Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение
высшего образования
«Уральский федеральный университет
имени первого Президента России Б.Н. Ельцина»

Институт радиоэлектроники и информационных технологий – РТФ

Школа бакалавриата

ОТЧЕТ

По проекту
«Разработка образовательных материалов и проектов в сфере Data Science»

по дисциплине «Проектный практикум»

Заказчик: Фамилия И.О. Ильинский Александр Дмитриевич

Куратор: Фамилия И.О. Ильинский Александр Дмитриевич
ученая степень, ученое звание, должность

Студенты команды _Data Wizards___

Фамилия И.О. Давыдова Елизавета Викторовна

Фамилия И.О. Ивченко Максим Степанович

Фамилия И.О. Пешкин Дмитрий Андреевич

Фамилия И.О. Писемский Михаил Валерьевич

Фамилия И.О. Рыбкин Макар Олегович

Екатеринбург, 2025

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

1.1 Цель

Получение практических навыков в области машинного обучения и Data Science через реализацию мини-проектов, основанных на обработке реальных данных: от их анализа, до построения и оценки машинных моделей.

1.2 Задачи

В рамках проекта были поставлены задачи реализовать две лабораторные работы:

а) первая — на основе датасета *Titanic* — посвящена задаче бинарной классификации;

б) вторая — на данных *Spotify* — ориентирована на решение задачи регрессии.

В рамках каждой лабораторной работы были выдвинуты задачи:

1) Проведение исследовательского анализа данных (EDA), включающего визуализацию, вычисление статистических характеристик и анализ корреляций с целевой переменной;

2) Осуществление feature engineering: создание новых признаков, оценка их значимости и влияния на целевую переменную;

3) Построение и сравнение моделей машинного обучения различных классов: линейные модели, деревья решений, алгоритмы градиентного бустинга и нейронные сети;

4) Проведение оценки моделей с применением кросс-валидации и выбор наилучшего решения по заданным метрикам;

5) Формирование аналитических выводов на всех этапах работы с данными и моделями.

1.3 Актуальность

Современные цифровые технологии всё глубже интегрируются в повседневную жизнь, изменяя подходы к принятию решений, персонализации сервисов и обработке информации. Во многих отраслях — от медицины до музыки — активно применяются методы автоматизированного анализа

данных и предсказательного моделирования. Умение работать с данными, выявлять закономерности и строить математические модели становится ключевой компетенцией специалистов в самых разных сферах.

Проект предоставляет возможность не только освоить базовые инструменты анализа данных и машинного обучения, но и применить их на практике — с ориентацией на реальные задачи и открытые источники данных. Такой подход способствует развитию критической оценки моделей и уверенного использования современных технологий обработки информации.

1.4 Планируемый результат

Создание 2 ноутбуков с анализов предложенных датасетов, работа над каждым из которых осуществляется в два этапа: сначала индивидуально, после чего на основе работ всех участников команды формируется итоговое групповое решение, объединяющее наиболее эффективные подходы.

ОСНОВНАЯ ЧАСТЬ

1.1 Общее решение Titanic

1.1.1 Исследовательский анализ данных (EDA)

Был проведен анализ данных о 891 пассажире Титаника. Ключевые наблюдения:

- Размер данных: 891 запись, 12 признаков.
- Пропуски:
  - Age – 177 пропусков.
  - Cabin – 687 пропусков.
  - Embarked – 2 пропуска.
- Распределение выживших:
  - 38% выживших, 62% погибших.
- Большинство пассажиров – из 2 и 3 классов.

Визуализация и ключевые закономерности

- Выживаемость по полу(рисунок 1):
  - Женщины выживали чаще , чем мужчины

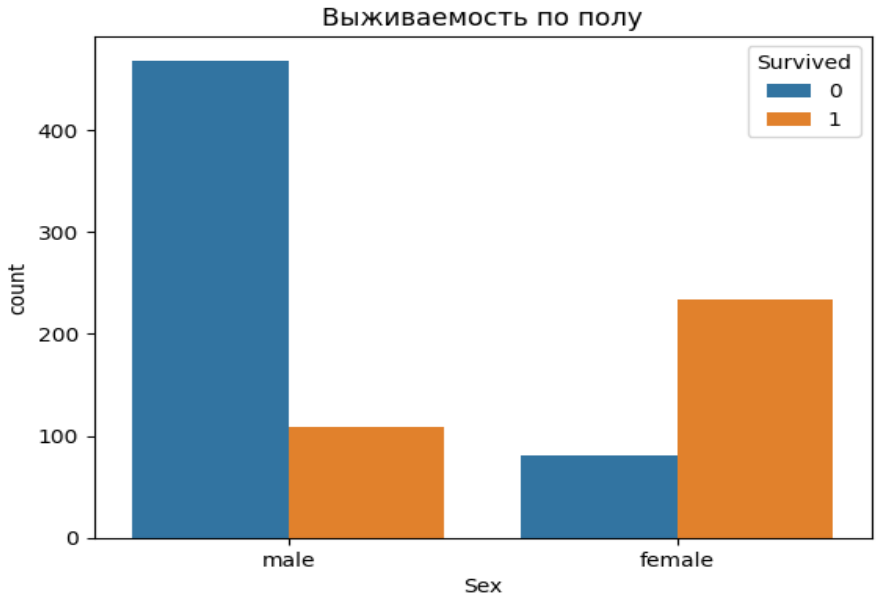

Рис. 1

- Выживаемость по классу билета(рисунок 2)
  - Люди с более высоким классом билета, выживали с большей вероятностью

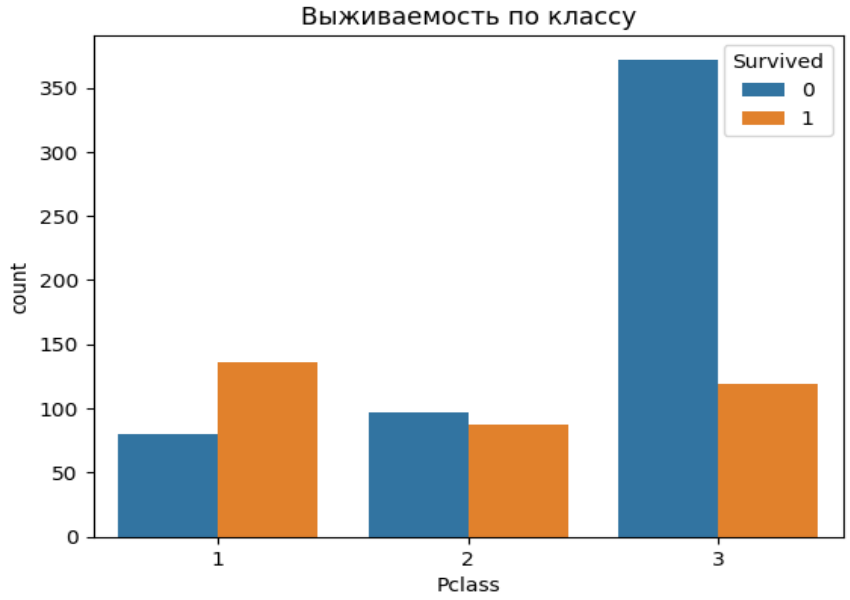

Рис. 2

- Выживаемость по возрасту(рисунок 3)
    - Дети (0–10 лет) имели наивысшую выживаемость.
    - Пассажиры старше 60 лет выживали реже.

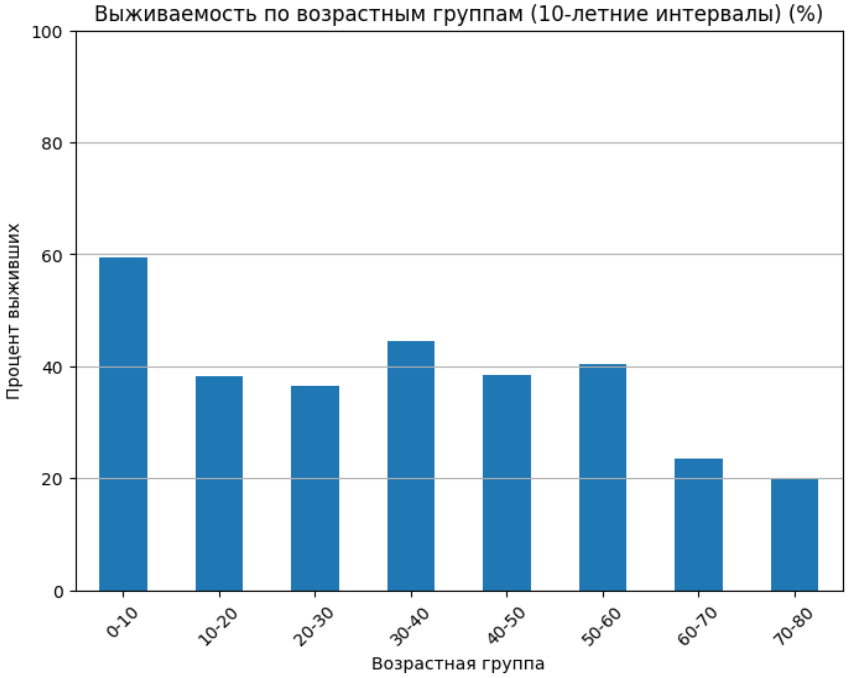

Рис. 3

- Выживаемость по размеру семьи(рисунок 4)
    - Одиночки и большие семьи (5+ человек) выживали реже.
    - Оптимальный размер семьи – 2–4 человека

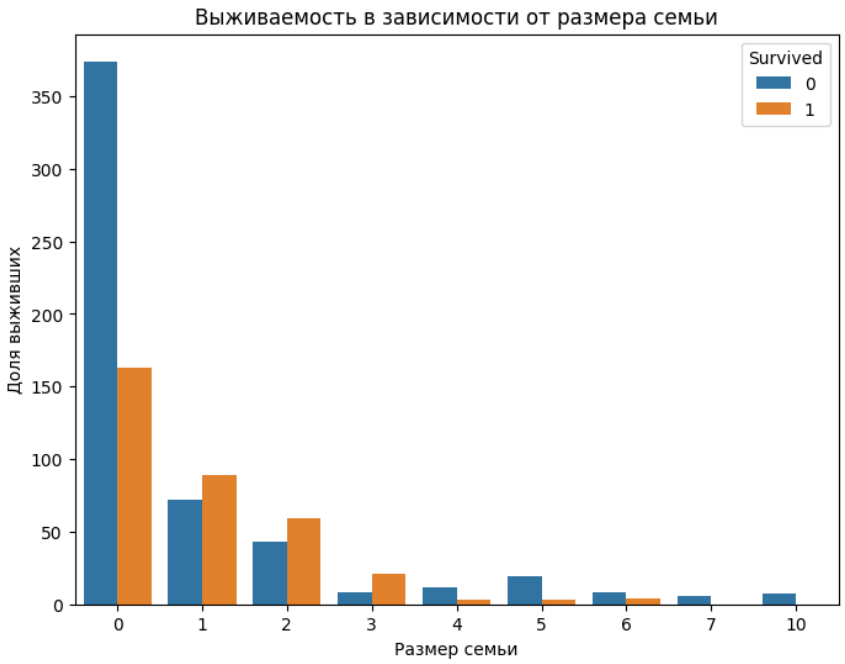

Рис. 4

- Выживаемость по стоимости билета(рисунок 5)
    - Пассажиры с билетами дороже $50 выживали чаще.

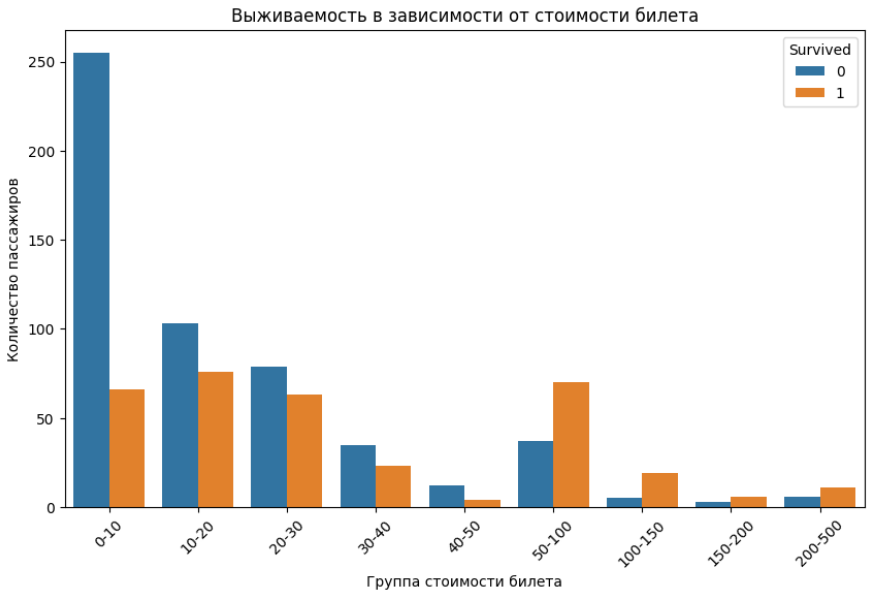

Рис. 5

1.1.2 Feature Engineering

Для улучшения качества данных были обработаны пропуски:

- Age – заполнена медианой по Pclass и Title.
- Fare – заполнена медианой.
- Cabin - неизвестные типы были заменены на U

и созданы новые признаки

- FamilySize размер семьи
- IsAlone = 1, показывает, есть ли родственники на корабле
- Title титул на основе приставки в имени(Mr, Mrs и тд).
- FareGroup и AgeBin – категоризация по интервалам.
- CabinDeck Часть корабля в которой находится каюта пассажира
- Редкие приставки были заменены на Rare.

Была построена матрица корреляции и сделаны выводы о важности каждого из признаков.(рисунок 6)

- Sex_female 0.54 - женщин выжило больше, чем мужчин (для Sex_male корреляция -0.54)
- Fare - чем дороже билет, тем выше вероятность выживания
- Pclass1 - чем выше класс каюты, тем выше вероятность выживания (для Pclass3 корреляция -0.32)
- Title_Miss и Title_Mrs 0.33 и 0.34 - так как сильная корреляция с полом, то и здесь тоже (для Title_Mr -0.55)
- Embarked_C - те кто сели на Титаник в Cherbourg с большей вероятностью выжили
- IsAlone - 0.20 те у кого на корабле были родственики с большей вероятностью выжили

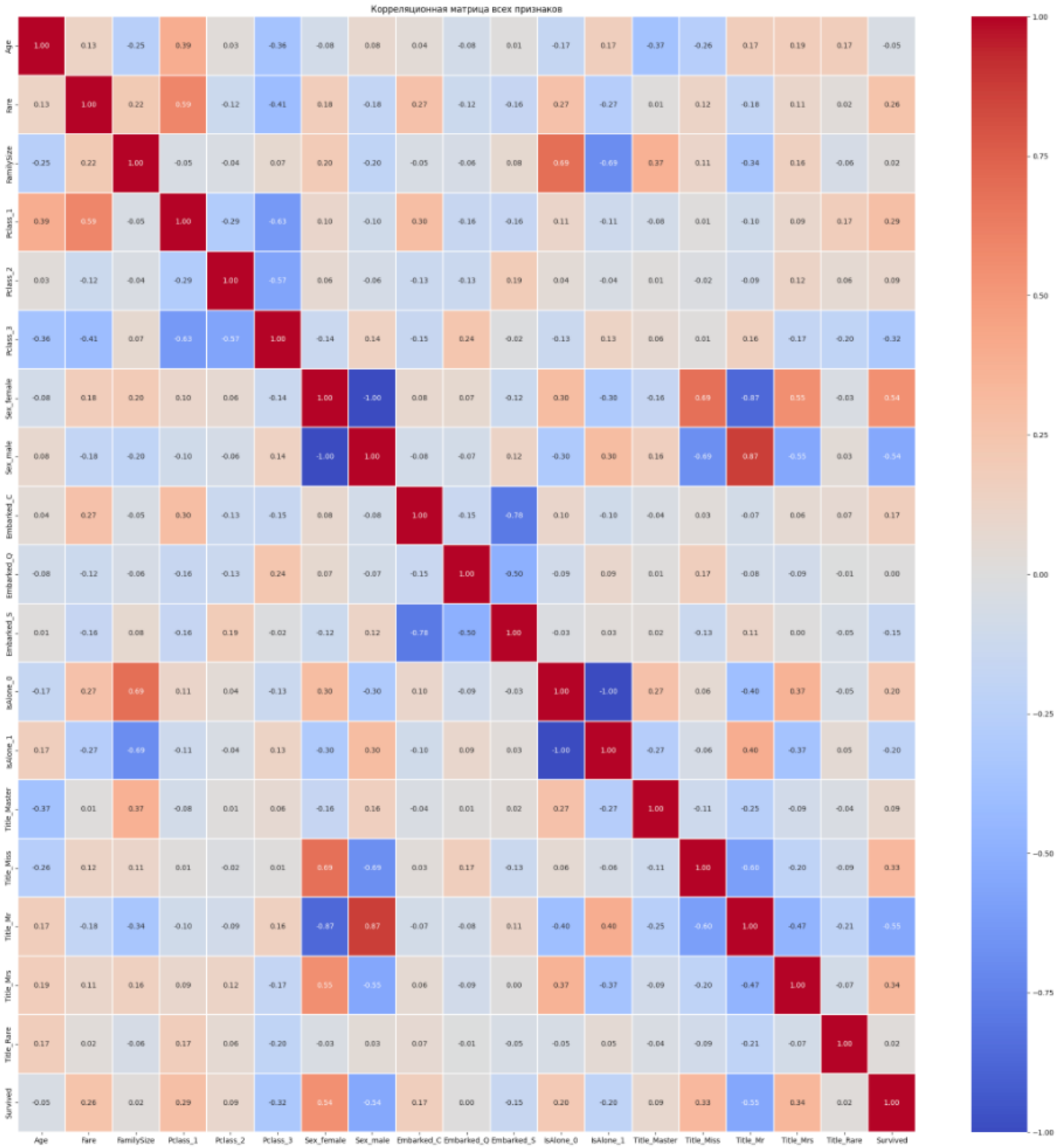

Рис. 6

Самые кореллирующие признаки были разделены на числовые('Age', 'Fare', 'FamilySize') и категориальные признаким('Age', 'Fare', 'FamilySize')

1.1.3 Построение и оценка моделей

Было протестировано 8 моделей с настройкой гиперпараметров через GridSearchCV (метрика – accuracy). Были выявлены лучшие параметры и их результаты для каждой из моделей.

Модель	Accuracy	ROC AUC	Лучшие параметры
SVC	83.15%	0.92	C=1, kernel='poly', degree=4, gamma=0.1
Gradient Boosting	83.02%	0.91	learning_rate=0.1, max_depth=3, n_estimators=200
Random Forest	82.87%	0.90	max_depth=4, n_estimators=500
XGBoost	82.87%	0.90	learning_rate=0.01, max_depth=4, n_estimators=200
LightGBM	82.59%	0.91	learning_rate=0.05, max_depth=8
Logistic Regression	82.17%	0.89	C=10, solver='saga'
CatBoost	82.03%	0.90	depth=6, iterations=300
KNN	81.89%	0.91	n_neighbors=3, weights='uniform'

Модель SVC показала наилучшие результаты с параметрами:

{C=1, kernel='poly', degree=4, gamma=0.1}

и результатом

Accuracy: 83.15%

ROC AUC: 0.92

Была проведена проверка на переобучение:

Accuracy на трейне: 83.15%,

Accuracy на валидации: 83.24%

Разница минимальна, из чего сделан вывод, что переобучения нет.

1.1.4 Ссылка на решение:

https://github.com/MakarRybkin/Titanic_Kaggle

1.2 Общее решение Spotify

1.2.1 Исследовательский анализ данных (EDA)

Был загружен датасет, содержащий 114000 записей, представляющих собой информацию о музыкальных треках с платформы Spotify. Данные были очищены от пропусков и дубликатов, удалены столбцы, не несущие полезной информации, размер датасета уменьшился до 89740 записей.

Был проведён анализ целевой переменной popularity — популярность трека от 0 до 100. (рисунок 7) Самая распространенная категория популярности 0.

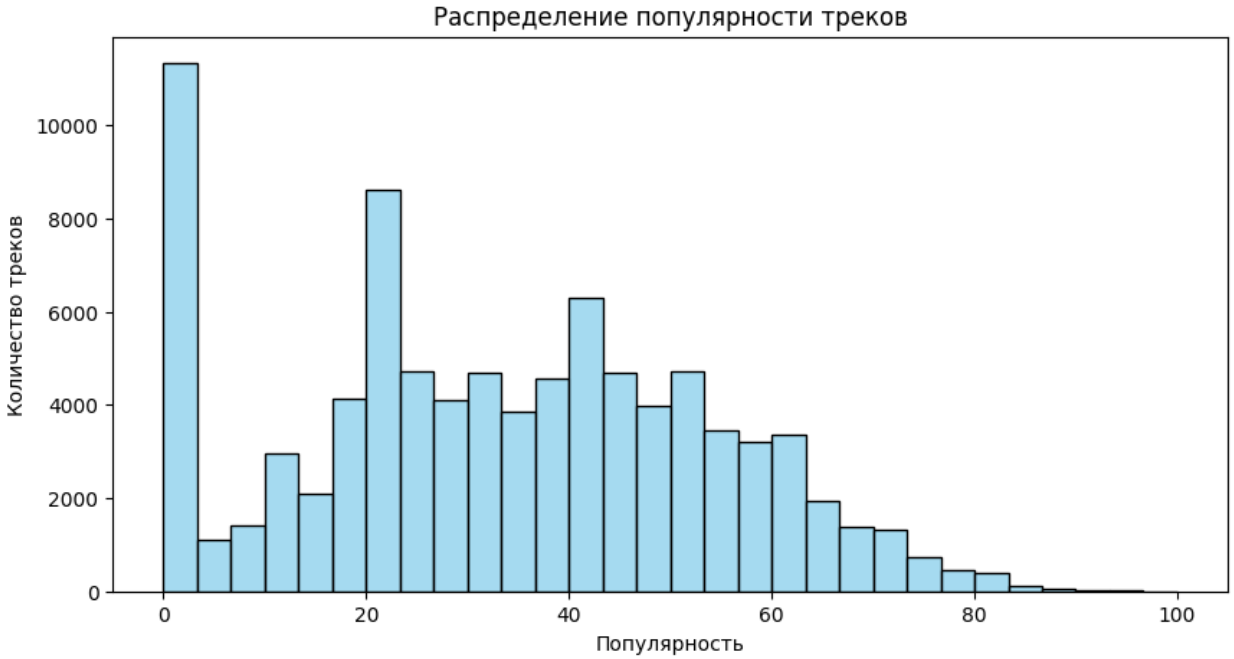

Рисунок 7. Распределение популярности треков

Самые популярные исполнители: Goerge Jones, my little airport, Beatles (рисунок 8).

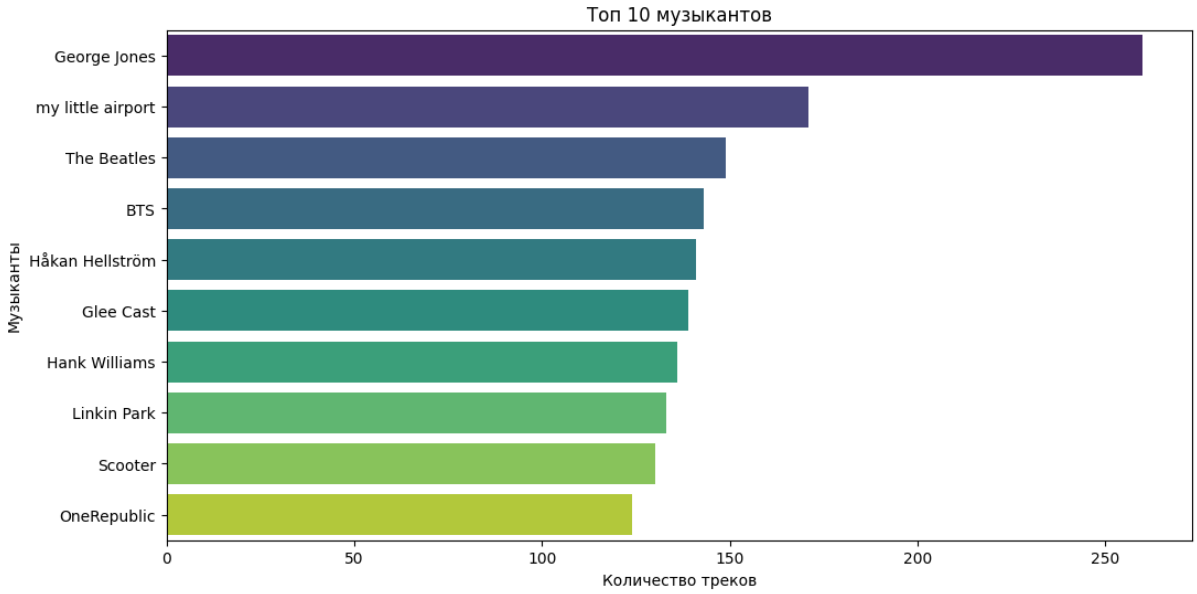

Рисунок 8. Самые популярные исполнители

Проведён анализ корреляции жанра трека с популярностью. (рисунки 9, 10) Наибольшую корреляцию имеет жанр k-film 0.13, а наименьшую имеет iranian -0.15. Видно, что жанр не сильно коррелирует с целевой переменной.

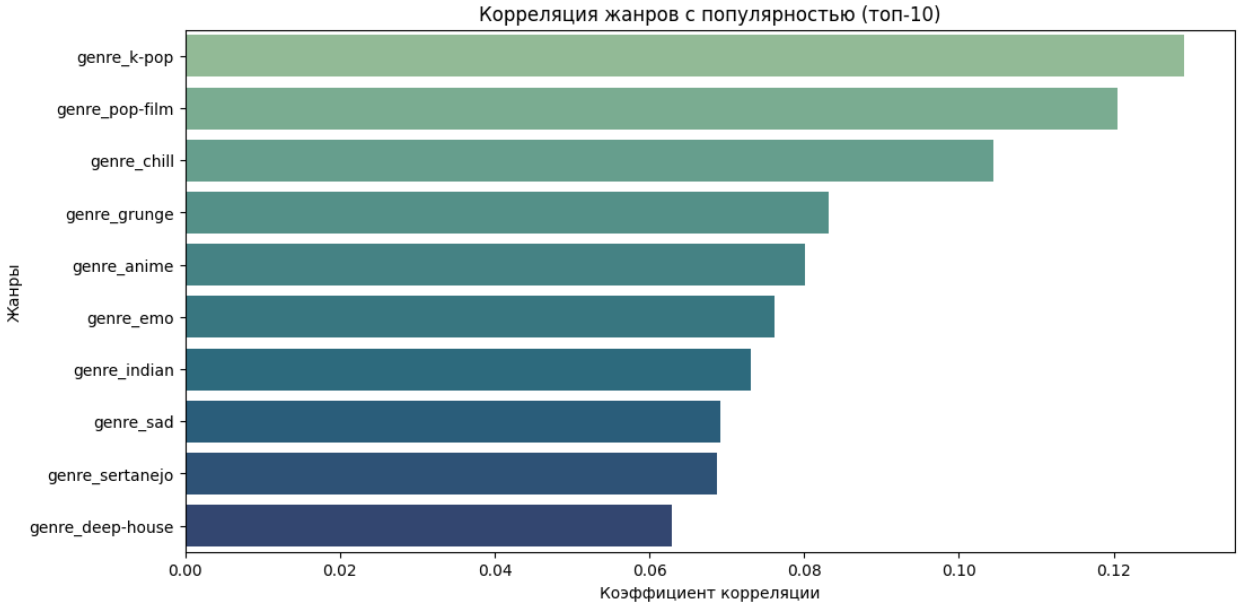

Рисунок 9. Самые популярные жанры треков

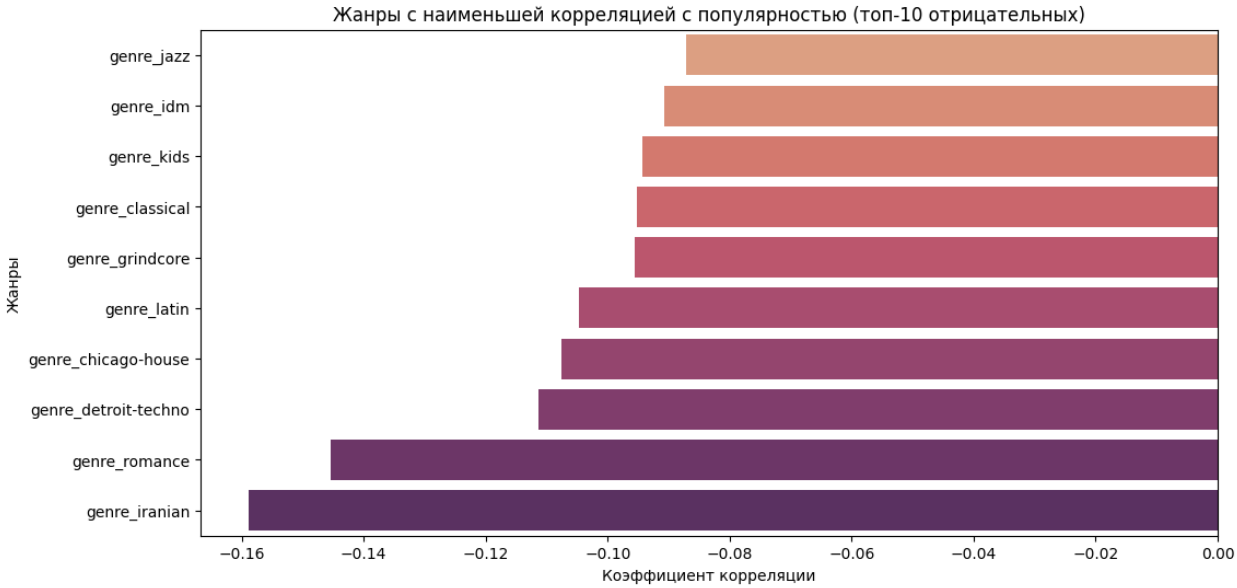

Рисунок 10. Самые непопулярные жанры треков

Для определения других зависимостей с целевой переменной была построена матрица корреляции (рисунок 11). Было определено, что с popularity есть слабая отрицательная корреляция у параметра instrumentalness (инструментальность). остальные корреляции с popularity незначительны

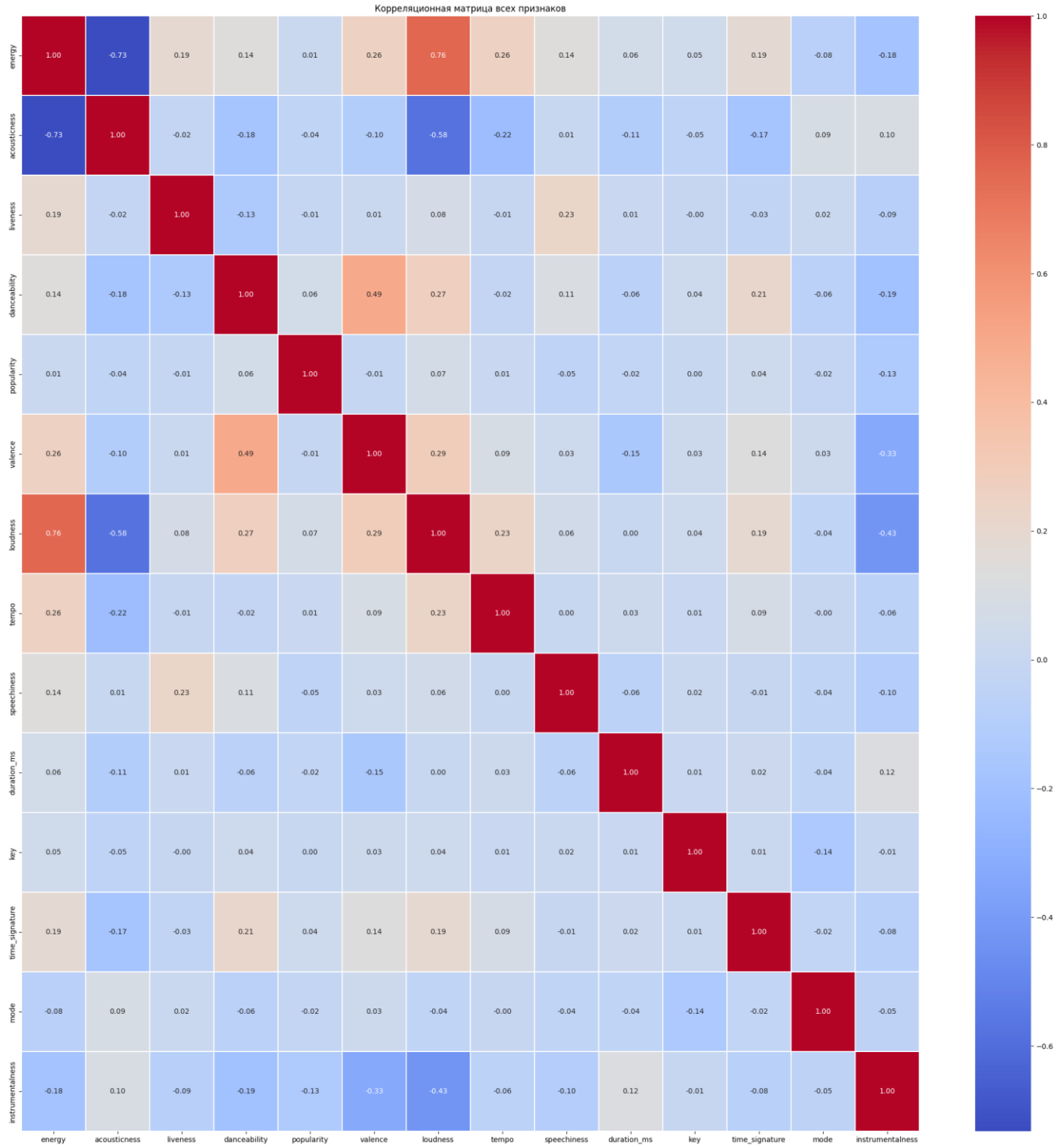

Рисунок 11. Корреляционная матрица всех признаков

1.2.2 Feature engineering

Для лучшего понимания зависимостей в данных создал новые переменные:

- energy_valence – отражает сочетание энергичности и эмоциональной окраски трека;
- dance_energy - комбинация танцевальности и энергии;
- is_spoken - бинарный индикатор разговорного трека (speechiness > 0.66);
- tempo_loudness - взаимодействие темпа и громкости.
- track_name_word_count - количество слов в названии трека;
- artist_track_count - количество треков исполнителя.

1.2.3 Построение и оценка моделей

Разделили данные на train и test выборки для обучения и локального тестирования модели. Для работы моделей с данными закодировали категориальные и масштабировали числовые переменные.

Использовали сетку GreadSearchCV из sklearn для подбора лучших параметров для каждой модели и выбора лучшей модели по метрике rmse (корень из средней квадратичной ошибки), получили такие значения для каждой из исследуемых моделей (таблица 2):

Таблица 2 — Подбор параметров

Модель	RMSE	Лучшие параметры
CatBoostRegressor	9.28	iterations = 1700, learning_rate=0.07, random_seed=42, has_time=True, depth=11, l2_leaf_reg=11, task_type="GPU"
Linear Regression	16.8657	-
Lasso Regression	16.9067	model__alpha = 0.01

Продолжение таблицы 2

Модель	RMSE	Лучшие параметры
Ridge Regression	16.8656	model__alpha = 1
ElasticNet	16.9832	model__alpha: 0.01, model__l1_ratio = 0.9
XGBoost	15.0323	model__colsample_bytree = 1.0, model__device = gpu, model__learning_rate = 0.3, model__max_depth = 7, model__n_estimators = 200
LightGBM	15.0726	model__device_type = gpu, model__feature_fraction = 1.0, model__learning_rate = 0.1, model__n_estimators = 200, model__num_leaves = 63
Random Forest	18.7602	model__max_depth = 9, model__n_estimators = 100
Gradient Boosting	15.1967	model__learning_rate = 0.3, model__max_depth = 7, model__n_estimators = 200

В итоге модель Catboost оказалась лучшей (минимальная средняя квадратичная ошибка на предсказаниях) с RMSE = 9.28 на тренировочных данных и 9.5 на тестовых.

1.2.4 Ссылка на решение

https://github.com/MakarRybkin/Spotify_popularity_reg

2 Отчеты участников

В этом разделе представлены отчеты по личным решениям каждого из участников команды

2.1 Давыдова Елизавета Викторовна

2.1.1 Titanic

2.1.2 Исследовательский анализ данных (EDA)

Были загружены и изучены данные о 891 пассажире «Титаника». С помощью базовых функций Pandas проанализирована структура таблицы, выявлены пропуски в столбцах «Возраст» и «Каюта». Далее с помощью графиков была проведена визуализация выживаемости в зависимости от пола, класса билета и стоимости. Отмечено, что женщины выживали значительно чаще мужчин, среди пассажиров преобладали 2 и 3 классы, а вероятность выживания возрастала при стоимости билета выше 50 долларов.

2.1.3 Feature Engineering

Для улучшения качества данных были созданы новые признаки. Из столбца с именем извлечён титул (например, Mr, Mrs), что позволило сделать предположения о семейном статусе. Дополнительно был рассчитан размер семьи на борту, а также выделена первая буква из номера каюты — возможный индикатор местоположения на судне. Эти признаки расширили исходное описание пассажиров и стали частью модели.

2.1.4 Построение и оценка моделей

После заполнения пропусков и кодирования категориальных переменных были построены различные модели классификации:

– **Random Forest** - точность 82,1%, ROC AUC - 0.88;

– **Logistic Regression** - точность 81%, ROC AUC - 0.88;

– **KNN** - точность 76.5%, ROC AUC - 0.82 (подбор параметров через GridSearchCV);

– **Decision Tree** - точность 82.1%, ROC AUC - 0.87 (подбор параметров через GridSearchCV);

– **LightGBM** - точность - 81%, ROC AUC – 89.3;

– **CatBoost** - точность – 82.7%, ROC AUC - почти 0.90;

– Также реализована простая нейронная сеть на PyTorch. Архитектура включала два скрытых слоя, обучение проходило 30 эпох. Итоговая точность составила 79,3%.

Наилучшие результаты показали ансамблевые модели, в особенности CatBoost, что позволяет рекомендовать их для решения данной задачи классификации.

2.1.5 Ссылка на ноутбук

https://colab.research.google.com/drive/1NZrFTkiuuInhNF3s5O7v4WtcDQ7zc-XT?usp=sharing

2.1.6 Spotify

2.1.7 Исследовательский анализ данных (EDA)

Был загружен датасет из 114 000 музыкальных треков с их характеристиками. Целевая переменная — popularity (популярность трека от 0 до 100).

Ключевые наблюдения:

– Распределение популярности — неравномерное, большинство треков малопопулярны;

– Корреляции с числовыми признаками низкие — линейная связь с популярностью слабая;

– Жанры оказались более коррелируемыми: genre_pop-film, genre_k-pop, dance_chill коррелируют с высокой популярностью, тогда как genre_romance и genre_iranian — с низкой;

– Визуализированы распределения всех числовых признаков — некоторые имеют выраженное смещение и пики.

Линейные зависимости с популярностью выражены слабо.

2.1.8 Feature Engineering

Для усиления модели предсказания популярности были добавлены новые признаки:

– energy_valence — отражает сочетание энергичности и эмоциональной окраски трека;

– dance_energy — комбинация танцевальности и энергии;

– is_spoken — бинарный индикатор разговорного трека (speechiness > 0.66);

– tempo_loudness — взаимодействие темпа и громкости.

Также была создана новая категория top_genre_or_other, где треки делятся на 10 жанров с наибольшей и наименьшей корреляцией с популярностью, остальные жанры объединены в «other».
Созданные фичи позволят дать модели больше информации о музыкальном характере трека.

2.1.9 Построение и оценка моделей

На этапе моделирования была поставлена задача регрессии — предсказание популярности трека. Данные были очищены от пропусков и подготовлены для обучения: категориальные признаки закодированы, числовые нормализованы при необходимости.

Обучено несколько моделей:

– **Random Forest**: MAE ≈ 12.5, R^2 ≈ 0.45;

– **Linear Regression**: MAE ≈ 18.4, R^2 ≈ 0.02;

– **Decision Tree**: MAE ≈ 13.9, R^2 ≈ 0.24;

– **Gradient Boosting**: MAE ≈ 12.0, R^2 ≈ 0.50;

– **MLP Regressor**: MAE ≈ 16.3, R^2 ≈ 0.16;

– **XGBoost**: MAE ≈ 9.80, R^2 ≈ 0.54;

– **MLP Regressor (усиленный)**: MAE ≈ 14.5, R^2 ≈ 0.27.

Модель XGBoost Regressor продемонстрировала наилучшую точность и объясняющую способность. Ансамблевые методы в целом показали более высокую эффективность в задаче предсказания популярности музыкальных треков.

2.1.10 Ссылка на ноутбук

https://colab.research.google.com/drive/1eRXr7kqQtjlYhGTnxP7rgLT9MF47wUMt?usp=sharing

2.2 Рыбкин Макар Олегович

Помимо личных решений лабораторных работ Titanic и Spotify я собирал ноутбуки общих решений всей команды для этих работ. Я тщательно проанализировал каждое решение участников, выделил лучшие стороны и лучшие параметры для каждой модели, протестировал модели каждого участника на Kaggle в учебном соревновании Titanic и выявил, что высшее accuracy(точность правильных ответов модели) = 0.787 среди всех моделей участников получила моя модель GradientBoosting для тестового набора данных Kaggle.

2.2.1 Titanic

2.2.2 Исследовательский анализ данных (EDA)

Проанализировал датасет с данными о пассажирах Титаника, выявил пропуски в столбцах Age, Cabin, Embarked и заменил их на медианные значения и на значения U(unknown) для Cabin. Вывел матрицу корреляций для всех признаков датасета и выявил основные корреляции с таргетным признаком (‘Survived’):

- Sex_female 0.54 - женщин выжило больше, чем мужчин;

-. Pclass1 0.29- чем выше класс каюты, тем выше вероятность выживания (для Pclass3 корреляция -0.32);

- Fare 0.26- чем дороже билет, тем выше вероятность выживания.

2.2.3 Feature Engineering

Создал несколько новых признаков:

- Family - количество родственников на корабле (включая самого себя)

- IsAlone - признак, показывающий есть ли родственники на корабле

- Title - Приставка перед именем человека (Mr, Miss, Mrs)

2.2.4 Построение и оценка моделей

Разделил данные на train и test выборки для обучения и локального тестирования модели.

Построил сетку GreadSearchCV из sklearn для подбора лучших параметров для каждой модели и выбора лучшей модели по метрике accuracy и получил такие значения для каждой из исследуемых моделей:

– **Logistic Regression** – accuracy 0.8217;

– **Random Forest** - accuracy 0.8287;

– **Gradient Boosting** - accuracy 0.8302;

– **SVC** - accuracy 0.8315;

– **KNN** – accuracy 0.8189;

– **XGBoost** - accuracy 0.8287;

Лучшую точность (accuracy) показали SVC, Gradient Boosting и XGBoost

После этого я проверил SVC с лучшими подобранными параметрами на локальной, искусственно созданной тестовой выборке, на возможное переобучение модели, но получил accuracy 0.829 из чего следует, что модель не переобучилась и хорошо классифицирует на новых данных. После этого проверил свою модель в соревновании на Kaggle Titanic и получил хорошее accuracy = 0.787

2.2.5 Ссылка на ноутбук

https://colab.research.google.com/drive/1R5wAS4nl2-S_FuTzIB2YB-fwTu0A57CQ?usp=sharing

2.2.6 Spotify

2.2.7 Исследовательский анализ данных (EDA)

Проанализировал датасет с данными о треках в Spotify. Увидел пропуск 1 в столбцах album_name, track_name, artists, но после того, как я удалил дубликаты в данных по track_id (уникальному идентификационному id из Spotify), то пропуски пропали и датасет уменьшился с 114 тысяч треков с дубликатами, до 90 тысяч уникальных треков. Вывел матрицу корреляций для всех признаков датасета и выявил одну слабую отрицательную корреляцию между признаком instrumentalness(инструментальность) и таргетным признаком (‘popularity’).

.

2.2.8 Feature Engineering

Новых показательных признаков для этого датасета придумать не удалось. Выделил категориальные и числовые признаки для модели Catboost. Для остальных моделей тоже выделил числовые признаки и закодировал некоторые категориальные данные (track_genre, explicit) с помощью OneHotEncoder из sklearn, остальные категориальные данные при кодировании и передаче в модели сильно увеличивали время подбора параметров, поэтому я решил их не использовать.

2.2.9 Построение и оценка моделей

Разделил данные на train и test выборки для обучения и локального тестирования модели.

Построил сетку GreadSearchCV из sklearn для подбора лучших параметров для каждой модели и выбора лучшей модели по метрике rmse (корень из средней квадратичной ошибки) и получил такие значения для каждой из исследуемых моделей:

– **Catboost**: RMSE = 9.28;

– **Linear Regression**: RMSE = 16.88 ;

– **Lasso Regression**: RMSE = 16.92 ;

– **Ridge Regression**: RMSE = 16.88 ;

– **Elastic Net**: RMSE = 16.99;

– **Random Forest**: RMSE = 19.03 ;

– **Gradient Boosting**: RMSE = 15.88;

– **XGBoost**: RMSE = 15.72;

В итоге модель Catboost оказалась самой лучшей (минимальная средняя квадратичная ошибка на предсказаниях) с RMSE = 9.28.

2.2.10 Ссылка на ноутбук

https://colab.research.google.com/drive/15dCcQXmrnBvNxcWqeP8Jqze3acD06rF-?usp=sharing

2.3 Пешкин Дмитрий Андреевич

2.3.1 Titanic

2.3.2 Исследовательский анализ данных (EDA)

Были загружены и изучены данные о 891 пассажире Титаника. Проведен анализ структуры данных с выявлением ключевых закономерностей:

- Визуализация выживаемости по классам показала, что пассажиры 1-го класса имели значительно более высокие шансы на спасение
- Анализ семейного статуса выявил, что одиночки и члены больших семей (5+ человек) имели меньшие шансы на выживание
- Возрастной анализ подтвердил приоритет спасения детей - группа 0-10 лет показала наивысшую выживаемость
- Корреляционный анализ выделил сильную зависимость выживания от пола (женщины), класса билета и его стоимости

2.3.3 Feature Engineering

Для улучшения качества данных были созданы и преобразованы признаки:

- Извлечены социальные титулы из имен с группировкой редких значений
- Рассчитан размер семьи и создан признак “IsAlone”
- Заполнены пропуски в возрасте на основе медианных значений по титулу и классу
- Категориальные признаки (пол, порт посадки) преобразованы в числовой формат

Анализ важности признаков показал, что пол , титул и класс наиболее значимы для предсказания

2.3.4 Построение и оценка моделей

Было протестировано 6 моделей с настройкой гиперпараметров через GridSearchCV:

- **Logistic Regression** - точность 84.9%, ROC AUC - 0.89

- **Random Forest** - точность 81.6%, ROC AUC - 0.90
- **SVC** - точность 86.0%, ROC AUC - 0.92 (лучший результат)
- **KNN** - точность 82.7%, ROC AUC - 0.91
- **Gradient Boosting** - точность 82.7%, ROC AUC - 0.91
- **XGBoost** - точность 82.7%, ROC AUC - 0.90

Модель SVC показала наилучшие результаты по всем метрикам. Финальное тестирование на полном наборе данных подтвердило точность модели на уровне 82.27%.

2.3.5 Ссылка на ноутбук:

https://colab.research.google.com/drive/1djQsfN87r1u8vZyjiES1b0AR4wThGpfb?usp=sharing

2.3.6 Spotify

Был проанализирован датасет, который содержал 114000 треков с 20 с толбцами.

В самом начале было решено провести очистку данных.

- Был удален столбец Unnamed: 0, который являлся повторением индексов.
- Были удалены строки с пропусками в столбцах artists, album_name, track_name.
- Было найдено и удалено 451 дубликатов, итоговый размер датасета: 113549 треков.
- Названия столбцов были приведены к нижнему регистру и пробелы заменены на _

2.3.7 Исследовательский анализ данных (EDA)

Проведен анализ структуры данных с выявлением ключевых закономерностей:

- **Распределение популярности**: Сильный перекос в сторону низких значений (медиана около 35)

- **Топ-артисты**: Выявлены наиболее продуктивные исполнители по количеству треков
- **Аудиохарактеристики**: Большинство признаков имеют специфические распределения
- **Корреляционный анализ:** Популярность слабо коррелирует со всеми другими признаками.

2.3.8 Feature Engineering

Были добавлены такие признаки:

- duration_min - длительность в минутах
- energy_danceability_ratio - соотношение энергии и танцевальности
- loudness_scaled - стандартизированные значения громкости
- tempo_category - категории по темпу

2.3.9 Построение и оценка моделей

Для начала было протестировано 6 регрессионных моделей.

Модель	**RMSE**	**R^2**
Linear Regression	21.73	0.060
Random Forest	16.63	0.449
Gradient Boosting	20.89	0.131
Decision Tree	22.89	-0.043
LightGBM	19.96	0.207
XGBoost	18.90	0.289

По итогу была выделена лучшая модель Random Forest. Далее была произведена оптимизация значений для этой модели с помощью GridSearchCV.

- Были выявлены лучшие параметры: max_depth: 50, max_features: 'sqrt', n_estimators: 100
- Лучший результат: RMSE: 15.91 R^2: 0.496

2.3.10 Ссылка на ноутбук:

https://colab.research.google.com/drive/1g9EtXMka7-71d0VCWJovhwxzJdsgHN00?usp=sharing

2.4 Писемский Михаил Валерьевич

2.4.1 Titanic

2.4.2 Исследовательский анализ данных (EDA)

Изучил датасет, содержащий 891 запись, представляющие собой сведения о пассажирах Титаника. Обработал пропущенные значения в данных. Проанализировал распределение числовых переменных, сделал следующие выводы: большинство пассажиров сели в Саутгемптоне, большая часть в возрасте от 20 до 40 лет, мужчин больше, чем женщин. Также сделал корреляционную матрицу, для определения корреляции переменных с целевым классом (Survived), сделал следующие выводы:

– Лучше всех с параметром Survived коррелирует пол: женщин выжило гораздо больше, чем мужчин;

– Чем выше класс каюты, тем больше вероятность выживания: Pclass_1 (0,29), Pclass_2 (0,093), Pclass_3 (-0,32);

– Чем выше стоимость билета, тем выше класс каюты. Fare (0,26);

– Embarked_C (0,17) — люди, севшие в Шербуре, выжили с большей вероятностью.

2.4.3 Feature engineering

Для лучшего понимания зависимостей в данных создал новые переменные:

– Title — титул пассажира. Создан на основе данных столбца с имеем пассажира, редко встречающиеся титулы были сгруппированы;

– FamilySize — размер семьи (количество родственников на корабле, включая самого пассажира). Создан на основе данных столбцов SibSp (количество братьев, сестёр или супругов, путешествующих с пассажиром) и Parch (количество родителей или детей, с которыми путешествовал пассажир);

— Dack — палуба, на которой находится каюта пассажира, редкие значения также были сгруппированы;

2.4.4 Построение и оценка моделей

Для лучшей работы моделей с данными закодировал категориальные и масштабировал числовые переменные. Разделил полученный датасет на тренировочную и тестовую выборки, отдельно выделив целевой признак Survived. После подбора гиперпараметров, для которого использовал GridSearchCV, получил следующие результаты точности классификации (метрика accuracy):

— **Logistic Regression**, точность = 0.8301%;

— **Decision Tree**, точность = 0.8343%;

— **Random Forest**, точность = 0.8343%;

— **Gradient Boosting**, точность = 0.8399%;

— **XGBoost**, точность = 0.8427%;

— **LightGBM**, точность = 0.8315%;

— **SVC**, точность = 0.8343%;

— **KNN**, точность = 0.8427%

После подбора гиперпараметров лучшую точность классификации на тренировочных данных показали модели KNN и XGBoost: 0.8427%. Проверил работу модели XGBoost на тестовой выборке, где точность была равна 0.8268%, из чего был сделан вывод об отсутствии переобучения и корректной работе модели.

2.4.5 Ссылка на ноутбук

https://colab.research.google.com/drive/1QEI7toqPcDt1w0JSCifawun4My9m0S9x?usp=sharing

2.4.6 Spotify

2.4.7 Исследовательский анализ данных (EDA)

Изучил датасет, содержащий 114000 записей о треках Spotify. Обработал пропущенные значения в данных и дубликаты. Проанализировал распределение числовых переменных, сделал следующие выводы:

— Распределение целевой переменной popularity (популярность трека от 0 до 100) скошено влево - большинство треков имеют среднюю популярность (40–70);

— Длительность трека имеет нормальное распределение с пиком около 180–240 секунд;

— Распределение энергичности трека смещено вправо - большинство треков имеют высокую энергичность;

— Общая громкость трека, имеет нормальное распределение с пиком около -10;

Также построил корреляционную матрицу для определения корреляции переменных с целевым классом (popularity), сделал следующие выводы: наибольшую корреляцию с целевой переменной(popularity): -0,13 имеет класс instrumentalness, остальные классы показывают значения <0,1.

2.4.8 Feature engineering

Для лучшего понимания зависимостей в данных создал новые переменные:

— track_name_word_count - количество слов в названии трека;

— artist_track_count - количество треков исполнителя.

2.4.9 Построение и оценка моделей

Для лучшей работы моделей с данными закодировал категориальные и масштабировал числовые переменные. Разделил полученный датасет на тренировочную и тестовую выборки, отдельно выделив целевой признак popularity. После подбора гиперпараметров, для которого использовал RandomizedSearchCV, сравнил модели по метрике RMSE (квадратный корень из средней квадратичной ошибки):

— **Linear Regression**, RMSE = 20.1340;

— **Decision Tree**, RMSE = 18.8404;

— **Random Forest**, RMSE = 15.9583;

— **Gradient Boosting**, RMSE = 15.9856;

— **XGBoost**, RMSE = 15.4745;

– **LightGBM**, 15.0380;

– **MLP Regressor**, 18.4453

После подбора гиперпараметров лучшее значение для RMSE на тренировочных данных показала модель LightGBM: 15.0380. Проверил работу модели LightGBM на тестовой выборке, где RMSE = 14.9054, из чего был сделан вывод об отсутствии переобучения и корректной работе модели.

2.4.10 Ссылка на ноутбук:

https://colab.research.google.com/drive/1mhVTVgc9HGGIEN37ZhWPgdF9MOElOZJx?usp=sharing

2.5 Ивченко Максим Степанович

2.5.1 Titanic

2.5.2 Исследовательский анализ данных (EDA)

В рамках предварительного анализа были загружены и изучены данные о 891 пассажире «Титаника». С помощью функций head(), info() и анализа пропусков (isnull().sum()) выявлены пропущенные значения в столбцах **Age**, **Cabin** и **Embarked**.

Далее проведён визуальный анализ распределения выживаемости:

- Построены гистограммы зависимости выживаемости от пола, класса билета и стоимости билета;
- Отмечено, что женщины выживали чаще мужчин;
- Выживаемость выше у пассажиров первого класса;
- Пассажиры с более дорогими билетами имели большую вероятность выживания.

2.5.3 Feature Engineering

Для улучшения качества признаков были добавлены новые переменные:

- **Title** — извлечён из столбца Name, содержит обращения Mr, Miss, Mrs и т.д.;
- **FamilySize** — сумма SibSp, Parch и 1 (сам пассажир);
- **IsAlone** — бинарный признак, показывающий, путешествовал ли пассажир один;

- **CabinLetter** — первая буква из номера каюты (или 'U', если неизвестно).

Обработка пропущенных значений:

- Age — заполнен медианой в разрезе Sex и Pclass;
- Embarked — заменён на наиболее часто встречающееся значение;
- Cabin — заполнен значением 'U'.

Категориальные признаки были преобразованы в числовые с помощью OneHotEncoding.

2.5.4 Построение и оценка моделей

Данные были разделены на обучающую и тестовую выборки. Построены и обучены следующие модели классификации:

- **Logistic Regression** — accuracy ≈ 79.3%
- **Random Forest** — accuracy ≈ 81.5%
- **Gradient Boosting** — accuracy ≈ 82.1%
- **XGBoost** — accuracy ≈ 82.7%
- **CatBoost** — accuracy ≈ 83.1%
- **K-Nearest Neighbors** — accuracy ≈ 77.0%
- **Decision Tree** — accuracy ≈ 80.1%

Для некоторых моделей построены ROC-кривые, рассчитаны метрики classification_report и confusion_matrix.

Также реализована простая **нейронная сеть с использованием Keras**. Архитектура включала два скрытых слоя. Модель обучалась в течение 30 эпох и показала точность **около 77.5%** на тестовой выборке.

Наиболее высокую точность продемонстрировали ансамблевые методы (CatBoost, XGBoost, Gradient Boosting), что делает их предпочтительными для решения задачи бинарной классификации выживаемости пассажиров.

Ссылка на ноутбук

Ссылка на ноутбук

2.5.5 Spotify

Данная часть лабораторной работы не была выполнена.

ЗАКЛЮЧЕНИЕ

В рамках командного проекта были успешно реализованы две учебные задачи в области Data Science: классификация выживших пассажиров Титаника и регрессия для предсказания популярности музыкальных треков Spotify.

Titanic

Решение задачи классификации на основе данных о пассажирах Титаника позволило команде последовательно пройти все ключевые этапы жизненного цикла модели: от предварительного анализа данных (EDA) до построения финальных ансамблевых моделей. Особое внимание было уделено созданию новых признаков (feature engineering) и подбору гиперпараметров через GridSearchCV. Итоговая модель, объединяющая лучшие практики участников, достигла точности (accuracy) 78.7% на платформе Kaggle, заняв 1884 место. Результаты и код доступны в репозитории:

GitHub: Titanic_Kaggle

Spotify

Задача регрессии по предсказанию популярности треков Spotify была решена с использованием современных методов анализа аудиоданных и предсказания числовых меток. Финальная командная сборка, объединяющая лучшие находки участников, реализована в ветке master репозитория. В процессе работы были протестированы различные алгоритмы, включая CatBoost, XGBoost и LightGBM. Итоговая модель на основе CatBoost достигла показателей RMSE = 9.28 и MAE = 6.7 на тестовом наборе данных. Результаты и код доступны в репозитории:

GitHub: Spotify_popularity_reg

Общий итог

В процессе работы команда приобрела важные практические навыки: от загрузки и очистки реальных данных до построения моделей, их оптимизации и оценки. Отчётная документация, визуализации и аккуратно оформленные ноутбуки подтверждают высокий уровень проработки каждой задачи. Проект

можно считать успешно завершённым: были достигнуты все заявленные цели, выполнены обе задачи, и на их основе сформированы качественные модели и аналитические выводы.