STA 521 Project 2

Eli Gnesin (ejg45) & Caleb Woo (csw57)

2022-12-06

Data Collection and Exploration

Summary of the Paper

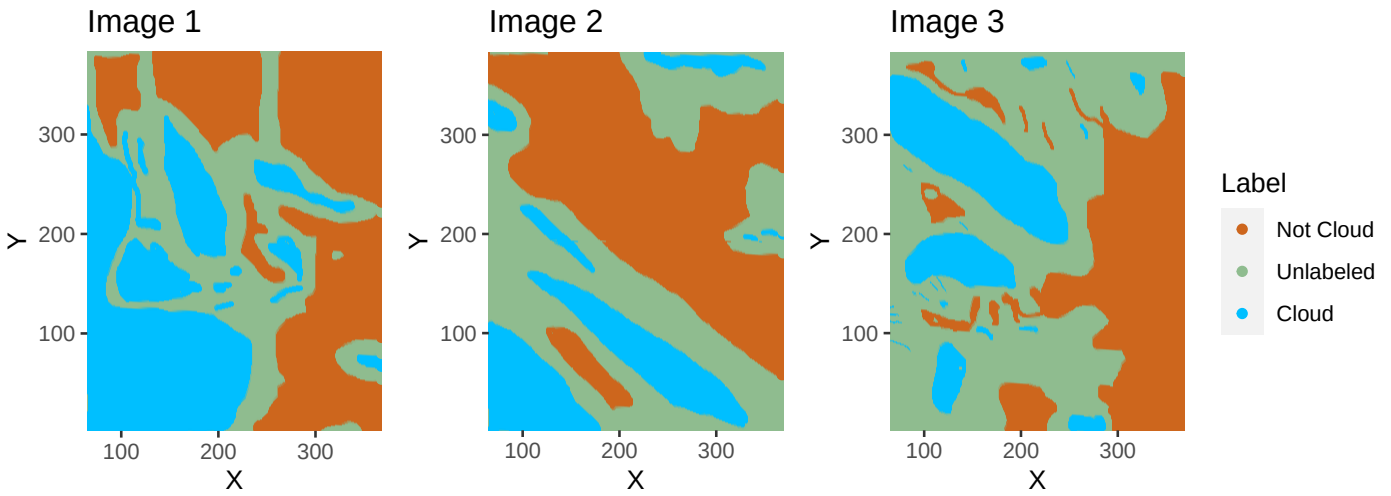
In 2008, Yu et al. published "Daytime Arctic Cloud Detection Based on Multi-Angle Satellite Data With Case Studies" (Shi et al. 2008). The purpose of the study was to explore the state of daytime Arctic cloud classification, and propose a new "Enhanced Linear Correlation Matching Algorithm" and classification system that is less reliant on manual human classification.

The data used in the study was collected by the Multiangle Imaging SpectroRadiometer (MISR) on NASA's Terra satellite, and for this study, 10 orbits of the satellite path over the Arctic, northern Greenland, and Baffin Bay, collected between April and September 2002, were used. Together, this dataset comprises 57 "data units" of three MISR blocks each (with a block being $\frac{1}{180}$ of a path), totaling 7.11 million pixels of 1.1 kilometer resolution (as well as 275 meter resolution for red radiation measurements). The labels of cloud cover and no-cloud cover (as well as an "ambiguous" label), were provided by industry experts.

The research team then used the data to construct three "physical features," CORR, SD, NDAI. CORR is the average of the linear correlations of two pairs of radiation measurements (Af/An and Bf/An), where higher values suggest a cloud-free image. The second, SD, is the standard deviation of the An radiation measurements, used to help detect "smooth cloud-free surfaces", and NDAI is the average of two radiation measurements over a 1.1 kilometer spacial resolution. Using these features, the research team created two decision rules to label a 1.1 km square pixel as clear, and with setting appropriate thresholds, reached over 91% agreement with the expert labels and 100% coverage in labeling. From this, the team demonstrated that the three physical features were sufficient to accurately classify cloud cover in arctic environments with better spatial coverage and real-time adaptive thresholding to improve the robustness of the model. Furthermore, the ELCM algorithm was used to train QDA to provide probability labels for partly cloudy scenes and was found to be effective in identifying cloud boundaries. By improving understanding of the flow of radiation through the atmosphere and how clouds respond to changes in arctic climate, this study is the first step towards analyzing how changing cloud properties may improve or destroy the changes in the Arctic brought about by climate change.

Summary of the Data

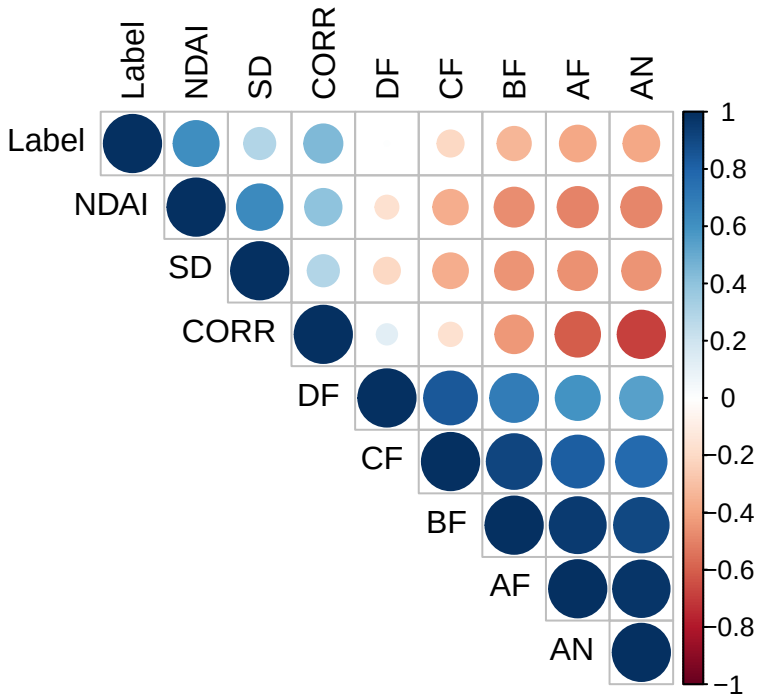
Label	% of pixels
Not Cloud	36.776
Unlabeled	39.789
Cloud	23.435



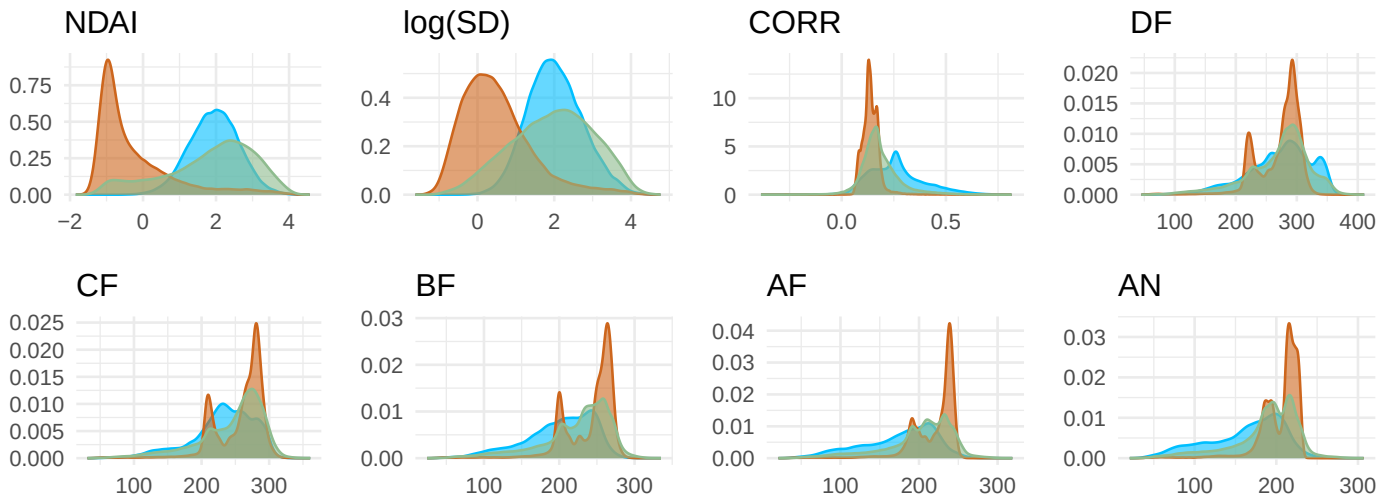
Similar to the ELCM-QDA results from the paper, we observe many ambiguous and unlabeled points around the cloud boundaries which separate the cloud pixels from the non-cloud pixels. An i.i.d. assumption for the samples cannot be justified for this dataset because individual pixels seem to depend on the pixels around it. If the data appeared to be i.i.d. we would expect a random scatter of pixels at any given region of an image. However, we observe clear spatial trends where cloud pixels are near other cloud pixels, unlabeled pixels border the clouds, and non-cloud pixels are near other non-cloud pixels. Therefore, individual pixels are essentially meaningless outside of the context of nearby pixels so we cannot make an i.i.d. assumption on the data.

Exploratory Data Analysis

We now continue with a short exploration of the data itself. First, we look at the pairwise correlations between the features, excluding the coordinates:



From these correlations, some trends stand out. All three of the "physical features" in the paper (NDAI, SD, CORR), are positively correlated with label, which indicates that *higher* values of all three of these features are associated with a label of +1, meaning cloud cover. In contrast, however, four of the five given radiance angles show negative correlation with the label variable, indicating that *lower* values of radiance are generally associated with cloud cover. We can also compare differences between the features in the two classes by considering the densities of the features separated by label:



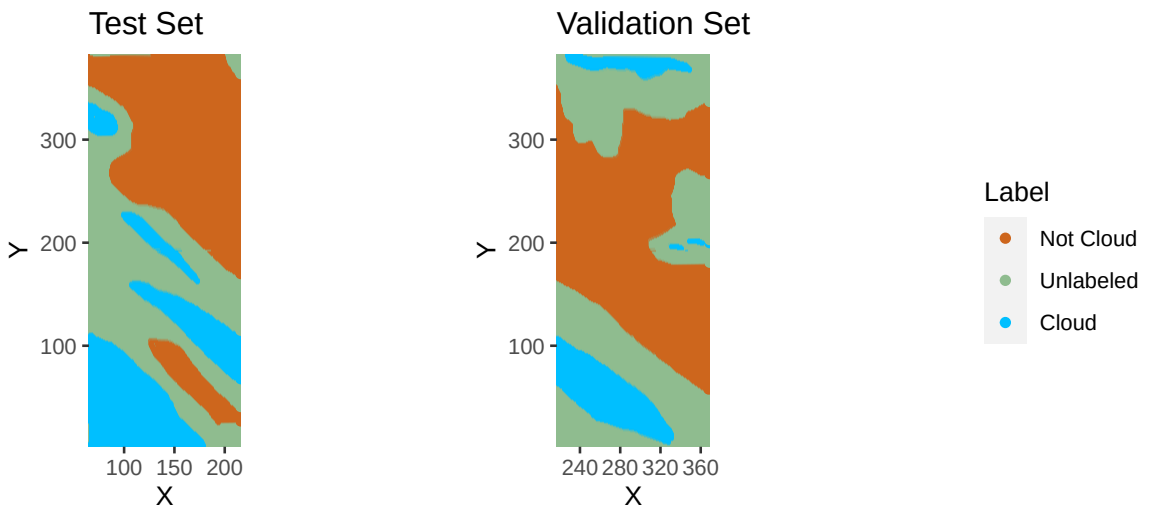
These density plots reinforce the same ideas seen in the correlations above. In the three "physical features", the "cloud" labeled points have densities higher than the "non-cloud" labeled points, and conversely, for four of the five radiances (all except for DF), the bulk of the density for the "non-cloud" labeled points is above that of the cloud labeled points. Interestingly, the radiance densities for the "cloud" labeled points are generally unimodal, whereas the radiances for the "non-cloud" labeled points are generally bimodal, with the larger mode above the cloud density mode and the smaller mode below. This helps explain the need for the physical features; the radiances are less uniformly separable, and thus have less predictive power than the physical features.

Preparation

Data Split

In splitting the dataset into training, validation, and test subsets, it is important to preserve the structure of the data, that is, that the observations are denoted by their X and Y spatial coordinates, and any split of the data would have to preserve this spatial structure. One method for doing so is to carve each image up into some k blocks, and then to randomly permute the block identifiers, the range of numbers from 1 to $3k$, and then take a split from the random block permutations. Such a method does preserve the spatial structure of the data, as each point is in a set with some (or all if not an edge) of the points around it. The random permutation of the blocks also ensures that, while the spatial structure of the points relative to each other is preserved, the structure of the image is not considered in the model, such that the orientation and structure of a individual image is a considered feature of the model.

A second method for splitting the data into three sets is to assign all data from images 1 and 3 to the training data. Then we can split the image 2 data in half to assign each half to the validation and test sets respectively. By doing so, we retain the majority of the data for training to better fit our classifiers. In addition, by splitting image 2 in half, we still retain the spatial structure of the data for the validation and test sets. We believe that image 2 is the best choice to split in half for our validation and test sets because each half contains numerous samples from each class. Looking at the maps of the validation and test sets below, we see that each set contains a sizable number of cloud and non-cloud samples. Since we would like to incorporate this image split method with the cross validation function, we will incorporate the validation data which is half of image 2 into the training data and use the validation data along with the training data from images 1 and 3 to train and validate on K folds of data before checking how the trained model will perform on the separate test data from image 2.



Baseline

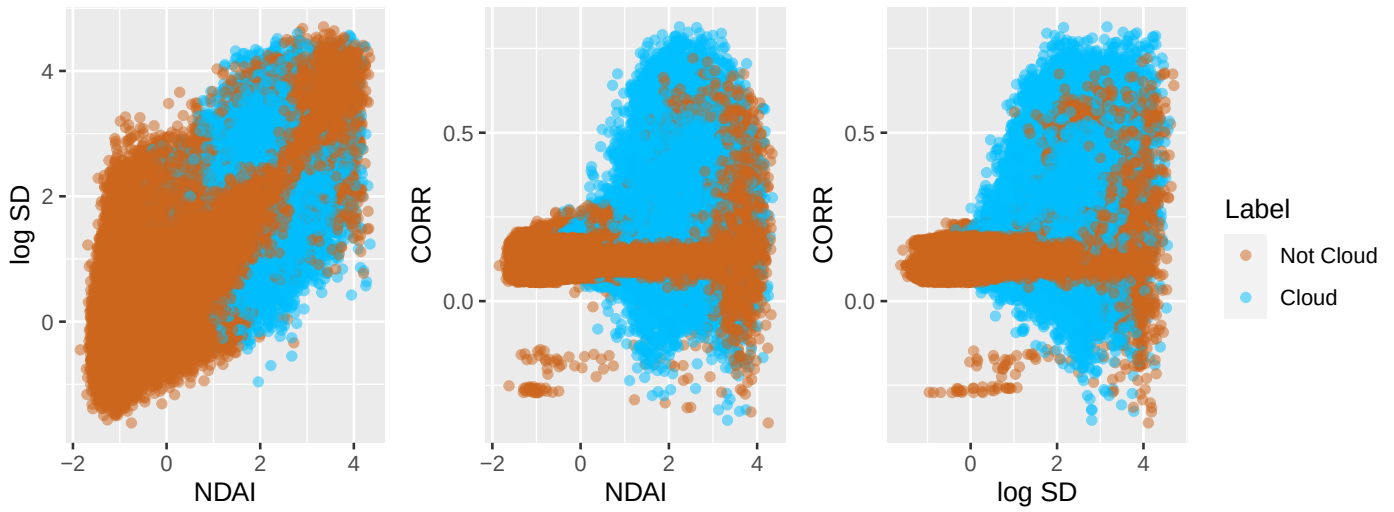
Before splitting the data, we first remove all observations with 0 labels which correspond to the unlabeled points because we are primarily focused on detecting the presence of clouds or no clouds in the images. Therefore, a binary classification task between the 1 and -1 labels is a more appropriate way of approaching this problem. Even if we approached this problem as a multi-classification task between 3 different labels and were predicting all 3 labels accurately, predicting the unlabeled points accurately is somewhat meaningless in the context of detecting the presence of clouds or no clouds in the images.

We can now split the data using the methods outlined above and run a "classifier" which just sets all points to label -1, indicating cloudlessness. For the block splitting method, we split each image into 64 blocks, for a total of 192 blocks, of which 30 are for testing, 30 are for validation, and 132 are for training. This method gives a validation set misclassification rate of 40.6% and a test set misclassification rate of 43.2%, which is about in line with the overall rate of -1 labels being 59.4% and 56.8% respectively. This classifier would yield high accuracy if the image data was unbalanced to be mostly cloudless images, but in a scenario where a slight majority of points are labeled -1, it is not a particularly accurate classifier.

Next, we will run this trivial classifier on our image splitting method where we split image 2 in half for the validation and test sets. This method gives a validation set misclassification rate of 21.1% and a test set misclassification rate of 36.4%, which is about in line with the overall rate of -1 labels being 78.9% and 63.6% respectively in the binary dataset (with unclassified points removed). Once again, this trivial classifier will have a high average accuracy if the majority of the points are -1 (cloudless). Since the majority of the points are cloudless in the validation and test sets, this trivial classifier appears to be fairly accurate even though it is just predicting the majority class each time.

First Order Importance

In general, the best set of features should include features that are strongly correlated with the response, in this case label, and features that are weakly correlated with each other. Looking at the correlation plot in the exploratory data analysis section, NDAI and CORR are fairly strongly correlated with label and are fairly weakly correlated with each other. Although SD is weakly correlated with CORR which would make it a good feature to include with CORR, it may not be preferable because it is strongly correlated with NDAI and weakly correlated with label. Looking at the scatter plots of the 3 combinations of these 3 variables below, we see that the -1 and 1 labels corresponding to the non-cloud and cloud labels are more easily separated with NDAI and CORR or with log(SD) and CORR. Therefore, we believe that NDAI and CORR are the best 2 features to select out of the 3 constructed physical features because they are both strongly correlated with the response label, are fairly weakly correlated with each other, and lead to more separable groups of labels.



Now looking at the radiance angles, the most promising features include BF, AF, and AN because they are all fairly highly correlated with label. Although the correlation of BF with label is slightly weaker, it has a weaker correlation with CORR which may make it a better feature to combine with NDAI and CORR.

Table 1: Mutual Information with Label

	NDAI	SD	CORR	DF	CF	BF	AF	AN
Entropy	0.426	0.315	0.271	0.106	0.094	0.18	0.206	0.194

Above is a table of the mutual information between each feature and the response label. The entropy of the empirical probability distributions are computed to quantify the amount of information obtained about label when observing one of the features. NDAI is certainly a good feature to select because it has the highest mutual information with label by a large margin. Although SD has a higher mutual information than CORR, since it is highly correlated with NDAI and CORR still has a high mutual information, we prefer to choose CORR alongside NDAI. As expected, the BF, AF, and AN radiance angles have a higher mutual information with label than the DF and CF radiance angles. Since the AF and AN radiance angles only have a slightly higher mutual information than the BF radiance angle, we may prefer to choose the BF radiance angle because it has a weaker correlation with CORR compared to the AF and AN radiance angles. The next table shows the table of the mutual information between the combination of NDAI and CORR with BF, AF, and AN respectively to see which combination of the 3 variables results in the greatest mutual information. The combination with BF results in the highest mutual information. This further justifies combining the BF radiance angle with NDAI and CORR because not only is it most weakly correlated with CORR, but it also gives the highest mutual information when in combination with NDAI and CORR.

Therefore, we have chosen the 3 best features to be NDAI, CORR, and the BF radiance angle because they are all fairly strongly correlated with the response label, they are only moderately correlated with each other, and they all have a relatively high mutual information with the response label.

Table 2: Mutual Information with Label

	NDAI, CORR, BF	NDAI, CORR, AF	NDAI, CORR, AN
Entropy	0.605	0.603	0.604

Generic Cross-Validation

We then wrote a function that allows us to pass in the name of a trivial classifier (or a function call to such a classifier), and a number of folds, and outputs the k-fold Cross-validation misclassification rate on the training data provided using the classifier. The function uses the block splitting method described in Section 2(a), with k^2 blocks per image. It then

partitions the blocks into k groups, and, for each partition, treats that partition as a test set while training the classifier on the other $k-1$ partitions. Finally, the function returns the mean of the misclassification rates of the classifiers predicting on each of the k test partitions. The function also allows for optional arguments, which are then passed to the generic classifier, so as to allow for classifiers to be modified from their generic versions.

The function also allows for the image splitting method described in Section 2(a). Taking images 1 and 3 from the training set and half of image 2 from the validation set, it splits these 2.5 images into K blocks which become the K folds used to train and validate the model in the function. The other half of image 2 is set aside as the test set so that after cross validation, the classifier is trained on the whole training set and the validation set before predicting and evaluating the performance on the test set.

Modeling

Table 3: Image Split Model Comparison

	LR	LDA	QDA	Tree	RF	GBM	Baseline
Fold 1 CV-loss	0.112	0.116	0.103	0.117	0.153	0.132	0.637
Fold 2 CV-loss	0.008	0.011	0.019	0.007	0.011	0.011	0.638
Fold 3 CV-loss	0.022	0.022	0.021	0.019	0.028	0.028	0.164
Fold 4 CV-loss	0.112	0.071	0.041	0.011	0.065	0.020	0.325
Fold 5 CV-loss	0.023	0.016	0.019	0.010	0.030	0.021	0.102
Fold 6 CV-loss	0.284	0.318	0.279	0.545	0.410	0.382	0.169
Fold 7 CV-loss	0.126	0.119	0.139	0.109	0.089	0.096	0.374
Fold 8 CV-loss	0.425	0.359	0.364	0.436	0.225	0.256	0.606
Average CV-loss	0.139	0.129	0.123	0.157	0.127	0.118	0.377
Test CV-loss	0.168	0.153	0.137	0.131	0.160	0.147	0.364

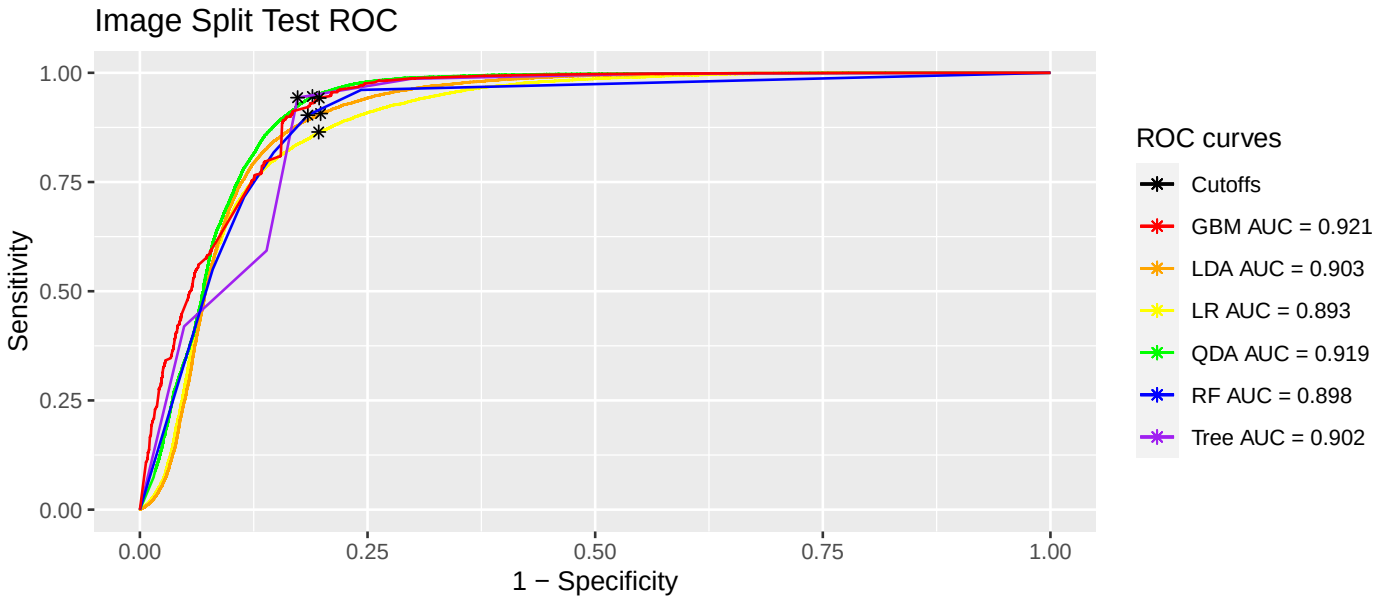
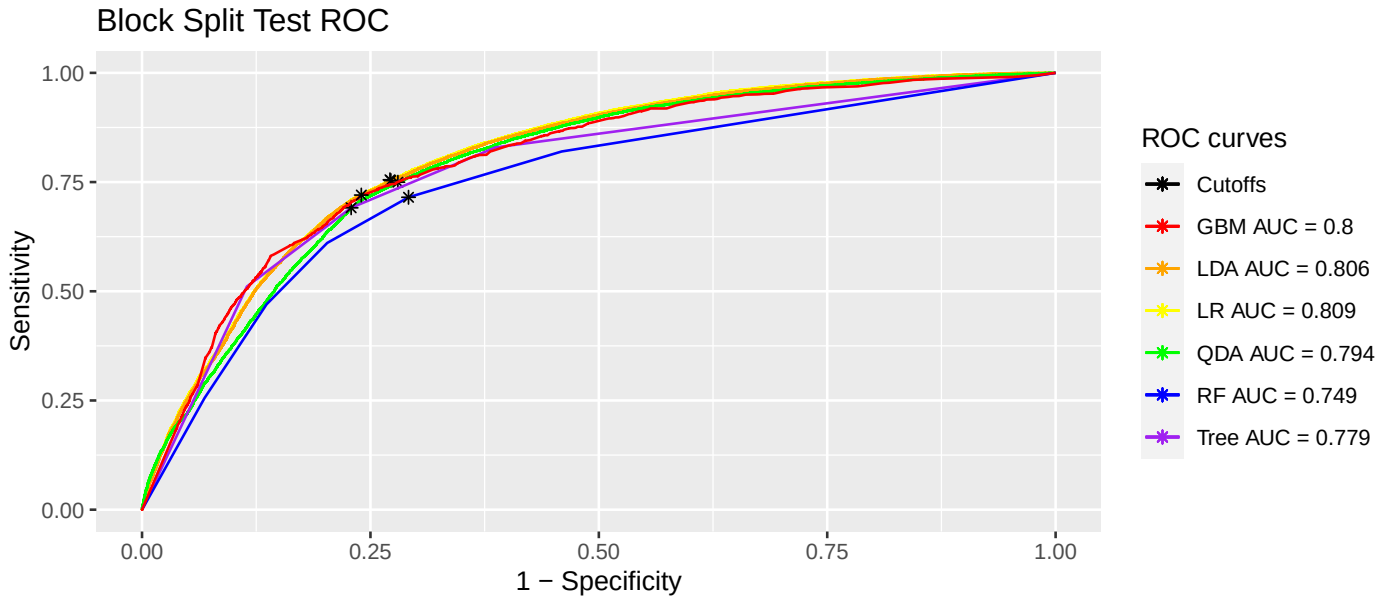


Table 4: Block Split Model Comparison

	LR	LDA	QDA	Tree	RF	GBM	Baseline
Fold 1 CV-loss	0.341	0.337	0.341	0.347	0.360	0.345	0.527
Fold 2 CV-loss	0.300	0.298	0.309	0.292	0.362	0.296	0.424
Fold 3 CV-loss	0.332	0.334	0.375	0.322	0.354	0.320	0.442
Fold 4 CV-loss	0.318	0.318	0.310	0.306	0.317	0.305	0.393
Fold 5 CV-loss	0.327	0.323	0.345	0.308	0.303	0.291	0.489
Fold 6 CV-loss	0.307	0.307	0.337	0.315	0.307	0.286	0.381
Fold 7 CV-loss	0.234	0.234	0.246	0.239	0.271	0.234	0.448
Fold 8 CV-loss	0.357	0.357	0.375	0.313	0.333	0.306	0.321
Average CV-loss	0.315	0.313	0.330	0.305	0.326	0.298	0.428
Test CV-loss	0.273	0.272	0.292	0.276	0.283	0.263	0.432

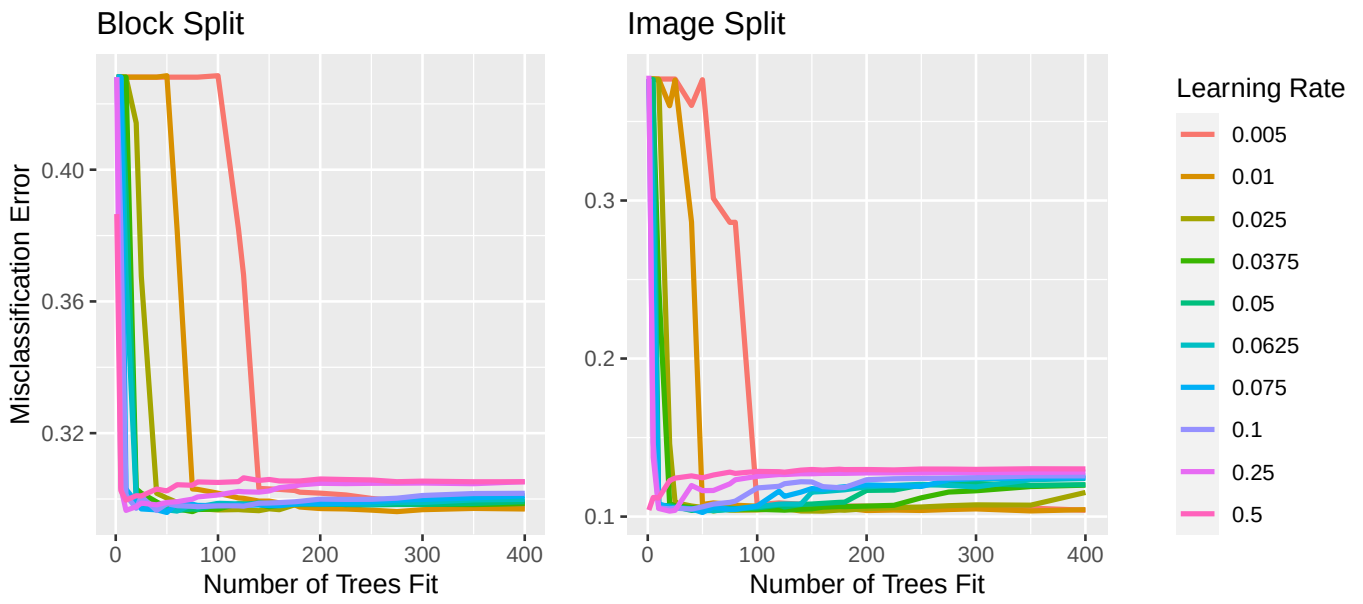


Since specificity is the true negative rate and sensitivity is the true positive rate, we would like cutoff values on the ROC curves that maximize both of them. To obtain the cutoff value of each ROC curve, we sum the specificity and the sensitivity at each threshold and then pick the threshold that has the largest sum. By doing so, we pick the cutoff value for each model that maximizes both the true negative rate and the true positive rate.

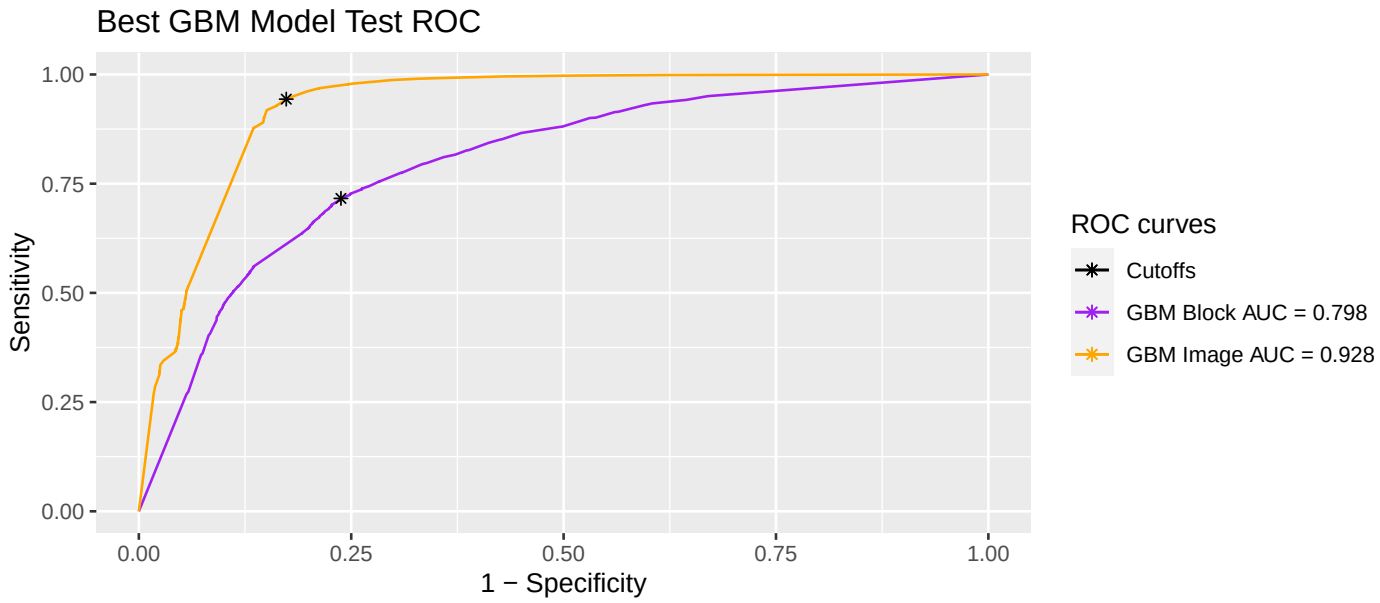
Diagnostics

Diagnostics of AdaBoost

Of the different classifiers we considered, one of the best performing ones was Generalized Boosted Models, specifically using AdaBoost exponential loss on binary outcome, where 0 indicated cloudlessness (changed from -1 because some modeling techniques need 0-1 binary outcomes) and 1 indicated clouds. In boosting models, there are 2 main hyperparameters to we are focused on estimating: the number of trees in the boosting model, and a shrinkage parameter applied to each tree. To determine the optimal parameters, we can fit a cross-validated model at different values of the hyperparameters and record the average misclassification error, as well as calculating the error on a test set with a model trained on the entire training set.



In total, 250 pairings of the two hyperparameters were trained on both split methods, with 25 distinct values for the number of trees and 10 distinct learning rate choices. In the block split method, at higher numbers of trees, the best performing learning rate is 0.01 with around 275 trees. For image split, higher trees meant significantly better performance with the 0.01 learning rate, compared to the other learning rates. However, the best parameter pair, for both split methods, by average cross-validated loss was 50 trees with learning rate of 0.075.



Looking at the ROC curve, we can choose a cutoff such that the sum of the specificity and sensitivity is maximized, since the goal is to have a model that has both maximum sensitivity and maximum specificity. We can also determine the best threshold for that cutoff, which for the image split method is a threshold of $\alpha_i \approx 0.59479$ and for the block split method is $\alpha_b \approx 0.23699$. With these thresholds, we can also calculate confusion matrices on the test sets for each split method. Looking at the confusion matrix, the image split method returns a test set predictive accuracy of $\approx 86.9\%$, with a sensitivity of $\approx 82.7\%$ and a specificity of 94.4%. Most significantly, the positive predictive value of the model, with cloudlessness as the positive class, is 0.9624, meaning that 96.24% of the cloudless points are predicted cloudless are actually cloudless. This suggests that the model is strong in that, if it predicts a pixel to be cloudless, it is very likely to actually be cloudless. In contrast, for the block split method, the test set predictive accuracy is $\approx 74.2\%$, with a sensitivity

of 76.2% and a specificity of 71.6%. Although weaker across the board than the image split model, the positive predictive value of the block split model is still 0.7796, and the model is still stronger in predicting cloudlessness than in predicting the existence of clouds.

Table 5: Image Split

	0	1
0	18890	739
1	3966	12352

Table 6: Block Split

	0	1
0	14864	4201
1	4640	10606

Comparing Misclassification Errors

Table 7: Image Split NDAI Range Errors

NDAI Range	Misclassification	P	FNR	N	FPR
(-1.85,-0.604]	0.000	0	NaN	63309	0.000
(-0.604,0.633]	0.066	1932	0.972	27060	0.001
(0.633,1.87]	0.358	29767	0.275	8014	0.663
(1.87,3.11]	0.195	34019	0.092	4391	0.997
(3.11,4.35]	0.489	2172	0.151	1450	0.994

Table 8: Block Split NDAI Range Errors

NDAI Range	Misclassification	P	FNR	N	FPR
(-1.69,-0.477]	0.028	5	1.000	45887	0.028
(-0.477,0.727]	0.405	2070	0.129	24296	0.429
(0.727,1.93]	0.212	27536	0.000	7387	1.000
(1.93,3.13]	0.112	32147	0.000	4039	1.000
(3.13,4.34]	0.436	2455	0.000	1895	1.000

Table 9: Image Split CORR Range Errors

CORR Range	Misclassification	P	FNR	N	FPR
(-0.363,-0.126]	0.139	78	0.051	73	0.233
(-0.126,0.109]	0.170	5036	0.623	22599	0.070
(0.109,0.344]	0.154	46616	0.223	81228	0.114
(0.344,0.579]	0.020	14548	0.000	303	1.000
(0.579,0.816]	0.013	1612	0.000	21	1.000

Table 10: Block Split CORR Range Errors

CORR Range	Misclassification	P	FNR	N	FPR
(-0.355,-0.124]	0.529	38	0.000	66	0.833
(-0.124,0.106]	0.149	6174	0.015	5956	0.288
(0.106,0.336]	0.188	42214	0.004	76435	0.290
(0.336,0.566]	0.062	14532	0.000	965	1.000
(0.566,0.797]	0.061	1255	0.000	82	1.000

Table 11: Image Split BF Range Errors

BF Range	Misclassification	P	FNR	N	FPR
(24.2,82]	0.466	612	0.000	533	1.000
(82,139]	0.174	4155	0.052	613	0.998
(139,197]	0.276	20931	0.129	6611	0.742
(197,254]	0.162	37135	0.253	49281	0.094
(254,312]	0.033	5057	0.240	47186	0.011

Table 12: Block Split BF Range Errors

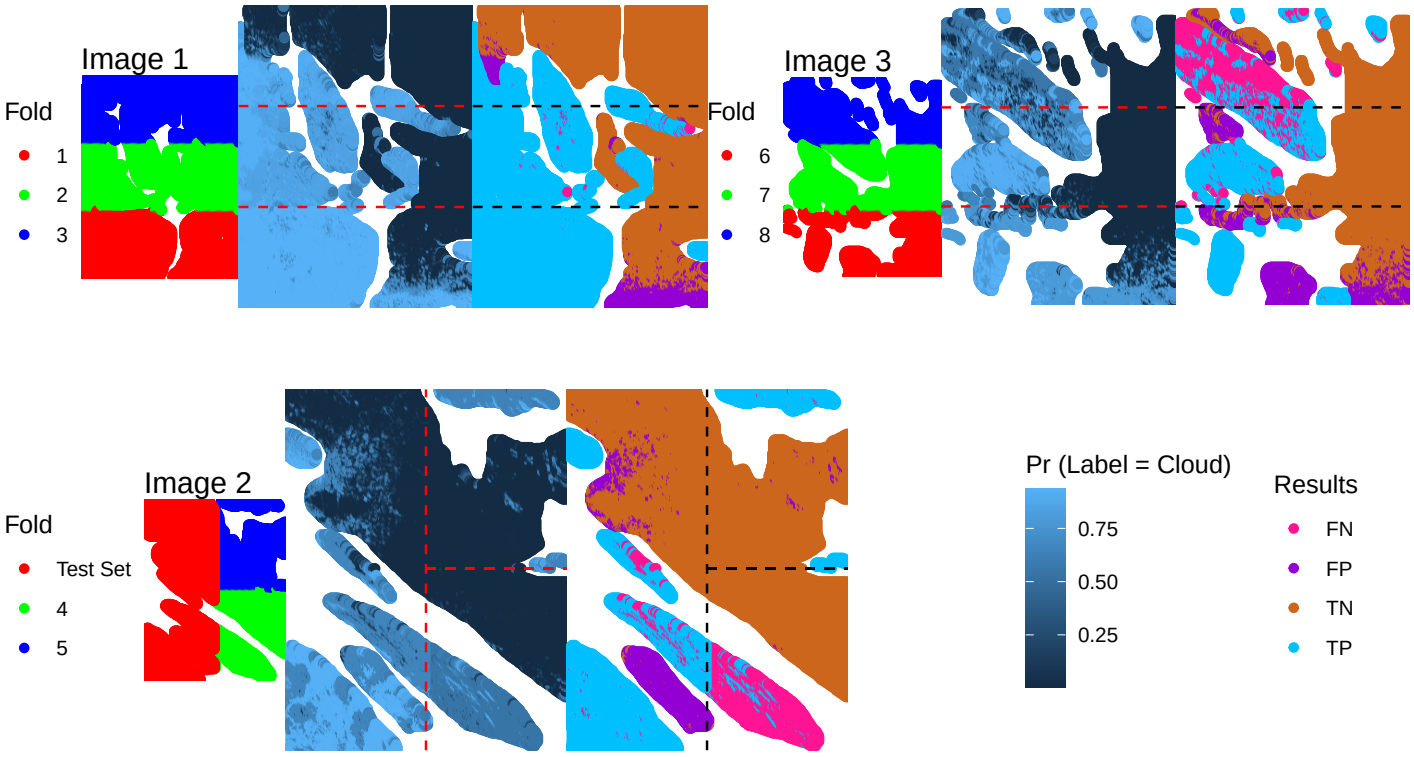
BF Range	Misclassification	P	FNR	N	FPR
(24.2,81.5]	0.490	770	0.000	739	1.000
(81.5,138]	0.140	6633	0.000	1084	1.000
(138,195]	0.142	17878	0.002	2951	0.987
(195,252]	0.198	32644	0.004	22299	0.483
(252,310]	0.153	6288	0.016	56431	0.168

Above are the same 3 density plots for each of our 3 best features separated by label from the exploratory data analysis section. Below the density plots are the tables of error rates of the 3 best features over 5 equally sized bins of the feature values for each of our data splitting methods. For each bin over the range of corresponding feature values, we have the overall misclassification rate, the number of cloud labels, the false negative rate, the number of cloudless labels, and the false positive rate. As evidenced by the densities of the cloud and cloudless labels for the NDAI and CORR features, the densities are well separated so that all 3 error rates over the 5 ranges of feature values reflect this.

Both data splitting methods have increasing false positive rates and decreasing false negative rates with increasing values of NDAI. These results align with our visual results because at lower values of NDAI, there is a high probability of observing cloudless labels so that there are a small number of cloud labels and a large number of cloudless labels that are mostly classified as cloudless which reflects the high false negative rate and the low false positive rate. As the range of NDAI values increase, the probability of observing cloud labels increases so that there are a large number of cloud labels and a small number of cloudless labels that are mostly classified as clouds which reflects the increasing false positive rate and the decreasing false negative rate. We observe fairly high misclassification rates of a large number of samples around NDAI values of 0.633 to 0.727 because this is about where the 2 densities of cloud and cloudless labels intersect at a high probability. The same pattern of increasing false positive rates and decreasing false negative rates is found with increasing values of CORR for the same reasons as NDAI. We observe the highest misclassification rates of a large number of samples around the CORR values of 0.106 to 0.344 because this is about where the 2 densities for CORR intersect at a high probability.

The pattern is reversed for increasing values of BF so that both data splitting methods have decreasing false positive rates and increasing false negative rates. These results align with our visual results because at lower values of BF, there is a high and more spread out density of cloud labels so that there are a large number of cloud labels and a small number of cloudless labels that are mostly classified as clouds which reflects the low false negative rate and the high false positive rate. As the range of BF values increase, the probability of observing cloudless labels drastically increases so that there are a small number of of cloud labels and a large number of cloudless labels that are mostly classified as cloudless which reflects the increasing false negative rate and the decreasing false positive rate. We observe fairly high misclassification

rates of a large number of samples around BF values of 195 to 254 because this is about where the 2 densities of cloud and cloudless labels intersect at a high probability.



The maps above correspond to the 3 images that are split using the image split method. Each row of maps represents one of the 3 images. The left most map of each row visualizes the points that are assigned to each fold from the image split method. In image 2, the left half is assigned to the test set as mentioned before while the right half is split into folds 4 and 5. The middle map of each row visualizes the probability of each point being a cloud label based on the cross validation response predictions for each fold or the test response predictions. Let us refer to the middle maps as the probability maps. The right most map of each row visualizes whether each point is a true positive, true negative, false positive, or false negative. Let us refer to the right most maps as the confusion matrix maps.

Based on the probability and confusion matrix maps, we immediately observe that the image 1 predictions are most accurate. The reason appears to be that image 1 has the largest number of expert labels provided so there are fewer gaps of unlabeled points which allows for better overall predictions. On the other hand, image 3 has numerous large gaps of unlabeled points and thus has numerous regions with high misclassification rates. The probability and confusion matrix maps also suggest that misclassifications most often occur in regions where the true cloud and cloudless labels are very close to each other. For example, the test set shows how a small cloudless region sandwiched between 2 larger cloud regions is almost completely misclassified as a cloud. This is less of a problem for a well labeled image like image 1, but even then it has many false positives on the left side of fold 3 where a region of cloudless points practically borders a region of cloud points. Furthermore, the high number of false positives and false negatives around the bottom border of fold 1, left border of fold 4, and all fold borders of image 3 suggest that the predictions often misclassify at the borders of the splitted image blocks. We believe that misclassifications at block borders is further exacerbated in the block split method because there are a larger number of smaller blocks. This would explain the higher misclassification rates across the board for the block split method compared to the image split method.

Finding a Better Classifier

Looking at the results from part 4(a), AdaBoost appears to be one of the stronger classifiers for predicting cloudless pixels, but a better classifier would have a negative predictive value closer to the positive predictive value of the model (for the

image split the NPV was 19% lower, and for the block split the NPV was 8% lower). A better classifier, either similar to AdaBoost with a different loss function, such as "huberized loss", might be more effective at handling this. Conversely, a classifier like soft-margin SVM might be a better classifier on this dataset, given it might be more robust to predicting future data correctly and predicting unlabeled points in the current dataset, but training SVM was computationally prohibitive for this analysis.

Comparing Split Methods

When looking at the results from the hyperparameter optimization on the AdaBoost model, the differences in the two split methods are immediately evident. In every iteration with more than 100 trees, with every learning rate, the image split method returned a misclassification error approximately 15%-20% lower than the misclassification error from the block split method. This was true for every model we attempted in part 3 as well. One possible explanation for this is that splitting the data into 64 blocks per image as opposed to 2-3 per image creates more violations of the spatial structure of the data, since each block has 4 edges so there are more points that are not connected to every point they are spatially adjacent to.

Conclusion

References

Shi, Tao, Bin Yu, Eugene E Clothiaux, and Amy J Braverman. 2008. "Daytime Arctic Cloud Detection Based on Multi-Angle Satellite Data with Case Studies." *Journal of the American Statistical Association* 103 (482): 584–93. https://doi.org/10.1198/016214507000001283.