Національний технічний університет України «КПІ ім. Ігоря Сікорського»

Факультет Інформатики та Обчислювальної Техніки

Кафедра інформаційних систем та технологій

Лабораторна робота №4

з дисципліни «Прикладні задачі машинного навчання»

на тему

«Класифікація методом k найближчих сусідів і набір даних Digits, частина 1»

15 варіант

Виконав:

студент групи ІС-21 Костюк А. С.

Викладач:

Нестерук А.О.

Київ – 2024

**Зміст**

[**1. Постановка задачі** 3](#_Toc164720906)

[**2. Виконання** 4](#_Toc164720907)

[**3. Висновок** 11](#_Toc164720908)

**1. Постановка задачі**

1. Для дослідження даних, візуалізуйте їх. Виведіть зображення перших 24 і 36 цифр з набору.

2. Розбийте дані на навчальні та тестові, за замовчуванням train\_test\_split резервує 75% даних для навчання і 25% для тестування, змініть це.

3. Створити та навчити модель.

4. Виконайте прогнозування класів.

5. Порівняйте прогнозовані цифри з очікуваними для перших 20, 24, 36 тестових зразків.

6. Поясніть результат, застосуйте метрики точності моделі.

6.1 Метод score оцінювача.

6.2 Матриця невідповідностей.

7. Виведіть звіт класифікації.

8. Використайте декілька моделей KNeighborsClassifier, SVC і GaussianNB для пошуку найкращої.

9. Налаштуйте гіперпараметр K в KNeighborsClassifier.

10. Зробити звіт про роботу.

**2. Виконання**

**1)** **Для дослідження даних, візуалізуйте їх. Виведіть зображення перших 24 і 36 цифр з набору.**

Зображення, що містить текст, знімок екрана, програмне забезпечення, Шрифт

Автоматично згенерований опис

Зображення, що містить текст, знімок екрана, Шрифт, програмне забезпечення

Автоматично згенерований опис

Зображення, що містить текст, електроніка, знімок екрана, число

Автоматично згенерований опис

Зображення, що містить текст, знімок екрана, програмне забезпечення, Шрифт

Автоматично згенерований опис

Перші 24 цифри:

Зображення, що містить ескіз, типографія, дизайн

Автоматично згенерований опис

Перші 36 цифр:

Зображення, що містить текст, знімок екрана, Шрифт, число

Автоматично згенерований опис

**2)** **Розбийте дані на навчальні та тестові, за замовчуванням train\_test\_split резервує 75% даних для навчання і 25% для тестування, змініть це.**

Зображення, що містить текст, знімок екрана, програмне забезпечення, Шрифт

Автоматично згенерований опис

**3)** **Створити та навчити модель.**

Зображення, що містить текст, знімок екрана, Шрифт

Автоматично згенерований опис

**4) Виконайте прогнозування класів.**

Зображення, що містить текст, Шрифт, знімок екрана, ряд

Автоматично згенерований опис

**5) Порівняйте прогнозовані цифри з очікуваними для перших 20, 24, 36 тестових зразків.**



Для перших 20 зразків:



Для перших 24 зразків:



Для перших 36 зразків:



**6) Поясніть результат, застосуйте метрики точності моделі.**

Серед перших 36 зразків лише одне значення не збіглось з очікуваним. Перевіримо точність нашої моделі більш детально за допомогою наступних методів:

**6.1) Метод score оцінювача.**

Зображення, що містить текст, Шрифт, знімок екрана, ряд

Автоматично згенерований опис

Як можна побачити з отриманого результату, модель продемонструвала доволі високу точність згідно до методу score оцінювача. Лише 2.22% прогнозів виявились хибними, що вказує на хорошу якість моделі.

**6.2) Матриця невідповідностей.**

Зображення, що містить текст, знімок екрана, монітор, програмне забезпечення

Автоматично згенерований опис

Згідно до матриці невідповідностей, 12 прогнозів виявились хибними.

**7) Виведіть звіт класифікації.**

Зображення, що містить текст, знімок екрана, програмне забезпечення

Автоматично згенерований опис

**8) Використайте декілька моделей KNeighborsClassifier, SVC і GaussianNB для пошуку найкращої.**

Зображення, що містить текст, знімок екрана, програмне забезпечення, Мультимедійне програмне забезпечення

Автоматично згенерований опис

Результат показав, що моделі KNeighborsClassifier та SVC виявились найбільш точними, причому KNeighborsClassifier виявився трохи кращим за SVC. Звідси можна зробити висновок, що ці дві моделі є найкращими для обробки наших даних. З іншого боку, модель GaussianNB показала найгірший результат, що може вказувати на те, що дана модель не підходить для нашої задачі.

**9) Налаштуйте гіперпараметр K в KNeighborsClassifier.**

Щоб визначити найкраще значення для гіперпараметра K, перевіримо точність моделі для значень К, наприклад, від 3 до 14:

Зображення, що містить текст, електроніка, знімок екрана, програмне забезпечення

Автоматично згенерований опис

Отриманий результат вказує на те, що модель досягає найбільшої точності при К = 3, отже саме це значення є найкращим для використання при роботі з нашими даними.

**3. Висновок**

В даній лабораторній роботі була розглянута класифікація методом k найближчих сусідів і набір даних Digits з використанням бібліотеки scikit-learn, обробка та візуалізація яких була здійснена за допомогою Python та його бібліотек scikit-learn та Matplotlib. Отримані знання були використані для обробки даних зображень написаних вручну чисел Digits. Обробка даних була виконана за допомогою бібліотеки scikit-learn, а саме використовуючи класифікатори SVC, KNeighborsClassifier та GaussianNB. Отримані результати показали, що класифікатор KNeighborsClassifier виявився найбільш точним на цьому наборі даних. Більш детальні результати роботи всіх класифікаторів представлені в лабораторній.