Sturfee Canidate Evaluation Task

Canidate: Albert Shaw

I trained a Alexnet-based Siamese network with cat and dog images. It was trained with shared weights between Siamese networks and with a Contrastive Margin Loss to try to seperate the Cat and Dog images. The final feature descriptor length is 10.

Design Choices

The network architecture is Alexnet-based, but has a some modifications due to the much simpler nature of the task and training/running time concerns. A feature size of 10 was chosen since it is a relatively simple binary separation we are learning. The last two layers was also reduced from 4096 length to 200 length since our task is simpler than imagenet. The first dropout layer was removed due and the second dropout layer was set to 0.25 dropout rate due to to training time concerns and the reduction in the previous layer size. Shared weights were chosen for the siamese networks since they work over the same image modality.

One particular modification I made to the default algorithm was normalizing the features to have a l2 norm of 1. This method is taken from the triplet-loss papers, but enforces that the features are always on the edge of a sphere and allows for an effective usage of constant margin parameter of 1.0. I hypothesize that this is why the visualizations are vaguely spherical(and in a way look linear).

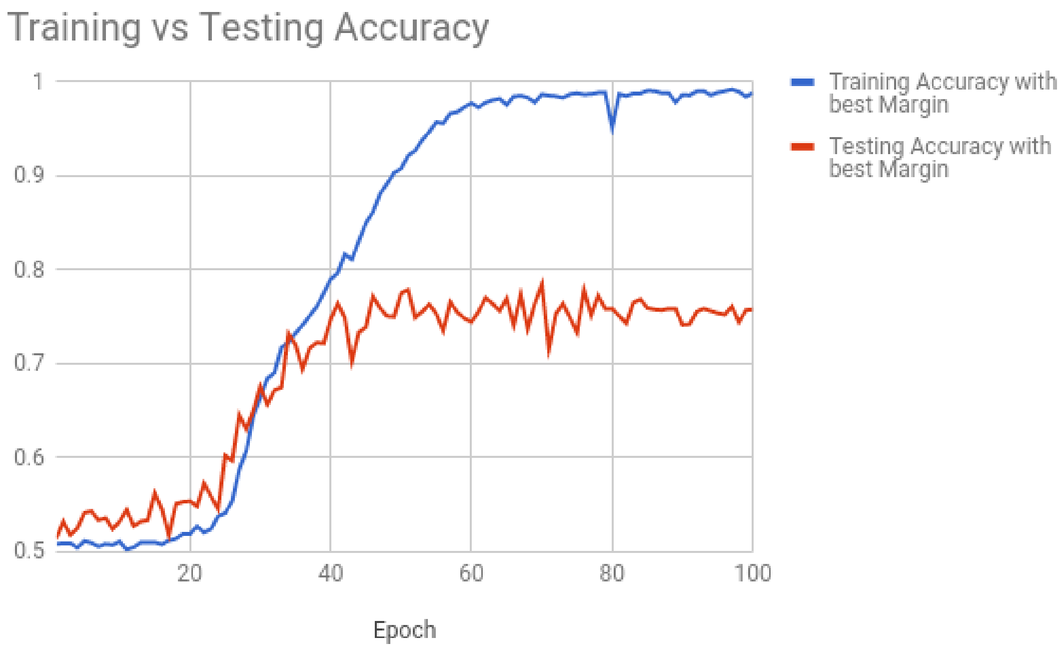
Data Augmentation

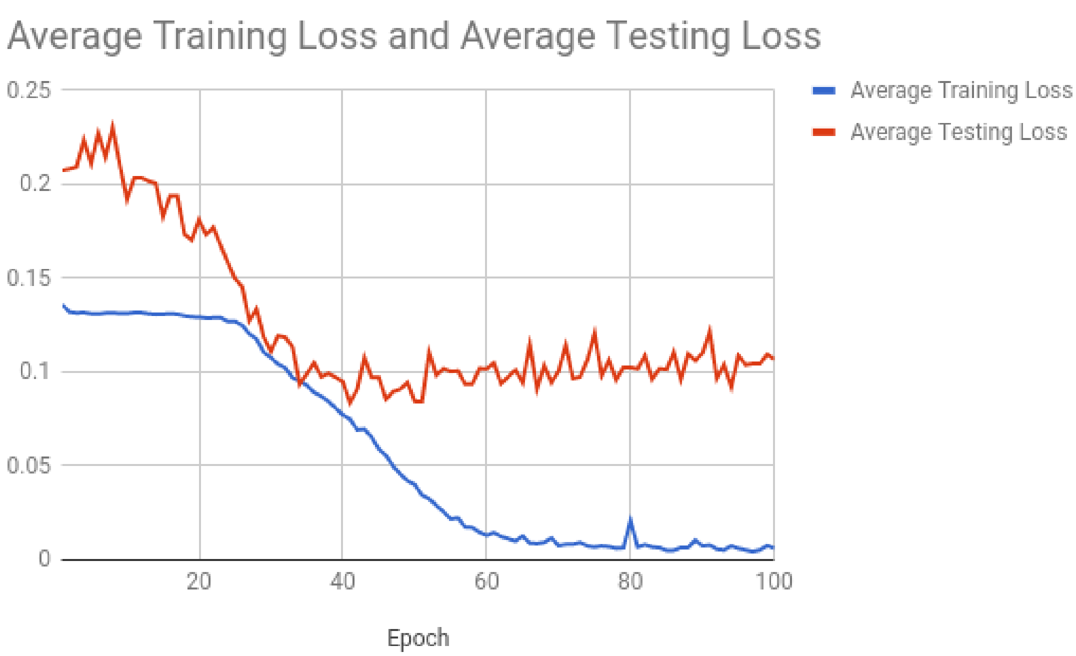
Very little data augmentation was done on this dataset. The images were simply resized to 224x224, and randomly flipped horizontally. This likely should be improved in the future since most images are not square. Naeve Cropping caused problems however due the variety of zooms and horizontal vs vertical photos.

Results

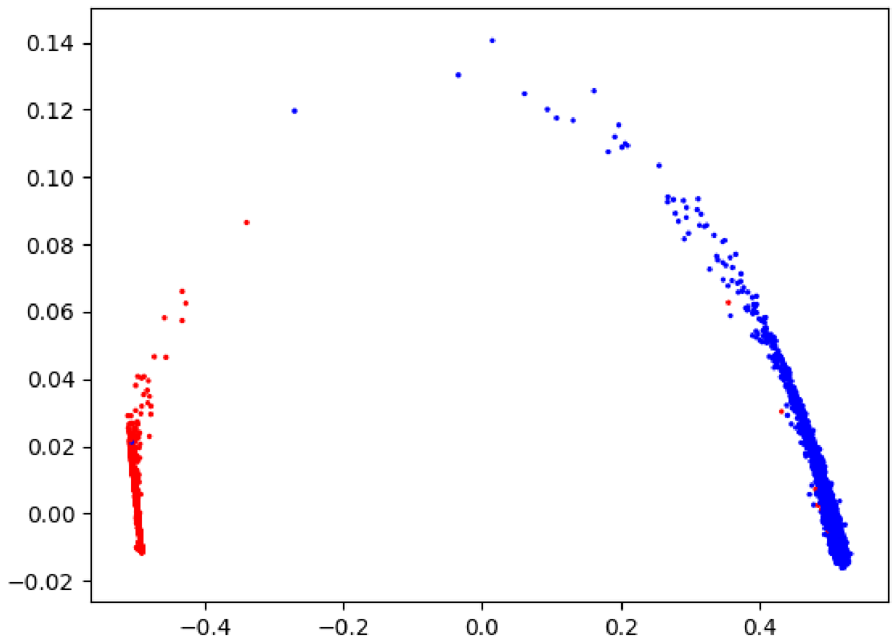
Since there was a very high likelyhood of overfitting, I did a 70/30 training-testing data split and evaluated the results. For some numerical data, I decided to evaluate it on the accuracy of determining if two random images are of the same type based on the distance between the feature descriptors. For each epoch, the optimal distance threshold was always used which maximized accuracy. This was done just for stability of the results since the optimal threshold changed as the network got better at enforcing the margin constraint. While this may seem strange compared to plotting the results with a fixed margin, it doesn’t significantly unfarily bias the data to higher accuracies. For example, running the final model over the testing data, any margin between 0.4-0.8 had an accuracy of 74-75% over all of the testing data. Looking at the training data feature visualization, since the margin is quite significant at the end, classification isn’t significantly affected by threshold even if it’s only roughly tuned.

While the training accuracy became very low, the network was highly overfit on the dataset. While this was counteracted to a degree by the introduction of the dropout layer, more normalization and data augmentation would be needed to better generalize the network.





Scatterplot of Features of Training data after PCA to 2 dimensions.



Scatterplot of Features of Testing data after PCA to 2 dimensions.