Daytime Arctic Cloud Classification Methods

Jessica Cherny: 26133071, Yiming Shi: 3031922583

I. DATA COLLECTION AND EXPLORATION

The paper explores two new algorithms for arctic cloud detection using Multiangle Imaging SpectroRadiometer (MISR) imagery because arctic cloud detection as it is currently is difficult and unreliable. As the abstract says, the goal is to "identify cloud-free surface pixels in the imagery instead of cloudy pixels as in the existing MISR operational algorithms". The data consists of 3 images of three data units collected from MISR blocks 20–22 over three consecutive orbits (i.e., 13257, 13490, and 13723). The features in this dataset include: y coordinates of the pixel, x coordinates of the pixel, expert label (+1 = cloud, -1 = not cloud, 0 unlabeled), NDAI (normalized difference angular index that characterizes the changes in a scene with changes in the MISR view direction), SD (the standard deviation of MISR nadir camera pixel values across a scene), CORR (the correlation of MISR images of the same scene from different MISR viewing directions), Radiance angle DF (MISR sensor camera angle at 70.5 degrees), Radiance angle CF (MISR sensor camera angle at 60 degrees), Radiance angle BF (MISR sensor camera angle at 45.6 degrees), Radiance angle AF (MISR sensor camera angle at 26.1 degrees), and Radiance angle AN (MISR sensor camera angle at 0 degrees).

The image data was collected over 144 days from April 28, 2002 to September 19, 2002 from 10 MISR orbits of path 26 over the Arctic, northern Greenland, and Baffin Bay. Six data units (MISR blocks 11–13, 14–16, 17–19, 20–22, 23–25, and 26–28) from each orbit were studied. Experts hand-labeled the image data as cloudy, not cloudy, or left it unlabeled if unsure. After the expert labeled a patch of image pixels, NASA's misrdump and misrlearn algorithms were employed to label the pixels from the MISR nadir camera as clear or cloudy. The paper concludes that the three most important physical features in predicting cloudy or not cloudy images are CORR, SD, and NDAI. The ELCM algorithm reliant on these three features is more accurate than the existing MISR algorithms for cloud detection in the Arctic. Moreover, QDA aided ELCM provided performance better than that of a more complicated classifier, like SVM. Having a better classifier of cloudy/non-cloudy images is important because more reliable labeling will lead to more accurate climate model simulations which will be useful when studying the effects of climate change in the Arctic brought on by increasing concentrations of atmospheric carbon dioxide.

A summary of the class distribution of the dataset is given in the table below. Both images 1 and 3 have a class imbalance problem, while image 2 has a more balanced representation from all classes.

	Image 1	Image 2	Image 3	Combined
Not Cloud	43.78%	37.25%	29.29%	36.78%
Unlabeled	38.45%	28.64%	52.27%	39.79%
Cloud	17.77%	34.11%	18.44%	23.43%

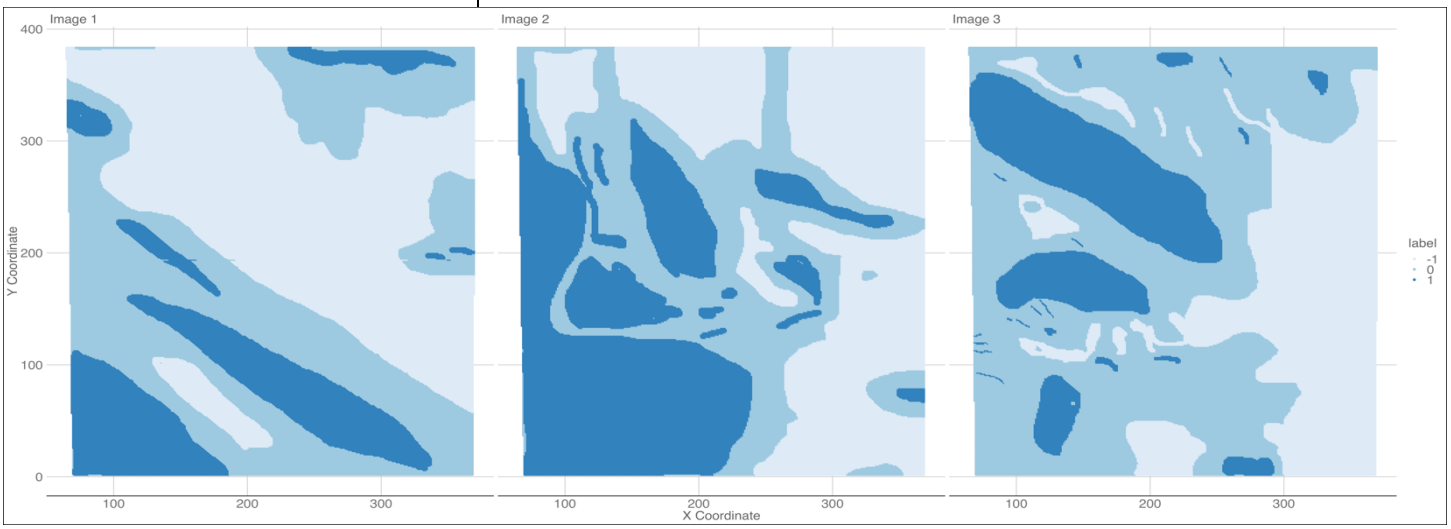
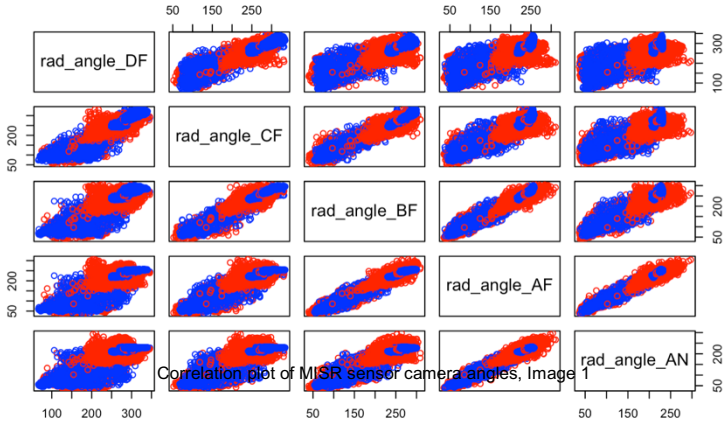
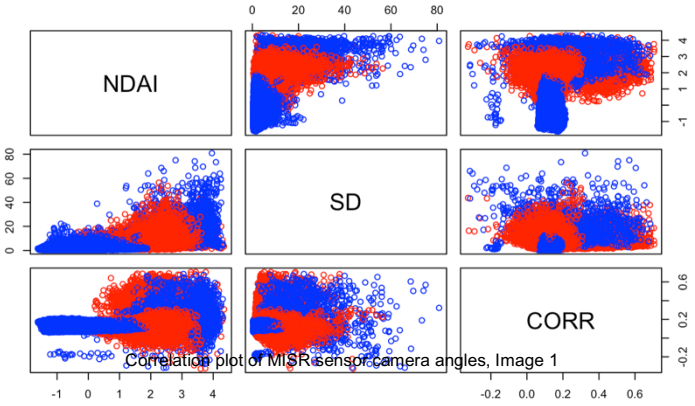


Image 3 has the highest percentage of unlabeled data, which may lead to lower classification accuracy later on when we disregard all the unlabeled entries and leave only the cloud/not-cloud labels for training and testing. As can be seen in the figure above, there seems to be a trend occurring: there are many unlabeled regions in between the expert-labeled regions. We can interpret this as the experts being unconfident in labeling anything by the boundary of cloudy and clear skies. Furthermore, we can say that the i.i.d. assumption is not justified because adjacent pixels cannot be independent from each other: a cloudy pixel will likely be part of an overall greater cloudy region, just like a non-cloudy pixel will most likely be part of a greater non-cloudy sky region.

We will begin our EDA by summarizing pairwise relationships between the features themselves and the expert labels. Pairwise scatterplots among the MISR sensor camera angles on Image 1 reveal that these angles are highly correlated with each other. This implies that including only one angle in a classification model will be necessary since including the other four angles will lead to issues due to multicollinearity. We get similar plots and conclusions when we plot image 2 and image 3, so we will omit them from this report.



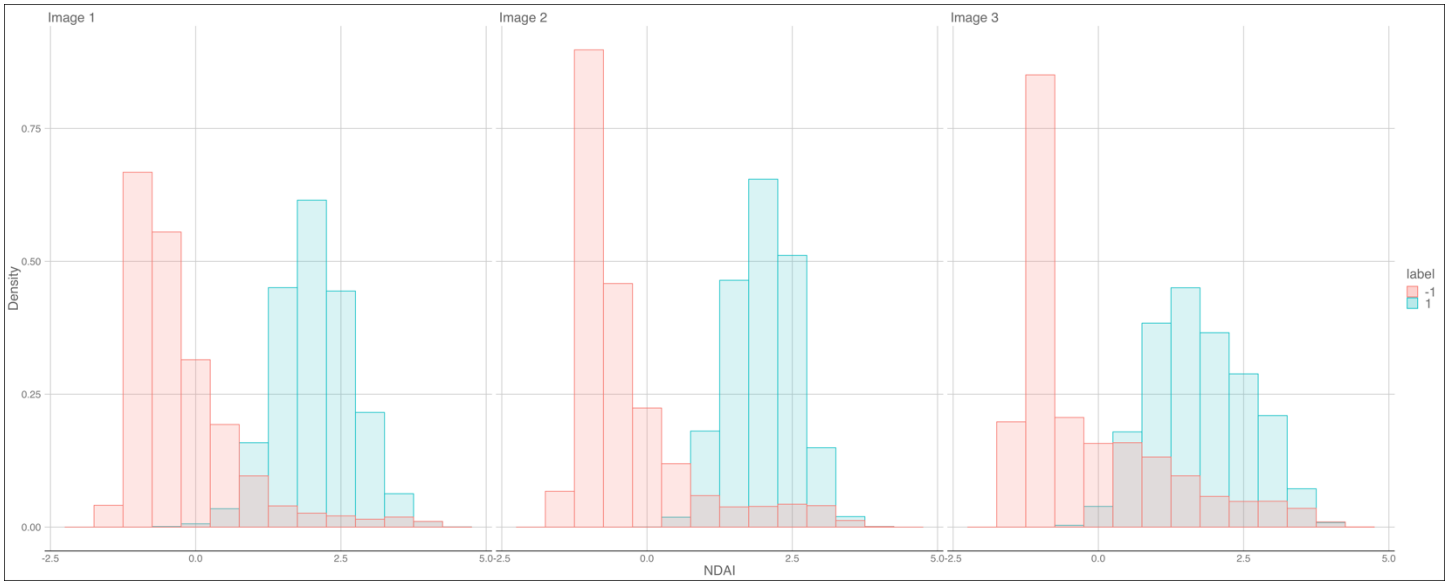
Correlation plot of MISR sensor camera angles, Image 1



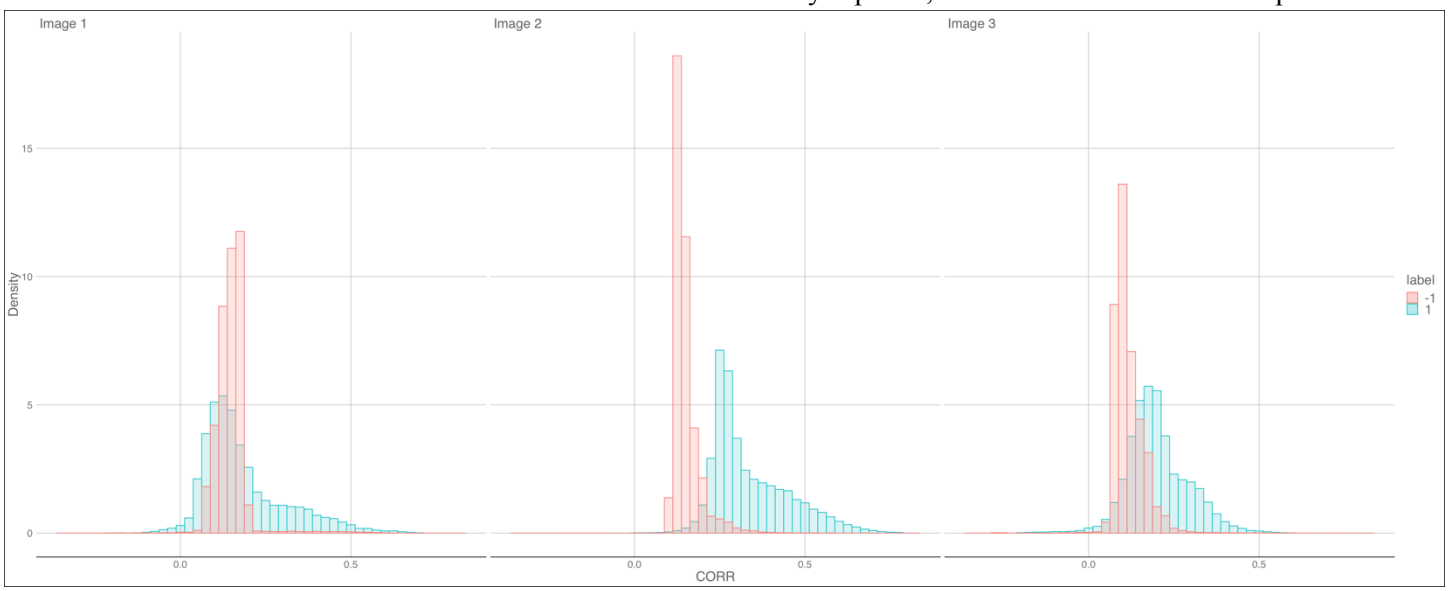
Correlation plot of MISR sensor camera angles, Image 1

Pairwise scatterplots among NDAI, SD, and CORR reveal that these features are relatively uncorrelated with each other. We can see SD and NDAI are somewhat correlated with each other but not too much. We get similar plots and conclusions when we plot image 2 and image 3, so we will omit them from this report.

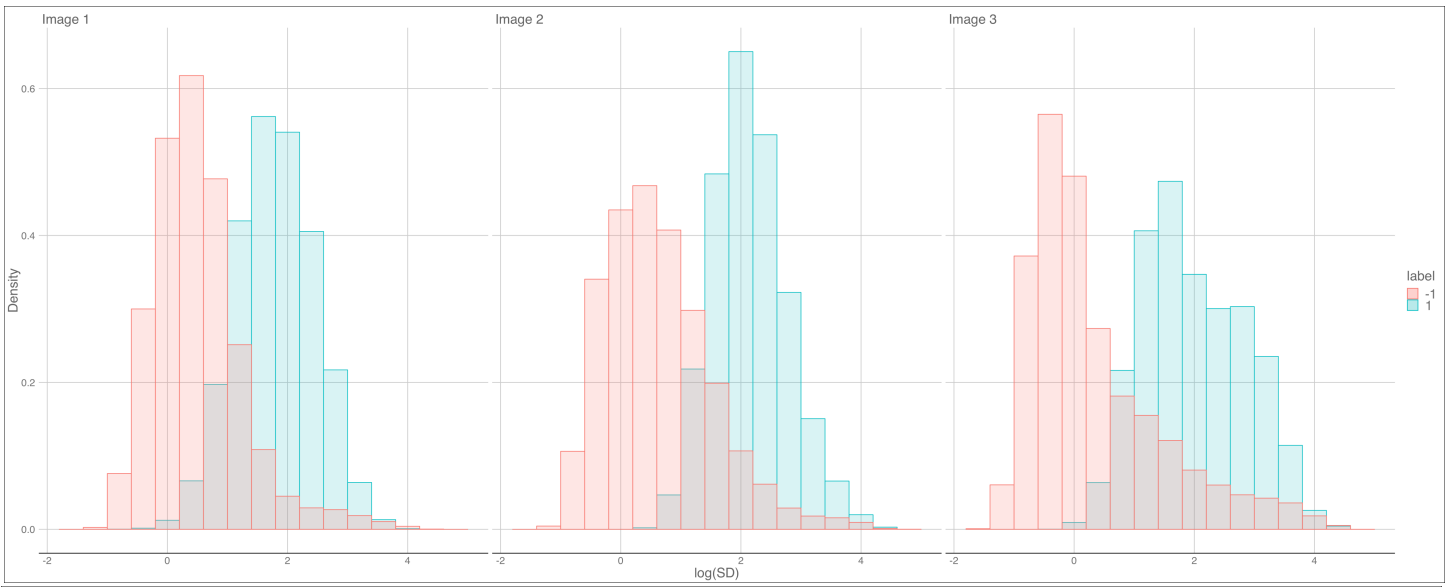
Now we will explore the relationship between the expert labels with the individual features through other visual and quantitative means. We will start by graphing the histogram distributions of important features like NDAI, CORR, SD, and Radiance Angle AN separated by label type (cloud vs. not cloud). From the figure below, we can see that the distribution of NDAI over cloudy and non-cloudy images is nicely separable here and will therefore be a good predictor of image labels.

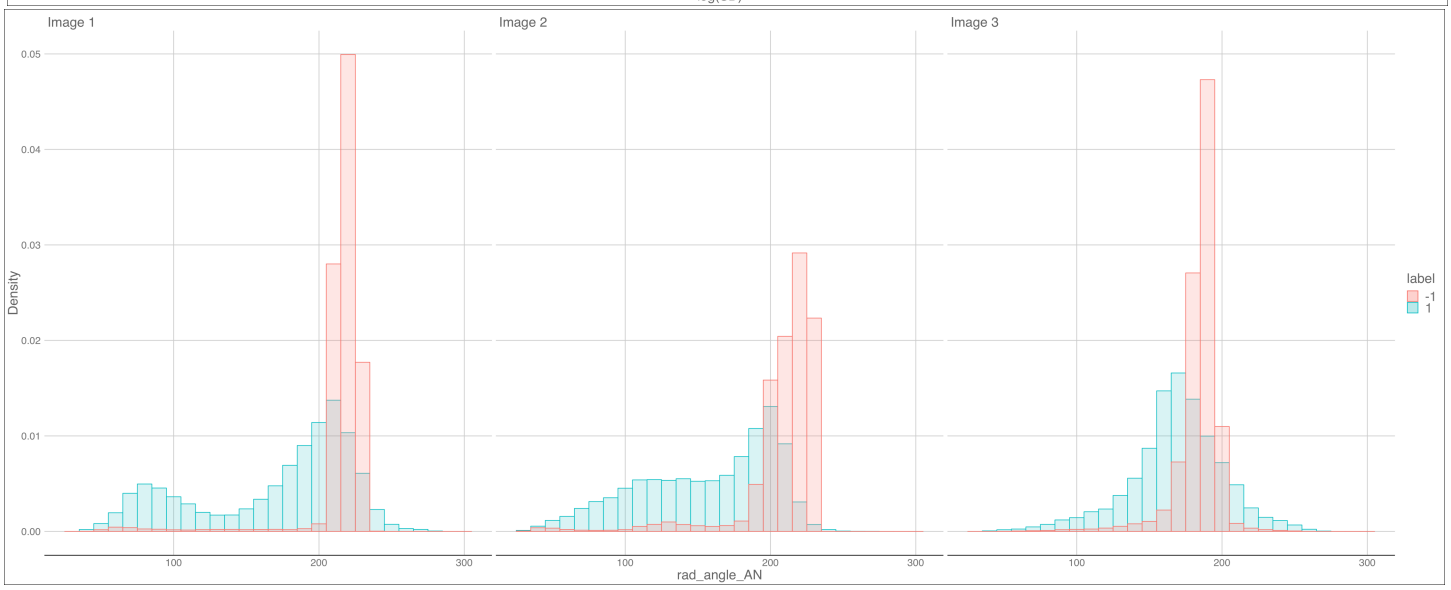


It's harder to clearly separate the distributions of CORR, as can be seen in the figure below. We can see that there may be a cutoff around CORR = 0.2 where the two distributions are relatively separate, but there is still much overlap.



We can see that the distribution of log(SD) over cloud and not-cloud images is nicely separable here (although not as nicely separable as NDAI) and will therefore be a good predictor of image labels.





It's harder to clearly separate the distributions of rad_angle_AN. We can see above that there may be a cutoff around 250 where the two distributions are relatively separate, but there is still much overlap. We decided to model the distributions of only one angle since angle features are highly correlated with each other and therefore produce highly similar histograms as well. Because Radiance Angle AN feature distribution is not very easily distinguishable, it might not be too important of a feature, but we will include it nonetheless.

II. PREPARATION

Since there are two important aspects of these images, we proposed two splitting methods: temporal split and spatial split. In the temporal split, we keep spatial features the same and split according to time. Following that, we assigned the first two images to be the training set, and equally divided image 3 to be the validation and test set. In the spatial split, we rectify the violation of i.i.d assumption: by grouping the pixels into blocks, we can roughly treat the blocks as i.i.d since points not on the perimeter of the blocks will be approximately independent of those in another block. Then we randomly assigned 70% of blocks from all three images to be the training set, 10% to be validation set, and 20% to be the test set. However, as we proceed with our investigation, we realized a fatal mistake in our temporal split: data points from the same time period are also dependent in the time dimension, which means predicting on different images without grouping the pixels might lead to a much lower accuracy than desired. This conjecture was confirmed later, when we saw that the CV accuracy and the test accuracy of the temporal split underperform compared to spatial split. Hence, we devised a new splitting method similar to the spatial split, where we deal with spatial and temporal dependence at the same time: we keep the block structure from the spatial split, but train a separate model on each image and predict using that model. 70% of the blocks of each image is assigned as the training set individually, 10% for validation set and 20% for test set. The accuracies for models trained on each of these images are reported separately later.

We considered a trivial classifier, one that predicts -1 (cloud-free) for all entries, for baseline comparison to make sure the classification task here is not trivial. Such a trivial classifier will have high average accuracy if the dataset has largely imbalanced classes with many more observations of not cloud than cloud, or if we are confident that future data will mostly be non-cloud, such as images from the Sahara Desert in Africa. Validation and test accuracies for each splitting method is reported in the table below, where accuracy is calculated after filtering out data points not labeled by human experts.

Accuracy \ Split	Temporal	Spatial	Revised: Image 1	Revised: Image 2	Revised: Image 3
Validation	61.36%	59.66%	69.28%	49.43%	62.02%
Test	61.39%	61.87%	74.91%	51.92%	60.59%

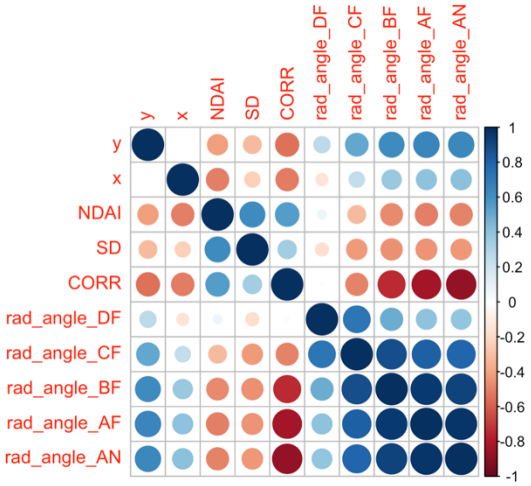
For the first order importance, the 3 best features we picked are SD, NDAI, and CORR. We can define the best feature criteria to imply features that help distinguish between the cloud and not-cloud classes easily. This means that when we map the histogram of a variable colored by the two different classes, there is a clear separation (two different distributions). If we look at the histograms from the last section, we can see the features SD, NDAI, CORR are (for the most part) non-overlapping. We have to set a threshold rule and see the accuracy in classification of that feature. (i.e. if NDAI > 1 then classify as cloud and if NDAI < 1 classify as non-cloud). We then see the area under the curve that overlaps. The accuracies of the top 3 features are below. By this rough approximation of classification, we can see that NDAI and SD classify labels well, while CORR does moderately so.

	NDAI	CORR	SD
Image 1	91.40%	50.20%	80.22%
Image 2	90.08%	69.83%	70.45%
Image 3	82.05%	81.26%	77.36%

We can also run logistic regression and see the importance of each feature in classifying labels by seeing the absolute value of each coefficient. Weights for all images are reported in the table below.

	NDAI	log(SD)	CORR	rad_angle_AN	rad_angle_DF	rad_angle_CF	rad_angle_BF	rad_angle_AF
Image 1	128.48	52.27	1.06	7.68	1.87	11.70	7.26	2.03
Image 2	87.75	31.74	47.47	40.07	47.81	11.46	5.82	23.99
Image 3	64.89	35.94	43.33	8.21	17.36	19.02	9.50	2.15

For image 1, the more important features are NDAI, log(SD), and rad_angle_CF; for image 2, they are NDAI, rad_angle_CF, CORR, and log(SD); while for image 3 they are NDAI, CORR, and log(SD). It seems that from this analysis NDAI, log(SD), rad_angle_CF, and CORR are the most important features. This finding aligns with the paper in confirming that NDAI, SD, and CORR are important features in classification.



Correlation Plot of Features of Image 2

We can see more visual evidence from the correlation plot above that the radiance angles are highly correlated with each other and are therefore redundant in importance. NDAI, SD, and CORR are good features because they are relatively uncorrelated with other features. This correlation plot used image 2 as an example, when in exploration we found correlation plots for all three images are very similar, so we only included one for simplicity.

III. MODELING

We started out with the idea of trying out logistic regression, decision trees, random forest, KNN, LDA, QDA, and SVM as classifiers on our data but decided to narrow down our classification methods to just logistic regression, LDA, QDA, and Random Forest. We made this decision because the runtime of SVM and KNN was far too long and infeasible to tune on. We decided to use MSE as our loss function when running CV generic but other possible loss functions include: classification error $\frac{1}{n}\sum_{i=1}^{n} I_{\{y_i=\hat{y}_i\}}$, hinge loss $1-\hat{y}_i y_i$, logistic loss $\hat{y}_i y_i + \log(1+e^{\hat{y}_i})$. In this case, MSE is the same as classification error because the MSE function will sum all the times our predicted values do not equal the actual value and divide the sum by the total number of points.

We decided to use the following covariates in our classification methods: CORR, SD, NDAI, and rad_angle_AN, since these are the important ones found in the EDA above. In our revised split method, we added in x and y as covariates since we know these two covariates will be useful precisely because the i.i.d. assumption of the pixels are not met. Note that the test accuracy of temporal split is much lower compared to the other two splitting methods, confirming our conjecture that there is another dependence relation not dealt with in temporal split. Five-fold CV is adopted for all analysis below.

A. Logistic Regression

Assumptions of logistic regression include: 1) the dependent variable is binary; 2) observations are independent of each other; 3) little or no multicollinearity among the independent variables; 4) linearity of independent variables; 5) large sample size.

The first assumption is satisfied if we disregard the unlabeled data points and classify cloud as 1 and no cloud as 0. The second assumption is technically not satisfied since the pixels are not i.i.d., although the spatial split and the revised split took this into consideration and by introducing the block structure into the data, we can assume the blocks are i.i.d. Assumption 3 is satisfied because we took out the highly correlated angle predictors and decided to only use rad_angle_AN as a covariate in our regression, along with SD, CORR, and NDAI (all of which have little collinearity between them). CV and test accuracy are reported below.

Split ~~Accuracy~~	CV fold 1	CV fold 2	CV fold 3	CV fold 4	CV fold 5	Test
Temporal	90.44%	90.56%	90.18%	90.21%	90.36%	78.31%
Spatial	88.70%	88.79%	88.80%	88.69%	88.90%	89.03%
Revised: Image 1	93.47%	93.30%	93.43%	93.39%	93.48%	93.44%
Revised: Image 2	98.19%	98.38%	98.19%	98.25%	98.35%	99.01%
Revised: Image 3	91.41%	92.37%	91.57%	92.14%	91.88%	92.16%

B. LDA

Assumptions for LDA include: 1) Multivariate normality: Independent variables are normal for each level of the grouping variable; 2) Homoskedasticity - the covariance matrix of each of the classes is identical; 3) No multicollinearity; 4) Independence.

The multivariate normality assumption and homoskedasticity assumption are technically not met but we will run the LDA classifier for the sake of comparison. The estimated covariance matrix for each class and normality checks are reported in part IV.

Split ~~Accuracy~~	CV fold 1	CV fold 2	CV fold 3	CV fold 4	CV fold 5	Test
Temporal	90.93%	91.02%	90.62%	90.71%	90.80%	79.37%
Spatial	89.24%	89.38%	89.44%	89.22%	89.41%	89.58%
Revised: Image 1	94.19%	94.26%	94.35%	94.29%	94.38%	94.01%
Revised: Image 2	97.77%	97.83%	97.60%	97.80%	97.87%	98.51%
Revised: Image 3	91.36%	92.32%	91.40%	92.06%	91.56%	92.14%

C. QDA

Assumptions for QDA include: 1) Variance-covariance matrices for each class are different from each other; 2) Multivariate Normality; 3) Independence.

Variance-covariance matrices for each class are indeed different (see part IV). Compared to LDA, there seems to be a small improvement in test accuracy in our spatial and revised splitting method, albeit the test accuracy decreased for image 3. The improvement is because we are no longer assuming homoskedasticity in the data. The decrease for image 3 could result from the significantly smaller number of useful pixels: as mentioned earlier, this could lead to instability in our model, hence lower test accuracy.

Split ~~Accuracy~~	CV fold 1	CV fold 2	CV fold 3	CV fold 4	CV fold 5	Test
Temporal	91.46%	91.52%	91.17%	91.26%	91.32%	78.35%
Spatial	89.37%	89.61%	89.50%	89.34%	89.77%	90.31%
Revised: Image 1	94.97%	95.18%	95.23%	95.11%	94.94%	94.70%
Revised: Image 2	98.57%	98.59%	98.41%	98.57%	98.59%	99.37%
Revised: Image 3	91.34%	92.18%	91.30%	91.53%	91.72%	90.89%

D. Random Forest

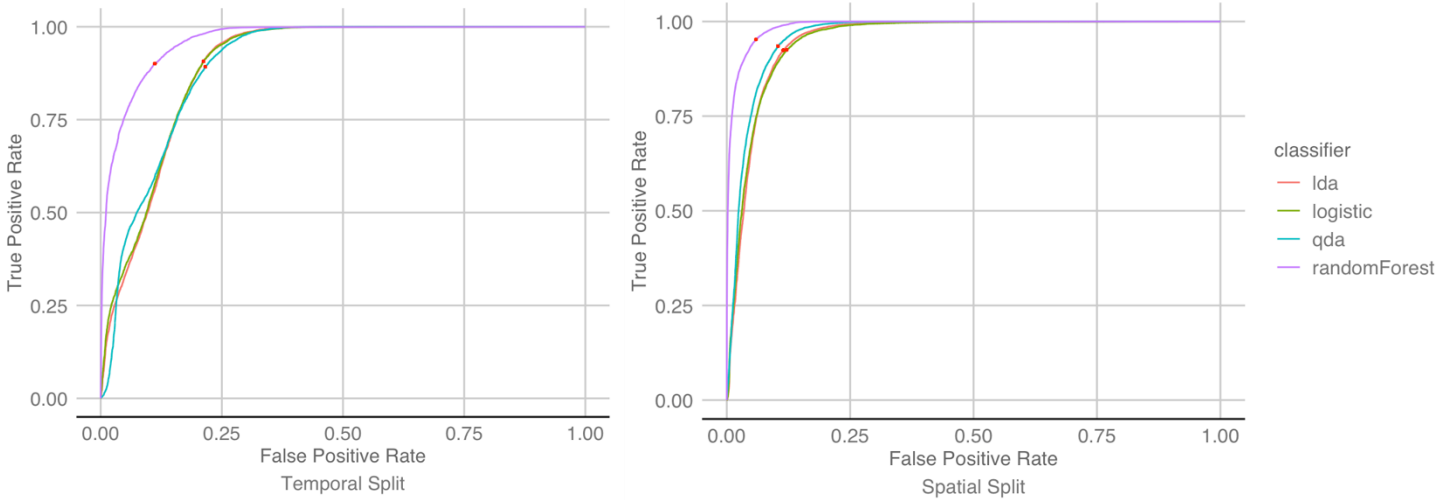
The only assumption for Random Forest is that data sample is representative. Because our data are 3 different orbital pictures of the same region along path 26 in the orbit, there are two ways to evaluate this assumption: if we further assume that all future data coming in is still from the same region, then our data sample is representative. However, if future data is from another region, then this might not be the case. For both of the cases mentioned above, we assume that all future data coming in is in the same format and have the same features, with x and y coordinates to cut the future image into blocks.

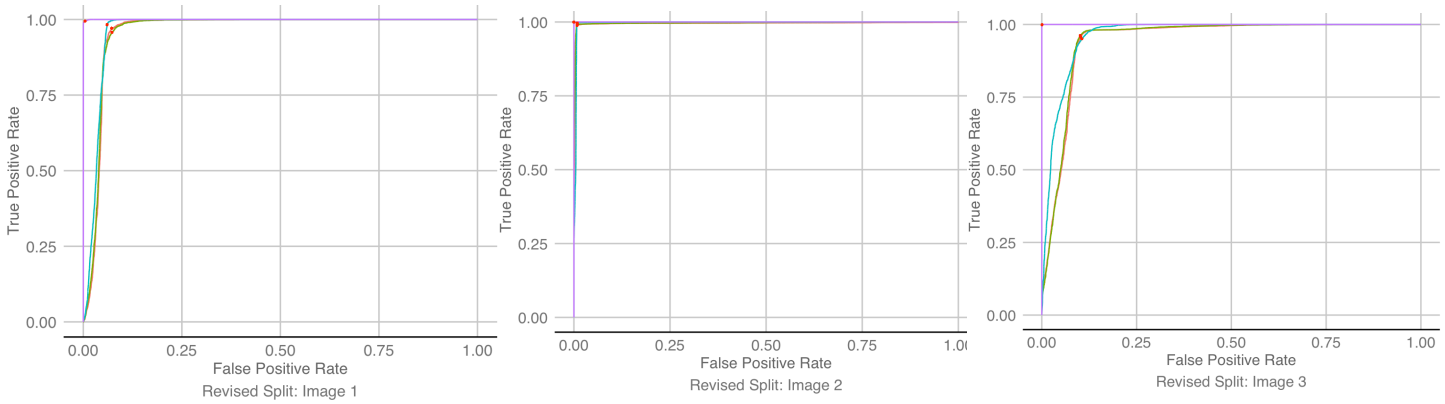
Split ~~Accuracy~~	CV fold 1	CV fold 2	CV fold 3	CV fold 4	CV fold 5	Test
Temporal	95.14%	95.29%	95.21%	95.03%	95.27%	89.25%
Spatial	94.62%	94.57%	94.27%	94.40%	94.56%	94.51%
Revised: Image 1	99.96%	99.98%	99.94%	99.97%	99.97%	99.43%
Revised: Image 2	100.00%	99.99%	99.99%	99.95%	99.98%	99.99%
Revised: Image 3	99.98%	99.99%	99.97%	99.99%	99.99%	99.87%

Among the four classifiers we tried, random forest seems to perform the best out of all. Random forest has the highest test accuracy as well as CV accuracy across all splitting methods as well, making it the best classifier by a large margin. It is worth noting that LDA, QDA, and logistic regression have comparable test accuracies. Different splitting methods lead to hugely different CV accuracies as well as test accuracies, suggesting some splitting methods are more preferable than others when it comes to maximizing test/CV accuracy.

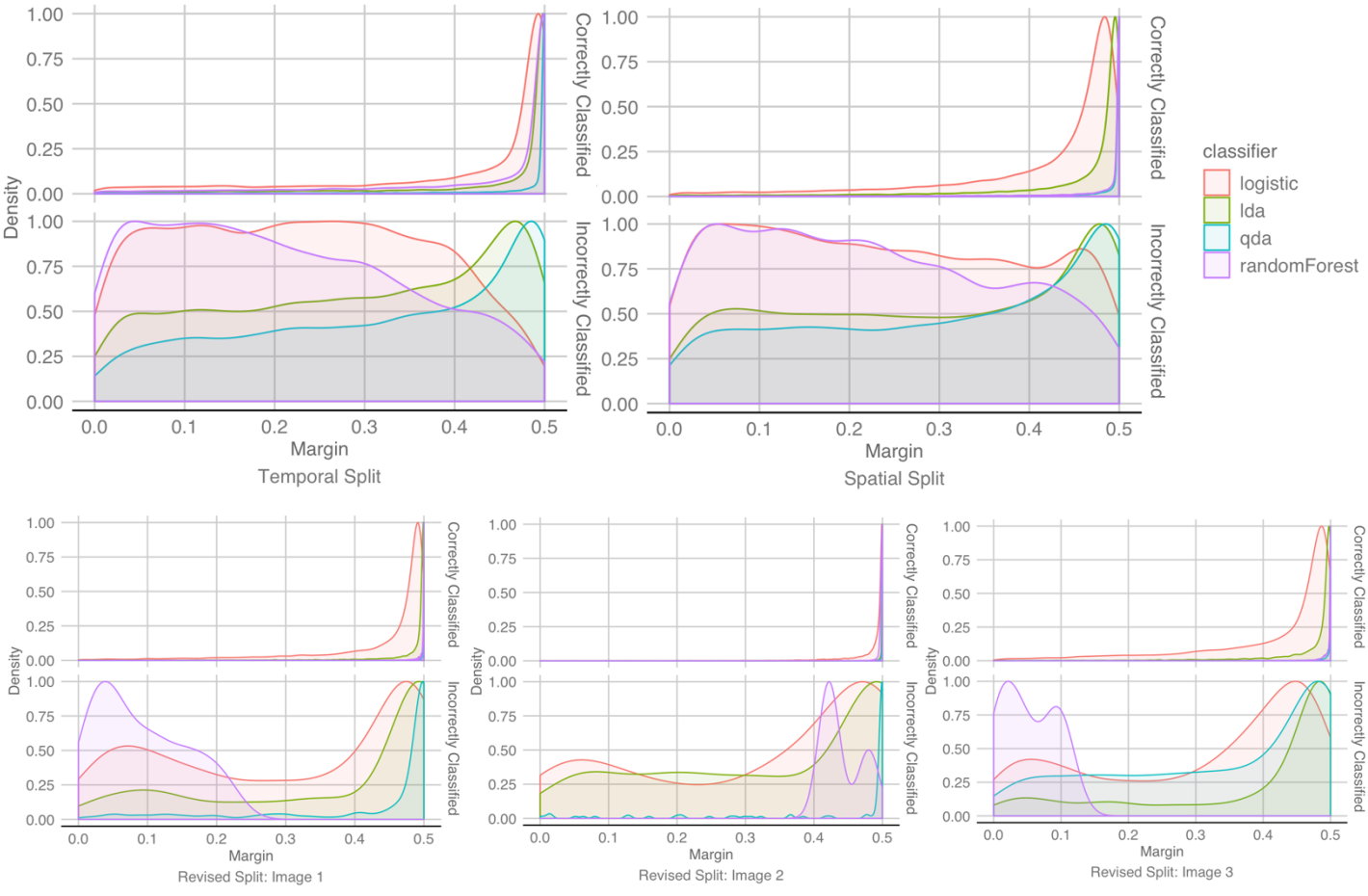
To help evaluate the performance of each classifier, we graphed ROC curves for all splitting methods, and computed the cutoff values across splitting methods and classifiers. The cutoff values are chosen by minimizing the distance to the perfect classifier which has (1,0) as (True Positive Rate, False Positive Rate), since we don't have a preference of lower false positive rate or false negative rate without more information. The cutoff values are graphed as the red dots on the curve individually, and the cutoff for true positive rates are reported in the table below. Note that LDA and logistic regression have very similar ROC curves hence cutoff values. It seems like random forest has the largest area under the curve, making it the best classifier in each splitting method.

Classifier ~~Split~~	Temporal	Spatial	Revised: Image 1	Revised: Image 2	Revised: Image 3
Logistic Regression	0.9069	0.9254	0.9575	0.9908	0.9620
LDA	0.9071	0.9240	0.9709	0.9886	0.9584
QDA	0.8923	0.9351	0.9828	0.9966	0.9506
Random Forest	0.9007	0.9525	0.9945	0.9996	0.9991





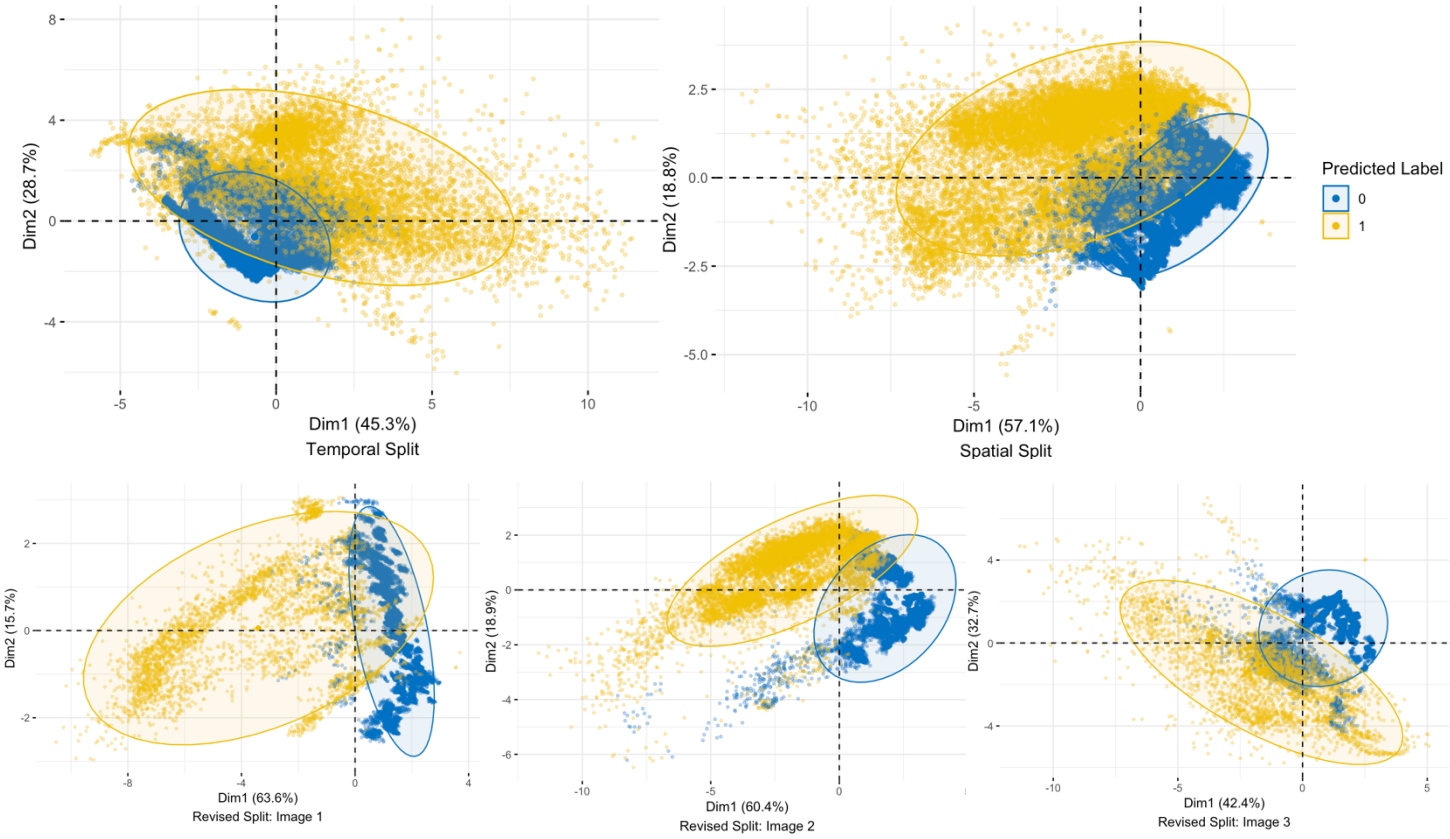
Besides ROC curve, we also examined the margin of predicted probabilities, which is the absolute difference between the predicted probability of being in class 1 and the traditional cutoff value 0.5. By looking at the distribution of margin for correctly predicted entries and incorrectly predicted entries separately below, we were able to get some feedbacks on how "confident" the classifier is on average, and use the margin as a proxy for model performance. It seems like random forest is the most "confident" classifier among all, having the highest percentage of large margin for correctly classified points across most splitting method. It is worth noting that while all classifiers have relatively large margins for correctly classified points, the margin for incorrectly classified points vary quite a bit. Ideally for incorrectly classified points, we want the margin to be as small as possible, i.e. if the classifier gets a prediction wrong, we'd rather have the predicted probability of being in class 1 to be close to 0.5 than 0. It can be observed that while random forest usually peaks around 0-margin for incorrectly classified points, it is not the case for image 2 in the revised split. In comparison, LDA, QDA and logistic regression usually peak closer to the maximum margin of 0.5 for incorrectly classified points across splitting methods. From the margin criteria, random forest is our best classifier because it is the most "confident" in correctly classified data points, and makes the least absolute error in prediction when it comes to incorrectly classified data points.



IV. DIAGNOSTICS

We decided to choose QDA as a good classification model of our choice because it had relatively high accuracy and is very interpretable compared to random forest, even though random forest has better accuracy. We will begin by checking the non-linear decision boundaries with PCA, then we will investigate the different variance-covariance matrix for each class, and finally check multivariate Gaussian assumptions.

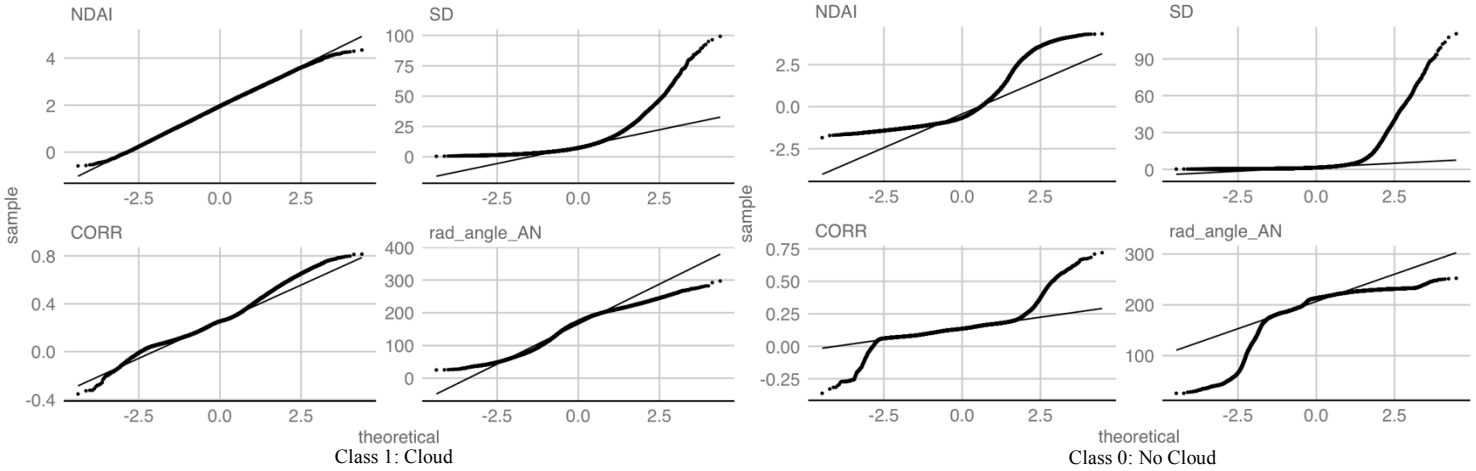
To check decision boundaries, we graphed the predicted labels projected into the first and second principal components in order to use the visual aid. It can be seen that there are clearly two separate classes in the PCA projections, and we can see that the two concentration ellipses of normal probability exhibit different radii, hinting at different covariance structure between classes. The decision boundaries are not linear, so QDA gives better fit than LDA in part III.



To estimate the parameters of QDA quantitatively, we looked at the two classes separately. We know from lecture that $\hat{\mu}_k = \frac{1}{N_k}\sum_{i:y_i \in k}^{N_k} x_i$, $\hat{\Sigma}_k = \frac{1}{N_k}\sum_{i:y_i \in k}^{N_k}(x_i - \hat{\mu}_{y_i})(x_i - \hat{\mu}_{y_i})^T$, so we used these formulas to estimate the QDA parameters. For class cloud (labeled 1) and class not-cloud (labeled 0 here), the order of the vector (and matrix) entries is NDAI, SD, CORR, and rad_angle_AN. The estimates are given below.

$$\hat{\mu}_1 = \begin{pmatrix} 1.95 \\ 9.84 \\ 0.26 \\ 163.03 \end{pmatrix}, \hat{\Sigma}_1 = \begin{pmatrix} 0.45 & 2.53 & 0.02 & -8.05 \\ 2.53 & 68.75 & 0.22 & -98.83 \\ 0.02 & 0.22 & 0.02 & -4.57 \\ -8.05 & -98.83 & -4.57 & 2087.32 \end{pmatrix}, \hat{\mu}_0 = \begin{pmatrix} -0.26 \\ 2.98 \\ 0.14 \\ 204.78 \end{pmatrix}, \hat{\Sigma}_0 = \begin{pmatrix} 1.12 & 4.05 & 0.02 & -16.09 \\ 4.05 & 34.35 & 0.09 & -82.51 \\ 0.02 & 0.09 & 0.00 & -0.45 \\ -16.09 & -82.51 & -0.45 & 656.36 \end{pmatrix}$$

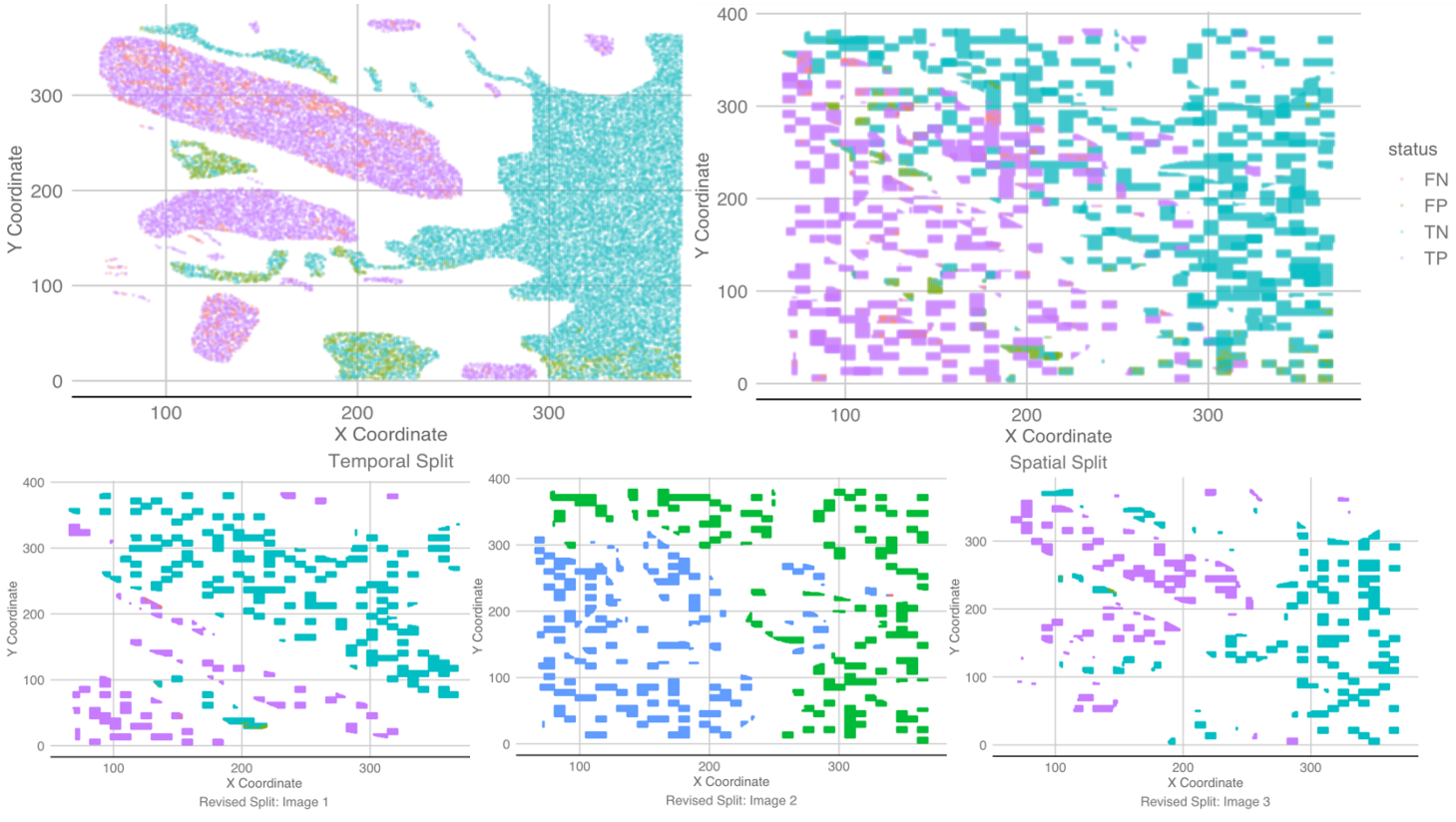
We also checked the Gaussian assumption of QDA, graphing QQ-plots for each covariate used in the fit. We can see here that Gaussian assumptions are not completely satisfied here for our predictors, which may explain why QDA isn't the best classifier when compared with random forest, but it is still good as a classifier. Note that all analysis done in this section so far is consistent across all splitting methods, since normality and homoskedasticity assumptions need to be satisfied for all data points available to achieve optimal results.



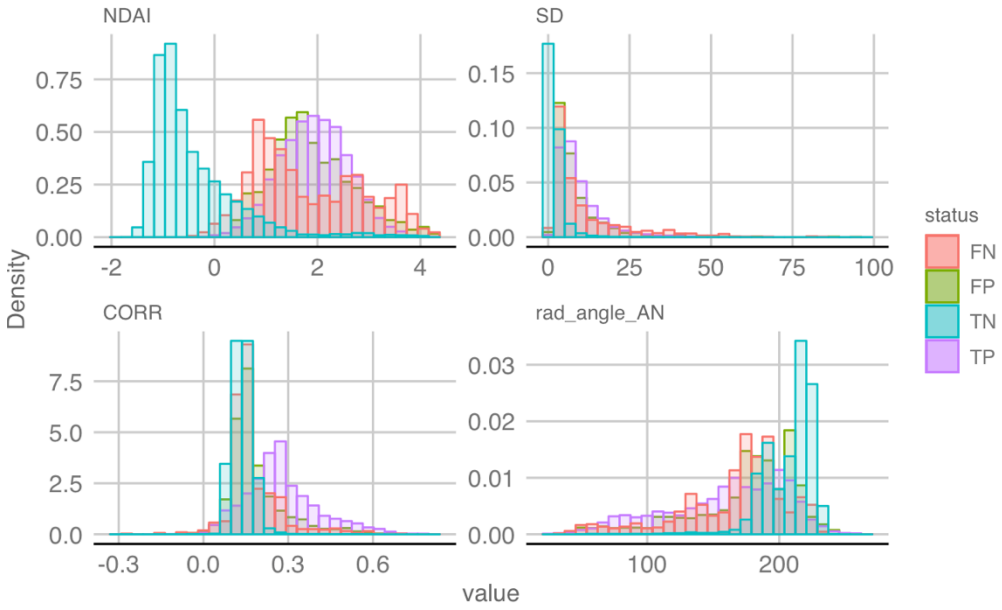
Class 1: Cloud

Class 0: No Cloud

Our best classification model is random forest, and a closer look at the misclassification entries reveals a distinct pattern. As can be seen from the graphs below, some points in the middle of a large cloud or large not-cloud area are misclassified in the temporal split, suggesting some dependence structure is not considered in this splitting method, as discussed in section II above. For other splitting methods, however, we see that misclassification only happens at the boundary of cloud and not-cloud. This could be because at the boundary there are not enough data points for the model to pick up the true relationship between the label and other features. Another reason could be that these misclassified points are sandwiched by data points from the opposite class, making them much harder to separate, hence to predict. Also note that there are no false positive points in image 2 of the revised splitting method, while there are only a handful of misclassified points in total in revised splitting method. From this, we can conclude that the pattern is different across splitting methods.



Patterns of misclassification in other features are also noticed. A graph of distributions for all features used is given below, broken up by their possible status: true positive, false positive, true negative, or false negative. This particular graph is taken from the spatial split, because in reality graphs across all splitting methods are very similar. As can be seen, true negative points have its distinct distribution for all features but CORR, and false negative points mimic the shape of true negative distributions in all features, although clustered together with true positives and false positives. This makes sense because data points with highly overlapping features are much harder to classify than linearly separable ones. However, it is possible that this analysis is not robust enough: random forest has very high accuracy across all splitting methods, especially in revised splitting method where only dozens of points are misclassified for each image. Hence, patterns recognized on these test sets can be based on very few data points and are not guaranteed to be true, especially not when the future data coming in is from a completely different region.



Given our analysis above, there are a few things that can possibly be done to improve the test accuracy. Since we only tried four classification methods, maybe soft-margin SVM is another option we can explore, because of the non-separable nature of the features in this dataset. Alternatively, we can consider the fact that most misclassification happens at the boundary and advise NASA to collect more data point in such boundary cases. Or, we can try to engineer a new feature that incorporates the neighborhood information of the pixels: if one pixel is within a small enough distance to a block of pixels known to be cloud or not cloud, the probability of this one particular pixel being in the same class should be very high. Currently, we are not yet at the level of engineering such complicated features, but this is something to come back to when we have more domain knowledge and experience.

When it comes to future performance, the case depends on the format of the future data. If the future data is from the same region as the three images here, and has the same feature with the same range of values, then our model should perform fairly well. However, if the future data coming in is from a different region where radiance angles play a different role in predicting labels, or if the values of features are extreme, then it is not surprising if our model doesn't do as well as we want. If it is the case that future data will change its format, be it different features, different regions, or different range of values, we can use the same EDA and data exploration methods employed in this report to study and pick the best features again, and train other models with different parameters. This report has set up a streamline of work that can be reused in the future for possible further EDA, hence we are confident that our model will perform above average if all steps mentioned are followed closely.

V. CONCLUSION

The goal of this project was to exploration and model cloud detection in the polar regions based on radiance recorded by the MISR sensor abroad the NASA satellite Terra. We attempted to build a classification model to distinguish the presence of cloud from the absence of clouds in the images using the features given to us. To begin with, we performed exploratory data analysis on each image and visualized the distributions and correlations of relevant features grouped by expert label type (cloud vs. not cloud) to better understand which features were best to use in our classification algorithms. After cross validation and accuracy measurement on various models and train-test splitting methods, we concluded that random forest on our revised splitting method performs the best, with CV accuracy of around 99.95% across 5 folds for each of the 3 images given. If we had more time to train longer and more computer memory capacity, we would've more deeply explored soft margin SVM and KNN as classifiers. Moving forward, we advise NASA to collect more data on the boundaries, and bring in more cloud labelling experts to engineer a feature that captures the spatial dependence better. For future image data, much of the analysis would be the same assuming we are given data with the same features and format, although in the case of different format of data our streamline of EDA and feature importance work can still be followed to obtain a new classification model.

CONTRIBUTION

Jessica worked on data collection and reproducibility section, while Yiming worked on the modeling section. We worked together on preparation and diagnostics section.

GITHUB REPOSITORY

https://github.com/jessica-cherny/stat154-proj2