Санкт-Петербургский Государственный Университет
Факультет Прикладной математики - Процессов управления

Программа
«Прикладная математика, фундаментальная информатика и программирование»
Кафедра теории систем управления электрофизической аппаратурой

Отчет по проектной работе
На тему «Применение методов машинного обучения на задачах регрессии, бинарной и многоклассовой классификации»

Студента 3 курса
Валиевой Н.Л.

Санкт-Петербург — 2025

1. Подготовка данных

1.1 Описание данных

Для решения задач машинного обучения использовались два связанных источника данных:

Таблица **cars_info** содержит основную информация о транспортных средствах, представленных в автопарке компании:

- car_id - уникальный идентификатор автомобиля, полезной информации для нашей задачи не несет, поэтому мы его уберем
- model - марка автомобиля
- car_type - класс автомобиля (седан, SUV, и т.д.)
- fuel_type - тип топлива
- car_rating - средняя оценка автомобиля пользователями
- riders - общее количество поездок, выполненных автомобилем
- year_to_start - год выпуска автомобиля
- year_to_work - год начала эксплуатации автомобиля в автопарке
- main_city - основной город эксплуатации (Москва или Санкт-Петербург)
- target_reg - количество дней до поломки (для задачи регрессии)
- target_class - класс поломки (9 категорий — для задачи многоклассовой классификации)
- target_bin - класс поломки (2 категории — для задачи бинарной классификации)

Дополнительно использовались данные о поездках из таблицы rides_info, включающей информацию о пользовательских оценках, расстояниях, скоростях и качестве поездок. Для повышения информативности исходного датасета данные о поездках были агрегированы по идентификатору автомобиля car_id:

- sum_rating - суммарный рейтинг автомобиля: произведение среднего рейтинга на количество поездок
- year_in_work - количество лет эксплуатации автомобиля в автопарке
- mean_distance - среднее пройденное расстояние в год эксплуатации
- riders_per_year - среднее количество поездок в год

1.2 Постановка задачи

Были поставлены три задачи машинного обучения. В задаче регрессии нужно предсказать целевую переменную target_reg, что означает количество дней до поломки. В задаче бинарной классификации нужно предсказать наличие поломки, связанной с двигателем, а в задаче многоклассовой классификации предсказать класс поломки. Для бинарной классификации категории, содержащие упоминание двигателя (engine_*), были объединены в один класс:

- 0 - 1247 наблюдений (без проблем с двигателем)
- 1 - 1090 наблюдений (поломка связана с двигателем)

Для оценки качества моделей данные были разделены в пропорции: 80:20, что соответствует 80% наблюдений для обучения и 20% — для тестирования. При разделении использовалась стратификация по целевой переменной для задач классификации, чтобы сохранить исходное соотношение классов.

В итоговом датасете после объединения и обогащения содержалось 2337 записей и 15 числовых

целевая метка	количество данных
engine_overheat	289
gear_stick	274
another_bug	279
engine_check	270
break_bug	270
engine_ignition	269
engine_fuel	262
electro_bug	249
wheel_shake	165

Таблица 1: Распределение по целевым меткам

	car_rating	year_to_start	riders	year_to_work	mean_rating	distance_sum	rating_min	speed_max	user_ride_quality_median	deviation_normal_count	user_uniq	sum_rating	year_in_work	mean_distance	riders_per_year
mean	4.354	2014.04	63711.444	2018.069	4.478	13569122.155	0.138	160.984	-0.329	174.0	171.624	285412.889	4.029	3579828.850	17412.336
std	0.996	1.685	33755.012	2.1594	0.418	4253208.772	0.163	33.211	9.581	0.0	3.158	154444.486	2.659	510188.107	33503.91
min	0.98	2011.0	23.0	2014.0	3.529	1478866.319	0.0	91.423	-38.116	174.0	136.0	96.652	-3.0	-24272815.761	-134781.0
25%	3.74	2013.0	41053.0	2016.0	4.155	10587577.624	0.1	123.609	-6.141	174.0	171.0	180214.616	2.0	1764125.028	4540.75
50%	4.36	2014.0	64415.0	2018.0	4.442	13218518.296	0.1	172.435	-0.195	174.0	172.0	288546.876	4.0	2788104.41	12302.142
75%	5.0	2015.0	86515.0	2020.0	4.675	16317876.612	0.1	188.597	5.37	174.0	173.0	382679.343	6.0	4795803.406	26948.25
max	8.9	2017.0	142862.0	2022.0	5.7	31197854.661	1.5	209.982	35.77	174.0	174.0	750383.665	11.0	31197854.661	138813.0

Таблица 2: Статистики для численных переменных

признаков.

Наиболее сильные корреляции наблюдаются между: riders и year_to_start (r = 0.99), mean_distance и riders_per_year (r = 0.88), year_to_work и year_in_work (r = 0.77).

Для подготовки данных к обучению были выполнены следующие шаги:

- Обработка пропусков - отсутствующие значения в числовых признаках заменялись на нули.
- Категориальные признаки model, car_type, fuel_type, main_city были преобразованы с помощью LabelEncoder.
- Нормализация числовых признаков - для алгоритмов, чувствительных к масштабу (например, логистическая регрессия и XGBoost), использовалось масштабирование с помощью StandardScaler.

2. Задача регрессии

В рамках задачи регрессии был проведён сравнительный анализ девяти моделей машинного обучения: Linear Regression, Polynomial Regression, SVR (linear и RBF), Decision Tree, Random Forest, XGBoost, CatBoost и LightGBM. Для всех моделей был выполнен перебор гиперпараметров с использованием байесовской оптимизации в заданных диапазонах, а также измерено время подбора параметров, время обучения и время предсказания. Основной метрикой качества выступала MAE (Mean Absolute Error).

Наилучшее качество показала модель Random Forest, достигнув MAE = 9.0785 при следующих опти-

	car_rating	year_to_start	riders	year_to_work	mean_rating	distance_sum	rating_min	speed_max	user_ride_quality_median	user_uniq	sum_rating	year_in_work	mean_distance	riders_per_year
car_rating	1.0	-0.02	-0.01	-0.02	0.02	0.0	0.06	-0.0	0.03	0.0	-0.01	-0.0	-0.02	-0.03
year_to_start	-0.02	1.0	0.99	0.06	0.0	0.01	0.0	-0.03	-0.02	-0.08	0.97	-0.59	0.05	0.23
riders	-0.01	0.99	1.0	0.05	0.0	0.01	0.01	-0.04	-0.03	-0.07	0.98	-0.59	0.05	0.24
year_to_work	-0.02	0.06	0.05	1.0	0.0	0.0	-0.0	-0.01	0.02	-0.06	0.05	0.77	-0.11	0.02
mean_rating	0.02	0.0	0.0	0.0	1.0	-0.13	0.36	-0.58	-0.01	0.04	0.18	-0.0	-0.04	-0.02
distance_sum	0.0	0.01	0.01	0.0	-0.13	1.0	-0.06	0.13	-0.0	0.05	-0.01	-0.0	0.22	0.01
rating_min	0.06	0.0	0.01	-0.0	0.36	-0.06	1.0	-0.27	-0.01	0.01	0.07	-0.01	-0.03	-0.02
speed_max	-0.0	-0.03	-0.04	-0.01	-0.58	0.13	-0.27	1.0	-0.02	-0.05	-0.14	0.01	0.04	0.01
user_ride_quality_median	0.03	-0.02	-0.03	0.02	-0.01	-0.0	-0.01	-0.02	1.0	-0.0	-0.03	0.03	0.01	0.01
user_uniq	0.0	-0.08	-0.07	-0.06	0.04	0.05	0.01	-0.05	-0.0	1.0	-0.06	-0.0	-0.0	-0.04
sum_rating	-0.01	0.97	0.98	0.05	0.18	-0.01	0.07	-0.14	-0.03	-0.06	1.0	-0.57	0.04	0.22
year_in_work	-0.0	-0.59	-0.59	0.77	-0.0	-0.0	-0.01	0.01	0.03	-0.0	-0.57	1.0	-0.12	-0.13
mean_distance	-0.02	0.05	0.05	-0.11	-0.04	0.22	-0.03	0.04	0.01	-0.0	0.04	-0.12	1.0	0.88
riders_per_year	-0.03	0.23	0.24	0.02	-0.02	0.01	-0.02	0.01	0.01	-0.04	0.22	-0.13	0.88	1.0

Таблица 3: Корреляции численных признаков

model	MAE	tune_time_sec	train_time_sec	predict_time_sec
RandomForest	9.0785	38.24	0.37	0.0036
CatBoost	9.2157	21.43	0.12	0.0006
XGBoost	9.2172	29.05	0.23	0.0017
LightGBM	9.2627	30.95	0.34	0.0015
DecisionTree	9.6036	0.58	0.01	0.0008
SVR_RBF	10.628	5.01	0.08	0.0296
PolynomialRegression	10.7004	5.79	0.02	0.0006
LinearRegression	11.7948	0.13	0.0	0.0
SVR_Linear	11.9224	12.55	0.05	0.0081

Таблица 4: Метрики и замеры времени для регрессии

мальных гиперпараметрах: n_estimators = 54, max_depth = 8, min_samples_split = 16, min_samples_leaf = 1.

Время подбора составило 38.24 с, обучение — 0.37 с, предсказание — 0.0036 с. Близкие результаты показал CatBoost (MAE = 9.2157), имеющий значительно меньшее время подбора и предсказания. Модели LightGBM и XGBoost также продемонстрировали высокое качество (MAE = 9.2627 и 9.2172 соответственно), подтверждая эффективность бустинговых методов на деревьях решений. Базовые линейные модели и SVR с линейным ядром показали значительно худшие результаты (MAE > 10.5), что говорит о выраженной нелинейности зависимости в данных. Модели SVR с RBF и полиномиальная регрессия улучшили качество, но уступили деревьям решений и ансамблям.

Лучшей моделью для задачи была выбрана Random Forest, поскольку она обеспечивает:

- Минимальную ошибку MAE среди всех моделей
- Устойчивость к шуму и выбросам благодаря ансамблированию
- Отсутствие необходимости масштабирования признаков, что упрощает весь конвейер обучения
- Хороший баланс между качеством, скоростью обучения и скоростью предсказания

CatBoost и LightGBM показали сопоставимое качество и превосходят Random Forest по скорости, однако небольшое преимущество Random Forest по точности оказалось критичным при выборе итоговой модели.

3. Бинарная классификация

В задаче бинарной классификации был проведён сравнительный анализ девяти моделей машинного обучения: Logistic Regression, Polynomial Logistic Regression, SVM (линейное и RBF ядро), Decision Tree, Random Forest, XGBoost, CatBoost и LightGBM. Для всех моделей был выполнен подбор гиперпараметров методом байесовской оптимизации с использованием библиотеки Optuna, основанной на алгоритме TPE (Tree-structured Parzen Estimator). Также измерены время подбора, обучения и предсказания. Основной метрикой качества являлся F1-score.

Наилучшее качество показала модель DecisionTreeClassifier, достигнув F1-score = 0.9543 и оптимальных гиперпараметров: max_depth = 5, min_samples_split = 14, min_samples_leaf = 8, criterion = entropy. Время подбора составило 0.46 с, обучение — 0.01 с, предсказание — 0.0005 с. Близкие результаты показали CatBoost (0.9498), Random Forest (0.9476), XGBoost (0.9476) и LightGBM (0.9474), что подтверждает эффективность ансамблей деревьев решений для бинарной классификации. Логистиче-

класс	0	1
0	239	16
1	4	209

Таблица 5: Матрица ошибок для бинарной классификации

Класс	Precision	Recall	F1-score	Support
0	0.98	0.94	0.96	255
1	0.93	0.98	0.95	213

Таблица 6: Метрики бинарной классификации для DecisionTreeClassifier

ская регрессия и линейное SVM показали худшие результаты, что свидетельствует о выраженной нелинейной структуре данных.

4. Многоклассовая классификация

Была также проведена серия экспериментов для сравнения эффективности различных алгоритмов машинного обучения в задаче многоклассовой классификации. Оценивались следующие характеристики моделей:

- качество классификации по метрике F1-macro;
- время подбора гиперпараметров (с использованием Optuna);
- время обучения;
- время предсказания;
- оптимальные параметры после тюнинга

Все гиперпараметры подбирались методом bayesian optimization (TPE — Tree-structured Parzen Estimator) через Optuna.

CatBoost показал наилучшее качество F1-macro = 0.796, при умеренном времени обучения, поэтому является лучшей моделью для текущей постановки задачи. RandomForest также продемонстрировал высокое качество при значительно меньших вычислительных затратах. XGBoost - третий по качеству, но с высокой стоимостью тюнинга. Линейные модели работают быстро, но уступают ансамблям и бустингу. LightGBM оказался наиболее тяжёлым по времени вычислений при сравнительно скромном результате.

Таким образом, если критерием является качество, предпочтение отдаётся CatBoost или RandomForest.

model	f1_score	tune_time_sec	train_time_sec	predict_time_sec
DecisionTree	0.9543	0.46	0.01	0.0005
CatBoost	0.9498	23.36	0.17	0.0006
RandomForest	0.9476	14.53	0.2	0.0055
XGBoost	0.9476	14.54	0.1	0.0016
LightGBM	0.9474	59.18	0.84	0.0036
PolynomialRegression	0.912	2.45	0.03	0.0005
SVM_RBF	0.9025	3.02	0.05	0.0092
SVM_Linear	0.771	22.63	0.03	0.0045
LogisticRegression	0.7661	0.31	0.01	0.0001

Таблица 7: Метрики и замеры времени для бинарной классификации

model	f1_score	tune_time_sec	train_time_sec	predict_time_sec
CatBoost	0.796	122.73	1.11	0.0008
RandomForest	0.7885	16.65	0.45	0.0104
XGBoost	0.784	118.22	0.67	0.0024
DecisionTree	0.7821	0.63	0.01	0.0015
LogisticRegression	0.7754	1.65	0.02	0.0002
SVM_RBF	0.7718	5.35	0.04	0.0221
LightGBM	0.7667	490.63	21.73	0.0321
SVM_Linear	0.7602	11.95	0.04	0.0092
PolynomialRegression	0.7432	27.4	0.17	0.0006

Таблица 8: Метрики и замеры времени для многоклассовой классификации

Класс	Precision	Recall	F1-score	Support
0	0.88	0.86	0.87	58
1	1.00	1.00	1.00	56
2	1.00	1.00	1.00	51
3	0.76	0.84	0.80	49
4	0.74	0.64	0.69	58
5	0.74	0.43	0.54	58
6	0.45	0.77	0.57	48
7	0.98	0.92	0.95	51
8	0.83	0.77	0.80	39

Таблица 9: Метрики многоклассовой классификации для CatBoost

5. Заключение

В ходе работы были изучены и сравнены девять алгоритмов машинного обучения по трём независимым задачам: регрессии, бинарной классификации и многоклассовой классификации. Для всех моделей был выполнен подбор гиперпараметров методом байесовской оптимизации, проведена оценка качества, а также измерены времена обучения и предсказания.

Лучшие модели для каждой постановки следующие:

- **Регрессия**: Random Forest — MAE = 9.0785
- **Бинарная классификация**: DecisionTreeClassifier — F1 = 0.9543
- **Многоклассовая классификация**: CatBoost — F1-macro = 0.796

Полученные результаты демонстрируют эффективность нелинейных деревообразных алгоритмов при работе с данными транспортного автопарка, а также оправдывают использование ансамблевых методов и продвинутой оптимизации гиперпараметров.

Код: https://github.com/bedanar/ml_proj_spbu/blob/main/project_final_v1.ipynb