

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

Лабораторная работа №2
По дисциплине: «ОМО»

Выполнил:
Студент 3-го курса
Группы АС-66
Николова М.С
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.

Ход работы
Вариант 6

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression, LogisticRegression
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import (
    mean_squared_error, r2_score,
    accuracy_score, precision_score, recall_score, confusion_matrix
)
print("\n==== РЕГРЕССИЯ: Boston Housing ===")
df_boston = pd.read_csv("BostonHousing.csv")

X = df_boston.drop('MEDV', axis=1)
y = df_boston['MEDV']

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

reg_model = LinearRegression()
reg_model.fit(X_train, y_train)
y_pred = reg_model.predict(X_test)

mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

print(f"MSE: {mse:.3f}")
print(f"R2: {r2:.3f}")

plt.figure(figsize=(8,6))
sns.scatterplot(x=df_boston['RM'], y=df_boston['MEDV'], alpha=0.6, label="Данные")
sns.regplot(x='RM', y='MEDV', data=df_boston, scatter=False, color='red', label="Линия регрессии")
plt.xlabel("Среднее количество комнат (RM)")
plt.ylabel("Медианная стоимость дома (MEDV)")
plt.title("Зависимость стоимости жилья от числа комнат")
plt.legend()
plt.show()

print("\n==== КЛАССИФИКАЦИЯ: Breast Cancer Wisconsin ===")
df_cancer = pd.read_csv("breast_cancer.csv")
```

```
df_cancer = df_cancer.drop(columns=["id", "Unnamed: 32"], errors="ignore")

df_cancer["diagnosis"] = df_cancer["diagnosis"].map({"M": 1, "B": 0})

print("Количество пропусков после очистки:")
print(df_cancer.isna().sum().sum())

X = df_cancer.drop("diagnosis", axis=1)
y = df_cancer["diagnosis"]

scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)

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

clf_model = LogisticRegression(max_iter=5000)
clf_model.fit(X_train, y_train)
y_pred = clf_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=(6,4))
sns.heatmap(cm, annot=True, fmt="d", cmap="Blues",
            xticklabels=["Злокачественная", "Доброположительная"],
            yticklabels=["Злокачественная", "Доброположительная"])
plt.xlabel("Предсказание")
plt.ylabel("Истина")
plt.title("Матрица ошибок")
plt.show()

tn, fp, fn, tp = cm.ravel()

print("\n==== Разбор матрицы ошибок ===")
print(f"Истинно отрицательные (TN): {tn}")
print(f"Ложно положительные (FP): {fp}")
print(f"Ложно отрицательные (FN): {fn}")
print(f"Истинно положительные (TP): {tp}")

print("\nПояснение:")
```

```
print("- TN (истинно отрицательные): модель правильно определила доброкачественные опухоли.")  
print("- TP (истинно положительные): модель правильно распознала злокачественные опухоли.")  
print("- FP (ложно положительные): модель ошибочно сочла доброкачественную опухоль злокачественной.")  
print("- FN (ложно отрицательные): модель ошибочно определила злокачественную опухоль как доброкачественную (опасно!).")
```

РЕГРЕССИЯ: Boston Housing

MSE: 14.801

R²: 0.798

КЛАССИФИКАЦИЯ: Breast Cancer Wisconsin

Количество пропусков после очистки:

0

Accuracy: 0.974

Precision: 0.976

Recall: 0.953

Разбор матрицы ошибок

Истинно отрицательные (TN): 70

Ложно положительные (FP): 1

Ложно отрицательные (FN): 2

Истинно положительные (TP): 41

Пояснение:

- TN (истинно отрицательные): модель правильно определила доброкачественные опухоли.

- TP (истинно положительные): модель правильно распознала злокачественные опухоли.

- FP (ложно положительные): модель ошибочно сочла доброкачественную опухоль злокачественной.

- FN (ложно отрицательные): модель ошибочно определила злокачественную опухоль как доброкачественную (опасно!).



