

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

ОТЧЕТ
по лабораторной работе №1
по дисциплине «Искусственные нейронные сети»
Тема: Многоклассовая классификация цветов

Студентка гр. 8382

Рочева А.К.

Преподаватель

Жангиров Т.Р.

Санкт-Петербург

2021

Цель работы.

Реализовать классификацию сортов растения ирис (Iris Setosa - 0, Iris Versicolour - 1, Iris Virginica - 2) по четырем признакам: размерам пестиков и тычинок его цветков.

Задание.

- Ознакомиться с задачей классификации
- Загрузить данные
- Создать модель ИНС в Keras
- Настроить параметры обучения
- Обучить и оценить модель

Выполнение работы.

Данные были загружены из файла iris.csv.

Создание модели (из двух слоев) проходит так:

```
model = Sequential()  
model.add(Dense(4, activation='relu'))  
model.add(Dense(3, activation='softmax'))
```

Обучение сети:

```
model.fit(X, dummy_y, epochs=75, batch_size=10, validation_split=0.1)
```

В работе было изучено обучение при различных параметрах функции `fit()`.

В функции `testEpochs()` были созданы модели с разными количествами эпох (`epochs`) — 75, 150, 225 и 300. Остальные параметры сети: три слоя (4, 4 и 3 нейрона), 10 `batches`, размер данных для валидации — 0.1 от исходных данных.

Результаты после четырех тестов представлены на рис.1-4.

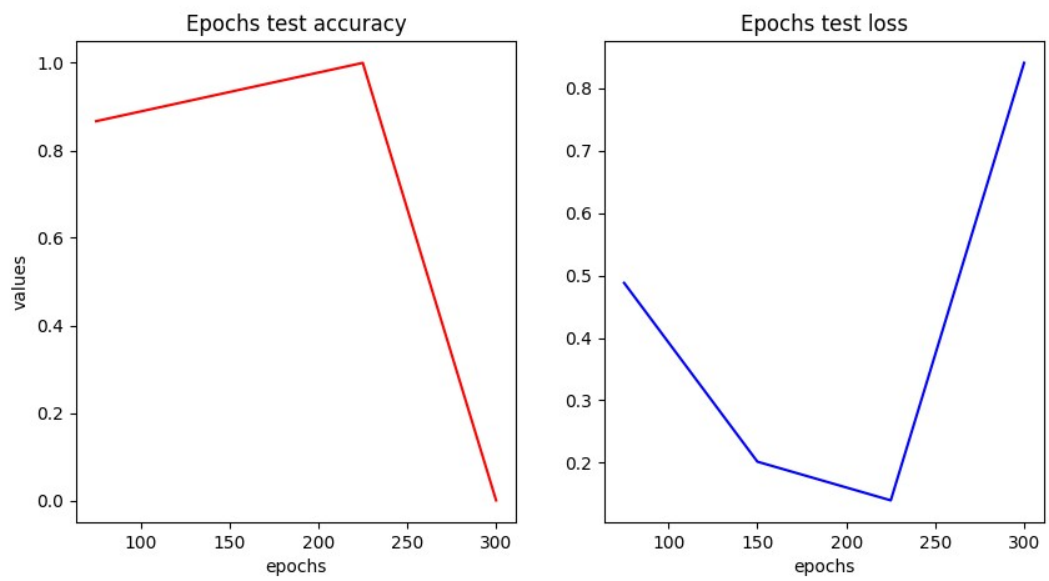


Рис. 1

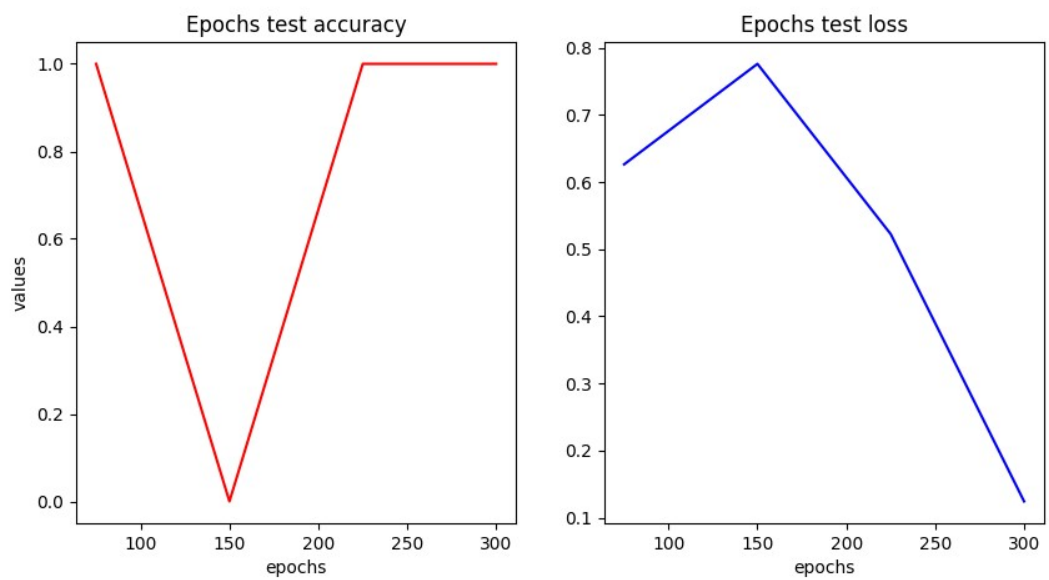


Рис. 2

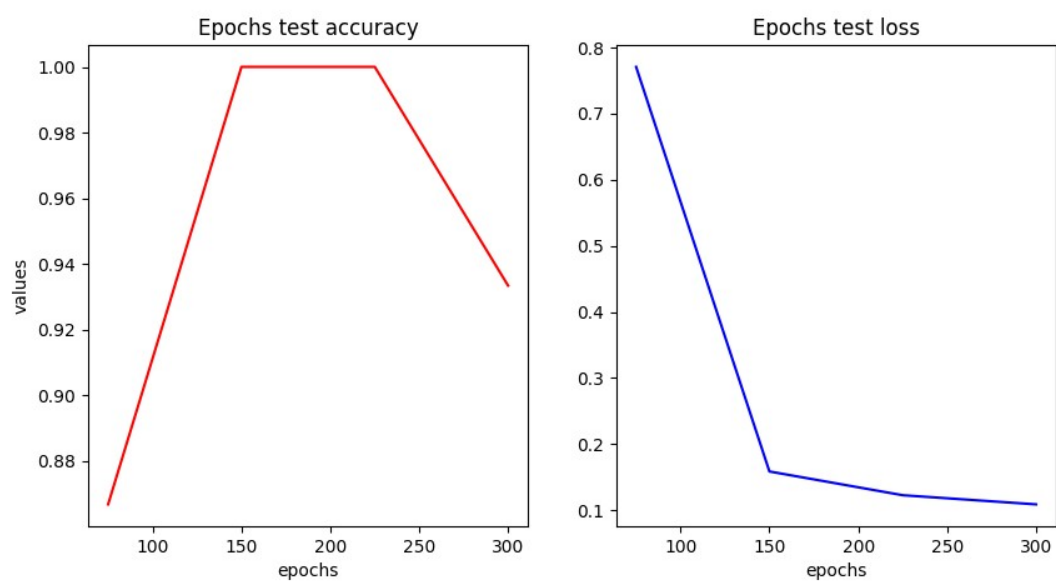


Рис. 3

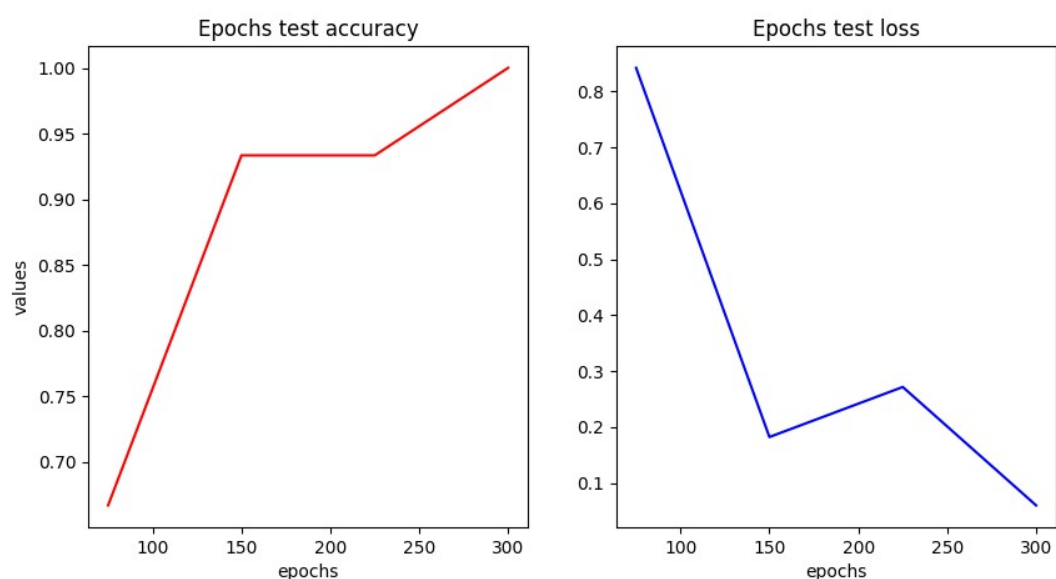


Рис. 4

Судя по графикам, до нуля никогда не опускались модели со значениями epochs равными 75 и 225, но при 225 три раза точность сети была единица.

Далее протестируем значения batchSize (для значений 10, 20, 30, 40, 50). Результаты представлены на рис.5-8.

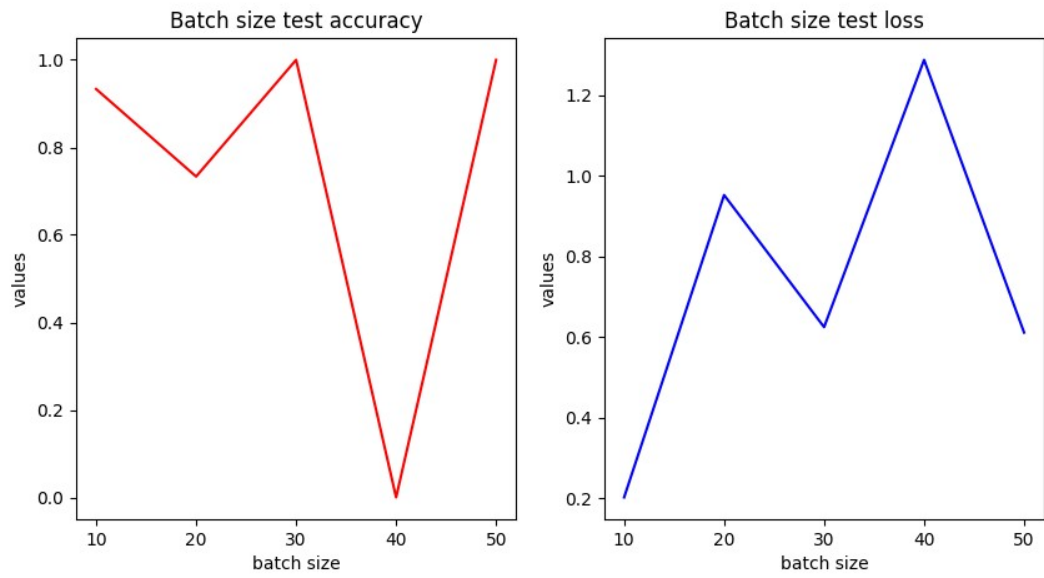


Рис. 5

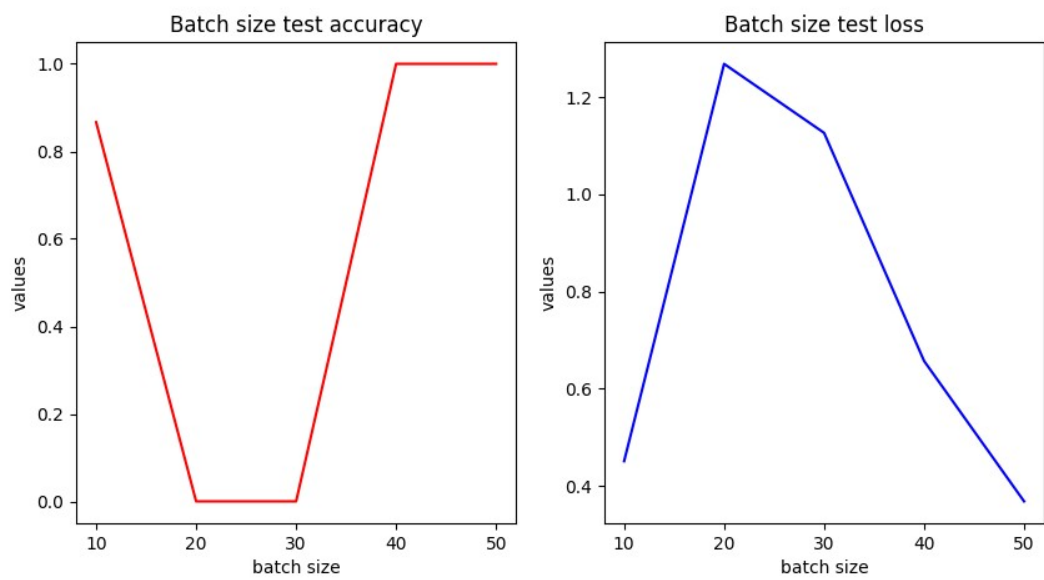


Рис. 6

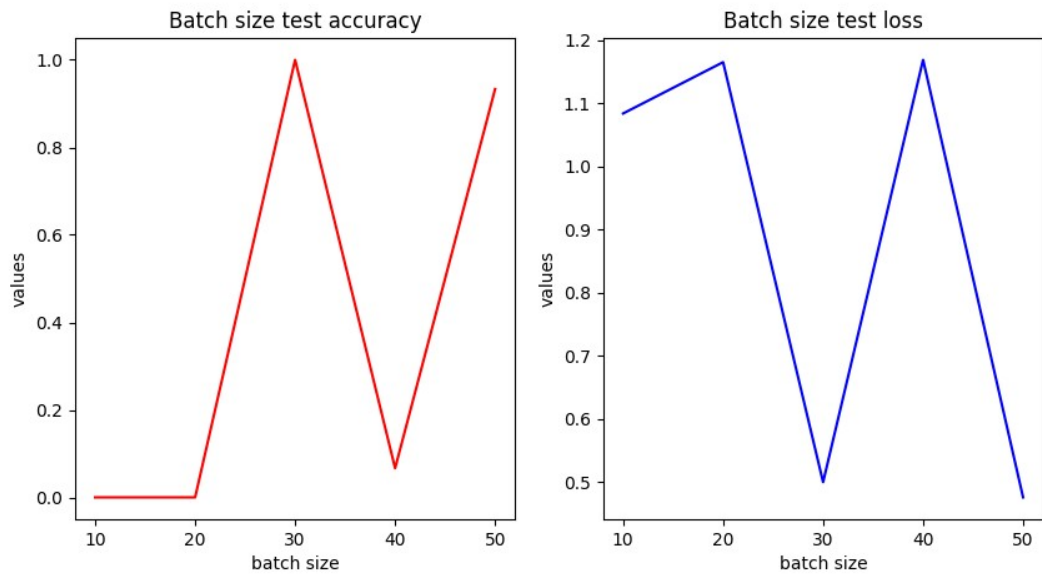


Рис. 7

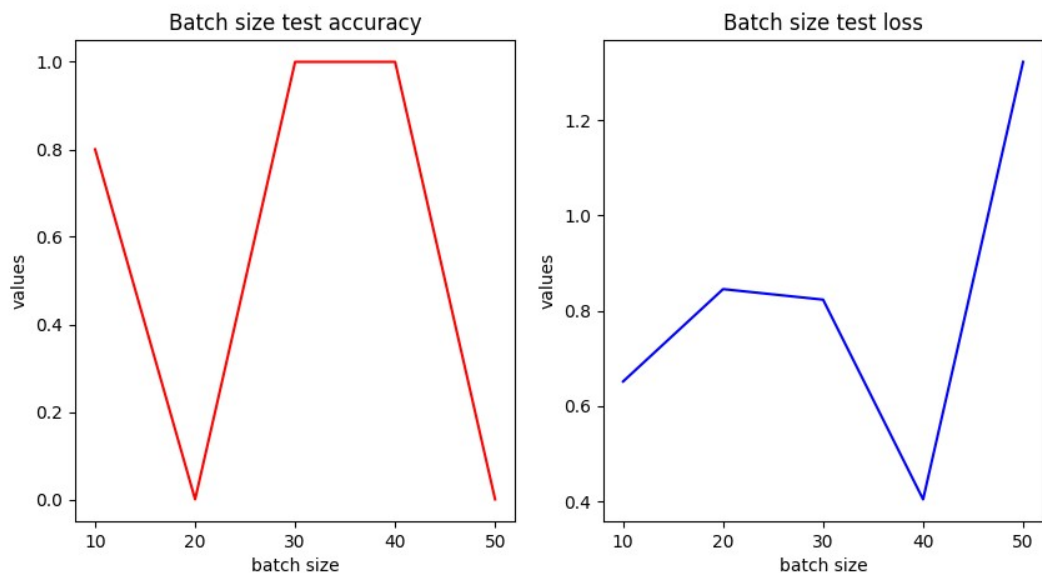


Рис. 8

При каждом числе batch size значение точности модели было и близким к единице, и равным нулю, так что можно сделать вывод, что нет особо большой разницы, какое число batch size из 10-50 выбрать.

Проверим, как значение validation split влияет на результат. Протестируем значения 0.1, 0.2 и 0.3 (рис.9-13).

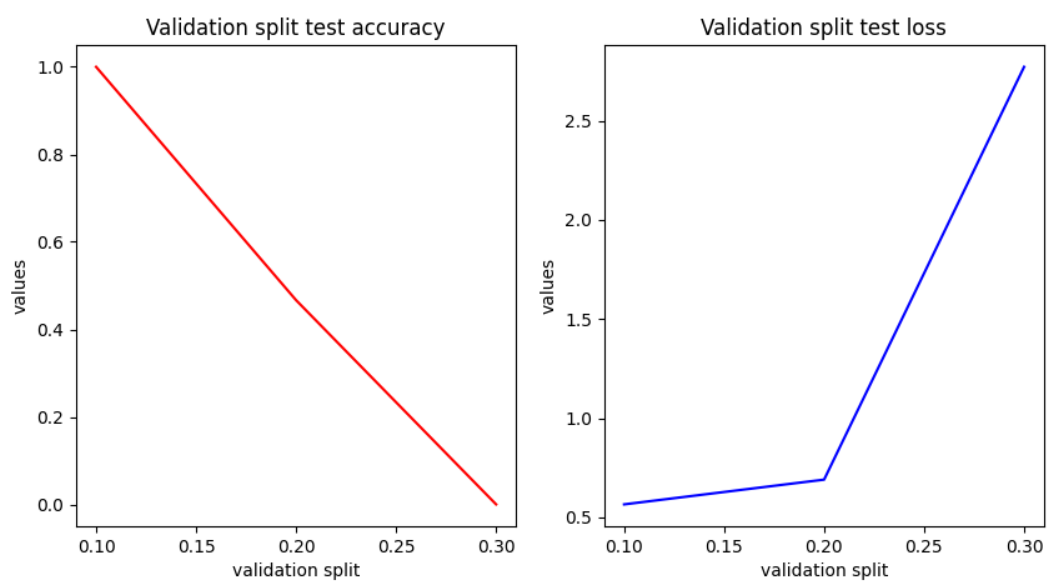


Рис. 9

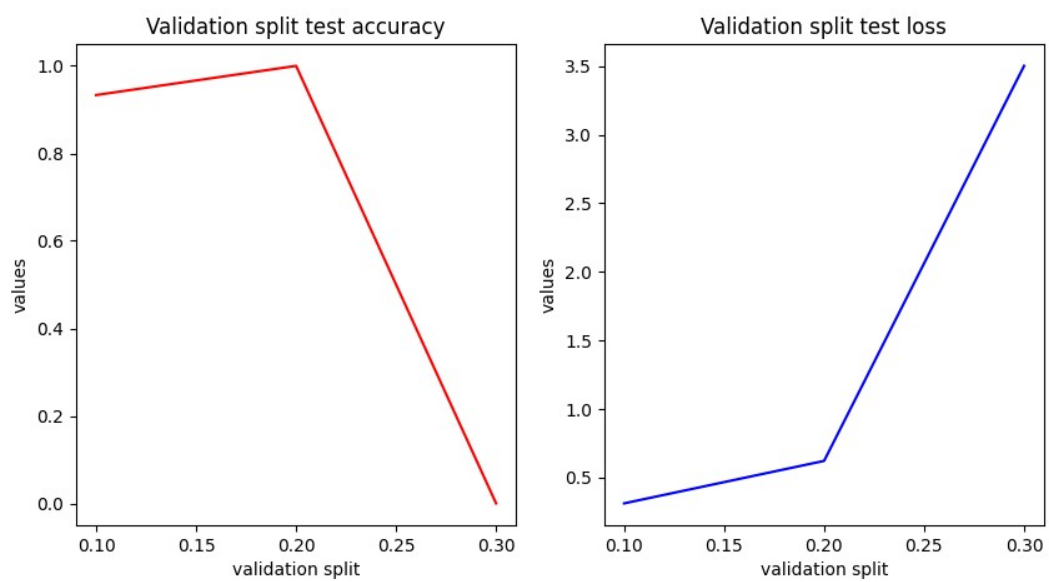


Рис. 10

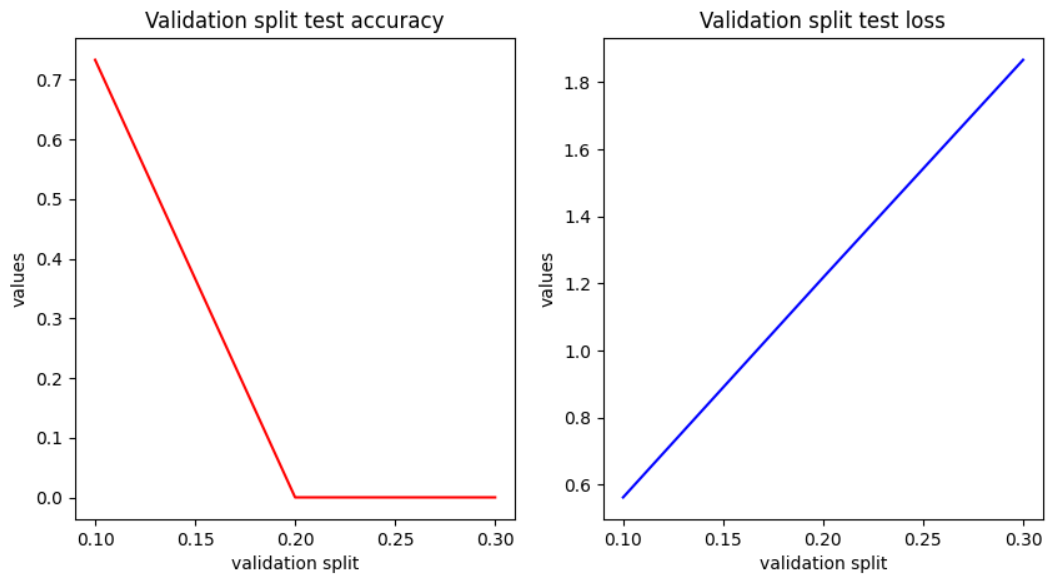


Рис. 11

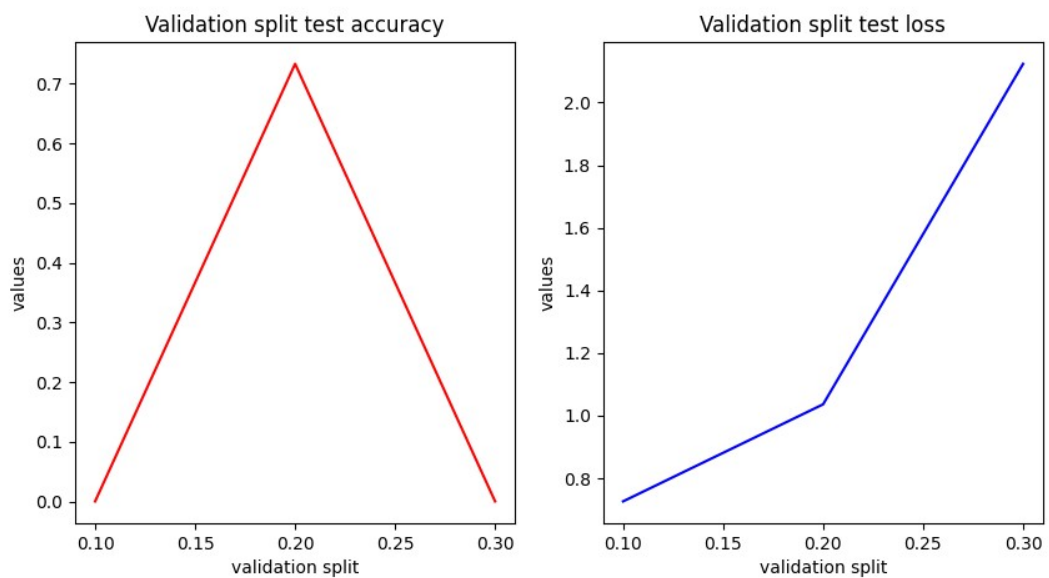


Рис. 12

По графикам можно понять, что лучше использовать 0.1 часть от начальных данных для валидации, но можно и 0.2.

Теперь протестируем кол-во нейронов для разного кол-ва слоев (3-6 слоев, [4,16,32,64,128] нейронов). Результаты показаны на рис.13-14.

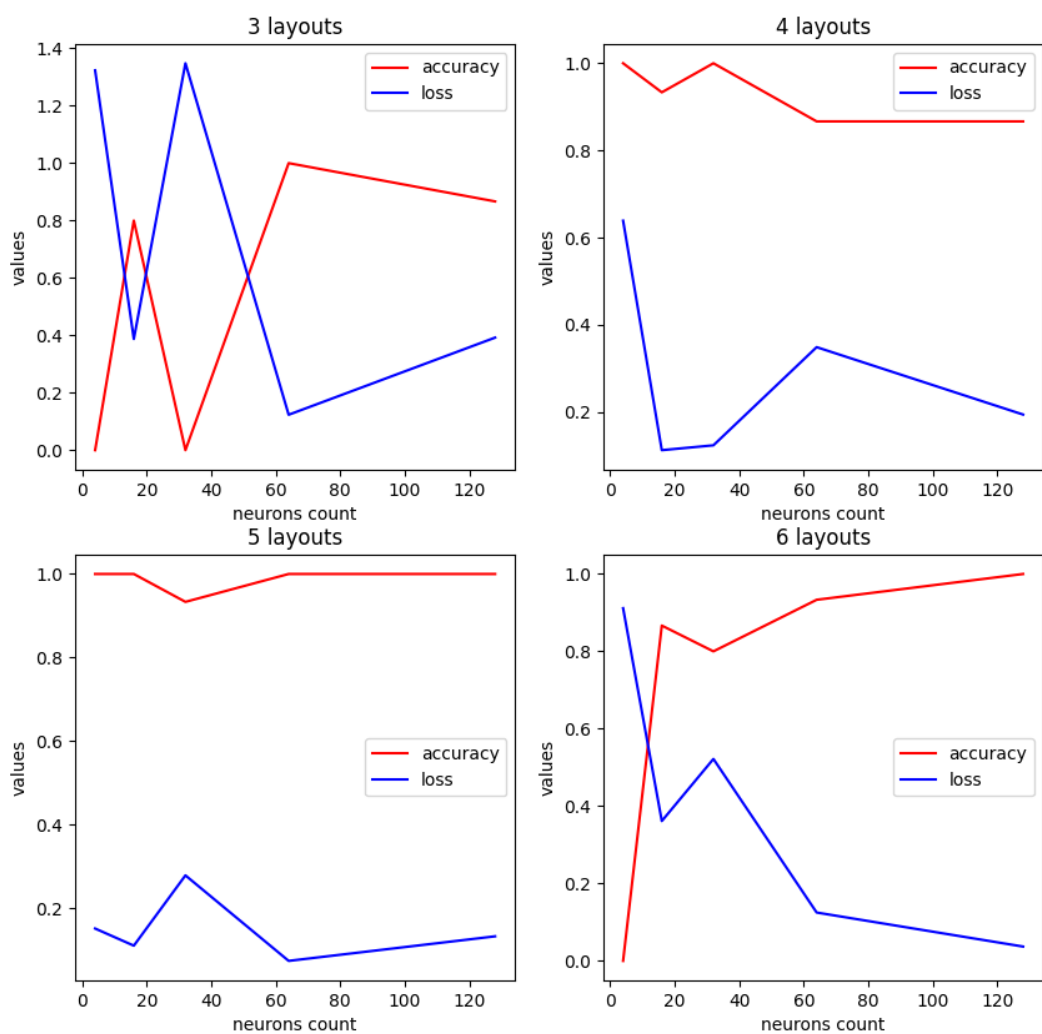


Рис. 13

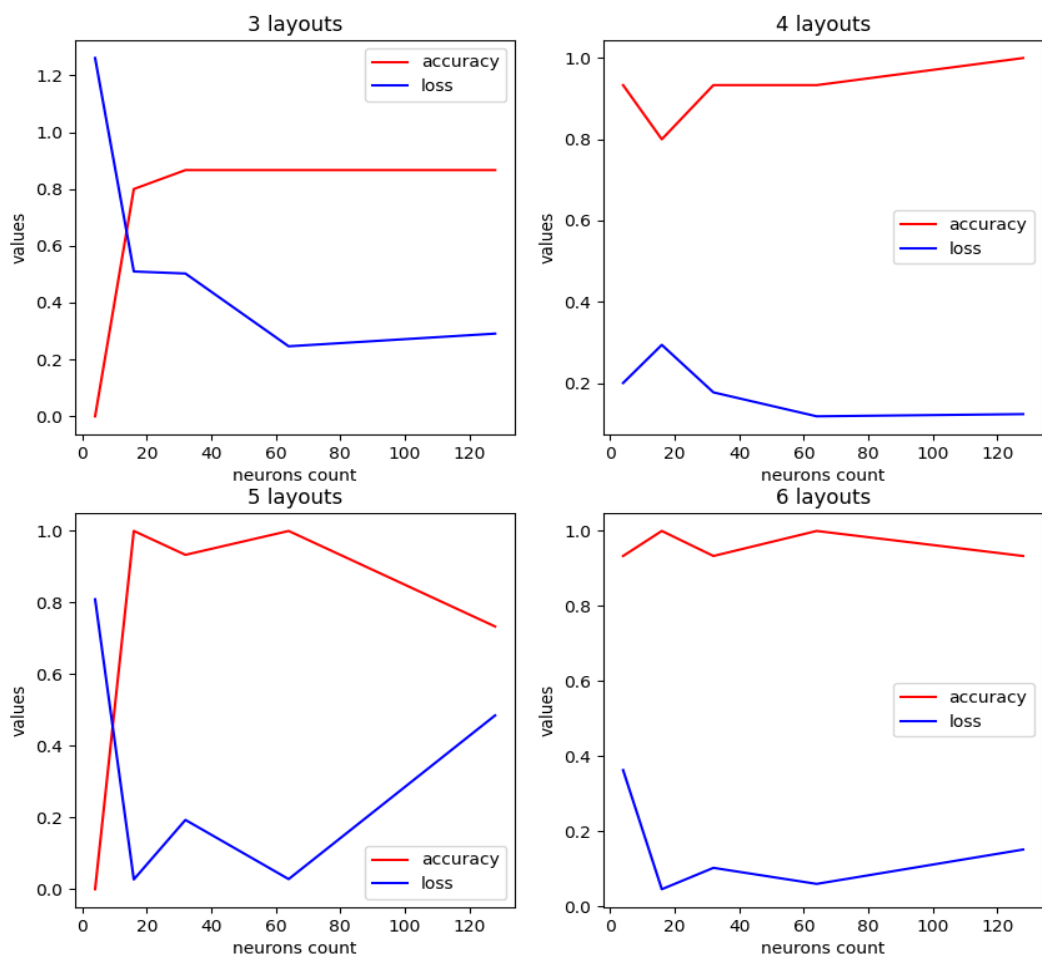


Рис. 14

Для слоев с 4-мя нейронами хорошо себя показала модель с 4-мя слоями. Для слоев с 16 нейронами — с 4-мя и 5-ью слоями. Для 32 нейронов — с 4-мя слоями. Для 64 нейронов — с 5-ью и 6-ью слоями. Для 128 нейронов — 4-мя и 6-ью слоями.

Для всех моделей значение точности не сильно меняется, не считая нулевых значений, но возможно, их можно считать промахами.

Из всех полученных данных можно попробовать построить хорошие модели и затем выбрать наилучшую. Во всех моделях будет 4 слоя — только

в моделях с 4-мя слоями не было значения 0 для точности. Графики моделей представлены на рис.15-17.

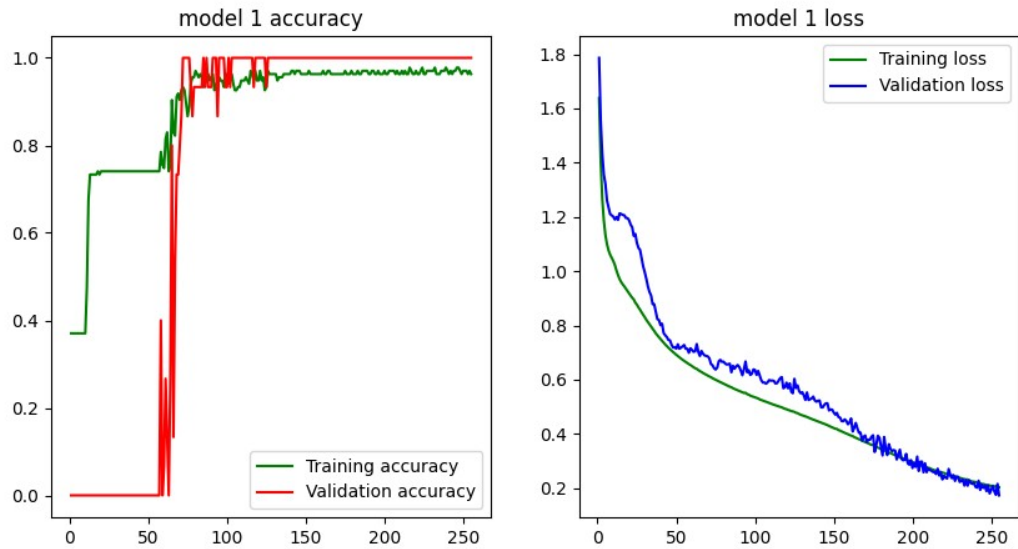


Рис. 15 — модель с 4-мя слоями и [4, 4, 4, 3] нейронами в слоях, epochs = 255, batch size = 10, validation split = 0.1

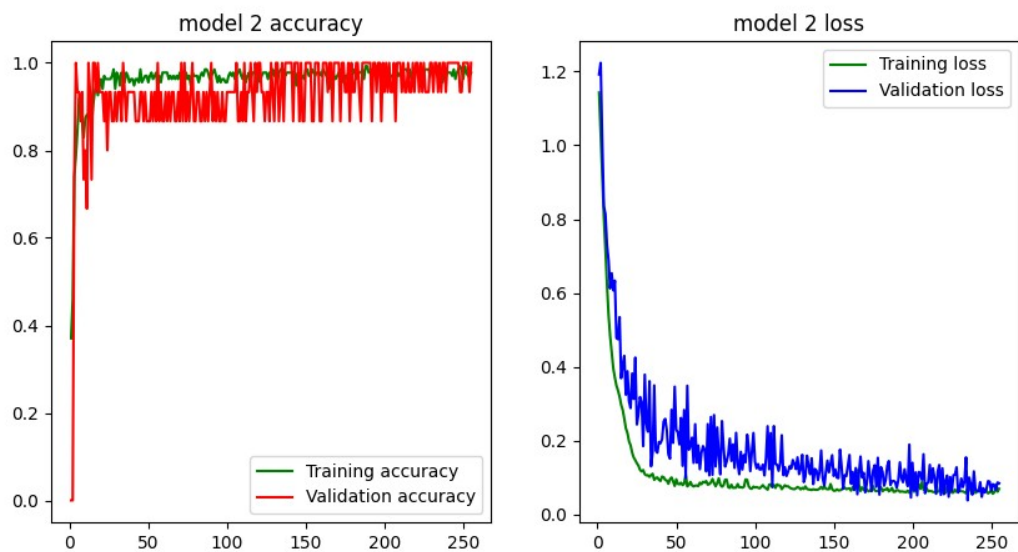


Рис. 16 — модель с 4-мя слоями и [4, 32, 32, 3] нейронами в слоях, epochs = 255, batch size = 10, validation split = 0.1

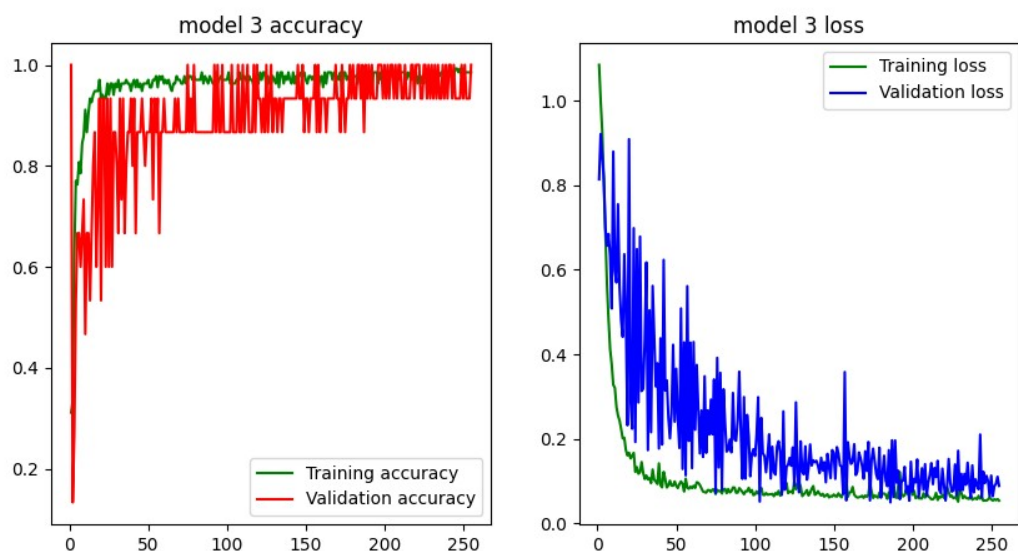


Рис. 17 — модель с 4-мя слоями и [4, 64, 64, 3] нейронами в слоях, epochs = 255, batch size = 10, validation split = 0.1

У моделей 2 и 3 точность достигается единицы гораздо быстрее, чем у первой модели. При этом у второй модели скачки наблюдаются реже, так что сделаем предположение, что она является наилучшей.

Выводы.

В ходе выполнения работы были получены навыки обучения простой сети на данных цветов. Так же были протестированы модели с разными параметрами обучения и было экспериментально выявлено, что модель с четырьмя слоями и 4, 32, 32, 3 количествами нейронов в них, кол-вом эпох равным 255 и batch size = 10 — наилучшая из исследованных.