Классификация цены дома (California Housing Prices)

Люзина Мария

М8О-307Б-23

Задача многоклассовой классификации

Исходные данные:

- 20 640 записей о домах Калифорнии (1990 г.)
- 9 числовых и 1 категориальный признак
- Целевая переменная – median_house_value

Задача:

Предсказать один их трех классов стоимости: low (< 150 000$), medium (150 000 – 300 000$), high (> 300 000$).

Классы сформированы по квантилям 33%/66%, сбалансированы

Базовая статистика датасета до обработки

	count	mean	std	min	25%	50%	75%	max
longitude	20640.0	-119.569704	2.003532	-124.3500	-121.8000	-118.4900	-118.01000	-114.3100
latitude	20640.0	35.631861	2.135952	32.5400	33.9300	34.2600	37.71000	41.9500
housing_median_age	20640.0	28.639486	12.585558	1.0000	18.0000	29.0000	37.00000	52.0000
total_rooms	20640.0	2635.763081	2181.615252	2.0000	1447.7500	2127.0000	3148.00000	39320.0000
total_bedrooms	20640.0	536.838857	419.391878	1.0000	297.0000	435.0000	643.25000	6445.0000
population	20640.0	1425.476744	1132.462122	3.0000	787.0000	1166.0000	1725.00000	35682.0000
households	20640.0	499.539680	382.329753	1.0000	280.0000	409.0000	605.00000	6082.0000
median_income	20640.0	3.870671	1.899822	0.4999	2.5634	3.5348	4.74325	15.0001
median_house_value	20640.0	206855.816909	115395.615874	14999.0000	119600.0000	179700.0000	264725.00000	500001.0000

Предобработка

- Избавление от выбросов в целевой переменной
- Заполнение пропусков медианой в total_rooms.

Feature Engineering

Простые фичи

- Среднее количество комнат на домохозяйство
- Среднее количество спален на домохозяйство
- Плотность населения в домохозяйстве

```
df['rooms_per_household'] = df['total_rooms'] / df['households']
df['bedrooms_per_household'] = df['total_bedrooms'] / df['households']
df['population_per_household'] = df['population'] / df['households']
```

Feature Engineering

Фичи на основе медианного дохода

- Логарифм дохода для нормального распределения ключевого предиктора
- Разделение на категории дохода

```
df['log_median_income'] = np.log1p(df['median_income'])
df['income_category'] = pd.qcut(df['median_income'], q=3, labels=['low_income','medium_income','high_income'])
df= pd.get_dummies(df, columns=['income_category'], prefix='income_category')
bool_cols = [col for col in df.columns if col.startswith('income_category_')]
df[bool_cols] = df[bool_cols].astype(int)
```

Feature Engineering

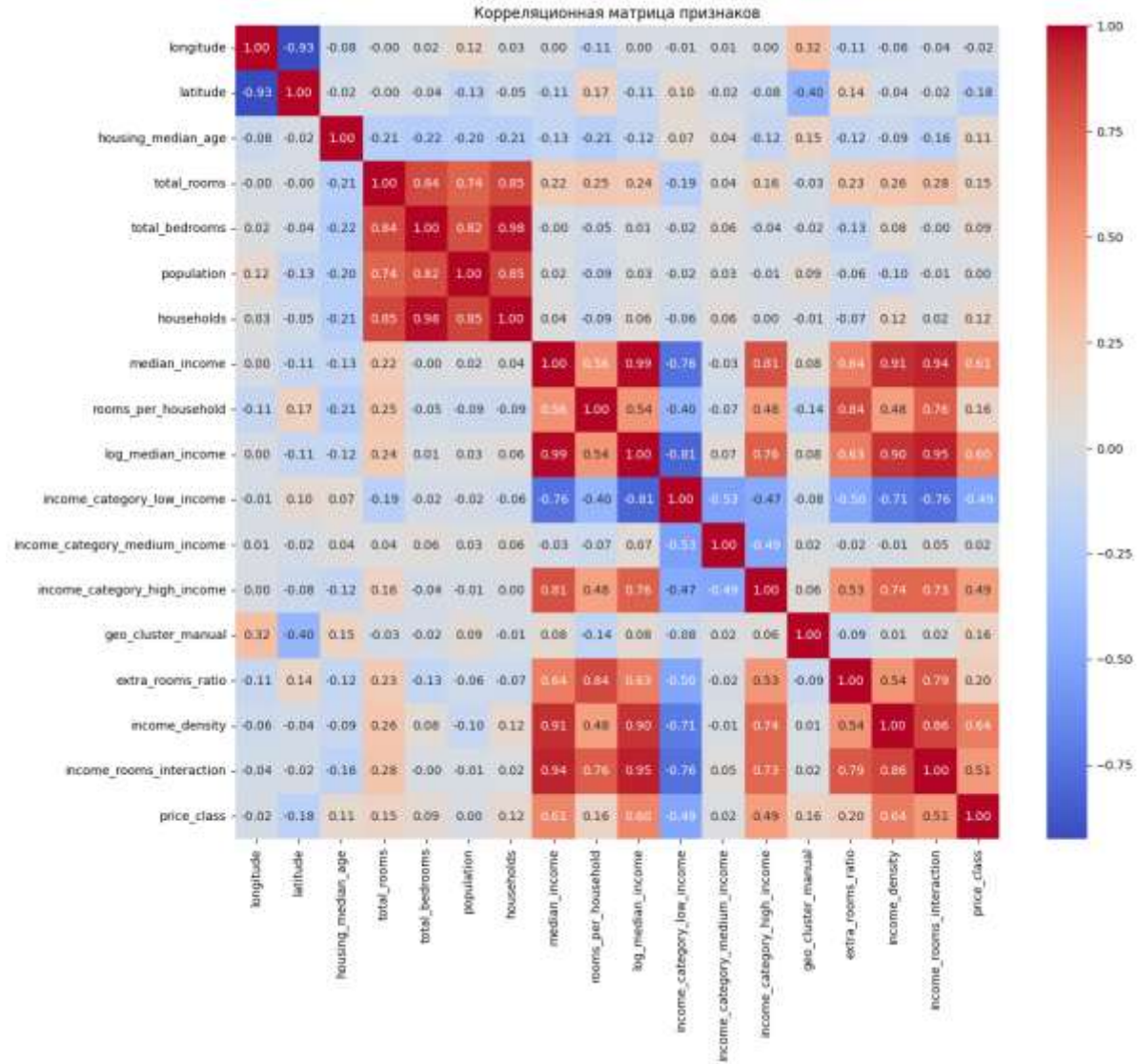
- Дифицит/избыток комнат
- Влияние плотности населения и дохода
- Полиномиальные взаимодействия

```
df['extra_rooms_ratio'] = (df['total_rooms'] - df['total_bedrooms']) / df['total_rooms']
df['income_density'] = df['median_income'] / (df['population_per_household'] + 1)
df['income_rooms_interaction'] = df['median_income'] * df['rooms_per_household']
```

Базовая статистика датасета после обработки

	count	mean	std	min	25%	50%	75%	max
longitude	19569.0	-119.562786	2.005764	-124.350000	-121.760000	-118.510000	-117.990000	-114.310000
latitude	19569.0	35.654159	2.151007	32.540000	33.930000	34.270000	37.730000	41.950000
housing_median_age	19569.0	28.352752	12.497772	1.000000	18.000000	28.000000	37.000000	52.000000
total_rooms	19569.0	2619.977260	2183.419302	2.000000	1438.000000	2110.000000	3123.000000	39320.000000
total_bedrooms	19569.0	538.821299	420.617106	2.000000	299.000000	435.000000	645.000000	6445.000000
population	19569.0	1442.788952	1145.011369	3.000000	797.000000	1181.000000	1749.000000	35682.000000
households	19569.0	501.394859	383.396308	2.000000	282.000000	411.000000	606.000000	6082.000000
median_income	19569.0	3.665568	1.557927	0.499900	2.522700	3.441200	4.572100	15.000100
median_house_value	19569.0	190852.301906	95438.555669	14999.000000	116200.000000	173200.000000	246700.000000	482200.000000
rooms_per_household	19569.0	5.357548	2.294996	0.846154	4.413567	5.181818	5.965142	132.533333
bedrooms_per_household	19569.0	1.101096	0.502175	0.121204	1.005464	1.048780	1.100000	34.066667
population_per_household	19569.0	3.098760	10.660526	0.692308	2.448193	2.839009	3.307692	1243.333333
log_median_income	19569.0	1.486294	0.329512	0.405398	1.259228	1.490925	1.717772	2.772595
income_category_low_income	19569.0	0.333333	0.471417	0.000000	0.000000	0.000000	1.000000	1.000000
income_category_medium_income	19569.0	0.333742	0.471561	0.000000	0.000000	0.000000	1.000000	1.000000
income_category_high_income	19569.0	0.332925	0.471272	0.000000	0.000000	0.000000	1.000000	1.000000
geo_cluster_manual	19569.0	2.719403	1.471490	0.000000	2.000000	3.000000	4.000000	5.000000
extra_rooms_ratio	19569.0	0.784123	0.064651	-1.824675	0.758681	0.795326	0.822378	0.962849
income_density	19569.0	0.960245	0.433428	0.008218	0.633345	0.913785	1.225132	4.814847
income_rooms_interaction	19569.0	20.795603	14.828568	0.856971	11.414449	17.465217	26.393909	612.966667

Матрица корреляции признаков



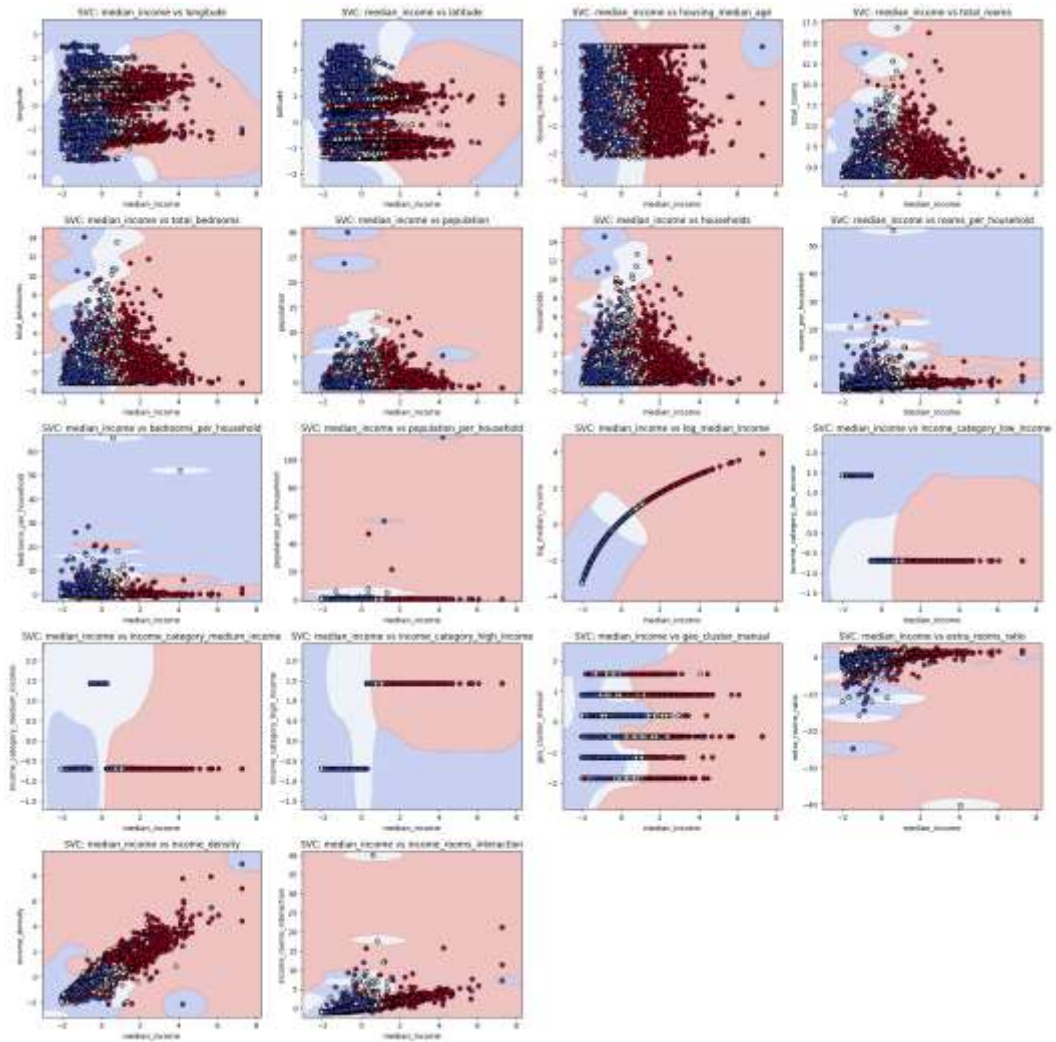
Метрики моделей

```
=== SVM ===
Accuracy: 0.742 ± 0.004
F1-score: 0.742 ± 0.004
=== Logistic Regression ===
Accuracy: 0.721 ± 0.002
F1-score: 0.721 ± 0.002
ROC-AUC: 0.877 ± 0.002
=== KNN ===
Accuracy: 0.710 ± 0.005
F1-score: 0.710 ± 0.004
ROC-AUC: 0.857 ± 0.002
```

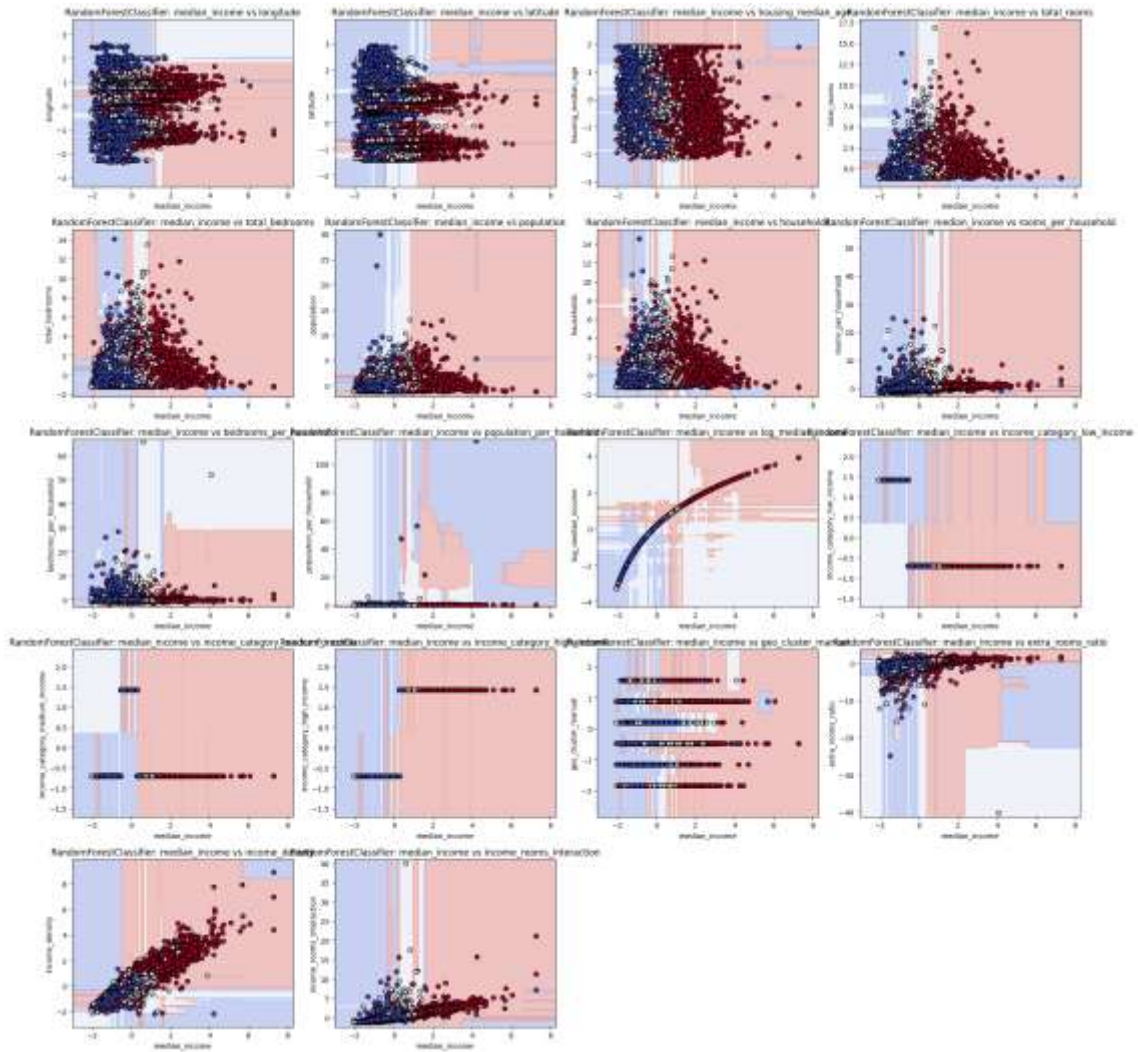
```
=== Boosting ===
Accuracy: 0.783 ± 0.006
F1-score: 0.782 ± 0.006
ROC-AUC: 0.924 ± 0.002
=== XGBoost ===
Accuracy: 0.773 ± 0.006
F1-score: 0.772 ± 0.007
ROC-AUC: 0.917 ± 0.003
=== CatBoost ===
Accuracy: 0.816 ± 0.003
F1-score: 0.816 ± 0.003
ROC-AUC: 0.942 ± 0.002
```

```
=== DesicionTree ===
Accuracy: 0.730 ± 0.008
F1-score: 0.731 ± 0.008
ROC-AUC: 0.798 ± 0.006
=== RandomForest ===
Accuracy: 0.778 ± 0.003
F1-score: 0.777 ± 0.003
ROC-AUC: 0.921 ± 0.002
```

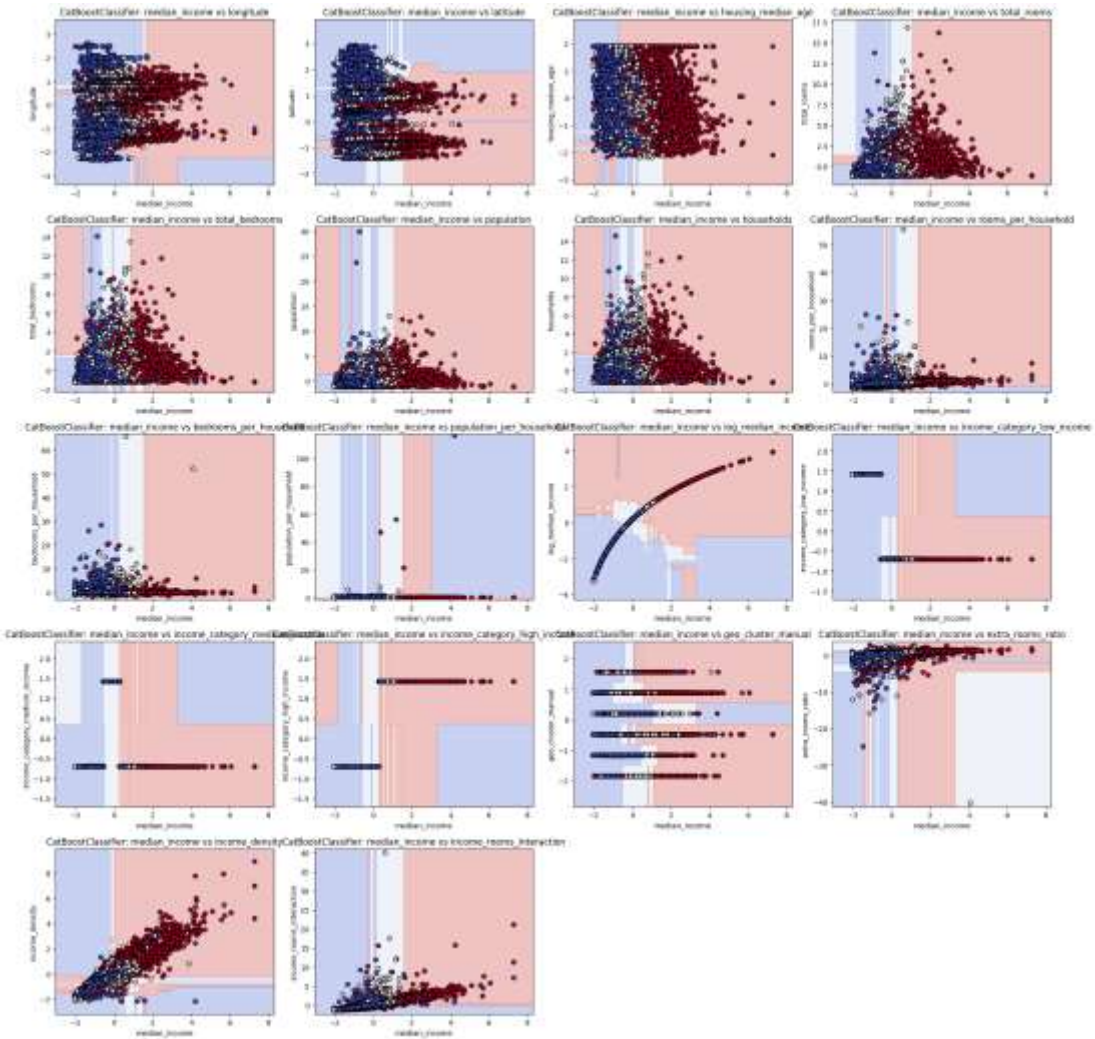
SVC-графики



SVC-графики



SVC-графики



Лучшая модель - CatBoost

Преимущества этой модели:

- Нативная обработка категориальных признаков
- Встроенная защита от переобучения
- Высокая точность
- Поддержка GPU/CPU без изменений кода

Вывод

В рамках лабораторной работы была решена задача многоклассовой классификации на датасете California Housing (20 640 объектов) с преобразованием непрерывной целевой переменной median_house_value в три сбалансированных класса: low, medium, high (по квантилям 33 % и 66 %).

Выполнены этапы:

- Исследовательский анализ данных
- Проектирование признаков
- Обучение и сравнение моделей