МИНОБРНАУКИ РОССИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

ВЫСШЕГО ОБРАЗОВАНИЯ

«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

(ФГБОУ ВО «ВГУ»)

МИНОБРНАУКИ РОССИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

ВЫСШЕГО ОБРАЗОВАНИЯ

“ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ”

ФГБОУ ВО «ВГУ»

Отчет по теме

Классификация. Наборы данных

Задание №15

09.03.02 Информационные системы и технологии

Программная инженерия в информационных системах

Зав. кафедрой\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ С. Д. Махортов, д.ф-м.н. \_\_.\_\_.20\_\_

Обучающийся\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ М. Д. Шеменев, 4 курс, 5.1 группа

Руководитель\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ И.В. Илларионов, доцент

Воронеж 2023

**Задание:**

Этапы работы:

1. Выделить обучающую и тестовую выборки.

2. Определить наилучшее значение k.

3. Оценить качество прогноза на тестовой выборке с помощью таблицы сопряженности.

4. Выдать процент ошибок, допущенных классификатором на тестовой выборке.

Входные данные:

T,2,8,3,5,1,8,13,0,6,6,10,8,0,8,0,8

I,5,12,3,7,2,10,5,5,4,13,3,9,2,8,4,10

D,4,11,6,8,6,10,6,2,6,10,3,7,3,7,3,9

N,7,11,6,6,3,5,9,4,6,4,4,10,6,10,2,8

G,2,1,3,1,1,8,6,6,6,6,5,9,1,7,5,10

S,4,11,5,8,3,8,8,6,9,5,6,6,0,8,9,7

B,4,2,5,4,4,8,7,6,6,7,6,6,2,8,7,10

A,1,1,3,2,1,8,2,2,2,8,2,8,1,6,2,7

J,2,2,4,4,2,10,6,2,6,12,4,8,1,6,1,7

M,11,15,13,9,7,13,2,6,2,12,1,9,8,1,1,8

X,3,9,5,7,4,8,7,3,8,5,6,8,2,8,6,7

O,6,13,4,7,4,6,7,6,3,10,7,9,5,9,5,8

G,4,9,6,7,6,7,8,6,2,6,5,11,4,8,7,8

M,6,9,8,6,9,7,8,6,5,7,5,8,8,9,8,6

R,5,9,5,7,6,6,11,7,3,7,3,9,2,7,5,11

F,6,9,5,4,3,10,6,3,5,10,5,7,3,9,6,9

O,3,4,4,3,2,8,7,7,5,7,6,8,2,8,3,8

C,7,10,5,5,2,6,8,6,8,11,7,11,2,8,5,9  
…………………………………………….

**Код проекта:**

import numpy as np  
import pandas as pd  
from sklearn.metrics import confusion\_matrix, accuracy\_score  
from sklearn.model\_selection import train\_test\_split  
from sklearn.neighbors import KNeighborsClassifier  
  
  
FILE\_NAME = "letter-recognition.data"  
data = pd.read\_csv(FILE\_NAME, sep=',')  
  
TAG\_KEY = "T"  
OBJECT\_RANGE = range(1, 16)  
  
TEST\_SIZE = 0.2  
RANDOM\_STATE = 43  
  
objects = data.iloc[:, OBJECT\_RANGE]  
tag = data[TAG\_KEY]  
  
  
def find\_best\_num\_neighbors():  
 best\_k = None  
 best\_accuracy = 0  
  
 for k in range(1, 20):  
 knn = KNeighborsClassifier(n\_neighbors=k)  
 knn.fit(X\_train, y\_train)  
 pred = knn.predict(X\_test)  
 accuracy = accuracy\_score(y\_test, pred)  
  
 if accuracy > best\_accuracy:  
 best\_accuracy = accuracy  
 best\_k = k  
 return best\_k  
  
  
def create\_conjugacy\_table():  
 knn = KNeighborsClassifier(n\_neighbors=n\_neighbors)  
 knn.fit(X\_train, y\_train)  
 pred = knn.predict(X\_test)  
  
 conf\_matrix = confusion\_matrix(y\_test, pred)  
 print("Таблица сопряженности:", np.array(conf\_matrix))  
 return pred  
  
  
def calculate\_classifier\_error():  
 error\_rate = 1 - accuracy\_score(y\_test, y\_pred)  
 print(f"Процент ошибок: {error\_rate \* 100:.2f}%")  
  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 print(data)  
  
 X\_train, X\_test, y\_train, y\_test = train\_test\_split(  
 objects, tag, test\_size=TEST\_SIZE, random\_state=RANDOM\_STATE  
 )  
 print("Наборы данных для обучения классификатора: ")  
 print(X\_train)  
 print(y\_train)  
  
 print("Набор тестовых данных: ")  
 print(X\_test)  
 print(y\_test)  
 print()  
  
 n\_neighbors = find\_best\_num\_neighbors()  
  
 print(f"Наилучшее значение k: {n\_neighbors}")  
  
 y\_pred = create\_conjugacy\_table()  
 calculate\_classifier\_error()

**Использованные функции и библиотеки:**

*NumPy* - применяют для математических вычислений: начиная с базовых функций и заканчивая линейной алгеброй.

*Sklearn*:

model\_selection - позволяет разделить данные на обучающую и тестовую выборки.

neighbors - представляет собой реализацию алгоритма K-Nearest Neighbors для классификации.

metrics - используются для оценки качества модели

**Вывод:**

В ходе работе, мы получили наилучшее значение k = 1, построили матрицу сопряженности, а процент ошибок оказался равным 4.70, что является хорошим результатом.