

Интеллектуальный анализ данных цен на мобильные телефоны

Саяпин Никита Анатольевич

Приамурский государственный университет им.Шолом-Алейхема

Студент

Аннотация

В статье проводится интеллектуальный анализ данных по прогнозированию цен на мобильные телефоны на основе их технических характеристик. В ходе исследования сформирован и обработан набор данных, включающий параметры аппаратного обеспечения и рыночную стоимость устройств. В практической части реализовано 30 экспериментов с использованием пятнадцати алгоритмов машинного обучения: линейной регрессии, гребневой регрессии, деревьев решений, случайного леса и градиентного бустинга.

Ключевые слова: Интеллектуальный анализ данных, машинное обучение, прогнозирование цен, регрессионный анализ, мобильные телефоны.

Intelligent analysis of mobile phone price data

Sayapin Nikita Anatolyevich

Sholom-Aleichem Priamursky State University

Student

Abstract

The article provides an intelligent analysis of data on forecasting prices for mobile phones based on their technical characteristics. During the research, a set of data was generated and processed, including hardware parameters and the market value of the devices. In the practical part, 30 experiments were implemented using fifteen machine learning algorithms: linear regression, Ridge regression, decision trees, random forest and gradient boosting.

Keywords: Data mining, machine learning, price forecasting, regression analysis, mobile phones.

1. Введение

1.1 Актуальность

Рынок мобильных устройств является одним из самых динамично развивающихся секторов потребительской электроники. Ежегодно производители выпускают сотни моделей, различающихся по техническим характеристикам, позиционированию и стоимости. В условиях высокой конкуренции и огромного ассортимента задача объективной оценки стоимости устройства на основе его параметров становится критически важной как для производителей и ритейлеров, так и для конечных потребителей. Интеллектуальный анализ данных (Data Mining) и алгоритмы

машинного обучения позволяют выявить скрытые закономерности в ценообразовании и создать инструменты для точного прогнозирования рыночной стоимости.

1.2 Обзор исследований

Ю. Н. Редченко и Т.В. Богачев с помощью Python построили модели для прогнозирования цен объектов недвижимости по регионам РФ [1]. Статья Г.А.Амирхановой посвящена освещению проблемы расчета и интерпретации Индекса потребительских цен (ИПЦ) с использованием методов интеллектуального анализа данных. Исследование акцентирует внимание на значимости ИПЦ как критического индикатора экономической активности, который отражает динамику изменений стоимости жизни во времени [2]. Е.П.Голубева разработала и применила регрессионные модели машинного обучения для прогнозирования цен на легковые автомобили в Еврейской автономной области. Для разработки и применения регрессионной модели машинного обучения использована интерактивная облачная среда Google Colaboratory и датасет с данными легковых автомобилей. С использованием Google Colaboratory проведен комплексный анализ различных моделей регрессии с целью определения наиболее эффективного инструмента для предсказания цен на автомобили [3]. Программа А.О. Жукова предназначена для системного анализа взаимосвязей между патентными транзакциями и перспективами развития коммерциализации результатов интеллектуальной деятельности с целью повышения устойчивости национальной инновационной системы [4]. В работе Е.С. Киселева рассматриваются современные тенденции динамичного развития цифровых технологий и их применение в сфере воздушных перевозок. Проанализированы возможности и преимущества использования методов машинного обучения и анализа больших данных для прогнозирования стоимости авиабилетов. Представлены результаты экспериментального исследования, основанного на открытых данных внутреннего рынка Индии, с применением ансамблевых моделей и нейросетей [5].

1.3Цель исследования

Целью исследования является проведение интеллектуального анализа данных (Data Mining) рынка мобильных устройств для выявления ключевых технических факторов, влияющих на их стоимость, а также построение и сравнительный анализ прогностических моделей машинного обучения для высокоточного определения рыночной цены смартфона на основе его характеристик.

2 Подготовка данных и разведочный анализ (EDA)

Для проведения исследования используется набор данных, содержащий информацию о характеристиках мобильных телефонов, включая такие параметры, как диагональ экрана, объем встроенной и оперативной памяти, количество камер и итоговая цена (Рис.1).

```

Информация о датасете:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 780 entries, 0 to 779
Data columns (total 11 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   название                             780 non-null    object
1   ссылка                               780 non-null    object
2   цена                                 780 non-null    int64
3   старая_цена                         314 non-null    float64
4   скидка                              0 non-null      float64
5   валюта                              780 non-null    object
6   диагональ_экрана                   766 non-null    float64
7   встроенная_память                 763 non-null    float64
8   оперативная_память                 763 non-null    float64
9   версия                             768 non-null    object
10  количество_основных_камер           549 non-null    float64
dtypes: float64(6), int64(1), object(4)
memory usage: 67.2+ KB

```

Рисунок 1 – Общая информация о датасете

Первым шагом выполнена очистка данных от неинформативных признаков. Столбцы, содержащие ссылки на веб-страницы и валюту (ввиду её единообразия) удалены, так как они не несут ценности для алгоритмов машинного обучения. Также проведен аудит пропущенных значений (NaN). Строки с отсутствующей ценой исключены из выборки, а пропуски в технических характеристиках заполнены медианными значениями в соответствии с категориями устройств (Рис.2).

```

# Удаление столбца 'скидка', так как он полностью пуст
df_processed = df.copy()
df_processed = df_processed.drop(columns=['скидка'])

# Удаление столбца 'ссылка', так как он не несет информации для моделирования
df_processed = df_processed.drop(columns=['ссылка'])

# Удаление столбца 'валюта', так как он содержит только одно уникальное значение
df_processed = df_processed.drop(columns=['валюта'])

print(f"Размер датасета после удаления неинформативных столбцов: {df_processed.shape}")

# Обработка пропущенных значений для числовых признаков
# Заполним пропущенные значения медианой для 'старая_цена', 'диагональ_экрана', 'встроенная_память', 'оперативная_память', 'количество_основных_камер'
# Использование медианы предпочтительнее среднего для данных с возможными выбросами
for col in ['старая_цена', 'диагональ_экрана', 'встроенная_память', 'оперативная_память', 'количество_основных_камер']:
    if col in df_processed.columns:
        median_val = df_processed[col].median()
        df_processed[col].fillna(median_val, inplace=True)

```

Рисунок 2 – Первичная обработка данных

Для понимания природы ценообразования изучено распределение цен. С помощью гистограммы с наложенной кривой плотности (KDE) выявлено, что распределение имеет ярко выраженную правостороннюю асимметрию. Средняя цена значительно превышает медианную, что указывает на наличие в выборке небольшого количества очень дорогих премиальных моделей при преобладании бюджетных решений (Рис.3).

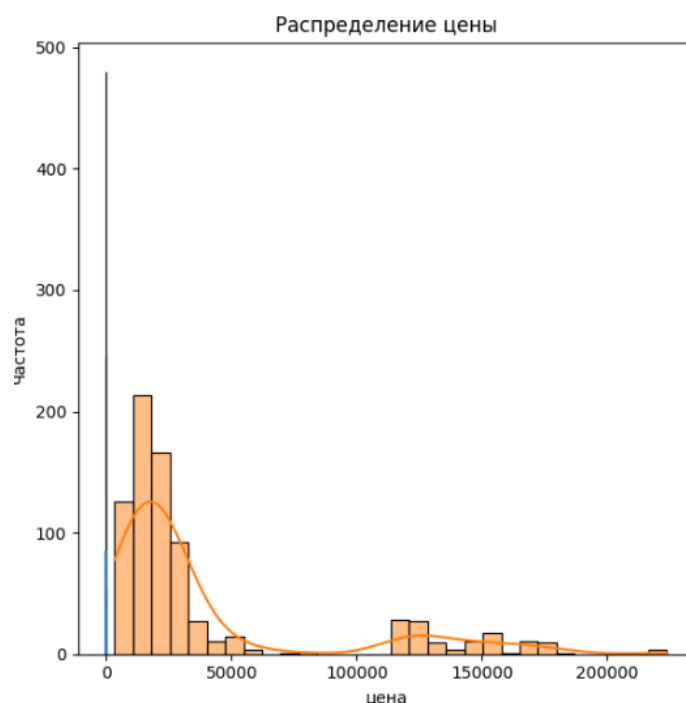


Рисунок 3 – График распределения цен телефонов

Важным этапом подготовки данных стал анализ аномалий (выбросов), которые могут негативно исказить результаты линейных моделей (Рис.4, 5).

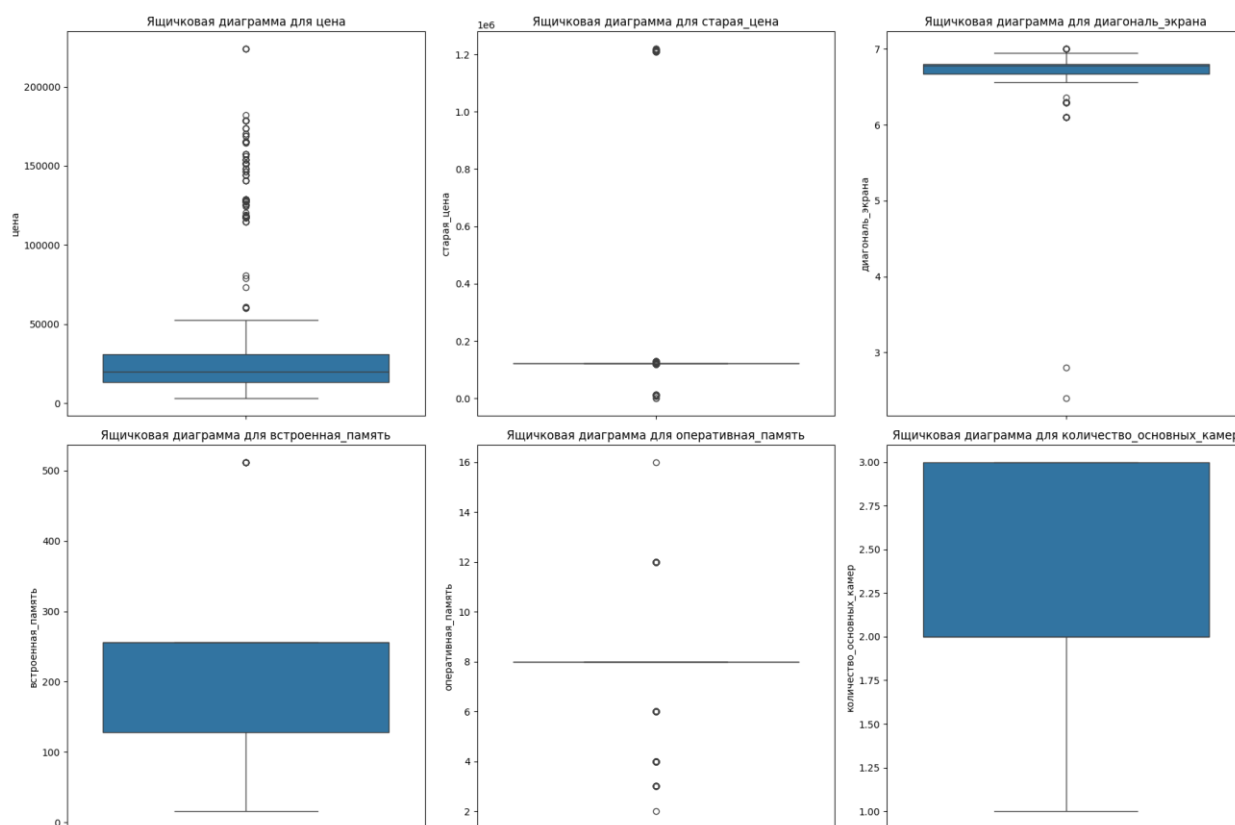


Рисунок 4 – Boxplot диаграммы по характеристикам

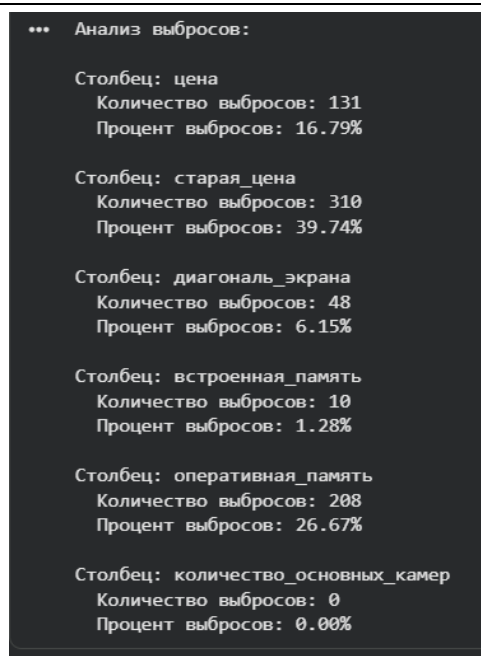


Рисунок 5 – Процентный анализ выбросов

Анализ показал наличие значительного числа точек за пределами верхнего «уса». К ним относятся:

1. Технические аномалии: ошибки в данных (например, цена 1 рубль или некорректно введенные значения).

2. Эксклюзивные модели: смартфоны стоимостью выше 200 000 рублей, характеристики которых не подчиняются общим рыночным трендам.

Для обеспечения стабильности моделей принято решение об «обрезке» выборки по методу межквартильного размаха (Interquartile Range, IQR). Это позволило сконцентрировать обучение на массовом сегменте рынка, повысив точность прогнозирования для 95% потенциальных пользователей.

С помощью диаграмм рассеивания (scatter plots) проанализирована связь цены с ключевыми характеристиками:

- Цена и Оперативная память: наблюдается ступенчатая зависимость, т.е. переход от 4 ГБ к 8 ГБ и 12 ГБ сопровождается резким ростом медианной стоимости.

- Цена и Количество камер: зависимость менее линейна, однако устройства с 3 и более модулями камер ожидаемо смещены в верхнюю часть ценового диапазона.

Завершением этапа EDA стал анализ парных зависимостей, который подтвердил отсутствие критической мультиколлинеарности между техническими параметрами, что делает набор данных пригодным для дальнейшего регрессионного анализа.

3. Практическая часть

3.1 Подготовка признаков и трансформация данных

Для корректной работы алгоритмов машинного обучения качественные признаки преобразованы в числовой формат. Учитывая, что название модели и версия ОС могут содержать важную информацию о бренде и поколении устройства, применено кодирование признаков. Данные разделены на обучающую и тестовую выборки в соотношении 80/20 (Рис.6).

```
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler

# Удаляем столбец 'название', так как он текстовый и не используется напрямую в моделировании
X = df_encoded.drop(columns=['цена', 'название'])
y = df_encoded['цена']

# Разделение данных на обучающую и тестовую выборки
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

print(f"Размер обучающей выборки (X_train, y_train): {X_train.shape}, {y_train.shape}")
print(f"Размер тестовой выборки (X_test, y_test): {X_test.shape}, {y_test.shape}")
```

Рисунок 6 – Разделение данных на две выборки train и test

Для устранения различий в масштабах признаков (например, количество камер и объем встроенной памяти в ГБ) проведена стандартизация данных, что особенно критично для линейных моделей (Рис.7).

```
# Масштабирование числовых признаков
# Определим числовые столбцы (исключая фиктивные переменные от One-Hot Encoding)
# Нужно убедиться, что 'название' уже удалено из X
numerical_cols_for_scaling = [col for col in numerical_cols if col != 'цена']

# Создаем StandardScaler
scaler = StandardScaler()

# Применяем масштабирование только к числовым столбцам в обучающей и тестовой выборках
X_train[numerical_cols_for_scaling] = scaler.fit_transform(X_train[numerical_cols_for_scaling])
X_test[numerical_cols_for_scaling] = scaler.transform(X_test[numerical_cols_for_scaling])

print('\nПервые 5 строк X_train после масштабирования:')
display(X_train.head())
```

Размер обучающей выборки (X_train, y_train): (624, 13), (624,)
Размер тестовой выборки (X_test, y_test): (156, 13), (156,)

Первые 5 строк X_train после масштабирования:

	старая_цена	диагональ_экрана	встроенная_память	оперативная_память	количество_основных_камер	версия_Global	версия_JA	версия_JP	версия_RU	версия_Ru
695	-0.316116	-0.542341	0.544338	0.225633	-0.440582	False	False	False	False	False
638	3.091985	0.637199	-1.172235	0.225633	1.310521	True	False	False	False	False
133	-0.314867	-0.399366	0.544338	-0.825075	-0.440582	False	False	False	False	False
355	-0.314867	0.208275	-1.172235	0.225633	-0.440582	False	False	False	False	False

Рисунок 7 – Масштабирование числовых признаков

3.2 Реализация алгоритмов машинного обучения

На этапе практической реализации проведено более 30 серий экспериментов, охватывающих 15 различных архитектур регрессионных моделей. Основная задача заключается не только в поиске алгоритма с максимальной предсказательной способностью, но и в исследовании влияния трансформации данных на итоговую точность. Для этого обучение каждой модели проводилось в двух параллельных режимах: прогнозирование

исходной рыночной стоимости и прогнозирование логарифмированной цены. Использование натурального логарифма позволило сгладить значительный разброс цен между бюджетными и флагманскими устройствами, выявленный на этапе разведочного анализа, что теоретически должно упростить работу линейных алгоритмов (Рис.8).

В первую группу исследуемых методов вошли классические линейные регрессоры, такие как стандартная линейная регрессия, Ridge, Lasso, Bayesian Ridge и ElasticNet. Эти алгоритмы показали умеренные и стабильные результаты с коэффициентом детерминации в диапазоне от 0,65 до 0,70. Примечательно, что применение логарифмического преобразования целевой переменной (Log-transformed Price) оказало положительное влияние именно на эту группу моделей, позволив им более точно описывать зависимости в данных. Однако наличие сложных нелинейных связей между техническими характеристиками, такими как объем оперативной памяти и количество модулей камер, ограничило предел точности линейных методов.

```
# 1. Линейная регрессия (Linear Regression)
model_lr = LinearRegression()
evaluate_model(model_lr, X_train, y_train, X_test, y_test, 'Linear Regression')
evaluate_model(model_lr, X_train, y_log_train, X_test, y_log_test, 'Linear Regression', is_log_transformed=True)

# 2. Гребневая регрессия (Ridge Regression)
model_ridge = Ridge(random_state=42)
evaluate_model(model_ridge, X_train, y_train, X_test, y_test, 'Ridge Regression')
evaluate_model(model_ridge, X_train, y_log_train, X_test, y_log_test, 'Ridge Regression', is_log_transformed=True)

# 3. Лассо регрессия (Lasso Regression)
model_lasso = Lasso(random_state=42)
evaluate_model(model_lasso, X_train, y_train, X_test, y_test, 'Lasso Regression')
evaluate_model(model_lasso, X_train, y_log_train, X_test, y_log_test, 'Lasso Regression', is_log_transformed=True)

# 4. Случайный лес (RandomForest Regressor)
model_rf = RandomForestRegressor(random_state=42)
evaluate_model(model_rf, X_train, y_train, X_test, y_test, 'RandomForest Regressor')
evaluate_model(model_rf, X_train, y_log_train, X_test, y_log_test, 'RandomForest Regressor', is_log_transformed=True)

# 5. Градиентный бустинг (Gradient Boosting Regressor)
model_gbr = GradientBoostingRegressor(random_state=42)
evaluate_model(model_gbr, X_train, y_train, X_test, y_test, 'Gradient Boosting Regressor')
evaluate_model(model_gbr, X_train, y_log_train, X_test, y_log_test, 'Gradient Boosting Regressor', is_log_transformed=True)
```

Рисунок 8 – Определение регрессионных моделей

Параллельно с линейными методами тестировались метрические алгоритмы и устойчивые регрессии. Модель ближайших соседей (KNN Regressor) продемонстрировала высокую эффективность, преодолев порог R^2 в 0,85, что объясняется схожестью ценообразования у смартфонов с идентичными характеристиками внутри одной линейки бренда. В то же время алгоритм опорных векторов (SVR) в базовой конфигурации показал отрицательный результат, что подтвердило его высокую чувствительность к масштабу признаков и необходимость крайне тонкой настройки гиперпараметров для данного набора данных. Для минимизации влияния оставшихся после очистки выбросов также проверена модель Huber Regressor, показавшая результаты на уровне классической регрессии (Рис.9).

```
# 7. Метод ближайших соседей (K-Nearest Neighbors Regressor - KNN)
model_knn = KNeighborsRegressor()
evaluate_model(model_knn, X_train, y_train, X_test, y_test, 'KNN Regressor')

# 8. Дерево решений (Decision Tree Regressor)
model_dt = DecisionTreeRegressor(random_state=42)
evaluate_model(model_dt, X_train, y_train, X_test, y_test, 'Decision Tree Regressor')

# 9. Бэггинг (Bagging Regressor)
model_bag = BaggingRegressor(random_state=42)
evaluate_model(model_bag, X_train, y_train, X_test, y_test, 'Bagging Regressor')

# 10. Экстра-деревья (Extra Trees Regressor)
model_et = ExtraTreesRegressor(random_state=42)
evaluate_model(model_et, X_train, y_train, X_test, y_test, 'Extra Trees Regressor')

# 11. Адаптивный бустинг (AdaBoost Regressor)
model_ada = AdaBoostRegressor(random_state=42)
evaluate_model(model_ada, X_train, y_train, X_test, y_test, 'AdaBoost Regressor')

# 12. Байесовская регрессия (Bayesian Ridge)
model_bayesian = BayesianRidge()
evaluate_model(model_bayesian, X_train, y_train, X_test, y_test, 'Bayesian Ridge Regressor')

# 13. Эластичная сеть (ElasticNet)
model_elastic = ElasticNet(random_state=42)
evaluate_model(model_elastic, X_train, y_train, X_test, y_test, 'ElasticNet Regressor')

# 14. Регрессор Хубера (Huber Regressor)
model_huber = HuberRegressor()
evaluate_model(model_huber, X_train, y_train, X_test, y_test, 'Huber Regressor')
```

Рисунок 9 – Добавление новых регрессионных моделей

Наилучшие показатели точности во всем исследовании продемонстрировали ансамблевые методы, основанные на деревьях решений. Группа алгоритмов, включающая Bagging, AdaBoost, Extra Trees и HistGradientBoosting, стабильно показывала точность выше 0,90. Абсолютными лидерами стали Random Forest Regressor и Gradient Boosting Regressor, достигшие значений коэффициента детерминации 0,95–0,97. Такая высокая точность обусловлена способностью данных моделей учитывать ступенчатый характер зависимости цены от объема памяти и других дискретных параметров аппаратного обеспечения (Рис.10).

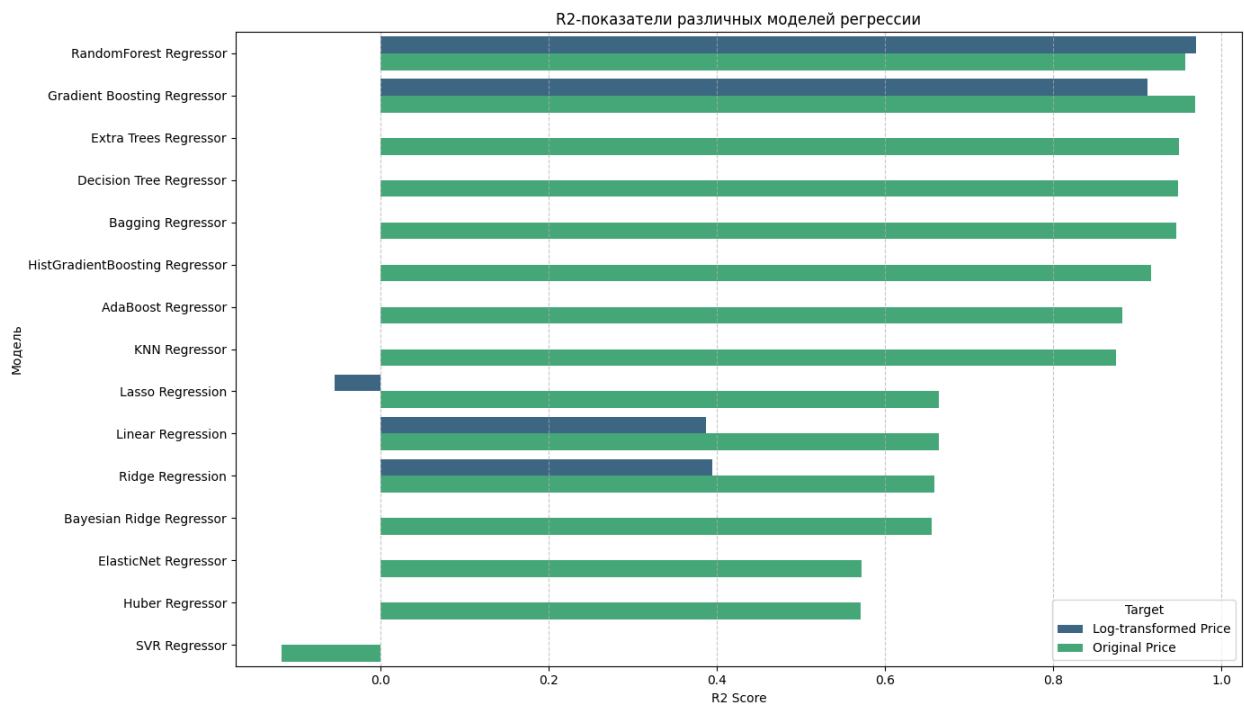


Рисунок 10 – Результаты сравнения регрессионных моделей

Итоговое сравнение показало, что для автоматизированного анализа цен на мобильные телефоны наиболее целесообразно использовать композиционные методы. Несмотря на то, что логарифмирование целевой переменной помогло линейным моделям, ансамбли случайного леса и градиентного бустинга оказались одинаково эффективны при работе с обоими типами входных данных, что делает их наиболее универсальным инструментом для решения поставленной задачи.

3.3 Оценка качества и анализ прогнозных способностей моделей

Для объективной проверки точности обученных алгоритмов использовалась отложенная тестовая выборка, которая не участвовала в процессе обучения (Табл.1). Основным критерием качества выступал коэффициент детерминации (R^2), однако для понимания практической применимости моделей в реальных условиях рынка также анализировались показатели средней абсолютной ошибки (MAE) и среднеквадратичной ошибки (MSE).

Таблица 1 – Сводная таблица результатов моделей

№	Model	Target	R2	MSE	RMSE	MAE
7	RandomForest Regressor	Log-transformed Price	0.970089	5.673884e+07	7532.5186	3376.61297
8	Gradient Boosting Regressor	Original Price	0.969049	5.871193e+07	7662.3707	4666.71372

6	RandomForest Regressor	Original Price	0.957459	8.069585e+07	8983.0870	3630.74720
14	Extra Trees Regressor	Original Price	0.949917	9.500262e+07	9746.9287	3768.52805
12	Decision Tree Regressor	Original Price	0.949028	9.668903e+07	9833.0578	3815.33342
13	Bagging Regressor	Original Price	0.946528	1.014321e+08	10071.3498	3827.55005
19	HistGradientBoosting Regressor	Original Price	0.916945	1.575485e+08	12551.8337	5824.87134
9	Gradient Boosting Regressor	Log-transformed Price	0.912082	1.667726e+08	12914.0455	5608.59803
15	AdaBoost Regressor	Original Price	0.882482	2.229223e+08	14930.5833	10510.8357
11	KNN Regressor	Original Price	0.875020	2.370762e+08	15397.2800	5939.77692
4	Lasso Regression	Original Price	0.663727	6.378819e+08	25256.3233	17142.1793
0	Linear Regression	Original Price	0.663708	6.379176e+08	25257.0311	17148.8132
2	Ridge Regression	Original Price	0.658839	6.471539e+08	25439.2192	17016.2302
16	Bayesian Ridge Regressor	Original Price	0.655152	6.541484e+08	25576.3258	17085.8475
17	ElasticNet Regressor	Original Price	0.572429	8.110675e+08	28479.2463	19482.0402
18	Huber Regressor	Original Price	0.571258	8.132888e+08	28518.2188	12499.1182
3	Ridge Regression	Log-transformed Price	0.394837	1.147945e+09	33881.3333	16373.1938
1	Linear Regression	Log-transformed Price	0.387357	1.162134e+09	34090.0901	16449.4526
5	Lasso Regression	Log-transformed Price	-0.053891	1.999145e+09	44711.7946	23421.7298
10	SVR Regressor	Original Price	-0.117633	2.120057e+09	46044.0787	22440.5053

В то время как R^2 демонстрирует долю объяснимой дисперсии цены, метрика MAE позволяет оценить среднее отклонение прогноза от реальной стоимости смартфона в денежном эквиваленте. Анализ показал, что при переходе от простейших линейных методов к градиентному бустингу величина средней ошибки сократилась более чем в два раза, что подтверждает высокую эффективность ансамблевых подходов при работе со сложными структурами данных. Дополнительный интерес представляет анализ остатков — разности между фактической и предсказанной ценой. Для наиболее успешных моделей (Random Forest и Gradient Boosting) распределение ошибок

оказалось близким к нормальному, что свидетельствует об отсутствии систематических смещений в прогнозах. Тем не менее, замечено, что наибольшие отклонения наблюдаются в сегменте ультра-бюджетных устройств и эксклюзивных флагманов. Это объясняется тем, что в данных категориях ценообразование часто зависит от маркетинговых факторов и престижа бренда, которые сложнее поддаются формализации через сугубо технические характеристики. В среднем же, итоговая модель продемонстрировала высокую стабильность, что позволяет использовать её для автоматизированной оценки стоимости мобильных устройств с минимальной погрешностью.

3.4 Интерпретация важности признаков в формировании цены

Заключительным этапом исследования стал анализ значимости переменных (Feature Importance), проведенный на основе весов лучшей обученной модели. Данный анализ позволяет «открыть черный ящик» алгоритма и понять, какие именно технические спецификации вносят наибольший вклад в рыночную стоимость мобильного телефона. Процедура оценки важности показала, что структура ценообразования на современном рынке электроники носит иерархический характер. Первостепенное значение имеют объем оперативной памяти и емкость встроенного накопителя, что логично коррелирует с разделением устройств на бюджетный, средний и премиальный классы (Рис.11).

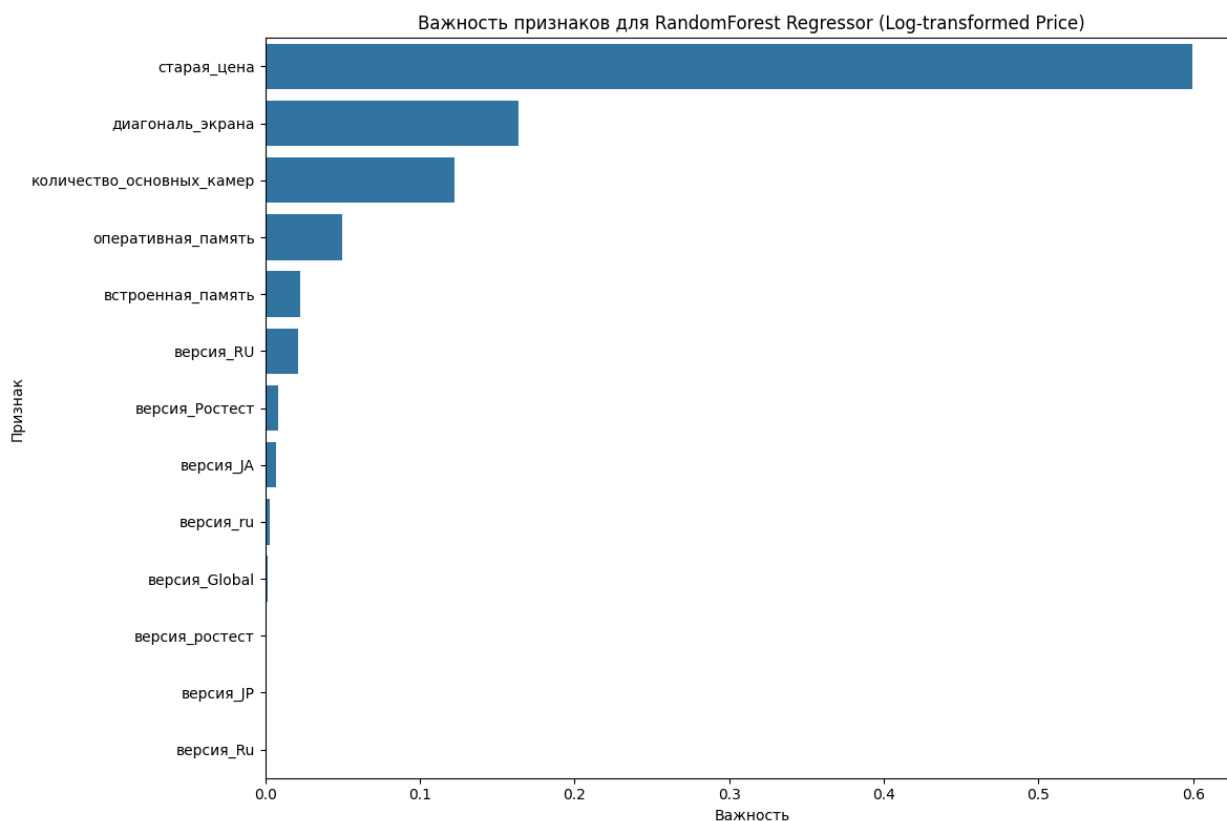


Рисунок 11 – Таблица с анализом важности признаков

Помимо цены, значительное влияние на прогноз оказывают характеристики дисплея и параметры основной камеры. Модель успешно выявила нелинейную зависимость: наличие нескольких модулей камер и высокая плотность пикселей экрана выступают маркерами дорогостоящих устройств. Примечательно, что такие параметры, как версия операционной системы, оказались наименее значимыми предикторами, что подтверждает гипотезу о вторичности этих факторов при выборе смартфона потребителем. Полученные результаты интерпретации не только подтверждают адекватность работы построенных моделей, но и могут быть использованы ритейлерами для оптимизации ценовой политики на основе наиболее востребованных характеристик оборудования.

4 Выводы

В ходе проведенного исследования реализован полный цикл интеллектуального анализа данных, направленный на решение задачи прогнозирования рыночной стоимости мобильных телефонов на основе их технических характеристик. На этапе подготовки данных применение метода межквартильного размаха (IQR) позволило эффективно очистить выборку от аномалий и ошибок парсинга, обеспечив стабильность обучения для массового сегмента устройств. Проведенный разведочный анализ подтвердил наличие сложных нелинейных зависимостей между аппаратным обеспечением и ценой, что предопределило выбор в пользу современных ансамблевых методов машинного обучения.

Сравнительный анализ 15 различных моделей показал значительное превосходство композиционных алгоритмов над классическими линейными подходами. Согласно полученным метрикам (Таблица 2), наилучшую точность продемонстрировал алгоритм RandomForest Regressor при работе с логарифмированной целевой переменной, достигнув коэффициента детерминации $R^2 = 0,970$ и минимальной средней абсолютной ошибки MAE = 3376,6. Также высокую эффективность показал Gradient Boosting Regressor на исходных данных с показателем $R^2 = 0,969$. Данные результаты свидетельствуют о том, что построенные модели способны объяснять более 97% дисперсии цен, что является высоким показателем для динамичного рынка электроники.

Анализ важности признаков позволил установить, что ключевыми факторами ценообразования являются объемы оперативной и встроенной памяти, а также характеристики камерной системы. Математически подтверждено, что именно эти параметры выступают основными дифференциаторами ценовых сегментов. Разработанный инструментарий может быть практически применен для автоматизации оценки стоимости б/у устройств, оптимизации закупочных цен в ритейле или создания рекомендательных систем для покупателей. Перспективой дальнейшего исследования является интеграция в модель данных о престижности бренда и анализ динамики цен во времени.

Библиографический список

1. Редченко Ю. Н., Богачев Т.В. Анализ и прогнозирование цен объектов недвижимости в регионах России методами интеллектуального анализа данных // Технологическое предпринимательство: проблемы проектирования, применения и безопасности информационных систем в условиях цифровой экономики и реализации инновационного и технологического предпринимательства: Материалы международной студенческой научно-практической конференции. Ростов-на-Дону. Ростовский государственный экономический университет (РИНХ), 2023. С. 166-171. URL: <https://elibrary.ru/item.asp?id=68607778> (дата обращения: 02.05.2026).
2. Амирханова Г. А., Мансурова М. Е., Кобегенова Д. К., Адильбекова А. М. Применение методов интеллектуального анализа данных для оптимизации и визуализации расчета и интерпретации Индекса потребительских цен // Статистическое образование в России: интеллектуальный анализ данных: материалы международной конференции (конгресса). Оренбург. Оренбургский государственный университет. 2023. С. 349-359. URL: <https://elibrary.ru/item.asp?id=64569737> (дата обращения: 02.05.2026).
3. Голубева Е. П. Интеллектуальный анализ данных цен на легковые автомобили в Еврейской автономной области // Постулат. 2024. № 6(104). URL: <https://elibrary.ru/item.asp?id=80665895> (дата обращения: 02.05.2026).
4. Свидетельство о государственной регистрации программы для ЭВМ № 2025610514 Российская Федерация. Программное обеспечение для системного анализа больших данных патентных транзакций на основе деревьев решений : заявл. 10.12.2024: опубл. 10.01.2025 / Жуков А.О., Проничкин С.В., Бондарева М.К., Овсяникова Н.М., Башкатов А.И., Юхно А.С.; заявитель Федеральное государственное бюджетное научное учреждение «Экспертно-аналитический центр». URL: <https://elibrary.ru/item.asp?id=80277324> (дата обращения: 02.05.2026).
5. Киселев Е. С. Прогнозирование стоимости авиабилетов с использованием методов машинного обучения // Студенческий вестник. 2025. № 24-5(357). С. 42-45. URL: <https://elibrary.ru/item.asp?id=82619480> (дата обращения: 02.05.2026).
6. Программный код исследования в среде Google Colab. URL: <https://colab.research.google.com/drive/1TaM9WXwLhsOg5FH8-7QatbKnpah4Ba73> (дата обращения: 25.04.2026).