CPSC 501 – Assignment 4

Artem Golovin
30018900

December 2019

Part 1

To optimize the given MNIST model, I performed the following:

- Added a Dense layer with 512 neurons

- To prevent overfitting, Dropout layer was added as well

- The algorithm was changed to 'Adam'

- Number of epochs increased from 1 to 10

Resulting model is 99% accurate on the training data and 98% accurate on the test data.

References

- https://www.tensorflow.org/tutorials/quickstart/beginner

Part 2

Part 2.1

Initial model for Part 2 was using the same layers as the model for Part 1, which can be seen here 14cec7c3. The accuracy with that model was roughly 93%. However, the model failed to identify image `user_inputs/image_A_lower.png`.

Part 2.2

To improve the model, I implemented a Convolution Neural Network, which increased the accuracy of notMNIST model to roughly 95%. This model was able to identify the image `user_inputs/image_A_lower.png`.

Besides the convolution network, adding more neurons via extra Dense layers and Dropout layers that prevent overfitting increased the predictions.

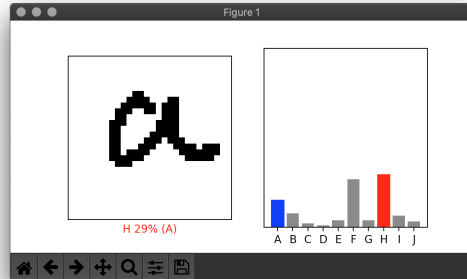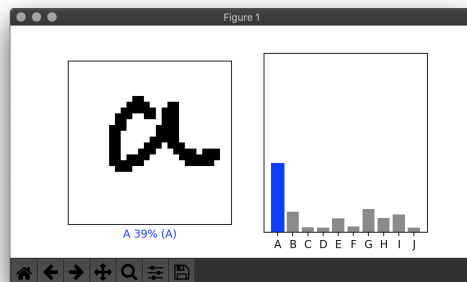Figure 1: Initial test with the first model

Figure 2: Test using the updated model

Part 3

Initially, the model was trained without taking in consideration overfitting. Resulted accuracy:

```
          [[{{node IteratorGetNext}}]]
56/56 [==============================] - 0s 6ms/step - loss: 0.3129 - accuracy: 0.8592
      37/Unknown - 0s 12ms/step - loss: 0.6468 - accuracy: 0.71352019-12-05 18:48:07.831887: W
equence
          [[{{node IteratorGetNext}}]]
37/37 [==============================] - 0s 13ms/step - loss: 0.6468 - accuracy: 0.7135
Model Accuracy: 71.4%
Model Loss:    0.65
```

After adding Dropouts and l2 regularizers, the model testing and training accuracy align and become roughly equal:

```
          [[{{node IteratorGetNext}}]]
28/28 [==============================] - 0s 6ms/step - loss: 0.6017 - accuracy: 0.7545
     19/Unknown - 0s 14ms/step - loss: 0.6683 - accuracy: 0.74052019-12-05 18:49:14.053099: W
equence
          [[{{node IteratorGetNext}}]]
19/19 [==============================] - 0s 15ms/step - loss: 0.6683 - accuracy: 0.7405
Model Accuracy: 74.1%
Model Loss:     0.67
```

References

- https://www.tensorflow.org/tutorials/load_data/csv

- https://www.tensorflow.org/tutorials/keras/overfit_and_underfit

- [view commit](#) • (a4) initial commit

- [view commit](#) • (a4) better project structure

- [view commit](#) • (p1) finish p1 with 98.4% (test) & 99.6 (train)

- [view commit](#) • (p2) train the model to be 93% accurate

- [view commit](#) • (p1, p2) update test files

- [view commit](#) • hide the image.png generated by grabimage

- [view commit](#) • ⚡ (p1) improve accuracy of MNIST model

- [view commit](#) • ⚡ (p2) train notMNIST model using conv. network

- [view commit](#) • ⚡ (p2) improve notMNIST model by tweaking the CNN

- [view commit](#) • ⚡ (p2) bump notMNIST accuracy to 95.2%

- [view commit](#) • (report) update the report

- [view commit](#) • ⚡ (p2) update report for p2 and add notes for image recognition

- [view commit](#) • (pipenv) add pipenv support

- [view commit](#) • (p1, p2) ensure the script's directory is in p1 or p2

- [view commit](#) • (pipenv) add project setup instructions

- [view commit](#) • (p3) save train and test csv and add necessary dependency

- [view commit](#) • (p3) split the heart csv data and build the initial model

- [view commit](#) • (p3) add information to the report

- [view commit](#) • ⚡ (p3) imrove CHD model accuracy by using dropouts and l2 regularizer

- [view commit](#) • (report) add notes for p2, p3 and screenshots