562 Final Project

Marichi Gupta, Chris Latimer, Ha Eun Lee, Sean McCaffery

December 10, 2019

1 Introduction

Proteins are one of the essential building blocks of life. They are crucial elements of almost all cellular functions and enable all life forms to develop, simple and complex. Understanding which proteins affect what parts of biological life is a complex exploratory task for scientists. Manual classification of proteins is a labor and time intensive task that can no longer meet the throughput of automated microscopy. Therefore biomedical image analysis is a necessary accelerator for the scientific communities around human cells and disease.

In this project, our goal is to classify mixed patterns of proteins in microscope images. We hope to identify specific proteins in crowded images to determine protein location in high-density images. Specifically our goals are 1) to label images with a predicted protein to allow humans to be able to select the proteins they wish to evaluate, and 2) to have these image classifications reflect the labeling that scientists would manually perform.

Microscope images are the natural choice for protein classification as automated microscope photography is outpacing the throughput of manual classification. Therefore labeling the images in an unprocessed form prevents any additional steps for scientists between photography and classification. The images may contain multiple proteins, meaning that this is a multi-classification problem where an input can and should be given multiple labels, and may be sparse as well.

We are using two approaches. First, we take the a bare bones approach to convolutional neural net (CNN) to create our baseline model. In the second approach, we changed the model to address certain deficiencies that were seen in the baseline model, creating a more sophisticated overall neural network model. We then evaluated the success of our approaches.

2 Related Work

Cellular classification is by no means a new concept. There is a vested interest in automated classification of images in the sciences as means of collecting data sore past the capacity of scientists to manually identify. Many top bio-tech companies are predicated on using Machine Learning to automate portions of previously manual and labor intensive scientific protocols such as cellular classification. The ability for researchers to have Machine Learning pipelines mass process and classify their data will drastically increase the speed of breakthroughs as data processing is a top blocker for scientific research.

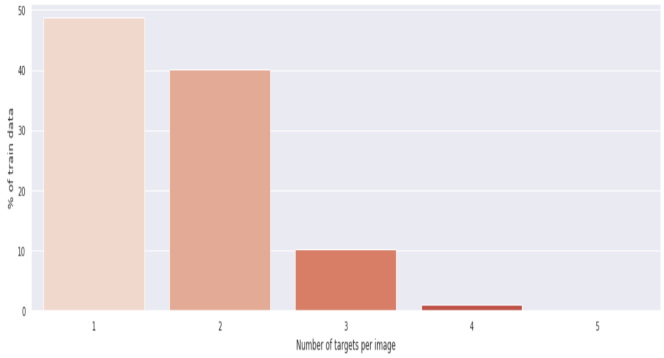
This project is based off of a featured prediction competition from Kaggle[1] to use Image Classification to label human proteins for The Human Protein Atlas[2]. We built upon their work for the data pre-processing steps, and built our own model configurations on top. Because our reasoning behind the model setup were different, we ended up with different model configurations and results.

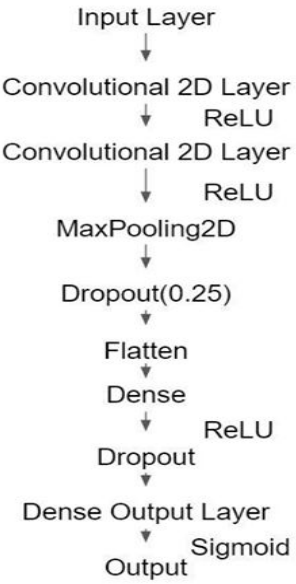
3 Dataset and Features

Cellular image data was obtained from thousands of PNG files and the correct labels for training data were obtained from a CSV provided by the Human Protein Atlas. This CSV contained 31072 samples, of which 11702 are for predictions. The data is stored in a one-hot table depicting multi-classification of proteins within the sample.

	Id	Target	Nucleoplasm	Nuclear membrane	Nucleoli	Nucleoli fibrillar center	Nuclear speckles	Nuclear bodies	Endoplasmic reticulum	Golgi apparatus	Peroxisomes
0	00070df0-bbc3-11e8-b2bc-ac1f6b6435d0	[16, 0]	1	0	0	0	0	0	0	0	0
1	000a6c98-bb9b-11e8-b2b9-ac1f6b6435d0	[7, 1, 2, 0]	1	1	1	0	0	0	0	1	0
2	000a9596-bbc4-11e8-b2bc-ac1f6b6435d0	[5]	0	0	0	0	0	1	0	0	0
3	000c995a-bba4-11e8-b2b9-ac1f6b6435d0	[1]	0	1	0	0	0	0	0	0	0
4	001838f8-bbca-11e8-b2bc-ac1f6b6435d0	[18]	0	0	0	0	0	0	0	0	0

The features of this data are each of the potential proteins that may be in the sample, with most samples having one or two targets.





4 Methods

Here, we discuss classification techniques and our application of Convolution Neural Networks (CNN) to the problem of identifying human proteins.

4.1 Baseline Model

The first model is called the Baseline Model, as it is a simple Convolution Neural Network (CNN) model. Implemented with a Keras Sequential Model, this means that it only has two 2D convolution layers each with a standard 3x3 kernel size and a Rectified Linear Unit (ReLU) activation function,

$$y = max(0, x)$$

and a dense output layer[3]. The final step in the model is a sigmoid activation function

$$y = \frac{1}{1 + e^{-z}}$$

which the output layer uses to determine what it should output.

Baseline Model Diagram:

4.1.1 Choice of Output Layer Activation Function

An unusual part of this model is that a sigmoid function is used as the output layer activation function and this is a multiclass classification problem. Usually the last layer in multiclass classification problems are softmax, a regression function whose goal is to minimize the loglikelihood,

$$l(\theta) = \sum_{i=1}^{n} \log \prod_{l=1}^{k} \left(\frac{exp(\theta_l^\top x^{(i)})}{\sum_{j=1}^{k} exp(\theta_j^\top x^{(i)})}\right)^{1\{y^{(i)}=l\}}$$

Where

$$p(y = i|x) = \phi_i$$

is given by

$$p(y = i|x) = \phi_i = \frac{exp(\theta_i^\top x)}{\sum_{j=1}^{k} exp(\theta_j^\top x)}$$

However, because more than one protein can be present in a given image, it's best to use the sigmoid function instead of the softmax.

4.2 Improved Model

The second model aims to improve upon the first by increasing training as well as the complexity of model. We added an intermediary Dense layer in between the convolution layer and the final layer,

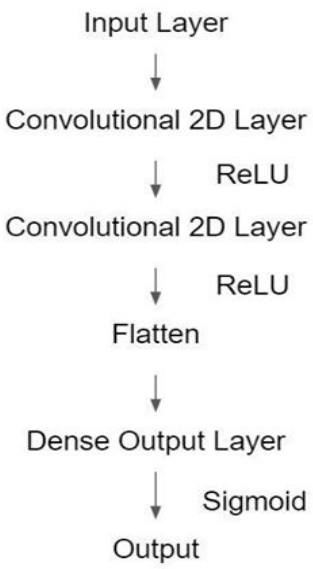
but this resulted in signs of overfitting so we attempted to combat this by integrating features that would help prevent overfitting.

We added in two dropout layers which essentially remove nodes in the processing of data to ensure that the weights we are calculating are not overly dependent upon single features. Additionally, we added a pooling layer to help make features transition independent, or that the model focuses on the actual features we care about rather than background noise.

As the major issue we faced in our initial attempts at fixing the problems of the Baseline model was overfitting, in addition to implementing measures inside the convolution networks, we also decremented the hyperparameter of the number of epochs, or iterations that we were running the network, to 4 as 5 seemed excessive and exacerbated overfitting.

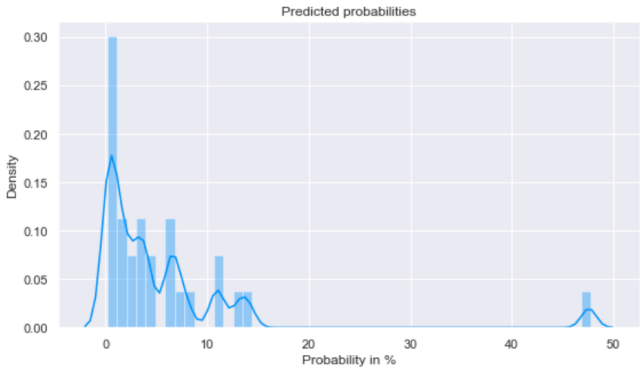
We also decreased the number of classes that the model was attempting to predict in a bid to reduce the validation error, or how inaccurate the model is at predicting that samples in fact belong to a class. Calling the classes that we would have the model focus on a "wishlist", which were a group of the most important proteins to classify, which we then had the model just attempt to predict.

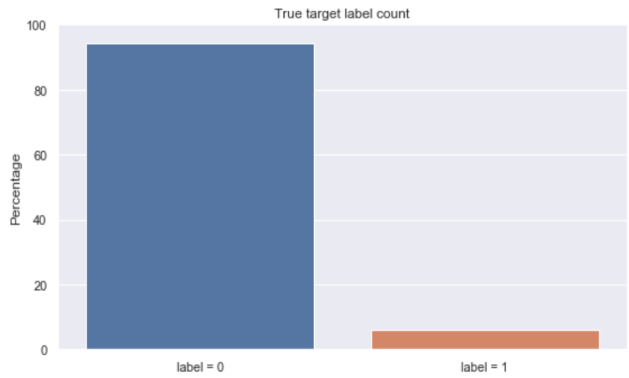
Improved Diagram:



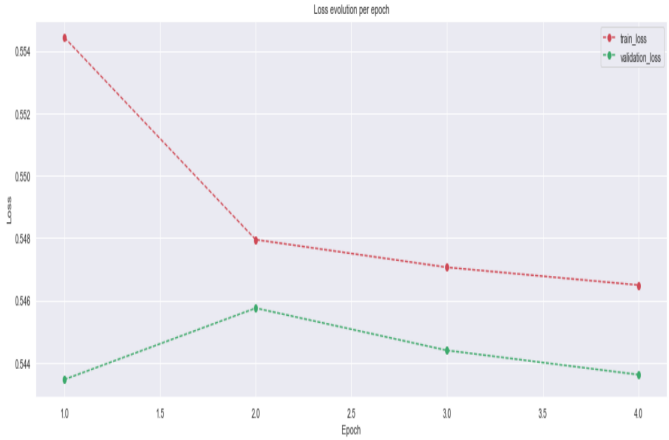
5 Results and Discussion

The Baseline model at face value had obtained extremely high accuracy, almost 95%. However, when we looked into the actual predictions further, that was shown to be misleading. Though the model was able to correctly identify when samples were not of a certain class consistently, the predicted probabilities for samples actually belonging to any of the classes were exceptionally low, as shown in Figures below.

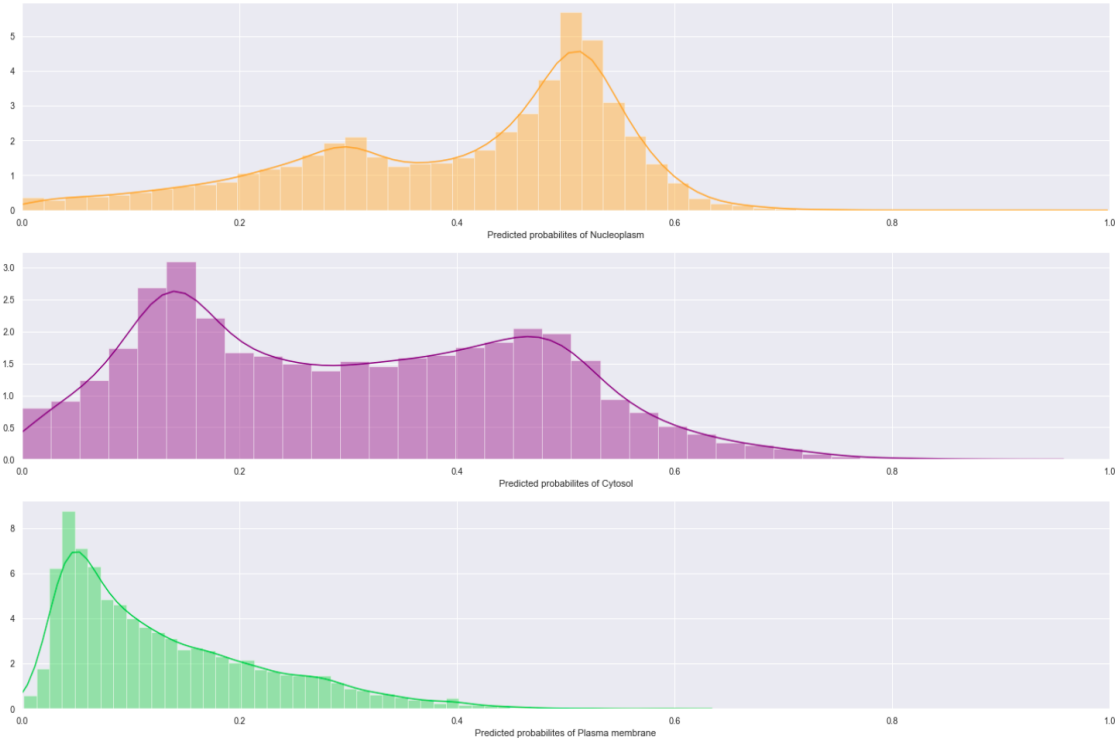




Along these same criteria, the Improved model performed much better than the baseline. We can see this improvement primarily in the decrease of validation error over the epochs, meaning that our model is getting better at distinguishing when a sample does in fact belong to a class.



This improved insight can be seen in the increasingly bimodal distribution of our probability predictions for samples. Whereas before the probabilities for if a sample belonged to a class were primarily clustered and all exceptionally low, as seen in these figures, they have become much more spread out and bimodal. The bimodality is important to us because this is what indicates that the model is now able to effectively differentiate between when a sample is or is not in a certain class.



6 Conclusions and Future Work

Of the two models that we implemented, the Improved one was clearly more effective. By using multiple epochs, our validation error decreased significantly, allowing our accuracy to more accurately reflect the robustness of the models. Moving forward, we would be interested in increasing the number of convolution layers that we include as well as the size of the kernels that we use in them. Also, increasing the size and diversity of the wishlist would allow us to obtain multiple accurate models that we could stitch together to have a comprehensive one that accurately predicts all of the classifiers.

7 References

References

[1] F., Laura. “Protein Atlas - Exploration and Baseline.” Kaggle, Kaggle, 12 July 2019, https://www.kaggle.com/allunia/protein-atlas-exploration-and-baseline

[2] “The Human Protein Atlas.” The Human Protein Atlas, Human Protein Atlas, 5 Sept. 2019, https://www.proteinatlas.org/.

[3] Convolutional Layers - Keras Documentation, https://keras.io/layers/convolutional/.