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-65

Нестюк Н.С.

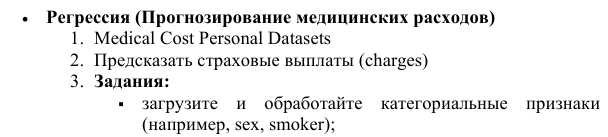
Проверил:

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.

Вариант 2



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

from sklearn.metrics import mean\_absolute\_error, r2\_score

url = "https://raw.githubusercontent.com/stedy/Machine-Learning-with-R-datasets/master/insurance.csv"

data = pd.read\_csv(url)

print("Размер данных:", data.shape)

print(data.head())

data\_encoded = pd.get\_dummies(data, drop\_first=True)

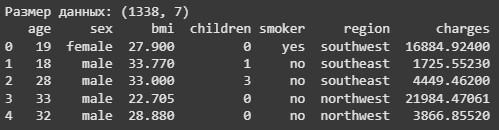
X = data\_encoded.drop("charges", axis=1)

y = data\_encoded["charges"]

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)



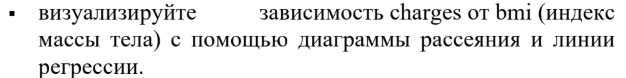
mae = mean\_absolute\_error(y\_test, y\_pred)

r2 = r2\_score(y\_test, y\_pred)

print(f"MAE (Средняя абсолютная ошибка): {mae:.2f}")

print(f"R2 (Коэффициент детерминации): {r2:.4f}")





plt.figure(figsize=(8,6))

sns.scatterplot(x=data["bmi"], y=data["charges"], alpha=0.6)

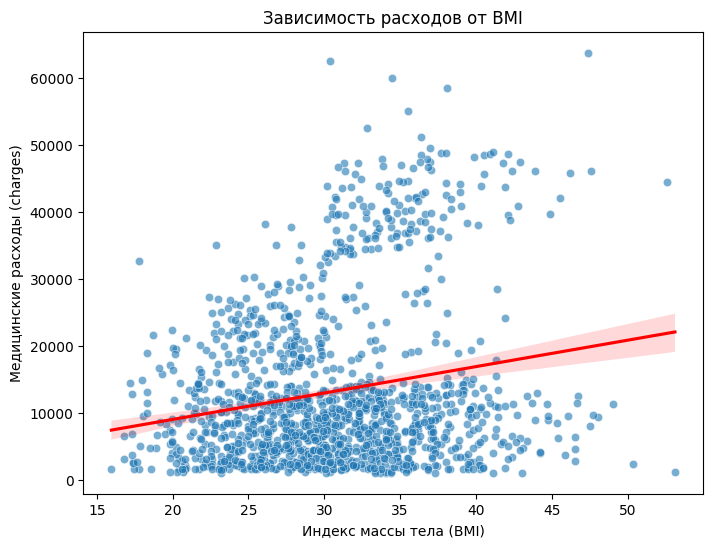
sns.regplot(x=data["bmi"], y=data["charges"], scatter=False, color="red")

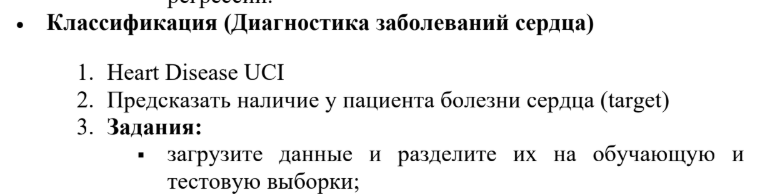
plt.xlabel("Индекс массы тела (BMI)")

plt.ylabel("Медицинские расходы (charges)")

plt.title("Зависимость расходов от BMI")

plt.show()





url = "https://raw.githubusercontent.com/sharmaroshan/Heart-UCI-Dataset/master/heart.csv"

data = pd.read\_csv(url)

print("Размер данных:", data.shape)

print(data.head())

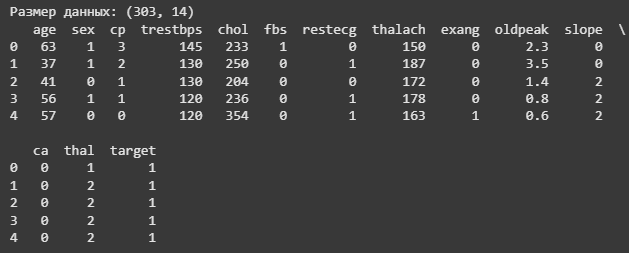
X = data.drop("target", axis=1)

y = data["target"]

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=1000)

model.fit(X\_train, y\_train)

y\_pred = model.predict(X\_test)



acc = accuracy\_score(y\_test, y\_pred)

prec = precision\_score(y\_test, y\_pred)

rec = recall\_score(y\_test, y\_pred)

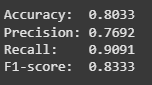
f1 = f1\_score(y\_test, y\_pred)

print(f"Accuracy:  {acc:.4f}")

print(f"Precision: {prec:.4f}")

print(f"Recall:    {rec:.4f}")

print(f"F1-score:  {f1:.4f}")





cm = confusion\_matrix(y\_test, y\_pred)

plt.figure(figsize=(6,5))

sns.heatmap(cm, annot=True, fmt="d", cmap="Blues",

            xticklabels=["Нет болезни", "Есть болезнь"],

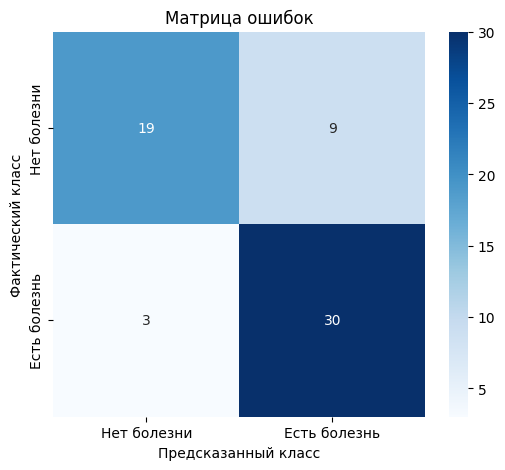
            yticklabels=["Нет болезни", "Есть болезнь"])

plt.xlabel("Предсказанный класс")

plt.ylabel("Фактический класс")

plt.title("Матрица ошибок")

plt.show()



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