Отчет

**Введение**

**Описания задач:**

|  |
| --- |
| 1. Проведите эксперименты - измените параметры сети. Попробуйте получить лучшие результаты (уменьшить количество ошибок, время обучения и т.п.). При каких параметрах сеть практически "ломается", очень плохо распознает?  3\*. Считается, что сети прямого распространения - не лучший вариант для распознавания. Возьмите другую сеть. Сравните результаты.  4\*. Сделайте программу распознавания букв. Получился полный аналог или что-то пришлось изменить? |

**Выбранный набор данных:**

|  |  |
| --- | --- |
| Задача 1 | Датасет с MNIST:  <https://media.githubusercontent.com/media/freebz/Make-Your-Own-Neural-Network/master/mnist_dataset/mnist_train_100.csv>  https://media.githubusercontent.com/media/freebz/Make-Your-Own-Neural-Network/master/mnist\_dataset/mnist\_test\_10.csv |
| Задача 3,4 | Датасет со ссылки:  https://www.kaggle.com/datasets/dhruvildave/english-handwritten-characters-dataset?resource=download. |

**Методы класстеризации:**

1. DFN-algorithm

3. CNN-algorithm from tensorflow module

4. DFN-algorithm as the same as 1-question

|  |
| --- |
| from tensorflow.keras import layers,models |

**Использованные модули:**

Задача 1 (Hand wright digit recognition)

|  |
| --- |
| import numpy as np  import scipy  import urllib.request as req  import csv  import codecs  import matplotlib.pyplot as plt |

Задача 3 (Letter recognize(CNN-model))

|  |
| --- |
| import numpy as np  import csv  from sklearn.model\_selection import train\_test\_split  import pandas  from PIL import Image  from tensorflow.keras import layers,models |

Задача 4 (Hand wright eng letter regcognition)

|  |
| --- |
| import numpy as np  import scipy  import csv  import codecs  from sklearn.model\_selection import train\_test\_split  import pandas  from PIL import Image  import matplotlib.pyplot as plt |

\*В задаче 1,4 не использовал готовые модули сетей

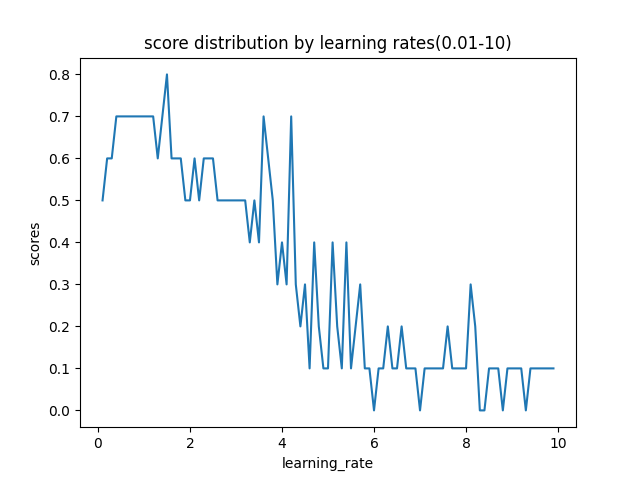
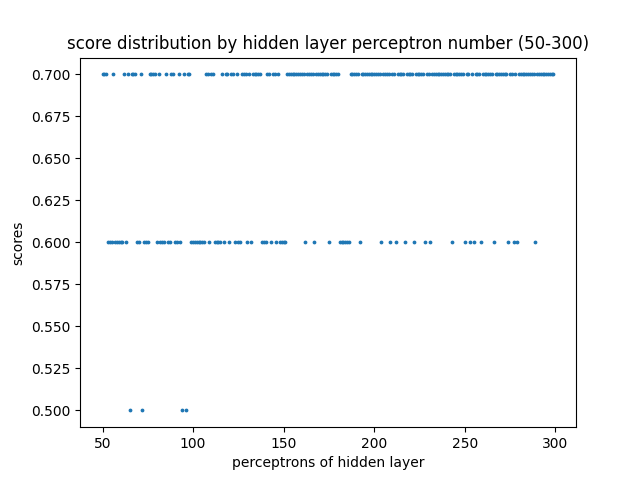
**Результат работы**

**Performance**

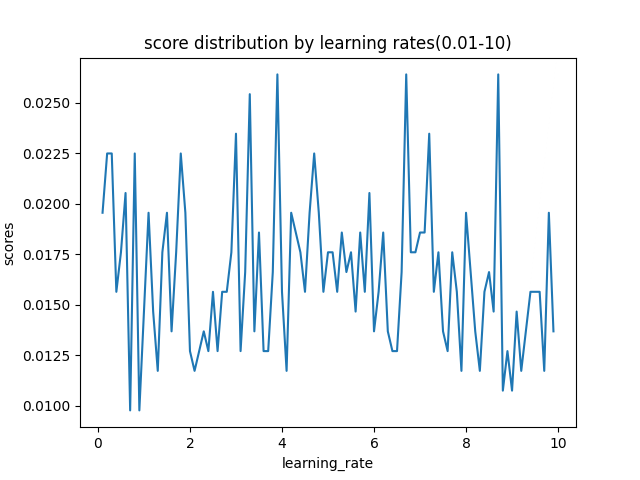
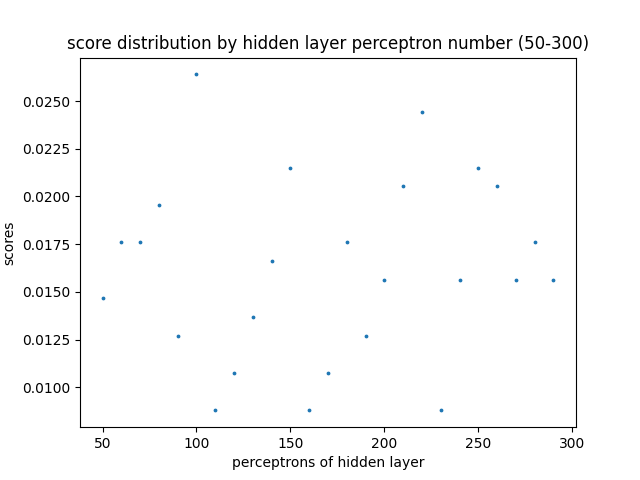
|  |  |  |
| --- | --- | --- |
| Задача 1 | IN\_node=784 / HI\_node=100 / OUT\_node=10 / learning\_rate=0.3 | 0.7 |
| Задача 3 | epochs=50, optimizer=adam | 0.5972629785 |
| Задача 4 | IN\_node=729 / HI\_node=300 / OUT\_node=62 / learning\_rate=0.0001 | 0.0136852394 |

**Графики:**

Задача 1



Задача 4



**Вывод**

Задача 1

|  |
| --- |
| В случае когда learning rate была в диапозоне 0.1-1.9 получил лучщий результат. Изменение кол-во скрытой ноды не влияло на результат. Хотелось бы посмотреть в случае больших количеств, но мой компьютер не мог поддерживать.  В случае когда learning rate была больше 2, результат начал ломаться. |

Задача 4

|  |
| --- |
| Использовал тотже самый алгоритм как в задаче 1. Получил очень плохой ответ. Пытался получить лучший ответ с изменением коэффициента нейрона, но как показывается в графике, получил без смысленный результат. Пришлось выбрать новый модель, либо исправить сам класс. Решил выбрать новый модель. |

Задача 3

|  |
| --- |
| Выбрал новый модель CNN(Convolutional Neural Network). Т.к. она очень эффективна к распознанию картинок. Получил performance 0.5-0.7. И скорость обучения стало очень быстрым. (Использавал оптимизатор adam) |