Федеральное государственное бюджетное образовательное учреждение высшего образования

**«ФИНАНСОВЫЙ УНИВЕРСИТЕТ ПРИ ПРАВИТЕЛЬСТВЕ**

**РОССИЙСКОЙ ФЕДЕРАЦИИ»**

**Департамент анализа данных и машинного обучения**

**Пояснительная записка к курсовой работе по дисциплине:**

**«Технологии анализа данных и машинного обучения»**

на тему:

**«Машинное обучение в задачах оценки активов»**

Выполнил:

Студент группы ПМ19-2

Путилов Дмитрий Сергеевич

\_\_\_\_\_\_\_\_\_\_\_

Научный руководитель:

доцент, к.т.н.

Болтачев Э. Ф.

\_\_\_\_\_\_\_\_\_\_\_

**Москва**

**2022**

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 2](#_Toc122436168)

[1 Теоретическая часть 3](#_Toc122436169)

[1.1 LogisticRegression 3](#_Toc122436170)

[1.2 KNeighborsClassifier 3](#_Toc122436171)

[1.3 Дерево решений 3](#_Toc122436172)

[1.4 MLPClassifier 4](#_Toc122436173)

[1.5 SVC 4](#_Toc122436174)

[2 Практическая часть 5](#_Toc122436175)

[2.1 Примеры реализаций 5](#_Toc122436176)

[2.2 Личная реализация 6](#_Toc122436177)

[2.2.1 Анализ и подготовка данных 6](#_Toc122436178)

[2.2.2 Базовая модель 9](#_Toc122436179)

[2.2.3 Улучшение модели 12](#_Toc122436180)

[2.2.4 Запуск моделей 13](#_Toc122436181)

[ЗАКЛЮЧЕНИЕ 14](#_Toc122436182)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 15](#_Toc122436183)

[Приложение ИСХОДНЫЙХОД 16](#_Toc122436184)

# ВВЕДЕНИЕ

В ходе выполнения данной курсовой работы будет создан инструмент по предсказанию роста цены акции, основанный на методах машинного обучения с учителем, а также на методах и приемах подготовительного и описательного анализа данных.

В процессе работы над инструментом будет использован язык Python, который является одним из самых популярных языков программирования. Средой разработки инструмента выбрана Jupyter Notebook.

Помимо разработки инструмента данная курсовая работа развивает навык составления технической документации, а именно пояснительной записки. Составленная пояснительная записка должна отвечать заданным требованиям, которые представляют собой не только определенное содержание, но также полноту и техническую корректность. Важно отметить, что грамотное составление технической документации является для программиста не менее важным навыком, чем разработка.

# Теоретическая часть

## LogisticRegression

Логистическая функция, также называемая сигмовидной функцией, была разработана статистиками для описания свойств роста населения в экологии, быстро растущих и максимизирующими пропускную способность окружающей среды. Это S-образная кривая, которая может взять любое вещественное число и сопоставить его в значение от 0 до 1.

## KNeighborsClassifier

Алгоритм классификации, реализующий голосование k-ближайших соседей. Количество соседей является основным решающим фактором.

Алгоритм имеет следующие шаги:

1. Рассчитать расстояние
2. Найти ближайших соседей
3. Провести голосование

Существуют такие меры расстояния: евклидово расстояние, расстояние Хэмминга, расстояние Манхэттена и расстояние Минковского.

## Дерево решений

В работе применяются два алгоритма DecisionTreeClassifier и RandomForestClassifier.

Дерево решений – модель, которая разделяет входные данные по их атрибутам, тем самым образуя дерево различных решений.

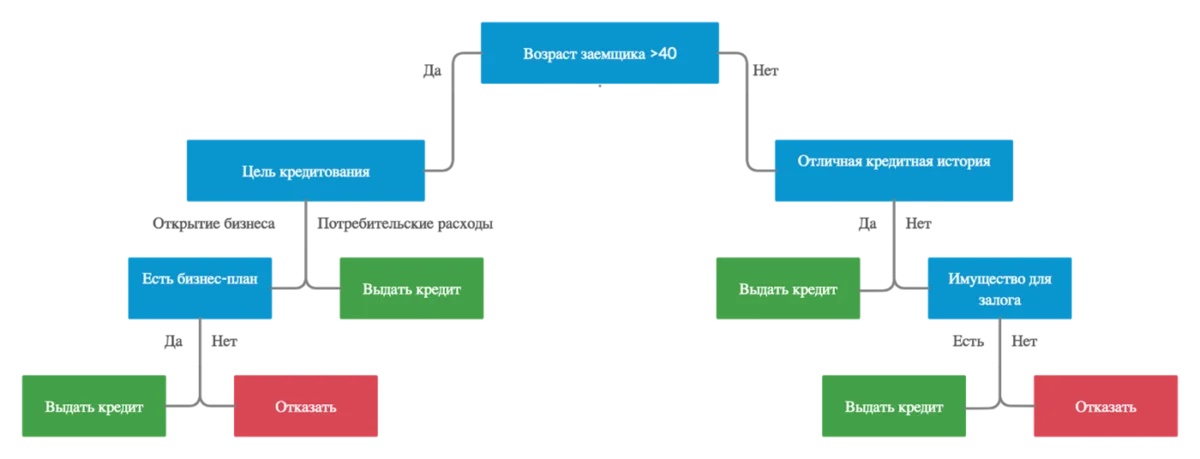


Рисунок 1.1 – Дерево решений на примере кредитного скоринга

Датасет делится на подмножества заданное количество раз, при этом модель самостоятельно выбирает атрибут, являющийся критерием разделения на том или ином шаге дерева (возраст заемщика, цель кредитования и т.д).

## MLPClassifier

Название расшифровывается как многослойный классификатор персептронов, то есть само название имеет отсылку к нейронной сети. В отличие от других алгоритмов классификации, MLPClassifier полагается на базовую нейронную сеть для выполнения задачи классификации.

## SVC

Метод опорных векторов с гауссовым и линейным ядром – метод машинного обучения с учителем, который может применяться как в классификации, так и в регрессии. Метод позволяет определить границу принятия решения вместе с максимальным зазором, который разделяет все точки датасета на два класса.

# Практическая часть

## Примеры реализаций

Основные примеры реализаций предсказания цены акции основаны на предсказании именно цены акции (регрессия), а не факта роста или падения (классификация), что значительно отличается от разрабатываемого мною решения.

Ниже я представлю визуализацию результатов использования алгоритмов нейронных сетей

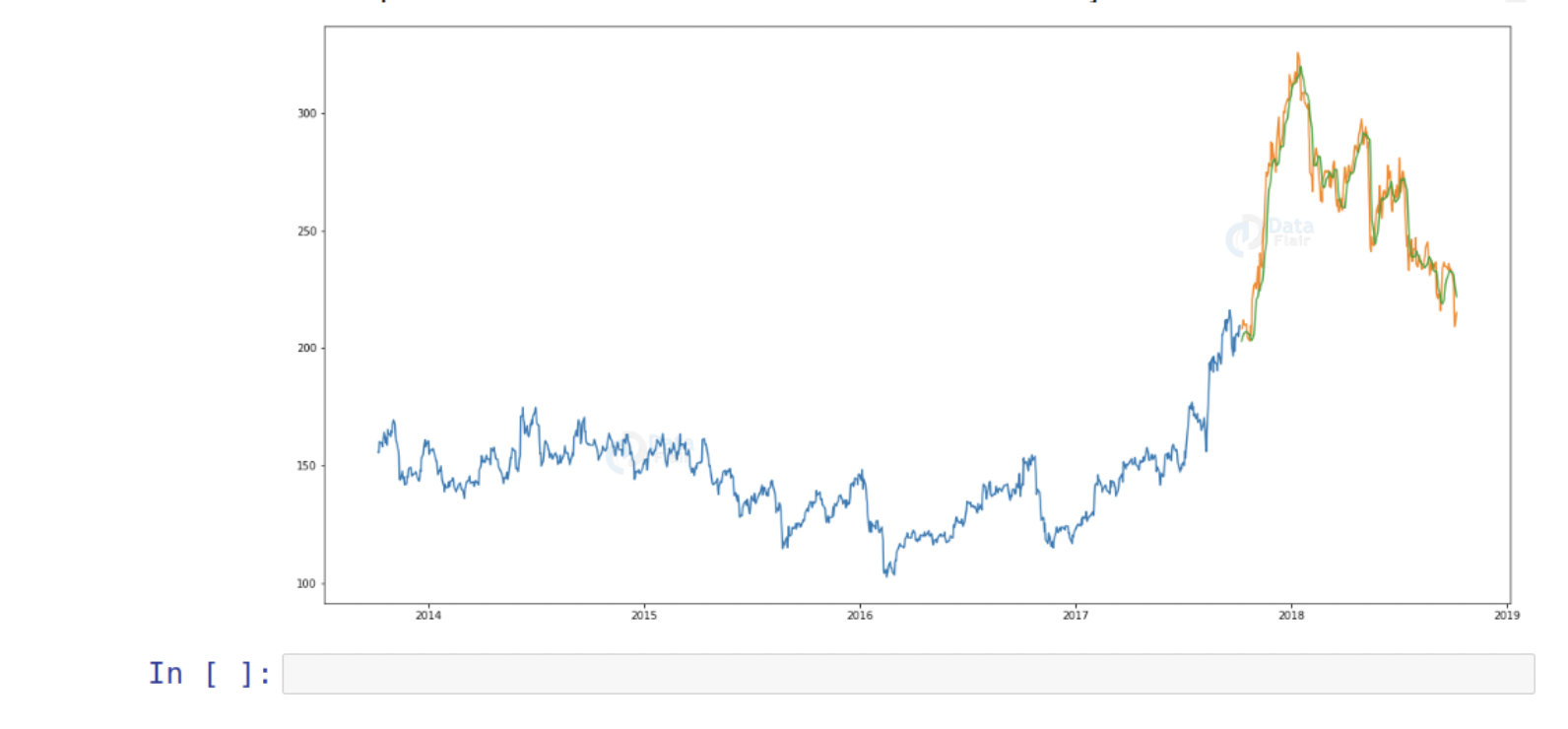


Рисунок 2.1 – LSTM neural network



Рисунок 2.2 – LSTM neural network 2

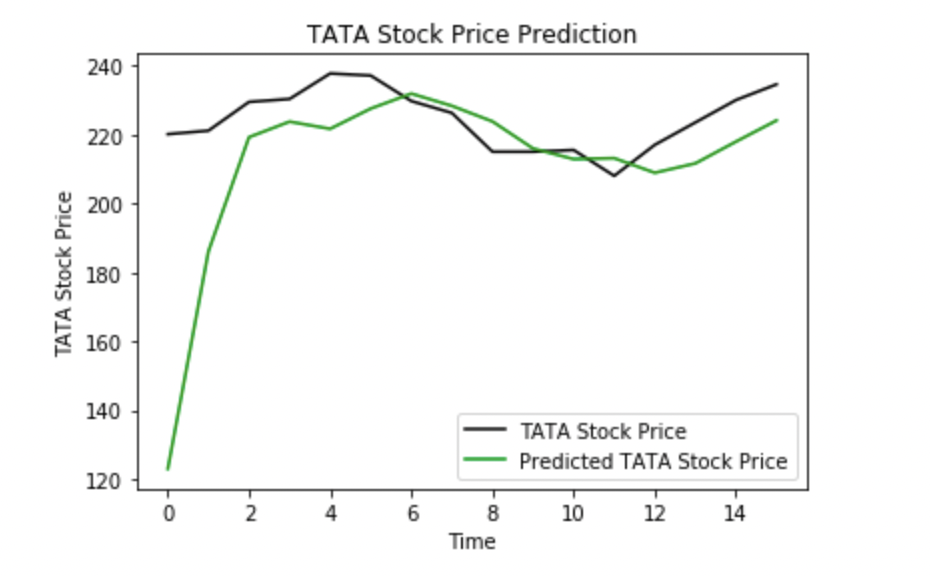


Рисунок 2.3 – LSTM neural network 3

## Личная реализация

### Анализ и подготовка данных

Цель моей курсовой работы – основываясь на исторических данных котировок акций уметь предсказывать рост или падение цены акции в разрезе 1 дня. То есть я решаю задачу классификации, где 1 – рост акции, 0 – падение акции.

Первая задача, которую я решил в своей курсовой работе – выбор набора данных и его анализ. С помощью библиотеки yfinance я могу получить котировки любой компании, акции которой торгуются на американской бирже NASDAQ. Я решил выбрать Microsoft – компанию с многолетней историей, которая является одним из мировых лидеров в области информационных технологий. С помощью библиотеки yfinance я загрузил данные за весь период торгов акций компании. Шаг датасата по таймлайну – 1 день.

Изображение выглядит как текст, стол

Автоматически созданное описание

Рисунок 2.4 – Загрузка датасета, выборка датасета

Рассмотрим значения атрибутов датасета

* Date – дата
* Open – цена акции на момент открытия биржи
* High – максимальная цена акции в течение торгового дня
* Low – минимальная цена акции в течение торгового дня
* Close - цена акции на момент закрытия биржи
* Volume – объем торгов
* Dividends – дивиденды по акции
* Stock splits – дробление акций

В своей работе я буду использовать все атрибуты кроме Dividends и Stock splits. Важно также отметить, что торги проводятся не каждый день, так как во время выходных и праздников биржа не работает. Это означает, что данные в колонке Date могут отличаться более чем на 1 день.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 2.5 – Информация о датасете

Также при анализе данных я построил график цены акции. Можно отметить, что начиная с 1986 цена акции росла достаточно медленно, однако примерно в 2015 начался крайне бурный рост. Однако в конце 2021 года началось стремительное падение, которое было связано с ухудшением ситуации на мировом рынке.

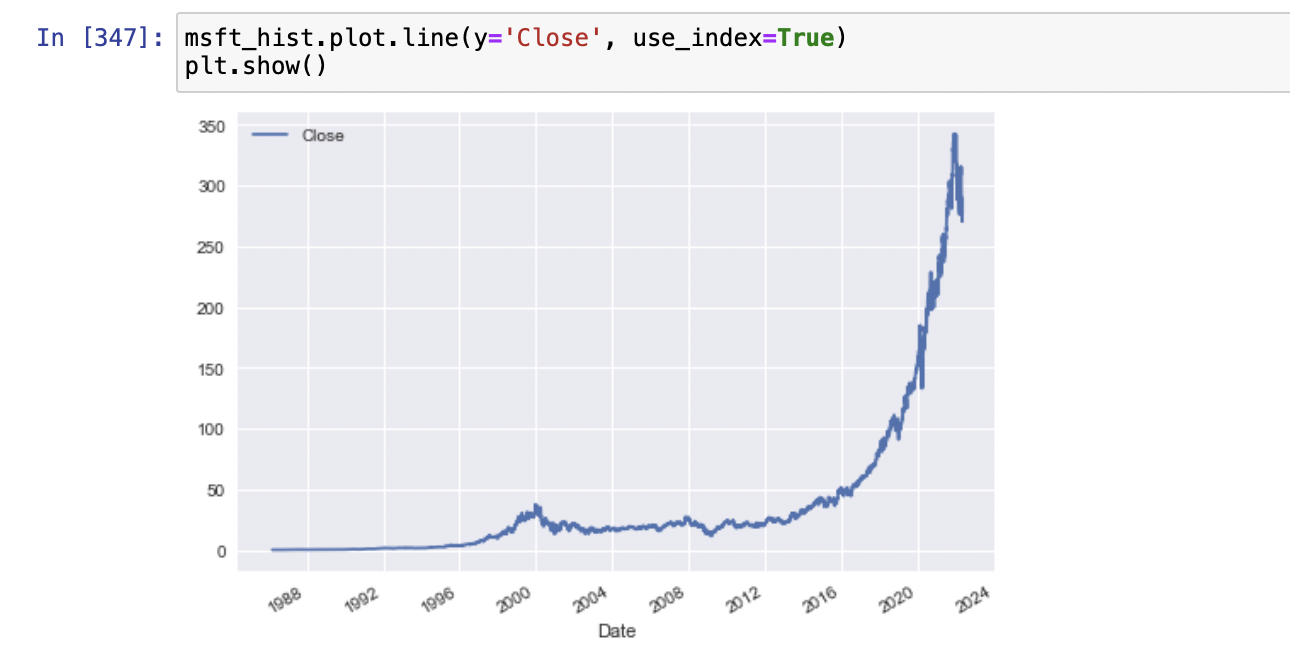


Рисунок 2.6 – График цены акции

После первичного анализа данных я приступил к подготовке данных к использованию. Как я писал выше, таргетом моей модели будет значение 1 или 0, где 1 – рост акции, 0 – падение акции. Важным моментом в этапе подготовки данных будет перемещение данных вперед на 1 день. Это будет сделано для того, чтобы не использовать данные за текущий день, чтобы делать на него прогноз.

Сперва я копирую данные столбца Close в рабочий датасет data, и переименовываю его в Actual\_Close, для того чтобы знать, какая реальная цена была в тот или иной день.

Далее я создаю колонку Target – которую и собираюсь предсказывать. Для этого я использую метод rolling, который агрегирует данные по текущей строке и по n-1 предыдущим строкам (n указывается как параметр метода, в моем случае n=2). Затем я применяю метод apply и lambda-функцию, с помощью которых создаю таргет (если значение предыдущего дня строго меньше значения текущего дня, то таргет равен 1).

После этого с помощью метода shift я сдвигаю все первоначальные данные на 1 день вперед.

Затем с помощью метода join я создаю итоговый датасет data, в котором есть колонки:

* Actual\_Close – реальная цена акции на момент закрытия биржи
* Target – бинарное значение, отвечающее за рост или падение акции
* Close, Volume, Open, High, Low – атрибуты, значения которых сдвинуты на 1 день вперед

### Базовая модель

Далее я решил создать базовую модель, чтобы протестировать ее, оценить результаты и продумать логику своей основной модели. Важно отметить, что из-за того, что данные представляют собой временной ряд, я не могу использовать кросс-валидацию данных. Иначе я бы использовал данные из будущего для предсказания прошлого, что было бы просто некорректным. Вместо этого я в дальнейшем буду несколько раз делить данные на тест и трейн, однако в базовой модели я выделю все данные, кроме последних 100 строк для трейна, а оставшиеся 100 строк будут тестовыми данными.

Для базовой модели я выбрала алгоритм RandomForestClassifier. Данный алгоритм является представителем дерева решений – модели, принципом которой является разделение входных данных при помощи ветвления по их признакам.

Далее я провел обучение модели. В качестве предикатов были выбраны столбцы Close, Volume, Open, High, Low. В качестве таргета столбец Target.

Теперь мне необходимо оценить, насколько хорошо моя модель предсказывает данные. Для этого я выбрал метрику precision, которая оценивает какую долю из всех предсказанных значений я предсказал верно. Я считаю, что для выбранной мной цели использования модели машинного обучения эта метрика является самой подходящей. В предсказании цены акции лучше всего минимизировать FP, чем минимизировать FN.

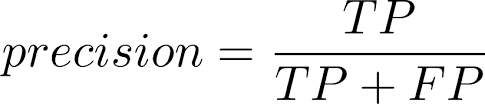


Рисунок 2.7 – Метрика precision

Для моей базовой модели метрика precision получилась 0.489, что меньше, чем если бы я принимал решение подбрасывая монетку. Думаю, что такое значение метрики вызвано рядом причин:

* для тестирования я беру всего 100 последних дней, что мало и в плане количества, и в то же время не совсем корректно с точки зрения анализа временных рядов
* в последние 100 дней по сравнению с жизнью всей акции происходит достаточно аномальная активность, а именно падение

Для более глубоко понимания поведения модели я хочу посмотреть на график значений таргета и предсказанного моделью значения (рис. 2.16). Как можно увидеть на графике в большинстве случаев модель предпочитает предсказывать рост акции, и лишь иногда она предсказывает падение цены акции.

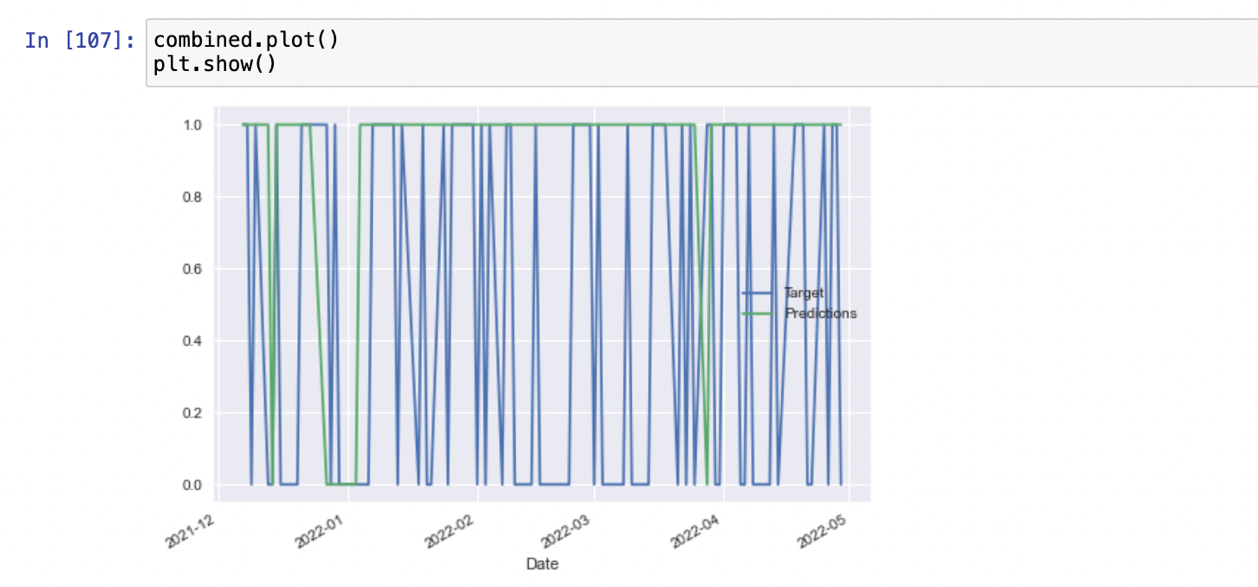


Рисунок 2.8 – График таргета и предсказанного значения

Также я построил визуализацию одного из деревьев решений, чтобы понимать, какие предикаты влияют на поведение модели.

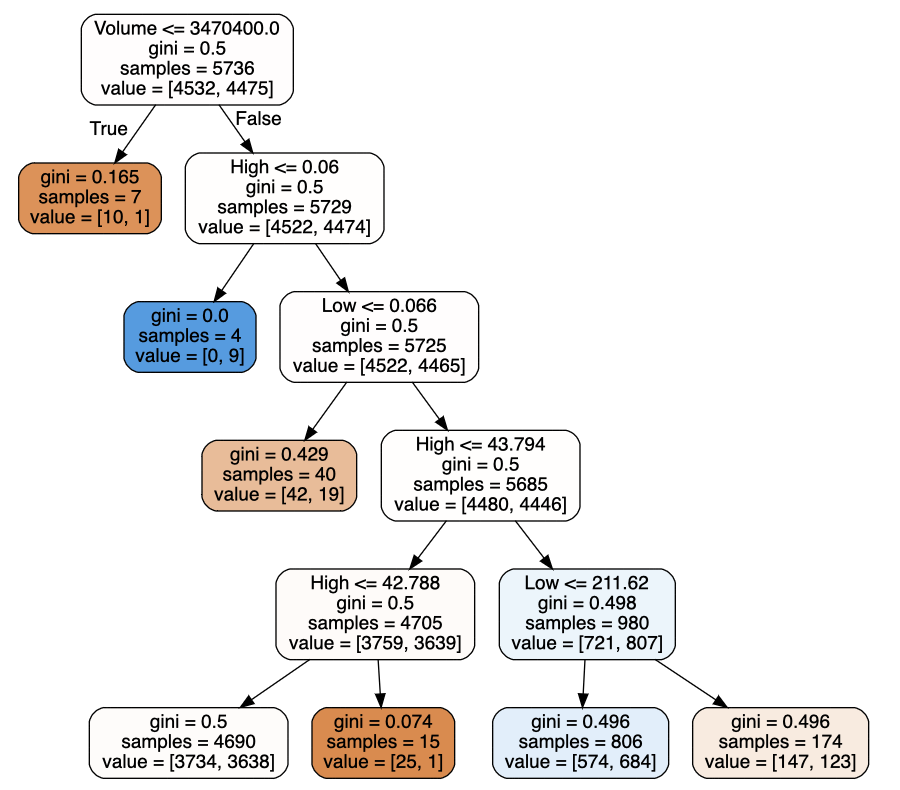


Рисунок 2.9 – Дерево решений

Проблему низкой метрики я собираюсь решить за счет улучшения модели по нескольким показателям:

* увеличение количества предикатов
* нормализация данных
* изменение подхода в разделении данных на тест и трейн.

### Улучшение модели

В первую очередь я хочу нормализовать свои данные. Для этого я написал функцию, основой которой является метод минимакс.

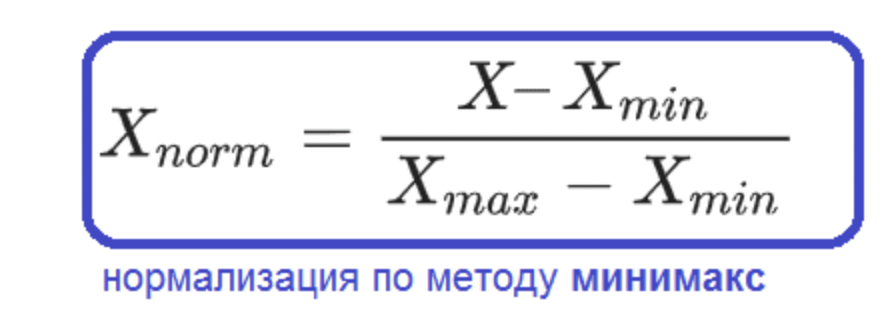
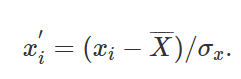


Рисунок 2.10 – Формула нормализации

Также существуют и другие способы нормализации данных, к примеру, Нормализация средним (Z-нормализация). Так, недостаткам минимаксной нормализации является наличие аномальных значений данных, которые «растягивают» диапазон, что приводит к тому, что нормализованные значения опять же концентрируются в некотором узком диапазоне вблизи нуля. Чтобы избежать этого, следует определять диапазон не с помощью максимальных и минимальных значений, а с помощью «типичных» — среднего и дисперсии:



Величины, полученные по данной формуле, в статистике называют Z-оценками. Их Абсолютное значение представляет собой оценку (в единицах стандартного отклонения) расстояния между x и его средним значением в общей совокупности.

И еще один метод для нормализации данных – Отношение.   
В этом методе каждое значение исходных данных делиться на некоторое, заданное пользователем число, или на значение статистического показателя, вычисленного по набору данных, например, среднее, стандартное отклонение, дисперсию, вариационный размах и др.

Также я написал функцию, которая собирает все манипуляции с данными, которые применялись на прошлых шагах.

После этого я задумался над добавлением новых предикатов, которые бы основывались на имеющихся данных. Я буду добавлять следующие значения:

* Среднее значение цены закрытия за неделю
* Среднее значение цены закрытия за месяц
* Среднее значение цены закрытия за квартал
* Среднее значение цены закрытия за год
* Тренд роста или падения цены за неделю
* Тренд роста или падения цены за месяц
* Отношение цены открытия к цене закрытия
* Отношение максимальной цены за день к цене закрытия
* Отношение минимальной цены за день к цене закрытия

Все эти новые предикаты я получаю на основе имеющихся данных, поэтому я завернул этот процесс в функцию rolling\_means.

После этого мне понадобилась функция, которая вмещала бы в себя все предыдущие шаги обработки данных, которые должны производиться в следующем порядке:

1. Формирование первоначального датасета со сдвигом предикатов
2. Нормализация данных
3. Создание новых предикатов на основе имеющихся
4. Очистка данных от NaN

Все эти шаги были реализованы в отдельных функциях, поэтому в функции final\_data мне требовалось просто собрать все это вместе.

После этого я хочу улучшить модель, обучая ее не только на последних 100 строках. Это даст мне более надежную оценку ошибки. Я буду использовать метод обратного тестирования, который гарантирует, что я использую данные только до того дня, когда я прогнозирую. Мой алгоритм обучается на start строках, а затем тестируется на последующих step строках.

Также с помощью параметра use\_predict\_proba я могу увеличивать порог уверенности предсказания роста цены до 0,65.

### Запуск моделей

Для тестирования я выбрал еще несколько алгоритмов:

* Логистическая регрессия
* Метод k ближайших соседей
* Классификатор случайный лес
* Классификатор дерева решений
* Многослойный перцептрон
* Метод опорных векторов с гауссовым ядром
* Метод опорных векторов с линейным ядром

После запуска всех моделей и получения метрик лучше всего себя показал RandomForestClassifier, presicion = 0,589.

# 

# ЗАКЛЮЧЕНИЕ

В ходе выполнения данной курсовой работы был создан инструмент по предсказанию роста цены акции, основанный на методах машинного обучения с учителем, а также на методах и приемах подготовительного и описательного анализа данных. Инструмент не идеален, и открыт к дальнейшим усовершенствованиям

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Devi M. K. N., Bhaskaran V. M. Impact of social media sentiments and economic indicators in stock market prediction //International Journal of Computer Science & Engineering Technology. – 2015. – Т. 6. – №. 4.
2. Kumbure M. M. et al. Machine learning techniques and data for stock market forecasting: a literature review //Expert Systems with Applications. – 2022. – С. 116659.
3. Jiang W. Applications of deep learning in stock market prediction: recent progress //Expert Systems with Applications. – 2021. – Т. 184. – С. 115537.
4. Géron A. Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow. – " O'Reilly Media, Inc.", 2022.
5. Chauhan V. K., Dahiya K., Sharma A. Problem formulations and solvers in linear SVM: a review //Artificial Intelligence Review. – 2019. – Т. 52. – №. 2. – С. 803-855.
6. Yu P., Yan X. Stock price prediction based on deep neural networks //Neural Computing and Applications. – 2020. – Т. 32. – №. 6. – С. 1609-1628.

# Приложение ИСХОДНЫЙ КОД

Ссылка на гихаб https://github.com/DimonSP0?tab=repositories