Project 2

Min Jae Shin & Eric Yang

May 3rd, 2019

1. Data Collection

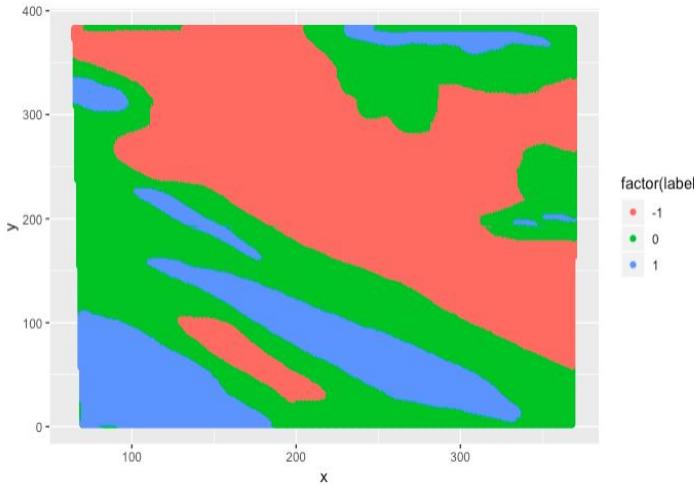
a)

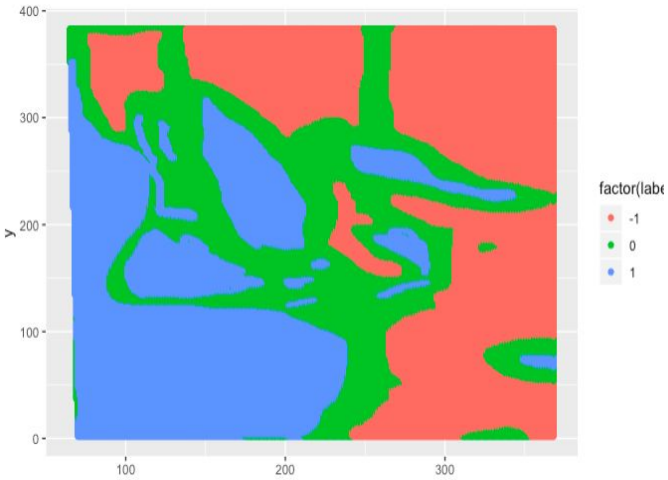
There is much discussion about global climate change and rise in temperature. Global climate models predict that global surface air temperatures will rise a certain amount when atmospheric carbon dioxide levels double. This model also predicts that the strongest effect of such model will occur in the Arctic. In the process of observing the increasing surface air temperature, cloud coverage is a crucial factor that has to be observed because clouds modulate the sensitivity of increase in surface air temperatures as atmospheric carbon dioxide levels increase. However, in the Arctic, the surface of the Arctic and the clouds have similar electromagnetic radiation and thus is hard to differentiate by using EM waves. As a result, this report seeks to find potential algorithms that will help identify the clouds properly. The data was first collected through the Multiangle Imaging SpectroRadiometer (MISR). This MISR sensor has nine cameras with each camera viewing Earth at different angles in 4 spectral bands. This MISR has a large data size because it operates fast around a 360-kilometer-wide swath on the surface and covers at high resolutions. The MISR collected data around a 16-day block over 233 MISR swath also known as paths. The MISR used to have these algorithms but were largely outdated. As a result, new algorithms that would enhance accurate and better spatial coverage cloud detection were needed. Through Exploratory Data Analysis (EDA), three features were constructed to allow more separability between the data and the constructed features and data were tested across various classification algorithms such as SVM and ELCM algorithms, the conclusion is that the ELCM algorithm itself has the best accuracy and spatial coverage than prior MISR algorithms in cloud detection. Now, the potential impact of this is that statistics can be used in the data processing process instead of just using statistics after data has been processed and this can be used in a variety of fields outside of just this particular topic as well although in this case, the report was able to use classification algorithms to analyze the data only after the data was preprocessed.

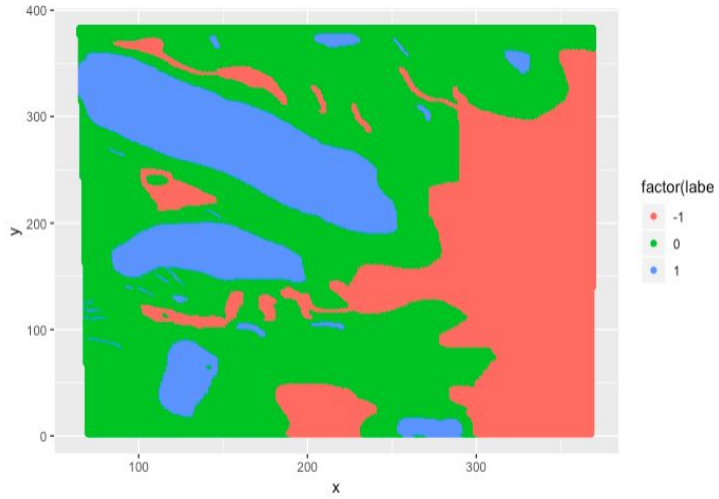
b)

	image 1 <dbl>	image 2 <dbl>	image 3 <dbl>
% of 1	17.76549	34.11172	18.43825
% of 0	38.45560	28.63522	52.26746
% of -1	43.77891	37.25306	29.29429

The following dataframe shows the proportion of the labels in percent within each image in which 1 means cloud, 0 is unknown data to classify, and -1 is the surface. Here are the images for each of the following images in the respective order of image 1, image 2, and image 3.

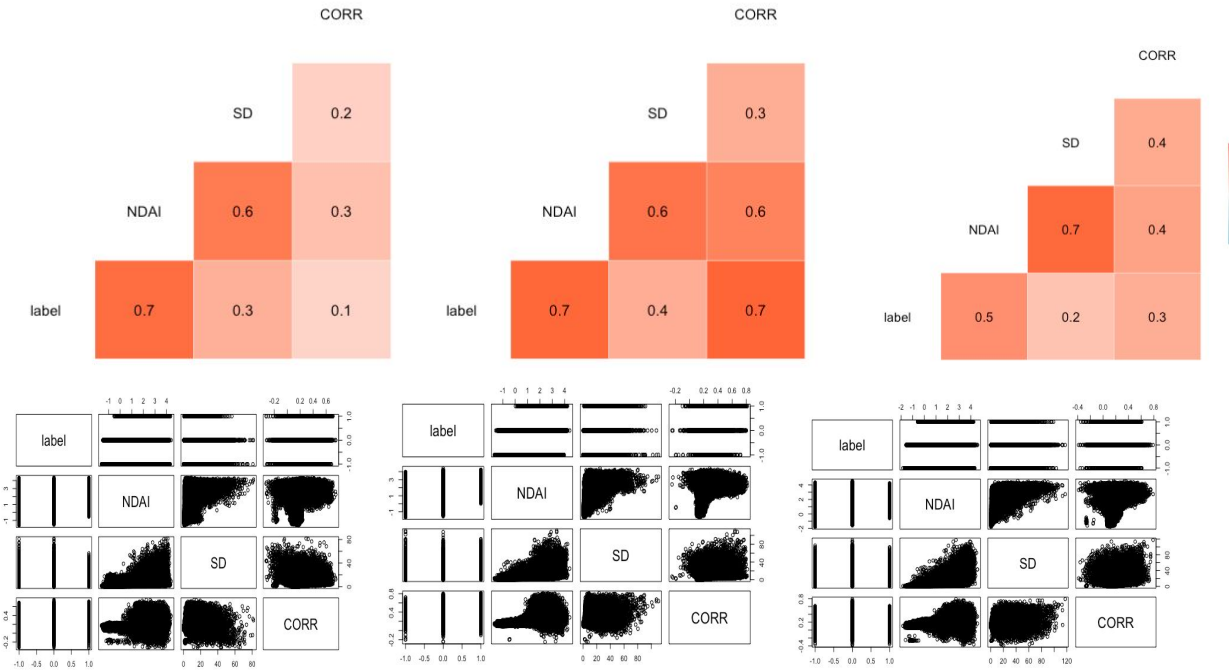






From the following trends, of the images, one noticeable commonality between the different images is that the images have are not adjacently placed. Labels of 1 and -1 all seem to cluster around its own labels or 0. There is no instance of different labels being placed adjacently to each other. This means that the image maps are spatially dependent of each other and as a result, we cannot make an assumption that the distribution of the labels are independent and identically distributed and when we classify or split data, we must do so based on specific locational groupings.

c)



From various EDA's in comparisons across different features in 3 different images, Label had high correlations to NDAI and CORR while between the features, NDAI and CORR had significant correlations across different images. Also, the distributions across various values of NDAI and CORR and SD all seemed consistent throughout the images, showing that the data is reliable based on its spread and looks based on its histograms although they are not included in the report. The values of the labels seem to span across similar ranges but, in the case of NDAI, the labels of 1 tend to be more nonnegative than the other values. Also, the values of 0 seem to be an evident combination between groups of 1 and -1 across the different distributions of features compared to labels. Other than that, it was difficult to observe other visual differences.

2. Preparation

a)

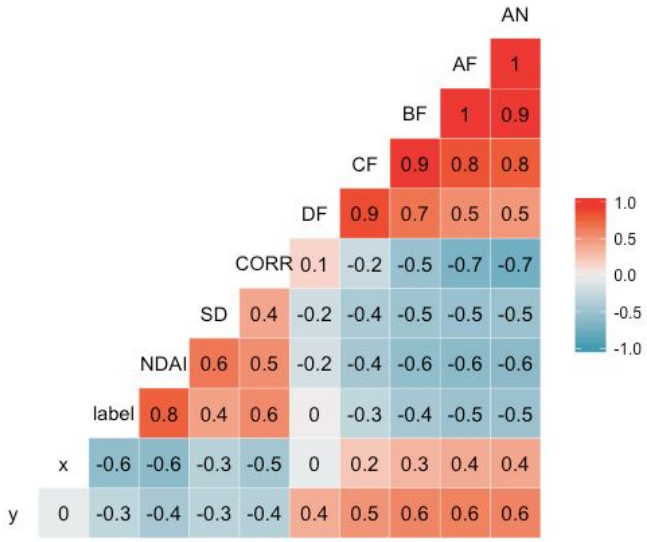
In terms of splitting data, this data cannot be split randomly because there is spatial dependence as stated above. This data depends on the strata that the data is placed in, as a result, there were two ways we thought of which would let us split data accurately. The first is to make 4 partitions of the image by X values and making 4 vertical strips. The second is to make 4 partitions of the image by Y values and therefore making 4 horizontal strips. We decided to partition the data by X values because of the Coriolis effect. Coriolis effect is the inertia motion that leads to a colon-shaped distortion in clouds and the surface and therefore we decided to partition the data by X values to correctly capture the Coriolis effect the data was first split based on the median values of x and y coordinates and then used percentile areas of x and y to split the data.

b)

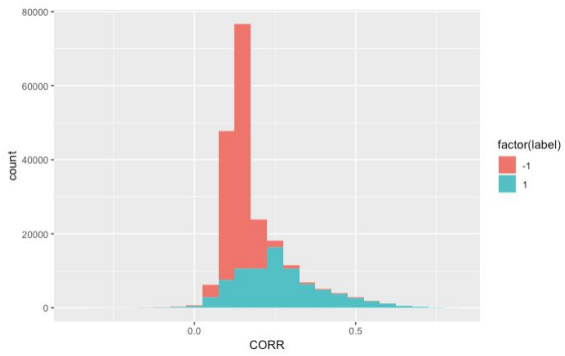
The accuracy of a trivial classifier that sets all labels to -1 on the validation set and on the test set gives an accuracy of 24.125 percent. The accuracy of the trivial classifier would be much more accurate when there is a class imbalance with a high proportion of -1 values on the data since then all of the values are -1 just as the trivial classifier states.

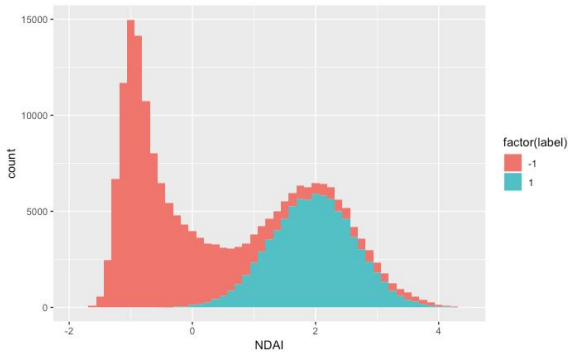
c)

We chose NDAI, AF, and AN as our three best features. To choose them, we looked at the correlations between the expert labels and provided features. Beyond the correlations, we also observed the distributions of the highly correlated features conditioned on the expert labels to see whether they were discernible.



NDAI, AF, CORR,and AN have high correlations between each other while other variables do not have as much correlation with the label and NDAI. However, from this, looking at possible distributions of the features was important to see if the data was separable.





The distribution of CORR and NDAI show that NDAI is a nice feature to select while CORR is not because the spreads of the data for CORR completely overlaps although there are different probability distributions. This makes the data difficult to identify which values are for which labels. On the other hand, the histogram for NDAI shows that the spreads are not completely caressed by the different factor values. This led to our decision of selecting NDAI, AF, and AN.

d)

```
import numpy as np
import pandas as pd
from sklearn.model_selection import KFold
```

```
def CVgeneric(features, labels, K, classifier, loss):
    errors = []
    folds = KFold(K)
    for train_idx, val_idx in folds.split(features):
        train_features, train_labels = features.iloc[train_idx,:], labels.iloc[train_idx]
        val_features, val_labels = features.iloc[val_idx,:], labels.iloc[val_idx]

        model = classifier.fit(train_features, train_labels)
        predictions = model.predict(val_features)

        errors.append(loss(predictions, val_labels))
    return errors
```

3. Modeling

For the following parts, plots on the left side correspond to the X-splitting method, and the plots on the right side correspond to the Y-splitting method. All cut-off values were chosen based on the classifier that had the largest difference between True Positive Rate and False Positive Rate, i.e. the classifier that was closest to True Positive Rate of 1 and False Positive Rate of 0.

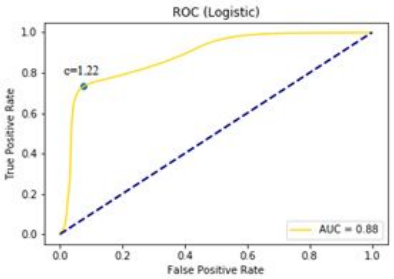
a,b)

i) Logistic

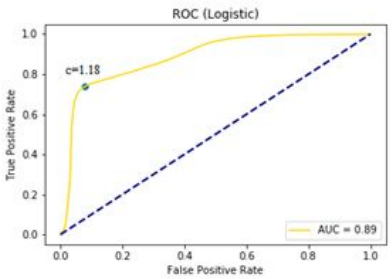
Assumptions:

- Dependent variable is binary (✔ 'label' ∈ {-1, 1})
- Observed data units are independent (□ data units are spatially dependent)
- little to no multicollinearity (✔ features are linearly independent)

Fold	Accuracy
1	0.851690
2	0.819034
3	0.815442
4	0.826339
5	0.810925



Fold	Accuracy
1	0.933691
2	0.911448
3	0.804338
4	0.724772
5	0.797280

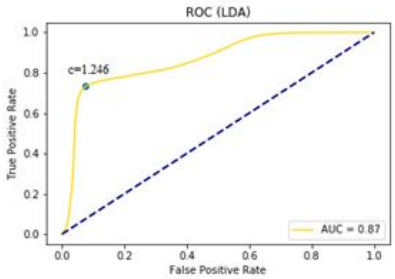


ii) LDA

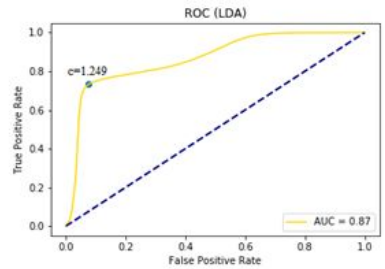
Assumptions:

- Multivariate normality in features (□ features not normally distributed)
- Observed data units are independent (□ data units are spatially dependent)
- little to no multicollinearity (✔ features are linearly independent)
- Homoscedasticity (□ class -1 and class 1 do not have equal variance)

Fold	Accuracy
1	0.845724
2	0.783862
3	0.776372
4	0.773165
5	0.768095



Fold	Accuracy
1	0.929147
2	0.888587
3	0.763780
4	0.692649
5	0.756998

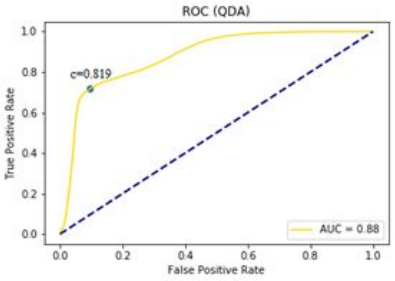


iii) QDA

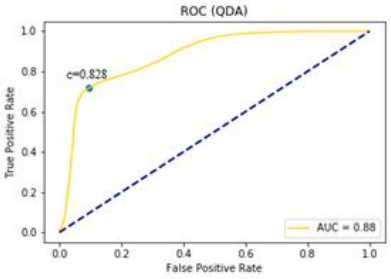
Assumptions:

- Multivariate normality in features (□ features not normally distributed)
- Observed data units are independent (□ data units are spatially dependent)
- little to no multicollinearity (✔ features are linearly independent)
- Heteroscedasticity (✔ class -1 and class 1 do not have equal variance)

Fold	Accuracy
1	0.879621
2	0.802966
3	0.723301
4	0.764371
5	0.737232



Fold	Accuracy
1	0.919645
2	0.853124
3	0.751489
4	0.677122
5	0.735411

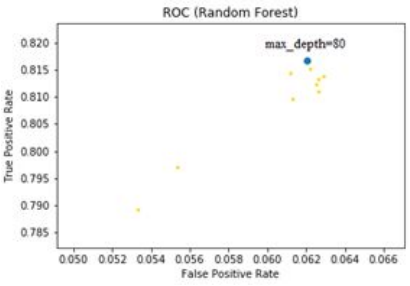


iv) Random Forest

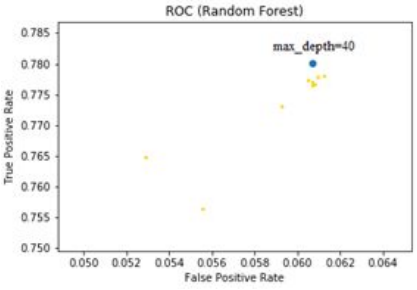
Assumptions:

- data exist in some space that can be partitioned (✔)

Fold	Accuracy
1	0.916483
2	0.878655
3	0.924446
4	0.928446
5	0.910618



Fold	Accuracy
1	0.945913
2	0.947048
3	0.902221
4	0.875676
5	0.905595

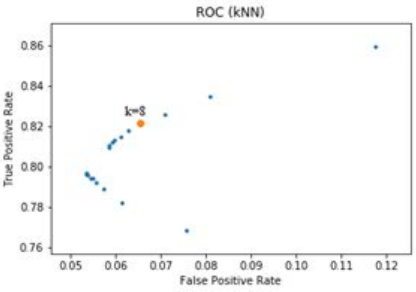


v) K-Nearest Neighbors

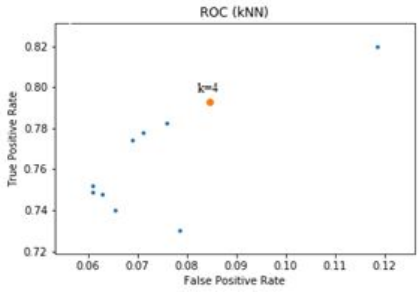
Assumptions:

- no assumption on underlying distribution of data, non-parametric
- data exist in some feature space with a measurable distance (✔ the data lies in the x-y space and we can measure distance)

Fold	Accuracy
1	0.924655
2	0.880241
3	0.927273
4	0.931929
5	0.914928



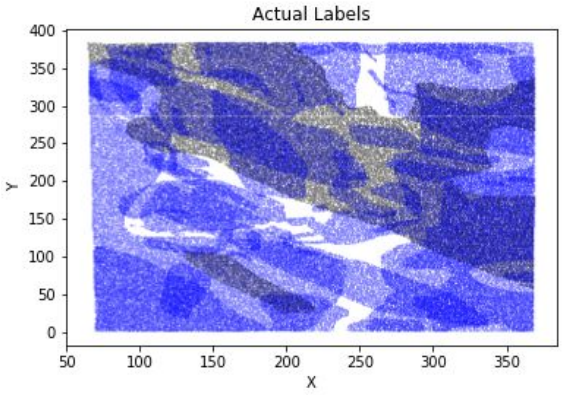
Fold	Accuracy
1	0.946877
2	0.950146
3	0.906318
4	0.879910
5	0.908797

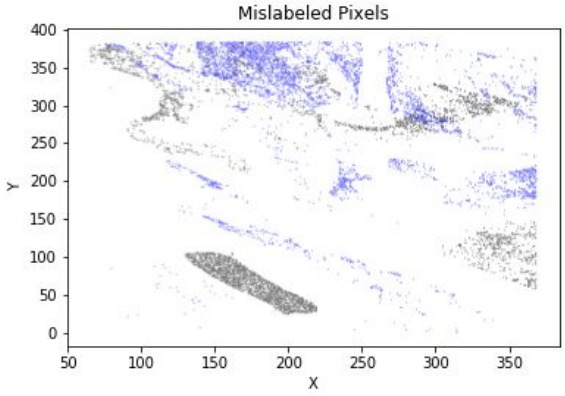


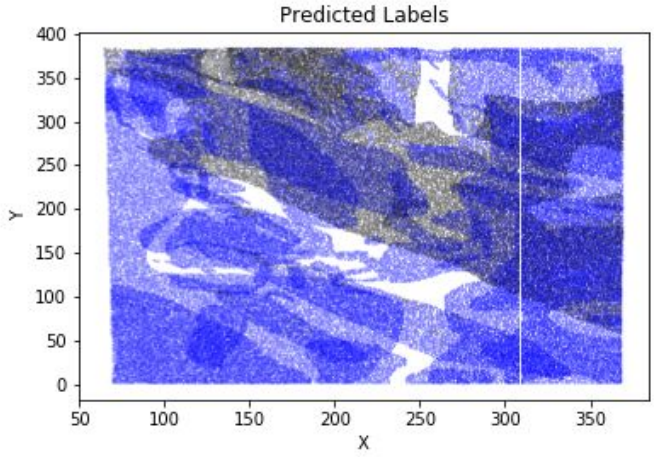
4. Diagnostics

a)

We thought that a good classification model from the above examples were K-Nearest Neighbors (KNN) because it showed great accuracy throughout the cross validations shown above. For simplicity, I will only compare the pictures for the X split portions of actual labels vs predicted labels, and where there were errors.







From the following images, the predicted, and actual labels on the plotted map almost look identical in the raw eye. Until specific locations are pointed out through the figure with mislabeled pixels, the error was hard to identify.

b)

There were issues in identifying the small cloud in the middle of the surface, and in an area with a lot of cloud, too many labels were predicted as clouds. This means that this classifier is unable to make detailed classifications while it gets the large picture. Also, this means that this classifier is aggressive in classifying values rather than conservatively asserting its values since any small

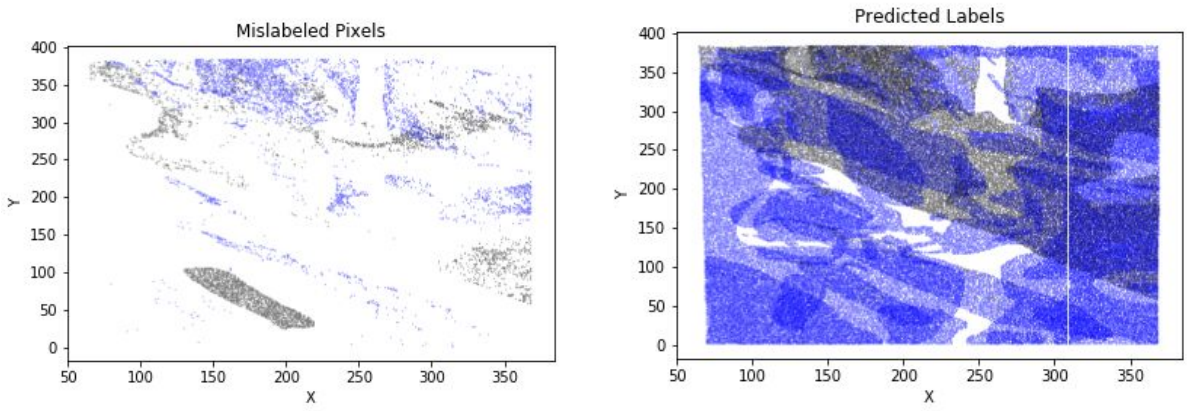
drop of cloud on a mostly surface area is automatically classified as a surface while large clouds are larger on the predicted labels than its actual observations.

c)

The KNN method still was accurate in its values. There might be better ways to improve the accuracy by choosing better features or modifying certain aspects of the features, but other than that, KNN seems like the most viable classifier among the values we have tested. The other classification models are not viable because they assume identical and independent distribution of the sample which is not true in this case because there is spatial dependence in the scatter plot of data.

d)

For another comparison, the misclassification errors will be posted for data split by y values.



Comparing these plots again with the one on the top, the mislabeled pixels seem to face similar errors as the data processed from being split by X values in terms of region even though the data processing method was different.

e)

After observing various methods, we concluded that KNN gave the highest five fold cross-validation score. KNN is a great estimator because this does not assume any sort of parametric distribution. This is a nonparametric method of classifying data while calculating the distance on a finite feature space. This comes at a cost of putting high weight on the data used. As a result, we saw commonalities in misclassification errors even though different methods of splitting data were chosen. We conclude that the accuracy of this particular scenario can be improved through selection and formulation of potentially better constructed features that are given across more spread and range of values so that the training data can incorporate much more information.

Acknowledgements:

Min Jae Shin was responsible for exploratory data analysis and writing up and making conclusions about KNN. Also responsible for the majority of the write up and formatting (Parts 1,2,4,5).

Eric Yang was responsible for selecting and implementing the classification algorithms (Parts 2,3,4,5). Set up the GitHub repository for reproduction of results.

GitHub Link: **https://github.com/ericuyang/stat154project2**