

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ
по лабораторной работе №4
по дисциплине «Искусственные нейронные сети»
Тема: Распознавание рукописных символов

Студент гр. 7382

Гиззатов А.С.

Преподаватель

Жукова Н.А.

Санкт-Петербург

2020

Цель работы.

Реализовать классификацию черно-белых изображений рукописных цифр (28x28) по 10 категориям (от 0 до 9).

Порядок выполнения работы.

1. Ознакомиться с представлением графических данных.
2. Ознакомиться с простейшим способом передачи графических данных нейронной сети.
3. Создать модель.
4. Настроить параметры обучения.
5. Написать функцию, позволяющая загружать изображение пользователя и классифицировать его.

Требования к выполнению задания.

1. Найти архитектуру сети, при которой точность классификации будет не менее 95%.
2. Исследовать влияние различных оптимизаторов, а также их параметров, на процесс обучения.
3. Написать функцию, которая позволит загружать пользовательское изображение не из датасета.

Основные теоретические положения.

MINIST – база данных рукописных цифр, имеющая подготовленный набор обучающих значений в размере 60000 примеров и тестовых значений из 10000 примеров. Это подмножество более широкого набора, доступное из NIST. Наш набор состоит из изображений размером 28x28, каждый пиксель которого представляет собой оттенок серого, цифры нормализованы по размеру и имеют фиксированный размер изображения. Таким образом, есть 10 цифр (от 0 до 9) или

10 классов для прогнозирования. Результаты сообщаются с использованием ошибки прогнозирования, которая является не чем иным, как инвертированной точностью классификации.

Ход работы.

Была создана модель для распознавания рукописных цифр. При обучении модели были взяты 4 оптимизатора:

- Adam
- RMSprop
- SGD
- Adagard

и было использовано несколько значений `learning_rate`, для определения самой лучшей комбинаций для обучения нейронной сети. Архитектура сети:

- Инициализация весов – normal.
- Epochs = 10, batch_size = 128, loss = categorical_crossentropy
- Слои:

```
model.add(Dense(128, activation='relu', input_shape=(784,)))  
model.add(Dense(100, activation='relu'))  
model.add(Dense(10, activation='softmax'))
```

1. Оптимизатор Adam

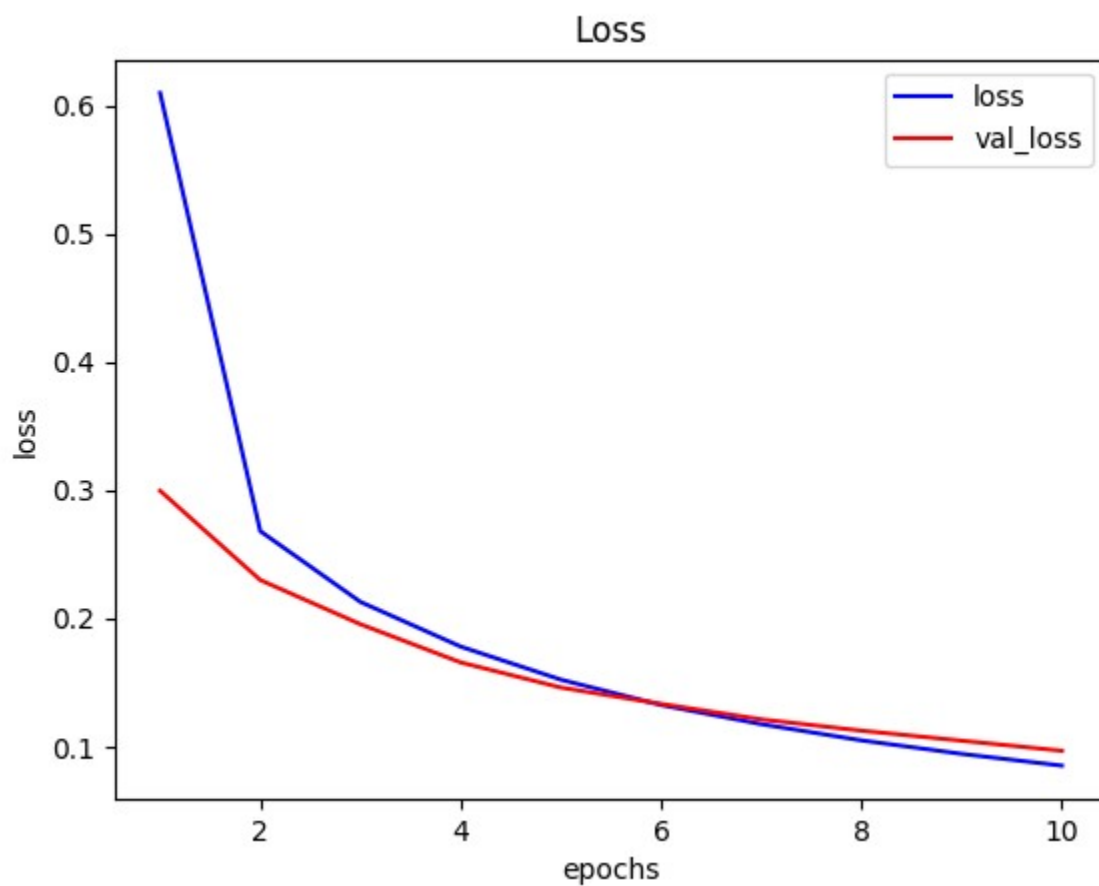


Рис.1 – Потери нс, learning_rate = 0.0002

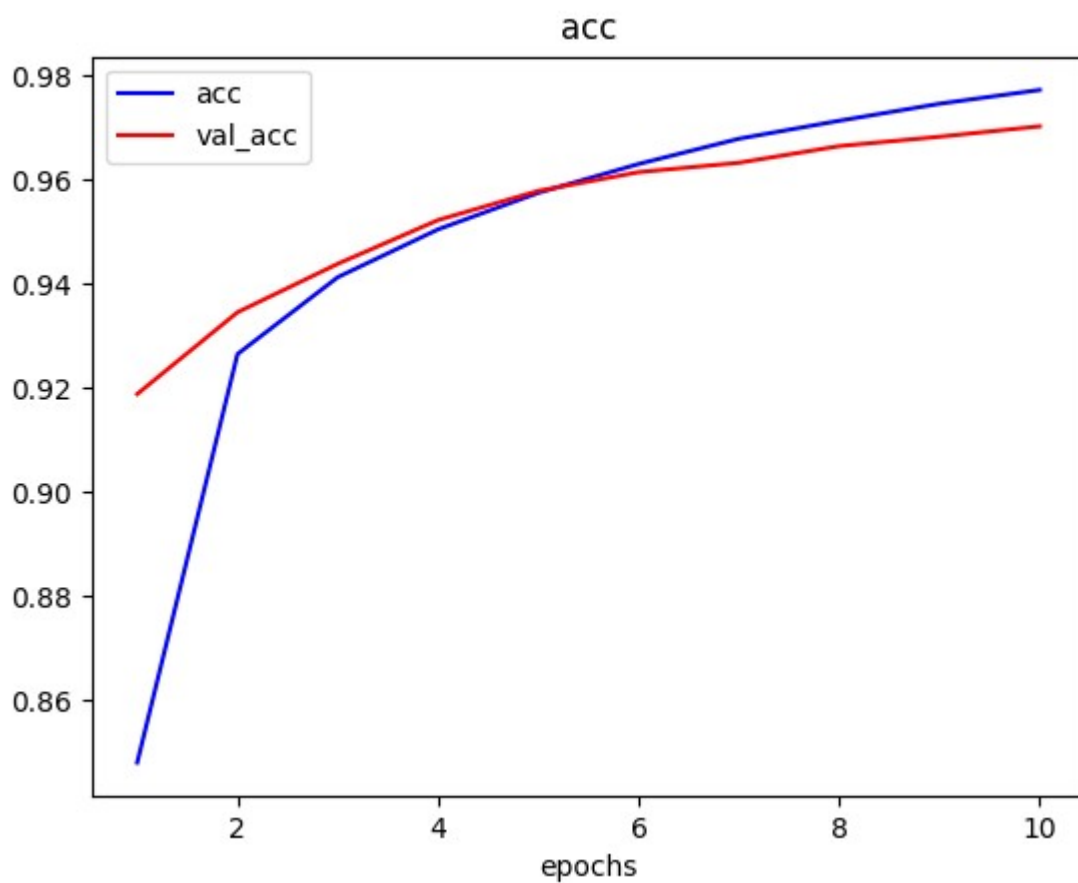


Рис.2 – Точность нс, learning_rate = 0.0002

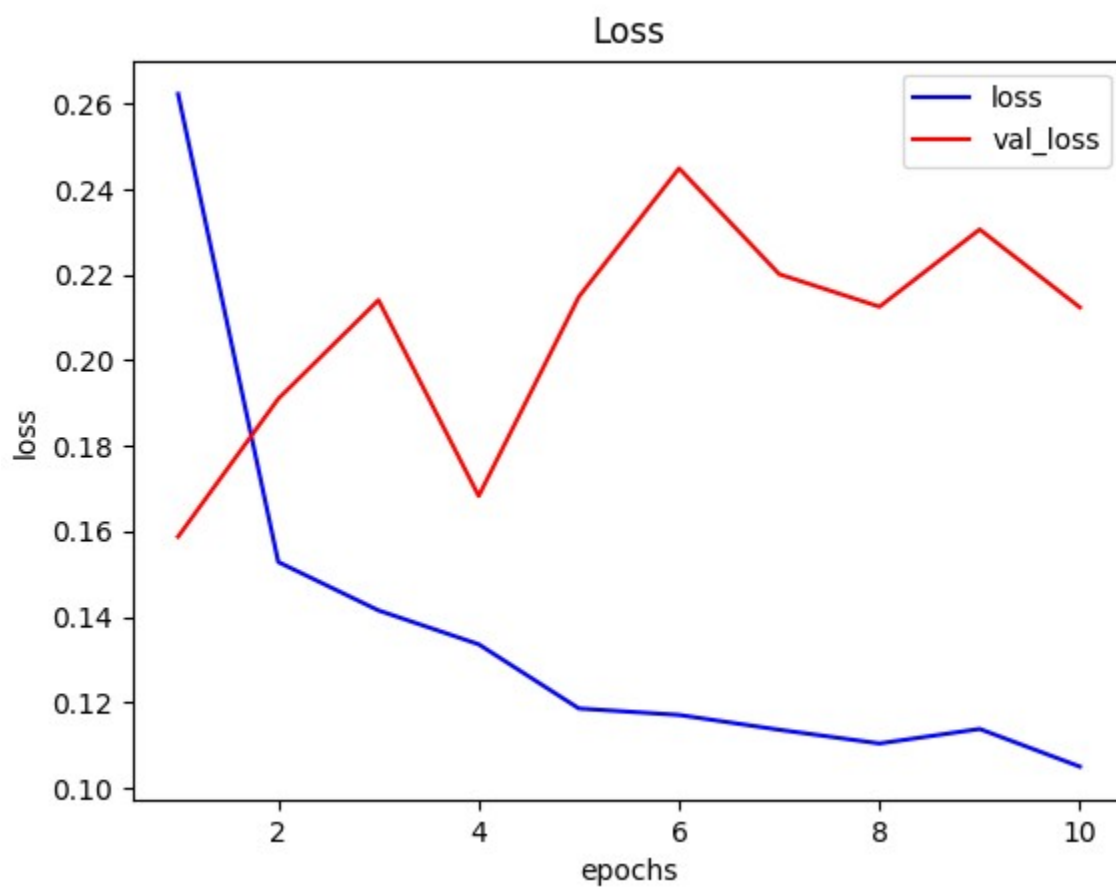


Рис.3 – Потери нс, learning_rate = 0.02

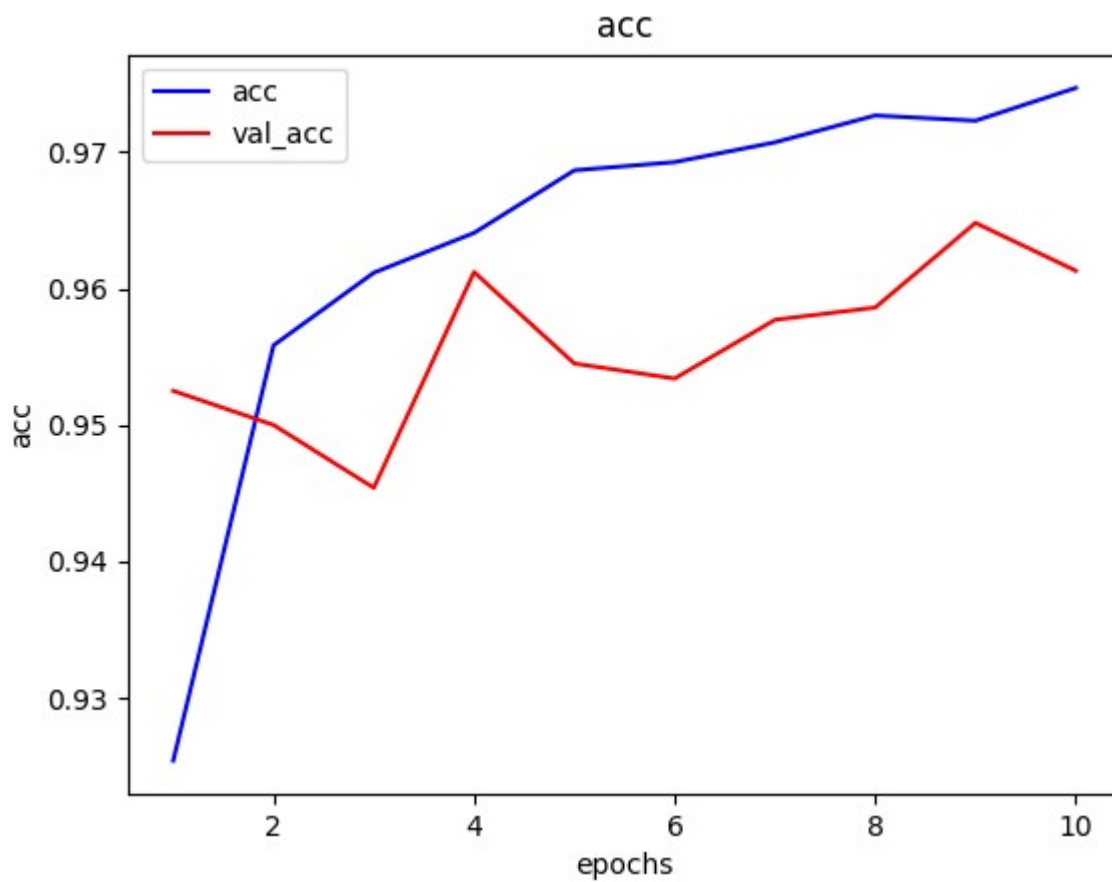


Рис.4 – Точность нс, learning_rate = 0.02

2. Оптимизатор RMSprop

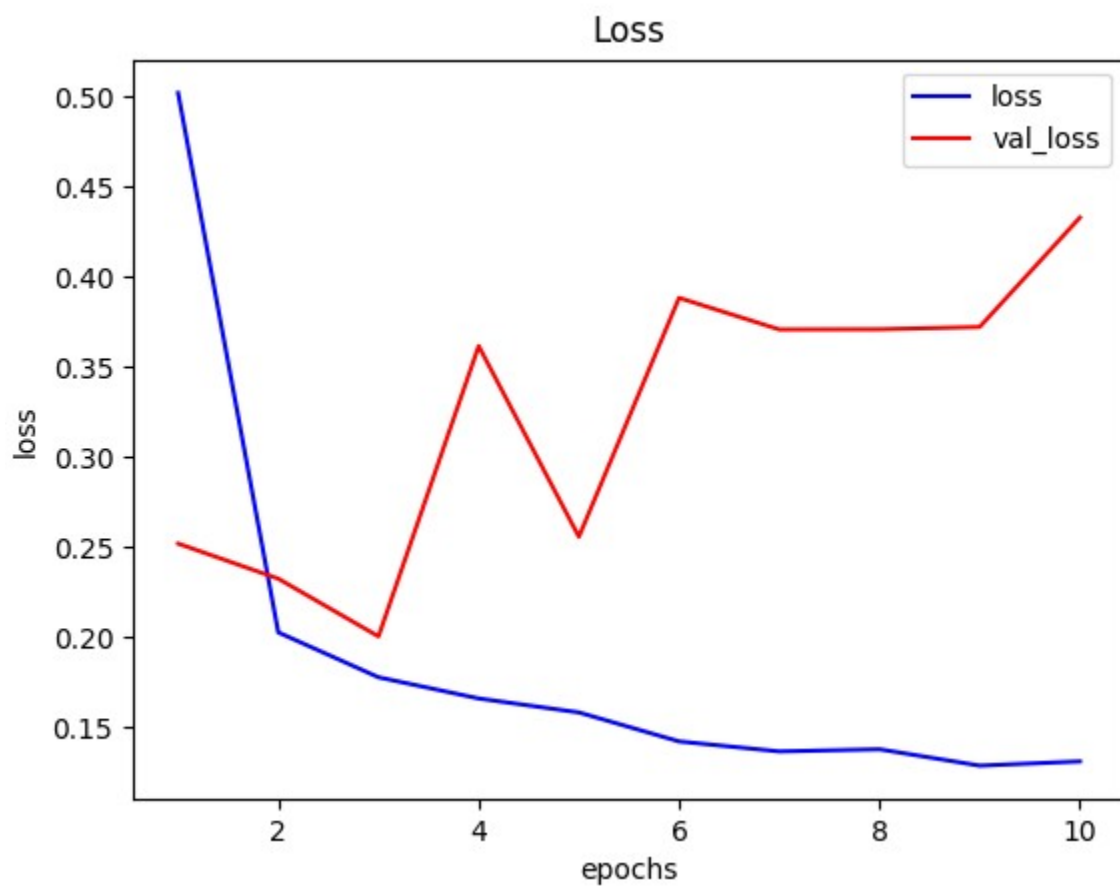


Рис.5 – Потери нс, learning_rate = 0.02

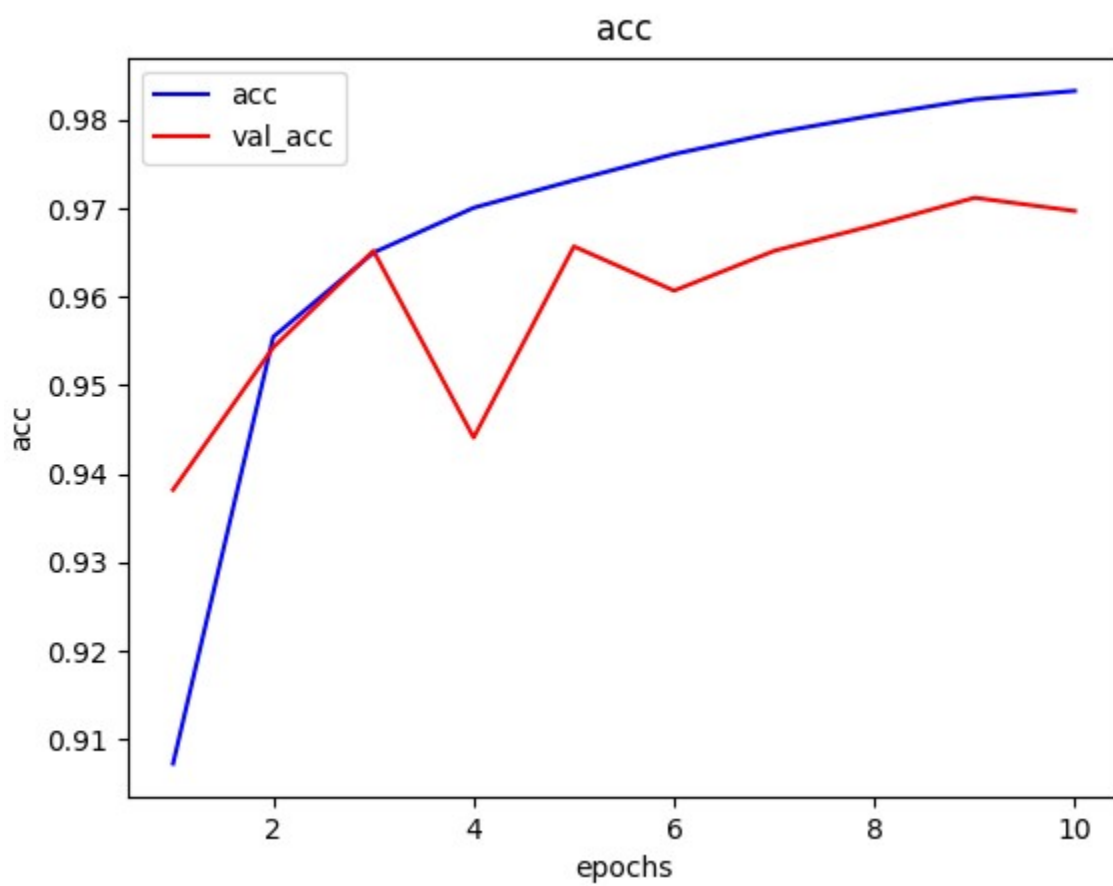


Рис.6 – Точность нс, learning_rate = 0.02

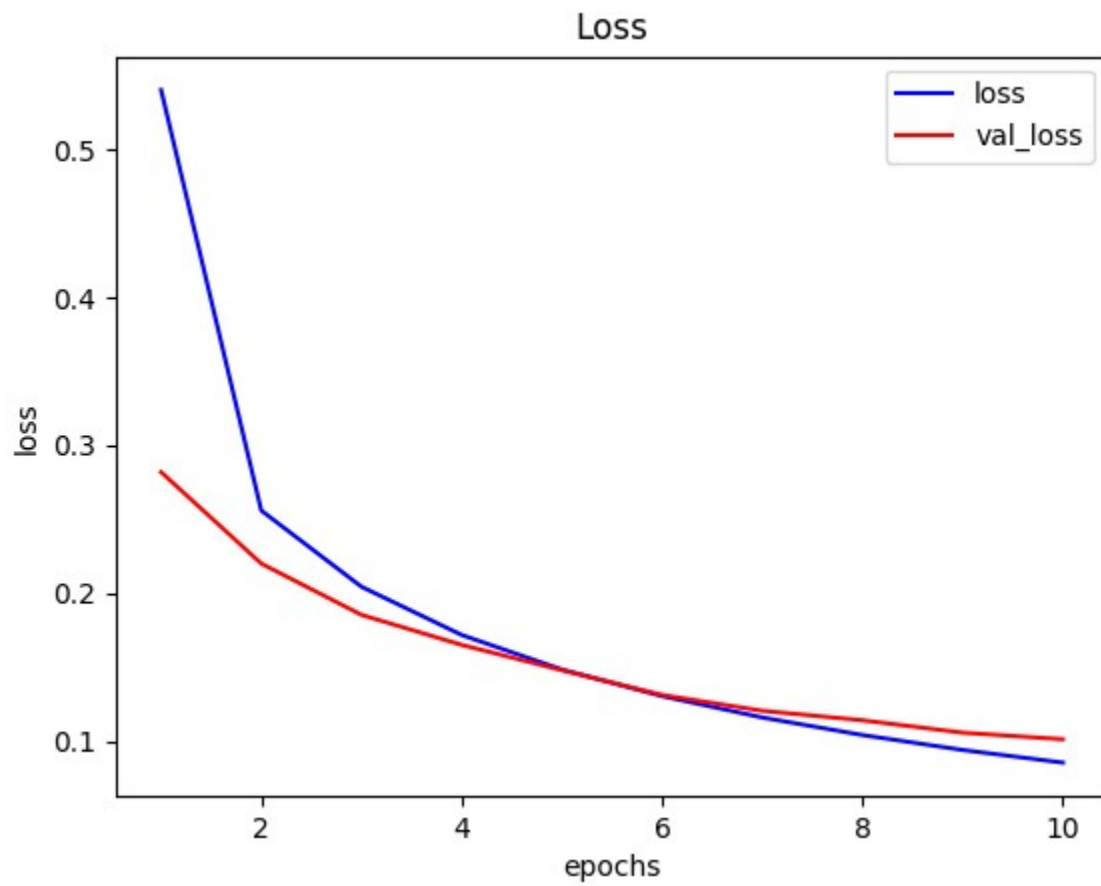


Рис.7 – Потери нс, learning_rate = 0.0002

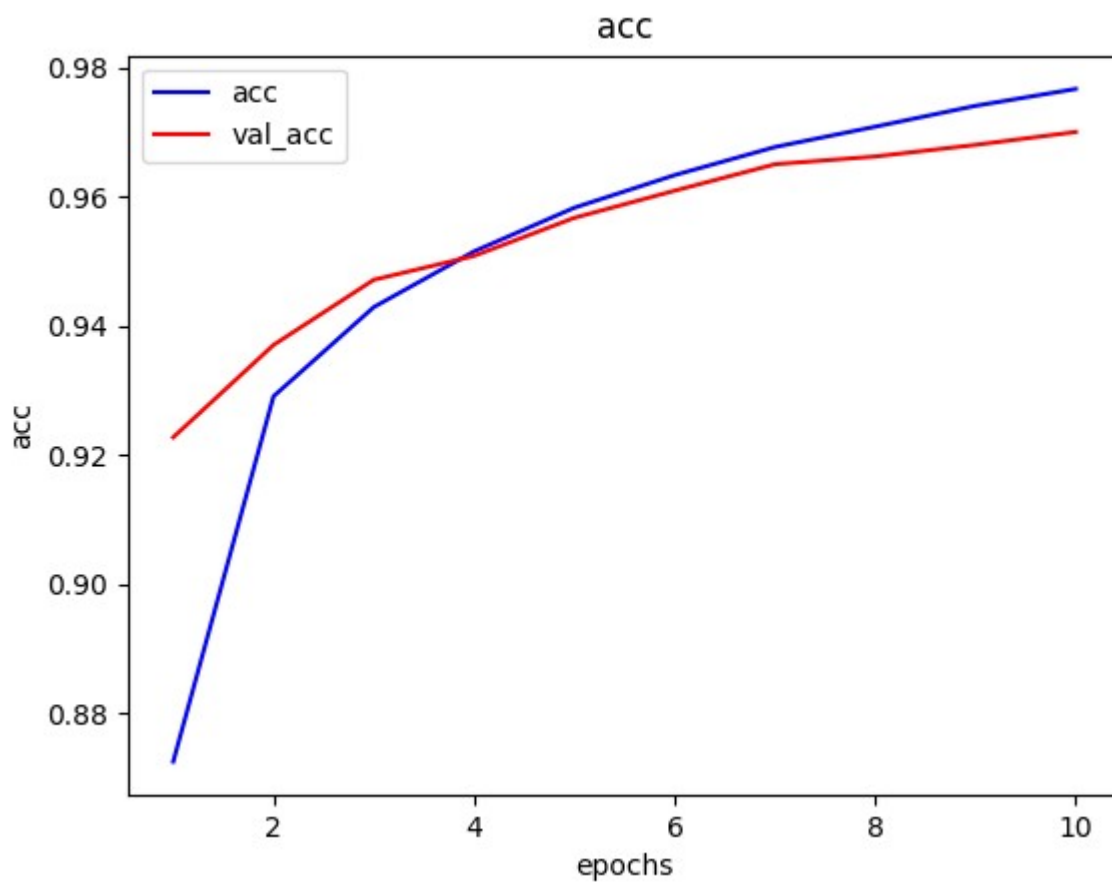


Рис.8 – Точность нс, learning_rate = 0.0002

3. Оптимизатор SGD

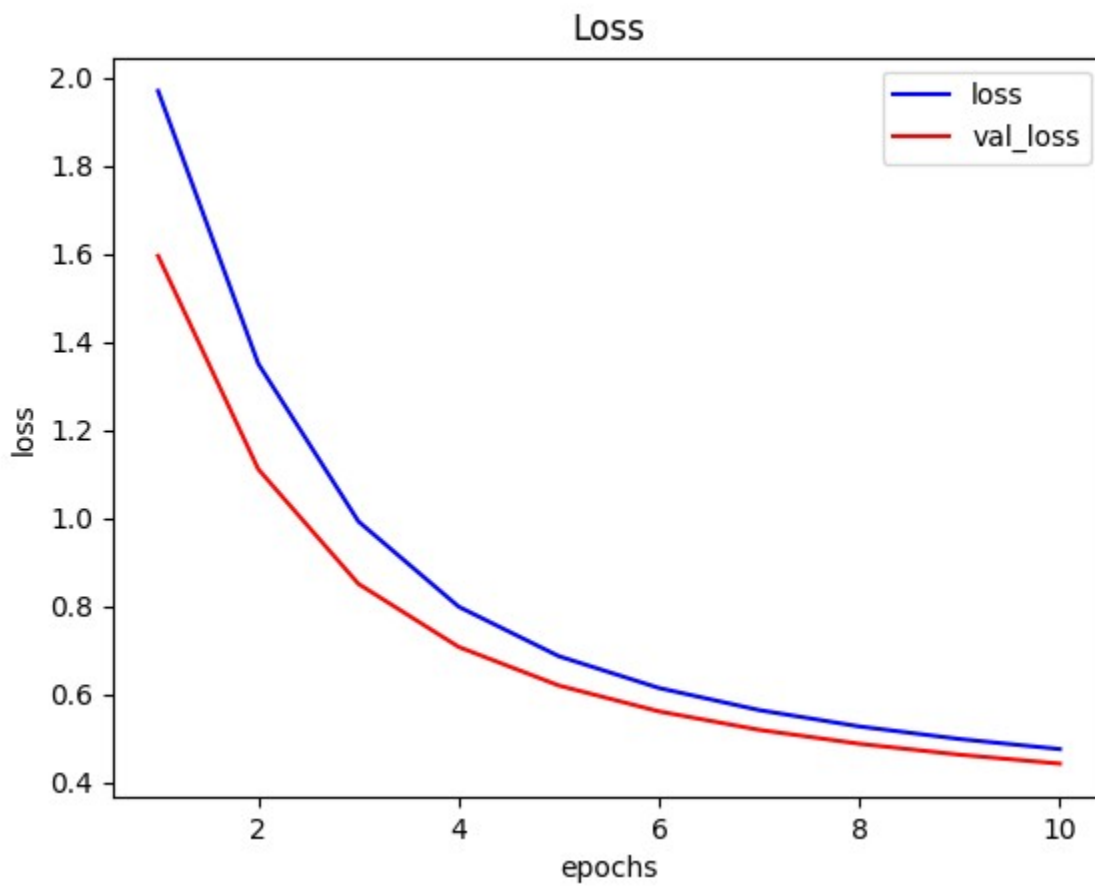


Рис.9 – Потери нс, learning_rate = 0.0002

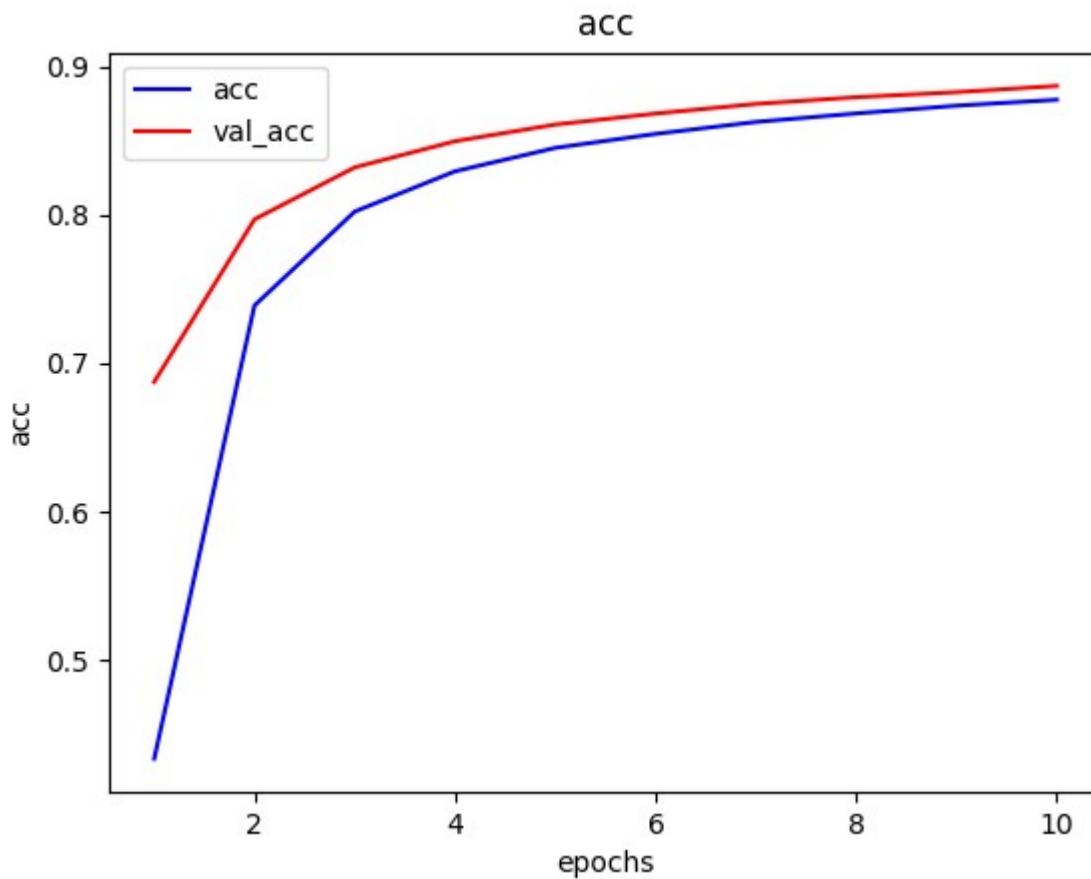


Рис.10 – Точность нс, learning_rate = 0.0002

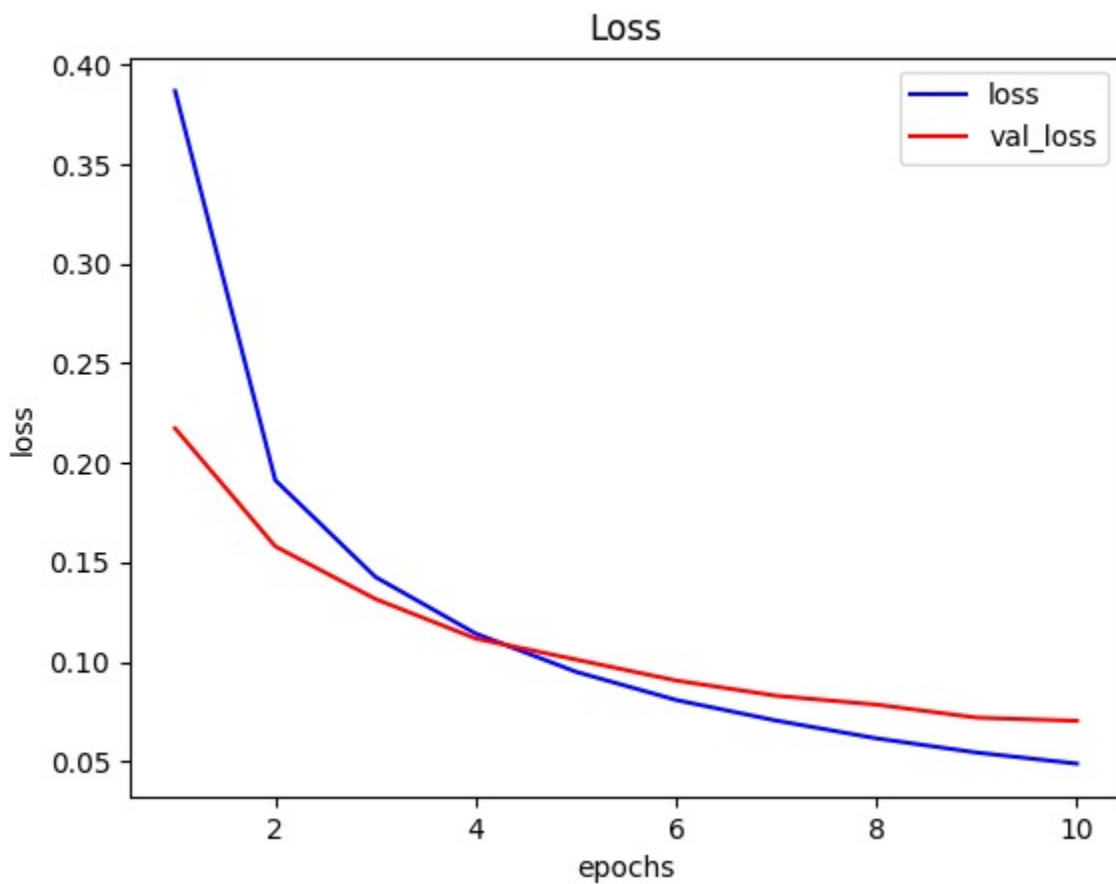


Рис.11 – Потери нс, learning_rate = 0.02

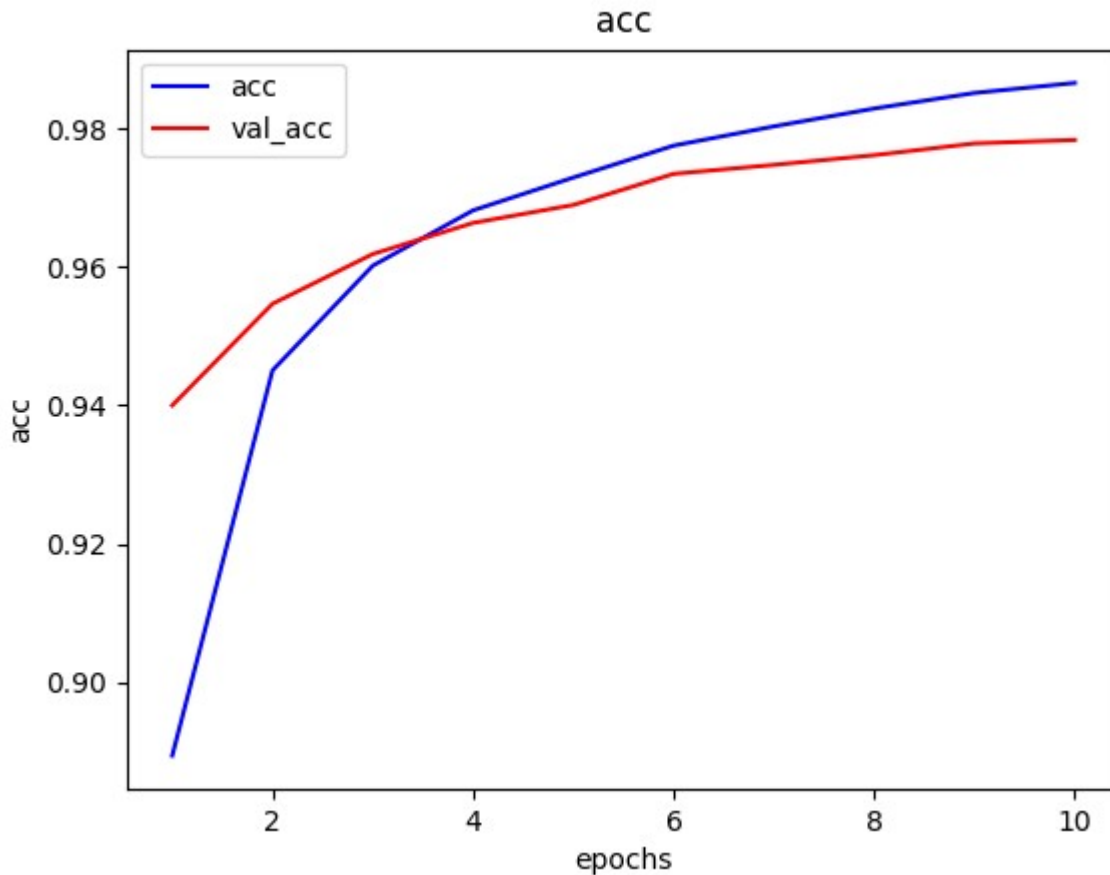


Рис.12 – Точность нс, learning_rate = 0.02

По требованиям к лабораторной работе была написана функция для считывания пользовательского изображения. Код функции:

```
def get_image(path):  
    img = image.load_img(path=path, grayscale=True, target_size=(28, 28, 1))  
    result = image.img_to_array(img)  
    return result
```

Функции передается путь к изображению и она возвращает array с формой (28,28,1) с которым можно работать.

Выводы.

В ходе работы была изучена задача классификации рукописных цифр с помощью базы данных MINIST. Рассмотрены влияния работы разных оптимизаторов с разными скоростями обучения на точность и потери нейронной сети. Из всех наблюдений можно сделать вывод, что для всех оптимизаторов кроме SGD чем меньше скорость обучения тем меньше ошибка и выше точность (до определенного значения, после которого нейронная сеть не обучается).