

Федеральное государственное бюджетное образовательное
учреждение высшего образования
«МИРЭА – Российский Технологический Университет»

Отчёт по практической работе №1: «Реализация алгоритма k-ближайших соседей»

Подготовила студентка
группы ББМО-01-21
Концевая Юлия

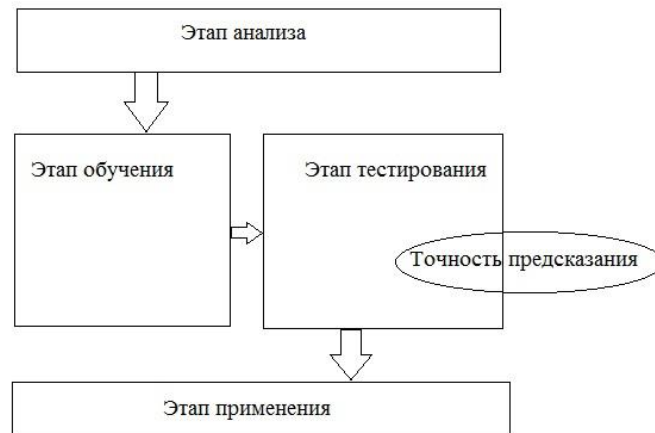
Описание

- Для написания алгоритма был использован высокоуровневый язык программирования Python в среде **Jupyter notebook**.
- **Jupyter Notebook** — это мощный инструмент для разработки и представления проектов в интерактивном виде.

Общая схема решаемой задачи

Общий подход к решению машинного обучения состоит из серии этапов. Эти этапы согласованы независимо от источника данных.

- **Этап анализа:** данные анализируются для выявления закономерностей, которые помогают создать новые признаки, для повышения точности обучения модели.
- **Этап обучения:** параметры данных, созданные на предыдущих этапах, используются для создания моделей машинного обучения. Фаза обучения - это итеративный процесс, модель может улучшать качество прогнозирования.
- **Этап тестирования:** модель машинного обучения, созданная на этапе обучения, тестируется на большом количестве данных, и оценивается её качество. На этом этапе тестируются данные, которые не использовались на предыдущем.
- **Этап применения:** использование модели на новых данных в производственной среде.



Используемые библиотеки

- Основной библиотекой для работы с данными является **pandas**, которая обеспечивает работу с табличными данными и служит для их первичного анализа и обработки.
- Для работы с алгоритмами машинного обучения была использована библиотека **sklearn**. **Scikit-learn** один из наиболее широко используемых пакетов Python для Data Science.
- В библиотеке **sklearn** реализован алгоритм kNN, возможность разделения данных на обучающую и тестовую выборку, а также построения матрицы ошибок.
- **seaborn** - библиотека для создания статистической графики на Python.

```
import pandas as pd
import numpy as np
from sklearn.neighbors import KNeighborsClassifier
import matplotlib.pyplot as plt;
import seaborn as sns;
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
```

Генерация данных

Набор данных был также сгенерирован с помощью библиотеки **sklearn**.

Набор данных представлял из себя:
15 семплов,
2 показателя
3 кластерных центра.

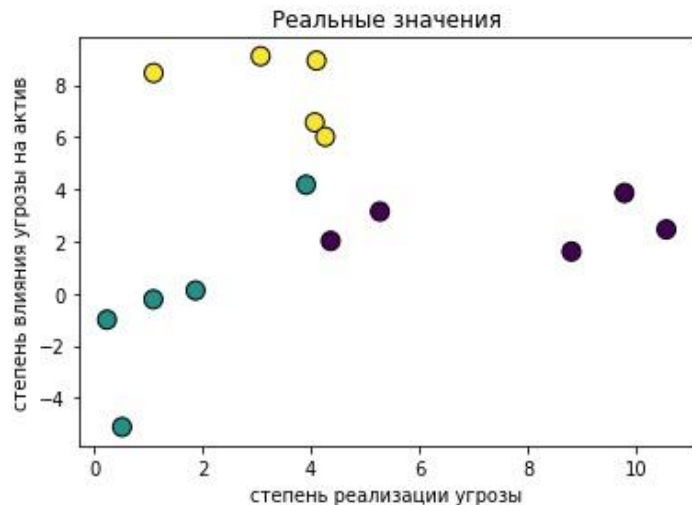
```
from sklearn.datasets import make_blobs
```

```
data = make_blobs(  
    n_samples = 15,  
    n_features = 2,  
    centers = 3,  
    center_box = (0, 10),  
    cluster_std = 1.8  
)  
  
X, y = data
```

Генерация данных

- После этого была построена визуализация для наглядной оценки данных.

```
: plt.scatter(data[0][:,0], data[0][:,1], c=data[1], s=100, edgecolors='k');  
plt.xlabel('степень реализации угрозы')  
plt.ylabel('степень влияния угрозы на актив');  
plt.title('Реальные значения');
```



Обучение алгоритма

- Разделение данных на обучающую и тестовую выборку.

```
X_train, X_test, y_train, y_test = train_test_split(X, y)
```

- Обучение выбранного алгоритма.

```
clf = KNeighborsClassifier().fit(X_train, y_train)
```

- Предсказание значений.

```
: y_pred = clf.predict(X_test)
```

Оценка качества модели

- Для расчета precision, recall, accuracy строится матрица ошибок (confusion matrix). Матрица ошибок показывает как часто модель ошибается при классификации.
- **Precision** - отношение правильно определённых положительных наблюдений ко всем определённым как положительные.
- **Recall** - отношение правильно определённых положительных случаев ко всем положительным: показывает какую часть положительных случаев модель правильно классифицирует.
- Точность модели составила **100%**.

```
from sklearn.metrics import classification_report, accuracy_score
print(classification_report(y_test, y_pred));
```

	precision	recall	f1-score	support
0	1.00	1.00	1.00	1
1	1.00	1.00	1.00	1
2	1.00	1.00	1.00	2
accuracy			1.00	4
macro avg	1.00	1.00	1.00	4
weighted avg	1.00	1.00	1.00	4

Федеральное государственное бюджетное образовательное
учреждение высшего образования
«МИРЭА – Российский Технологический Университет»

Реализация алгоритма k-ближайших соседей

Подготовила студентка
группы ББМО-01-21
Концевая Юлия