Министерство образования и науки Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
**«Уральский федеральный университет имени первого Президента России Б.Н.Ельцина» (УрФУ)**

Институт *Радиоэлектроники и информационных технологий-РТФ*

Департамент *Школа бакалавриата*

# ОТЧЕТ
## о лабораторной работе
## по дисциплине «Разработка образовательных материалов и проектов в сфере Data Science»

по теме: **Titanic и Spotify**

Студент:

Федоров Ярослав Владимирович
Коноплянников Николай Дмитриевич
Ирзутов Станислав Дмитриевич

Группа: РИ-230934

Екатеринбург
2025

СОДЕРЖАНИЕ



**Добавлено примечание ([ИГ1]):**

**Добавлено примечание ([ИГ2]):** Между введение и заключением (не включая их) Сделать обычным текстом, не капсом.

**Добавлено примечание ([ИГ3]):** ВВЕДЕНИЕ и ЗАКЛЮЧЕНИЕ не нумеруются
БИБЛИОГРАФИЧЕСКИЙ СПИСОК

**Добавлено примечание ([L4]):**

# Введение

В современном мире анализ данных играет ключевую роль при построении интеллектуальных систем. Целью настоящей работы является проведение полного цикла анализа данных: от разведочного анализа (EDA), генерации признаков, построения моделей машинного обучения и сравнения их качества на двух различных задачах:

- **Классификация выживания пассажиров на Титанике (Titanic)**
- **Анализ популярности музыкальных треков (Spotify)**

В ходе работы были сформированы гипотезы, проведены эксперименты с различными семействами моделей (линейные, деревья решений, бустинг, нейросети), а также реализована кросс-валидация для выбора оптимального подхода.

# Основная часть

## Titanic: EDA, гипотезы, моделирование

### *Задача*

Предсказать, выживет ли пассажир на Титанике по известным признакам.
Тип задачи: **Бинарная классификация**

### Датасет

- Кол-во записей: 891
- Целевая переменная: Survived (0 — нет, 1 — да)
- Признаки: Pclass, Sex, Age, SibSp, Fare, Embarked и др.

### Предварительный анализ и гипотезы

- **Пол и выживание:** Женщины выживали значительно чаще мужчин.
- **Класс обслуживания (Pclass):** Пассажиры 1 класса имели больше шансов выжить.
- **Возраст:** Дети и молодые женщины имели наибольшую выживаемость.
- **Количество родственников:** одиночки выживали реже.

## EDA (визуализация)

- Графики распределения Survived по полу, классу, возрасту.
- Проверка на пропущенные значения (Age, Cabin).
- Корреляционная матрица показала высокую связь Sex, Pclass, Fare с таргетом.

## Feature Engineering

- Кодирование Sex, Embarked (Label/OneHot)
- Создан новый признак FamilySize = SibSp + Parch + 1
- Категоризация возраста (ребенок, подросток, взрослый, пожилой)

## ML-эксперименты

Модели, примененные к задаче:

- **Logistic Regression**
- **Decision Tree**
- **Random Forest**
- **Gradient Boosting (CatBoost)**
- **Нейросеть (простая полносвязная модель на Keras)**

## Кросс-валидация (5 фолдов)

- Наилучший результат показал **CatBoostClassifier**: accuracy ≈ **0.84**
- Нейросеть дала ≈ 0.82 accuracy, но хуже устойчивость на разных фолдах

### Выводы

- Основные признаки: Sex, Pclass, Fare, Age, FamilySize
- Модель бустинга лучше остальных по стабильности и качеству
- Нейросеть применима, но переобучается при малом объеме данных

---

## Spotify: EDA, гипотезы, моделирование

***Задача***

Анализ и предсказание **популярности** музыкальных треков по аудиохарактеристикам.
Тип задачи: **Регрессия или классификация (по уровням популярности)**

### Датасет

- Кол-во записей: 114000
- Целевая переменная: popularity (0–100)
- Признаки: danceability, energy, loudness, acousticness, valence, tempo и др.

### Гипотезы

- Танцевальность (danceability) влияет положительно на популярность
- Слишком "громкие" треки — не всегда популярные (нелинейная связь)
- Энергичность (energy) и valence повышают популярность
- Длительность и explicit не всегда влияют линейно

***EDA***

- Построены boxplot и pairplot для основных признаков
- Корреляция: danceability, energy, valence — умеренная положительная
- Обнаружены выбросы (в основном в loudness, tempo)

**Feature Engineering**

- Категоризация popularity: low (0–30), medium (31–60), high (61–100)
- Масштабирование числовых признаков (MinMaxScaler)
- OneHotEncoding жанров и explicit

**ML-эксперименты**

- **Logistic Regression** (на категориальной популярности)
- **Random Forest Classifier**
- **CatBoostClassifier**
- **Neural Network (Keras)**

**Кросс-валидация (5 фолдов)**

- Наиболее устойчивой оказалась модель **CatBoost**: accuracy ≈ **0.75**
- Нейросеть показала ≈ 0.73 accuracy, но требовала больше ресурсов

**Выводы**

- Популярность можно частично предсказать по аудио-признакам
- Лучшие признаки: danceability, valence, energy, acousticness

- Оптимальная модель — **CatBoost** с кросс-валидацией

---

**Требования заказчика**

- Titanic: Предсказание выживания пассажира для интеграции в систему рекомендаций спасательных операций
- Spotify: Определение признаков успешности трека для платформы рекомендаций

---

**Аналоги**

- Titanic: Kaggle Challenge с открытым лидбордом
- Spotify: Spotify Recommender Engine, проекты Spotify API и Audio Analysis

---

**Используемый стек**

- **Python 3.10**

**Библиотеки:**

- pandas, numpy, matplotlib, seaborn — для анализа
- scikit-learn — модели и предобработка
- catboost — градиентный бустинг
- keras / tensorflow — нейросети
- shap — объяснение моделей

**Датасеты:**

- Titanic: Titanic.csv с Kaggle
- Spotify: предоставлен CSV + Google Drive

---

**План действий**

1. Загрузка и очистка данных
2. EDA и визуализация
3. Формулировка гипотез
4. Feature Engineering
5. Обучение и сравнение моделей
6. Кросс-валидация
7. Итоговые выводы

## Заключение

В ходе выполнения лабораторных работ были рассмотрены две задачи — классификация выживаемости на Титанике и оценка популярности треков Spotify. Проведен EDA, выделены значимые признаки, построены и сравнины модели. Наилучшие результаты показал алгоритм **CatBoost**, который обеспечивал высокую точность и устойчивость.

Обе задачи подтвердили, что качественный предварительный анализ и проработка признаков играют ключевую роль в машинном обучении.

---

## Источники использованные


1. Kaggle Titanic Dataset: https://www.kaggle.com/c/titanic
2. Spotify Dataset: https://www.kaggle.com/datasets/zaheenhamidani/ultimate-spotify-tracks-db
3. Scikit-learn documentation: https://scikit-learn.org
4. CatBoost documentation: https://catboost.ai
5. Keras & TensorFlow: https://keras.io
6. Matplotlib, Seaborn: https://seaborn.pydata.org
7. SHAP: https://github.com/slundberg/shap