Central Washington University

High Performance Computing

Spring 2019

Image Classification Using Keras

Author:
Hermann Yepdjio

Supervisor:
Dr. Szilard Vajda

May 29, 2019

Contents

1

1   Introduction

Project 3 was about classifying images using Keras and Tensorflow. The goal was to run some experiments, changing some parameters each time or increasing the number of hidden layers in the network in order to improve the classification accuracy of the model. The data set used in this project to train and test the model as well as the python code used to create the initial model were obtained from [1]. The code was modified to adjust the parameters and add more layers to the network, and the resulting models were evaluated for the classification accuracy. The results of the experiments are discussed in the following section.

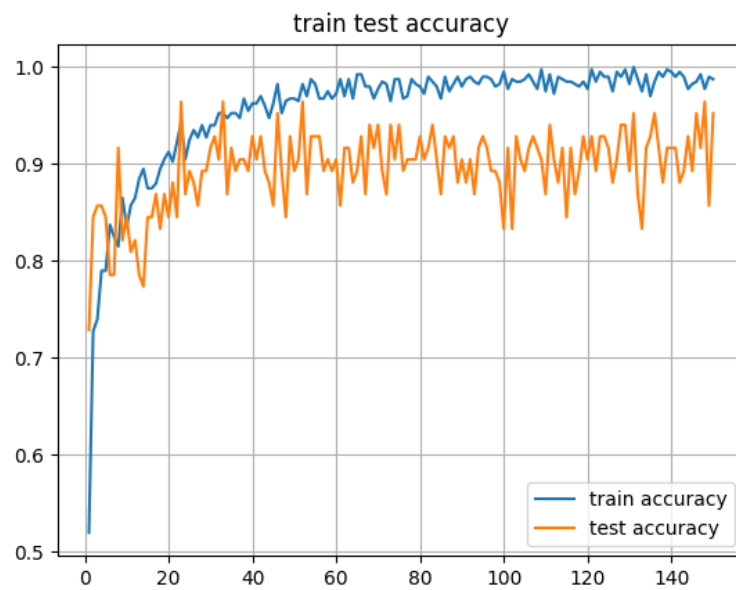# 2   Results for Experimentation Using Training and Testing Sets

2.1   Increasing the Number of Convolution Layers and the Number of Epochs
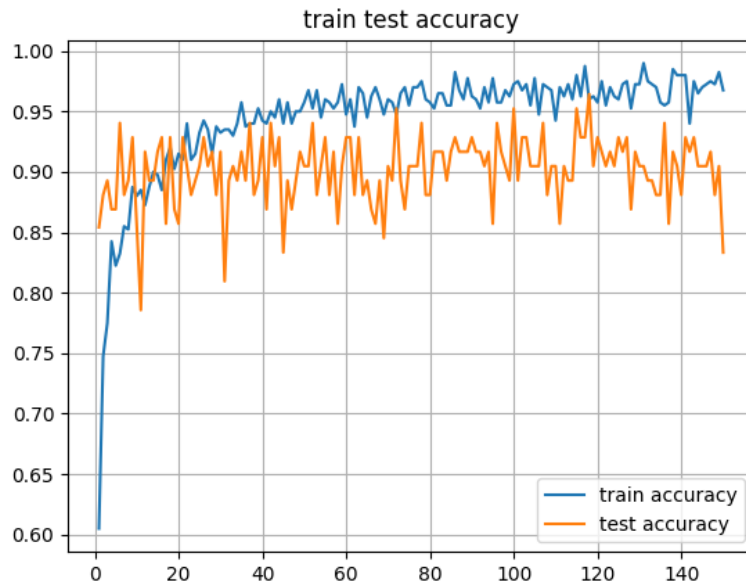
Figure 1: Accuracy Vs Epoch for a Network with Three Convolution Layers
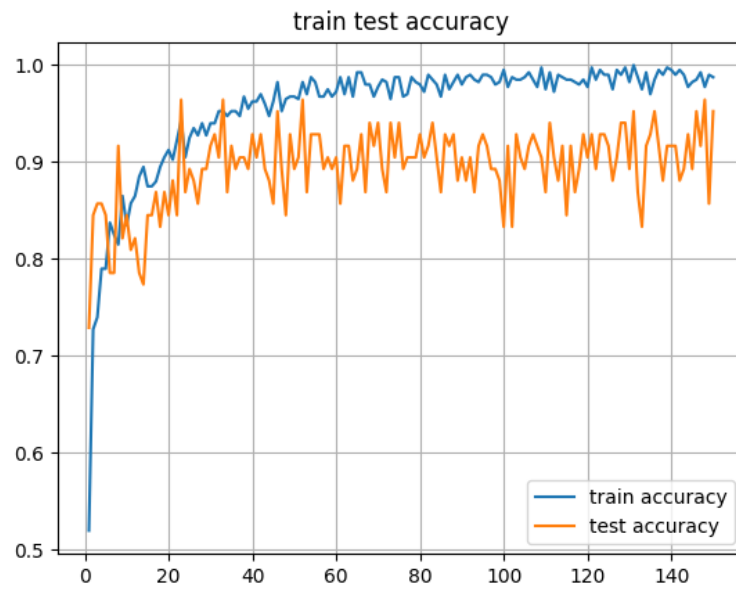
**Figure 2:** Accuracy Vs Epoch for a Network with Five Convolution Layers

Figure 3: Accuracy Vs Epoch for a Network with Seven Convolution Layers

As figures 1, 2 and 3 above show, the train classification accuracy tends to increase as the epoch number increases while the test classification accuracy pretty much remains stable (between .75 and .95). Those figures also show that the train classification accuracy improves as the number of convolution layers increases.
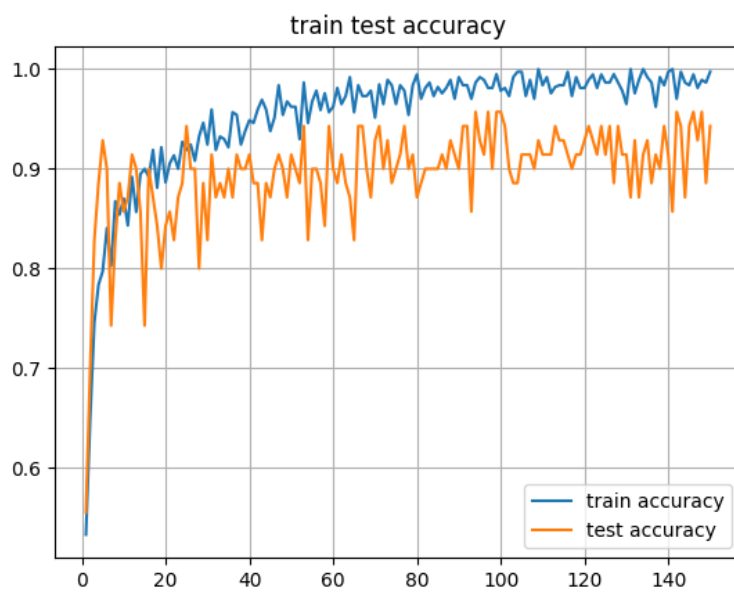
2.2 Increasing the Batch Size and the Number of Epochs
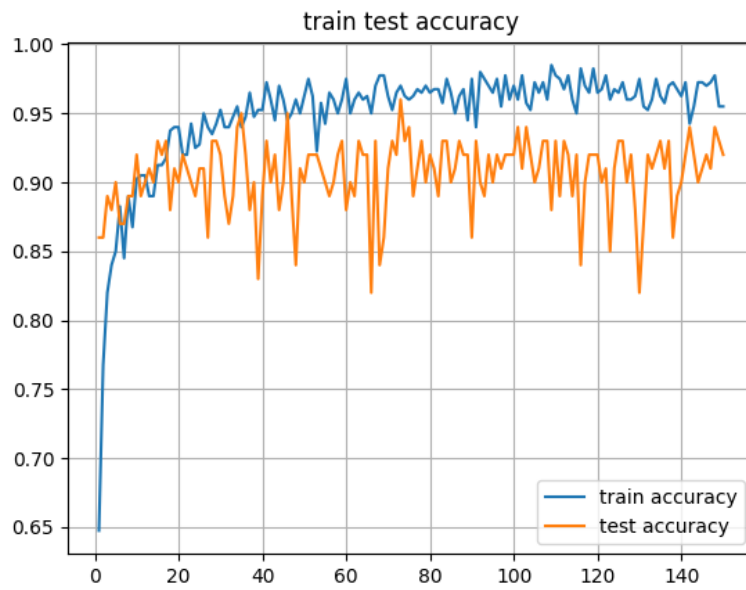
**Figure 4:** Accuracy Vs Epoch with Batch Size = 10
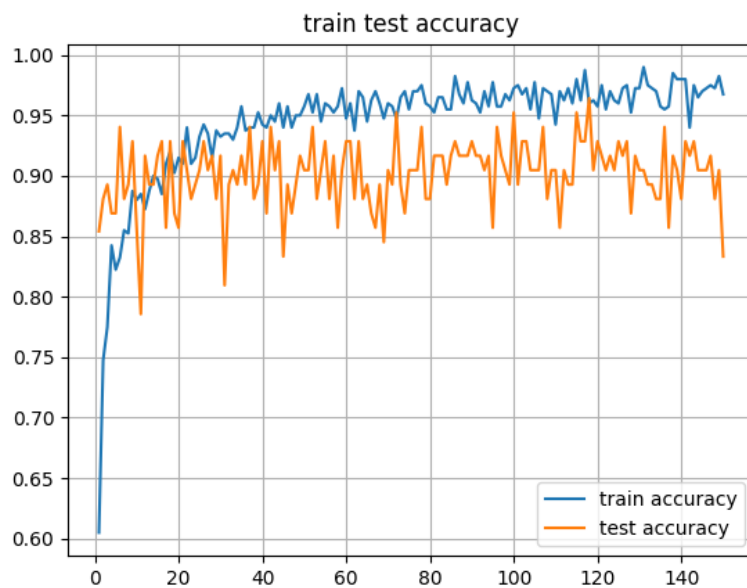
**Figure 5:** Accuracy Vs Epoch with Batch Size = 16

Figure 6: Accuracy Vs Epoch with Batch Size = 30

The patterns seen in section 2.1 can be observed in figures 4, 5, and 6 as well. That is, the train classification accuracy tends to increase as the epoch number increases while the test classification accuracy remains stable (between .75 and .95) and the train classification accuracy improves as the batch size increases.
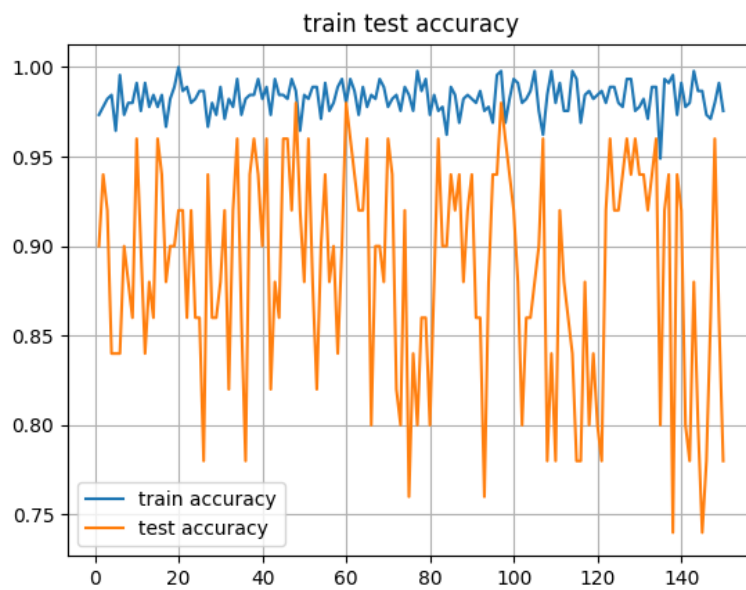
3 Result of Experimentation Using Ten Folds Cross Validation
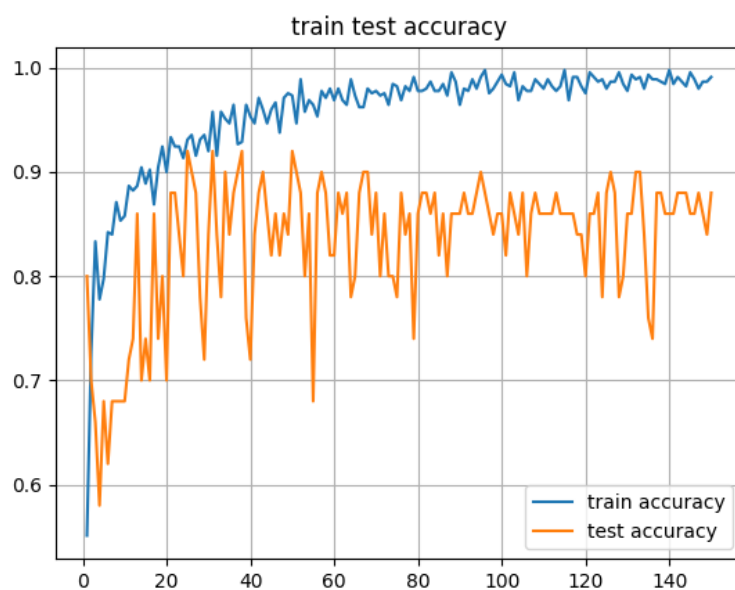
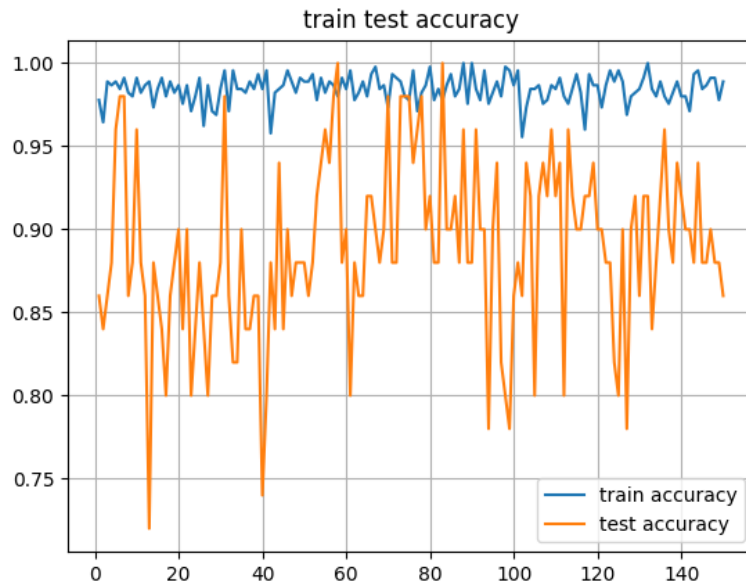**Figure 7:** Accuracy Vs Epoch for Fold 1
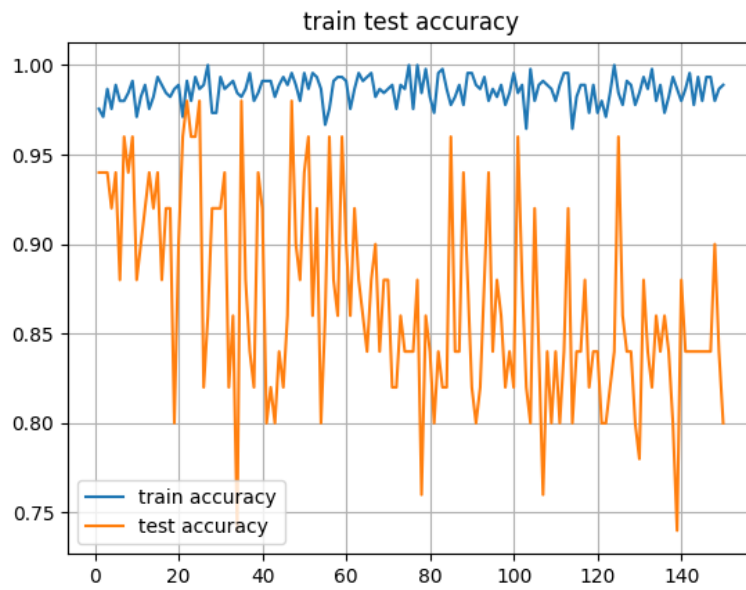
Figure 8: Accuracy Vs Epoch for Fold 2

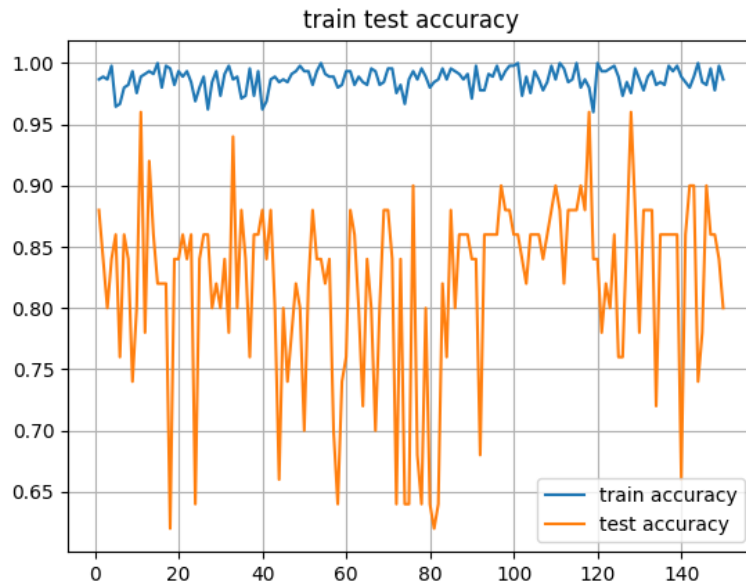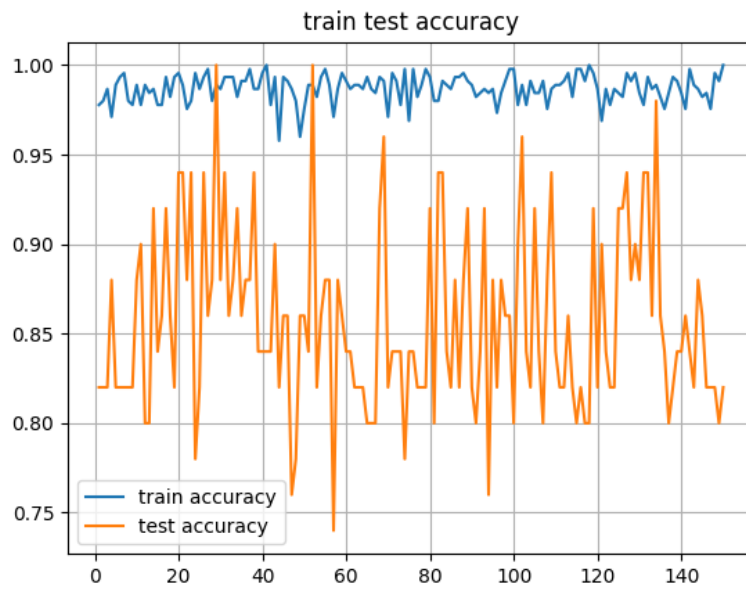**Figure 9:** Accuracy Vs Epoch for Fold 3

Figure 10: Accuracy Vs Epoch for Fold 4

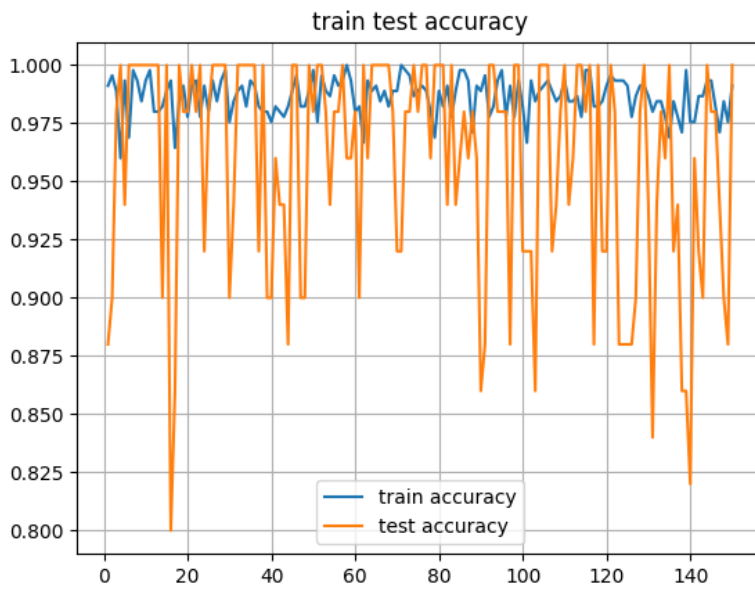**Figure 11:** Accuracy Vs Epoch for Fold 5
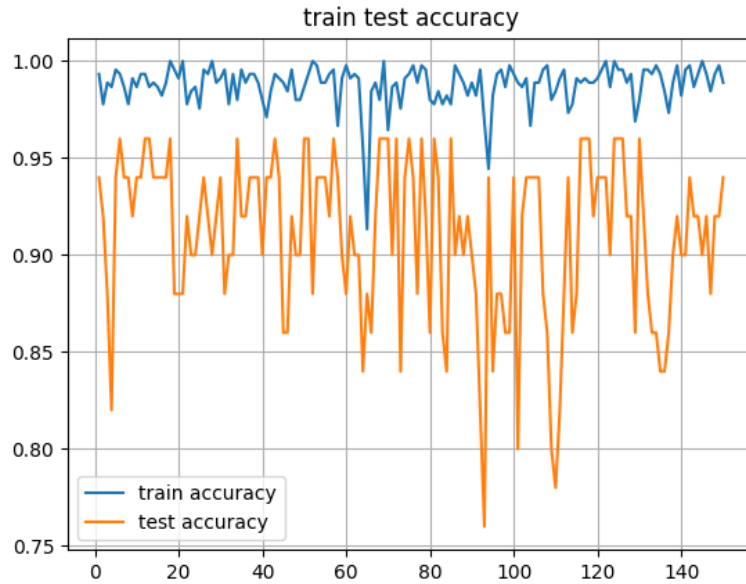
8

Figure 12: Accuracy Vs Epoch for Fold 6

Figure 13: Accuracy Vs Epoch for Fold 7

Figure 14: Accuracy Vs Epoch for Fold 8

Figure 15: Accuracy Vs Epoch for Fold 9

Figure 16: Accuracy Vs Epoch for Fold 10

Figures 7 show that both the train and test classification accuracies increase as the number of epochs increases for Fold 1. However, figures 8 - 16 show that the train classification accurancy remain stable (between .90 and 1) while the test classification accuracy is really unstable (varies between .62 and 1). Figure 15 show that fold 9 has the most unstable test classification accuracy across the epochs as it varies between .6 and 1.

4 Conclusion

The experiments ran in project 3 show that changing the configuration of a network can affect its outcome in good or in bad. Increasing the number of convolution layers or the batch size did not really affect the test classification accuracy. However, using 10 folds cross validation produced the highest test classification accuracy (which was 1 according to Figure 15) for fold 9, but overall it also produced the most unstable classification accuracy across the epochs.

References

[1] NitishGangwar, *Python—Image Classification using keras* https://www.geeksforgeeks.org/python-image-classification-using-keras/