STA521 PROJ2 Arctic Cloud Detection

Huiying Lin(hl457@duke.edu), Yanjiao Yang(yy361@duke.edu)

December 6, 2022

1 Data Collection and Exploration

1.1 Summary

According to today's global climate models, doubling of atmospheric carbon dioxide levels will cause global surface air temperatures to increase by 1.5-3.5 K throughout the 21st century. Climate in the Arctic may be the most sensitive, and study of the topic requires accurate Arctic-wide measurements, among which cloud coverage plays an important role. Different from cloud coverage measurements in other regions, similar scattering properties between clouds and ice- and snow- covered surfaces bring additional challenges. Therefore, the study aims to propose new operational Arctic cloud detection algorithms.

The data were collected with the help of the Multiangle Imaging SpectroRadiometer (MISR). The MISR sensors contain cameras viewing Earth scenes at nine different angles (70.5° Df, 60.0° Cf, 45.6° Bf, 26.1° Af in the forward direction, 0.0° An in the nadir direction, 26.1° Aa, 45.6° Ba, 60.0° Ca, 70.5° Da in the aft direction.) in four spectral bands (blue, green, red and near-infrared). There are 233 geographically distinct but overlapping MISR paths, paths are covered every 16 days, each path has 180 blocks, and each MISR pixel covers a 275m $\times$ 275m region on the ground. The vast amount of data produced by MISR also calls for new algorithms. There are two algorithms based on MISR used in other origins, while they both have difficulties in polar regions. The stereo-derived cloud mask(SDCM) compares retrieved object heights with the known terrain heights, and the angular signature cloud mask (ASCM) compares solar spectral reflctances between different view angles, which is called the band-differenced angular signature, but both of them can't detect low clouds. The study attempts to solve the problem by searching for surface pixels instead of cloudy ones.

Specifically, the data used in this study were collected from 10 MISR orbits of path 26 over the Arctic, northern Greenland, and BaffinBay from April 28 through September 19, 2022. After excluding data units of open water surfaces, the data contained 57 data units with 7114248 1.1-km resolution pixels with 36 radiation measurements for each pixel. Moreover, to evaluate the algorithms, one of the authors handlabeled the data, and the expert labels cover about 71.5% of the total valid pixels.

To conclude, the study presents new and operational algorithms for Arctic cloud detection and further climate study. It's shown that three physical features, the linear correlation of radiation measurements from different MISR view directions, the standard deviation of MISR nadir red radiation measurements within a small region, and a normalized difference angular index is enough to tell clouds from ice- and snow-covered surfaces.

More abstractly, the study demonstrates the great promise of statistical science. Statisticians can be directly involved in the data processing and closely collaborated with scientists from other fields, and work out innovative and effective solutions.

1.2 Labeled Maps

Table 1 reports the percentage of pixels for the three classes: cloudy, clear, and unlabeled.

Table 1: Summary of percentage of pixels.

	Cloudy	Clear	Unlabeled
Image 1	34.1%	37.3%	28.6%
Image 2	17.8%	43.8%	38.5%
Image 3	18.4%	29.3%	52.3%

Figure 1 suggests that no single pixels are labeled as cloud. In other words, pixels are spatially related to each other in a sense that the neighbour of a pixel labeled as cloud tend to be cloudy. As a result, the assumption that the samples are independent does not hold for the data.

Figure 1: Expert labels for the three images. White represents cloud; black represents unlabeled pixels; and grey represents no cloud.

1.3 EDA

To explore the pairwise relationship between features, we plot the correlation matrix for each image. The result for the first image is presented as a representative in Figure 2 because the three images share similar pairwise relationship between features.

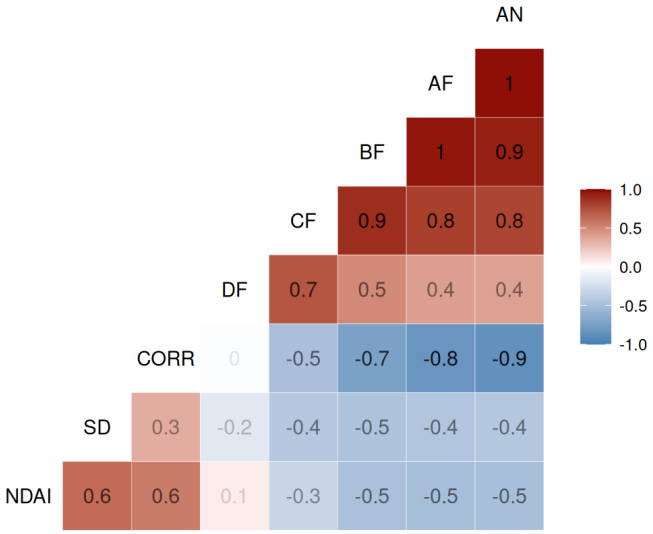

Figure 2: Pairwise relationship between features of image 1

Figure 2 suggests that CORR is positively related to the fifth MISR direction An. It is reasonable as the definition of CORR involves the radiance angle An. In addition, NDAI, the feature being used to detect low altitude clouds, is positively related to SD, the feature being used to identify smooth surfaces but the correlation is not so strong as that among radiance angles An, Af, Bf, and Cf. And the radiance angle Df has positive correlation with angle Cf.
The difference between cloud and no cloud regarding these features is presented in Figure 3. Notice that since the difference in boxplots is similar for the three images, we again select image 1 as the representative. High NDAI and SD correspond to cloud which is consistent with the conclusion in [SYCB08]. Since high CORR is expected to suggest either no cloud or low altitude cloud, the figure indicates the presence of low altitude cloud. As for the radiance angles, high values of An, Af, Bf are likely to relate to no cloud.

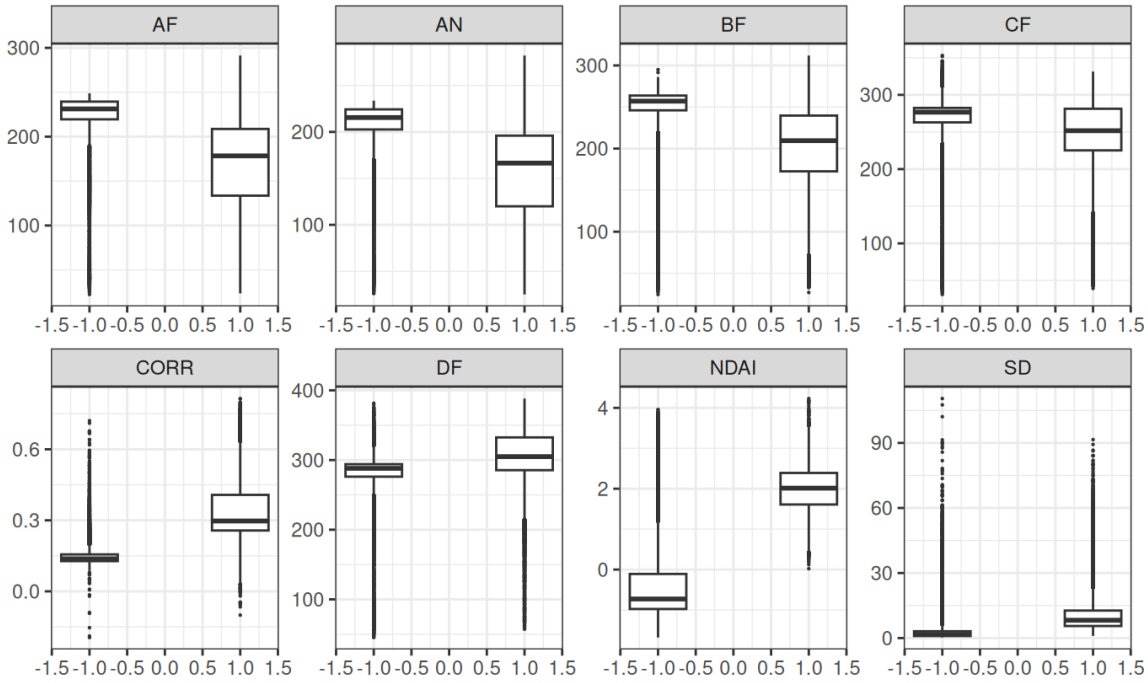

Figure 3: EDA: Difference between cloud and no cloud of image 1

2 Preparation

2.1 Data Split

We first use the systematic assignment [RBC+17] to split each image and the pattern for the split in image 1 is shown in Figure 4. The systematic assignment outperforms the random split especially when the data is spatially correlated because it ensures lower dissimilarity between folds. For the cloud data, due to the aforementioned spatial correlation between pixels in each image, the systematic assignment can effectively reduce the dependence between the training set and the validation set. Among all the 12 blocks of the three images, 2 blocks are randomly selected as the test set. As for the remaining 10 blocks, one is used for validation and the rest for training in turn. The following section uses 10-fold cross validation to assess the fit of models. We claim that such split of validation set and training set is reasonable because the number of folds is usually between 5 and 10.

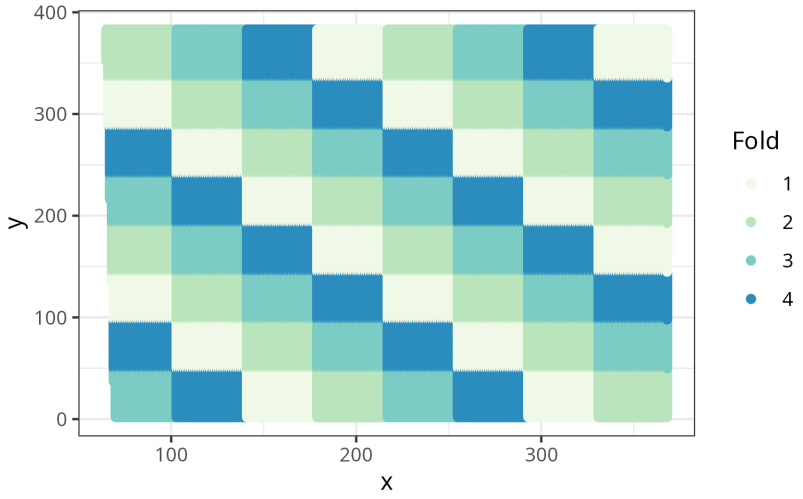

Figure 4: Image 1 split by systematic assignment

The second method separates the training set from the validation set and the test set by creating a buffer. The cross in the middle of the image is used as the test set and the remaining 12 corners in the three images are used for cross validation. Specifically, one block is for validation and the remaining 11 blocks for training in turn. Although the pixels in the buffer area are not used for either training or testing, the width of the buffer area can be modified such that the fraction of the unused data can be controlled. One advantage of this split over the first one is that it generates spatially separated folds and ensures that no validation data or test data abuts the training data.

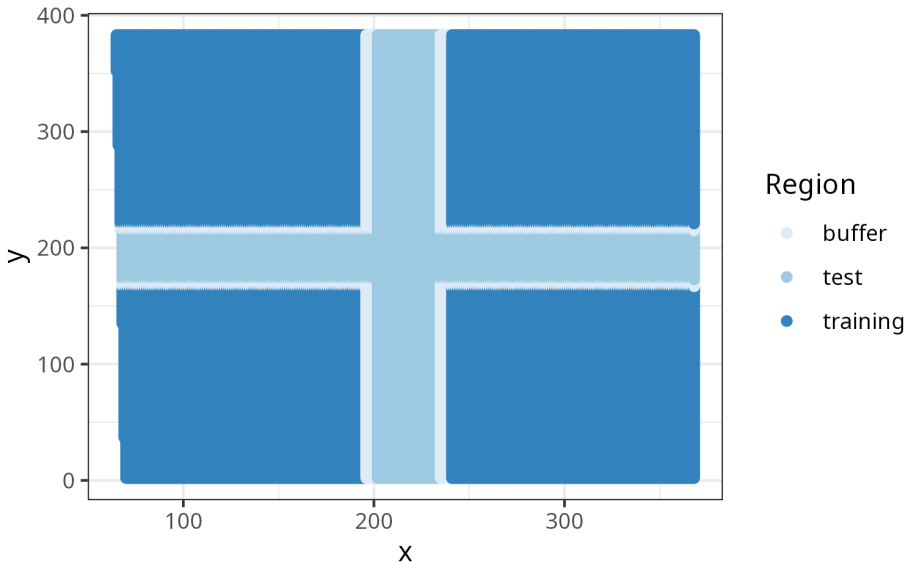

Figure 5: Image 1 split by creating the buffer

2.2 Baseline

To ensure the ensuing classifier in section 3 is not trivial, we first examine the accuracy of a trivial classifier that assumes all labels to be no cloud on the validation set and test set for both splits. The accuracy is summarized in Table 2. The accuracy of the trivial classifier is mainly determined by the proportion of pixels labeled as no cloud.

Table 2: Accuracy of a trivial classifier

Split	Set	Accuracy
Systematic	Validation	46.4%
Systematic	Test	57.5%
Buffer	Validation	47.0%
Buffer	Test	59.1%

2.3 First Order Importance

Since we assume the expert labels as the truth, we define the *best* features to have high correlation with the expert labels and have large difference between the cloudy and clear groups so that the *best* features achieve better classification performance. As shown in Table 3, we calculate the correlation between the eight features and the label. We can see that features of NDAI, SD, CORR, Radiance angle BF, Radiance angle AF, and Radiance angle AN have a high correlation. Moreover, combining the box-plots in Figure 3 and considering that information from radiance angle is integrated, we choose NDAI, SD and CORR as the three *best* features.

Table 3: Correlation between features and label

Feature	Correlation
NDAI	0.617
SD	0.295
CORR	0.444
Radiance angle DF	0.007
Radiance angle CF	-0.208
Radiance angle BF	-0.338
Radiance angle AF	-0.390
Radiance angle AN	-0.389

2.4 CVmaster Function

In this part, we write a cross validation function in CVmaster.R. We take classifiers, features, labels, K-folds, loss functions as inputs, and output each fold's results of loss functions as a matrix. Classifiers can be *logistic*, *LDA*, *QDA*, *Naive Bayes*, *knn*, *rf*, and *adaboost* while loss functions can be *accuracy*. To take data split method into consideration, we put the number of data split fold into the input. When K becomes larger, we can extend our data split to get more blocks for k-folds cross validation. In our study, we choose $K = 10$.

3 Modeling

3.1 Accuracy

In this section, seven classification methods are applied and compared including logistic regression, linear discriminant analysis (LDA), quadratic discriminant analysis (QDA), Naive Bayes, k-nearest neighbors algorithm (kNN), random forest, and Adaboost. We use 10-fold cross validation to assess the fit of each classification and compute the corresponding 10-fold cross validation loss on the training set based on the CVmaster function. The accuracy across the 10 folds split by the systematic assignment is summarized in Table 4. Table 5 shows the accuracy across folds for the second split and Table 6 displays the test accuracy for both splits.

Table 4: Accuracy across folds by first split

Model	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Average
Logistic	0.69	0.87	0.95	0.83	0.83	0.90	0.95	0.92	0.88	0.98	0.88
LDA	0.82	0.90	0.88	0.95	0.92	0.70	0.98	0.84	0.95	0.87	0.88
QDA	0.87	0.91	0.94	0.88	0.81	0.98	0.86	0.96	0.71	0.88	0.88
Naive Bayes	0.67	0.84	0.85	0.90	0.87	0.89	0.87	0.84	0.89	0.77	0.84
KNN	0.90	0.88	0.94	0.86	0.85	0.92	0.89	0.85	0.97	0.78	0.88
Random Forest	0.93	0.99	0.96	0.95	0.81	0.85	0.94	0.90	0.94	0.90	0.92
Adaboost	0.79	0.88	0.91	0.89	0.88	0.94	0.99	0.97	0.87	0.90	0.90

Table 5: Accuracy across folds by second split

Model	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Average
Logistic	0.77	0.96	0.91	0.81	0.96	0.89	0.37	0.99	0.88	0.68	0.82
LDA	0.96	0.41	0.99	0.87	0.99	0.90	0.81	0.90	0.77	0.96	0.85
QDA	0.99	0.98	0.74	0.64	0.85	0.97	0.99	0.99	0.83	0.49	0.85
Naive Bayes	0.92	0.99	0.67	0.70	0.70	0.83	0.83	0.97	0.83	0.97	0.84
KNN	0.59	0.97	0.73	0.98	0.94	0.94	0.73	0.70	0.72	0.43	0.77
Random Forest	0.99	0.94	0.82	0.69	0.76	0.73	0.91	0.60	0.98	0.96	0.84
Adaboost	0.93	0.99	0.71	0.78	0.98	0.95	0.42	0.94	0.88	0.99	0.86

Table 6: Test accuracy by two data split methods

Model	Systematic assignment	Creating the buffer
Logistic	0.88	0.90
LDA	0.88	0.91
QDA	0.89	0.88
Naive Bayes	0.85	0.82
KNN	0.95	0.93
Random Forest	0.96	0.96
Adaboost	0.95	0.94

Table 4 suggests that random forest outperforms the other classification methods with an average accuracy of 92% across all the folds, closely followed by Adaboost with an average accuracy of 91%. LDA, KNN, and QDA have similar average accuracy that are around 88% while Naive Bayes presents the lowest average accuracy that is 84%. The accuracy across folds for the second split is shown in Table 5. Adaboost achieves the best accuracy across folds on average while KNN attains the lowest average accuracy. The accuracy of other models falls between 0.82 to 0.85. Table 6 shows that random forest, Adaboost, KNN are the top three models in terms of test accuracy under both data split methods.
We further examine the assumptions and properties of each classification.

Logistic regression: The basic logistic regression assumes that the response is binary, which is consistent with this case because the label only contains cloud and no cloud after the unlabeled data is removed. It also assumes that the observations are independent of each other, which is not satisfied in this case since the data are spatially correlated. The assumption that there is no multi-collinearity among features also does not hold because the features are found to be correlated with each other in section 1.
LDA: LDA assumes that the data within each label is normal distributed with a common covariance matrix. Nonetheless, due to the EDA in section 1, we conclude that this assumption is not satisfied.
QDA: QDA assumes that the data within each label is normal distributed but each label can have a different covariance matrix, which is also not satisfied.
Naive Bayes: Naive Bayes assumes that the predictors are independent within in each label. However, since the predictors are correlated as mentioned in section 1, such assumption does not hold in this setting.
KNN: KNN is a model-free classification method which only assumes that close data points have similar labels. It has no assumptions on the distribution of the data.
Random Forest and Adaboost: Random forest decorrelates the decision trees by randomly choosing a subset of predictors for each split and it makes no assumptions on the distribution of the data. Adaboost makes use of base learners and reweight each sample at each iteration to build the tree and has no assumptions on the distribution of the data.

3.2 ROC Curve

In this part, we use ROC curves to compare the above models. As shown in the Figure 6, ROC curves present models' performance at all classification thresholds. The closer the curve is to the top-left corner, the larger the AUC, and the better the model. We conclude that the random forest model performs the best. Moreover, we choose cut off values that are closest to the top-left corner and have better performance both in specificity and sensitivity, which are marked in the figure.

3.3 Bonus

In this part, we use confusion matrices and F1 score to compare the above models. To get better performance of each model, we use the corresponding cut off values we get from ROC curves. The results of confusion matrices are shown in Figure 7. We can conclude that all the seven models have a relatively low error rate, and false positive happens a bit more often that false negative. Among all the models, random forest and Adaboost stand out again.
F1 score is calculated based on the precision and recall and hence combines both False Positive and False negative. The F1 score for each model trained from the first split is summarized in Table 7. Combining the accuracies, confusion matrices and the F1 score, we define random forest to be the best model.

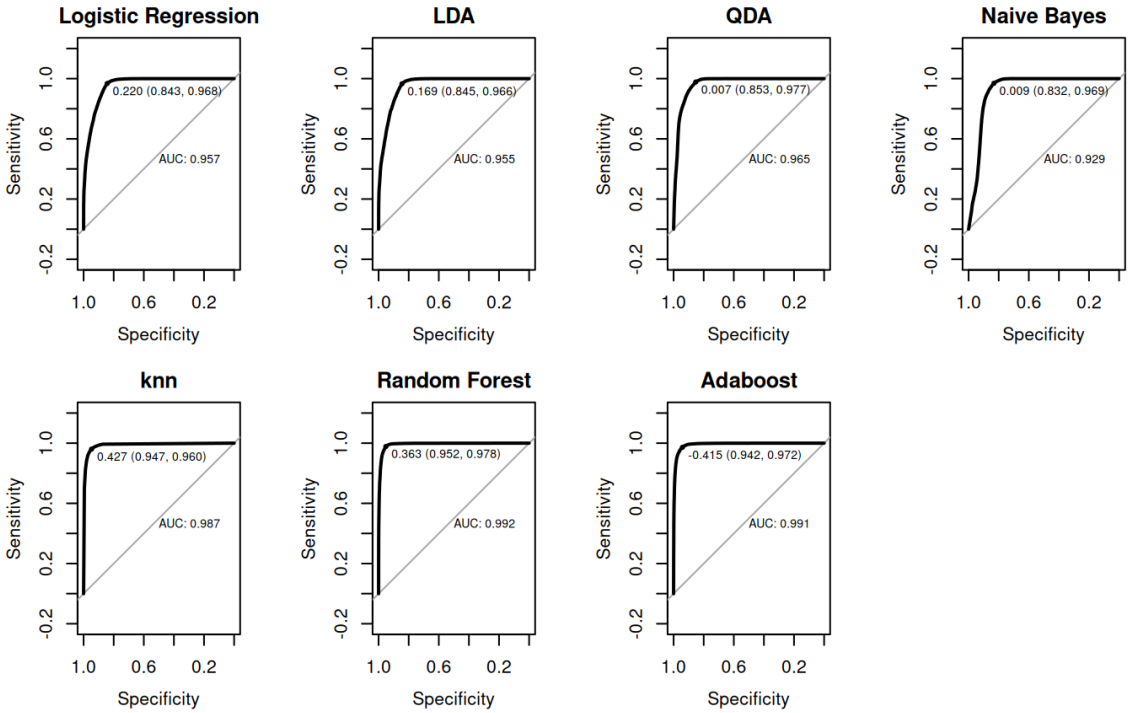

Figure 6: ROC curve by first split method

Logistic	Actually 0	Actually 1
Predicted 0	19440	523
Predicted 1	1499	15883

LDA	Actually 0	Actually 1
Predicted 0	19564	532
Predicted 1	1375	15874

QDA	Actually 0	Actually 1
Predicted 0	19612	289
Predicted 1	1327	16117

Naïve Bayes	Actually 0	Actually 1
Predicted 0	19508	382
Predicted 1	1431	16024

KNN	Actually 0	Actually 1
Predicted 0	19285	410
Predicted 1	1430	14907

Radom Forest	Actually 0	Actually 1
Predicted 0	19932	367
Predicted 1	1007	16039

Adaboost	Actually 0	Actually 1
Predicted 0	19717	461
Predicted 1	1222	15945

Figure 7: Confusion matrix by first split method

Table 7: F1 score by first split method

Model	F1 score
Logistic	0.730
LDA	0.903
QDA	0.912
Naive Bayes	0.897
KNN	0.958
Random Forest	0.969
Adaboost	0.964

4 Diagnostics

4.1 In-depth Analysis of Random Forest

In this section, we conduct further diagnostics for random forest. To start with, Figure 8 displays the tree as a sample to visualize the model. The tree has 3 leaves in total. As for the bottom left leaf, 55% of the data is in the leaf among which 99.8% are no cloud. Such result indicates a relatively pure leaf. It can be seen from the tree that NDAI and angle AN are used to split the node. The tree first

split the root node based on whether NDAI is less than 0.82 and then split the left branch based on whether the angle AN is greater than 169.

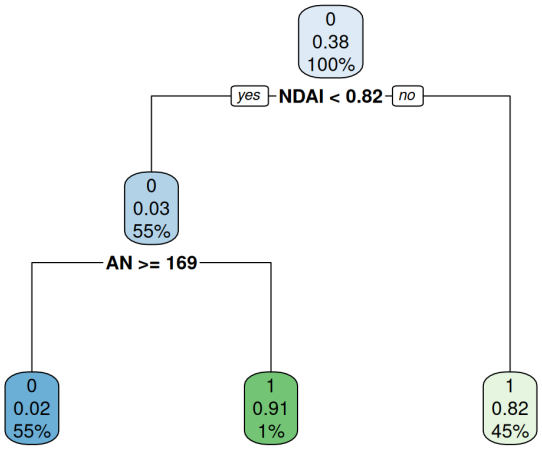

Figure 8: Tree generated from training data by the first split. 0: no cloud, 1: cloud.

Next, we explore the performance of random forest model under different hyper-parameters. Figure 9 shows accuracy as a function of the number of trees, and each colored line corresponds to a different value of m, the number of predictors available for splitting at each interior tree node. Random forests perform better than bagging, and the number of trees larger than 200 is better.

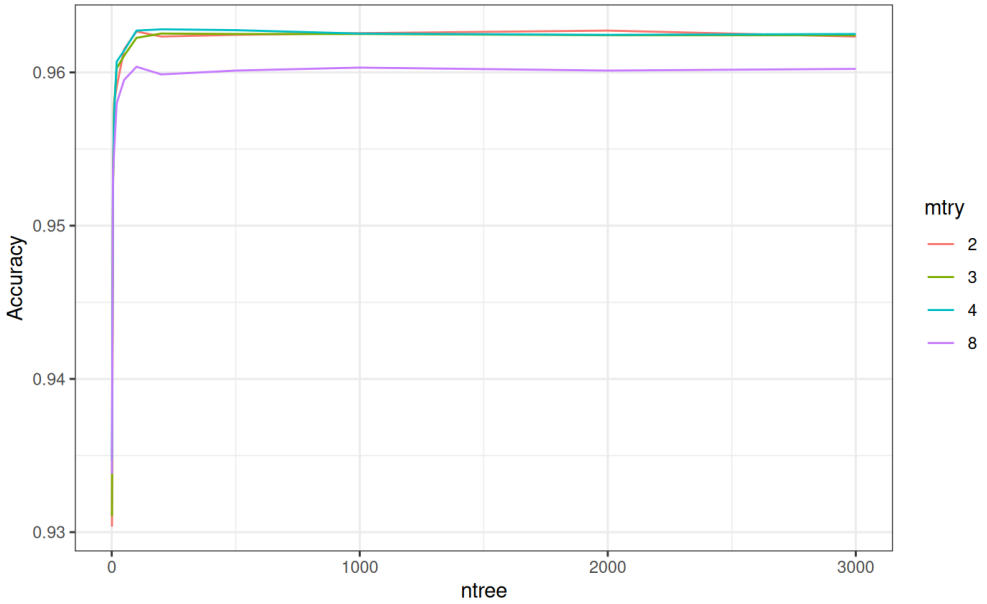

Figure 9: Accuracy under different hyper-parameters by first split method

Third, we check the variable importance. According to the results under different parameters, we choose mtry=4, ntree=500. Both Gini importance(mean decrease in impurity, MDI) and permutation importance(mean decrease in accuracy, MDA) are shown in Figure 10. We can see that NDAI is absolutely the most important.

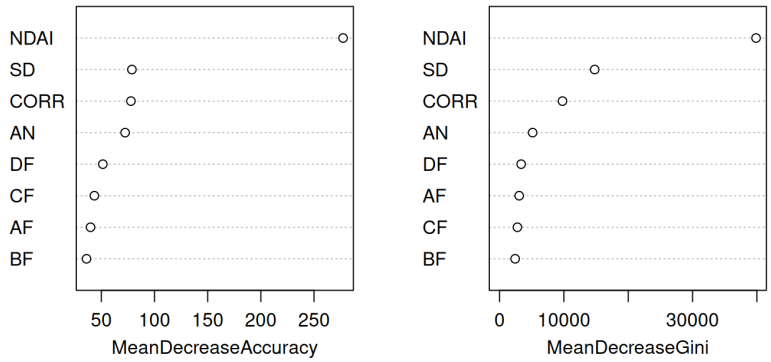

Figure 10: Variable importance by first split method

4.2 Patterns in the Misclassification Error

To explore the pattern in the misclassification error, we compare the predicted results with the real labels and visualize the misclassified pixels in Figure 11.

We summarize two patterns from the figure. First, it can be seen that most misclassified pixels are near the boundary of cloud and no cloud, and the predicted probabilities in these areas are around 0.5. It is reasonable and consistent with [SYCB08] because the pixels near the boundary is usually partly cloudy and can't be simply labeled as cloudy or clear. We deal with the problem in the next part. Second, the number of clear pixels mislabeled as cloudy is larger than that of cloudy pixels mislabeled as clear. The ROC curve and confusion matrix also tell that false positive rate is higher than false negative rate.

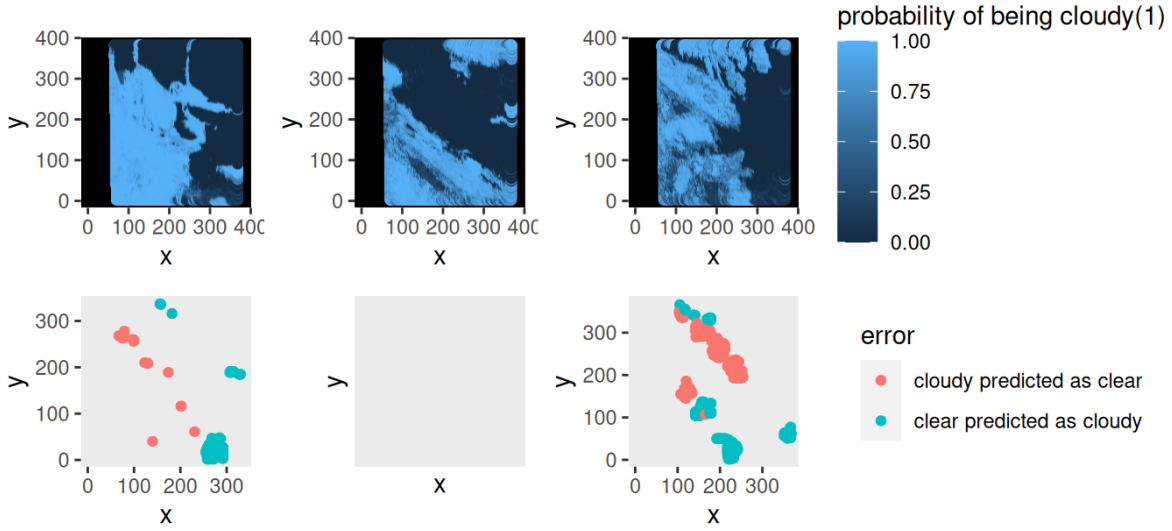

Figure 11: Predicted results and misclassification by first split method

4.3 A Better Classifier

Based on previous patterns, a better classifier should improve the classification performance at the boundary of cloud and no cloud. We could first apply random forest and do cross validation after splitting the images by systematic assignment or creating a buffer. Our previous analysis shows that random forest has satisfying performance on pixels other than the boundary. To improve the boundary performance, multiple classification methods can be applied to help decide the prediction. For instance, we can apply LDA, QDA, and logistic regression on these pixels and report the proportion of cloudy pixels for each model. The final prediction can be based on the majority vote. We put forward this idea inspired by ELCM-QDA algorithm in [SYCB08].

We state our model perform pretty well for the future unlabeled data based on the previous analysis and diagnostics. The splitting method effectively preserve the spatial structure while reduce the

dependence between the training set and the validation set. The model turns out to achieve high accuracy on the test set and the possible misclassifcation issue on the boundary pixels can be improved by the aforementioned method.

4.4 Modify the Way of Splitting the Data

We change the splitting method and present the results as follows. For the second split, the tree is visualized in Figure 12 which has 6 leaves in total. In general, the tree is deeper than the first tree and uses more features. Specifically, NDAI, CORR, SD, and angle AN are used to split the node. The threshold for NDAI at the root node and the threshold for angle AN is similar to that in Figure 8.

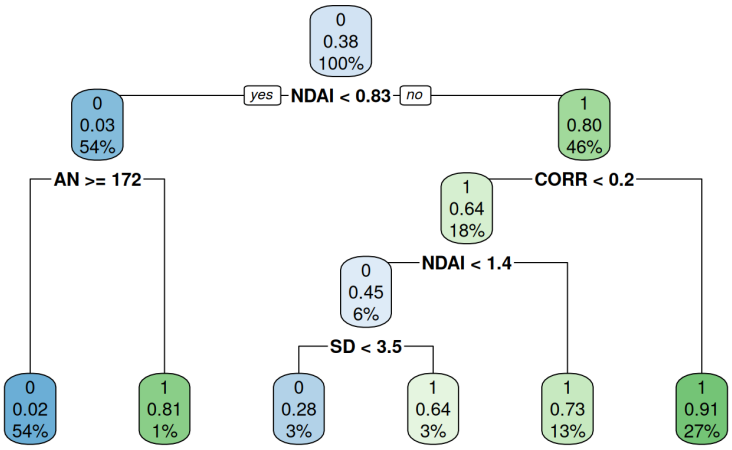

Figure 12: Tree generated from training data by the second split

To see how the model performs for the second split, we tune the number of trees and visualize the convergence of the accuracy in Figure 13. Similar to Figure 9, Figure 13 shows that for all colored line which represents a different value of m in the random forest, the accuracy converges to a satisfying level after the number of trees is larger than 200. For the second splitting data, we can also conclude that the random forest has better performance than bagging in terms of accuracy.

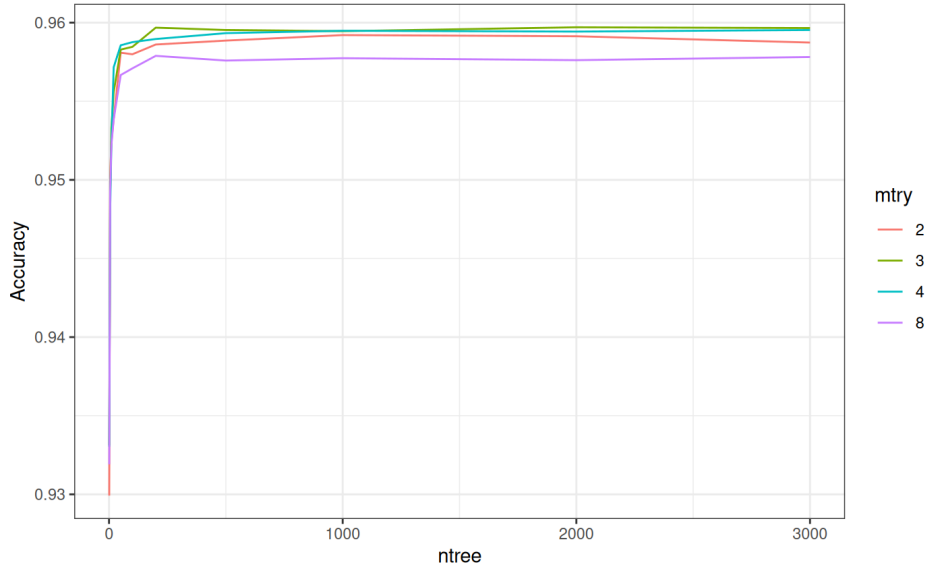

Figure 13: Accuracy under different hyper-parameters by second split method

As for the variable importance, both Figure 10 and Figure 14 give the same conclusion that the top three important variables are NDAI, CORR, and SD among which NDAI is always the most important

feature.

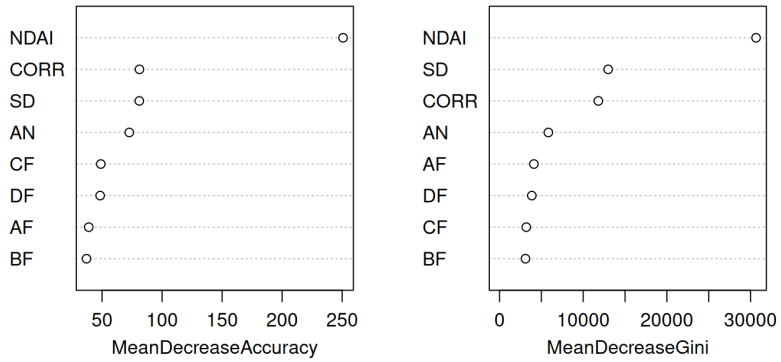

Figure 14: Variable importance by second split method

To compare the pattern in the misclassification errors between the two splits, we visualize the misclassified pixels for the second split in Figure 15. In brief, the aforementioned pattern in section 4.2 still holds. Hence, the improved method that we have proposed in section 4.3 is still suitable.

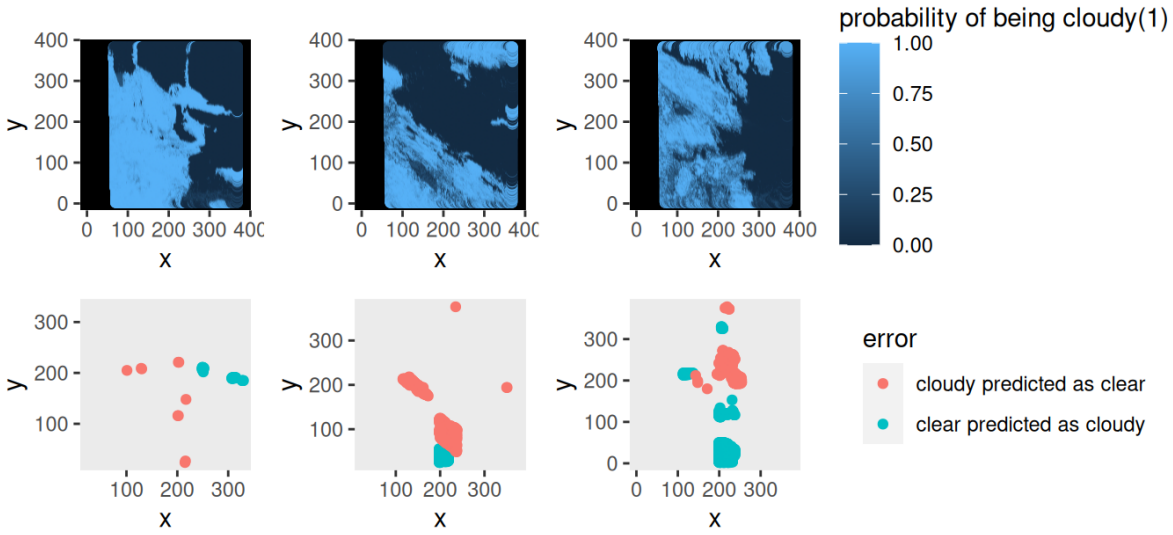

Figure 15: Predicted results and misclassification by second split method

4.5 Conclusion

In summary, we apply seven classification methods for two splitting methods and assess the model performance in terms of accuracy and f1 score. It turns out that the random forest is the best model. As for the diagnostics, we conduct an in-depth analysis of the random forest model under both data split methods. In particular, we tune the number of trees to see the convergence of the accuracy, visualize the variable importance and plot the misclassified pixels to explore the underlying pattern. To conclude, the random forest model has high accuracy, sensitivity and specificity and is a good classifier for Arctic cloud detection. Results for the two split methods are similar, which indicates that random forest have a robust satisfactory performance. Besides, we identify a pattern that misclassified pixels tend to appear at the boundary of cloud and no-cloud. Based on such finding, an improved classifier is proposed which will conduct a detailed classification on misclassified pixels by utilizing multiple models and giving a more reliable classification.

5 Reproducibility

To reproduce our results, we provide the following files:

- data: the three images obtained by the MISR sensor
- PROJ2-writeup.tex: the raw Latex used to generate the report
- PROJ2-code.rmd: the code written for all parts. One can get all the figures and plots by running the code chunk by chunk. In detail, part 1 is to load data, plot labeled maps and perform EDA, part 2 is to do data split, calculate baseline accuracy and find the best features, part 3 is to compare different models using multiple ways, and part 4 is to further diagnose the random forest model.
- CVmaster.R: the generic cross validation function that outputs the K-fold corss validation loss on the training set.

6 Acknowledgement

This project was motivated by [SYCB08] and the split methods were inspired by [RBC+17]. The code and write up were jointly completed by both Huiying Lin and Yanjiao Yang. For the first section of this paper, Huiying finished the summary of the original paper and Yanjiao conducted the EDA. As for the following sections, we collaborated and discussed about each sub-question together and refer to *An introduction to statistical learning with applications in R* for the classification methods.

References

[RBC+17] David R Roberts, Volker Bahn, Simone Ciuti, Mark S Boyce, Jane Elith, Gurutzeta Guillera-Arroita, Severin Hauenstein, José J Lahoz-Monfort, Boris Schröder, Wilfried Thuiller, et al. Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure. *Ecography*, 40(8):913–929, 2017.

[SYCB08] Tao Shi, Bin Yu, Eugene E Clothiaux, and Amy J Braverman. Daytime arctic cloud detection based on multi-angle satellite data with case studies. *Journal of the American Statistical Association*, 103(482):584–593, 2008.