**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

Федеральное государственное бюджетное образовательное учреждение высшего образования

**«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Кафедра «Измерительно-вычислительные комплексы»

**Отчёт по лабораторной работе №2**

**по дисциплине «Методы и технологии интеллектуальной обработки и анализа данных»**

**Тема «Исследование методов классификации»**

**Выполнила:**

студентка группы ИСТмд-11

Фролова М.В.

**Проверил:**

Доцент каф. «Измерительно-

вычислительные комплексы»

к.т.н.

Шишкин В.В.

Ульяновск, 2025

**Формулировка проблемы:**

Одной из ключевых задач в машинном обучении является задача классификации данных. Её целью является отнесение объекта к одной из заранее известных категорий (классов) на основе построенного прогноза. Для решения задачи используются различные методы классификации (линейные, ансамблевые, деревья решения и другие).

Однако универсального классификатора, который одинаково эффективно работал бы на разных наборах данных, не существует. По этой причине, поиск подходящего классификатора, который позволил бы достигнуть наилучшего результата в обучении модели для конкретного датасета, становится проблемой.

**Гипотеза:**

Эффективность методов классификации напрямую зависит от характеристик конкретного набора данных. В частности, от объёма выборки, количества и качества признаков, количества и сбалансированности классов. Таким образом, проанализировав данные, можно выявить наиболее подходящие алгоритмы для классификации.

**План исследования:**

1. Подготовить набор данных для исследования, определить целевую переменную и признаки.

2. Выбрать несколько разного рода классификаторов для сравнения.

3. Определить связь признаков с целевой переменной.

4. Провести тестирование, применив выбранные классификаторы для обучения моделей.

5. Оценить результаты обучения, сделав сравнительный анализ метрик качества классификаторов.

6. Сделать выводы по результатам исследования.

**План эксперимента:**

Датасет для исследования: файл ‘Coffee\_sales.csv’, размер 3547х11.

Целевая переменная: ‘coffee\_name’ (разновидность кофе).

Признаки: ‘hour\_of\_day’ (время дня), ‘money’ (стоимость), ‘Weekdaysort’ (день недели), ‘Monthsort’ (месяц).

Исследуемые классификаторы:

RandomForestClassifier() => Метод «Случайный лес»,

KNeighborsClassifier() => Метод «k-ближайших соседей»,

DecisionTreeClassifier() => Метод «Дерево решений»,

AdaBoostClassifier() => Метод «Адаптивный бустинг»,

GaussianNB() => Метод «Наивный Байес».

Используемые библиотеки: – import pandas as pd – import matplotlib.pyplot as plt – import sklearn (модули .preprocessing, .model\_selection, .metrics, .neighbors, .tree, .ensemble, .naive\_bayes)

Для оценки связи признаков с целевой переменной была построена диаграмма корреляции.

Для анализа эффективности классификаторов было проведено троекратное тестирование и сравнение метрик качества. Размер тестовой выборки – 30%.

Метрики для сравнения:

Accuracy – точность, представляющая собой долю правильных предсказаний среди всех предсказаний модели. Рассчитывается как:

(TP + TN) / (TP + FP + FN + TN).

Precision – точность положительных предсказаний, то есть доля объектов, которые действительно принадлежат предсказанному моделью классу. Рассчитывается как:

TP / (TP + FP).

Recall – полнота, представляющая собой долю объектов класса, которые модель смогла правильно отнести к этому классу. Рассчитывается как:

TP / (TP + FN).

F1-Score – гармоническое среднее между Precision и Recall, позволяющее сбалансировать их значения. Рассчитывается как:

2 \* (Precision \* Recall) / (Precision + Recall).

Где исходя из матрицы ошибок:

TP – истинно положительное решение;

TN – истинно отрицательное решение;

FP – ложно положительное решение;

FN – ложно отрицательное решение.

**Реализация эксперимента:**

Связь признаков с целевой переменной:

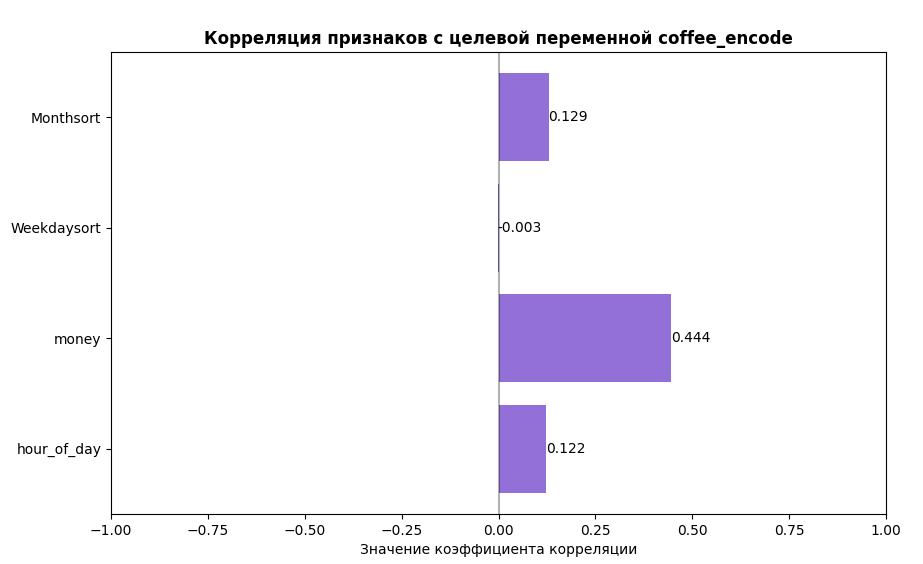


Рисунок 1 – Корреляция признаков с целевой переменной

Таким образом, по диаграмме видим, что самым значимым признаком является «money» (цена), самым незначимым – «weekdaysort» (день недели).

Проведём троекратное тестирование для оценки классификаторов.

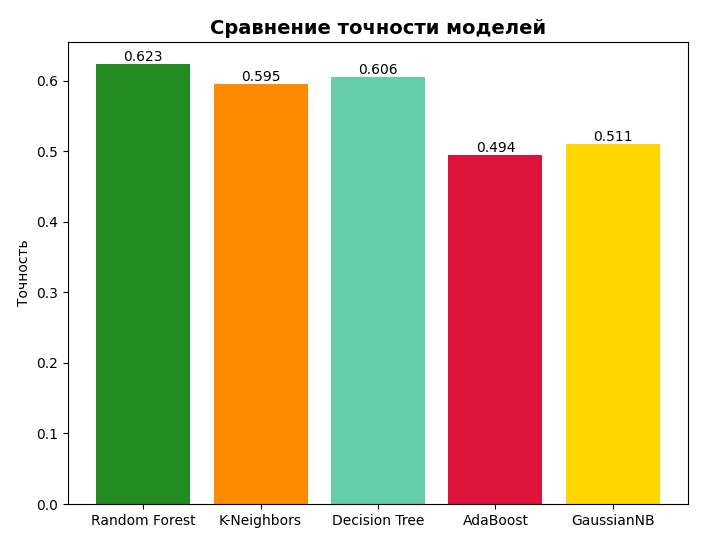


Рисунок 2 – Диаграмма сравнения точности (accuracy) классификаторов №1

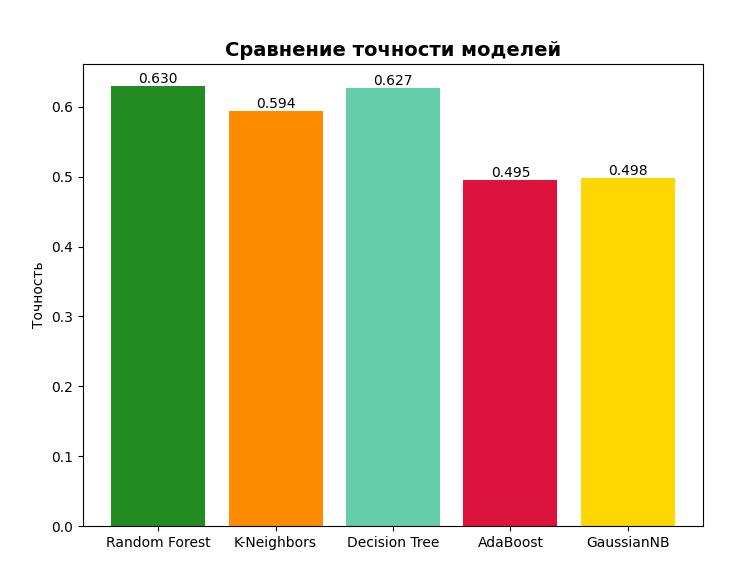


Рисунок 3 – Диаграмма сравнения точности классификаторов №2

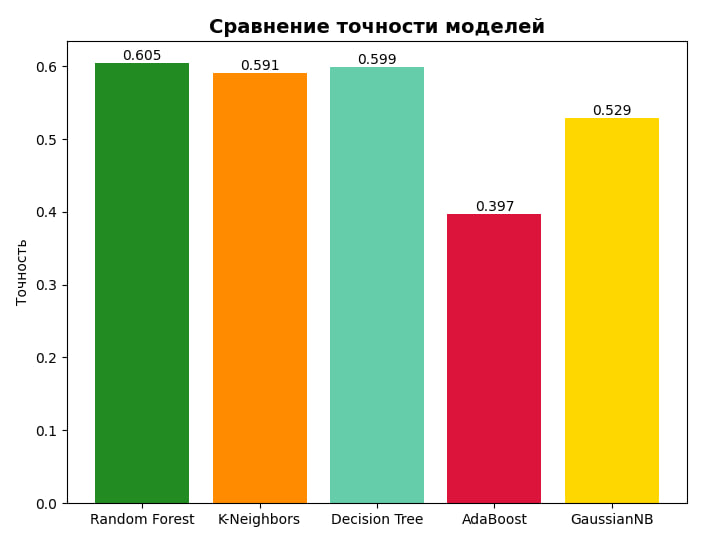
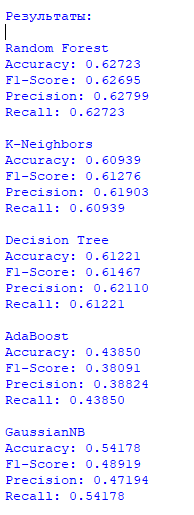


Рисунок 4 – Диаграмма сравнения точности классификаторов №3

Сопоставив диаграммы, можно сделать вывод, что получаемая точность моделей относительно стабильна, за исключением AdaBoost. Это связано с последовательной природой алгоритма адаптивного бустинга.

Самая высокая точность моделей у классификатора RandomForestClassifier(), затем следуют DecisionTreeClassifier() и KNeighborsClassifier(). Точность GaussianNB() заметно ниже. Самая низкая точность у моделей AdaBoostClassifier().



По результатам вычисления метрик, можно заметить, что метрики у моделей Random Forest, K-Neighbors, Decision Tree практически совпадают, а у AdaBoost и GaussianNB отличаются – метрика Accuracy больше,   
чем F1-Score, то есть модели в основном предсказывают самые частые классы.

Уберём самый значимый признак «money» и сравним результаты.

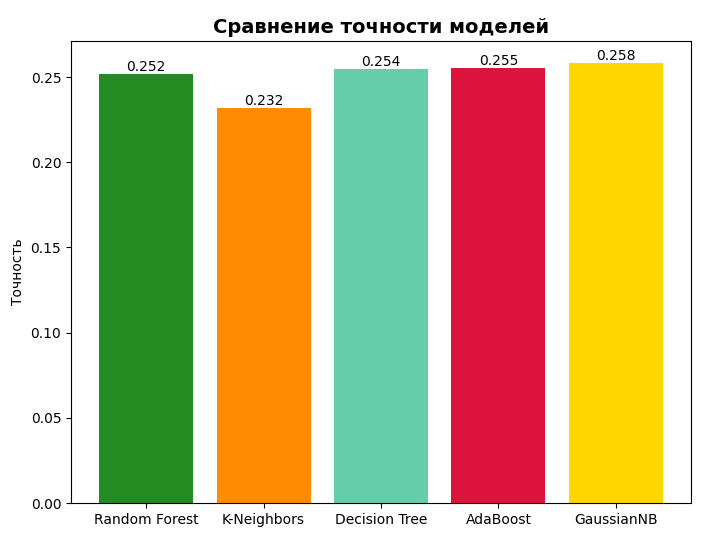
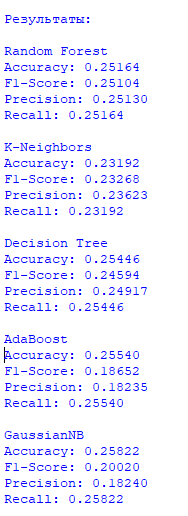


Рисунок 5 – Диаграмма сравнения точности (accuracy) классификаторов №4



По значениям метрик и диаграмме видно, что точность моделей уменьшилась более чем в 2 раза и стала примерно одинаковой у всех моделей. Кроме того, у модели K-Neighbors оказалась самая низкая доля правильных предсказаний. Это связано с тем, что после исключения «money» оставшиеся признаки недостаточно хорошо разделяют классы, что необходимо для KNeighborsClassifier().

Теперь оставим только самый значимый признак – «money» и посмотрим на результаты.

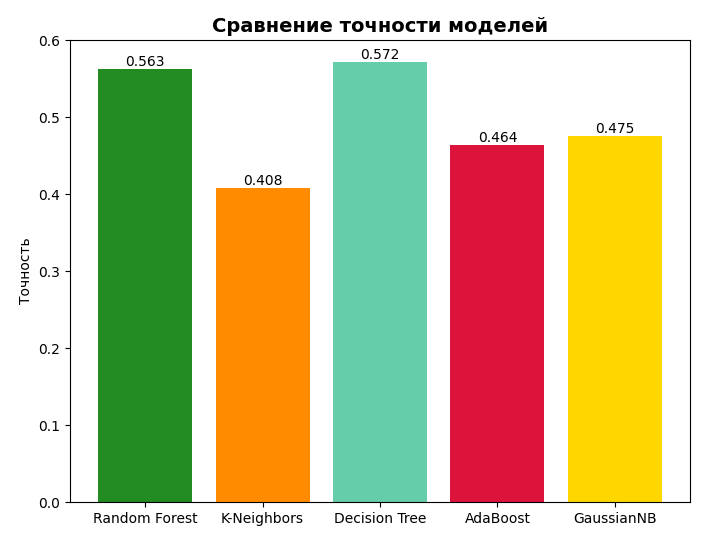
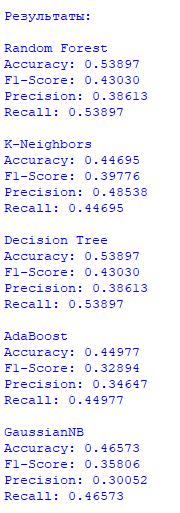


Рисунок 6 – Диаграмма сравнения точности (accuracy) классификаторов №5



По значениям метрик и диаграмме видно, что точность моделей несколько уменьшилась. Также стоит отметить, что значения accurancy и f1-score довольно отличаются у всех моделей. У Random Forest и Decision Tree сильно различаются метрики Precision и Recall. Это говорит о том, несмотря на правильные предсказания, модели также делают много ложных предсказаний. У K-Neighbors самая низкая доля правильных предсказаний <– один признак, однако Precision, напротив, больше чем Recall.

**Выводы по результатам:**

– Для датасета ‘Coffee\_sales.csv’ лучшие результаты показал классификатор RandomForestClassifier(), использующий метод случайного леса. Худшие результаты оказались у AdaBoostClassifier(), использующего метод адаптивного бустинга, и GaussianNB(), использующий метод наивного Байеса.

– Наиболее восприимчивым к количеству признаков является классификатор KNeighborsClassifier().

– У моделей, обученных классификаторами GaussianNB() и AdaBoostClassifier(), в ходе тестирования метрика Recall всегда оказывалась больше, чем Precision, что говорит о наличии большого количества ложных предсказаний.

– Самые нестабильные результаты точности предсказаний показывал классификатор AdaBoostClassifier().

– Самые стабильные результаты точности предсказаний показывали классификаторы RandomForestClassifier() и DecisionTreeClassifier(), использующие деревья решений.

– У моделей, обученных классификатором KNeighborsClassifier(), во всех рассмотренных в ходе исследования случаях Precision > Recall, то есть модели реже ошибаются в предсказаниях, но много пропускают.