

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-65
Сунцова М. Д.
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 8

Регрессия (Прогнозирование качества вина)

1. Wine Quality

2. Предсказать оценку качества вина (quality) как непрерывную величину

3. Задания:

- загрузите данные

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.linear_model import LinearRegression, LogisticRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error, r2_score, accuracy_score,
precision_score, recall_score, confusion_matrix
```

```
df = pd.read_csv('winequality-white.csv', sep=';')
print(df.head())
```

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	ph	sulphates	alcohol	quality
0	7.0	0.27	0.36	20.7	0.045	45.0	170.0	1.0010	3.00	0.45	11.8	6
1	6.3	0.30	0.34	1.6	0.049	14.0	132.0	0.9960	3.30	0.49	9.5	6
2	8.1	0.28	0.40	6.9	0.050	30.0	97.0	0.9954	3.26	0.44	10.1	6
3	7.2	0.23	0.32	8.5	0.058	47.0	186.0	0.9956	3.19	0.40	9.9	6
4	7.2	0.23	0.32	8.5	0.058	47.0	186.0	0.9956	3.19	0.40	9.9	6

- обучите модель линейной регрессии на всех доступных признаках;
- рассчитайте MSE и R²;

```
X_reg = df.drop('quality', axis=1)
y_reg = df['quality']
```

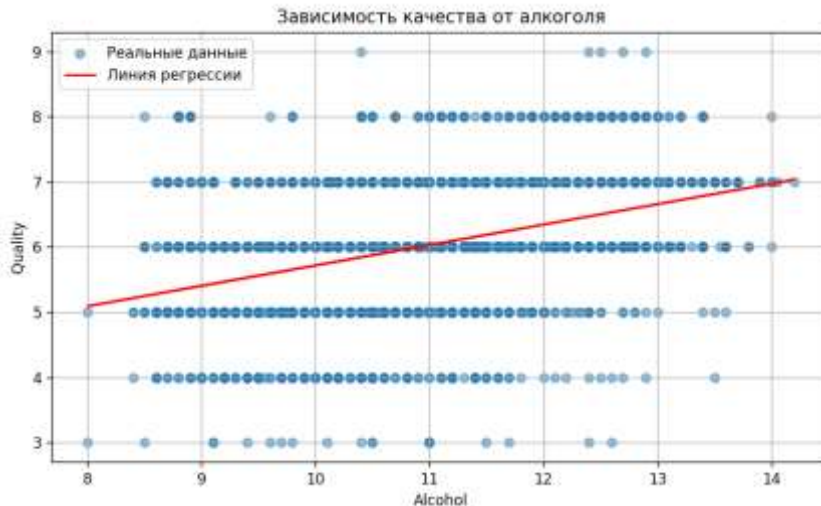
```
X_train_reg, X_test_reg, y_train_reg, y_test_reg = train_test_split(X_reg,
y_reg, test_size=0.2, random_state=42)
reg_model = LinearRegression()
reg_model.fit(X_train_reg, y_train_reg)
y_pred_reg = reg_model.predict(X_test_reg)
print("MSE:", mean_squared_error(y_test_reg, y_pred_reg))
print("R²:", r2_score(y_test_reg, y_pred_reg))
```

```
MSE: 0.5690247717229248
R²: 0.2652750042179164
```

- визуализируйте зависимость quality от alcohol с линией регрессии.

```
plt.figure(figsize=(8, 5))
X_alcohol = df[['alcohol']]
reg_alcohol = LinearRegression()
reg_alcohol.fit(X_alcohol, y_reg)
y_pred_alcohol = reg_alcohol.predict(X_alcohol)
```

```
plt.scatter(df['alcohol'], df['quality'], alpha=0.5, label='Реальные данные')
plt.plot(df['alcohol'], y_pred_alcohol, color='red', label='Линия регрессии')
plt.xlabel('Alcohol')
plt.ylabel('Quality')
plt.title('Зависимость качества от алкоголя')
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()
```



Классификация (Определение "хорошего" вина)

1. Wine Quality
2. Классифицировать вино как "хорошее" ($quality \geq 7$) или "плохое" ($quality < 7$)
3. Задания:
 - создайте новый бинарный целевой столбец на основе столбца quality;
 - обучите модель логистической регрессии;

```
df['quality_binary'] = df['quality'].apply(lambda x: 1 if x >= 7 else 0)
print(df[['quality', 'quality_binary']].head())
```

```
X_clf = df.drop(['quality', 'quality_binary'], axis=1)
y_clf = df['quality_binary']
```

```
X_train_clf, X_test_clf, y_train_clf, y_test_clf = train_test_split(X_clf,
y_clf, test_size=0.2, random_state=42)
clf_model = LogisticRegression(max_iter=1000)
clf_model.fit(X_train_clf, y_train_clf)
y_pred_clf = clf_model.predict(X_test_clf)
```

	quality	quality_binary
0	6	0
1	6	0
2	6	0
3	6	0
4	6	0

- оцените модель с помощью Accuracy, Precision, Recall для класса "хорошее";

```
print("Accuracy:", accuracy_score(y_test_clf, y_pred_clf))
print("Precision (класс хорошие):", precision_score(y_test_clf, y_pred_clf))
print("Recall (класс хорошие):", recall_score(y_test_clf, y_pred_clf))
```

```
Accuracy: 0.7836734693877551
Precision (класс хорошие): 0.5773195876288659
Recall (класс хорошие): 0.24669603524229075
```

- постройте матрицу ошибок.

```
conf_matrix = confusion_matrix(y_test_clf, y_pred_clf)
plt.figure(figsize=(6, 4))
sns.heatmap(conf_matrix, annot=True, fmt='d', cmap='Blues', xtick-
labels=['Плохие', 'Хорошие'], yticklabels=['Плохие', 'Хорошие'])
plt.xlabel('Предсказано')
plt.ylabel('Факт')
plt.title('Матрица ошибок')
plt.tight_layout()
plt.show()
```



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