Cloud Detection from Images

Chanhaeng Lee(3033269280), Tiffy Tsay (3031903057)

1. Data Collection and Exploration

To understand the data exploration and modeling in this paper, the source of the data and context of the study must first be addressed. The data and background for this continued exploration comes from a research paper published in 2006 on Arctic cloud detection. The purpose of this original study is to propose algorithms of classifying images of both ice-covered and snow-covered surfaces in the Arctic area into cloud coverage so that this data could be used in context with climate change over time.

Data was collected by the launch of the Multiangle Imaging SpectroRadiometer (MISR) on board the NASA's Terra satellite in 1999. MISR sensors are composed of nine cameras at a different angle to an object in four spectral bands (blue, green, red, and near-infrared). The cameras cover about a 360-km-wide swath on the Earth's surface from the Arctic to Antarctica. Some of the swaths are overlapping and it takes 16 days to cover the same exact swath. Each swath is subdivided into 180 blocks and the number of the blocks increases from the Arctic to Antarctica. Although each pixel from the cameras produces 275m by 275m region on the Earth's surface, due to the size of the whole data and transmission issues, only the red radiances and all channels from the nadir camera (only one camera in the direction) at the full resolution and others at a 1.1 km by 1.1 km resolution. The specific data used in this study is from 10 MISR orbits of path 26 over the Arctic, northern Greenland, and Baffin Bay. The reason why this particular path was chosen is its geographical diversity such as snow-covered and snow-free coastal mountains, glacial snow and ice and sea ice.

Several algorithms of identifying clouds are tested and evaluated. In the comparisons of MSIR operational algorithms, enhanced linear correlation matching (ELCM) and ELCM-QDA, and offline SVM, ELCM and ELCM-QDA algorithm is found to be the best. There are two types of potential impacts brought by this study. First, the statisticians' role in dealing with huge data of Earth science is crucial and the study shows the power and contribution of statistical thinking and analysis. Second, by being able to classify clouds on the Arctic region, we can reach a better understanding of how visible and infrared radiation differ and clouds response to a climate change in the area.

The models in this paper is from three images from the satellite with the following features: the x and y coordinates, the expert labels, normalized difference angular index (NDAI), standard deviation of nadir values (SD), correlation of MISR images from different directions (CORR), and 5 radiance angle measures (DF, CF, BF, AF, and AN). In Figure 1, we can see that there are some patterns in where the cloud regions are located in the three images. The maps that we have created using the experts show where clouds are located (in orange), no clouds are located (in blue), and the unknown area (in yellow).Typically, clouds are close to each other, so if a neighboring pixel is marked as cloudy then it's likely that this pixel is labeled as cloudy too. The unlabeled regions are ones that are between clouds or between a cloud and uncloudy, very few (if any) are within a large cloudy region. Overall, image 1 has roughly 17% cloudy, 38% unlabeled, and 43% uncloudy pixels, image 2 has 34% cloudy, 28% unlabeled, and

37% uncloudy, while image 3 has 18% cloudy, 52% unlabeled, and 30% uncloudy pixels. Between image 2 and 3, most of the clouds seem to be located on the left half of the image, while image 1 seems to have it more evenly distributed. All 3 images seem to show that clouds tend/can exist at any y coordinate. Nevertheless, based on these maps we can see that an independent and identically distributed assumption is not reasonable since the existence of clouds look to be location-based or regionally clustered.

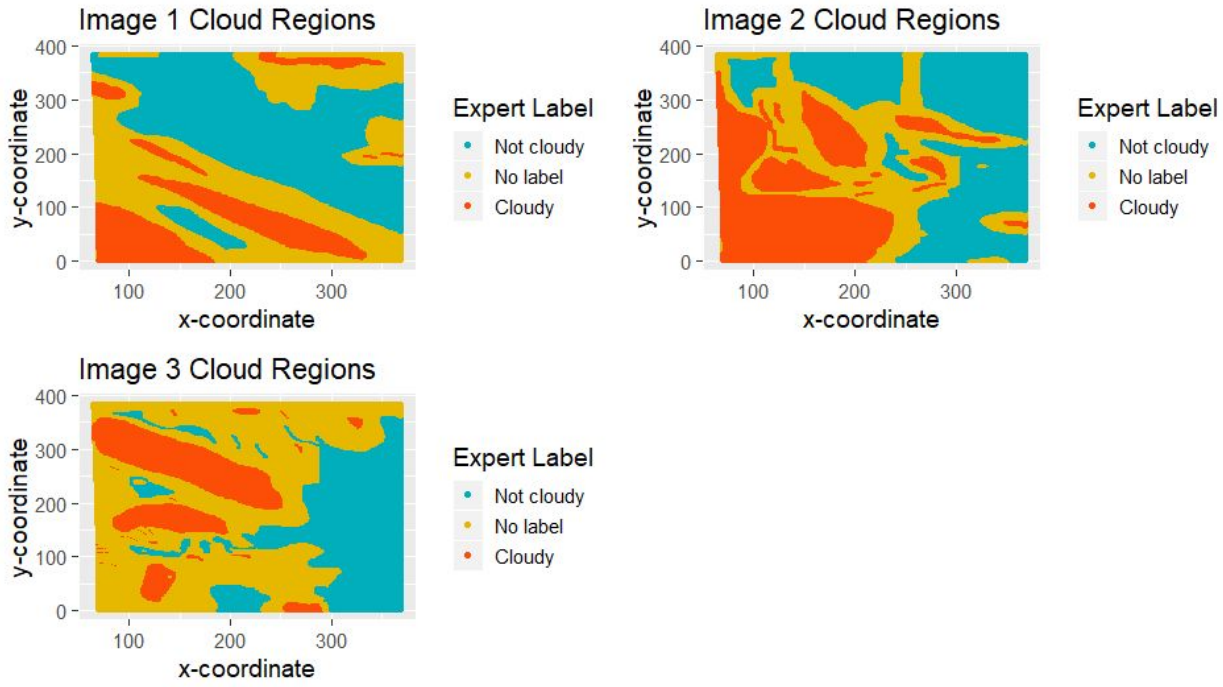

Figure 1 (Cloud labels)

Through more thorough EDA we can explore the relationships between the features as well as those with the expert labels. The ten features (since we are ignoring expert labels) have relationships between each other. Figure 2 can show us at a quick glance just exactly what those pairwise relationships look like in Image 2, but these relationships carry over to all three images. There are negative relationships between: CORR and radiance angle AN/AF. Meanwhile, there are positive relationships between all radiance angle measurements. The strongest association is between radiance angle AF and AN, BF and AF. When we aggregated the all three images our quantitative analysis confirmed what we inspected visually for image 2. There are correlations of: -0.60 between CORR and radiance angle AF, -0.68 between CORR and radiance angle AN. The lowest correlation among the radiance angles was between radiance angle AN and radiance angle DF which was only 0.54 while the highest correlation was between radiance angle AN and radiance angle AF which was 0.97.

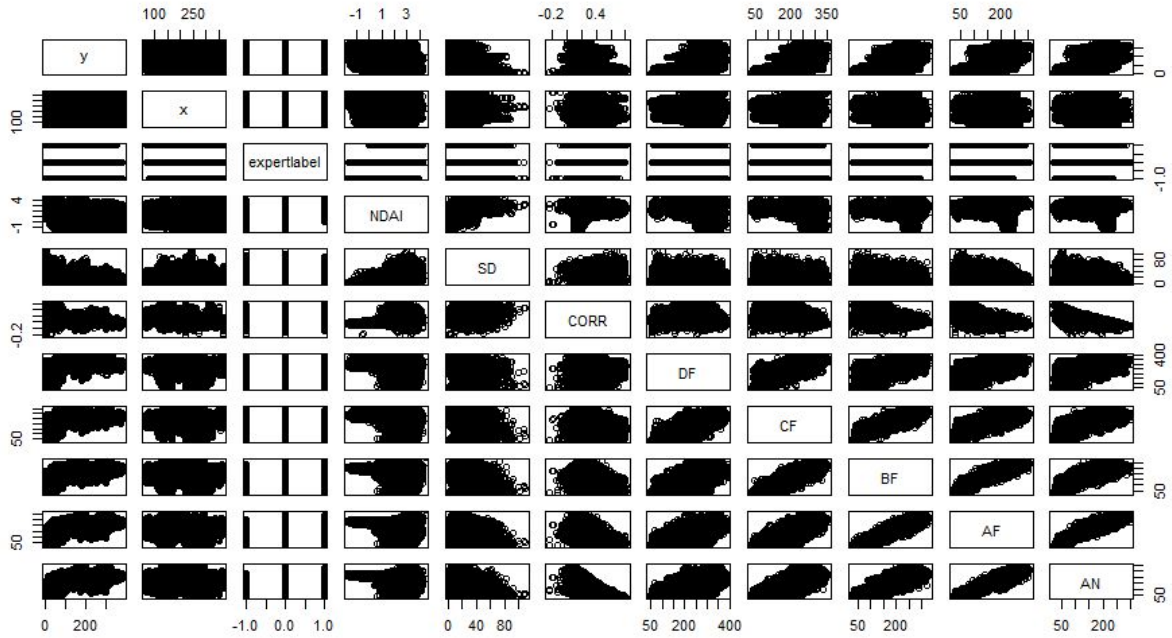

Figure 2 (Pairwise scatterplot for Image 2)

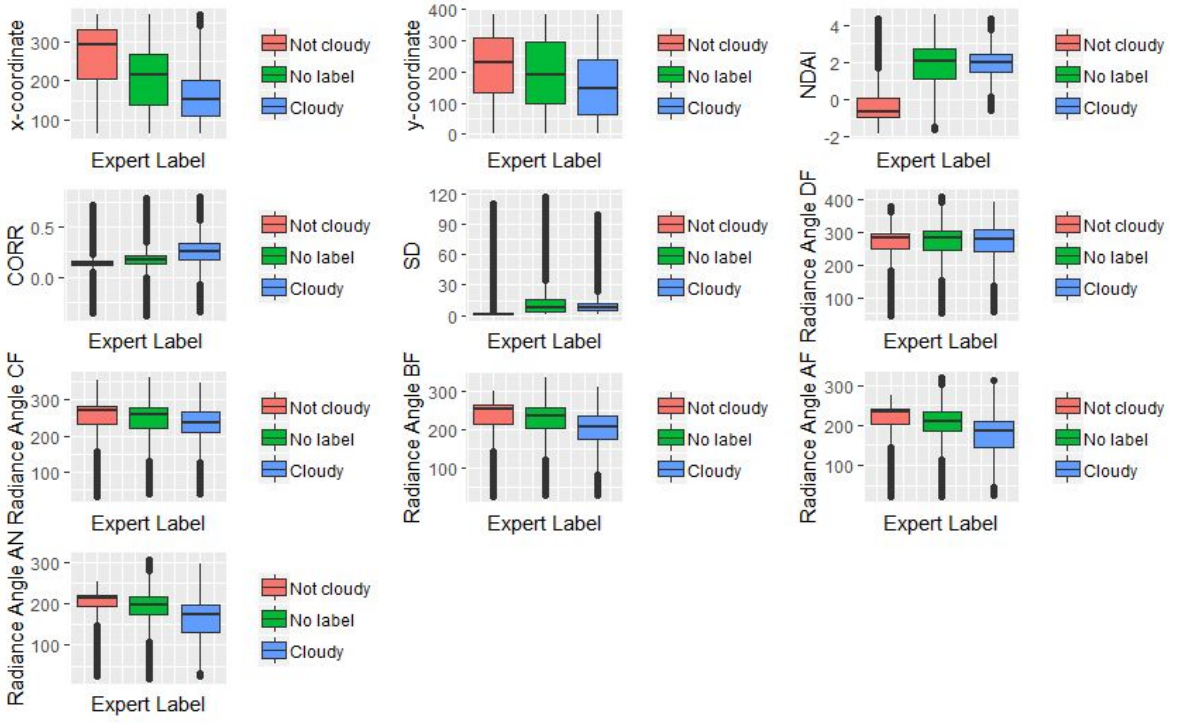

Figure 3 (Feature Distributions)

On the aggregated data, there is clearly some features that are differently distributed based on the cloud's label. Figure 3 shows us that there are very clear differences in where the data is distributed for the x-coordinate, NDAI, CORR, SD, and radiance angles BF, AF, and AN.

2. Preparation

To prepare to train our model the aggregated dataset needs to be split into three portions, training, validation and test sets. The current problem with just naively randomly splitting is that the labels are not independent of each other, if all of a pixel's neighbors are cloudy or uncloudy then it becomes much more likely that it is of the same type. This is also expected of future data which will not have any expert labels on them and only the other features will be used to predict. Due to the importance of locationality of where the clouds are, splits are generated for three images and then combined afterwards instead of splitting the aggregated data. By doing this, we can preserve more of the innate location information encoded within each image. Both of the next two methods try and account for the non-independence between pixels in different ways.

Our blocking method splits each image into different square regions using the x-coordinate and y-coordinate. A reasonable section from visual inspection indicates that a 50x50 region should ensure that there is not too much mixing of different labeled clouds. Most of the time if half of that region's pixels were selected then the majority vote of those would be able to label the remaining the half's with high accuracy. Using this logic, from each region of the image 80% would be randomly taken into the train and validation split and the remaining 20% would be taken into the test split. This way even though pixels were randomly selected, they were selected from a specific region which ensures that roughly 200 pixels selected for training from that region should be representative of the 50 pixels that are put into the test data.

Our second method is based around the x-axis and splits the image into rows. Each row would then have 80% randomly taken into the train and validation split and the remaining 20% into the test split. From our previous visual exploratory data analysis 2 out of the 3 images seemed to have a tendency of having clouds on the left-half of the image, but it's also possible that this trend does not continue for our future data. In order to avoid overfitting, the row split allows us still to preserve locationality in the dimension (x instead of y-coordinates) that shows the most clear difference in label distribution but also allows for each row to have the flexibility to select different pixels.

Our baseline accuracy would be to just predict a trivial classifier which labels all pixels in an image as uncloudy. Using both our blocking and x-axis method we achieve roughly a 60-61% accuracy on both the test and validation set. The only time our trivial classifier would have high average accuracy is that it's typical to be no clouds present in our images, which means that we already know what the cloud coverage looks like.

Using the expert labels as the truth, modeling relies on the good features to achieve the best results. In order to narrow down our features to three of the best ones, we judged the features based on how accurately each individual feature by itself would be able to predict the label. Intuitively, these would features that are distributed the most differently between non-cloudy and cloudy labels. During initial exploration, the boxplots help to find ones that have different areas for the different labels. NDAI does this really well, while the x-coordinate, CORR, radiance angles AF and AN seem to be promising. Our goal is to now narrow this down to 3 of the best. The visual density graph in Figure 4 help us visualize just how much the two labels overlap for a single feature. NDAI looks the best still, followed by CORR and SD, and then

radiance angle AF/AN. This overlap can also be quantified through integration to help us decide exactly which three features are the best. NDAI has roughly 11% of overlap, , SD has 18%, CORR has 20%, x has 31%, AN has 31%, and AF has 32%. One last consideration we want is that we do not necessarily want to take highly correlated features as the only features because this means we are essentially losing potential information. While these feature might have some association we can see that none of them have correlation above 0.63. Therefore, we have selected the 3 best features as: NDAI, CORR, SD.

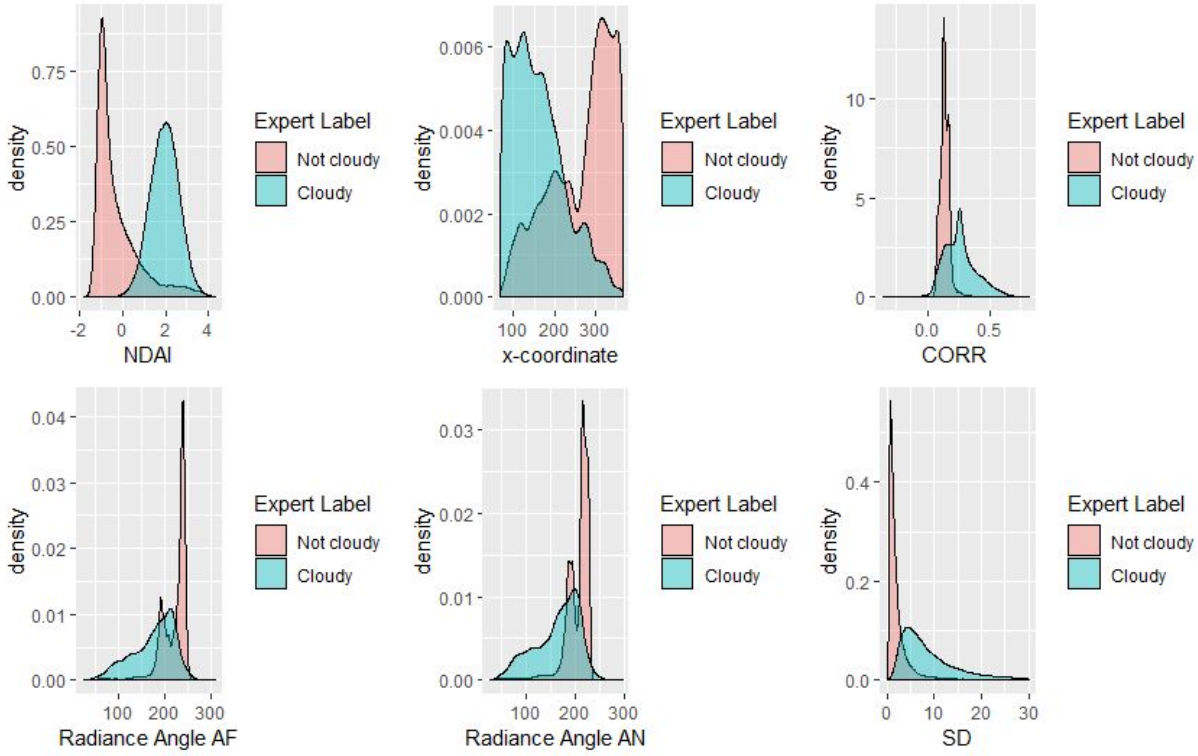

Figure 4 (Feature Expert Label Overlap)

3. Modeling

A total of four methods (logistic regression, decision trees, QDA and LDA) was used to decide which model would produce the best results. The main criteria that these methods were compared across each other was accuracy. Our goal is to be able to predict correctly whether or not a pixel is cloudy or uncloudy so accuracy is a good metric. Additionally, false positives are no worse than a false negative since neither class is more important than the either.

Logistic regression model has a few assumptions that need to be satisfied in order for the method to work properly. We assume that a pixel is either cloudy or uncloudy, and it can't be both, which is satisfied. Another assumption is that the features should be roughly linear with the logit of the outcome, which is not the case for some features such as the y-coordinate. The last assumption is that there is not multicollinearity, but using both the variable inflation factor as well as the original pairs plot, we can see that the radiance angles are highly correlated. The hyperparameter for the link function was set/default to binomial to produce the logits. The blocking method for all 10 folds of our cross validation achieves accuracies of 89% (89.7, 89.4,

89.5, 89.3, 89.5, 89.7, 89.4, 89.0, 89.7, 89.4), for an overall accuracy of 89.41% and an overall accuracy of 89.37%. The overall test accuracy from the best logistic model (the one with the highest accuracy) achieved an accuracy of 89.3%. The x-axis method for all 10 folds achieves accuracies of 89% (89.5%, 89.6%, 89.4%, 89.4%, 89.4%, 89.3%, 89.4%, 89.5%, 89.4%, 89.2%) for an overall accuracy of 89.46% on the train and 89.43% on the test set. When we check the coefficients for both of these logistic models as a metric for feature importance, the top 3 features would be NDAI, Radiance angle AN, and CORR (in that order).

The decision tree model also has a few assumptions that need to be satisfied. Once again, there is an assumption that there is not multicollinearity, otherwise the decision tree only greedily chooses the best feature. The hyperparameter of complexity was tuned through cross-validation of within the caret package, with the highest accuracy being achieved when the model complexity was the lowest (0.01~), this helps to also ensure that we avoid overfitting. The blocking method resulted in fold accuracies of 90-92% (90.8%, 91.0%, 91.8%, 90.6%, 92.0%, 90.1%, 90.6%, 91.5%, 90.99%, 91.5%), for an overall accuracy of 91% and a test accuracy of 92.3%. The x-axis method for all 10 folds achieves accuracies of 90-92% (92.2%, 90.9%, 91.2%, 91.2%, 91.6%, 91.3%, 92.1%, 91.5%, 90.7%, 90.7%) for an overall accuracy of 91.3% on the train and 90.6% on the test set. When we check the decision tree using the first 3 splits as cutoffs for feature importance, the top 3 features are NDAI, CORR, x/y (in that order).

In order for us to use LDA model, we assumed that each observation in binary classes are normally distributed and multicollinearity and also that the covariance of two classes is identical. The blocking method resulted in fold accuracies of 89% (89.85%, 89.97%, 89.81%, 89.85%, 89.84%, 89.82%, 89.63%, 89.99%, 89.75%, 89.74%), for an overall accuracy of 89.8% and a test accuracy of 89.79%. The x-axis method for all 10 folds achieves accuracies of 89-90% (89.77%, 90.00%, 90.35%, 89.61%, 89.61%, 89.40%, 89.87%, 89.94%, 89.70%, 90.31%) for an overall accuracy of 89.8% on the train and 89.83% on the test set. When checking coefficients of linear discriminants, we can find that the top 3 important features are NDAI, SD, and x (in that order). The average of LDA accuracies across folds depending on features are compared. We can compare 5 different combination of features; the first is all features. (1st: all features, 2nd: ‘x’, ‘y’, ‘NDAI’, ‘SD’, ‘CORR’, 3rd: ‘x’, ‘y’, 4th: ‘NDAI’ ,’SD’, ‘CORR’, and 5th: top 3(NDAI, SD, x). For LDA with blocked data, the model with the top 3 features give the highest average of accuracies across folds, 90.22% compared to that with all feature considered, 89.82% and that with the 2nd, 90.05%. We find a slight improvement in using top 3 in LDA method.

For the QDA model, we obviously do not assume that the covariances of each class is the same. The blocking method resulted in fold accuracies of 90% (90.30%, 90.56%, 90.37%, 90.43%, 90.23%, 90.43%, 90.50%, 90.51%, 90.87% and 90.55%) for an overall accuracy of 89.4% and the test accuracy of 90.54%. The x-axis method resulted in fold accuracies of 90% (90.40%, 90.72%, 90.58%, 90.54%, 90.89%, 90.05%, 90.19%, 90.76%, 90.31%, 90.95%) for an overall accuracy of 90.5% and test accuracy of 90.50%. The top 3 important features are NDAI, SD, and x (in that order). For QDA, we can also compare how different combinations of features affect accuracy. This time, when all features considered, the mean accuracies across folds is 90.47%, which is the highest when compared the other three; 89.67%, 82.86%, 89.62%, and 89.36%.

ROC curves are a good way to compare the different methods since it compares the true positive rate with the false positive rate. The cutoff values were selected based on the ROC curve, our goal is to choose ones that ensure that we have the lowest false positive rate and highest true positive rate which is indicated by the steepest point. Figures 5 and show the different ROC curves for all methods. The method with the best AUC was decision tree with blocked data, 0.916. For logistic regression, the cutoff was 0.87 sensitivity and 0.91 specificity. The decision tree the cutoff was 0.97 sensitivity and 0.85 specificity. For QDA, the cutoff was 85.16 sensitivity and 94.05 specificity. For LDA, the cutoff was 89.12 sensitivity and 90.34 specificity.

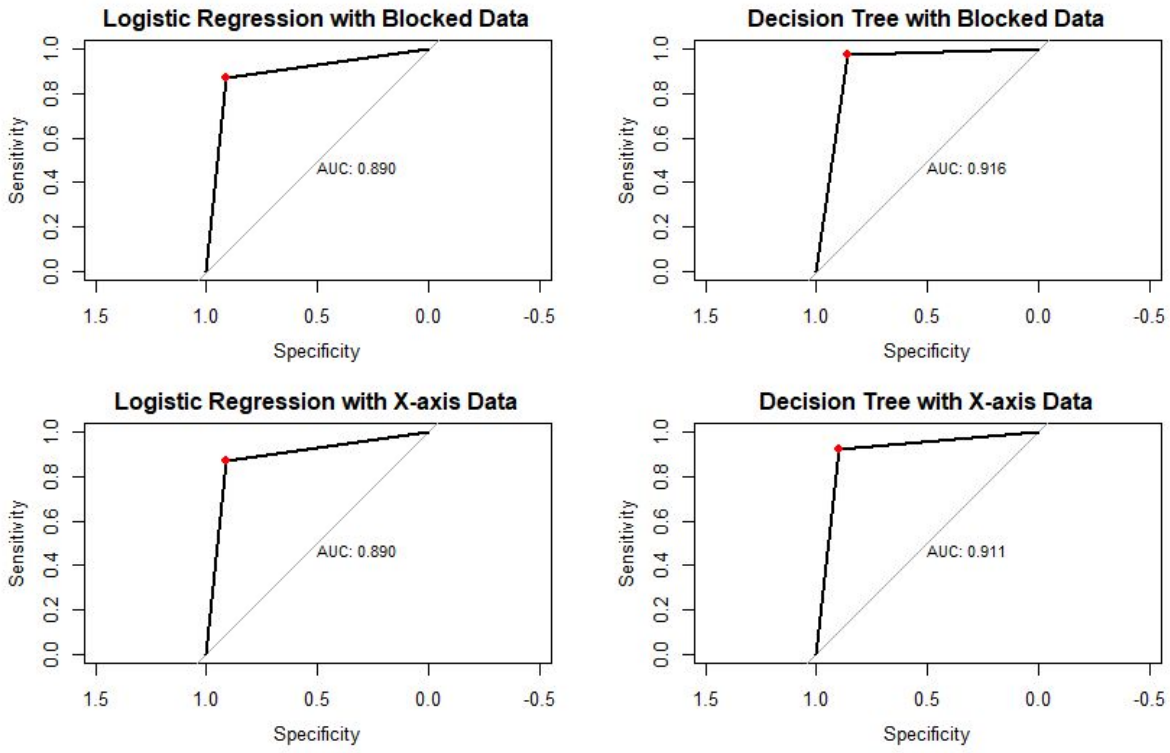

Figure 5 (Logistic Regression and Decision Tree ROC Curve)

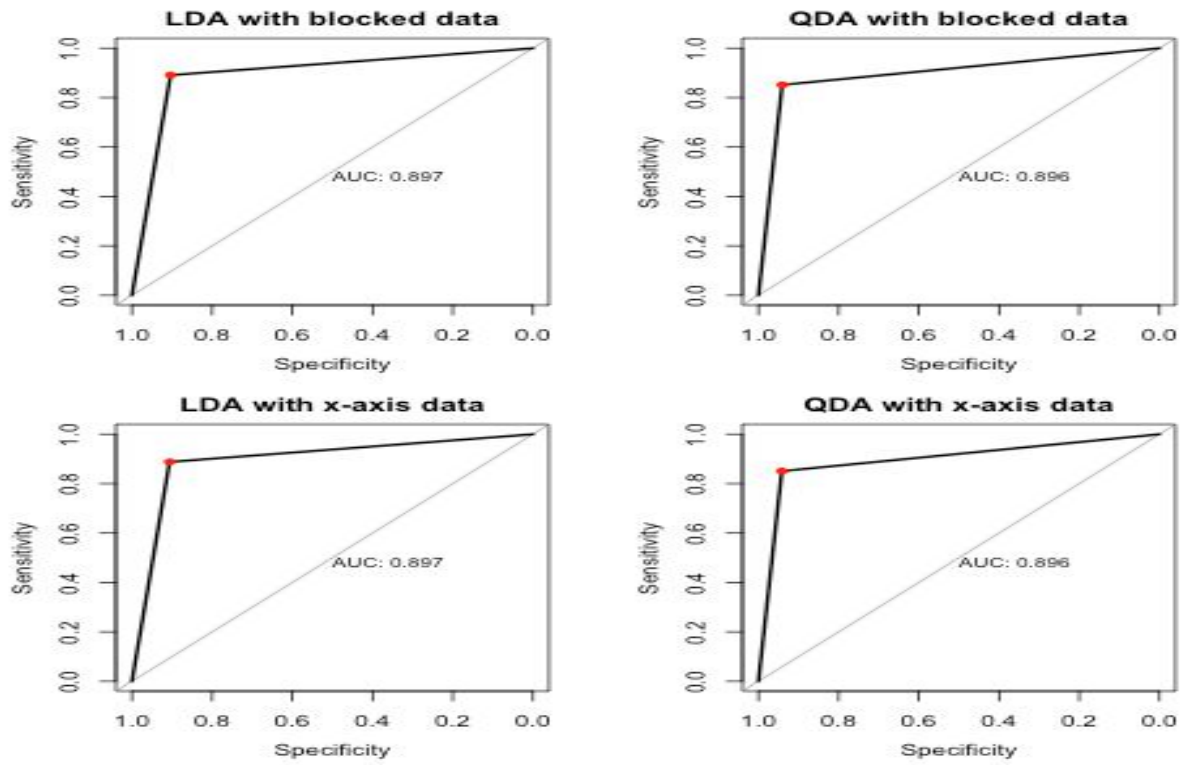

Figure 6 (LDA and QDA ROC curve)

Besides accuracy, other metrics can help us understand where and how are models are performing poorly or well. The different methods are now compared with respect to positive predictive value (PPV) and negative predictive value (NPV). PPV is how many predictions on positive class are right and vice versa for NPV. Since neither prediction is more important to us, we hope that the PPV and NPV rates are relatively the same. We found that the decision tree with x-axis gives the highest PPV and LDA with blocking gives the highest NPV value.

Method	PPV	NPV
LDA (Blocking)	85.44%	92.74%
LDA (x-axis)	85.69%	92.61%
QDA (Blocking)	90.21%	90.73%
QDA (x-axis)	90.19%	90.68%
Logistic Regression (Blocking)	86.73%	91.05%
Logistic Regression (x-axis)	87.03%	90.95%
Decision Tree (Blocking)	93.07%	91.84%
Decision Tree (x-axis)	86.8%	91.0%

4. **Diagnostics**

4 types of classification methods, LDA, QDA, logistic regression, and decision tree, were used to find the best classification method. Since we used two different ways of sectioning data that utilize cross-validation in the process, bootstrap resampling is used for each fold. Also, every features is pre-processed of re-scaling and centering. The best model that seemed to balance both PPV, NPV, and accuracy was the decision tree with blocking.

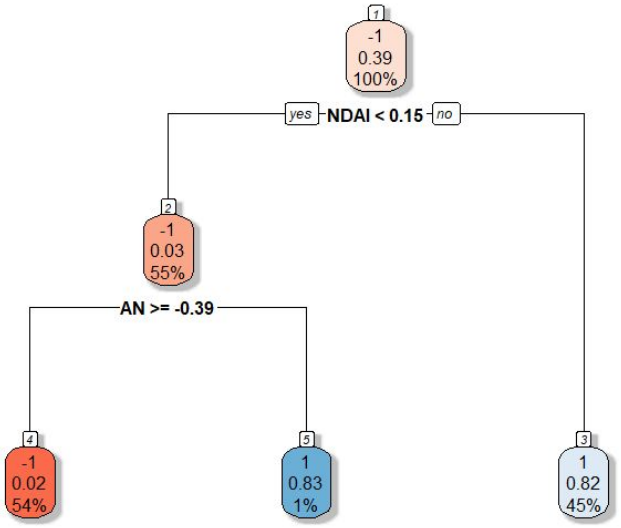

Figure 7 shows the exact decisions that the decision tree uses. One of the advantages to decision trees is that it is easy to interpret and can be confirmed by expertise knowledge.

Figure 7 (Decision Tree splits)

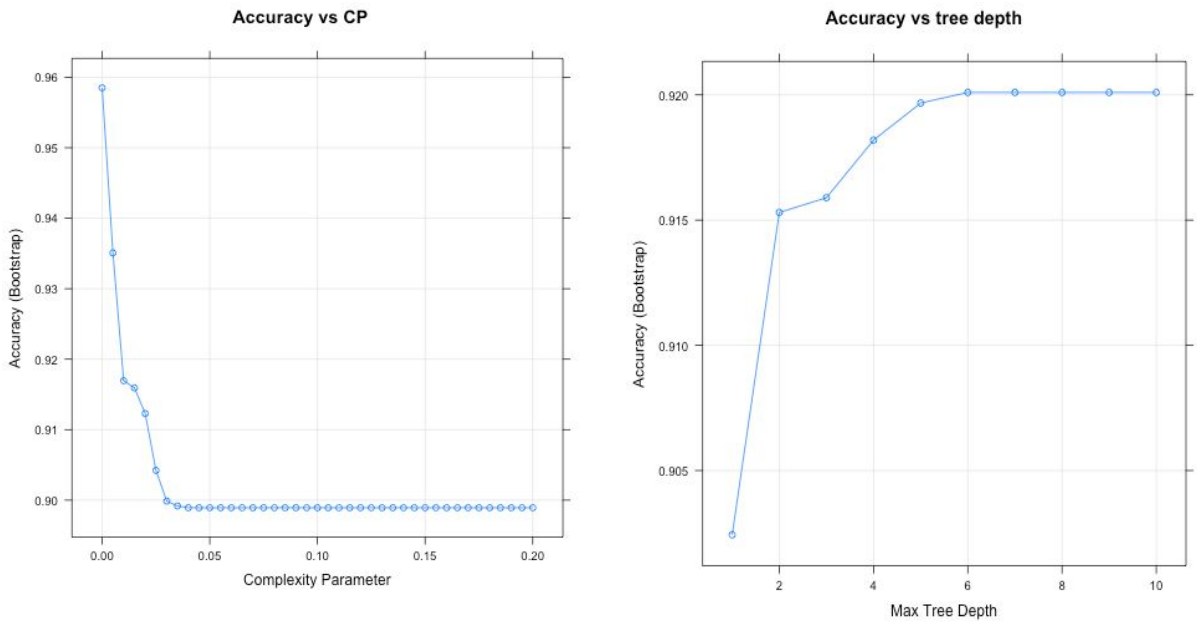

Figure 8 (Decision tree hyperparameter)

Figure 8 explores the convergence of different hyperparameters and the accuracy that it achieves. We can see that with a very shallow tree between a stump and a depth of 2 there is the largest increase in accuracy. Continuing to increase the depth gives us diminishing returns and there is almost no difference between using 5 or more features. This is reflected in how we achieve the highest accuracy when the complexity parameter (which controls depth, minimum leaf nodes, etc.) is the lowest and that increasing the complexity parameter leads to lower accuracy. Different features might be selected at different levels depending on the depth of the tree (since decision trees greedily select features), but NDAI seems to be the most indicative for our purpose. Even at different cutoffs NDAI is almost always the first feature required. One of the benefits of this blocking method is we can see that the best model selected does not actually use the x-coordinate or y-coordinate since we do not expect clouds to be located in the same region over time.

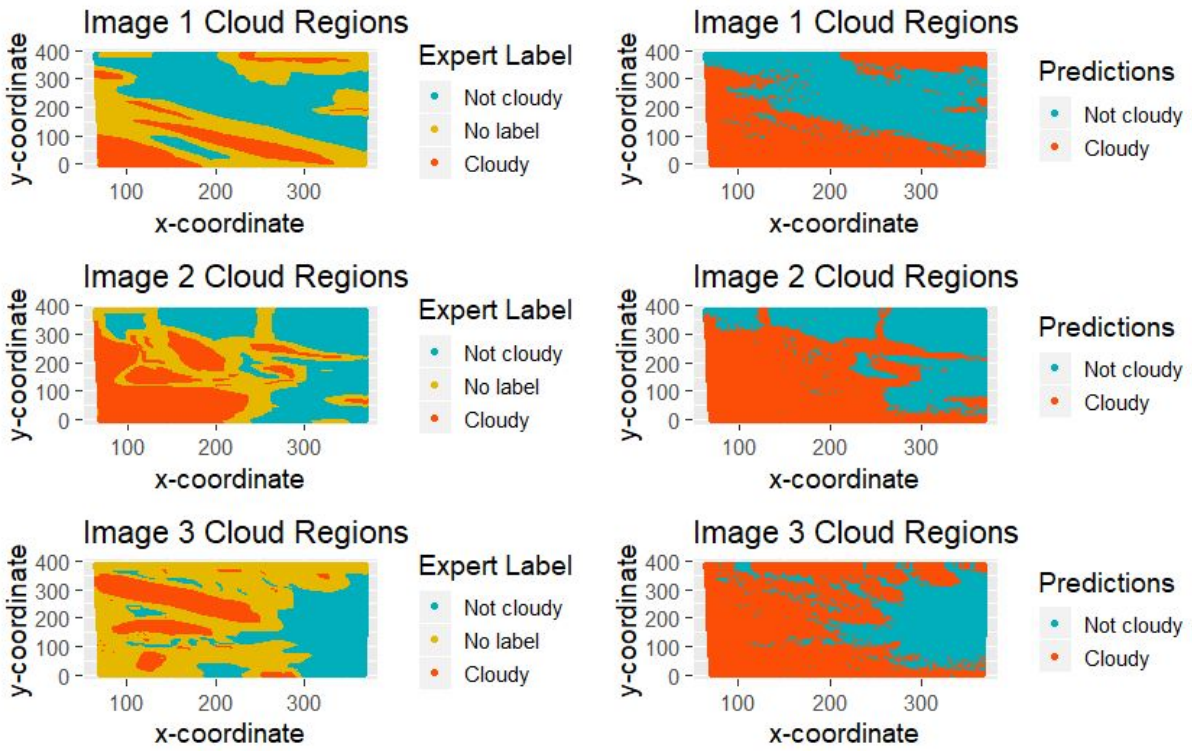

Figure 9 (Decision Tree Blocking Method Predictions)

Figure 9 shows us side by side what our model is predicting relative to the original images. The figure shows us that we tend to miss the uncloudy gaps that a cloud region might have. Both the small gap on the bottom of image 1 and the gaps on the left half of image 3 are missed. However we are classifying clouds in the bottom right corner of the images when they might not occur, this is likely due to image 1's contribution. We also tend to misclassify more clouds on the left half of the image than the right. Between all 3 images we misclassified more pixels as cloudy than uncloudy (6244 cloudy classified as uncloudy and 9703 uncloudy classified as cloudy). Another problem that this method has is that we are having small pixel breaks within regions, that our clouds and uncloudy regions are not well formed since in the center areas there will be some pixels that are mislabeled.

Since we are only limited to 3 training images, it's possible that our model will not work well with future data. Decision trees have a tendency to overfit to the training data, and given the small sample size we aren't sure if the trends that are present here will carry over to other future data just yet. The most important labels that would help us test the strength of our classifier are mostly unlabeled, so we would expect the decision tree to have a harder time predicting correctly at these unlabeled value areas (see the original boxplots of distributions). To fix this problem in specific and continue using a decision tree is to increase training dataset, have images completely labeled. Another thing to note is that we violated the multicollinearity assumption. Another potential is to use random forests, which would help us avoid the multicollinearity problem, avoid overfitting to our small sample, and reduce variance. The only concern with random forests would be the amount of time and resources needed to generate it.

We see not too much difference between the way of splitting for decision trees since both the features selected are the same with slightly different cutoffs and the misprediction rate and areas are the same. The accuracy is also similar, and but we do see that the PPV and NPV values are different between the two. The blocking method generates more stable PPV and NPV values (closer to each other) than the x-axis method. Figure 10 shows the predictions for the x-axis method for decision trees. Overall it looks very similar, but one noticeable difference is that this one successfully predicts uncloudy in the bottom right corner. For the x-axis splitting method we see that the top 3 feature importance is still the same, if we force it to split the same amount. However, splitting on the x-axis creates more complex trees to generate the same degree of accuracy, the complexity factor was roughly eight times greater than the blocking-method and naturally uses 7 features instead of 2. Since our training set was so small it's possible that the x-axis method could outperform the blocking method, but additional testing would be needed to make that conclusion. However, overall both methods can achieve the same performance but may require different levels of tuning/depths in the tree.

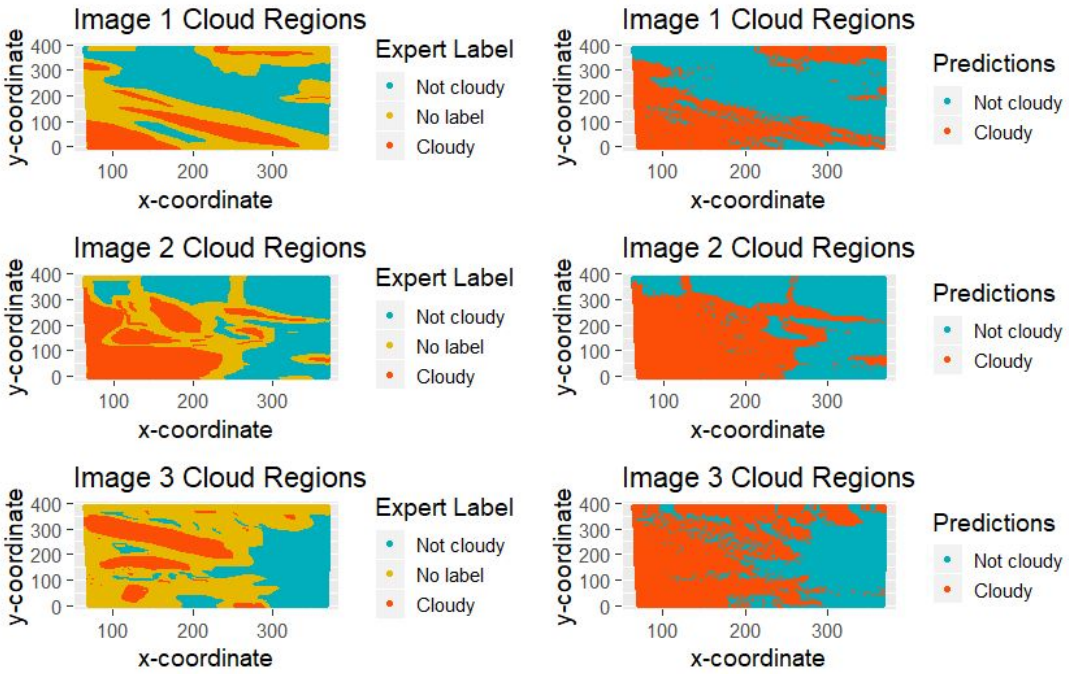

Figure 10 (Decision Tree X-axis Method Predictions)

Being able to detect clouds from surfaces with similar characteristics such as ice- and snow-covered surfaces in the Arctic region is crucial when it comes to broadening our understanding of climate changes with clouds over the area. Using data collected from three images took by MISR cameras on Terra satellite, we develop classification methods of identifying clouds. Through our data exploratory analysis, we know how features are associated with expert label, which is an indicator of cloud class. We come up with two unique ways of sectioning data that preserve much of the variability to use cross-validation method. One is to make 50x50 grids for each image(blocking method). The other method is just section image pixels by x-coordinates(x-axis method). They are applied when making cross-validation sets and are compared to each other while they are implemented for different classifying methods; logistic regression, decision trees, linear discriminants analysis, and quadratic discriminants analysis. All of the four methods have more than 89% of accuracies across folds and test set. QDA and decision trees work better than the other two when comparing not only accuracies but also PPV and NPV values. Changes occured by selection between blocking and x-axis methods, are not over 1% when calculated naively in accuracies, however, the blocking method give overall better results, which makes a sense because it clouds can be shaped in any direction.

5. Acknowledgements

Most of the work in this paper was based on the study, Daytime Arctic Cloud Detection Based on Multi-Angle Satellite Data With Case Studies. The dataset was also provided from them as well as insight into how these features were generated and background knowledge about the environment. Chanhaeng focused on understanding and summarizing the study and working on the QDA and LDA methods, while Tiffany focused on exploratory data analysis and using decision trees and logistic regression. All code and information can be found in this repository: https://github.com/tiffytsay/Stat154Project2.