

Построения модели для прогнозирования цены ноутбуков разных моделей в среде Google Colaboratory и в системе Orange

Вихляев Дмитрий Романович

Приамурский государственный университет имени Шолом-Алейхема

Студент

Аннотация

В данной статье рассматривается метод создания регрессионных моделей для предсказания цен, на основе датасета. Для реализации используются среда Google Colaboratory и система Orange. В результате исследования будут созданы модели для предсказания цен на ноутбуки по 5 характеристикам.

Ключевые слова: машинное обучение, Python, регрессионная модель.

Sending a model to predict prices for laptops of previous models in the Google Collaboration environment and in the Orange system

Vikhlyayev Dmitry Romanovich

Sholom-Aleichem Priamursky State University

Student

Abstract

This article discusses a method for creating regression models for price prediction based on a dataset. The Google Colaboratory environment and the Orange system are used for implementation. As a result of the research, models will be created to predict the prices of laptops based on 5 characteristics.

Keywords: machine learning, Python, regression model.

1 Введение

1.1 Актуальность

Цены на ноутбуки могут существенно колебаться в зависимости от множества факторов, таких как технологические инновации, сезонные скидки, изменения спроса и предложения, а также макроэкономические условия. Прогнозирование цен на ноутбуки с использованием систем машинного обучения (ML) становится все более актуальным в условиях динамичного рынка. Это важно, как для производителей и ритейлеров, так и для потребителей. Машинное обучение предоставляет мощные инструменты для прогнозирования цен на ноутбуки, предлагая значительные преимущества в точности, адаптивности и персонализации. Несмотря на существующие вызовы, потенциал ML для улучшения бизнес-процессов и удовлетворенности клиентов делает его незаменимым инструментом в современном мире технологий. Разработка и внедрение таких систем позволит компаниям более эффективно управлять своими ресурсами и

стратегиями, а потребителям получать более релевантные предложения и информацию.

1.2 Обзор исследований

А.В.Шихина, Т.В.Ягодкина исследовали повышение точности предсказания цены электроэнергии за счет применения комбинированных моделей прогноза [1]. А.Д.Леонова применила глубокое обучение для предсказания цен акций [2]. М.Д.Никифорова предсказал цены на автомобили с помощью методов машинного обучения [3]. К.А.Рудеев, Н.О.Чубурь в своей статье использовали искусственный интеллект для предсказания цен в гостиничном бизнесе [4]. Е.В.Ларионова, А.В.Левенец применили нейро-нечеткие сети для задачи прогнозирования [5].

1.3 Цель исследования

Целью данного исследования является построения моделей прогнозирования цены ноутбуков по производственным характеристикам на основе регрессионного анализа.

2 Материалы и методы

В качестве среды разработки применяется веб сервис Google Colaboratory, специализированный для языка python. Дополнительно используется система Orange, как готовый инструмент, использующий различные библиотеки машинного обучения.

3 Результаты и обсуждения

Данные для исследования были взяты с сайта kaggle [7]. В наборе данных формата csv присутствуют различные модели ноутбуков разных производителей. Для прогнозирования цены использованы фирма производитель, вид процессора, размеры оперативной и постоянной памяти, а также размер экрана (рис. 1).

index	brand	processor	Ram	ROM	display_size
0	HP	5th Gen AMD Ryzen 5 5600H	8GB	512GB	15.6
1	HP	12th Gen Intel Core i3 1215U	8GB	512GB	15.6
2	Acer	11th Gen Intel Core i3 1115G4	8GB	512GB	14.0
3	Lenovo	12th Gen Intel Core i5 1240P	16GB	512GB	14.0
4	Apple	Apple M1	8GB	256GB	13.3
5	Acer	12th Gen Intel Core i5 1240P	8GB	512GB	14.0
6	Dell	12th Gen Intel Core i3 1215U	8GB	512GB	15.6
7	Acer	13th Gen Intel Core i5 13420H	16GB	512GB	15.6
8	Asus	12th Gen Intel Core i5 12500H	8GB	512GB	15.6
9	Samsung	12th Gen Intel Core i5 1240P	16GB	512GB	13.3
10	Lenovo	12th Gen Intel Core i7 1255U	16GB	512GB	15.6
11	Ultimus	Intel Celeron N4020	4GB	128GB	14.1
12	Asus	12th Gen Intel Core i5 12500H	16GB	512GB	15.6
13	Primebook	MediaTek MTK8788	4GB	64GB	11.6
14	Lenovo	7th Gen AMD Ryzen 3 7320U	8GB	512GB	15.6
15	Asus	11th Gen Intel Core i5 11400H	8GB	512GB	15.6
16	MSI	11th Gen Intel Core i3 1115G4	8GB	512GB	14.0
17	Infinix	13th Gen Intel Core i9 13900H	32GB	1TB	15.6
18	HP	12th Gen Intel Core i5 12450H	16GB	512GB	15.6
19	Lenovo	11th Gen Intel Core i5 11300H	8GB	512GB	15.6
20	Asus	5th Gen AMD Ryzen 5 5600H	8GB	512GB	16.0
21	Apple	Apple M2	8GB	256GB	13.6
22	Acer	12th Gen Intel Core i5 12450H	16GB	512GB	15.6

Рисунок 1. Фрагмент таблицы с набором данных

В первую очередь необходимо импортировать библиотеки, выбрать используемые и целевые параметры, после чего определится с типами данных и единицами измерений для каждого поля. Таким образом, для постоянной памяти, единицей измерения будет гигабайт (рис.2).

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.neighbors import KNeighborsRegressor
from sklearn.metrics import mean_squared_error, r2_score

def convert_rom(rom_value):
    if 'TB' in rom_value:
        return int(rom_value.replace('TB', '')) * 1000
    else:
        return int(rom_value.replace('GB', ''))

features = ['brand', 'processor', 'Ram', 'ROM', 'display_size']
target = 'price'
```

Рисунок 2. Подготовка функций и переменных

Далее необходимо загрузить и обработать данные. Разделить данных на признаки и целевую переменную (рис.3).

```
file_path = 'data.csv'
df = pd.read_csv(file_path)

df_model = df[features + [target]].copy()

df_model = pd.get_dummies(df_model, columns=['brand', 'processor'])

df_model['Ram'] = df_model['Ram'].str.replace('GB', '').astype(int)
df_model['ROM'] = df_model['ROM'].apply(convert_rom)

X = df_model.drop('price', axis=1)
y = df_model['price']
```

Рисунок 3. Обработка данных

Ниже представлен код реализующий функции: разделение данных на тренировочную и тестовую выборки, масштабирование данных, создание и обучение модели ближайших соседей с параметром 5, прогнозирование на тестовом наборе, Оценка модели и вывод результатов оценки модели.

Чтобы узнать насколько хорошая модель получилась, снова применяется модуль библиотеки sklearn. Более важным является `r2_score`, так как данный модуль поможет узнать коэффициент детерминации модели (R^2). Коэффициент детерминации для модели при верном расчете принимает значения от 0 до 1. Для приемлемых моделей предполагается, что коэффициент детерминации должен быть не менее 70 %. У полученной

модели линейной регрессии коэффициент детерминации составляет 53%, что говорит о том, что модель не очень хорошая (рис.4).

```
x_train, x_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)

knn_model = KNeighborsRegressor(n_neighbors=5)
knn_model.fit(X_train_scaled, y_train)

y_pred_knn = knn_model.predict(X_test_scaled)

mse_knn = mean_squared_error(y_test, y_pred_knn)
r2_knn = r2_score(y_test, y_pred_knn)

mse_knn, r2_knn

(1598400254.155978, 0.5335377762499747)
```

Рисунок 4. Расчет коэффициента детерминации общей модели ближайших соседей

Следующая используемая модель случайного леса с параметрами, указанными на рисунке 5 показывает результат оценки коэффициента детерминации в 77%, что соответствует поставленным требованиям.

```
x_train, x_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)

# Создание и обучение модели случайного леса
rf_model = RandomForestRegressor(n_estimators=100, random_state=42)
rf_model.fit(X_train_scaled, y_train)

# Прогнозирование на тестовом наборе
y_pred_rf = rf_model.predict(X_test_scaled)

# Оценка модели случайного леса
mse_rf = mean_squared_error(y_test, y_pred_rf)
r2_rf = r2_score(y_test, y_pred_rf)

mse_rf, r2_rf

(766323586.4010237, 0.7763632711548394)
```

Рисунок 5. Коэффициент детерминации модели случайный лес.

Аналогичный анализ проведен в системе Orange, однако в данном случае использовалось больше регрессионных моделей.

Первым делом внесены полученные данные в систему Orange и разделены на тестовые и тренировочные данные в пропорции 103/43 с помощью объекта Date Sampler. Далее добавляется ещё несколько объектов: Scatter Plot и Correlations. Они связываются через промежуточный инструмент Test and Score. Уже от Test and Score проводится к нужным

моделям, с отрегулированными параметрами. Среди выбранных моделей были: Random Forest, SVM, kNN и Linear Regression (рис.6).

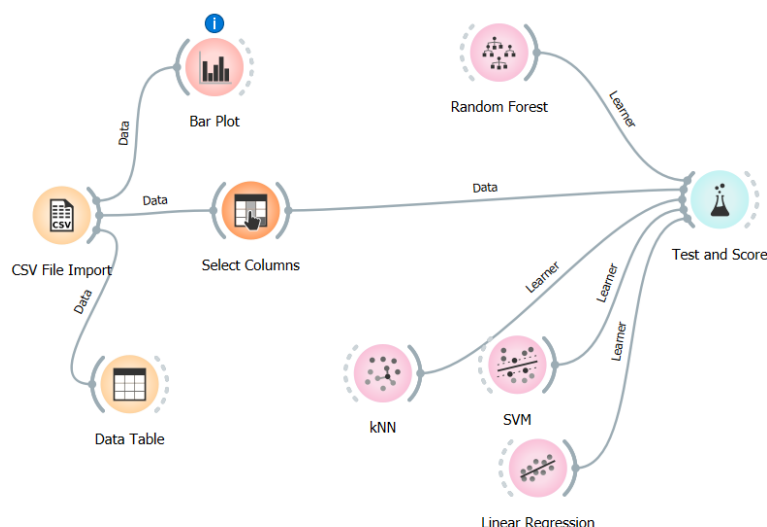


Рисунок 6. Полная модель

На основе внесённых данных, программа Orange самостоятельно реализует анализ. Результат можно увидеть в окне инструмента Test and Score. Главным показателем выбран коэффициент детерминации (рис.7).

Model	MSE	RMSE	MAE	MAPE	$\hat{R}^2$
SVM	3722784756.560	61014.627	41663.744	0.718	-0.006
kNN	747261366.037	27336.082	15246.276	0.178	0.798
Linear Regression	725512560.971	26935.340	16347.784	0.218	0.804
Random Forest	362681948.327	19044.210	11077.067	0.138	0.902

Рисунок 7. Результат в Test and Score.

Самой лучшей получилась модель Random Forest, с коэффициентом детерминации выше 90 процентов.

В результате исследования построены модели прогнозирования в среде Google Colaboratory и системе Orange. Модель можно использовать для покупки и продажи ноутбуков разных типов, основываясь на 5 характеристиках.

Библиографический список

1. Шихина А.В., Ягодкина Т.В. Повышение точности предсказания цены электроэнергии за счет применения комбинированных моделей прогноза // Вестник Московского энергетического института. 2020. № 6. С. 119-128.

2. Леонова А.Д. Применение глубокого обучения для предсказания цен акций // В сборнике: Информационно-телекоммуникационные технологии и математическое моделирование высокотехнологичных систем. материалы Всероссийской конференции с международным участием. Москва, 2023. С. 362-367.
3. Никифорова М.Д. Предсказание цен на автомобили с помощью методов машинного обучения // В сборнике: Инжиниринг предприятий и управление знаниями (ИП&УЗ-2022). Сборник научных трудов XXV Российской научной конференции. В 2-х томах. Под научной редакцией Ю.Ф. Тельнова. Москва, 2022. С. 161-167.
4. Рудеев К.А., Чубырь Н.О. Применение искусственного интеллекта для предсказания цен в гостиничном бизнесе // В сборнике: Прикладная математика: современные проблемы математики, информатики и моделирования. Материалы V Всероссийской научно-практической конференции, молодых ученых. Краснодар, 2023. С. 103-108.
5. Ларионова Е.В., Левенец А.В. Применение нейро-нечетких сетей для задачи прогнозирования // В сборнике: Информационные технологии XXI века. Сборник научных трудов. Редакционная коллегия: ответственный редактор В.В. Воронин [и др.]. Хабаровск, 2020. С. 83-87.
6. Галимьянов А.Ф. Особенности прогнозирования стоимости товаров веблена с помощью машинного обучения // Научный аспект. 2023. Т. 17. № 6. С. 2206-2208.
7. Набор данных цен на ноутбуки URL: <https://www.kaggle.com/datasets/jacksondivakarr/laptop-price-prediction-dataset/data> (дата обращения 02.06.2024).