Міністерство освіти і науки України

Львівський національний університет імені Івана Франка

Факультет електроніки та комп’ютерних технологій

Кафедра оптоелектроніки

Звіт

Про виконання лабораторної роботи №5

З курсу «Системи машинного навчання»

Нейронні мережі та PyTorch/Keras

Виконав:

Студент групи ФеС-32

Бойко Кирило

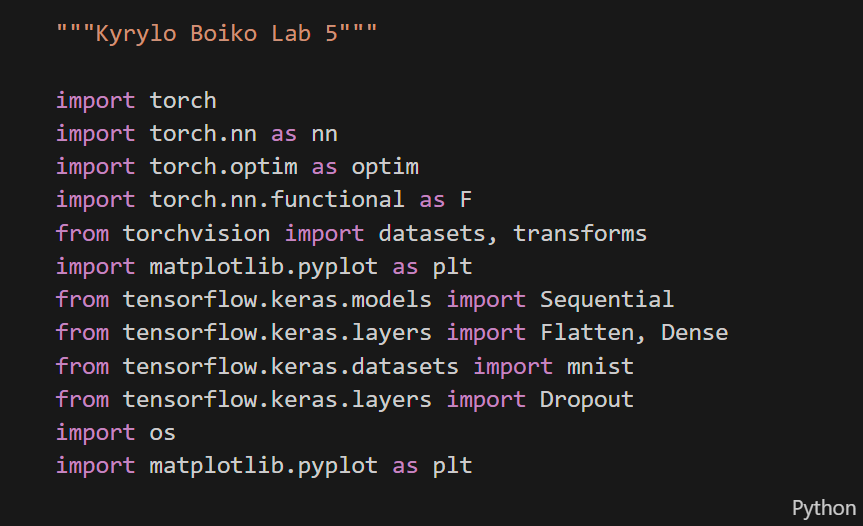
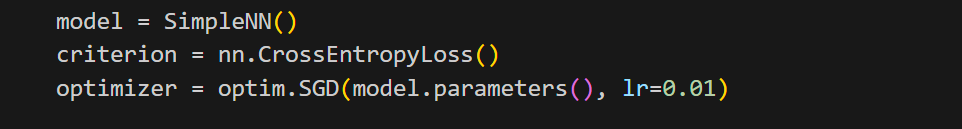
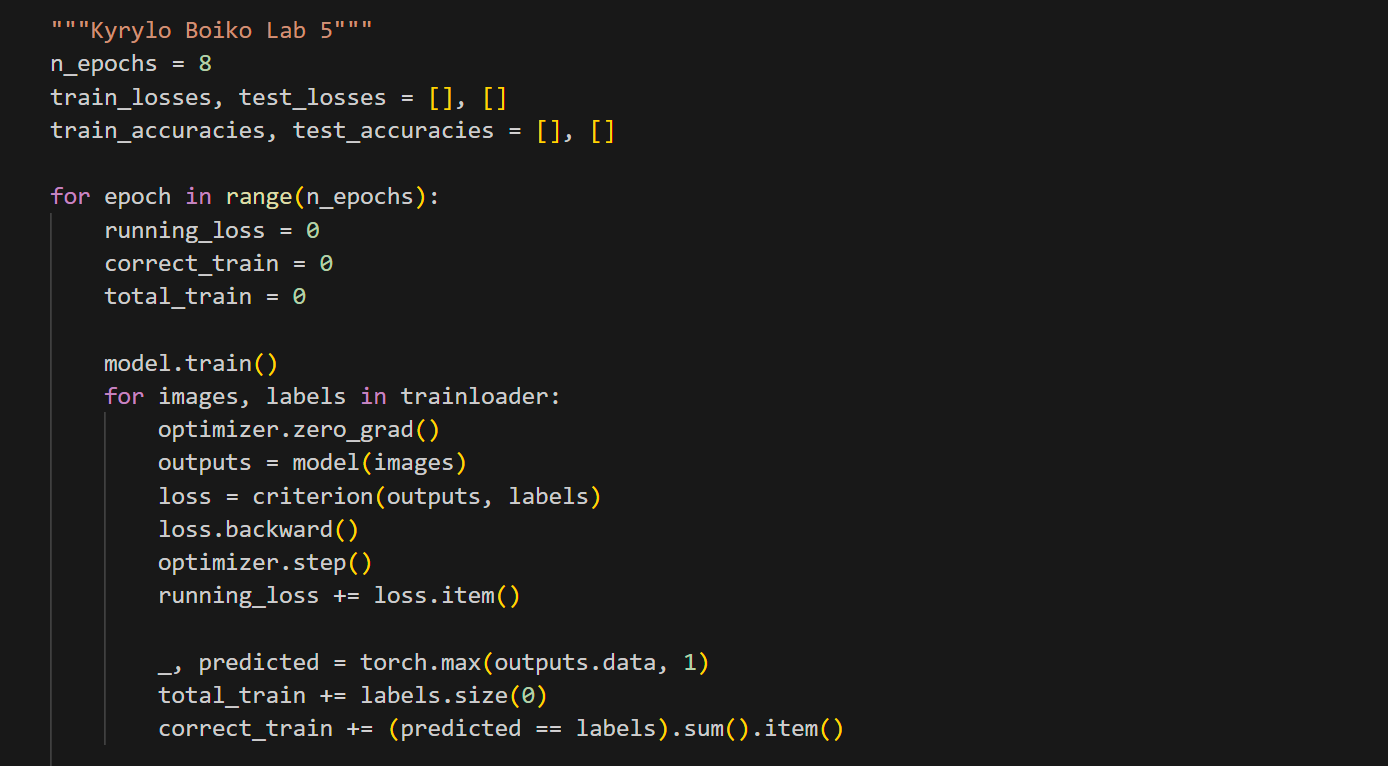
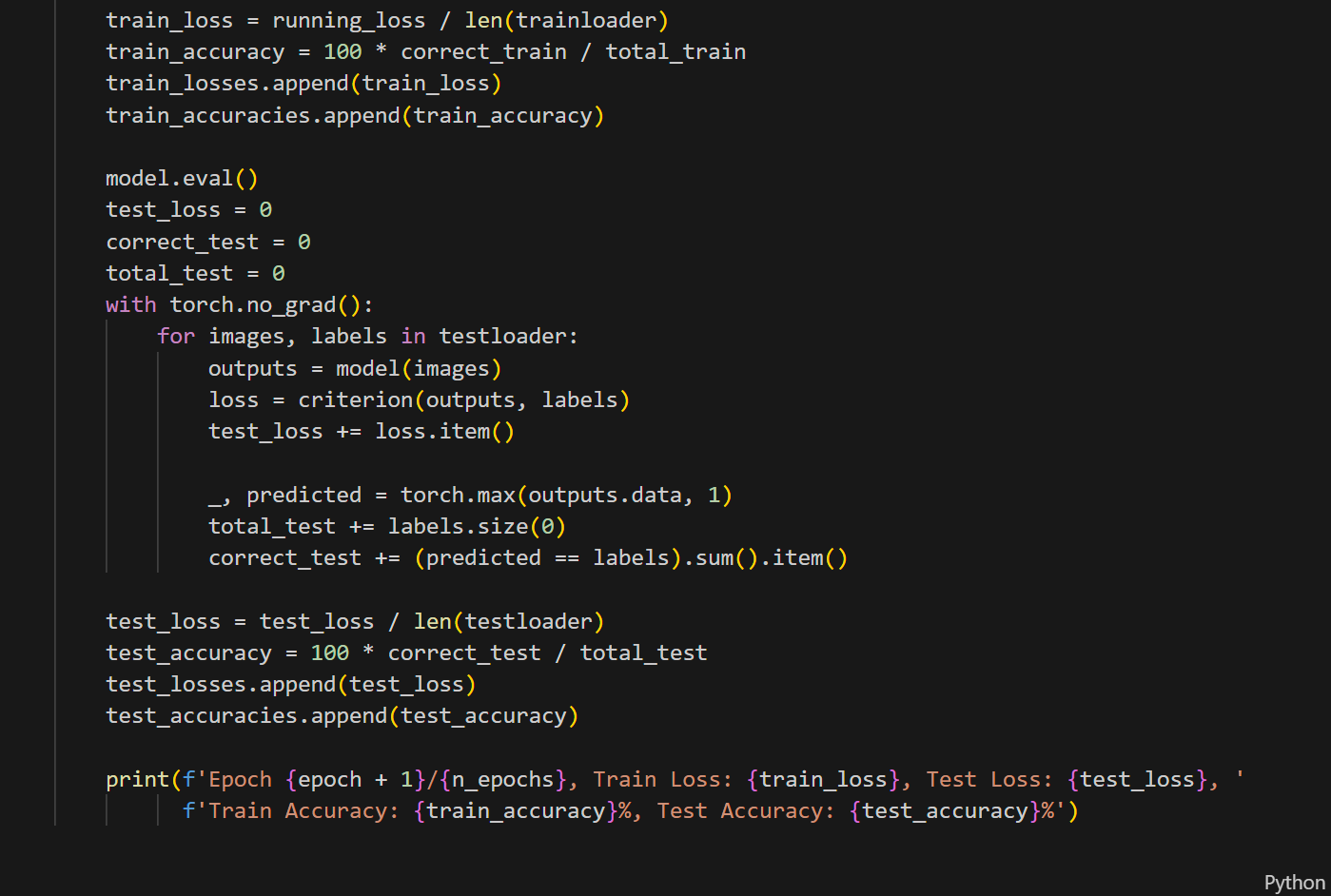
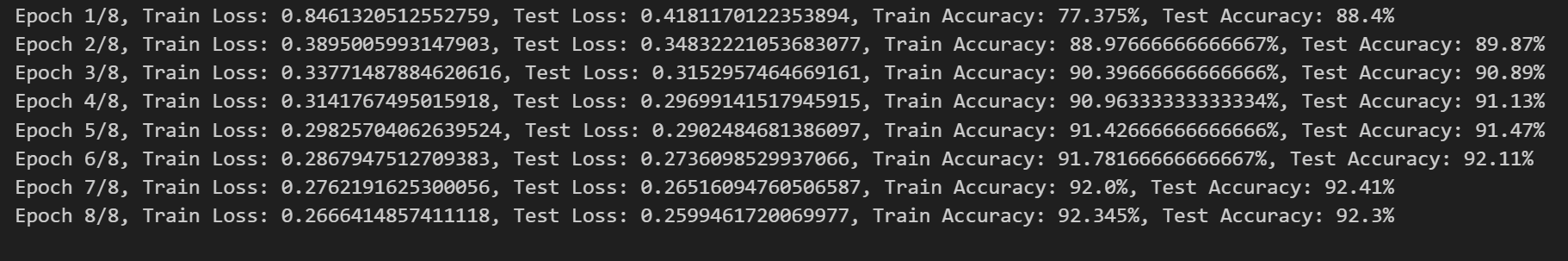
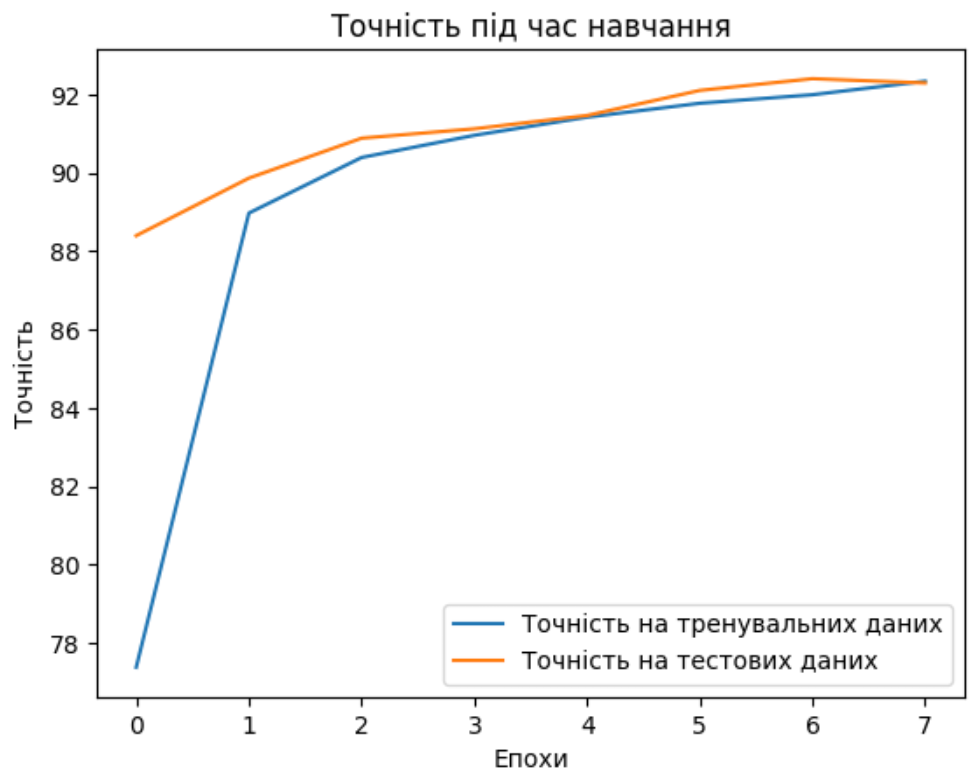
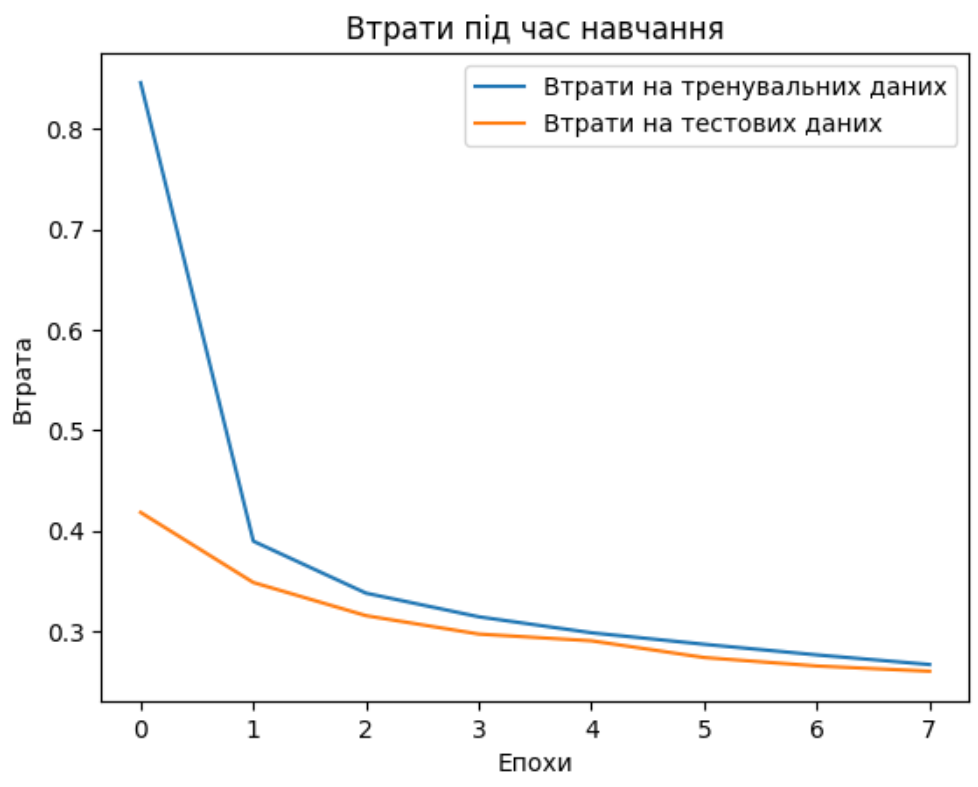
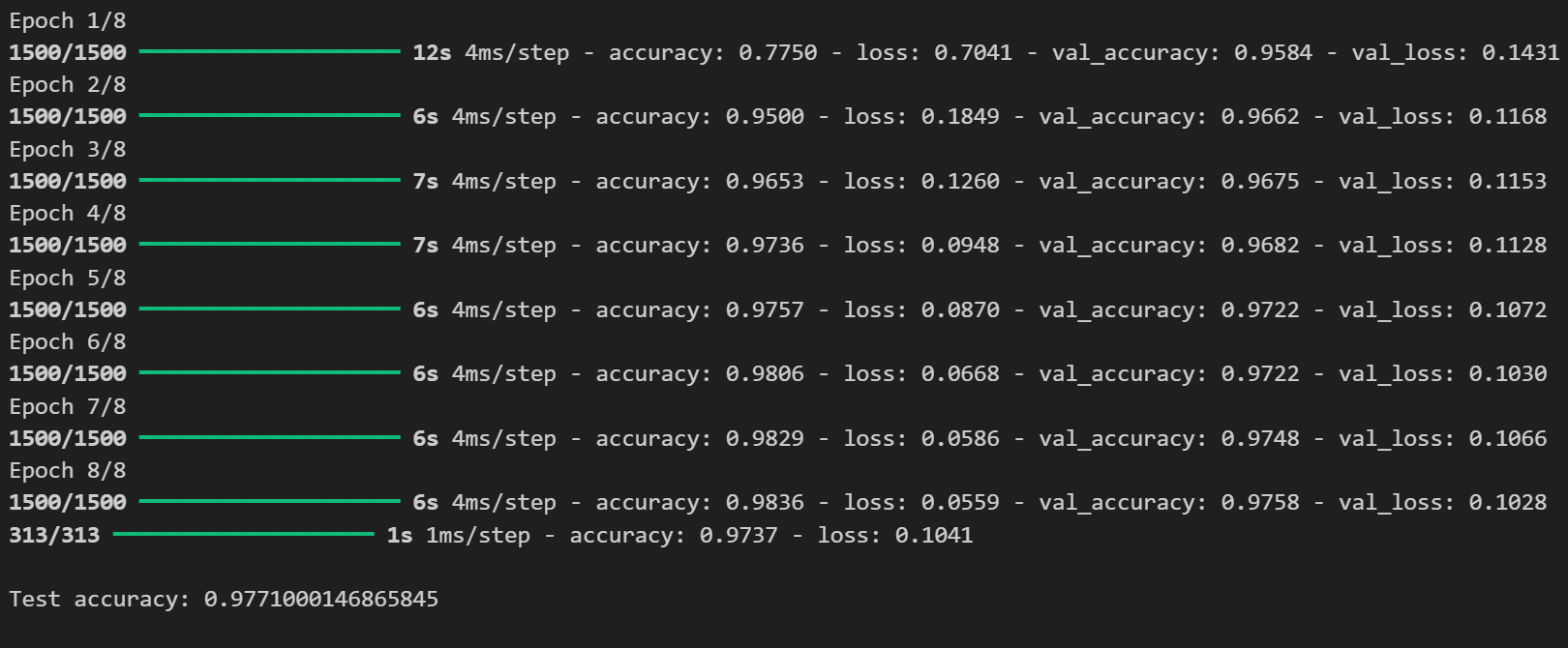
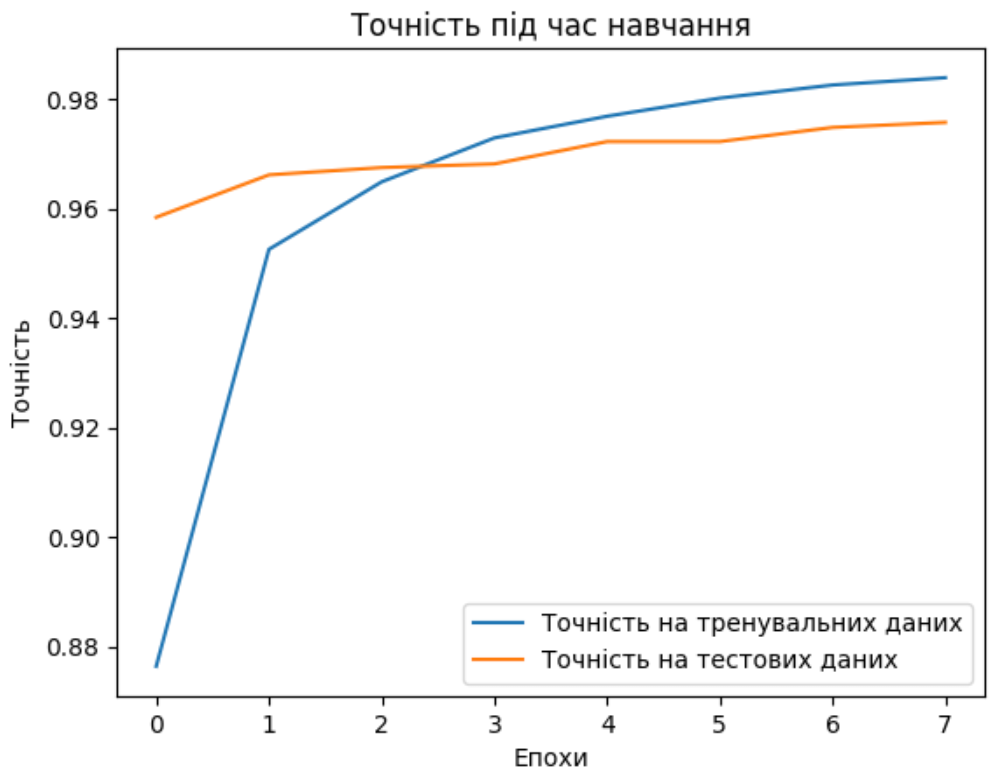
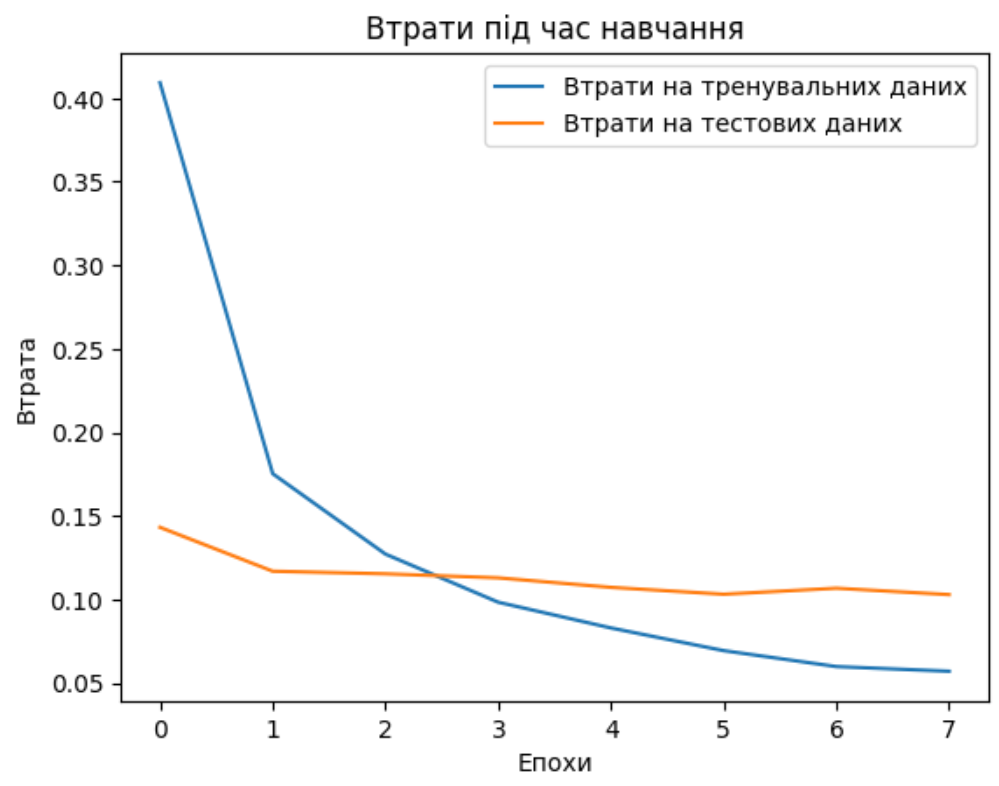
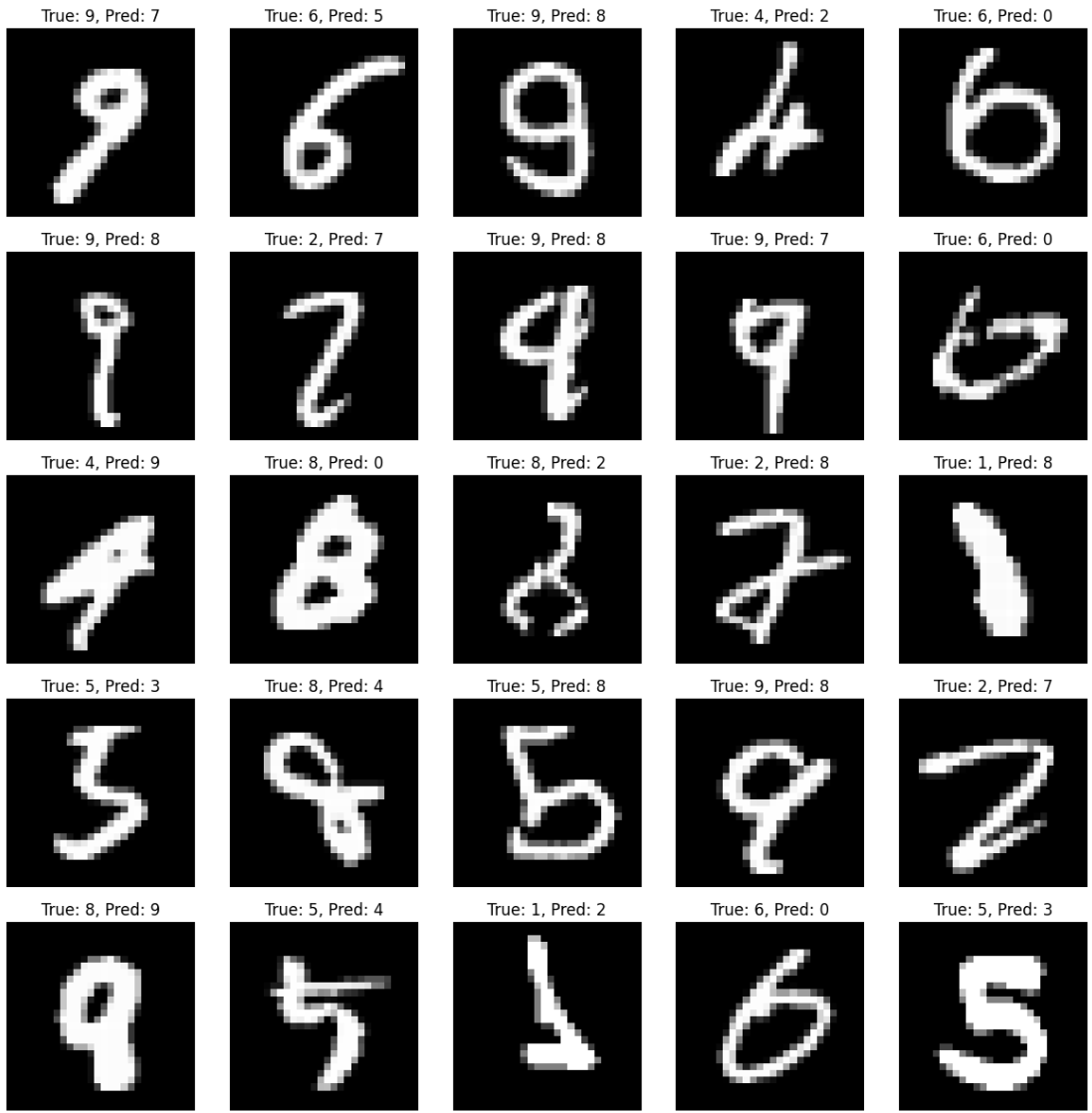
Перевірив:

Доцент Колич І.І.

Львів 2024

**Мета:** навчитися будувати та навчати прості нейронні мережі для задач класифікації та регресії.

**Хід роботи:**

1. Підготовка середовища
   1. Встановіть необхідні бібліотеки, якщо вони ще не встановлені: PyTorch, Keras, Matplotlib
   2. Імпортуйте необхідні бібліотеки в Python.
2. Створення простої нейронної мережі, що відмінна від прикладу
   1. Створіть та налаштуйте модель нейронної мережі за допомогою Keras (або PyTorch):
      * Використовуйте послідовну модель (Sequential).
      * Додайте шари до моделі (Dense). Кількість шарів має бути іншою, ніж у прикладі
      * Вкажіть кількість нейронів у кожному шарі та функції активації
   2. Сконфігуруйте модель для навчання (в прикладі позначено <…>):
      * Виберіть алгоритм оптимізації (наприклад, sgd).
      * Вкажіть функцію втрат (наприклад, categorical\_crossentropy для задач класифікації).
      * Вкажіть метрику для оцінки моделі (наприклад, accuracy).
3. Навчання нейронної мережі на наборі даних
   1. Завантажте та підготуйте дані:
      * Використайте набір даних, наприклад, Fashion-MNIST.
      * Розділіть дані на тренувальну та тестову вибірки.
      * Нормалізуйте дані, якщо необхідно.
   2. Навчіть модель на тренувальних даних:
      * Вкажіть кількість епох.
      * Вкажіть розмір пакета (batch size).
      * Збережіть історію навчання для подальшої візуалізації.
4. Оцінка та візуалізація результатів
   1. Оцініть модель на тестових даних:
      * Використайте метод оцінки (evaluate) для отримання метрик якості.
   2. Візуалізуйте результати навчання та оцінки:
      * Побудуйте графіки втрат та точності на тренувальних та тестових даних
      * Візуалізуйте помилково класифіковані приклади.
5. Оптимізація гіперпараметрів
   1. Експериментуйте з різними параметрами моделі, щоб підвищити її точність:
      * Змініть кількість шарів та нейронів.
      * Використайте різні функції активації.
      * Спробуйте різні алгоритми оптимізації.
   2. Порівняйте результати:
      * Запишіть результати експериментів.
      * Проаналізуйте, як зміна параметрів впливає на якість моделі.
6. Оформлення звіту:
   1. Завантаження бібліотек:  
      
   2. Розробка класу для простої нейронної мережі, завантаження набору даних Fashion-MNIST, його нормалізація та поділ на тренувальну і тестову вибірки:  
      
   3. Вибір для моделі простої нейронної мережі, функції для класифікації та оптимізатора:  
      
   4. Тренування моделі простої нейронної мережі, 8 епох:  
      
   5. Оцінювання моделі:  
      
   6. Результати навчання моделі за епохами показують, що з підвищенням кількості епох точність моделі збільшується:  
      
   7. Графіки, що відображають точність та втрати відносно епохи тренування простої нейронної мережі:  
      
   8. Створення та навчання нейронної мережі з послідовними шарами, де як бекенд обрано "torch", хоча це вже застаріла версія; наразі частіше використовується tensorflow. Для оптимізації застосовано оптимізатор adam, а кількість нейронів становить 8:  
      
   9. Результат тренування моделі на 8 епохах: 
   10. Графіки, що відображають точність та втрати відносно епохи тренування послідовної нейронної мережі:  
       
   11. Неправильно класифіковані значення в моделі послідовних шарів:  
       

**Висновок:** Виконавши дану лабораторну роботу, я навчився тренувати прості та послідовні нейронні мережі, а також налаштовувати їх гіперпараметри.