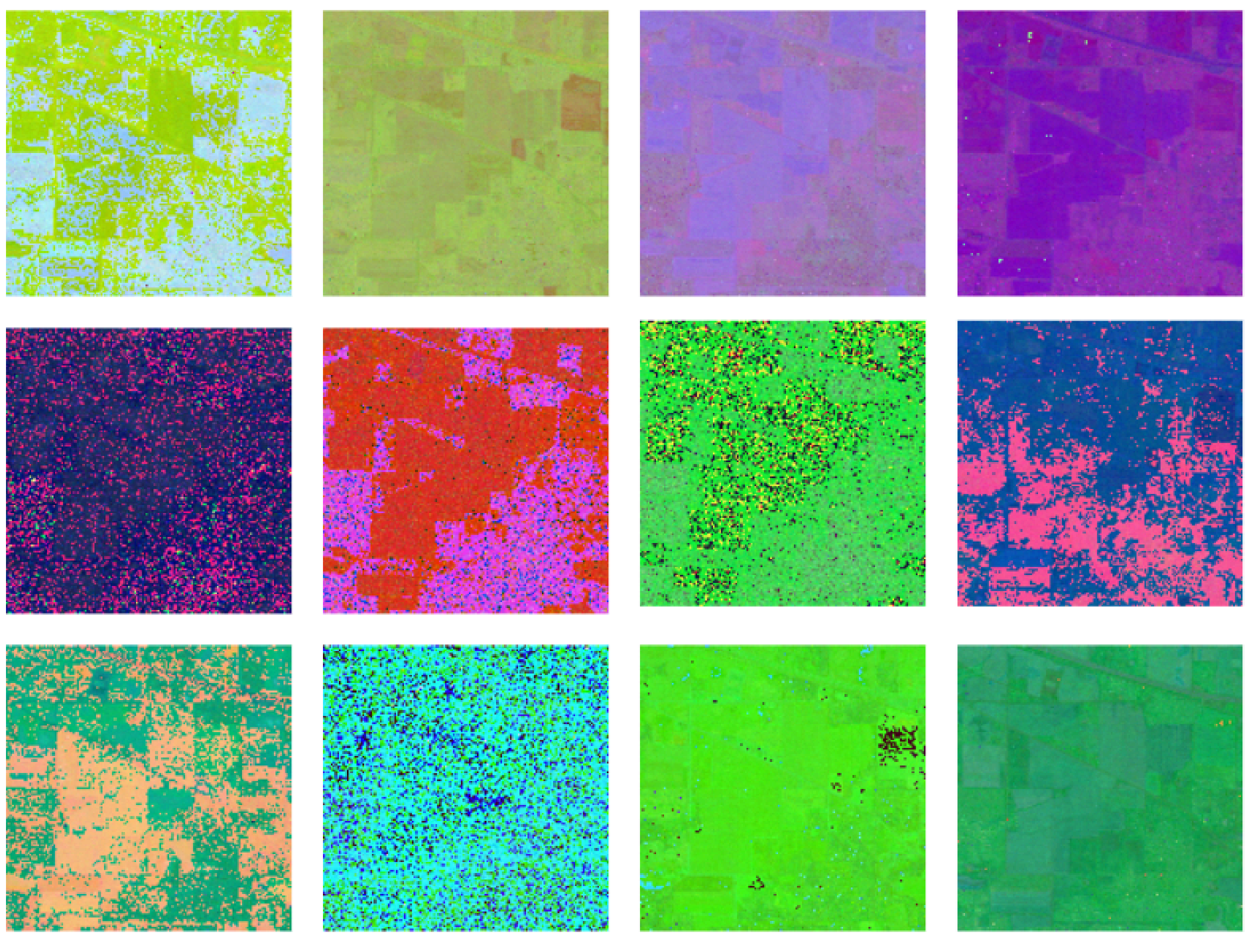
Indian Pines Hyperspectral Image Classification



By: Mitchell Dodson and Josh Chapman

Introduction

The goal of this project was to build a multi-layer perceptron based classifier that could accurately predict an object type from the Indian Pines dataset based on its hyperspectral intensity characteristics. Figure 1 shows a monochrome sample image of the dataset. Figure 2 shows a RGB sample image of the Indian Pines scene.The Indian Pines dataset contains spectral radiance data in the visible and near infrared range from the airborne hyperspectral camera (AVIRIS) over a landscape in Indiana, US [1]. Each pixel location contains a given intensity (W cm-2 nm-1 sr-1) in 220 unique reflectance bands. The Indian Pines scene contains many types of crops, vegetation, and many other types of objects. About two-thirds of the scene is agriculture and the other third is forestry or other vegetation [4]. Some of the water absorption bands have anomalously high reflectance values.

This dataset contains 16 different classes with some classes having a larger proportion of samples compared to others. Figure 3 shows each class location in the Indian Pines scene. Table 1 shows the ground truth classes with each class's number of samples in the dataset.

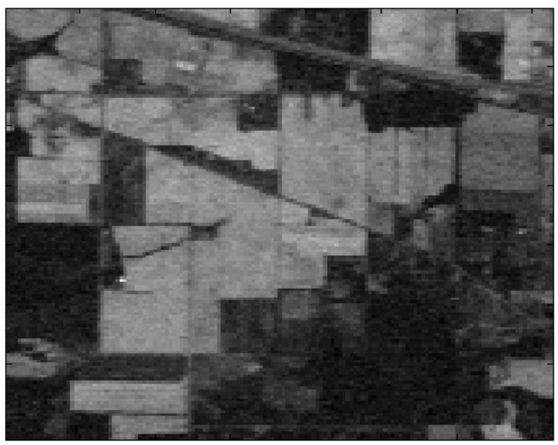


Figure 1: Sample Band of Indian Pines Dataset [4]

Figure 2: Domain RGB Truecolor

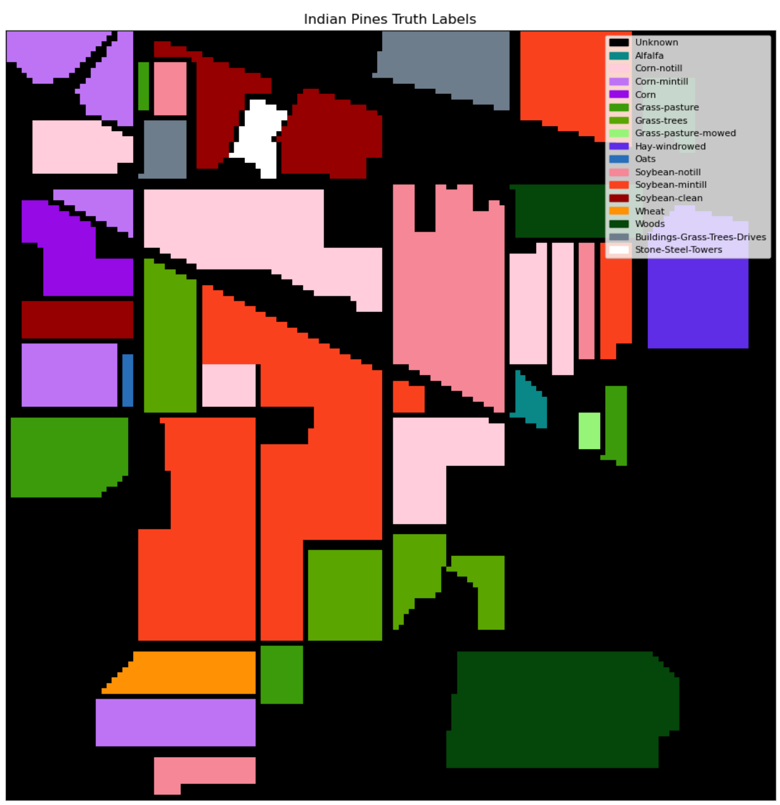


Figure 3: Indian Pines Truth Labels

Groundtruth classes for the Indian Pines scene and their respective samples number

#	Class	Samples
1	Alfalfa	46
2	Corn-notill	1428
3	Corn-mintill	830
4	Corn	237
5	Grass-pasture	483
6	Grass-trees	730
7	Grass-pasture-mowed	28
8	Hay-windrowed	478
9	Oats	20
10	Soybean-notill	972
11	Soybean-mintill	2455
12	Soybean-clean	593
13	Wheat	205
14	Woods	1265
15	Buildings-Grass-Trees-Drives	386
16	Stone-Steel-Towers	93

Table 1: Groundtruth Classes for Indian Pines

Method

To improve classification accuracy and performance, several preprocessing steps were performed. To handle unbalanced classes, sampling methods such as stratified sampling are used to sample an equal amount of classes for training models [2]. Figure 4 shows how proportionate and disproportionate sampling can be used to sample a given population. Stratified sampling can prevent overfitting to overrepresented classes when training a model. Proportionate sampling was used for training our models. After the proportionate sampling was performed, we limited the maximum number of training samples for a given class. For different models, we varied this maximum number of samples and compared loss and validation loss.

Another preprocessing method we used was normalization of the data. More specifically, We trained models on datasets that are linearly normalized to a standard gaussian. This is done either with separate mean and standard deviation coefficients for each band, or by using the universal statistics. Ultimately, we found that independently normalizing bands was best because it mitigated the anomalously high absorption channel intensities. Figure 5 shows the Indian Pines spectral intensity before the normalization, and Figure 6 shows the normalized spectral intensity.

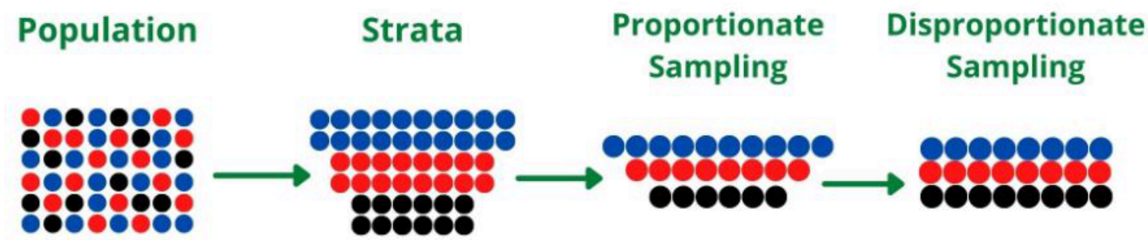


Figure 4: Sampled Population Strategies

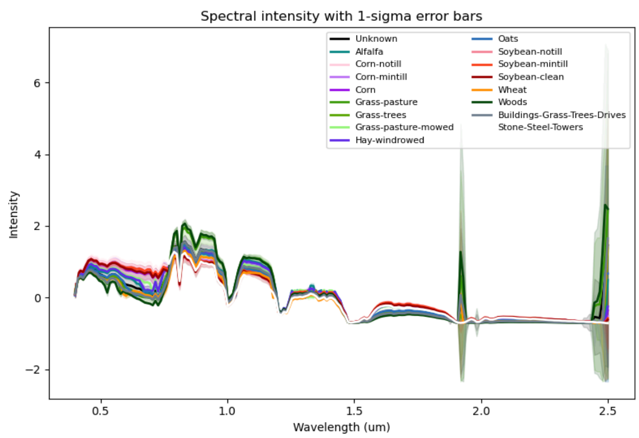


Figure 5: Indian Pines Spectral Intensity

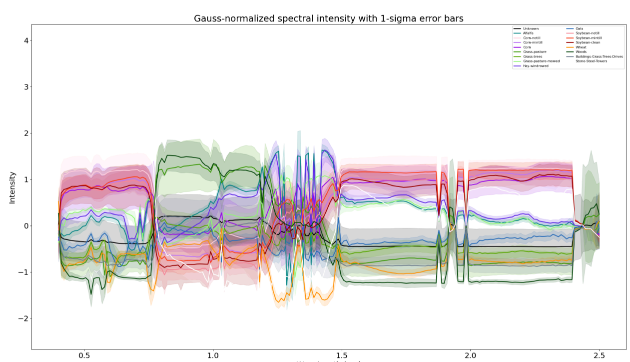


Figure 6: Indian Pines Normalized Spectral Intensity

Results

Figure 7 shows the full collection of learning curves associated with all of the training runs we executed while varying hyperparameters, and Figure 8 displays the training and validation loss values after training each of the models for at most 256 epochs. Training was stopped before reaching this threshold in cases where the validation loss failed to decrease for more than 64 epochs. The most performant models had a categorical cross-entropy loss value around .25 for training data and .5 for validation data.

We found that this problem is very vulnerable to overfitting, probably due to the large variety of sample counts and small data volume. For this reason, models with more than 2 or 3 layers and with layers that have more than about 128 nodes tended to perform worse than more reduced models.

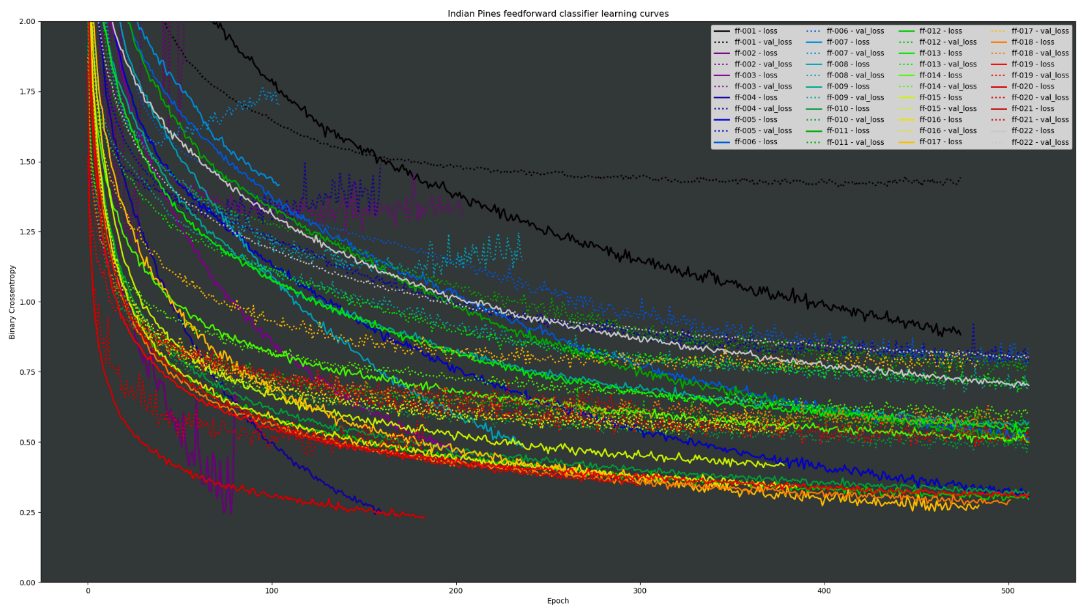


Figure 7: Indian Pines Feedforward Classifier Learning Curves

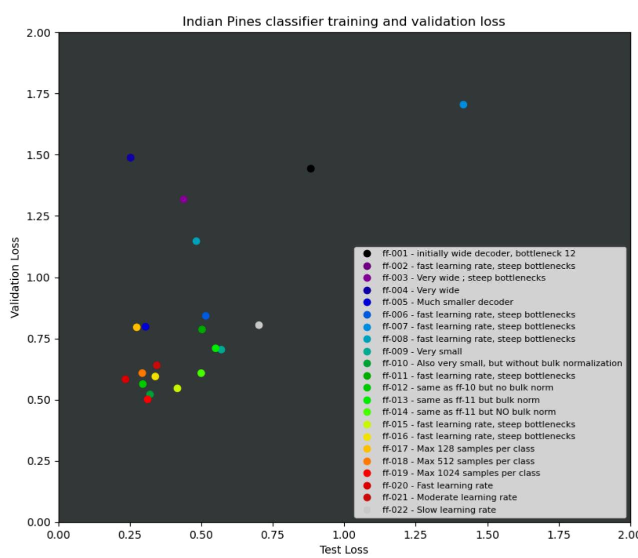


Figure 8: Indian Pines Classifier Training and Validation Loss

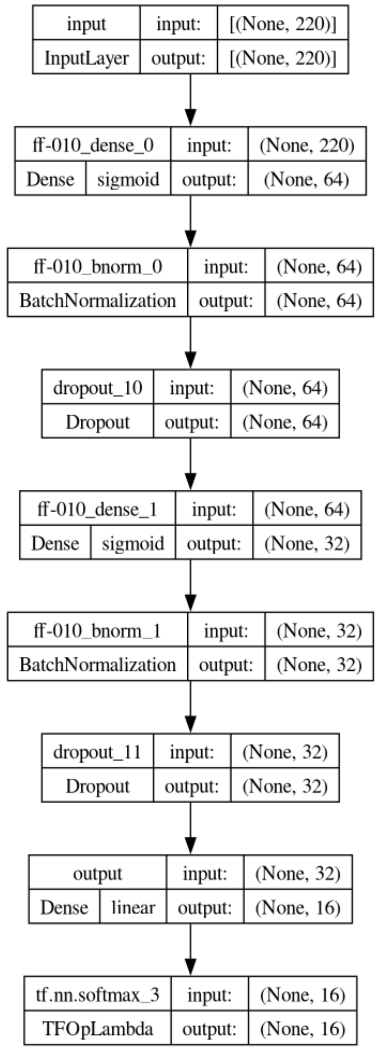


Figure 9: Best Model Layer Architecture

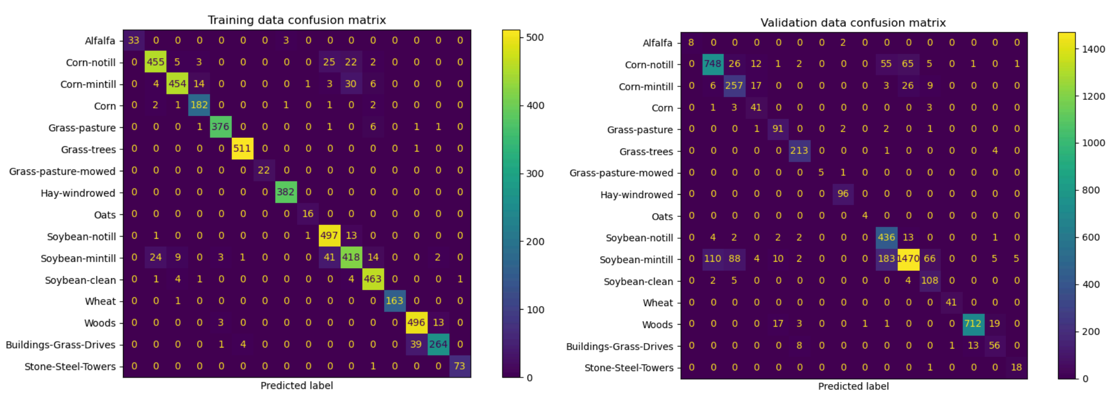


Figure 10: Training and Validation Data Confusion Matrices

Figure 9 shows the architecture of the best model we identified, which uses batch normalization, a dropout rate of 20%, learning rate 1e-4, and a logistic loss function. The results from this model are characterized by Figure 10 and Figure 11. This model only has 2 hidden layers consisting of 64 and 32 nodes, but was better at identifying classes having few samples than models with more layers or more nodes per layer.

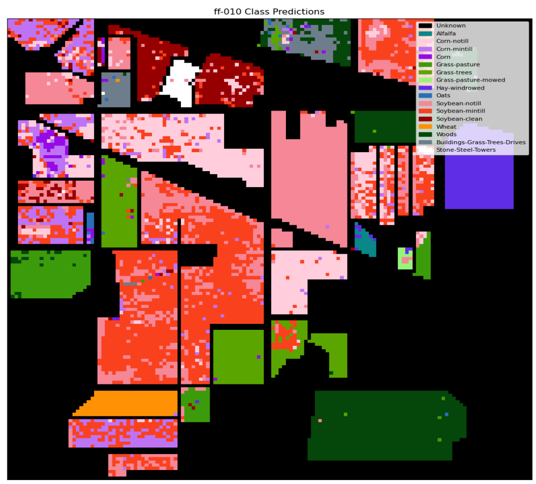


Figure 11: Model ff-010 Class Predictions

Unsurprisingly, the model had a tendency to confuse classes that consist of the same crop at different stages in its cultivation, for instance soybeans with minimum or no tilling. Soybeans and corn were most often mistaken for each other outside of their class type, but other surface classes like Oats, Stone-steel-towers, Grass, and Woods were classified confidently by the model, despite having few samples in some cases.

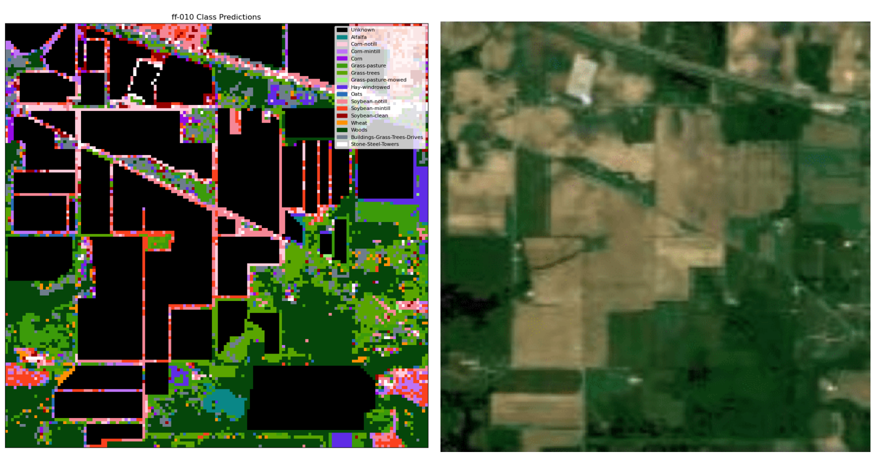


Figure 12: Unknown Class Predictions and true color RGB

Figure 12 shows the predictions made by the model over pixels marked “unknown” in the dataset, compared to an RGB true-color image over the domain. Subjectively, the model seems to do a good job of picking up the boundaries of treelines and buildings/parking lots. It also seems to consistently predict the corresponding crop type of pixels adjacent to labeled fields, and to classify roadways as “stone”.

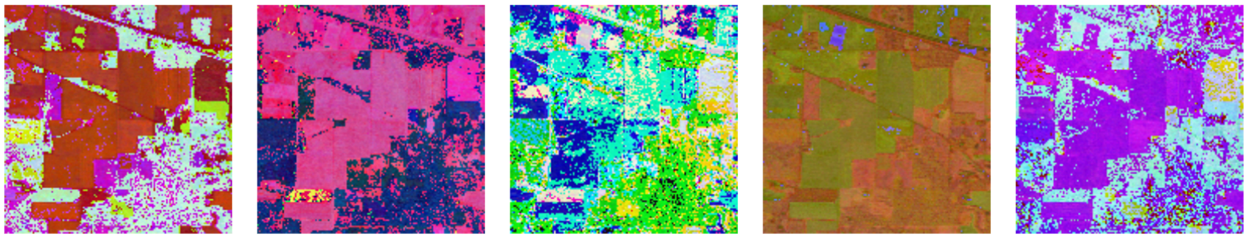


Figure 13: Final Hidden Layer Principle Components

Figure 13 demonstrates the first 15 principle components of the 16-dimensional final hidden layer. Each image is generated by setting the red green and blue channels of the RGB image to the next 3 principle component activations of the layer given the full dataset. While they don’t necessarily convey interpretable meaning, they may give some indication of how the model is determining its predictions. For example, the first principle component forming the red channel of the first image seems to strongly separate tilled fields from the adjacent buildings and vegetation, and the 3rd (blue) principle component in the fourth image tends to emphasize buildings and roadways. The wheat field in the lower left is clearly resolved in each of the RGBs, which may correspond to the model’s near-perfect performance in predicting it.

Figure 14 shows a second experiment we conducted where the model described above was re-trained while varying two hyperparameters: learning rate, and the maximum number of samples allowed from a specific class. The results suggest that starting with an initially fast learning rate enables the model to quickly find a fairly good solution, but continuing to train causes overfitting, as indicated by the distance between the validation and training curves. Moderate learning rates and slow learning rates exhibit less noisy learning curves, but converge more slowly and with lower accuracy.

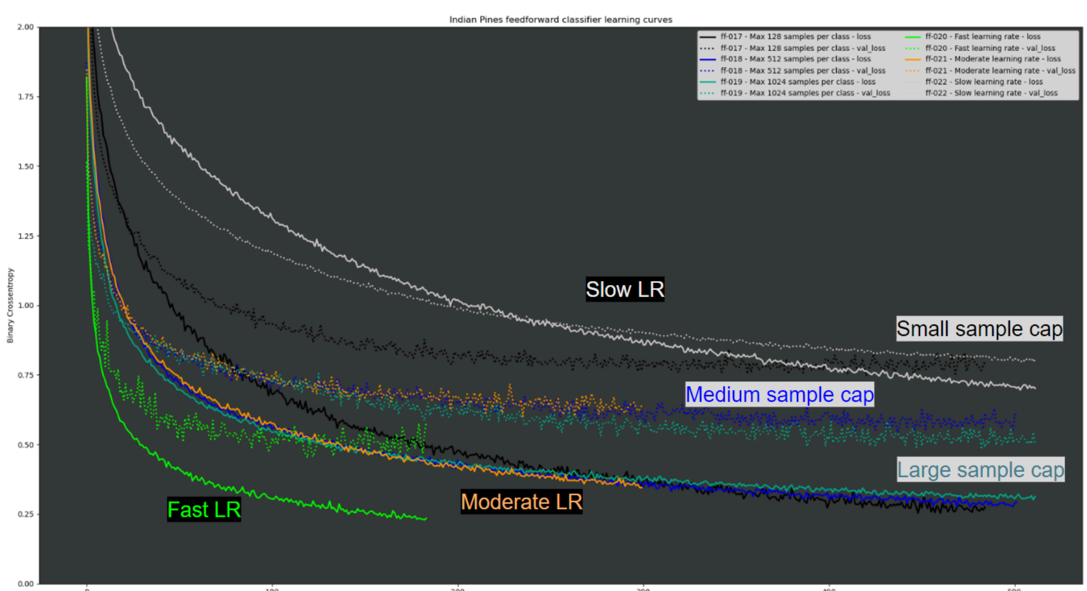


Figure 14: Indian Pines Classifier Learning Curves for Varied Learning Rates and Max Samples

Furthermore, separately increasing the maximum number of samples from each class included in the training data universally caused the validation loss curve to decrease, and the training loss curve to slightly increase. This is deceptive, though, because adding additional training samples for large classes only improved the model's ability to predict (and over-predict) those classes. In general, increasing the sample cap diminished the prediction accuracy of samples with few classes.

References

[1] https://paperswithcode.com/dataset/indian-pines
[2] https://www.geeksforgeeks.org/stratified-sampling-in-pandas/
[3]https://www.linkedin.com/advice/0/how-can-you-use-sampling-prevent-overfitting-your-dun5f#:~:text=One%20approach%20is%20to%20employ,to%20specific%20patterns%20or%20outliers.
[4]https://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes#Indian_Pines