Project 2 Cloud Data Report

Authors: Rubina Aujla (SID 3031804725), Linshanshan Wang (SID 3032082808)

1. Data Collection and Exploration

1.a Summary of Paper

The paper presents a case study of detecting daytime cloud coverage in the Arctic. Levels of atmospheric carbon dioxide have increased throughout the twenty-first century, and Arctic surface air temperatures show strong sensitivity to changes in atmospheric carbon dioxide. This sensitivity has motivated scientists to analyze changes in Arctic cloud coverage, ice- and snow-covered surfaces, and atmospheric water vapor to determine whether or not future warming will occur in the region. The paper focuses specifically on the characterization of clouds over the Arctic, by classifying surfaces as belonging to one of two classes: "cloud-free" or "cloudy ice- and snow-covered."

The study used data collected from 10 Multiangle Imaging SpectroRadiometer (MISR) sensors, each composed of nine cameras, over the Arctic, northern Greenland, and Baffin Bay regions. Of the 233 geographically-distinct paths captured by the cameras every 16 days, path 26 was selected because it contained a variety of surface features: permanent sea ice, coastal mountains, and melting sea ice. Of the 60 data units taken from 10 orbits, 57 data units with 7,111,248 1.1-km resolution pixels with 36 radiation measurements per pixel were combined with 275-m red radiation measurements to form the data set.

To solve the Arctic cloud detection problem, the following strategy was utilized: construct 3 features (CORR, SD, NDAI), set thresholds on each feature and apply an enhanced linear correlation matching (ECLM) algorithm to each data unit, and train Fisher's QDA on ELCM algorithm-produced labels to predict probability of cloudiness. Results showed that the ELCM algorithm had a 91.80% agreement label rate for the 5 million testing pixels labeled by the expert in the study, which was higher than the agreement label rates for the MISR ASCM algorithm (83.23%), SFCM algorithm (80.00%), and offline SVM algorithm (80.99%). Thus, the statistical study proved that three physical features could indeed be used to distinguish clouds from ice- and snow-covered surfaces. The study also serves as an example of how statistics can be used to analyze data on current scientific research, with results eventually allowing scientists to understand how shifting cloud properties in the Arctic are related to increasing levels of atmospheric carbon dioxide.

1.b Summary of Data

Excluding unlabeled pixels from the total number of classified pixels in our dataset, we find that the proportion of pixels per class is:

- Not Cloud (label = -1): 0.6107824
- Cloud (label = +1): 0.3892176

When we plot the *x, y* coordinates of the pixels, with color of the region based on expert label, we observe that pixels per class tend to correspond to the same class across images (*Figure 1*). For example, majority of the pixels at position $x > 300$ and $50 < y < 300$ for each of the 3 plots are classified as "not cloud," while majority of the pixels at position $x < 200$ and $0 < y < 200$ for each of the 3 plots are classified as "cloud." It is likely that areas in close proximity to one another will be classified as the same type of surface: "cloud" or "not cloud." Thus, we conclude the data contains trends of spatial dependence, meaning we cannot assume that the samples are identically and independently distributed.

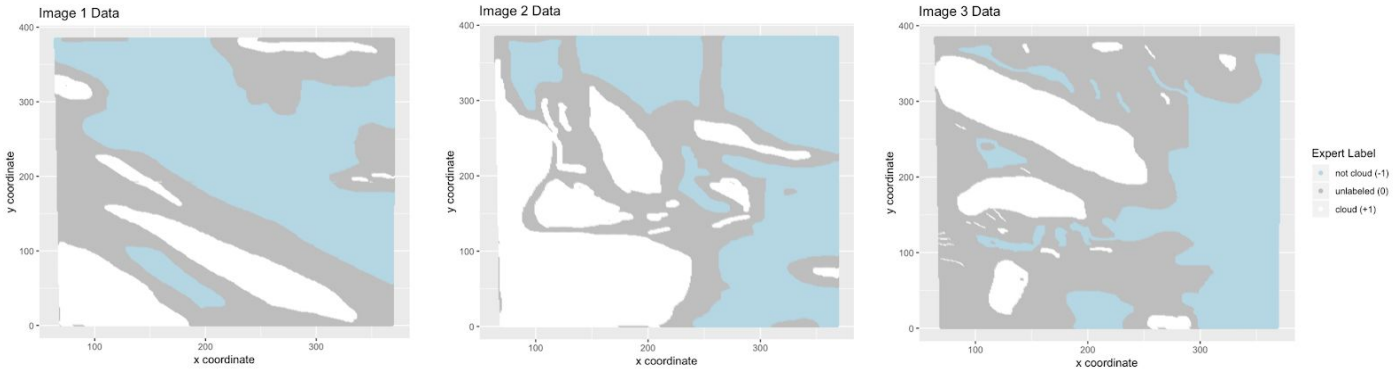

Figure 1: scatterplot of data by image (white = cloud, blue = non-cloud, gray = unlabeled)

1c. Exploratory Data Analysis

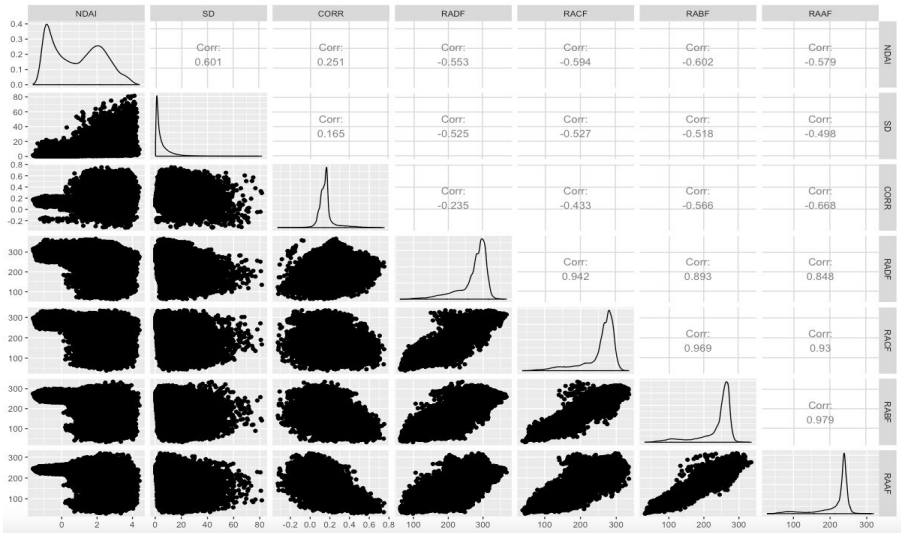

Figure 2: pairwise correlation plot between explanatory variables

(i) As shown in figure 2, the following pairs of features are strongly[1] correlated with each other: RADF and RACF (r = 0.942), RADF and RABF (r = 0.893), RADF and RAAF (r = 0.848). RACF and RABF (r = 0.969), RACF and RAAF (r = 0.93), RABF and RAAF (r = 0.979). Besides, NDAI and SD are moderately correlated with all the other features except for CORR. CORR is moderately correlated with RABF (r = -0.566) and RAAF (r = -0.668).

(ii) Differences can be noticed between two classes (cloud, no cloud) based on the features(Figure 3). NDAI and SD for *no cloud* class are both generally lower than those of *cloud* class. It can also be observed that or *cloud* class, the distribution of SD value is less peaked. Boxplots for five radiances values show similar patterns: the radiance for *no cloud* class are higher in values than those of *cloud* class, and the the distributions are peaked and highly left-skewed for no cloud class whereas for cloud class, the values are more spread out.

[1] Pearson correlation coefficient(r) is interpreted as showing: a perfect linear relationship if |r|=1, a strong relationship if 0.7<|r|<1, a moderate relationship if 0.5<|r|<0.7, a weak relationship if 0.3<|r|<0.5, and no relationship if |r| < 0.3.

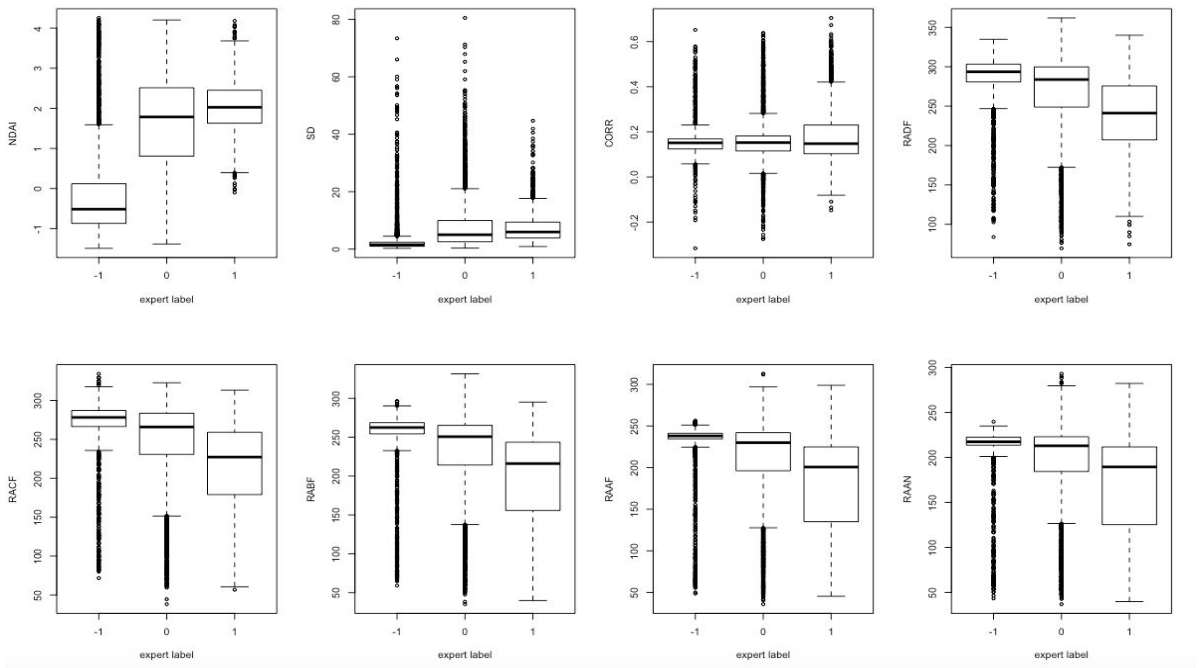

Figure 3: boxplots revealing relationship between the expert labels with each feature

2. Preparation

2a. Data Split

Taking into account the non-iid nature of the data, we propose the following two methods for splitting the data into training, validation and test sets: 10 blocks from a *2 x 5* array of the data and 10 vertical blocks from the data. For both methods, we first cut each image evenly into 10 blocks, with approximately the same number of datapoints in each block. Then 1 out of the 10 blocks is randomly assigned as the test set, 2 other blocks are randomly assigned as the validation set, and the rest are assigned as the training set. After performing the same procedure for all three images, we combine the individual training, validation and test sets together to form our full splitted dataset. *Figure 4* (left plot) shows an example of how Image 1 is split into 10 blocks by dividing data into a *2 x 5* array, and *Figure 4* (right plot) shows an example of how Image 1 is split into 10 vertical blocks from the data.

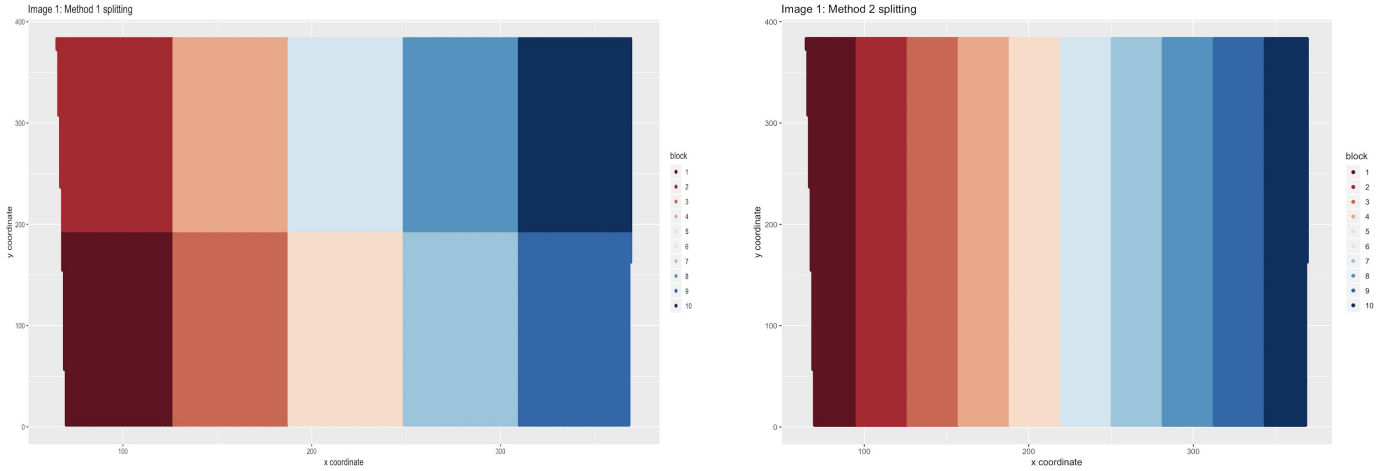

Figure 4: Splitting Image 1 into 10 blocks with Method 1 and 10 vertical slices with Method 2

2b. Baseline

The accuracy of a trivial classifier, which sets all labels to -1 (cloud-free) on the validation and test sets that exclude unlabeled points, is as follows for each method:

Method	Test Set Accuracy	Validation Set Accuracy
1	0.4815895	0.6173262
2	0.6657902	0.4904572

This classifier would have high average accuracy if majority of the data points in the validation and test sets were labeled as cloud-free (label = -1). However, it is not useful to merely classify data points based on the majority class of training data, as future data could have a different majority class. Thus, we should consider other metrics, besides accuracy, when comparing different classification models.

2c. First Order Importance

To identify the features that are the most relevant to predicting the class label, we investigated the distribution of the values of each of the eight features (NDAI, SD, CORR, RADF, RACF, RABF, RAAF and RAAN) using the histograms and density plots (Figure 5). We only used the data from the training set because feature selection is also a training process, thus the testing data should not be accessed at this stage. To find the features that can help us in doing the classification, we were looking for the histograms that show huge disparities between the distribution of the values of cloud vs no cloud class. NDAI and SD are the features we chose based on this criteria because their histograms show the greatest disparity between the two groups. The *no cloud* class has much lower NDAI and SD values when compared with the *cloud* class and the peaks of the density plots from the two classes are well-separated for both NADI and SD.

To choose the third feature, we standardized all the features (by their standard deviation) and for each class(*cloud* and *no cloud*), we computed the average value for each feature. We then took the absolute values of the differences between the two classes for each feature. The result is shown in Table 5.

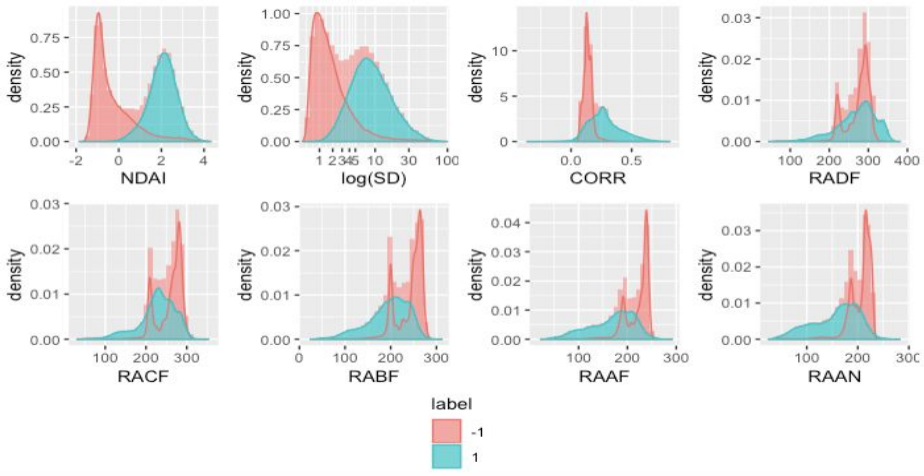

Figure 5: Comparing histogram of each feature in cloud vs no cloud class.

feature	NDAI	SD	CORR	RADF	RACF	RABF	RAAF	RAAN
Absolute difference	1.64	1.09	1.24	0.05	0.72	1.06	1.18	1.18

Table2: Absolute difference of the mean value for each feature between the two classes

We want to find the features that have high absolute difference between the two classes. We would consider such features useful because a high absolute difference shows that the two classes are distinguishable based on this feature. We found that besides the two features that we have chosen(NDAI and SD), our best candidate is CORR. RAAF and RAAN also have high absolute difference of mean value, but we think it is better to use CORR because it captures and condenses information from the radiances measures. The paper(Shi, Tao et al.) suggests that high CORR values are expected over cloud-free areas because the radiation measured by MISR camera is scattered from the same surface, while CORR values for the cloud area are expected to be lower. Our matric, together with the domain knowledge, suggest that CORR is a strong candidate.

2d. Generic Cross Validation Function

The generic cross validation (CV) function *CVgeneric* takes in a generic classifier, training features, dataset, training labels, number of folds *K*, and a loss function and outputs the *K*-fold CV loss for the training data. The R code is stored in the associated GitHub repository.

3. Modeling

3.a Classification Methods

We fit four classification models on both training sets: Logistic Regression, Bayes Generalized Linear Model, QDA (Quadratic Discriminant Analysis), and CART (Classification and Regression Tree). We performed K-fold cross-validation, using K=4, with accuracies across folds and test accuracies listed in the table below. Assumptions for each of these models are also included below. We used K=4 to ensure that each divided dataset would contain enough variability to represent the population dataset, as a larger value of K would create more small sets with potentially increased bias.

Our results show that all four models, each fit on two types of training and test sets, achieved close prediction accuracy values across folds; the average prediction accuracy across folds ranged from 0.8741 to 0.9098. Accuracy across folds for the Logistic Regression and Bayes Linear Regression models were highly similar, which is likely due to the underlying similarity in how both models fit to data. Overall, the CART model appears to achieve the highest prediction accuracy across folds for both training and test sets; while we attempted to control for spatial dependence when splitting data into training and test sets, a nonparametric method, such as CART, could control for this condition when classifying data. As expected, test set error slightly decreased from training set error for each of the models. Because these decreases were minor, it appears that the training sets were representative of the test sets. If we were to select our "best" classification model from these results, we would choose QDA, as it achieved the highest test set prediction accuracy across folds.

- Logistic Regression
    - The logistic regression model assumes a binary response variable, little to no multicollinearity among independent variables, a large sample size, independent observations, and a linear relationship between independent variables and log odds. The surface labels are binary (-1, +1), our features were selected such that they would not be highly multicollinear, and our sample size is quite large. While our raw data is not independent, we believe our data split methods would control for spatial dependence and consider the assumption met. After computing the log odds of predicted probabilities for a logistic regression model on the first training set, we find that the correlation of each feature (NDAI, CORR, SD) with the log odds is somewhat linear ($r = 0.952$, $r = 0.733$, $r = 0.533$), so the assumption is met.
- Bayes Generalized Linear Model

    - The assumptions of the Bayesian GLM are similar to the logistic regression model. However, this approach assumes that the unobserved parameters are random and utilizes the given observations (training data) to approximate and assign a prior distribution to the unknown parameters.
- Quadratic Discriminant Analysis (QDA)
    - QDA assumes that observations within each class follow a multivariate Normal distribution, little to no multicollinearity among independent variables, and independent observations. Because we fit this model on a large number of samples (training data) from the population, we may assume that the normality condition is satisfied by the Central Limit Theorem. As discussed above for Logistic Regression, we also assume the absence of multicollinearity among independent variables and independent observations.
- CART
    - The CART model uses a decision tree model to classify data by setting a threshold to divide values into intervals related to decisions. This method is non-parametric nad makes decisions solely based on the observed data, which means there are no assumptions about data or error term distributions. The data is described by independent variables such that a decision tree would produce logical predictions of labels.

Logistic Regression: Accuracy across Folds

Input Data	*Training Set (Folds Method 1)*	*Test Set 1 (Folds Method 1)*	*Training Set 2 (Folds: Method 2)*	*Test Set 2 (Folds: Method 2)*
Method 1 Data (10 blocks)	0.8889561 0.9082494 0.9146731 0.8698524 *Avg: 0.8954*	0.9018965 0.8348526 0.8658296 0.9670216 *Avg: 0.8924*	0.8909880 0.9107539 0.8978372 0.8872215 *Avg: 0.8967*	0.8684245 0.8311942 0.8730450 0.9269264 *Avg: 0.8748*
Method 2 Data (10 slices)	0.8915126 0.9026032 0.9105545 0.8592920 *Avg: 0.8909*	0.8727268 0.8175459 0.8664441 0.9721279 *Avg: 0.8822*	0.8818386 0.9135920 0.8892533 0.8809958 *Avg: 0.8914*	0.8667791 0.7990210 0.8874195 0.9309263 *Avg: 0.8710*

Bayes Linear Regression: Accuracy across Folds

Input Data	*Training Set (Folds Method 1)*	*Test Set 1 (Folds Method 1)*	*Training Set 2 (Folds: Method 2)*	*Test Set 2 (Folds: Method 2)*
Method 1 Data (10 blocks)	0.8889561 0.9082494 0.9146731 0.8698593 *Avg: 0.8954*	0.9018965 0.8348526 0.8658296 0.9670216 *Avg: 0.8925*	0.8909951 0.9107539 0.8978372 0.8872215 *Avg: 0.8967*	0.8684461 0.8311942 0.8730450 0.9269435 *Avg: 0.8749*
Method 2 Data (10 slices)	0.8915196 0.9026032 0.9105470 0.8592920 *Avg: 0.8910*	0.8727515 0.8175237 0.8664441 0.9721279 *Avg: 0.8822*	0.8818314 0.9136062 0.8892533 0.8809958 *Avg: 0.8914*	0.8667791 0.7990210 0.8874195 0.9309263 *Avg: 0.8710*

QDA: Accuracy across Folds

Input Data	*Training Set (Folds Method 1)*	*Test Set 1 (Folds Method 1)*	*Training Set 2 (Folds: Method 2)*	*Test Set 2 (Folds: Method 2)*
Method 1 Data (10 blocks)	0.9015878 0.9197678 0.9145400 0.8750104 *Avg: 0.9027*	0.9193406 0.8307276 0.8836290 0.9693722 *Avg: 0.9007*	0.9000982 0.9156777 0.9021294 0.8880870 *Avg: 0.9014*	0.9004416 0.8413691 0.8828855 0.9398661 *Avg: 0.8911*
Method 2 Data (10 slices)	0.9015620 0.9129666 0.9043150 0.8617279 *Avg: 0.8951*	0.8956927 0.7988179 0.8813484 0.9735289 *Avg: 0.8873*	0.8901282 0.9190807 0.8891411 0.8784065 *Avg: 0.8941*	0.8986600 0.8051215 0.8939813 0.9411252 *Avg: 0.8847*

CART: Accuracy across Folds

Input Data	*Training Set (Folds Method 1)*	*Test Set 1 (Folds Method 1)*	*Training Set 2 (Folds: Method 2)*	*Test Set 2 (Folds: Method 2)*
Method 1 Data (10 blocks)	0.9105077 0.9213331 0.9167665 0.8760074 *Avg: 0.9061*	0.9176869 0.8412795 0.8524372 0.9829692 *Avg: 0.8985*	0.9156896 0.9177609 0.9099047 0.8959224 *Avg: 0.9098*	0.8955492 0.8428765 0.9199381 0.9047537 *Avg: 0.8907*
Method 2 Data (10 slices)	0.9113438 0.9167291 0.9161542 0.8730291 *Avg: 0.9043*	0.8924154 0.8519691 0.8530885 0.9841216 *Avg: 0.8953*	0.9137201 0.9223072 0.8959941 0.8924798 *Avg: 0.9061*	0.8607174 0.8269435 0.9219379 0.9133004 *Avg: 0.8807*

Table 3: Accuracies across folds and test accuracies for each model

3.b ROC Curves

We generate the ROC curves for each of the four models(Logistic Regression, Bayes Generalized Linear Model, QDA, and CART) on the two training sets created using the different methods in 2.a. For each plot the true positive rate (Sensitivity) is plotted in function of the 1 - false positive rate (Specificity) for different cut-off points. Each point on the ROC curve represents a sensitivity/specificity pair corresponding to a particular decision threshold. A test with perfect discrimination (no overlap in the two distributions) has a ROC curve that passes through the upper left corner (100% sensitivity, 100% specificity). Therefore the closer the ROC curve is to the upper left corner, the higher the overall accuracy of the test (Zweig & Campbell, 1993).

The AUC(Area Under the Curve) is reported for each ROC curve(Figure 6). The closer the AUC value is to 1, the better the performance of the model. However, since these AUC values only reflects the model's performance on the training set, we do not consider it as a reliable metric for comparing the models. Thus, to better compare the models, we also calculated he AUC for the test sets(Table 4).

Our results show that for test set 1(data splitted using method 1), the AUC values range from 0.901 to 0.923, with CART having the highest AUC. For test 2(data splitted using method 2), however, the AUC values are slightly higher, ranging from 0.963 to 0.978. QDA has the highest AUC value. The discrepancies between the AUC of the two test sets might be due to the randomness the data splitting procedure.

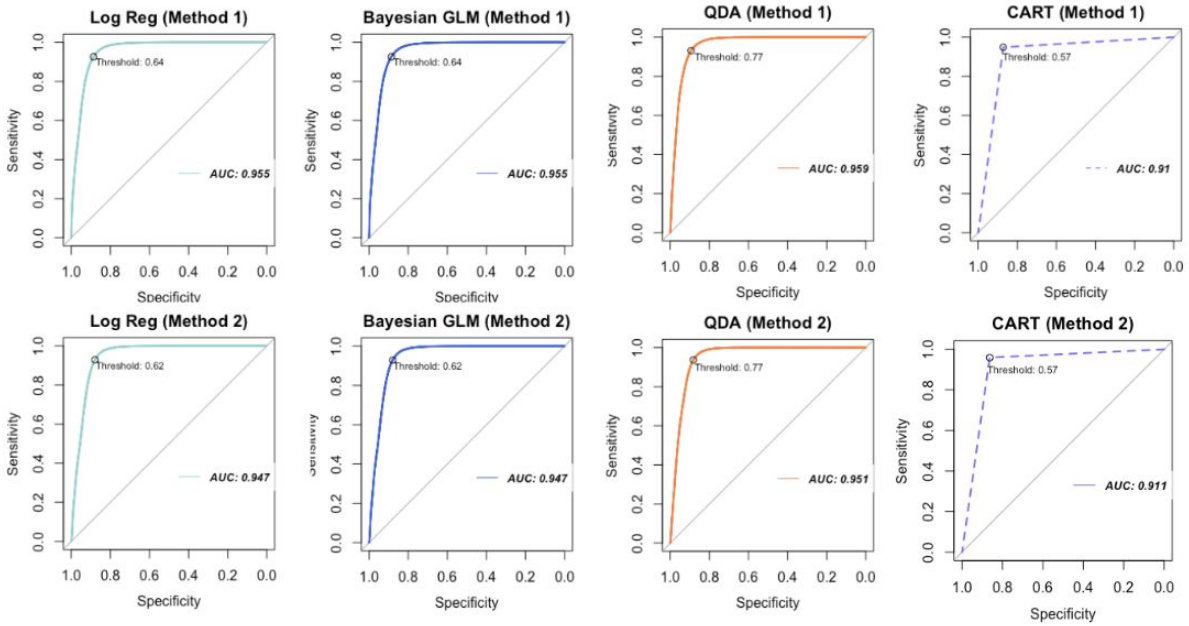

Figure 6: ROC curve of each model(Logistic Regression, Bayes Generalized Linear Model, QDA, and CART) generated with training data. The best cutoff value(circled) is selected such that it is the point closest to the top-left part of the plot with perfect sensitivity or specificity.

Column1	*Logistic Regression*	*Bayes Linear Regression*	*QDA*	*CART*
AUC (splitting method 1)	*0.901*	*0.901*	*0.921*	*0.932*
AUC (splitting method 2)	*0.975*	*0.975*	*0.978*	*0.963*

Table 4: AUC of each model(Logistic Regression, Bayes Generalized Linear Model, QDA, and CART) for the test sets.

To find the optimal cutoff value, we find the point closest to the top-left part of the plot. The optimality criterion is:

$$min((1 - sensitivities)^2 + (1 - specificities)^2)$$

The threshold that meets this criterion would be our optimal cutoff value because ideally we would like our model to have perfect sensitivity and specificity, which is point (1,1) on the top-left corner of the plot. Thus, for all the points located on the ROC curve, the point that is the closest to (1,1) would be the best model we can get given other parameters of the model fixed. This selection criterion for optimal cutoff value also take into account the fact that for this classification task, false positive and false negative are considered equally important. Therefore, sensitivities and specificities have equal weights in the formula.

3.c Relevant Metrics

We first examined the confusion matrix of each model (8 models in total: 4 algorithms, each trained on 2 training sets that are obtained using the two different splitting methods). Optimal cutoff value of each model is chosen based on the criterion described in 3.b. We used the optimal cutoff values as thresholds to assign predicted labels to the testing sets. The numbers of TP, FP, TN and FN cases are computed for each of the eight models(Figure 7) .

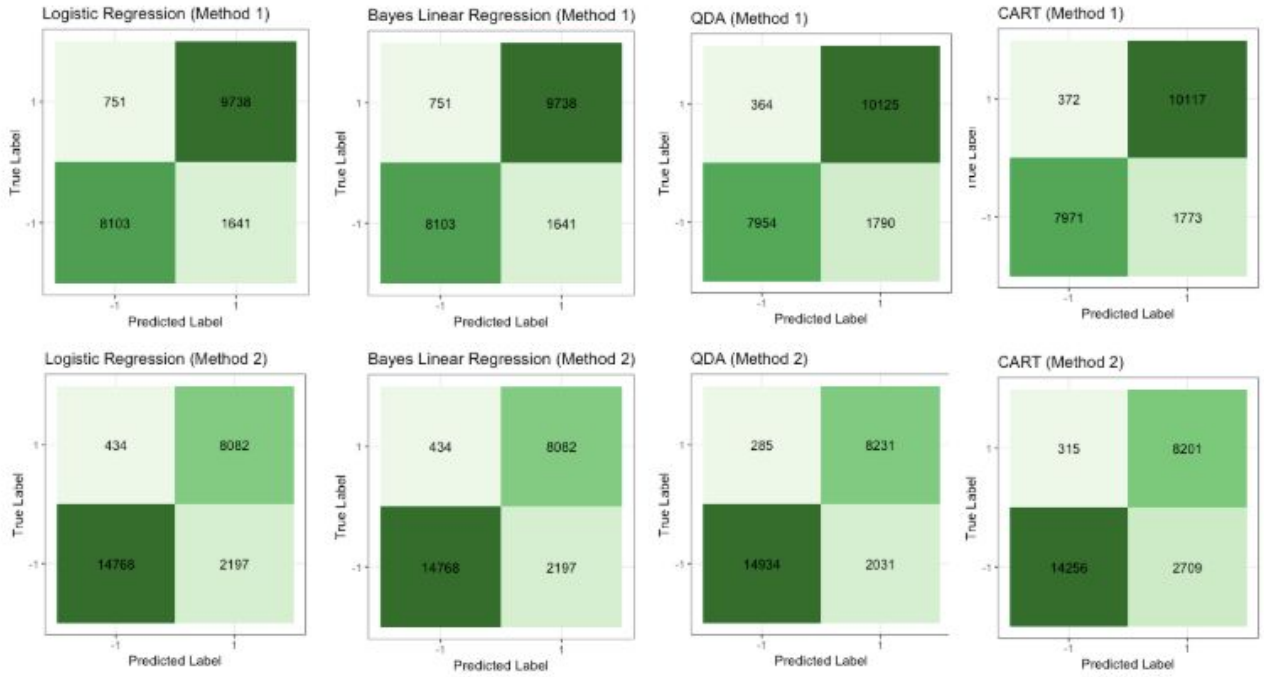

Figure 7: Confusion Matrix for each model on both test sets. Optimal cutoff value for each model is chosen based on the criterion defined in 3.b.

From the confusion matrices, we can observe that the performance of Logistic Regression and Bayes Linear Model are very similar to each other. QDA and CART give fewer false negatives (*predicted: no cloud, true label: cloud*) but more false positives (*predicted:cloud, true label: no cloud*) when compared with Logistic Regression and Bayes Linear Model. In order to better compare the models and access their performances, we calculated *Specificity, Sensitivity, Precision, Recall, F1 score and MCC* for each model. The results are summarized in Table 5.

Model		Specificity	Sensitivity	Precision	Recall	F1 score	MCC
Logistic Regression	(method1)	0.9284	0.8315	0.9152	0.8316	0.8714	0.7654
	(method2)	0.9490	0.8704	0.9715	0.8705	0.9182	0.7880
Bayesian GLM	(method1)	0.9284	0.8316	0.9152	0.8316	0.8714	0.7654
	(method2)	0.9490	0.8704	0.9715	0.8705	0.9182	0.7880
QDA	(method1)	0.9653	0.8163	0.9652	0.8163	0.8807	0.7937
	(method2)	0.9665	0.8803	0.9813	0.8803	0.9280	0.8147
CART	(method1)	0.9645	0.8180	0.9554	0.8180	0.8814	0.7943
	(method2)	0.9630	0.8403	0.9783	0.8403	0.9041	0.7658

Table 5: Model performances

It can be observed that for all of our models, specificity is higher than sensitivity. Specificity refers to the model's ability to correctly designate a *no cloud* data point as *no cloud* (have few false positive results), while sensitivity refers to the model's ability to not miss any *cloud* data point (have few false negative results). Our results show that out models are highly specific but not so much sensitive.

Precision is to measure the quality of our predictions only based on what our predictor claims to be positive regardless of all it might miss. However, Recall is to measure such quality with respect to the mistakes we did. Our models have high precision but lower recall, which suggests that there are many cases that should have been predicted as *cloud* are flagged as *no cloud*. Neither precision not recall can give us a holistic view of the model's performance because they both hide some information individually, but F1 score and MCC(Matthews correlation coefficient), combining information from the first four metics, can be more indicative of the model's performance. Comparing the F1 score and MCC for each model, we find that QDA model produces the best F1 score and MCC, thus the most suitable for making predictions of this classification problem.

4. Diagnostics

4.a In-Depth Analysis

To examine the stability of using QDA algorithm for prediction on our dataset, we visualized the decision boundaries created by the algorithm on training set 1 verses training set 2. Since it is impossible to visualize the quadratic decision boundaries in two dimension, we only looked at the their projections in the two-dimensional space created by each combination of the input variables. Take into account the computational complexity of generating the plot, we randomly sampled one block from training set 1 and one slice from training set 2 as representatives of their corresponding training set, and generated the plot below(Figure 8).

By visual inspection, in spite of the difference between the two splitting method and the random sampling of the training block/slice, the decision boundaries created by fitting QDA on training set 1 vs training set 2 are extremely similar in shape. This result gives us confidence that our model has a small variance, and is able to stably capture the difference between the two classes. When applied to make predictions on future datasets, we expect the model to have reasonable variability of test accuracy.

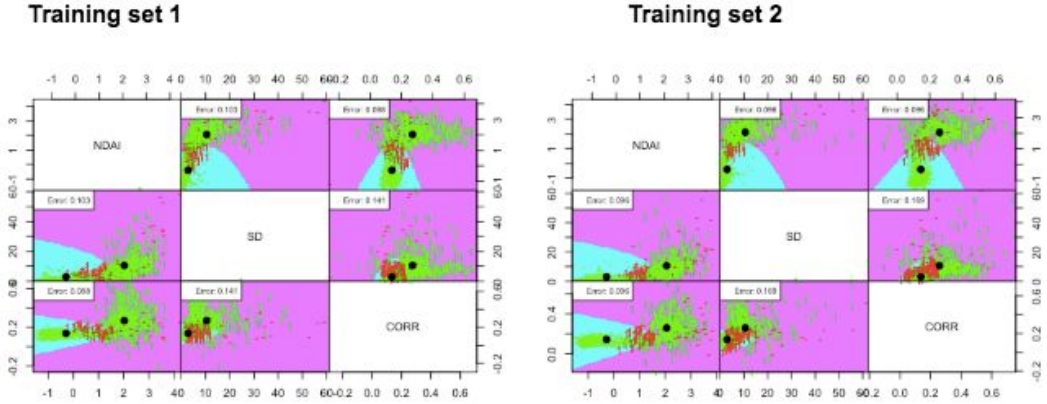

Figure 8: Visualization of QDA boundaries(estimated) when fitted on training set 1 vs training set 2.

Besides, in the two-class classification problem, QDA models each class density as a multivariate Gaussian distribution with mean μ_k and covariance matrix Σ_k where k=-1, 1 denotes the class label. The posterior probability of $x \in class\ k$ is given by Bayes rule. The estimated posterior probability serves as the predicted probability of cloudiness of a data point. In 3.b. The optimal cutoff value of the posterior probabilities are identified so that those probabilities can be used to classify the pixel as either cloud or no cloud. Now, besides using the probabilities to make discrete predicted labels(either -1 or +1) for each datapoint, we also

regard them as a continuous variable that indicates the cloudiness of each pixel. As we expected, the distribution of the predicted posterior probabilities(Figure 9) is bimodal, with values clustered on both ends. Furthermore, we looked at the agreement of the the predicted cloudiness with the expert label in the test sets. For instance, if a pixel is labeled as cloud(+1) and our predicted probability of cloudiness is 0.95, then the agreement between our prediction and the expert label on this pixel is 0.95. On the other hand, if for a pixel that is labeled as non cloud(-1) our predicted probability of cloudiness is 0.95, then the agreement for this pixel is 0.05. Ideally, we would like to see our predicted probability of cloudiness has high agreement with the expert label. The histogram(Figure 10) indeed shows that in spite of some outliers with low agreement, most data points have very high agreement(mean= 0.866, median=0.999), which suggests that our model performs well not only in classifying the pixels to the two classes, but also in providing a cloudiness label to each pixel.

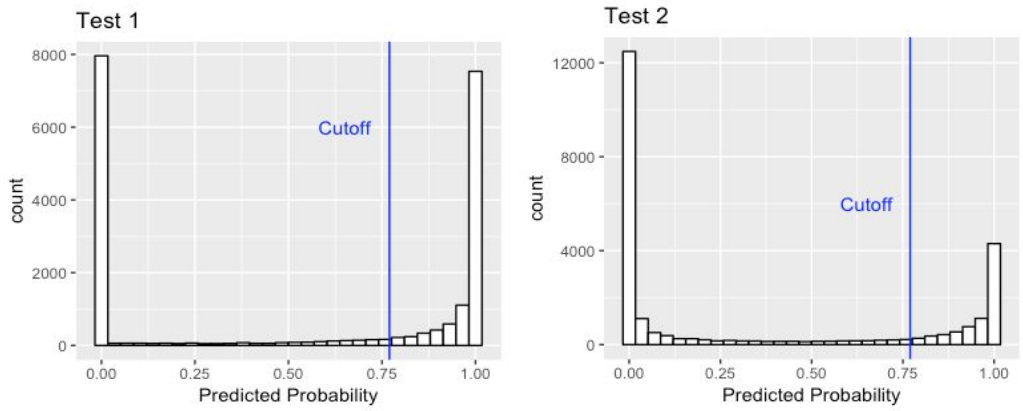

Figure 9: distribution of the predicted posterior probabilities.

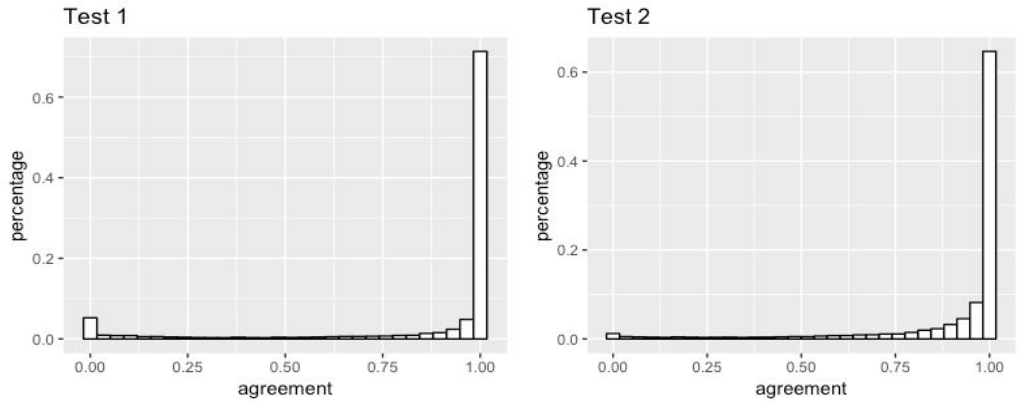

Figure 10: Distribution of agreement score between predicted probabilities and expert labels.

4.b Misclassification Errors

Upon fitting a QDA model and CART model on each of the two training sets, we find that misclassification errors on the training sets are concentrated in particular areas of the images. As seen in Figure 9, misclassified cloud areas (red) tend to be located along the boundaries of large non-cloud areas, and a high proportion of misclassified non-cloud areas (dark blue) tend to fall in the region of $x > 200$ and $0 < y < 80$, for both QDA and CART methods. When comparing the median values of *NDAI, SD,* and *CORR* for the training sets, it appears that data points with high *NDAI* and *SD* values are likely to be misclassified; these results are summarized in Table 6 below for both models.

Misclassification Patterns by Median Feature Values: QDA and CART						
Training Set	Feature	Median	Missed Cloud (QDA)	Missed Cloud (CART)	Missed Non-Cloud (QDA)	Missed Non-Cloud (CART)
1	*NDAI*	0.3413	1.0332	0.6015	2.334	1.6568
	SD	2.7218	4.5257	3.8936	9.1964	5.1488
	CORR	0.1533	0.15744	0.1831	0.1703	0.1506
2	*NDAI*	0.4348	0.9857	0.5490	2.373	1.8380
	SD	2.9081	4.5305	3.9641	9.7188	6.0122
	CORR	0.1564	0.16195	0.17951	0.1720	0.1557

Table 6: Median values of features for misclassified observations in both training sets for QDA and CART

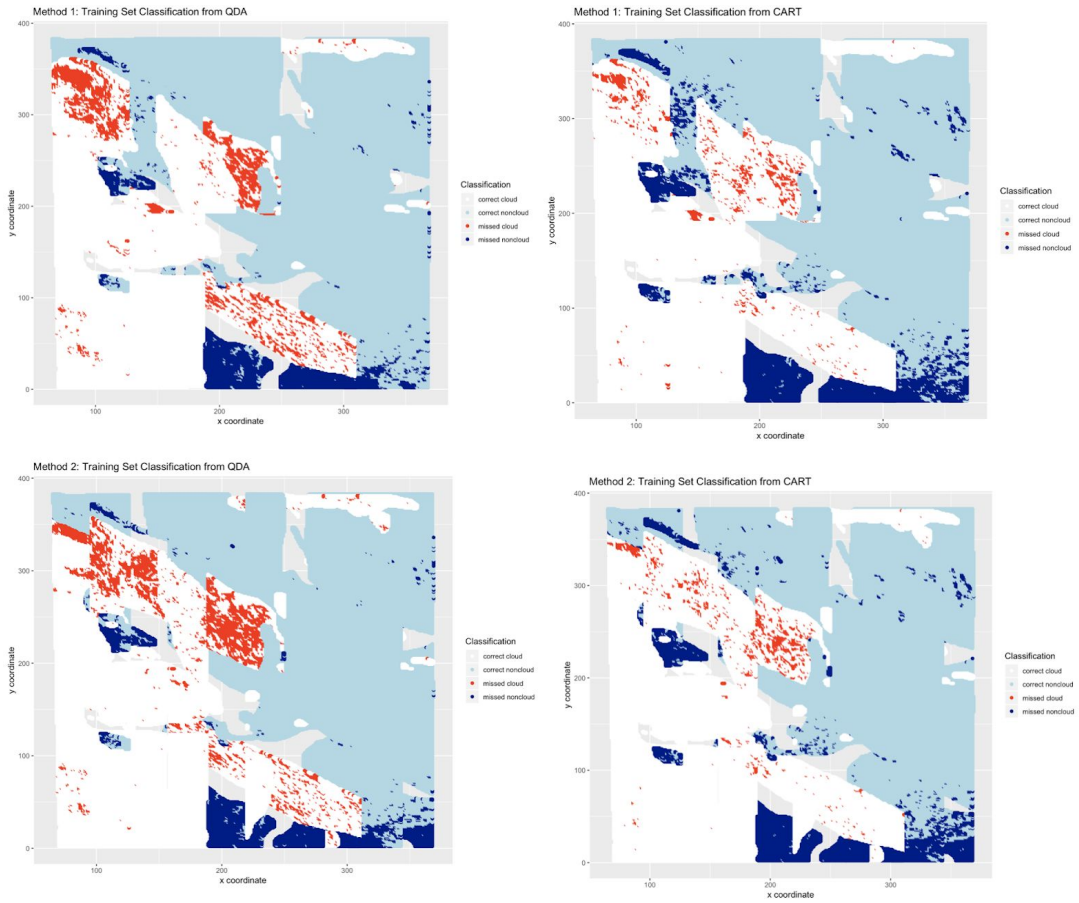

Figure 11: Misclassification patterns in both training sets for QDA and CART

4.c Better Classifier

Our results in 4(a) and 4(b) demonstrate that QDA and CART models achieve similar accuracy in classifying observations in both training sets. QDA showed the highest model performance of the four models we tested, in terms of F1 and MCC values, which suggests that QDA is our best classification model. However, we have only trained and tested the model's accuracy on labeled data, which means we cannot say with full certainty how the model will perform on future data without expert labels. A better classifier would make use of the unlabeled data points we excluded when training the model to help with classifying future data. Pseudo-labeling is one method that could be used for this purpose. One could train a classification model on labeled data whilst simultaneously classifying unlabeled data, based on the observations' similarity to known

classes. By including unlabeled data in the training set, the model would likely be more stable when working on future data without expert labels.

One could also improve the classifier by adding more features to the model. Because the MISR radiation measurements are highly similar in terms of captured information added to the model, we fit separate models with a single angle featured added to the original QDA model. Across both types of training data, we find that all average test accuracies across folds for K-fold cross-validation for models with added features are lower than those of the original QDA model (Table 7). Thus, to avoid overfitting whilst improving the model, it would perhaps be more beneficial to fit a different classification model or utilize pseudo-labeling, rather than add features to the model.

Average Test Accuracy across Folds (K = 4)			
Added Feature	Input Training Set	Test Set 1 (Folds Method 1)	Test Set 2 (Folds: Method 2)
NONE	1	**0.9007**	**0.8911**
	2	**0.8873**	**0.8847**
RAAF	1	0.8747	0.8821
	2	0.8767	0.8746
RADF	1	0.8862	0.8854
	2	0.8820	0.8768
RACF	1	0.8870	0.8844
	2	0.8799	0.8746
RABF	1	0.8794	0.8843
	2	0.8779	0.8756
RAAN	1	0.8761	0.8821
	2	0.8793	0.8741

Table 7: Average test accuracy across folds for K-fold cross validation with added features to QDA

4.d. Modify Data Splitting

We used two methods of splitting data - 10 blocks from a *2x5* array and 10 vertical slices from our dataset - to obtain two training sets to fit our QDA and CART models. Our results in 4(a) and 4(b) show that while there are minor differences in the number of misclassified observations in a particular area, both data splitting methods yield similar results in model fitting and misclassification patterns. For instance, majority of the misclassified non-cloud surfaces fall in the lower right quadrant of our plots in Figure 8, and observations with high median *NDAI* and *SD* values tend to be misclassified for both training sets. Therefore, we would expect QDA to perform with reliable accuracy on similar datasets in the future.

4.e. Conclusion

Resources

- http://deeplearning.net/wp-content/uploads/2013/03/pseudo_label_final.pdf
- https://arxiv.org/pdf/1805.06118.pdf
- https://cran.r-project.org/web/packages/caret/vignettes/caret.html
- http://topepo.github.io/caret/train-models-by-tag.html
- https://rdrr.io/cran/caret/man/models.html

GitHub: https://github.com/raujla/STAT154-Project2