Report

For this project, we are going to predict field anomalies for each pixel in an image. In this report, we did a thoroughly analysis on our training data set to see the data distribution pattern. Four features are used in the analysis, including 3 RGB values and 1 infrared value. For computational efficiency and data visualization purposes, we took one random pixel from each training image (512 x 512 pixels) and got a total of 12901 uncorrelated pixels. Table 1 shows the label distribution of the selected pixels.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | background | cloud\_shadow | double\_plant | planter\_skip | standing\_water | waterway | weed\_cluster |
| Number | 10173 | 396 | 96 | 23 | 204 | 149 | 1860 |

Table 1. label distribution of the selected pixels.

From Table 1, we can see that majority of the pixels in the training data are background label, which is reasonable from our inspection.

We plotted the data distribution of the selected pixels regarding to the RGB and infrared values.

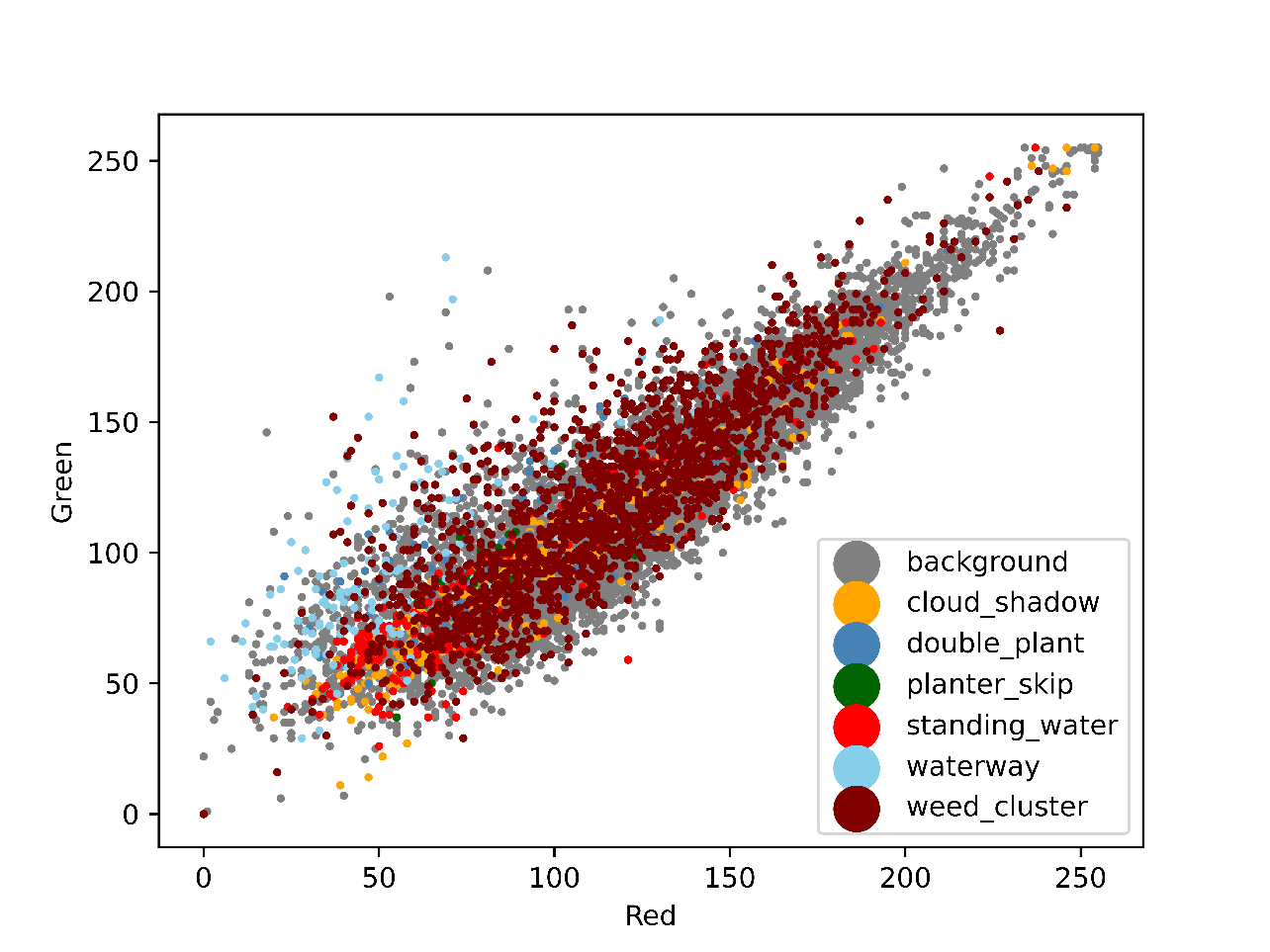


Figure 1. data distribution in red and green domain.

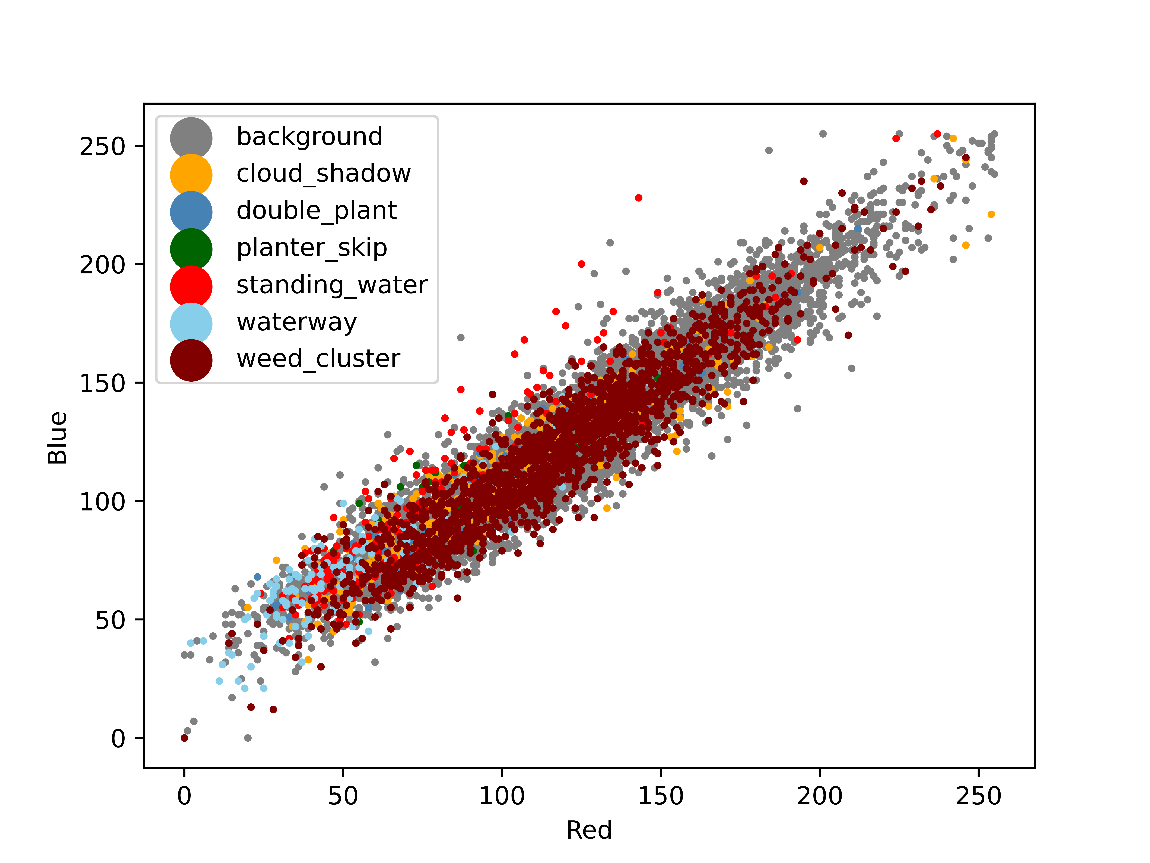


Figure 2. data distribution in red and blue domain.

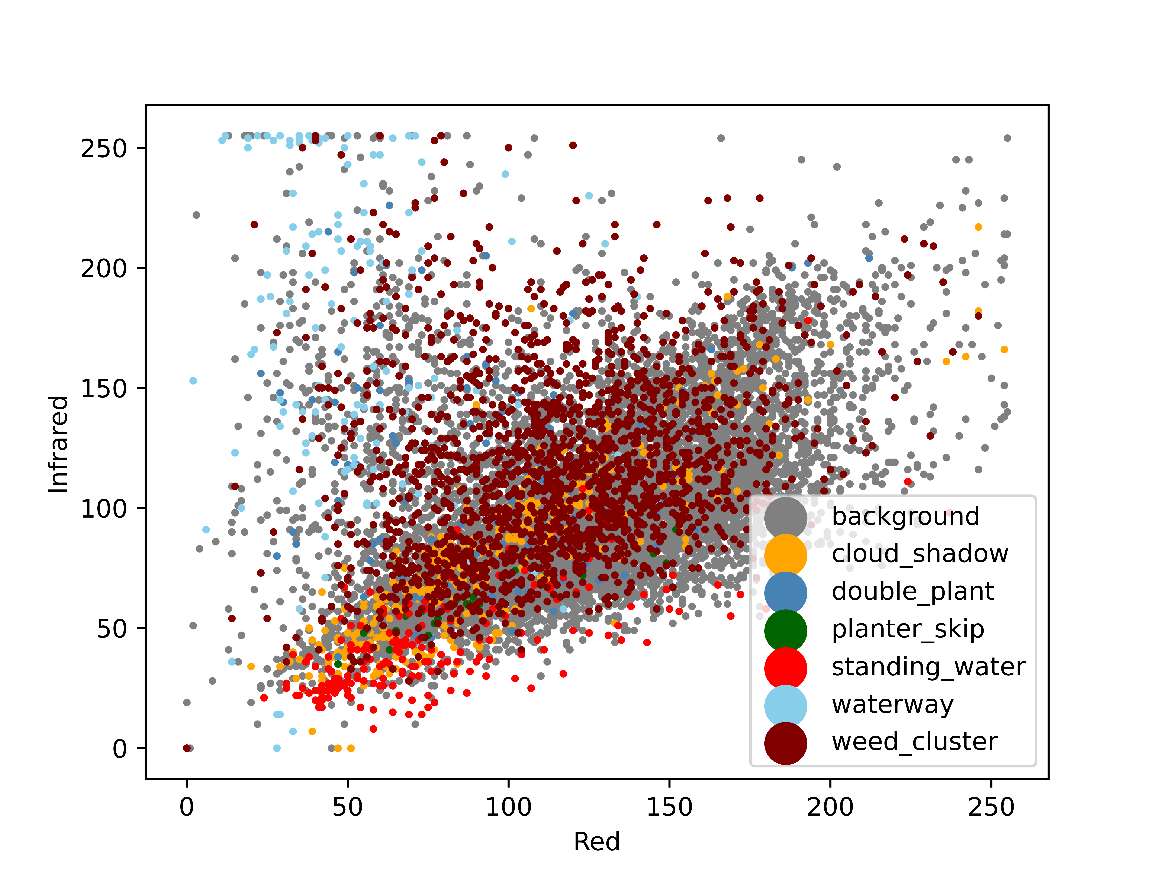


Figure 3. data distribution in red and infrared domain.

From Fig 1 and 2, we found that there is no clear clustering pattern and noticeable outliers for any label. Comparing to other labels, “standing\_water”(red) and “waterway”(lightblue) pixels generally have low RGB values. It is also noticeable that the background label spans across the figure, indicating the difficulty of separating it from other labels.

In the infrared domain, clearer clustering pattern can be seen for “standing\_water”(red) and “waterway”(lightblue) labels. We also saw outliers for “waterway” label with high infrared value.

From the data inspection, we can see that the RGB values of the pixels carry limited information. Data points show a linear trend with very few outliers and all labels show uniform distribution. On the contrary, the infrared value provides more information showing separation between labels. The mixed data could cause a lot of problems for machine to learn, because it would be hard for machine to separate each label, especially the “background” label shares similar characteristics as other labels.

After the general analysis, we further investigate pixels in a single specific image to see the differences between field anomaly and background labels.

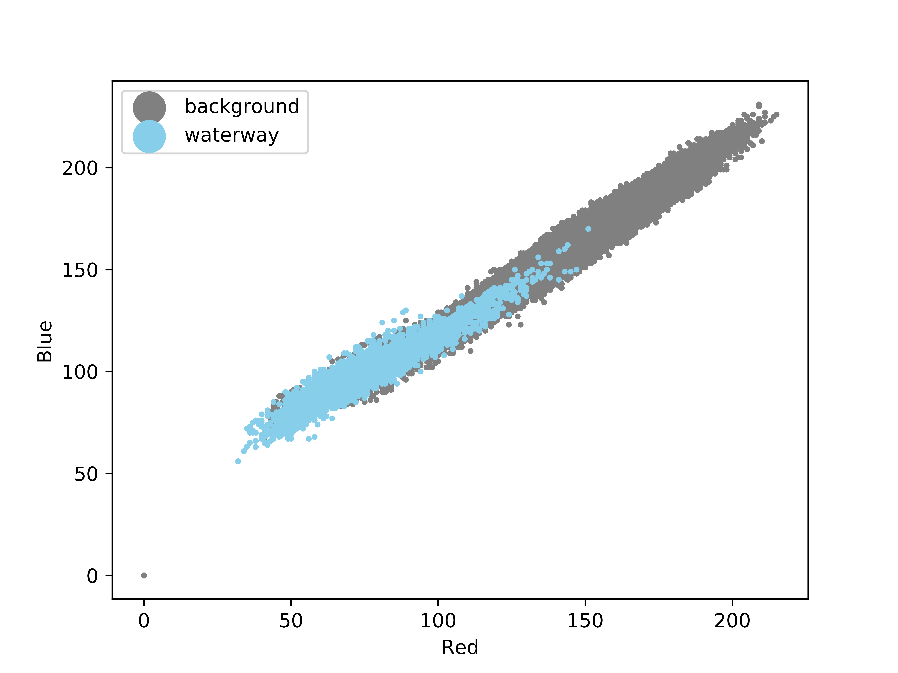


Figure 4. example image and waterway vs background RGB value distribution

From single image analysis, although pixels are still mixed in RGB values, better separation can be found for the two labels. However, for both labels we saw a wide range red and blue values (~50 – 200), indicating a weak clustering pattern. In conclusion, single image analysis results show a better label separation than the previous random pixel analysis, illustrating the importance of the neighboring effect of the pixel group instead of treating every pixel as independent.

At this stage, we only considered four features of the training data set (red, green, blue, and infrared values). However, red, green, and blue values share very similar characteristic, and could be reduce to the average of the three values. We believe that adding more features is necessary for better training result.

We used simple linear methods, logistic regression and SVM, to train the dataset. In both methods, we randomly selected 14000 pixels from the training dataset. Among these training data points, each label 0-6 has 2000 randomly chosen pixels.

**Training with logistic regression**

**1. Using 4 features: 3 RGB and 1 nir values**

We choose learning rates among 1e-7, 5e-7, 1e-6, 5e-6, and regularization values from 5e4, 1e5, 5e5, 2e7, 1e8.

The loss of the model is 1.9458943392059134. The following table shows the training accuracy at each learning rate and regularization:

|  |  |  |
| --- | --- | --- |
| Learning rate | Regularization | Training Accuracy |
| 1e-7 | 5e4 | 0.035714 |
| 1e-7 | 1e5 | 0.142857 |
| 1e-7 | 5e5 | 0.142857 |
| 1e-7 | 2e7 | 0.159429 |
| 1e-7 | 1e8 | 0.157571 |
| 5e-7 | 5e4 | 0.265143 |
| 5e-7 | 1e5 | 0.142857 |
| 5e-7 | 5e5 | 0.136857 |
| 5e07 | 2e7 | 0.159857 |
| 5e-7 | 1e8 | 0.164071 |
| 1e-6 | 5e4 | 0.048571 |
| 1e-6 | 1e5 | 0.142929 |
| 1e-6 | 5e5 | 0.154357 |
| 1e-6 | 2e7 | 0.158357 |
| 1e-6 | 1e8 | 0.187214 |
| 5e-6 | 5e4 | 0.161071 |
| 5e-6 | 1e5 | 0.158929 |
| 5e-6 | 5e5 | 0.159143 |
| 5e-6 | 2e7 | 0.172500 |
| 5e-6 | 1e8 | 0.290714 |

The highest accuracy trained by this multi-class logistic regression is 0.290714 using learning rate 5e-6 and reg 1e8. The confusion matrix for these parameters is:

[[1701 0 0 228 0 71 0]

[ 787 0 0 1213 0 0 0]

[1224 0 0 75 0 701 0]

[ 958 0 0 1030 0 12 0]

[ 731 0 0 1269 0 0 0]

[ 659 0 0 2 0 1339 0]

[1955 0 0 0 0 45 0]]

From this matrix, we can clearly see the phenomenon that the trained model predicts most of the pixels to label 0. This meets the previous analysis that it’s hard to separate the background with other labels, since they have close RGB values. Beside, we observe that label 5 have a better prediction, where above 60% of the points belonging to it are labeled correctly after training. This meets our expectation that ‘waterway are more easily to be classified than other labels, as it has much lower RGB values.

**2. Using two features: the average of 3 RGB values and 1 nir values**

When plotting the cluster difference, we found that the plots for r&g and r&b have a high similarity. Thus, we supposed that the 3 RGB values are dependent with one another to some extent. Then, we reduced features to only the average value of the RGB and the nir value for the training model.

The loss of this model is 1.9459041334179215, which is very close to the previous model.

We again choose same learning rates and regularization values. The following table shows their training accuracy:

|  |  |  |
| --- | --- | --- |
| Learning rate | Regularization | Training Accuracy |
| 1e-7 | 5e4 | 0.033071 |
| 1e-7 | 1e5 | 0.157643 |
| 1e-7 | 5e5 | 0.087143 |
| 1e-7 | 1e8 | 0.148286 |
| 5e-7 | 5e4 | 0.212286 |
| 5e-7 | 1e5 | 0.227500 |
| 5e-7 | 5e5 | 0.142857 |
| 5e-7 | 1e8 | 0.148357 |
| 1e-6 | 5e4 | 0.140429 |
| 1e-6 | 1e5 | 0.142857 |
| 1e-6 | 5e5 | 0.143000 |
| 1e-6 | 1e8 | 0.142857 |
| 5e-6 | 5e4 | 0.143071 |
| 5e-6 | 1e5 | 0.147143 |
| 5e-6 | 5e5 | 0.148286 |
| 5e-6 | 1e8 | 0.147929 |

The highest accuracy trained by this multi-class logistic regression is 0.227500 using learning rate 5e-7 and reg 1e5. The confusion matrix for these parameters is:

[[ 991 0 0 0 604 405 0]

[ 684 0 0 0 1149 167 0]

[1334 0 0 0 420 246 0]

[ 964 0 0 0 322 714 0]

[ 35 148 0 0 1793 24 0]

[1575 0 0 0 24 401 0]

[ 678 0 0 0 1078 244 0]]

From this matrix, we observe that the background still has more pixels to be classified as, and label 4, ‘waterway’, is getting a better prediction. However, label 5 has more wrong predictions. Thus, we can assume that RGB values for ‘standing water’ and ‘waterway’ are lower than other labels, which makes them much convenient to be separated from others. Nevertheless, their average values might be similar, making the model hard to distinguish between them.

The logistic regression fails to capture the prediction. Since there are only few features for the logistic regression to learn, which is lack of information, the training is extremely underfitting and unable to classify the correct label.

**Training with SVM**

**Next Step**

For our next step, we will start using mini batch gradient descent to continuous train small sections of the dataset until the whole data points are reached. Besides, we will build a convolutional neural network to form layers starting from 3 to see how each hidden layer will perform on the images and try different activation functions on them, such as ReLU, sigmoid, tanh, and softmax.