Cloud Finder

Hongze Liu, 3032170077

Data Collection and Exploration:

Purpose of Study:

Clouds play an important role in modulating the sensitivity of the Arctic to increasing surface air temperatures, and MISR sensors can effectively detect accurate characterization of clouds.

Data Collection Method:

MISR comprises nine cameras, with each camera viewing Earth scene in different angles. MISR collects data from all path on a repeat for every 16 days. Each path is subdivided into 180 blocks, with blocks number increasing from North Pole to South Pole, and each pixel covers 275m*275m. The purpose of study, is therefore to build an operational algorithm that efficiently process large data collected by MISR, named ELCR algorithm. ELCR focuses on three physical features: the linear correlation of radiation measurements from different MISR view directions(CORR), the standard deviation of MISR nadir red radiation measurements within a small region (*SD*An), and a normalized difference angular index (NDAI)—contain sufficient information to separate clouds from ice- and snow-covered surfaces. ELCR algorithm composed of three parts: construct three features based on EDA and domain knowledge, build an ELCM cloud detection algorithm by setting threshold on each feature and apply ELCM to each data to produce first cloud detection product, and predict probability of cloudiness, the second cloud detection product.

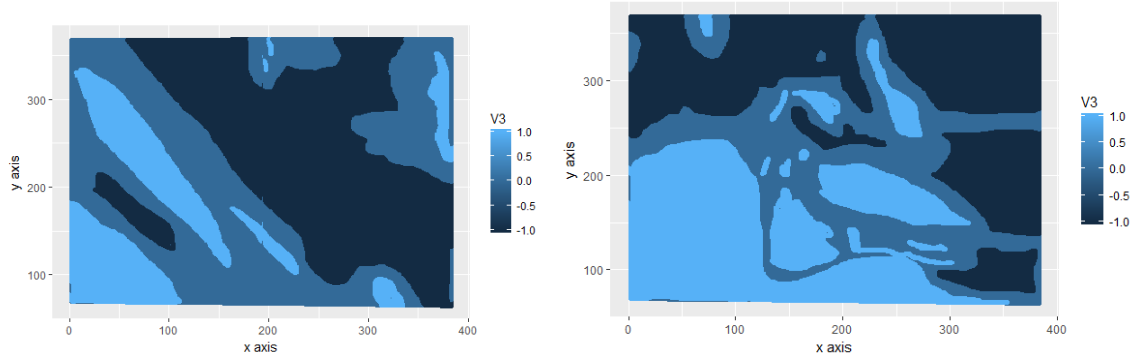
Impact of ELCM:

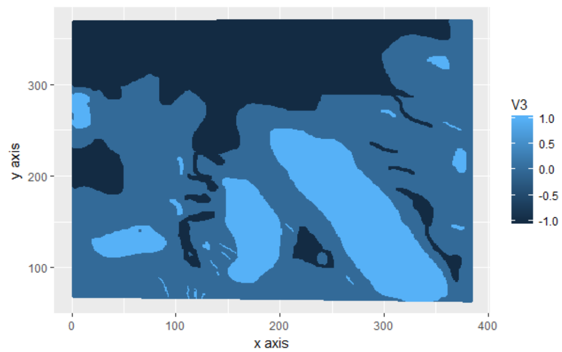
ELCM algorithm has the highest rate of receiving agreement by experts(91.8%), while sustaining 100% coverage rate. It is more accurate and provide better spatial coverage.

Conclusion:

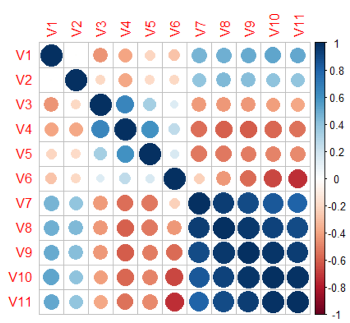
The effectiveness of the algorithm demonstrates the importance of statistician in analyzing Earth Science data, and demonstrates power of statistical thinking.

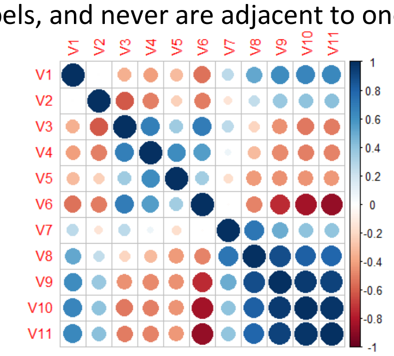
Image 1 has 17.76% with label of cloud, 38.45% with no label and 43.77% with no cloud. Image 2 has 37.25% with label of cloud, 28.63% with no label and 34.11% with no cloud. Image 3 has 18.44% with label of cloud, 52.26% with no label and 29.29% with no cloud.

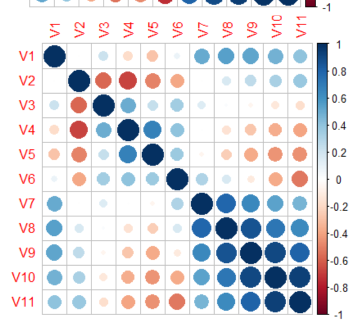




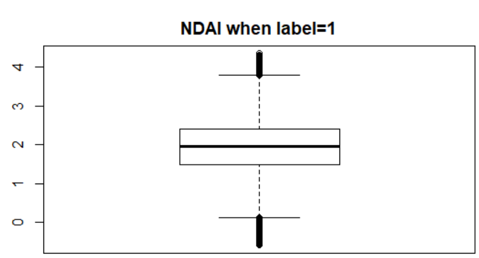
Above are respective images with labels by experts, with has cloud being light, no label in between and no cloud being dark. From three images, clearly same labels are aggregating together, with clouds in clusters, no labels in clusters and no clouds in clusters. In addition, label with cloud and no cloud are separated by no labels, and never are adjacent to one another.

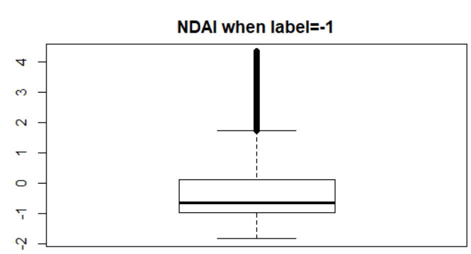


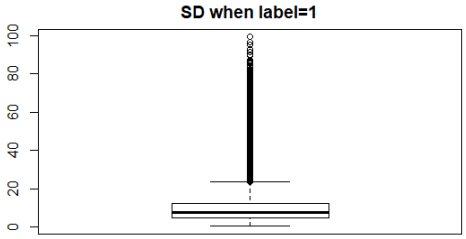


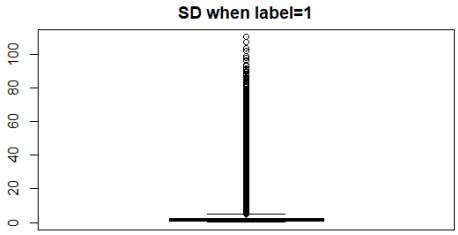


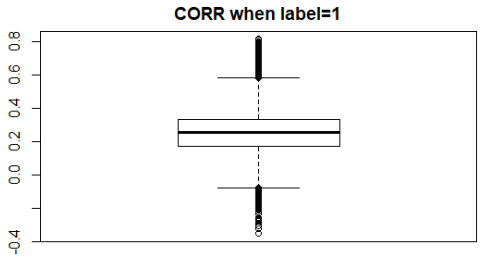
Expert labels are strongly correlated with NDAI, less correlated with SD and nearly not correlated with CORR for image 1 and image 2, but has some correlation with CORR in image 3. NDAI is strongly correlated with SD for image 1 and image 2, and less correlated with SD in image 3. NDAI is not correlated with CORR in all three images. SD is not correlated with CORR in image 1 and image 3, but has a little correlation with CORR in image 2.

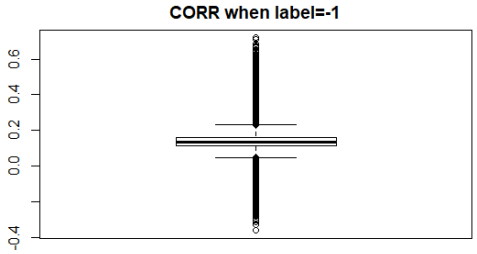












From the box plot, cloud will have higher NDAI comparing to no cloud. In addition, cloud will have higher SD then no cloud. CORR for both data appear to have similar value, but cloud has slightly higher CORR then no cloud. In addition, variation of CORR of no cloud is much smaller than cloud.

Preparation

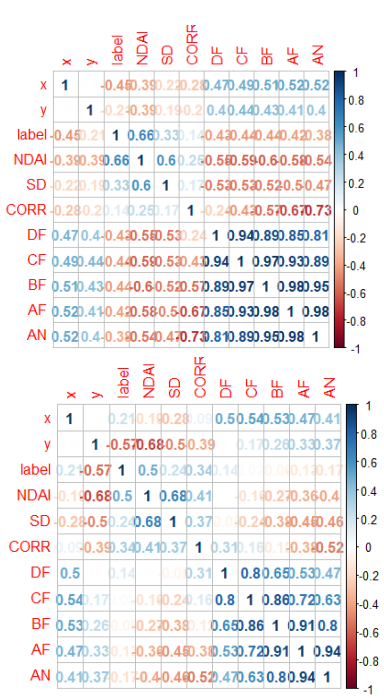
a.

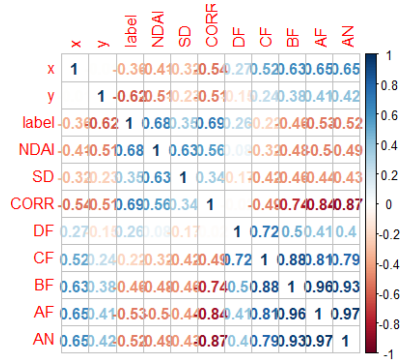
In this project, I use two ways to split data into training, testing and validation set. The first way is splitting each image into 9 small images by separating data into categories of x and y coordination mod 3. For example, small image 1 will have data with x coordinate mod 3=0 and y coordinate mod 3=0. Spliting data into modulos of 3 will give total of 27 small images. In order to further decrease dependency of data, testing, training and validation data will select 9 random small images to build new data. The second way of spliting data is the method of "Superpixel". A superpixel is built based on 9 adjacent observations from the original data, and data in the superpixel will be the average of 9 observations. However, taking average of 9 obeservations will cause label to be a non-integer. Therefore, blurring must be apply for superpixel method: if observations' label value is not an integer, then those observations will be remove from the data frames.

b.

Applying trivial classifier on modulo splitting method will give accuracy rate around 61%, and the 95% confidence interval of the accuracy is (60.65%, 61.37%). If applying trivial classfier on different splitting method, the accuracy rate of trivial classfier will change. Applyiing trivial classifier on superpixel data splitting method gives accuracy rate of 78.61%, with 95% confidence interval of accuracy rate is (77.7%, 79.49%). Changing data spillting method is one

way of inproving accuracy rate, superpixel increased accuracy ny 17.6%. Baseline for non-trivial classfication method is therefore 78.6%.



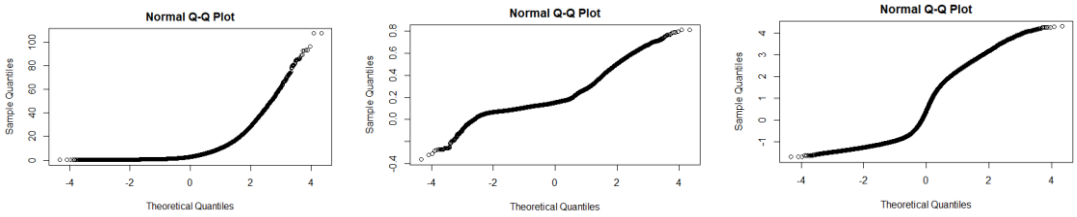


In image1, label has positive correlation only with NDAI, SD and CORR. In image 2, NDAI, SD, CORR have the highest correlation with expert label, with value of 0.69,0.68 and 0.35. In image 3, NDAI, SD, CORR have the highest correlation with expert label, with value of 0.5,0.34 and 0.24. We can conclude that NDAI, SD, CORR are three features that have highest correlation with label. Other variable such as DF, CF have no or negative correlation with value of label, so we can conclude that the best three features are CORR, SD and NDAI.

Modeling

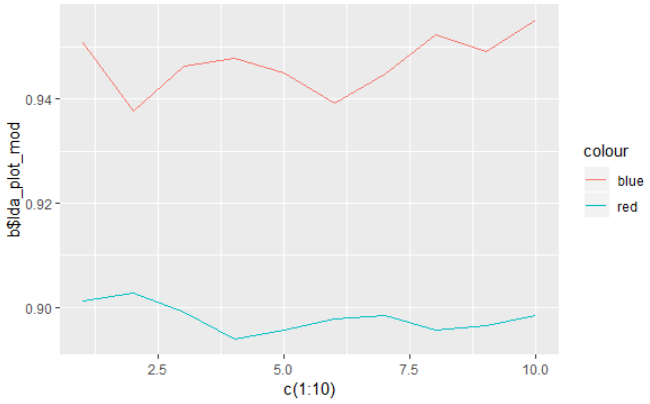
a.

LDA: LDA model is based on the assumption that data is gaussian.



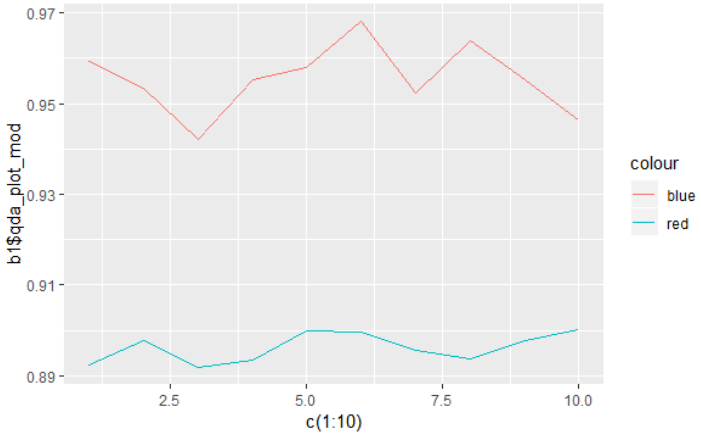
From quantile-quantile plot on SD, CORR and NDAI, CORR does not have a bell curve shape while SD and NDAI follows a bell curve shape. Therefore, LDA model is worth for testing on model. In addition, LDA requires the equality of covriance among the predictor across all levels of Y. Therefore, it is necessary t rescale data to have mean 0 and variance 1. Further, LDA requires data to have more sample than variable, and it is true for data I am using.

Applying cross validation for LDA classfier on splitting data using modulo and data using superpixel across 10 folds gives us accuracy plot on the right across 10 folds. Applying lda model on modulos gives average 89.79% accuracy, while applying on superpixel gives average 94.68% across 10 folds.



QDA

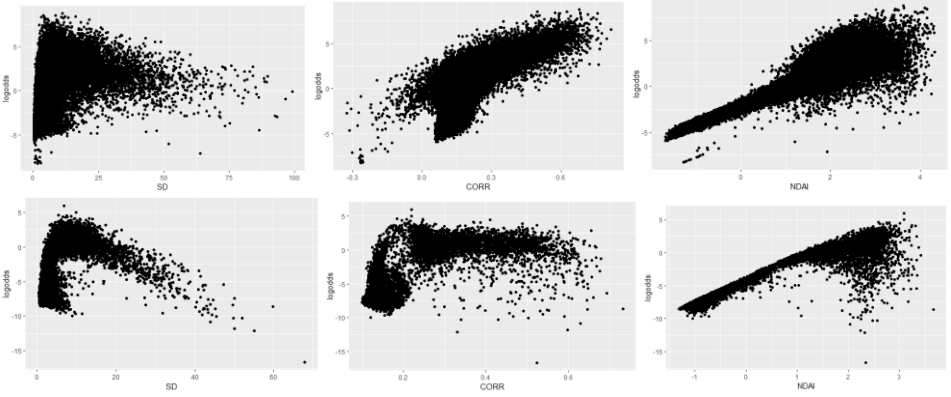
Similar to LDA, QDA assumes that data is gaussian which is proven in LDA section. Unlike LDA, QDA relaxes the assumption of equal covariance, so data doesn't need to be scale. QDA requires data to have more sample than variable, and it is true for data I am using.



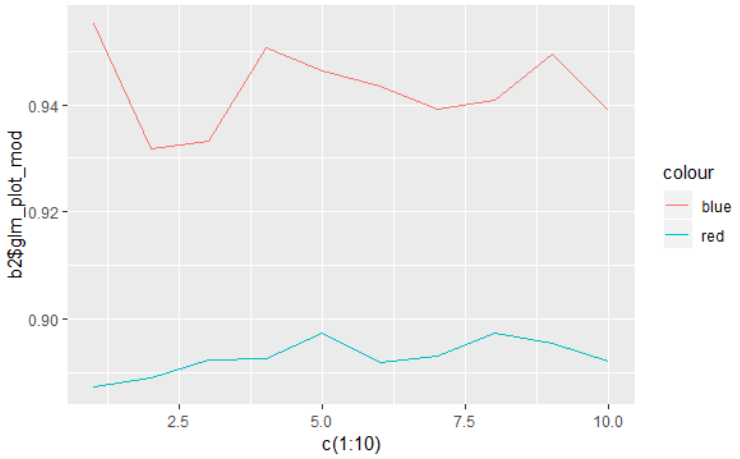
Applying cross validation for QDA classfier on splitting data using modulo and data using superpixel across 10 folds gives us accuracy plot on the right across 10 folds. Applying lda model on modulos gives average 89.62% accuracy, while applying on superpixel gives average 95.54% across 10 folds.

Logistic Regression

Logistic regression requires laarge set of data, and also assumes the linearlity of independent variable and lof odds.



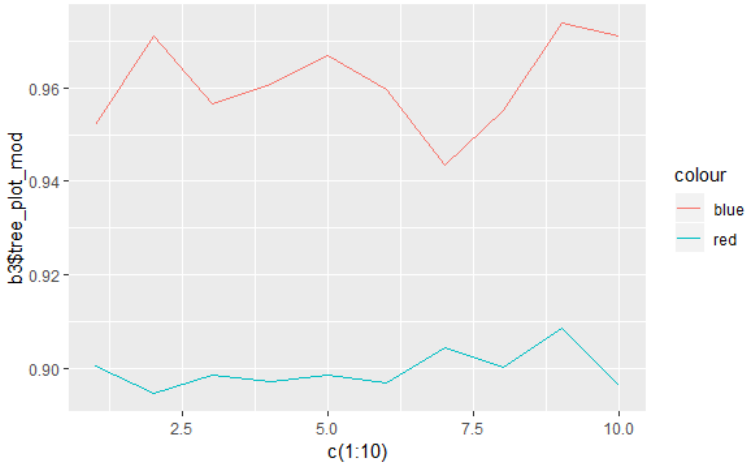
Pictures are plots of independent features selected for model and their log odd plots. First row pictures are from modulos, and second row pictures are from superpixel. From above pictures, only CORR meets the assumption of linearity.



Applying cross validation for logistic regression classfier on splitting data using modulo and data using superpixel across 10 folds gives us accuracy plot on the right across 10 folds. Applying lda model on modulos gives average 89.30% accuracy, while applying on superpixel gives average 94.29% across 10 folds

Decision Tree

Unlike logistic regression, LDA or QDA method, decision tree does not need any assumption on model. Therefore decision tree can be directly applied to the data without modifying data.



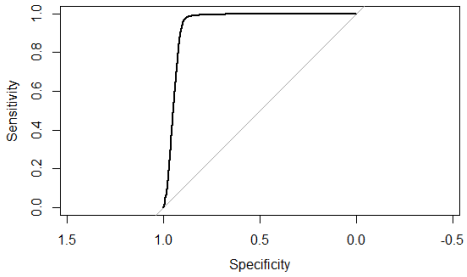
Applying cross validation for logistic regression classfier on splitting data using modulo and data using superpixel across 10 folds gives us accuracy plot on the right across 10 folds. Applying lda model on modulos gives average 89.96% accuracy, while applying on superpixel gives average 96.10% across 10 folds

Conclusion:

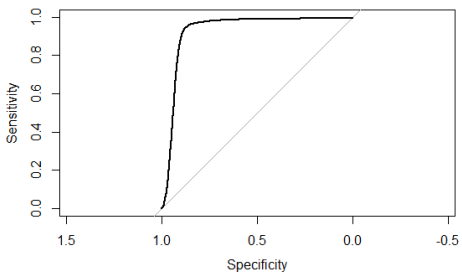
All four classifiers perform better on superpixel method comparing to modulo. In all four classfiers, decision tree has the best performance, lda, qda and logistic regression have relatively same accuracies.

2. Model Comparision with ROC

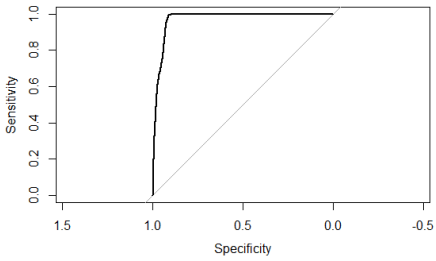
From the previous part, clearly superpixel has higher accuracy than modulo. So the next section I focus only on superpixel method. ROC curve's x axis indicates the amount of false positive, meaning the percent of data that does not belong to the category being classified to the category. Its y axis indicates the amount of true positive, meaning the percent of data that belongs to the category being classified to the category. Desired model should have high true positive rate and low false positive rate. Below are ROC curve fr four classifiers:



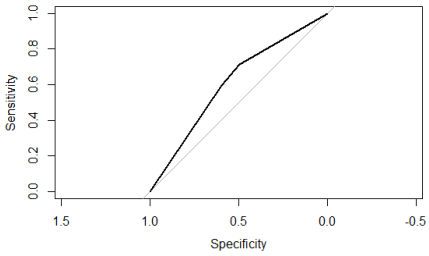
LDA



Logistic Regression



QDA



Decision Tree

The AUC value for LDA, Logistic Regression, QDA and Decision Tree are 0.9463, 0.9284, 0.9713, 0.612. The best cut-off value are at the point where it is the closest point to (0,1). (0,1) is the ideal point indicating no false postive and all true positive classification. For four models, the best cut-off are at y=0.9728, y=0.9427, y=0.9939, y=0.7147, with respect to the order of graphs. QDA particularly stand out for ROC curve comparison, as its AOC is closest to 1. Decision tree turns out to have large false positive rate comparing to other models.

c. Confusion Matrix

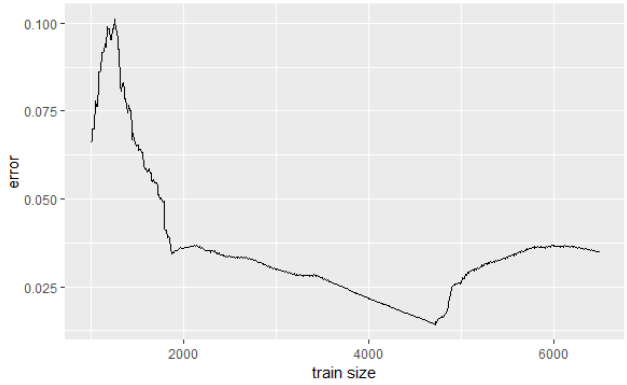
Comparing to ROC curve, confusion matrix can not only show the ratio of true postive rate to false positive rate, but also can disply other relevant data such as true negative rate and false negative rate. The accuracy rate is confusion matrix is (true positive+true negative)/(positive+negative)

	-1	1
-1	5061	3049
1	4932	3178

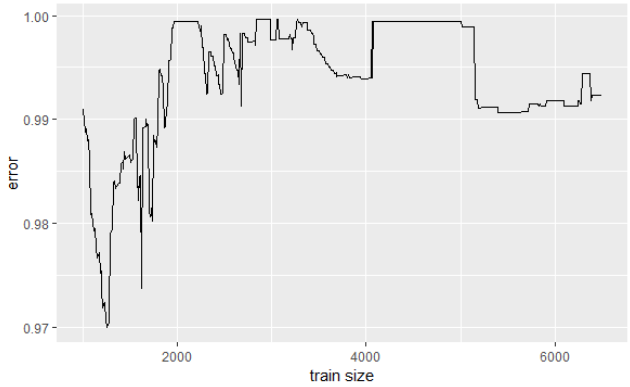
The accuracy for decision tree in confusion is 50.79%. For other models, logistic regression has accuracy of 78.69%, LDA has accuracy of 91.09%, and QDA has accuracy of 94.44%. From Confusion matrix and ROC curve, we can conclude that QDA is the best model for cloud detection because of its high accuracy, and LDA is good model with accuracy slightly below QDA. Logistic regression although has a satisfactory performance in ROC curve and cross validation, but it has a low accuracy in confusion matrix. Decision tree, although has the best performance in cross validation, fail both ROC curve and confusion matrix test.

Diagnostic

a. From part 3 of ROC curve, confusion matrix and cross validation tests, QDA classifier has the best performance among all four classifiers, therefore, I decide to run diagnostic on QDA model on superpixel.



As training size increases, the training error increases when size is small, and decreases almost monotonically until x=4500 and increases, and reach to a convergence of error=0.0375. Therefore, it is fair to say the error for qda model converges, and the model is fairly stable.



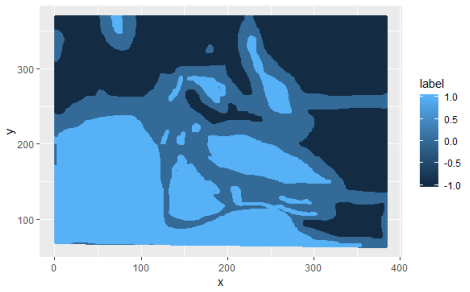
Similar to training error, the optimal cutoff value of ROC curve also converges to around 0.992 when training size is bigger than 4500. When data is smaller than 4500, the optimal cutoff value is quite unstable.

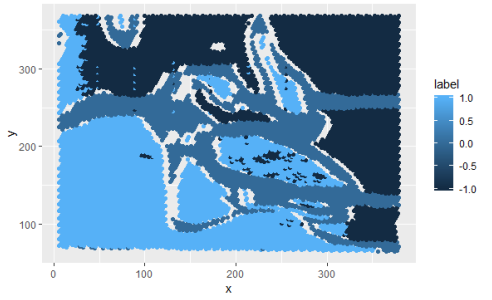
b.

For the qda prediction class and the actual image we have

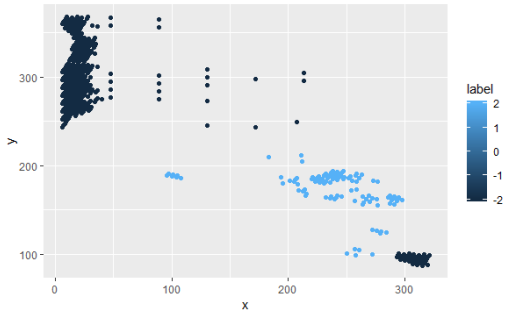
```
          clear         1
Qda     0.5027127 0.4972873

image   0.5326757 0.4673243
```

Percentage of misclassification is close to the actual image label. The error might come from the method of “blurring” from superpixel. Data at boundary of cloud with more than one label value is abandoned in the data set. The error might come from the data section being blurred.



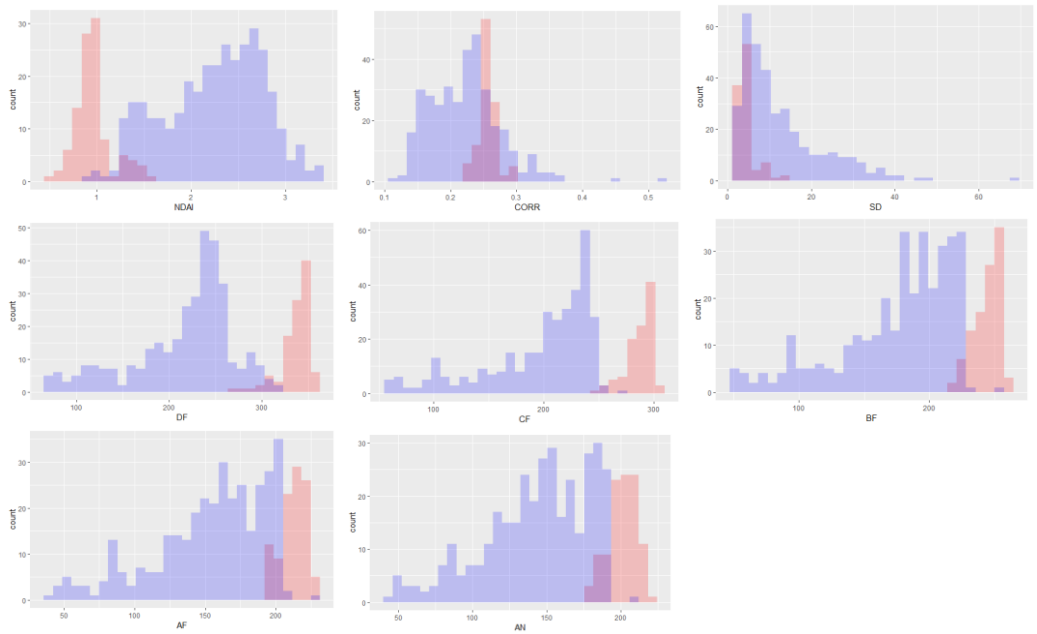


Above are picture of the original data and the data predicted by using QDA model. The picture on the left are data being misclassified by using QDA model. Dark points are clear being miss classfied as cloud, and light are cloud being misclassfied as clear. Missclafficiation occur at the top left corner, at the center, and at the bottom right of the image. Collecting other relevant fetures of misclassified data:




```
NDAI                SD                CORR
 Min.   :0.4866   Min.   : 1.643   Min.   :0.1155
 1st Qu.:1.3308   1st Qu.: 4.077   1st Qu.:0.1969
 Median :2.0843   Median : 7.319   Median :0.2365
 Mean   :1.9462   Mean   :10.785   Mean   :0.2300
 3rd Qu.:2.5648   3rd Qu.:13.793   3rd Qu.:0.2581
 Max.   :3.3486   Max.   :67.882   Max.   :0.5238
Comparing to the correct data
      NDAI                SD                CORR
 Min.   :-1.3056   Min.   : 0.3975   Min.   :0.01915
 1st Qu.:-0.5624   1st Qu.: 2.2569   1st Qu.:0.14071
 Median : 1.5795   Median : 6.0152   Median :0.20013
 Mean   : 1.0713   Mean   : 8.1343   Mean   :0.23225
 3rd Qu.: 2.2799   3rd Qu.:11.3214   3rd Qu.:0.27597
 Max.   : 3.7016   Max.   :67.8821   Max.   :0.73230
```

I find that misclassfied data has roughly equal CORR with the correct data, slightly higher SD than the original data and much higher NDAI comparing that of the original data. Taking a closer look to misclassfied data for all 8 features:



With cloud being misclassfied as clear as red, clear being misclassfied as cloud as blue. From camera angles, misclassfied data locates at around 200. Misclassfied data are ususally comes from high radiance readings, according to the plot. Clear being classified as cloud has lower radiance comparing to cloud being clear.

c.

From section b, there is some abnormality of features in misclassfied data. One way to diminish the effect is to reduce the scale of abnormality in order to increase accuracy of model. Among all three major features, NDAI has the highest abnormality. Therefore, I rescale NDAI in the data by the equatiom: $NDAI = log(|abs(NDAI + 1))|)$ Taking the natural log can reduce the scale of NDAI, and applying absolute value plus one can ensure the value inside of log function is bigger than 1.

After		Before	
1	0.9536232	1	0.9667149
2	0.9579710	2	0.9667149
3	0.9681159	3	0.9434783
4	0.9506531	4	0.9724238
5	0.9464544	5	0.9580925
6	0.9536232	6	0.9521045
7	0.9507959	7	0.9521739
8	0.9623734	8	0.9507959
9	0.9608696	9	0.9522431
10	0.9609262	10	0.9479016
mean	0.9578498	mean	0.9549552

The accuracy of cross validation arises by 0.3%. For percent of misclassification:

	Clear	Cloud
Misclassified	0.5059186	0.4940814
Actual	0.5326757	0.4673243

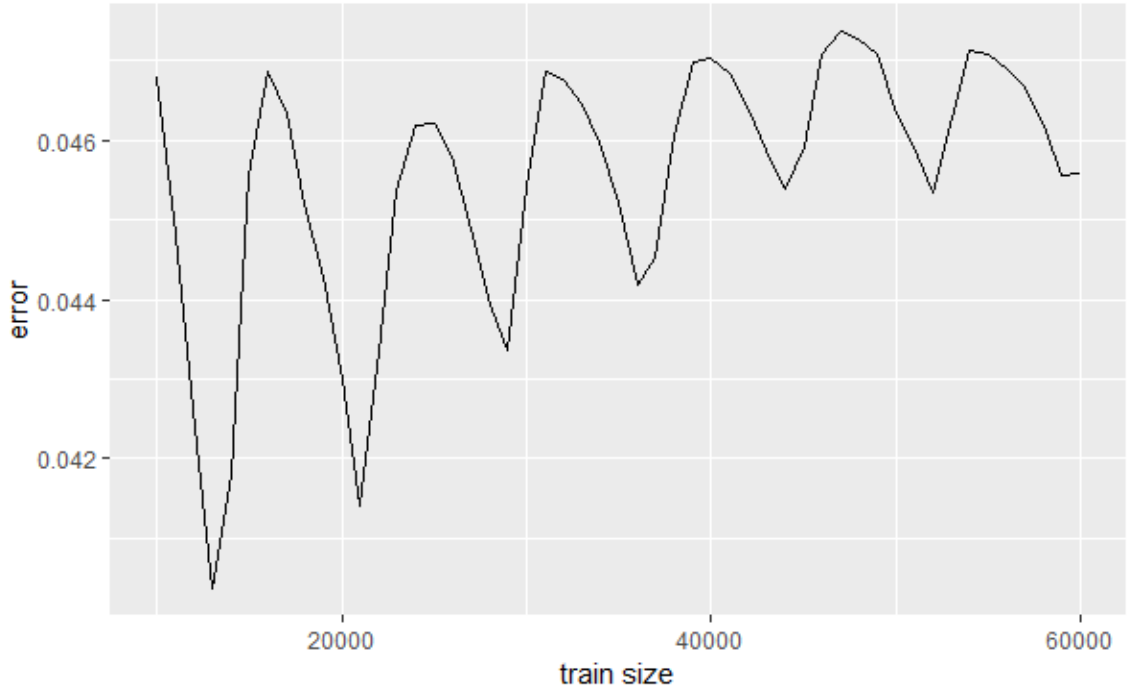
The difference of percentage between prediction and reality for the new feature decreases. The refore, I can conclude that the modified feature provides higher accuracy.

d.

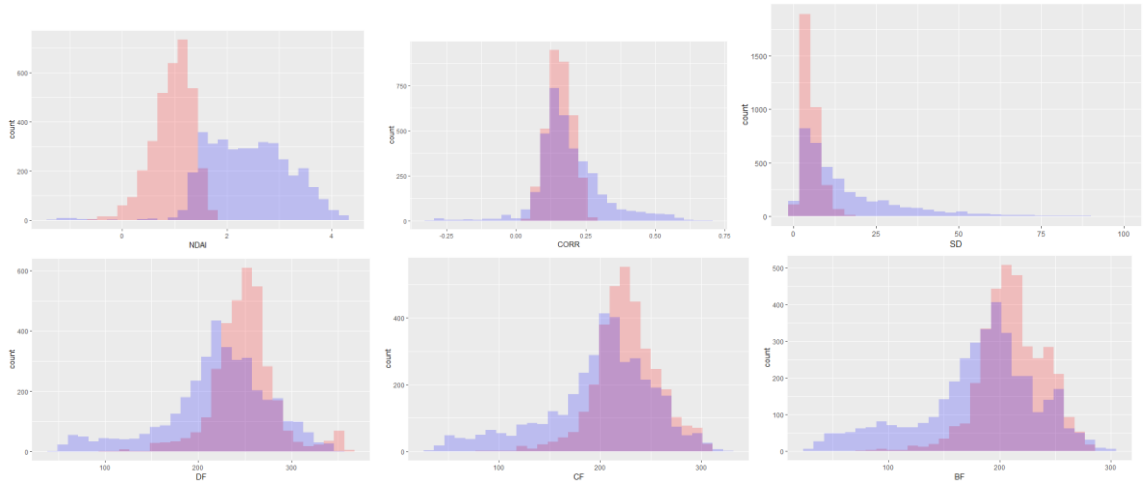
Switching the data splitting method to modulo, for the percentage of misclassification:

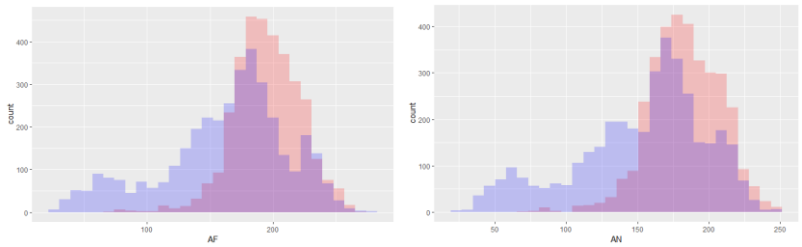
	Clear	Cloud
Misclassified	0.5463897	0.4536203
Actual	0.6133864	0.3866136

The difference of percentage between prediction and reality for the new data splitting method i ncrease.



Error of modulo splitting method oscillates as training size increases, but as size of data increase s, the size of oscillation decreases and the error will converge at the value around 0.046, base on the graph. But comparing to the converged training error of 0.0375 in method 1, training error of modulo increases.





Above are the plots of misclassfied data: NDAI, CORR, SD, DF, CF, BF,AF,AN. Opposing to superpixel method, modulo does not have a clear difference in radiance between cloud being clear and clear being cloud. The only difference is that cloud being clear has smaller variance then clear being cloud. However, in NDAI, there is a clear difference: cloud being clear has much smaller value then that of clear being cloud.

Conclusion

In this project, I study Professor's paper of cloud detection. After performing EDA on data, I select CORR, NDAI, SD to be the three main features. To split data into training, validation and testing, I use superpixel method and modulo method, then, I select four models: logistic regressiom, LDA, QDA and decision tree. After performing ROC curve, cross validation and confusion matrix, QDA method out-performed the other three and superpixel has better performance than modulo.
From QDA method on superpixel, there are two main clusters of error in the image. They might come from camera angle and aggregation of small cloud. In addition, NDAI value of misclassfied data has some abnormality. To reduce error, I revise the NDAI feature by taking natural log of the absolute value of (NDAI+1) to reduce the error.

Acknowledgment: All by myself

Github: https://github.com/hongzeliu/stat154