**ДНІПРОВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ОЛЕСЯ ГОНЧАРА**

**Факультет прикладної математики**

**Кафедра математичного забезпечення ЕОМ**

**Звіт**

з лабораторної роботи № 1

з курсу «Deep Learning»

на тему: «Ознайомитися з можливостями хмарного середовища для навчання нейронних мереж Google Colaboratory»

Виконав:

студент гр. ПЗ-21у-1

Войцехов М.О.

м. Дніпро

2023 р.

**Постановка задачі**

1. Ознайомитися з можливостями хмарного середовища для навчання нейронних мереж Google Colaboratory або його аналогів. Вміти завантажувати дані, запускати код на виконання, обирати GPU або TPU тощо.

2. Ознайомитися з сайтом https://www.tensorflow.org/. Розуміти, як знайти необхідну інформацію.

3. Обрати набір даних для задачі навчання з вчителем (класифікація, регресія), з яким вам буде цікаво навчати нейронну мережу.

Можливі варіанти:

1. Працювати з розглянутими на заняттях прикладами

2. Знайти інші приклади у tutorials tensorflow тощо

3. Завантажити датасет застосовуючи tfds

4. Знайти датасети зі сторонніх ресурсів, наприклад:

1) archive.ics.uci.edu/ml/datasets

2) kaggle.com/datasets

Здійснити навчання нейронної мережі на обраних у п.2 даних. Провести експерименти та спробувати підібрати найоптимальніші для вашого датасету гіперпараметри (кількість шарів, кількість нейронів, метод навчання, швидкість навчання (різні варіанти налаштування), розмір батчу, функції активації, застосування dropout, регуляризації тощо). Усі результати проведених вами експериментів мають бути збережені у коді.

Для виконання роботи було обрано дані Auto MPG (<https://archive.ics.uci.edu/dataset/9/auto+mpg>).

Далі загрузив датасет, з усіма стовпцями, видалив усі рядки з пустими даними. Значення стовпців перевів з числових у строковий та розбив дані на навчалий та тестовий набор.

url = 'http://archive.ics.uci.edu/ml/machine-learning-databases/auto-mpg/auto-mpg.data'

column\_names = ['MPG', 'Cylinders', 'Displacement', 'Horsepower', 'Weight',

                'Acceleration', 'Model Year', 'Origin']

raw\_dataset = pd.read\_csv(url, names=column\_names,

                          na\_values='?', comment='\t',

                          sep=' ', skipinitialspace=True)

dataset = raw\_dataset.copy()

dataset = dataset.dropna()

dataset['Origin'] = dataset['Origin'].map({1: 'North America', 2: 'Europe', 3: 'Asia'})

dataset = pd.get\_dummies(dataset, columns=['Origin'], prefix='', prefix\_sep='')

train\_dataset = dataset.sample(frac=0.8, random\_state=0)

test\_dataset = dataset.drop(train\_dataset.index)

train\_features = train\_dataset.copy()

test\_features = test\_dataset.copy()

train\_labels = train\_features.pop('MPG')

test\_labels = test\_features.pop('MPG')

train\_dataset.describe().transpose()[['mean', 'std']]

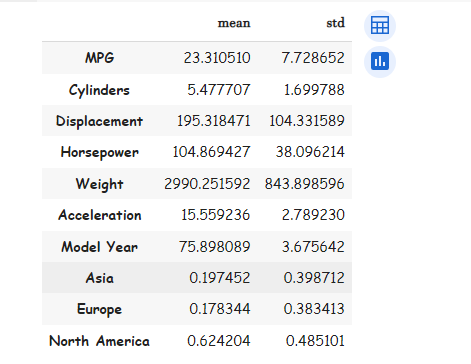


Рисунок 1 – Виведення даних

Наступним кроком створив нормалізатор на основі навчальних даних, вимірюючи середні значення ознак, щоб потім використовувати ці параметри для нормалізації даних у процесі навчання моделі. Та вивів середнє значення кожної ознаки.

normalizer = tf.keras.layers.Normalization(axis=-1)

normalizer.adapt(np.array(train\_features))

print(normalizer.mean.numpy())



Рисунок 2 – Середнє значення кожної ознаки

Далі написав для прогнозування значення MPG на основі характеристик автомобілів, таких як потужність двигуна.

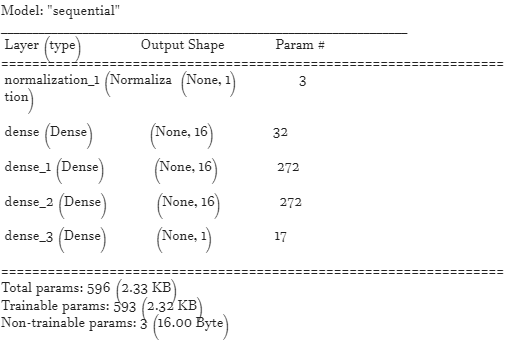


Рисунок 3 – Прогнозування значення

Побудував графік для візуалізації динаміки зміни помилки в процесі навчання нейронної мережі.

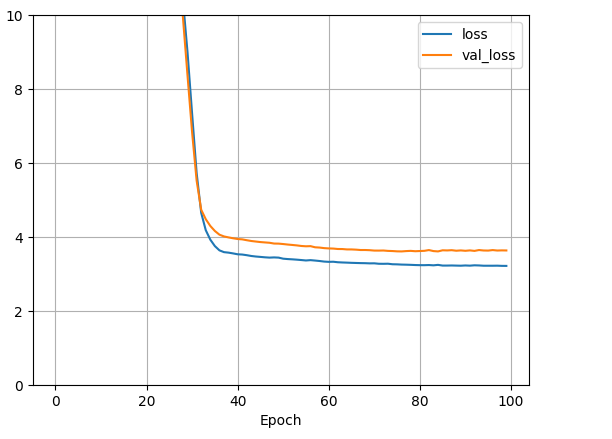


Рисунок 4 – Графік

Візуалізіровал взаємозв’язок між потужністю автомобіля та ефективності використання палива.

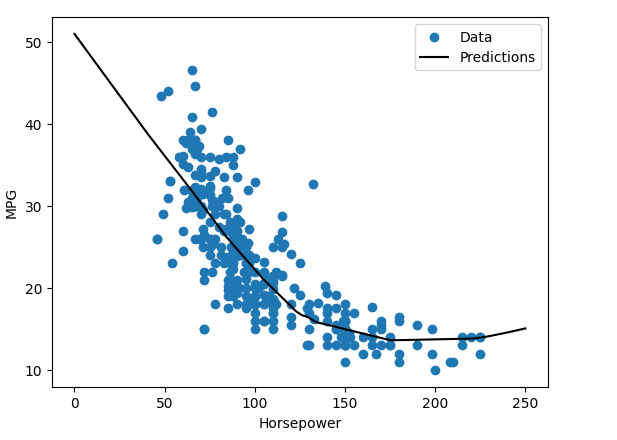


Рисунок 5 – Взаємозв’язок

Зведеної інформації про модель нейронної мережі в TensorFlow і Keras вивів у вигляді архітектури моделі.

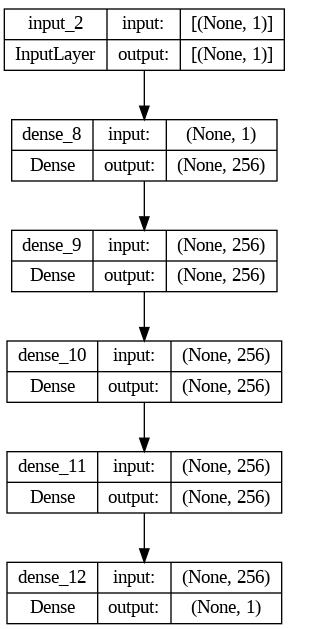


Рисунок 6 – Архітектура моделі

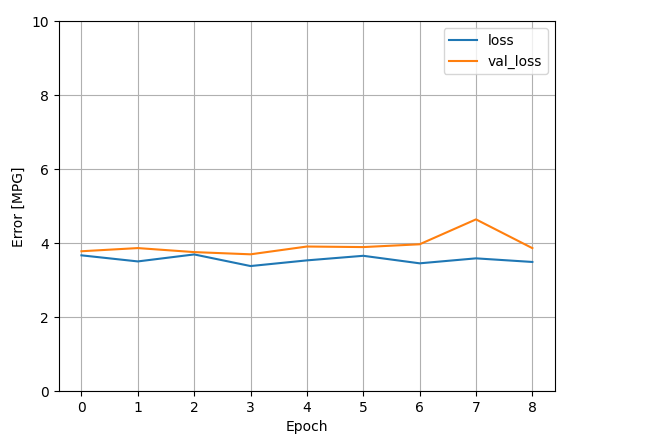


Рисунок 7 – Графік втрати

В кінці вивів графік, який наглядно показує здатність моделі прогнозувати витрату палива на основі потужності двигуна.

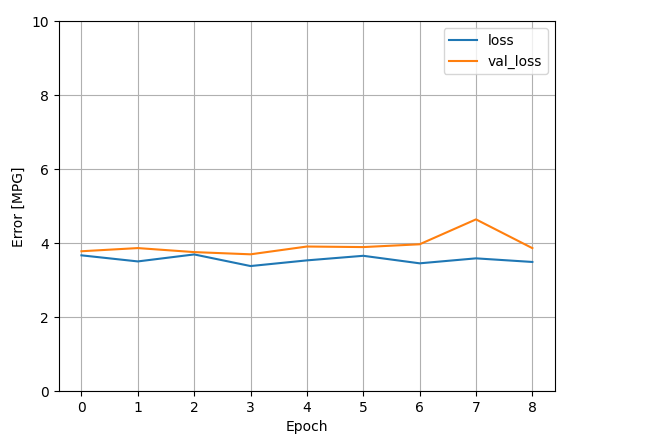


Рисунок 8 - Графік прогнозування