Рекомендации по построению графика ошибки обучения в Задании №1

В функции тренировки изменить окончание, добавив возврат ошибки:

# тренировка нейронной сети

    def train(self, inputs\_list, targets\_list):

изменить окончание:

**return output\_errors**  # возврат ошибок выходного слоя

 # pass

Внести изменения в цикл обучения:

# тренировка нейронной сети

# переменная epochs указывает, сколько раз тренировочный

# набор данных используется для тренировки сети

epochs = 25

**# Добавить**

error\_epochs = [] # ошибки за эпоху

for e in range(epochs):

    # перебрать все записи в тренировочном наборе данных

    sum\_error = 0   # суммирование ошибок по всем записям

    for record in training\_data\_list:

        # получить список значений из записи, используя символы

        # запятой ',' в качестве разделителей

        all\_values = record.split(',')

        # масштабировать и сместить входные значения

        inputs = (numpy.asfarray(all\_values[1:]) / 255.0 \* 0.99) + 0.01

        # создать целевые выходные значения (все равны 0.01,

        # за исключением желаемого маркерного значения, равного 0.99)

        targets = numpy.zeros(output\_nodes) + 0.01

        # all\_values[0] - целевое маркерное значение для данной записи

        targets[int(all\_values[0])] = 0.99

**# Измененить и добавить**

        errors = n.train(inputs, targets) # ошибки вых. слоя на тек. эпохе

        e = numpy.linalg.norm(errors) # норма L2 вектора ошибок (СКО)

        sum\_error = sum\_error + e     # суммир. ошибок по всем записям

        pass

    error\_epochs.append(sum\_error/60)  # массив СКО за каждую эпоху

    pass

# рисуем график

matplotlib.pyplot.plot(error\_epochs)

# показываем график

matplotlib.pyplot.show()

