**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ**

**«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

###### Центр непрерывного образования

###### Факультета компьютерных наук

**ИТОГОВЫЙ ПРОЕКТ**

Анализ данных о фондовом рынке акций и

разработка моделей для прогнозирования цены акции \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Название темы

|  |  |
| --- | --- |
|  | Выполнил (а): |
|  |  |
|  | Александрова Юлия Игоревна  Ф.И.О. |
|  | Руководитель: |
|  | Паточенко Евгений Анатольевич  Ф.И.О. |

Москва 2025

**Оглавление**

[I. Введение](#_30j0zll)

[II. Проведение разведочного анализа данных (Exploratory Data Analysis – EDA)](#_1fob9te)

[III. Методы обнаружения аномалий и оценка качества алгоритмов](#_3znysh7)

[IV. Эксперименты](#_2et92p0)

[V. Заключение](#_tyjcwt)

[Приложение](#_3dy6vkm)

[Список литературы:](#_1t3h5sf)

1. **Введение**

Машинное обучение – это область искусственного интеллекта, которая занимается разработкой алгоритмов и моделей, позволяющих компьютерам, обучаясь на основе данных и извлекая закономерности из данных, делать прогнозы и принимать решения, повышая производительность прогностических моделей с течением времени.

Сегодня машинное обучение – не только предмет исследований в области компьютерных наук, оно играет важную роль во многих сферах экономики, а также в нашей повседневной жизни. Машинное обучение применяют во многих сферах, где есть цифровые данные: торговля и маркетинг, производство, транспорт и логистика, кибербезопасность, наука и исследования, сельское хозяйство, образование. Машинное обучение активно используется в области медицины, биологических и фармацевтических исследованиях, а также в исследованиях проблем изменения климата.

В данном итоговом проекте будет применен тип машинного обучения с учителем на примере задачи прогнозирования цены акции. Набор данных для разработки модели взят с сайта [www.kaggle.com](http://www.kaggle.com) и содержит исторические данные о фондовом рынке, извлеченные из Yahoo Finance. Набор данных включает следующие столбцы:

«Date» дата, соответствующая зафиксированным данным фондового рынка;

«Open» цена открытия торгов акциями на указанную дату;

«High» максимальная цена акции, достигнутая в течение торгового дня;

«Low» минимальная цена акции, зафиксированная в течение торгового дня;

«Close» цена закрытия торгов акциями на определенную дату;

«Volume» объем акций, проданных в указанную дату;

«Dividends» дивидендные выплаты, произведенные компанией на эту дату;

«Stock Splits» информация о дроблении акций, произошедшем в указанную дату;

«Company» краткое буквенное обозначение компании.

Набор данных содержит 602 962 строки и 9 столбцов.

Основная цель обучения с учителем – обучить на данных модель, которая позволит делать прогнозы относительно незнакомых или будущих данных. Определение «с учителем» относится к набору обучающих примеров (входных данных), для которых желаемые выходные сигналы (метки) уже известны. Обучение с учителем представляет собой процесс моделирования взаимосвязи между входными данными и метками. Если модель может выдавать точные прогнозы на ранее не встречавшихся данных, значит она обладает способностью обобщать результат на тестовые данные.

Максимальная обобщающая способность модели заключается в компромиссе между переобучением и недообучением. Переобучение происходит, когда модель слишком точно подстраивается под особенности обучающего набора данных, в результате чего модель хорошо работает на обучающем наборе данных, но не умеет обобщать результат на новые данные. С другой стороны, если модель слишком проста, то она будет плохо работать даже на обучающем наборе данных, что будет считаться недообучением. Задача итогового проекта: исследовать данные, а также построить модель для прогнозирования цены акции, которая будет обладать максимальной обобщающей способностью.

1. **Проведение разведочного анализа данных**

**(Exploratory Data Analysis – EDA)**

В начале работы настраиваем проверку кода линтерами. Для этого необходимо установить pycodestyle. Утилита pycodestyle проверяет код на Python на соответствие стандартам PEP 8.

Разведочный анализ данных – это исследование данных с помощью статистических и визуальных методов для выявления основных характеристик, закономерностей, аномалий в данных, а также преобразование данных для последующего их использования в разработке прогностических моделей.

Формат представления кода: **Jupyter Notebook (.ipynb)**.

Импортируем библиотеки для работы с массивами данных, обработки, анализа структурированных табличных данных, выполнения математических операций и визуализации данных:

**import numpy as np**

**import pandas as pd**

**import matplotlib.pyplot as plt**

**import seaborn as sns**

Загружаем данные из csv-файла. Прописывается путь, где размещен файл с данными, затем с помощью функции **pd.read\_csv()** выполняется чтение данных из csv-файла.

С помощью метода **info()** выводим краткую оценку структуры и качества набора данных: количество строк и столбцов, названия и тип данных столбцов, количество ненулевых значений в столбцах, общее использование памяти. Данный метод удобен тем, что собирает в единое целое краткую информацию о наборе данных.

С помощью метода **describe()** выводим статистические характеристики числовых столбцов: количество заполненных строк в каждом столбце; среднее значение; стандартное отклонение, показывающее разброс значений; минимальное и максимальное значения; значения по процентилям, показывающие распределение значений в выборке.

В наборе данных содержится информация по акциям 491 компании. Для вывода всех компаний и подсчета их количества используем методы **unique()** и **nunique()**.

Преобразовав с помощью **pivot** строки с указанием компаний в столбцы, выводим цену закрытия торгов по датам в виде нового датасета data\_Close. Такое преобразование позволяет понять, что по некоторым компаниям данные имеются не по всем датам. Методом **isnull().sum()** подсчитываем количество пропущенных значений. Затем полученную информацию преобразуем в DataFrame. Методом **reset\_index()** преобразуем индекс в столбец. С помощью функции **rename()** переименовываем столбец с количеством пропущенных значений. Таким образом, выводим перечень компаний с пропущенными значениями. Пропущенные значения заменяем на «серединное» значение упорядоченного ряда чисел – медиану, используя функцию **fillna()**.

Используя датасет data\_Close подсчитываем функцией **mean()** среднюю цену закрытия торгов по компаниям за весь период наблюдения и преобразовываем полученные данные в DataFrame.

Имеется существенный разброс в данных. Построим диаграмму рассеяния значений средней цены закрытия торгов по компаниям за весь период наблюдения. Используем функцию **plt.scatter()**. В данных есть несколько значений, существенно отклоняющихся от среднего значения. Ориентируясь по графику, выводим перечень компаний, по которым наблюдается существенное отклонение от среднего значения в большую сторону.

Столбец «Date», содержащий информацию о датах, имеет тип данных object. Для дальнейшего исследования данных необходимо его преобразование. Сначала с помощью метода **astype(str)** преобразуем в строковый тип данных, затем с помощью метода **str.slice()** извлекаем часть строки: год, месяц, день, указанные через дефис. Далее методом **to\_datetime()** преобразуем данные в формат datetime для дальнейшего выделения в отдельные столбцы года, месяца, дня с помощью атрибутов **dt.year**, **dt.month**, **dt.day**.

С помощью возможностей библиотеки Matplotlib визуализируем динамику изменения цены закрытия торгов за весь период наблюдения по выбранной компании, а также одновременно по трем компаниям для сравнительного анализа. Смотреть на графике динамику одновременно всех компаний неудобно по причине количества компаний, поэтому можно посмотреть динамику по одной или трем компаниям. Информацию с кратким буквенным обозначением компании необходимо ввести в поле для ввода данных Input.

Удаляем ранее преобразованный столбец «Date» функцией **drop()**.

Выводим среднюю по годам цену закрытия торгов по компаниям. Находим среднее значение столбца «Close» для каждой комбинации значений столбцов «Company» и «Year»: **groupby(["Company", "Year"])["Close"].mean()**. Затем полученные результаты преобразуем в DataFrame и выводим динамику изменения средней по годам цены закрытия торгов за весь период наблюдений по всем компаниям.

Большинство алгоритмов машинного обучения используют числовые типы данных. Столбец «Company» имеет тип данных object. Данные не имеют естественного порядка, поэтому для преобразования категориальных переменных в данном случае используем метод One Hot Encoding функцию pd.get\_dummies().

Создаем матрицу корреляции и визуализируем результат в виде тепловой карты. По результатам видно, что максимальная корреляция присутствует между ценой открытия торгов, ценой закрытия торгов, а также колебаниями в течение дня.

Разделяем данные на целевую переменную и признаки. Целевой переменной назначается столбец «Close» с информацией о цене закрытия торгов. Все остальные столбцы – признаки.

Импортируем функцию **train\_test\_split** из подмодуля sklearn.model\_selection библиотеки Scikit-learn. Разделяем набор данных на два подмножества: для обучения и для тестирования. Разделение данных на тренировочные и тестовые помогает избежать переобучения и обеспечивает объективную оценку производительности модели. В данном случае размер тестовой выборки составляет 20%, то есть 20% данных будет использовано для тестирования, 80% данных – для обучающей выборки.

Импортируем класс **StandartScaler** из модуля preprocessing библиотеки Scikit-learn. StandartScaler масштабирует признаки, то есть приводит данные к стандартному нормальному распределению с центром в нуле и стандартным отклонением, равным единице. Масштабируются только признаки, целевую переменную не масштабируем, поскольку масштабирование может изменить ее оригинальное значение, что приведет к неправильным прогнозам. StandartScaler используем, поскольку признаки имеют разный порядок чисел, не применение масштабирования может сказаться на качестве модели.

Выводы:

В процессе разведочного анализа данных удалось исследовать состав и структуру набора данных. Выявлены компании, по которым данные собраны не по всем датам. Выявлены компании с наиболее высокой ценой закрытия торгов.

Визуализация динамики изменения цены закрытия торгов показывает тренд, наличие спадов и подъемов.

Для дальнейшего использования данных с целью разработки моделей прогнозирования преобразованы столбцы с датой и кратким буквенным обозначением компании.

В завершение подготовки данных для разработки моделей прогнозирования данные были разделены на обучающий и тестовый наборы данных, а также проведено масштабирование данных, то есть приведение данных к стандартному нормальному распределению.

**Приложение**

**Список литературы**

При подготовке данного итогового проекта использовались следующие источники информации:

* Машинное обучение с PyTorch и Scikit-Learn Себастьян Рашка, Юси (Хэйден) Лю, Вахид Мирджалили. Астана: Фолиант, 2024;
* Введение в машинное обучение с помощью Python. Руководство для специалистов по работе с данными Андреас Мюллер, Сара Гвидо. Москва 2016-2017;
* Грокаем машинное обучение Серрано Луис. Санкт-Петербург: Питер, 2024;
* Сайт [www.kaggle.com](http://www.kaggle.com), источник данных Yahoo Finance. Набор данных был собран с помощью Yahoo Finance API или методом парсинга данных для получения ежедневной информации о котировках акций указанных компаний.