Effect of Training Size on Image Prediction with Neural Networks

# Background

Data scientists take great care to make sure that models are trained on diverse datasets to increase accurate predictions on unseen data. The logic here is simple: a model’s ability to predict hinges on having seen enough examples to learn necessary patterns. Data scientists address these issues in preprocessing via stratified sampling or in model redevelopment due to concept drift. The size of the training set may also be important - the more data used in training, the more examples seen by the model and therefore more accurate predictions on unseen data.

This report will explore the relationship between training size and prediction accuracy on an image classification problem. The Cats and Dogs training dataset from Kaggle will be used for training, validation, and testing. Specific features in image classification including data augmentation and pre-trained models will be discussed in relation to the importance of the size of the training set and its impact on model accuracy.

# Limitations

During my time in the MSBA program, one topic frequently discussed is the computational expense of machine learning models / algorithms. Often, this comes up in discussions about the tradeoff between computational expense with model selection, hyperparameter tuning, validation techniques and more. When running image classification models, machines with a GPU are often required to handle the needed computing power, especially when pre-trained convents/networks are being deployed dur to the number of parameters. While my MacBook Air has served me well in this program, it finally hit its limit with image classification tasks. Often the models would make it through validation, but during the finalization process would shut down R, preventing me from using the model in the test phase.

While this was often disappointing and frustrating it has provided me with a better understanding of computational expense and machine limitations. Up until this point, this was a topic that was often discussed but maintained an air of the abstract since it had not been an issue. I will reference these limitations in the report below, and all code is provided in the associated R files.

# Building the “From Scratch” Models

The from scratch models will be built using three different layers: convolution layers, max pooling layers and dense layers. Convolution layers are a best practice for image classification because they learn local patterns which are translation invariant, and they can learn special hierarchies of patterns. The max-pooling layer aggressively down samples feature maps to force the model to look at the totality of the input and to reduce overfitting through parameter reduction. Finally, the dense layer it used to make the final prediction.

To make the model more generalizable data augmentation and a dropout layer will be added to the model. Data augmentation generates more training data from the existing training data by implementing a series of random transformations which should help reduce overfitting. Adding a dropout layer randomly eliminates a specified proportion of units in seen and unseen layers in order lessen complex co-adaptions which can capture noise and lead to overfitting.

Chart, scatter chart

Description automatically generated

While the construction of the models was the same, the three models were developed on various amounts of training data with equal amounts of cat and dog photos: 1000 samples, 2000 samples and 4000 samples. The same validation and test set of 500 samples each was used for the various models. The results can be seen below.

Model Trained on 4000 Samples

Model\_1000 test accuracy: **75.8%**

Model\_2000 test accuracy: **76.8%**

Model\_4000 test accuracy: **Undetermined** (due to computational expense.). *Highest validation accuracy was 81%. See visual above.*

Even though accuracy improved as the training samples increased, the gains were relatively small. For example, the difference in accuracy between the Model\_2000 and Model\_1000 was only **1%** even though the former had twice as many training samples. The reasons for this can be explained by the properties of convolution and data augmentation.

As previously stated, convolution is translation invariant which means that it can recognize a pattern in differing locations, which requires less samples in applying patterns. Furthermore, data augmentation generates more training data from the existing training set. Both of these techniques allow for convolution neural networks to make accurate image predictions with fewer training samples then might be expected.

# Pre-trained Convent

Next, I will build a model using a pre-trained convent - a common and effective technique used when working with image classification problems with small datasets. The pretrained network was built via testing on the ImageNet dataset. The pre-trained network is large enough and generalizable so that features learned can be applied to the dog and cats image classification problem. Specifically, I will be using the VGG16 architecture developed by Karen Simonyan and Andrew Zisserman in 2014 for feature selection. I will implement this by using their convolution bases in my model. I will also freeze the convolution base so that the representations and weights learned in the pre-trained convolution base are not modified during my training. In development of this model, I will also be implementing fine-tuning which consists of unfreezing the top layers after the model has been trained with frozen layers in order adjust the model and make it more relevant to the dogs and cats classification problem. As in the previous models, dropout was used a regularization technique.

Graphical user interface, chart, histogram

Description automatically generated

Model Trained on 4000 Samples

As stated at the beginning of the report, the computational expense of running image classification tasks on my machine is great – and often meant that the model did not finalize after the final validation due to the lack of a GPU. However, all the code was written and executed and is included in the associated word file. The image to the left shows the validation accuracy using a 4,000 samples during training was around 90%. We can compare to some extent to the results in our text book which used a sample training size of 2000. Here again we do not see large changes in accuracies in this case because of data augmentation, but because the base of our convolution model was trained on a large data set.

# Conclusion

Image classification tasks can be trained on relatively small samples and achieve a high-degree of accuracy on unseen data. These techniques that make this possible include: **Convolution Layers:** Translation Invariant allows for pattern recognition in differing locations; **Data Augmentation:** Creators additional training data via random transformations; **Pre-Trained Networks :** Use generalizable pre-trained layers that were developed on large training sets. These techniques prove powerful at developing accurate image classification models on relatively small training sets of data, lowering the barriers for image classification tasks.