МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение высшего образования
«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
Кафедра «Измерительно-вычислительные комплексы»

«Методы искусственного интеллекта»

Отчёт по лабораторной работе №5

Вариант №6

Выполнил:
студент группы ИСТбд-41
Евтушенко Александр
Проверил:
доцент кафедры ИВК, к.т.н.
Шишкин В.В.

Ульяновск
2022

Задание 1.
Ознакомится с классификаторами библиотеки Skikit-learn.

Ознакомиться с классификаторами библиотеки Scikit-learn.

Результат.

Мы ознакомились с классификаторами библиотеки Scikit-learn.

Задание 2.
Выбрать для исследования не менее трёх классификаторов.

Необходимо выбрать для исследования не менее трёх классификаторов.

Результат.

Нами были выбраны такие классификаторы:

1. Метод логистической регрессии
2. Метод опорных векторов
3. Метод k ближайших соседей

Задание 3.
Выбрать набор данных для задач классификации из открытых источников.

Из перечисленных источников необходимо выбрать набор данных для задач классификации:

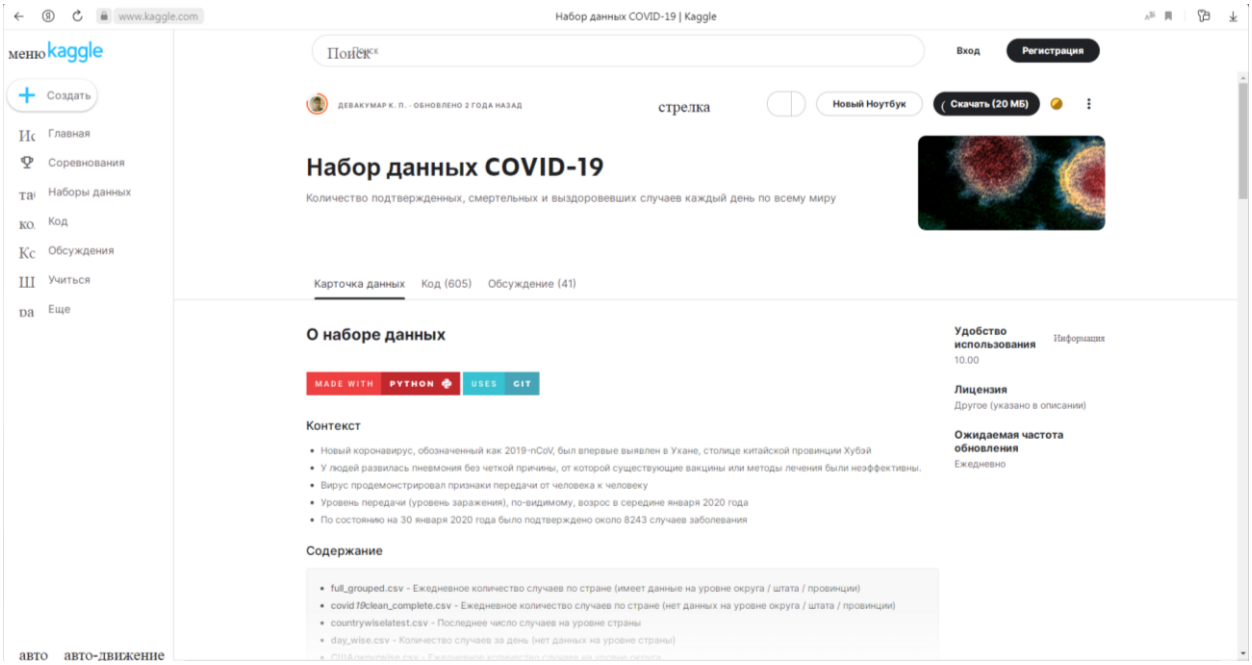
- https://tproger.ru/translations/the-best-datasets-for-machine-learning-and-data-science/
- https://vc.ru/ml/150241-15-proektov-dlya-razvitiya-navykov-raboty-s-mashinnym-obucheniem
- https://archive.ics.uci.edu/ml/index.php
- https://habr.com/ru/company/edison/blog/480408/
- https://www.kaggle.com/datasets/

Результат.

Нами был выбран dataset с сайта https://www.kaggle.com/datasets/ :

Набор данных COVID-19.

https://www.kaggle.com/datasets/imdevskp/corona-virus-report?resource=download



Задание 4.
Выбор классификаторов и набора данных утвердить у преподавателя.

Выбор классификаторов и набора данных утвердить у преподавателя.

Результат.

Выбор классификаторов и набора данных мы утвердили у преподавателя.

Задание 5.
Для каждого классификатора определить целевой столбец и набор признаков. Обосновать свой выбор. При необходимости преобразовать типы признаковых данных.

Для каждого классификатора определить целевой столбец и набор признаков. Обосновать свой выбор. При необходимости преобразовать типы признаковых данных.

Результат.

Для всех классификаторов мы выбрали столбец who_region в качестве целевого т.к. сочетание остальных параметров характеризует исследуемый регион.

Признаковые данные мы преобразовали в числовые.

Задание 6.

Подготовить данные к обучению.

Подготовить данные к обучению.

Результат.

Мы подготовили данные для обучения.

Задание 7-8.
Провести обучение и оценку моделей на сырых данных. Провести предобработку данных.

Провести обучение и оценку моделей на сырых данных. Провести предобработку данных.

Результат.

Данные пункты мы пропускаем т.к. данные уже очищены.

Задание 9.
Провести обучение и оценку моделей на очищенных данных.

Провести обучение и оценку моделей на очищенных данных.

Результат.

Мы провели обучение и оценку моделей на очищенных данных.

Задание 10.
Проанализировать результаты.

Проанализировать результаты.

```
Точность предсказаний k ближайших соседей: 31.868131868131865%
Точность предсказаний логистической регрессии: 35.16483516483517%
Точность предсказаний методом опорных векторов: 42.857142857142854%
```

Результат.

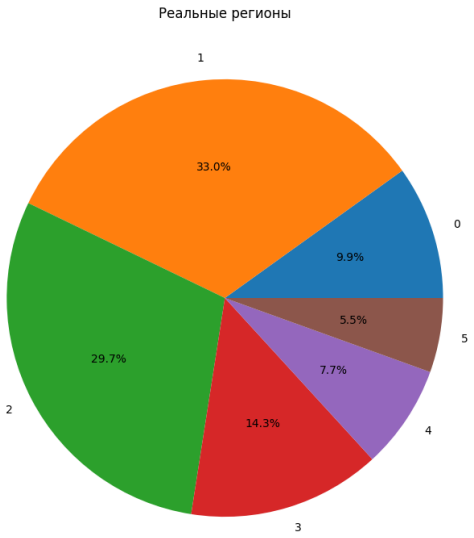
С имеющимися данными высокую эффективность показал классификатор методом опорных векторов. Методы k ближайших соседей и логистической регрессии показали чуть более низкую точность по сравнению с методом опорных векторов.

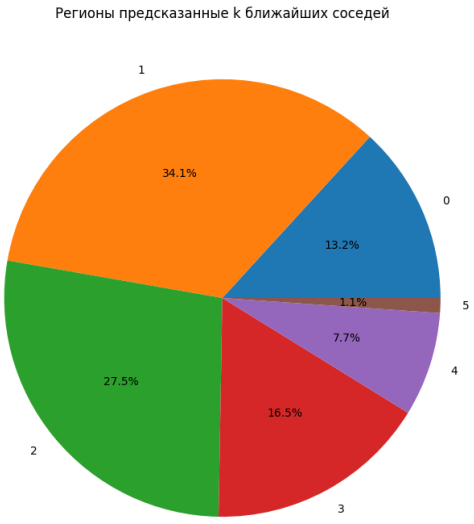
Задание 11.
Результаты анализа предоставить в табличной и графической форме.

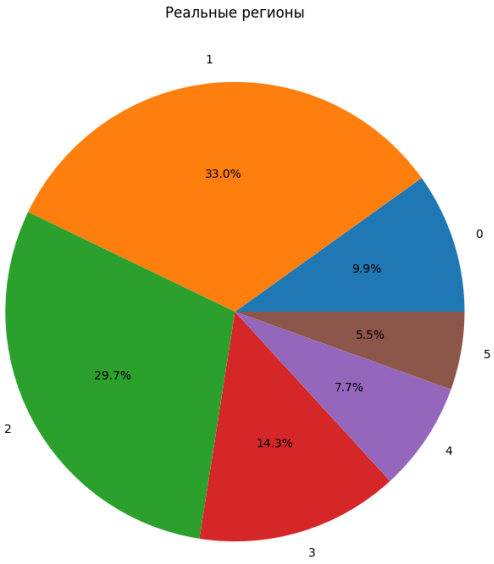
Результаты анализа предоставить в табличной и графической форме.

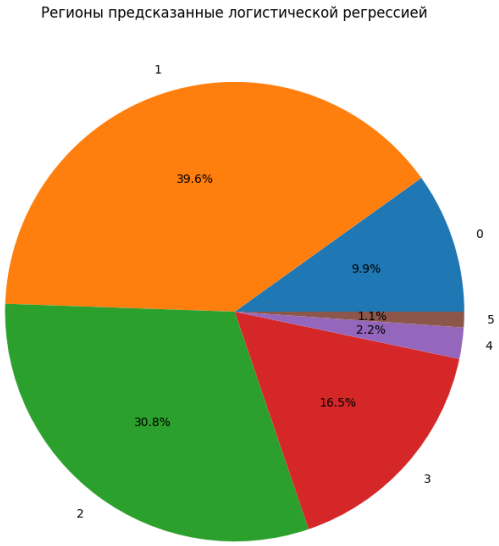
Результат.

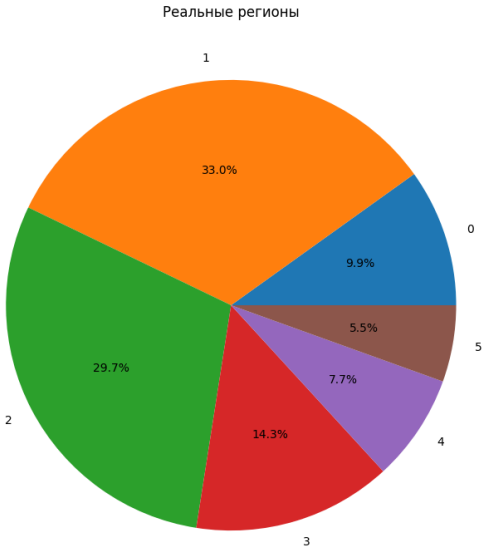
Мы предоставили результаты анализа в графической форме.

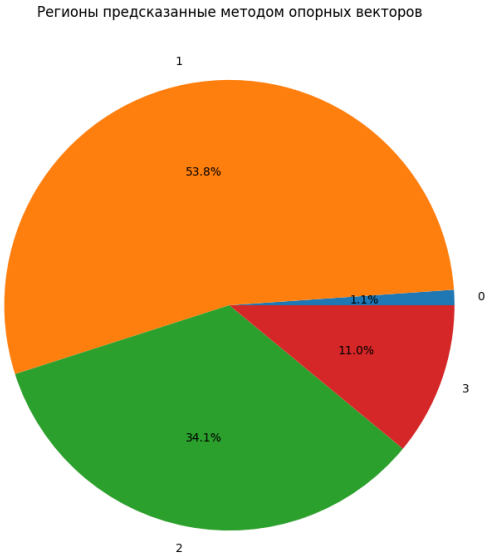












Задание 12.
Сформулировать выводы.

Сформулировать выводы.

Результат.

В ходе проведённой нами работы мы провели классификацию выбранного набора данных тремя методами классификации. В результате мы выявили, что среди: метода k ближайших соседей, метода опорных векторов и метода логистической регрессии; метод опорных векторов является самым эффективным методом классификации с учётом имеющихся данных.

Код.

```
import pandas
import matplotlib.pyplot as pyplot
import pylab
import numpy as np
from sklearn.linear_model import LogisticRegression
from sklearn.neighbors import KNeighborsClassifier
from sklearn.preprocessing import StandardScaler
from sklearn.svm import SVC
from sklearn.model_selection import train_test_split

#Reading data from csv
teach_df = pandas.read_csv('country_wise_latest.csv')
teach_df.replace([np.inf, -np.inf], np.nan, inplace=True)
teach_df.dropna(inplace=True)
teach_df['Country/Region'] = pandas.factorize(teach_df['Country/Region'])[0]
teach_df['WHO Region'] = pandas.factorize(teach_df['WHO Region'])[0]
x_teach_df , x_test_df , y_teach_df , y_test_df =
train_test_split(teach_df.drop('WHO Region', axis=1), teach_df['WHO Region'],
test_size=0.5, random_state=0)
x_teach_df = pandas.DataFrame(x_teach_df, index=x_teach_df.index,
columns=x_teach_df.columns)
x_test_df = pandas.DataFrame(x_test_df, index=x_test_df.index,
columns=x_test_df.columns)

#Scaling the data
scaler = StandardScaler()
X_teach = scaler.fit_transform(x_teach_df)
X_test = scaler.fit_transform(x_test_df)

#We run the test data through classifiers and output the accuracy of predictions

#k nearest neighbors
knn = KNeighborsClassifier(n_neighbors=4).fit(X_teach, y_teach_df)
knn_predictions = pandas.Series(knn.predict(X_test))
print('Точность предсказаний k ближайших соседей: ' + str(knn.score(X_test,
y_test_df)*100) + '%')

#Logistic regression method
lr = LogisticRegression().fit(X_teach, y_teach_df)
lr_predictions = pandas.Series(lr.predict(X_test))
print('Точность предсказаний логистической регрессии: ' + str(lr.score(X_test,
y_test_df)*100) + '%')

#The method of support vectors
svm = SVC(kernel = 'rbf').fit(X_teach, y_teach_df)
```

```
svm_predictions = pandas.Series(svm.predict(X_test))
print('Точность предсказаний методом опорных векторов: ' +
str(svm.score(X_test, y_test_df)*100) + '%')

#We draw graphs with predicted and real values of regions

#k nearest neighbors
pylab.figure(figsize=(20,10))
pylab.subplot(1, 2, 1)
pyplot.pie(y_test_df.value_counts().sort_index(), labels = sorted(y_test_df.unique()),
autopct='%1.1f%%')
pyplot.title('Реальные регионы')
pylab.subplot(1, 2, 2)
pyplot.pie(knn_predictions.value_counts().sort_index(), labels =
sorted(knn_predictions.unique()), autopct='%1.1f%%')
pyplot.title('Регионы предсказанные k ближайших соседей')
pyplot.show()

#Logistic regression method
pylab.figure(figsize=(20,10))
pylab.subplot(1, 2, 1)
pyplot.pie(y_test_df.value_counts().sort_index(), labels = sorted(y_test_df.unique()),
autopct='%1.1f%%')
pyplot.title('Реальные регионы')
pylab.subplot(1, 2, 2)
pyplot.pie(lr_predictions.value_counts().sort_index(), labels =
sorted(lr_predictions.unique()), autopct='%1.1f%%')
pyplot.title('Регионы предсказанные логистической регрессией')
pyplot.show()

#The method of support vectors
pylab.figure(figsize=(20,10))
pylab.subplot(1, 2, 1)
pyplot.pie(y_test_df.value_counts().sort_index(), labels = sorted(y_test_df.unique()),
autopct='%1.1f%%')
pyplot.title('Реальные регионы')
pylab.subplot(1, 2, 2)
pyplot.pie(svm_predictions.value_counts().sort_index(), labels =
sorted(svm_predictions.unique()), autopct='%1.1f%%')
pyplot.title('Регионы предсказанные методом опорных векторов')
pyplot.show()
```