0, 3].plot(x, y_BS4, label='BS')
axs[0, 4].plot(x, y5, label='maket')
axs[0, 4].plot(x, y_mt5, label='FAT')
axs[0, 4].plot(x, y_BS5, label='BS')
axs[1, 0].plot(x, y6, label='maket')
axs[1, 0].plot(x, y_mt6, label='FAT')
axs[1, 0].plot(x, y_BS6, label='BS')
axs[1, 1].plot(x, y7, label='maket')
axs[1, 1].plot(x, y_mt7, label='FAT')
axs[1, 1].plot(x, y_BS7, label='BS')
axs[1, 2].plot(x, y8, label='maket')
axs[1, 2].plot(x, y_mt8, label='FAT')
axs[1, 2].plot(x, y_BS8, label='BS')
axs[1, 3].plot(x, y9, label='maket')
axs[1, 3].plot(x, y_mt9, label='FAT')
axs[1, 3].plot(x, y_BS9, label='BS')
axs[1, 4].plot(x, y10, label='maket')
axs[1, 4].plot(x, y_mt10, label='FAT')
axs[1, 4].plot(x, y_BS10, label='BS')

# add legend

for i in range(2):

    for j in range(5):

        axs[i, j].legend(fontsize='small', loc='upper right')

# display the figure containing the subplots

plt.tight_layout()
plt.show()

```

5. Statistical errors

5.1 Calculating statistical errors *RPE* and *ARPE* for FAT model

```
RPEs_mt = calculating_RPE(calls_mt_table, chosen_calls)
ARPEs_mt = calculating_ARPE(calls_mt_table, chosen_calls)
```

5.2 Calculating statistical errors *RPE* and *ARPE* for Black-Shoulse model

```
RPEs_BS = calculating_RPE(calls_BS_table, chosen_calls)

ARPEs_BS = calculating_ARPE(calls_BS_table, chosen_calls)
```

5.3 Getting statistical characteristics for calculated errors

```
# RPEs_mt
```

```
means = [statistics.mean(i) for i in RPEs_mt] mean = statistics.mean(means)

medians = [statistics.median(i) for i in RPEs_mt] median = statistics.median(medians)

std_devs = [np.std(i) for i in RPEs_mt] std_dev = np.std(std_devs)

print("RPEs_mt Mean:", mean)
print("RPEs_mt Median:", median)
print("RPEs_mt Standard Deviation:", std_dev)
print("")
```

```
# ARPEs_mt
```

```
means = [statistics.mean(i) for i in ARPEs_mt] mean = statistics.mean(means)

medians = [statistics.median(i) for i in ARPEs_mt] median = statistics.median(medians)

std_devs = [np.std(i) for i in ARPEs_mt] std_dev = np.std(std_devs)

print("ARPEs_mt Mean:", mean)
print("ARPEs_mt Median:", median)
print("ARPEs_mt Standard Deviation:", std_dev)
print("")
```

```
# RPEs_BS
```

```
means = [statistics.mean(i) for i in RPEs_BS] mean = statistics.mean(means)

medians = [statistics.median(i) for i in RPEs_BS] median = statistics.median(medians)

std_devs = [np.std(i) for i in RPEs_BS] std_dev = np.std(std_devs)

print("RPEs_BS Mean:", mean)
print("RPEs_BS Median:", median)
print("RPEs_BS Standard Deviation:", std_dev)
```

```

print("")

# ARPEs_BS

means = [statistics.mean(i) for i in ARPEs_BS] mean = statistics.mean(means)

medians = [statistics.median(i) for i in ARPEs_BS] median = statistics.median(medians)

std_devs = [np.std(i) for i in ARPEs_BS] std_dev = np.std(std_devs)

print("ARPEs_BS Mean:", mean)
print("ARPEs_BS Median:", median)
print("ARPEs_BS Standard Deviation:", std_dev)
print("")

```

6. Moneyiness

```
moneyiness = moneyiness_for_all_days(Ps, Ks)
```

7. Regression

7.1 Data for regression

```
ARPEs_mt_for_regesion = toOneDimension(ARPEs_mt, len(ARPEs_mt), len(ARPEs_mt[0]))
```

```
y_mt = ARPEs_mt_for_regesion
```

```
ARPEs_BS_for_regesion = toOneDimension(ARPEs_BS, len(ARPEs_BS), len(ARPEs_BS[0]))
```

```
y_BS = ARPEs_BS_for_regesion
```

```
# moneyiness separately
```

```
Xm = toOneDimension(moneyiness, len(moneyiness), len(moneyiness[0]))
```

```
# time to maturity separately
```

```
Xt = np.zeros(len(moneyiness)*T)
```

```
count = 6
```

```
for i in range(len(Xt)):
```

```
    Xt[i] = count
```

```
    if count > 0:
```

```
        count -= 1
```

```
    else: count = 6
```

```
X = []

for i in range(len(moneyness)):

    for j in range(T):

        X.append([Xm[i], (T-1-j)])
```

7.2 Regresing *ARPE*s for FAT model and on moneyness

```
reg_m_mt = LinearRegression().fit(Xm.reshape((-1, 1)), y_mt)

print(reg_m_mt.score(Xm.reshape((-1, 1)), y_mt))

print(reg_m_mt.coef_)

print(reg_m_mt.intercept_)
```

7.3 Regresing *ARPE*s for Black-Shouls model on moneyness

```
reg_m_BS = LinearRegression().fit(Xm.reshape((-1, 1)), y_BS)

print(reg_m_BS.score(Xm.reshape((-1, 1)), y_BS))

print(reg_m_BS.coef_)

print(reg_m_BS.intercept_)
```

7.4 Visualising regression on moneyness

```
# Creating subplots
fig, axs = plt.subplots(nrows=1, ncols=2, figsize=(10, 5))
# Subplot for y_mt
axs[0].scatter(Xm, y_mt, color='blue', label='Data')
axs[0].plot(Xm, reg_m_mt.predict(Xm.reshape((-1, 1))), color='red', label='Regression Line')
axs[0].set_title('Regression for ARPE on Moneyness for FAT')
axs[0].set_xlabel('Moneyness')
axs[0].set_ylabel('ARPE')
axs[0].legend()
# Subplot for y_BS
axs[1].scatter(Xm, y_BS, color='green', label='Data')
axs[1].plot(Xm, reg_m_BS.predict(Xm.reshape((-1, 1))), color='purple', label='Regression Line')
axs[1].set_title('Regression for ARPE on Moneyness for Black-Shoulse')
axs[1].set_xlabel('Moneyness')
axs[1].set_ylabel('ARPE')
axs[1].legend()
plt.tight_layout()
plt.show()
```

7.5 Regresing *ARPE*s for FAT model on Time to maturity

```
reg_t_mt = LinearRegression().fit(Xt.reshape((-1, 1)), y_mt)
```

```
print(reg_t_mt.score(Xt.reshape((-1, 1)), y_mt))
```

```
print(reg_t_mt.coef_)
```

```
print(reg_t_mt.intercept_)
```

7.6 Regresing *ARPE*s for Black-Shouls model on Time to maturity

```
reg_t_BS = LinearRegression().fit(Xt.reshape((-1, 1)), y_BS)
```

```
print(reg_t_BS.score(Xt.reshape((-1, 1)), y_BS))
```

```
print(reg_t_BS.coef_)
```

```
print(reg_t_BS.intercept_)
```

7.4 Visualising regression on Time to maturity

```
# Creating subplots
```

```
fig, axs = plt.subplots(nrows=1, ncols=2, figsize=(10, 5))
```

```
# Subplot for y_mt
```

```
axs[0].scatter(Xt, y_mt, color='blue', label='Data')
```

```
axs[0].plot(Xt, reg_t_mt.predict(Xt.reshape((1, 1))), color='red', label='Regression Line')
```

```
axs[0].set_title('Regression for ARPE on Time to maturity for FAT')
```

```
axs[0].set_xlabel('Time to maturity')
```

```
axs[0].set_ylabel('ARPE')
```

```
axs[0].legend()
```

```
# Subplot for y_BS
```

```
axs[1].scatter(Xt, y_BS, color='green', label='Data')
```

```
axs[1].plot(Xt, reg_t_BS.predict(Xt.reshape((-1, 1))), color='purple', label='Regression Line')
```

```
axs[1].set_title('Regression for ARPE on Time to maturity for Black- Shoulse')
```

```
axs[1].set_xlabel('Time to maturity')
```

```
axs[1].set_ylabel('ARPE')
```

```
axs[1].legend()
```

```
plt.tight_layout()
```

```
plt.show()
```

7.6 Regresing *ARPE*s for FAT model on Moneyness and Time to maturity

```
reg_mt = LinearRegression().fit(X, y_mt)
```

```
print(reg_mt.score(X, y_mt))
```

```
print(reg_mt.coef_)
```

```
print(reg_mt.intercept_)
```

7.6 Regresing *ARPE*s for Black-Shoulse model on Moneyness and Time to maturity

```

reg_BS = LinearRegression().fit(X, y_BS)

print(reg_BS.score(X, y_BS))

print(reg_BS.coef_)

print(reg_BS.intercept_)

X = np.array(X)
X1 = np.array([x[0] for x in X])
X2 = np.array([x[1] for x in X])

# Prepare grid for plotting the plane

x_range = np.linspace(X1.min(), X1.max(), num=10)

y_range = np.linspace(X2.min(), X2.max(), num=10)

x_range, y_range = np.meshgrid(x_range, y_range)

# Predict z-values across the grid

z_mt_range = reg_mt.predict(np.c_[x_range.ravel(), y_range.ravel()])

z_mt_range = z_mt_range.reshape(x_range.shape)

# Predict z-values across the grid

z_BS_range = reg_BS.predict(np.c_[x_range.ravel(), y_range.ravel()])

z_BS_range = z_BS_range.reshape(x_range.shape)

fig = plt.figure(figsize=(15, 7))

ax1 = fig.add_subplot(121, projection='3d') # first subplot

ax2 = fig.add_subplot(122, projection='3d') # second subplot

# For the first subplot
ax1.scatter(X1, X2, y_mt)
ax1.plot_surface(x_range, y_range, z_mt_range, alpha=0.5, rstride=1, cstride=1, cmap='viridis',
edgecolor='none')
ax1.set_xlabel('Moneyiness')
ax1.set_ylabel('Time to maturity')
ax1.set_zlabel('ARPE')
ax1.set_title('Regression Plane for FAT model')
# For the second subplot
ax2.scatter(X1, X2, y_BS)
ax2.plot_surface(x_range, y_range, z_BS_range, alpha=0.5, rstride=1, cstride=1, cmap='viridis',
edgecolor='none')
ax2.set_xlabel('Moneyiness')
ax2.set_ylabel('Time to maturity')

```

```
ax2.set_ylabel('ARPE')  
ax2.set_title('Regression Plane for Black-Shoulse model')  
plt.tight_layout()  
plt.show()
```

Додаток В. Зібрані дані щодо кол опціонів

Date	Last	Strike			
09.Feb	38,20	330.00		10.Feb	12,50 352.50
09.Feb	40,00	332.50		10.Feb	10,89 355.00
09.Feb	33,00	335.00		10.Feb	9,65 357.50
09.Feb	33,80	337.50		10.Feb	8,30 360.00
09.Feb	34,00	340.00		10.Feb	7,20 362.50
09.Feb	26,35	342.50		10.Feb	6,30 365.00
09.Feb	24,55	345.00		10.Feb	5,50 367.50
09.Feb	23,60	347.50		10.Feb	4,50 370.00
09.Feb	19,85	350.00		10.Feb	3,95 372.50
09.Feb	18,70	352.50		10.Feb	3,22 375.00
09.Feb	16,88	355.00		10.Feb	2,71 377.50
09.Feb	15,40	357.50		10.Feb	2,26 380.00
09.Feb	12,82	360.00		10.Feb	1,91 382.50
09.Feb	11,45	362.50		10.Feb	1,62 385.00
09.Feb	9,91	365.00		10.Feb	1,53 387.50
09.Feb	8,85	367.50		10.Feb	1,17 390.00
09.Feb	7,65	370.00		10.Feb	0,99 392.50
09.Feb	6,65	372.50		13.Feb	31,25 325.00
09.Feb	5,81	375.00		13.Feb	32,40 327.50
09.Feb	4,95	377.50		13.Feb	29,58 330.00
09.Feb	4,24	380.00		13.Feb	27,75 332.50
09.Feb	3,60	382.50		13.Feb	25,75 335.00
09.Feb	3,11	385.00		13.Feb	23,92 337.50
09.Feb	2,52	387.50		13.Feb	21,25 340.00
09.Feb	2,16	390.00		13.Feb	18,00 342.50
09.Feb	2,00	392.50		13.Feb	17,70 345.00
09.Feb	1,60	395.00		13.Feb	15,40 347.50
09.Feb	1,30	397.50		13.Feb	13,50 350.00
09.Feb	1,12	400.00		13.Feb	11,80 352.50
10.Feb	36,80	322.50		13.Feb	10,40 355.00
10.Feb	39,03	325.00		13.Feb	8,98 357.50
10.Feb	33,15	327.50		13.Feb	7,55 360.00
10.Feb	33,95	330.00		13.Feb	6,46 362.50
10.Feb	40,00	332.50		13.Feb	5,40 365.00
10.Feb	26,35	335.00		13.Feb	4,50 367.50
10.Feb	33,80	337.50		13.Feb	3,70 370.00
10.Feb	22,28	340.00		13.Feb	3,03 372.50
10.Feb	23,15	342.50		13.Feb	2,45 375.00
10.Feb	19,25	345.00		13.Feb	1,91 377.50
10.Feb	15,85	347.50		13.Feb	1,53 380.00
10.Feb	14,10	350.00		13.Feb	1,21 382.50

13.Feb	0,97	385.00	15.Feb	8,55	352.50
13.Feb	0,77	387.50	15.Feb	6,33	355.00
13.Feb	0,59	390.00	15.Feb	5,02	357.50
13.Feb	0,46	392.50	15.Feb	3,98	360.00
14.Feb	34,03	325.00	15.Feb	4,15	362.50
14.Feb	32,40	327.50	15.Feb	2,78	365.00
14.Feb	29,00	330.00	15.Feb	1,65	367.50
14.Feb	27,68	332.50	15.Feb	1,43	370.00
14.Feb	25,46	335.00	15.Feb	1,02	372.50
14.Feb	23,13	337.50	15.Feb	0,99	375.00
14.Feb	21,17	340.00	15.Feb	0,56	377.50
14.Feb	18,50	342.50	15.Feb	0,43	380.00
14.Feb	16,50	345.00	15.Feb	0,35	382.50
14.Feb	14,50	347.50	15.Feb	0,33	385.00
14.Feb	12,97	350.00	15.Feb	0,24	387.50
14.Feb	11,20	352.50	15.Feb	0,13	390.00
14.Feb	9,80	355.00	15.Feb	0,13	392.50
14.Feb	8,34	357.50	16.Feb	33,65	325.00
14.Feb	7,21	360.00	16.Feb	28,58	327.50
14.Feb	6,00	362.50	16.Feb	28,75	330.00
14.Feb	5,03	365.00	16.Feb	28,85	332.50
14.Feb	4,10	367.50	16.Feb	24,47	335.00
14.Feb	3,34	370.00	16.Feb	18,20	337.50
14.Feb	2,72	372.50	16.Feb	18,25	340.00
14.Feb	2,16	375.00	16.Feb	16,30	342.50
14.Feb	1,77	377.50	16.Feb	13,46	345.00
14.Feb	1,34	380.00	16.Feb	11,04	347.50
14.Feb	1,06	382.50	16.Feb	9,40	350.00
14.Feb	0,82	385.00	16.Feb	7,00	352.50
14.Feb	0,64	387.50	16.Feb	5,50	355.00
14.Feb	0,51	390.00	16.Feb	3,92	357.50
14.Feb	0,36	392.50	16.Feb	2,55	360.00
15.Feb	33,95	325.00	16.Feb	1,72	362.50
15.Feb	31,21	327.50	16.Feb	1,01	365.00
15.Feb	28,86	330.00	16.Feb	0,64	367.50
15.Feb	27,99	332.50	16.Feb	0,39	370.00
15.Feb	24,79	335.00	16.Feb	0,24	372.50
15.Feb	22,75	337.50	16.Feb	0,14	375.00
15.Feb	19,16	340.00	16.Feb	0,09	377.50
15.Feb	17,40	342.50	16.Feb	0,05	380.00
15.Feb	15,03	345.00	16.Feb	0,03	382.50
15.Feb	12,98	347.50	16.Feb	0,03	385.00
15.Feb	11,34	350.00	16.Feb	0,02	387.50

16.Feb	0,01	390.00	17.Feb	0,51	347.50
16.Feb	0,03	392.50	17.Feb	0,01	350.00
17.Feb	30,00	315.00	17.Feb	0,01	352.50
17.Feb	28,18	317.50	17.Feb	0,03	355.00
17.Feb	27,50	320.00	17.Feb	0,01	357.50
17.Feb	23,88	322.50	17.Feb	0,01	360.00
17.Feb	23,05	325.00	17.Feb	0,01	362.50
17.Feb	20,00	327.50	17.Feb	0,01	365.00
17.Feb	17,49	330.00	17.Feb	0,01	367.50
17.Feb	14,94	332.50	17.Feb	0,01	370.00
17.Feb	12,74	335.00	17.Feb	0,01	372.50
17.Feb	11,00	337.50	17.Feb	0,01	375.00
17.Feb	8,20	340.00	17.Feb	0,01	377.50
17.Feb	5,65	342.50	17.Feb	0,01	380.00
17.Feb	3,00	345.00	17.Feb	0,02	382.50

Додаток С. Історичні дані

Date	Last				
04/26/2023	321.15	02/27/2023	323.03	12/27/2022	284.17
04/25/2023	322.55	02/24/2023	317.15	12/23/2022	294.965
04/24/2023	329.02	02/23/2023	323.65	12/22/2022	297.75
04/21/2023	327.98	02/22/2023	334.88	12/21/2022	297.96
04/20/2023	325.35	02/21/2023	337.50	12/20/2022	288.19
04/19/2023	323.12	02/17/2023	347.96	12/19/2022	288.30
04/18/2023	333.70	02/16/2023	350.71	12/16/2022	290.71
04/17/2023	332.72	02/15/2023	361.42	12/15/2022	290.41
04/14/2023	338.63	02/14/2023	359.96	12/14/2022	317.83
04/13/2023	346.19	02/13/2023	358.57	12/13/2022	320.34
04/12/2023	331.03	02/10/2023	347.36	12/12/2022	315.18
04/11/2023	338.21	02/09/2023	362.50	12/09/2022	320.01
04/10/2023	338.99	02/08/2023	366.83	12/08/2022	310.26
04/06/2023	339.33	02/07/2023	362.95	12/07/2022	308.42
04/05/2023	342.35	02/06/2023	361.48	12/06/2022	305.56
04/04/2023	346.75	02/03/2023	365.90	12/05/2022	312.59
04/03/2023	348.28	02/02/2023	366.89	12/02/2022	320.41
03/31/2023	345.48	02/01/2023	361.99	12/01/2022	316.95
03/30/2023	338.43	01/31/2023	353.86	11/30/2022	305.53
03/29/2023	332.03	01/30/2023	353.11	11/29/2022	280.96
03/28/2023	323.52	01/27/2023	360.77	11/28/2022	281.17
03/27/2023	327.66	01/26/2023	364.87	11/25/2022	285.54
03/24/2023	328.39	01/25/2023	367.96	11/23/2022	291.50
03/23/2023	320.37	01/24/2023	363.83	11/22/2022	286.69
03/22/2023	293.90	01/23/2023	357.42	11/21/2022	285.05
03/21/2023	305.79	01/20/2023	342.50	11/18/2022	287.98
03/20/2023	305.13	01/19/2023	315.78	11/17/2022	295.28
03/17/2023	303.50	01/18/2023	326.33	11/16/2022	306.02
03/16/2023	310.06	01/17/2023	326.22	11/15/2022	310.20
03/15/2023	303.79	01/13/2023	332.82	11/14/2022	299.27
03/14/2023	294.94	01/12/2023	330.13	11/11/2022	290.13
03/13/2023	293.51	01/11/2023	327.26	11/10/2022	274.97
03/10/2023	292.76	01/10/2023	327.54	11/09/2022	254.66
03/09/2023	297.78	01/09/2023	315.17	11/08/2022	263.46
03/08/2023	311.79	01/06/2023	315.55	11/07/2022	258.60
03/07/2023	308.47	01/05/2023	309.70	11/04/2022	260.79
03/06/2023	312.03	01/04/2023	309.41	11/03/2022	269.06
03/03/2023	315.18	01/03/2023	294.95	11/02/2022	273.00
03/02/2023	311.88	12/30/2022	294.88	11/01/2022	286.75
03/01/2023	313.48	12/29/2022	291.12	10/31/2022	291.88
02/28/2023	322.13	12/28/2022	276.88	10/28/2022	295.72

10/27/2022	296.94	08/23/2022	224.55	06/16/2022	173.35
10/26/2022	298.62	08/22/2022	226.54	06/15/2022	180.11
10/25/2022	291.02	08/19/2022	241.16	06/14/2022	167.54
10/24/2022	282.45	08/18/2022	245.17	06/13/2022	169.69
10/21/2022	289.57	08/17/2022	241.15	06/10/2022	182.94
10/20/2022	268.16	08/16/2022	245.69	06/09/2022	192.77
10/19/2022	272.38	08/15/2022	249.11	06/08/2022	202.83
10/18/2022	240.86	08/12/2022	249.30	06/07/2022	198.61
10/17/2022	245.10	08/11/2022	242.70	06/06/2022	197.14
10/14/2022	230.00	08/10/2022	244.11	06/03/2022	198.98
10/13/2022	232.51	08/09/2022	229.94	06/02/2022	205.09
10/12/2022	220.87	08/08/2022	233.49	06/01/2022	192.91
10/11/2022	214.29	08/05/2022	226.78	05/31/2022	197.44
10/10/2022	229.98	08/04/2022	229.91	05/27/2022	195.19
10/07/2022	224.75	08/03/2022	226.73	05/26/2022	191.40
10/06/2022	240.02	08/02/2022	221.42	05/25/2022	187.83
10/05/2022	236.73	08/01/2022	226.21	05/24/2022	180.34
10/04/2022	240.74	07/29/2022	224.90	05/23/2022	187.44
10/03/2022	239.04	07/28/2022	226.02	05/20/2022	186.35
09/30/2022	235.44	07/27/2022	226.75	05/19/2022	183.48
09/29/2022	239.71	07/26/2022	213.91	05/18/2022	177.19
09/28/2022	245.20	07/25/2022	218.51	05/17/2022	190.56
09/27/2022	224.36	07/22/2022	220.44	05/16/2022	186.51
09/26/2022	224.07	07/21/2022	223.88	05/13/2022	187.64
09/23/2022	226.41	07/20/2022	216.44	05/12/2022	174.31
09/22/2022	237.05	07/19/2022	201.63	05/11/2022	166.37
09/21/2022	236.87	07/18/2022	190.92	05/10/2022	177.66
09/20/2022	242.85	07/15/2022	189.11	05/09/2022	173.10
09/19/2022	243.63	07/14/2022	174.78	05/06/2022	180.97
09/16/2022	240.13	07/13/2022	176.56	05/05/2022	188.32
09/15/2022	235.38	07/12/2022	174.45	05/04/2022	204.01
09/14/2022	224.12	07/11/2022	177.34	05/03/2022	199.87
09/13/2022	218.13	07/08/2022	186.975	05/02/2022	199.46
09/12/2022	236.53	07/07/2022	189.27	04/29/2022	190.36
09/09/2022	233.57	07/06/2022	184.06	04/28/2022	199.52
09/08/2022	227.44	07/05/2022	185.88	04/27/2022	188.54
09/07/2022	228.96	07/01/2022	179.95		
09/06/2022	218.39	06/30/2022	174.87		
09/02/2022	226.11	06/29/2022	178.36		
09/01/2022	230.04	06/28/2022	179.60		
08/31/2022	223.56	06/27/2022	189.14		
08/30/2022	220.65	06/24/2022	190.85		
08/29/2022	224.57	06/23/2022	181.71		
08/26/2022	223.28	06/22/2022	178.89		
08/25/2022	233.98	06/21/2022	170.91		
08/24/2022	229.61	06/17/2022	175.51		

