

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

Лабораторная работа №2
По дисциплине: «ОМО»
Тема : “линейные модели
для задач регрессии и классификации”

Выполнил:
Студент 3-го курса
Группы АС-66
Цеван К.А.
Проверил:
Крощенко А.А.

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

Вариант 2

- Регрессия (Прогнозирование медицинских расходов)

1. Medical Cost Personal Datasets

2. Предсказать страховые выплаты (charges)

3. Задания:

§ загрузите и обработайте категориальные признаки (например, sex, smoker);

§ обучите модель линейной регрессии для предсказания charges;

§ рассчитайте MAE (Mean Absolute Error) и R2;

§ визуализируйте зависимость charges от bmi (индекс массы тела) с помощью диаграммы рассеяния и линии регрессии.

- Классификация (Диагностика заболеваний сердца)

1. Heart Disease UCI

2. Предсказать наличие у пациента болезни сердца (target)

3. Задания:

§ загрузите данные и разделите их на обучающую и тестовую выборки;

§ обучите модель логистической регрессии;

§ оцените модель с помощью Accuracy, Precision, Recall и F1-score;

§ постройте матрицу ошибок.

charges:

```
import os
```

```
import sys
```

```
import numpy as np
```

```
import pandas as pd
```

```
import matplotlib.pyplot as plt
```

```
from sklearn.linear_model import LinearRegression
```

```
DATA_FILE = "medical_cost_personal_dataset.csv"
```

```
IMG_FILE = "charges_vs_bmi.png"
```

```
def check_file(filename: str):
```

```
    if not os.path.isfile(filename):
```

```
        print(f"Ошибка: файл '{filename}' отсутствует.")
```

```
        sys.exit(1)
```

```

def plot_and_save(x, y, model, out_path: str):
    # Создаем точки для линии регрессии
    x_line = np.linspace(x.min(), x.max(), 200).reshape(-1, 1)
    y_line = model.predict(x_line)

    plt.figure(figsize=(8, 6))
    plt.scatter(x, y, alpha=0.5, label="Исходные данные")
    plt.plot(x_line, y_line, color="red", linewidth=2, label="Линия регрессии")
    plt.xlabel("BMI")
    plt.ylabel("Charges")
    plt.title("Зависимость страховых выплат от BMI")
    plt.legend()
    plt.tight_layout()
    plt.savefig(out_path, dpi=200)
    plt.close()
    print(f'График сохранён: {out_path}')

def main():
    check_file(DATA_FILE)

    data = pd.read_csv(DATA_FILE)
    print("Файл загружен:", DATA_FILE, "| размер:", data.shape)

    if {"bmi", "charges"} - set(data.columns):
        print("Ошибка: столбцы 'bmi' и/или 'charges' отсутствуют.")
        sys.exit(1)

    # Берём только нужные данные
    subset = data[["bmi", "charges"]].dropna()
    if subset.shape[0] < 2:
        print("Недостаточно данных для построения графика.")
        sys.exit(1)

    X = subset[["bmi"]].values
    y = subset["charges"].values

    # Линейная регрессия
    model = LinearRegression()
    model.fit(X, y)

    plot_and_save(X, y, model, IMG_FILE)
    print("Готово.")

if __name__ == "__main__":
    main()

```

```
2.
import os
import sys
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

from sklearn.linear_model import LinearRegression

DATA_FILE = "medical_cost_personal_dataset.csv"
IMG_FILE = "charges_vs_bmi.png"

def check_file(filename: str):
    if not os.path.isfile(filename):
        print(f"Ошибка: файл '{filename}' отсутствует.")
        sys.exit(1)

def plot_and_save(x, y, model, out_path: str):
    # Создаем точки для линии регрессии
    x_line = np.linspace(x.min(), x.max(), 200).reshape(-1, 1)
    y_line = model.predict(x_line)

    plt.figure(figsize=(8, 6))
    plt.scatter(x, y, alpha=0.5, label="Исходные данные")
    plt.plot(x_line, y_line, color="red", linewidth=2, label="Линия регрессии")
    plt.xlabel("BMI")
    plt.ylabel("Charges")
    plt.title("Зависимость страховых выплат от BMI")
    plt.legend()
    plt.tight_layout()
    plt.savefig(out_path, dpi=200)
    plt.close()
    print(f"График сохранён: {out_path}")

def main():
    check_file(DATA_FILE)

    data = pd.read_csv(DATA_FILE)
    print("Файл загружен:", DATA_FILE, "| размер:", data.shape)

    if {"bmi", "charges"} - set(data.columns):
        print("Ошибка: столбцы 'bmi' и/или 'charges' отсутствуют.")
        sys.exit(1)

    # Берём только нужные данные
    subset = data[["bmi", "charges"]].dropna()
    if subset.shape[0] < 2:
```

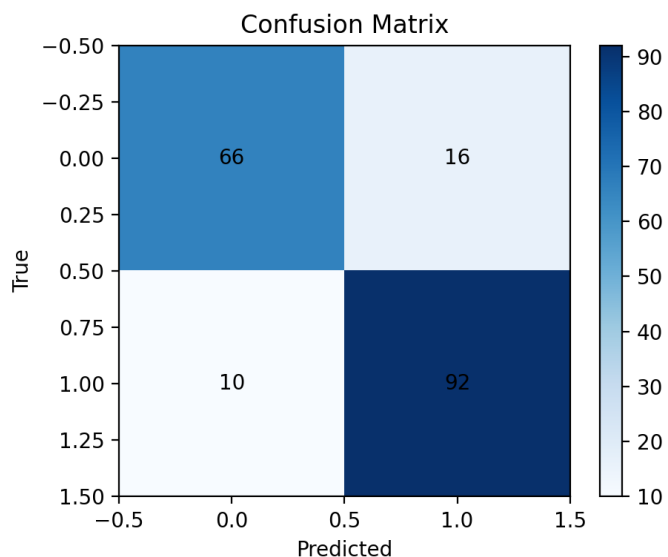
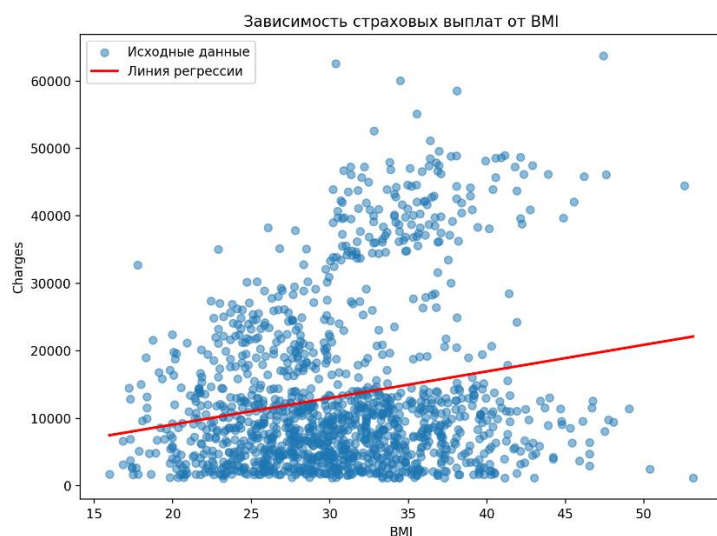
```
print("Недостаточно данных для построения графика.")
sys.exit(1)
```

```
X = subset[["bmi"]].values
y = subset["charges"].values
```

```
# Линейная регрессия
model = LinearRegression()
model.fit(X, y)
```

```
plot_and_save(X, y, model, IMG_FILE)
print("Готово.")
```

```
if __name__ == "__main__":
    main()
```



Вывод: Изучили применение линейной и логистической регрессии для решения практических задач.