

Министерство образования Республики Беларусь  
Учреждение образования  
«Брестский Государственный технический университет»  
Кафедра ИИТ

**Лабораторная работа №2**

**По дисциплине:** «Основы машинного обучения»

**Тема:** «Линейные модели для задач регрессии и классификации»

**Выполнила:**

Студентка 3 курса

Группы АС-66

Прокурат В. Д.

**Проверил:**

Крощенко А. А.

**Брест 2025**

**Цель работы:** изучить применение линейной и логистической регрессии для решения практических задач. Научиться обучать модели, оценивать их качество с помощью соответствующих метрик и интерпретировать результаты.

## Вариант 9

### • Регрессия (Прогнозирование цены автомобиля)

1. Car Price Prediction

2. Предсказать цену автомобиля (price)

3. Задания:

- загрузите данные. Выберите 5-6 числовых признаков (например, horsepower, citympg, enginesize);
- обучите модель **линейной регрессии**;
- рассчитайте **R<sup>2</sup>** и **MAE**;
- визуализируйте зависимость price от horsepower с линией регрессии.

### • Классификация (Детекция мошеннических транзакций)

1. Credit Card Fraud Detection

2. Определить, является ли транзакция мошеннической (Class = 1)

3. Задания:

- загрузите данные. Обратите внимание на сильный дисбаланс классов;
- обучите модель **логистической регрессии**;
- рассчитайте **Accuracy, Precision и Recall**. Объясните, почему Accuracy в данном случае не является показательной метрикой;
- постройте **матрицу ошибок**.

### Код программы (регрессия):

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.linear_model import LinearRegression
from sklearn.metrics import r2_score, mean_absolute_error
from sklearn.model_selection import train_test_split

df = pd.read_csv("CarPrice_Assignment.csv")

features = ['horsepower', 'citympg', 'enginesize', 'curbweight', 'carwidth',
            'carlength']
X = df[features]
y = df['price']
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)
```

```
model = LinearRegression()
model.fit(X_train, y_train)
```

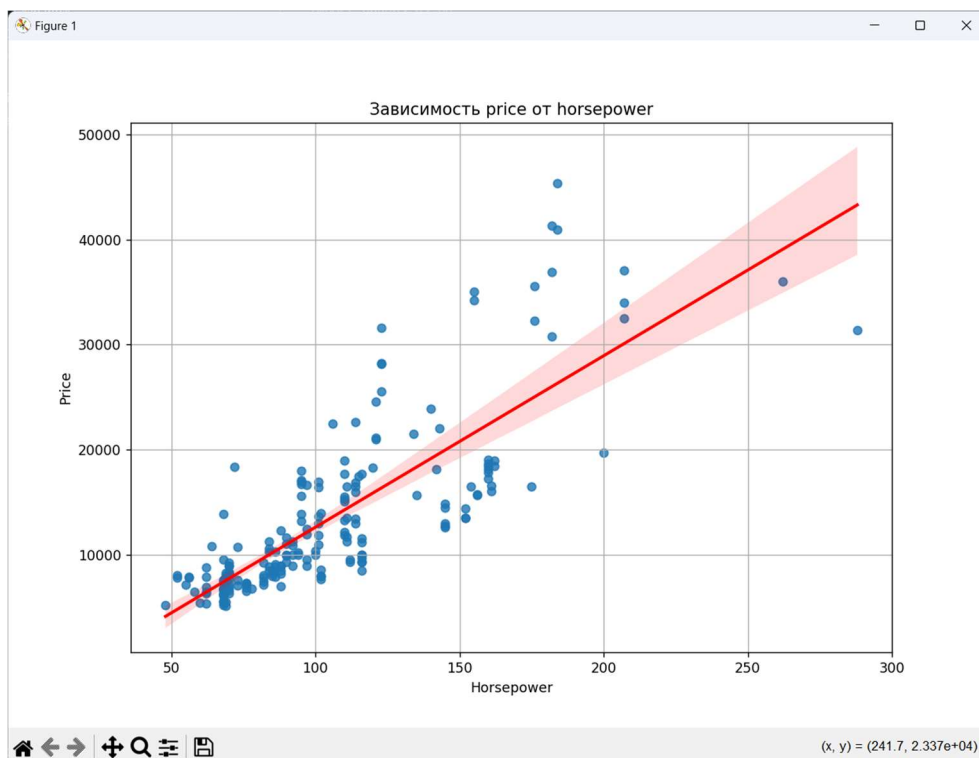
```
y_pred = model.predict(X_test)
```

```
r2 = r2_score(y_test, y_pred)
mae = mean_absolute_error(y_test, y_pred)
print(f"R²: {r2:.3f}")
print(f"MAE: {mae:.2f}")
```

```
plt.figure(figsize=(10, 7))
sns.regplot(x=df['horsepower'], y=df['price'], line_kws={"color": "red"})
plt.title("Зависимость price от horsepower")
plt.xlabel("Horsepower")
plt.ylabel("Price")
plt.grid(True)
plt.show()
```

## Результат работы программы:

```
PS D:\политех\ЛАБЫ\5 семестр\ОМО\2 лаба> &
x/ЛАБЫ/5 семестр/ОМО/2 лаба/regression.py"
R²: 0.814
MAE: 2772.52
```



## Код программы (классификация):

```
import pandas as pd
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, precision_score, recall_score,
confusion_matrix
import seaborn as sns
import matplotlib.pyplot as plt

df = pd.read_csv("creditcard.csv")

X = df.drop(columns=['Class'])
y = df['Class']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42, stratify=y)

model = LogisticRegression(max_iter=12345)
model.fit(X_train, y_train)

y_pred = model.predict(X_test)

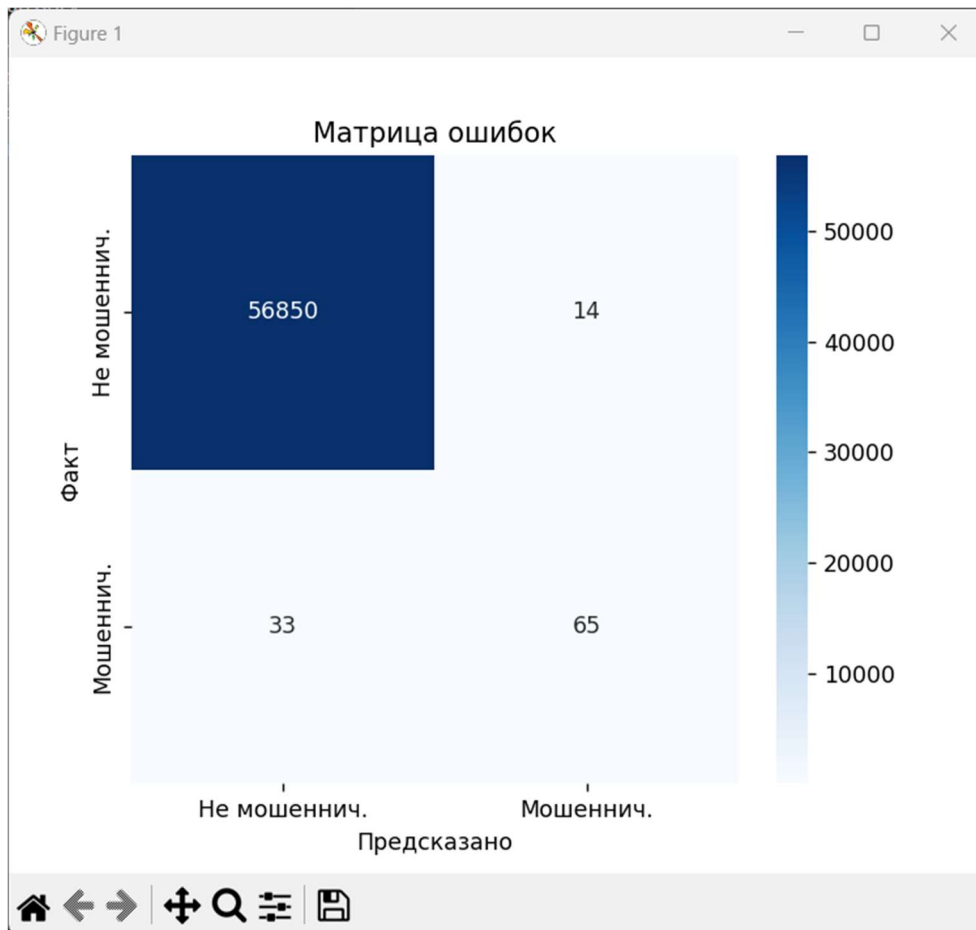
accuracy = accuracy_score(y_test, y_pred)
precision = precision_score(y_test, y_pred)
recall = recall_score(y_test, y_pred)

print(f"Accuracy: {accuracy:.3f}")
print(f"Precision: {precision:.3f}")
print(f"Recall: {recall:.3f}")

cm = confusion_matrix(y_test, y_pred)
plt.figure(figsize=(6, 5))
sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', xticklabels=['Не мошеннич.',
'Mошеннич.'], yticklabels=['Не мошеннич.', 'Мошеннич.'])
plt.title("Матрица ошибок")
plt.xlabel("Предсказано")
plt.ylabel("Факт")
plt.show()
```

## Результат работы программы:

```
PS D:\политех\ЛАБЫ\5 семестр\ОМО\2 лаба> & C:/U
x/ЛАБЫ/5 семестр/ОМО/2 лаба/classification.py"
Accuracy: 0.999
Precision: 0.823
Recall: 0.663
```



Ассигасу не является показательной метрикой при дисбалансе классов, так как большинство транзакций - не мошеннические. Модель может просто предсказывать '0' и получать высокий Accuracy.

**Вывод:** изучила применение линейной и логистической регрессии для решения практических задач. Научилась обучать модели, оценивать их качество с помощью соответствующих метрик и интерпретировать результаты.