Lab 4 - Cloud Detection
Stat 215A, Fall 2017

Alan Dong, Amy Ko, Xiao Li

November 16, 2017

Note: The Rnw file contains pre-generated plots only to save computation time. The code used to generate these plots are contained in "code" folder.

1 Introduction

Understanding the effects of the increasing levels of carbon dioxide on Earth's climate is of great interest to scientists and policy makers alike. A key part of understanding the global climate change involves the study of the Arctic, which is predicted to have the strongest dependencies between the surface air temperature and the level of carbon dioxide. In particular, cloud coverage is an important climatic factor as it modulates the effects of the rising air temperature in the Arctic. Despite the importance of acquiring accurate cloud measurements, however, cloud detection in the Arctic is challenging due to the similar light scattering properties between cloud and ice. In this report, we use randiances recorded by the Multiangle Imaging SpectroRadiometer (MISR) sensor to model and predict cloudy regions.

2 Data

In 1999, the National Aeronautics and Space Administration (NASA) launched the Terra satellite equipped with the MISR sensor, which consists of nine cameras with different viewing angles. Furthermore, each camera records in four spectral bands: blue, green, red, and near-infrared. The Terra satellite makes several passes over the Arctic and Antarctic every day, collecting a large number of images in paths approximately 360 km wide. For this report, we only use a tiny portion of the data: three "images" of Arctic ice and clouds, each with a coverage of approximately 8700 square km. Each "image" is composed of five red-color channel images of the same patch of Earth from five different angles: four forward-facing angles (called DF, CF, BF, and AF) and the nadir (straight down) angle (AN). Each pixel is labeled with an x and y coordinate and covers approximately 275-by-275 meters of the earth's surface. The data also contains three additional features, NDAI, SD, and CORR, based on domain knowledge, which are described in detail in [1]. By calculating the correlation between the radiances of different angles and the features NDAI, SD, and CORR, we see that they are highly correlated as shown in Figure 2.

3 Feature Selection

In this section, we explore the features of the MISR images. In particular, we will use images 1 and 2 to select the best features to train our prediction models, which will be discussed in Section 4. We can see from Figure 1 that NDAI stands out visually as a good predictor of the expert labels. Furthermore, when we plot the kernel density estimation of each feature grouped by the expert label (Figure 3), we see that the first three features, NDAI, SD and CORR, seem to better predict the expert label than the others. In fact, the conditional distributions of these three features (conditional on cloudy/clear) have very different range, medians, and modes. For other features, however, the conditional distributions tend to overlap.

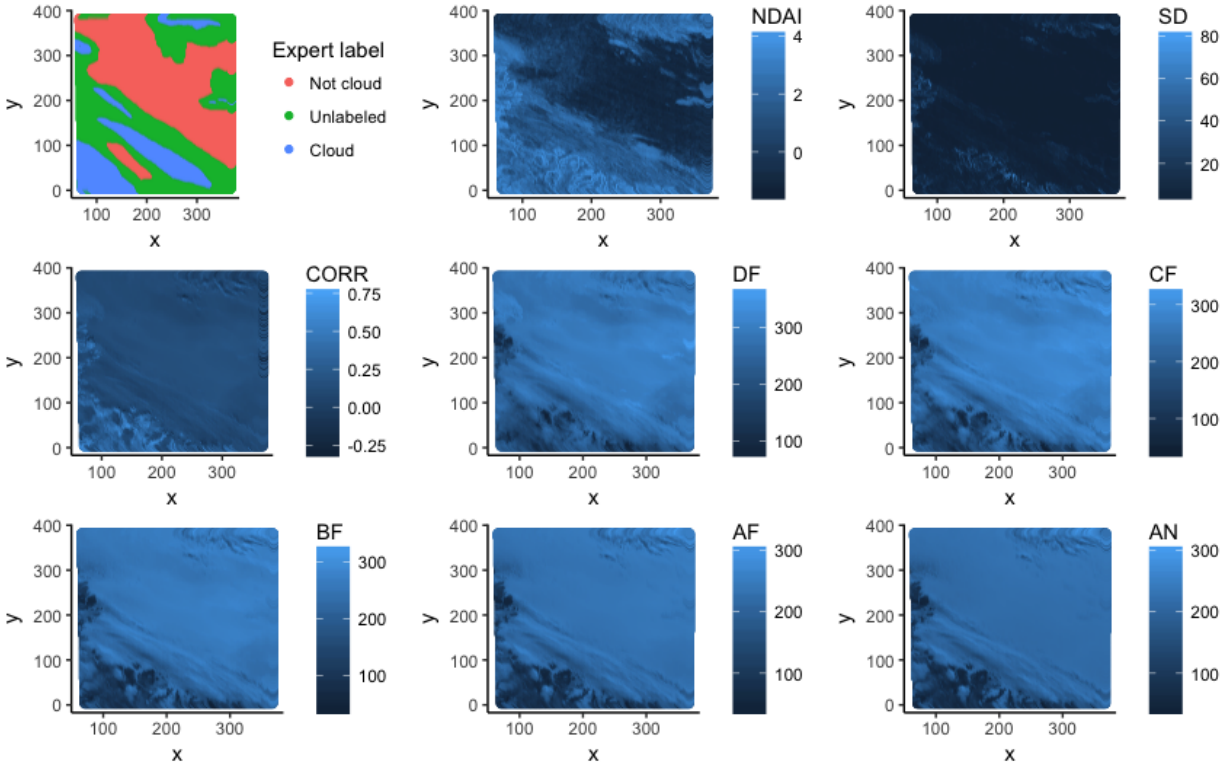


Figure 1: The expert label and the 9 features for image 1.

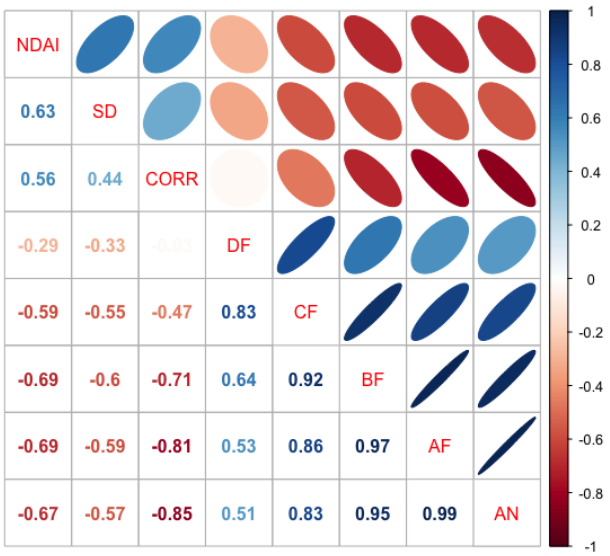


Figure 2: Pairwise correlation between features.

To quantify the difference in the conditional distributions, we compute their total variation distance, or $\mathbf{d_{TV}}$, which takes value between 0 (if the distributions are identical) and 1 (if the distributions are orthogonal). On Figure 3, this is the size of the overlap of the red and blue area. We have,

$$\mathbf{d_{TV}}(\text{dist}_{\text{clear}}(\text{NDAI}), \quad \text{dist}_{\text{cloudy}}(\text{NDAI})) = 0.87,$$

$$\mathbf{d_{TV}}(\text{dist}_{\text{clear}}(\text{SD}), \quad \text{dist}_{\text{cloudy}}(\text{SD})) = 0.70,$$

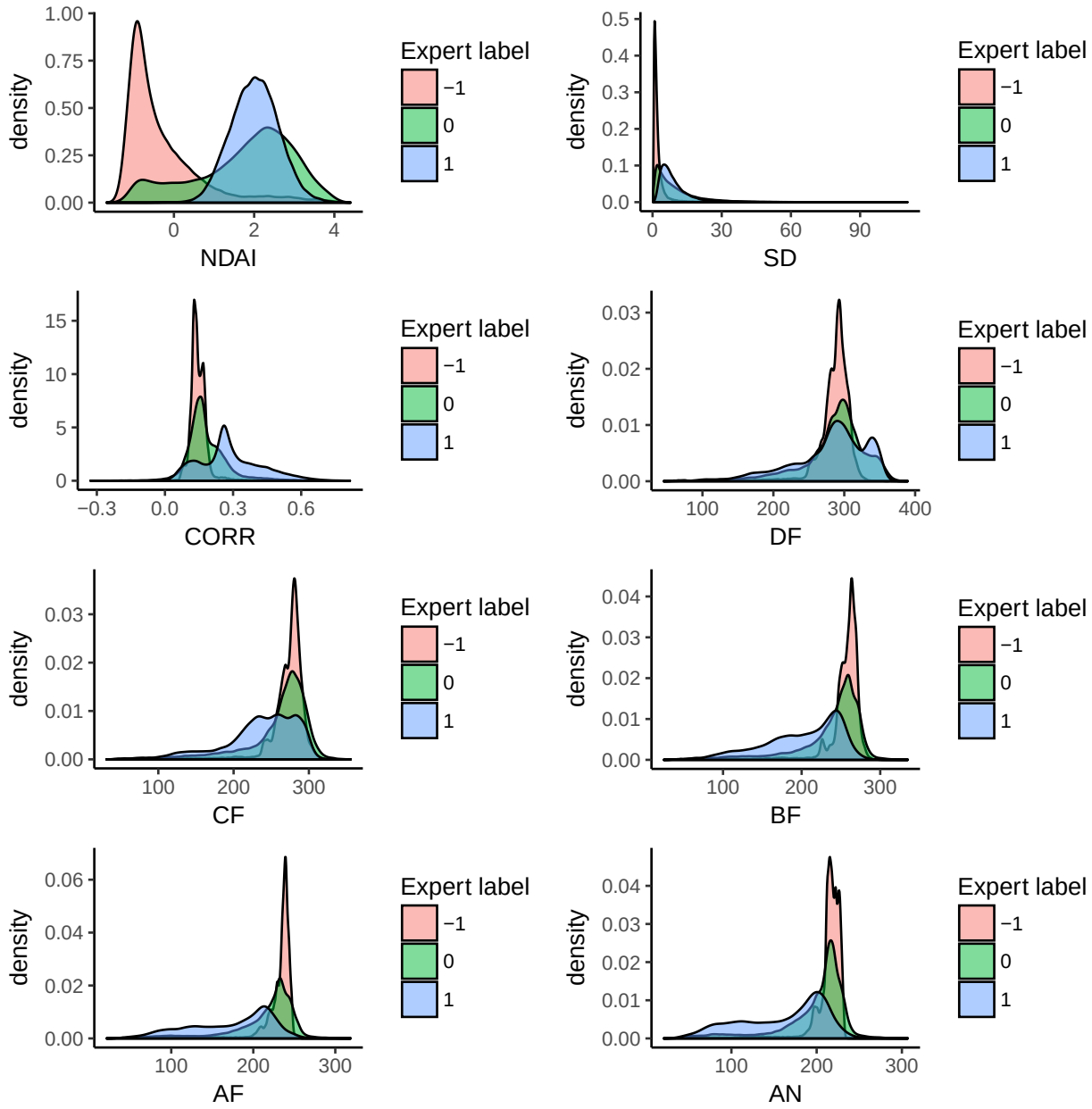


Figure 3: Kernel density estimation for different features, grouped by the expert labels. "1" corresponds to cloud, "-1" non-cloud, and "0" unlabeled.

$$\mathbf{d_{TV}}(\text{dist}_{\text{clear}}(\text{CORR}), \quad \text{dist}_{\text{cloudy}}(\text{CORR})) = 0.68.$$

For other features, this distance is generally smaller. Therefore we select NDAI, SD and CORR as predictors among those available in the dataset to train our prediction models.

4 Models

For the following models, we will frame our problem as a binary classification problem where unlabeled observations are removed from both training and testing stages. Here we explore three methods: quadratic discriminant analysis, random forest, and least directional standard deviation, or LDSD. Although these

represent just a few of the possible methods we can try, they take quite different approaches to classification, which will be described further below. Furthermore, we will use image 1 only as the training data, and leave out images 2 and 3 as the validation set.

4.1 Quadratic Discriminant Analysis

Quadratic Discriminant Analysis, or QDA, is a parametric method for estimating class probabilities. QDA is more flexible than linear discriminant analysis, or LDA, in that it does not require both classes to share a common covariance matrix, which leads to a quadratic decision boundary rather than a linear one. A quick survey of Figure 3 indicates that the variance of each feature (NDAI, CORR, and SD) is different between the classes, which suggests that QDA may be a better approach than LDA for this problem. QDA also assumes that the features from each class are normally distributed. As the normal QQ plots show in Figure 4, however, the features deviate significantly from the normal distribution. Although the normality assumption is not met, we try this model as a benchmark to compare the accuracy between various prediction models.

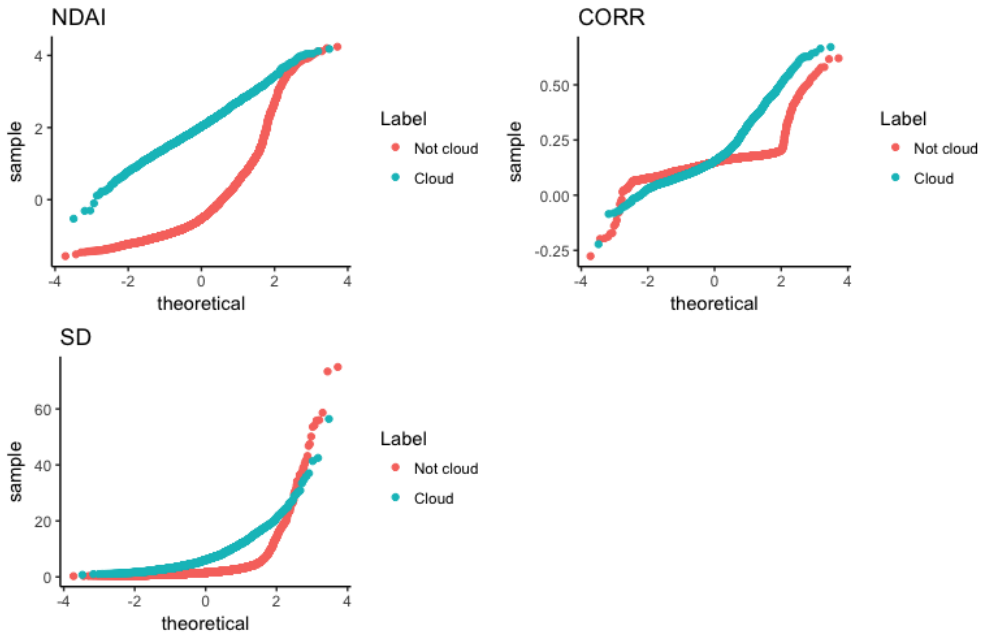


Figure 4: Normal QQ plots for NDAI, CORR, and SD for image 1. The distributions of the features deviate from the normal distribution.

4.2 Random Forest

Random forest is a non-parametric method for classification where many decisions trees are grown based on the bootstrap samples of the training data. The prediction is then based on the average of the trees. The only assumption of random forest is that the sample is representative of the population.

Random forest depends on two tuning parameters: the number of trees and the size of the random set of predictors at each split, or $mtry$. Since we have three predictors, we set $mtry = \sqrt{3} \approx 2$ as a rule of thumb. Increasing the number of trees does not overfit the data, so in practice, we choose a sufficiently large number of trees for the accuracy rate to plateau. To choose an appropriate number of trees, we first divided image 1 into 5-by-5 blocks, then performed a 5-fold cross validation on the blocks. Figure 5 shows the cross validation error on image 1 as a function of the number of trees. We can see that the error rate settles down around 300 tress. For training our model, therefore, we set the number of trees to 300.

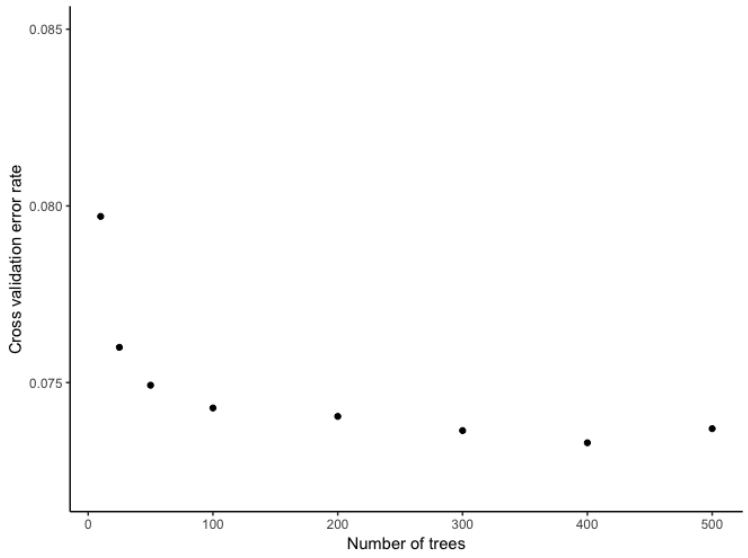
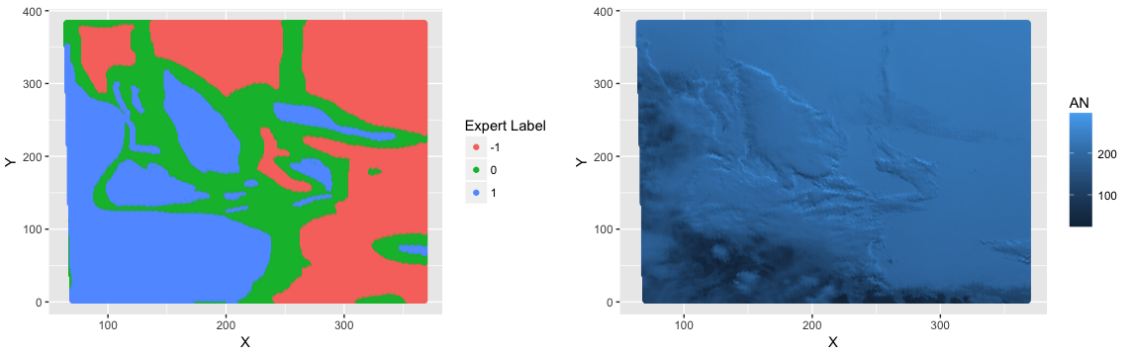


Figure 5: Cross validation error for random forest on the training data (image1) as a function of number of trees. The error rate settles down around 300 trees.

4.3 Least Directional Standard Deviation (LDSD)

Here, we explore designing a new feature. As we can see in Figure 6, AN is informative of the expert labels. The AN measurements are much smoother in the clear regions than in the cloudy regions. In fact, if we appeal to this smoothness, we can even identify the cloudy regions by eye using the plot of AN. This inspired us to cook up another feature which makes use of this spatial information of AN.



(a) Expert Label (b) AN

Figure 6: Plot of expert label (ground truth) and AN for image 2

To capture the spatial smoothness of AN, we define a new feature called Least Directional Standard Deviation (LDSD). The construction of the feature is based on a simple observation: if a pixel is indeed clear, then there should be at least one direction along which the pixels are also clear within a short range. Intuitively, this is true because the clear pixels should be connected. A clear pixel cannot be surrounded by clouds as there cannot be any holes in the clouds. Specifically, for a pixel with coordinates (x, y), its LDSD is defined as

$$\text{LDSD} = \min\{SD_E, SD_W, SD_S, SD_N\},$$

where

$$SD_E = SD(AN_{x,y:(y+9)}) = \sqrt{\frac{1}{10}\sum_{i=0}^{9}(AN_{x,y+i} - \frac{1}{10}\sum_{i=0}^{9} AN_{x,y+i})^2},$$

$$SD_W = SD(AN_{x,y:(y-9)}) = \sqrt{\frac{1}{10}\sum_{i=0}^{9}(AN_{x,y-i} - \frac{1}{10}\sum_{i=0}^{9} AN_{x,y-i})^2},$$

$$SD_S = SD(AN_{x:(x+9),y}) = \sqrt{\frac{1}{10}\sum_{i=0}^{9}(AN_{x+i,y} - \frac{1}{10}\sum_{i=0}^{9} AN_{x+i,y})^2},$$

and

$$SD_N = SD(AN_{x:(x-9),y}) = \sqrt{\frac{1}{10}\sum_{i=0}^{9}(AN_{x-i,y} - \frac{1}{10}\sum_{i=0}^{9} AN_{x-i,y})^2}.$$

This feature is in some sense similar to the SD feature, but different in important ways. For a clear pixel near the boundary of clear regions, its SD is relatively large because there are cloudy pixels in its neighborhood, and is often larger than the SD of an interior cloudy pixel. This causes the conditional distributions of SD to overlap. However, for such a pixel its LDSD is small because, say, it is clear to the east of the pixel. In fact, for images 1 and 2, we have

$$\mathbf{d_{TV}}(\text{dist}_{\text{clear}}(\text{LDSD}), \quad \text{dist}_{\text{cloudy}}(\text{LDSD})) = 0.93,$$

Figure 7 shows the kernel density estimation of this new feature and its spatial distribution. We can see that the distribution for the cloud class does not overlap much with the distribution for the non-cloud, and the spatial distribution reflects the expert labels quite well (see Figure 6 (b) for the expert labels).

With this new feature, it is possible to define an extremely simple classification rule:

$$\text{A pixel is labeled clear if LDSD} < \text{threshold}_{\text{LDSD}}.$$

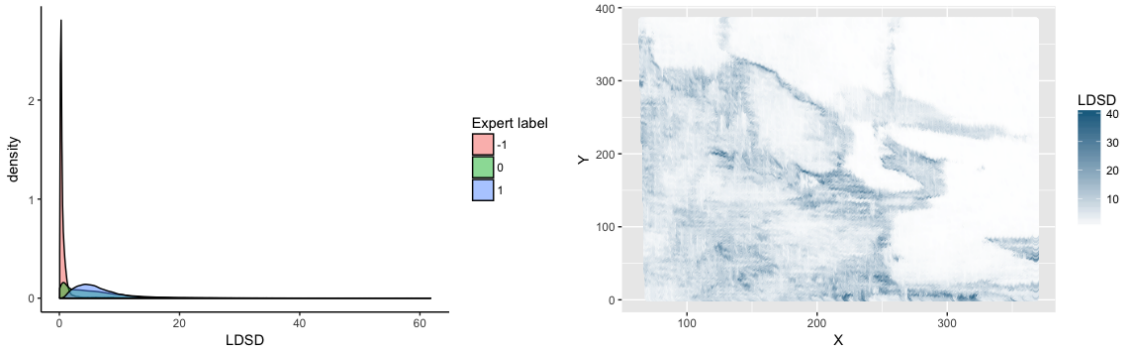
We find the cutoff $\text{threshold}_{\text{LDSD}} = 1.5$ with image 1, and test it with images 2 and 3. After predicting the labels by thresholding LDSD, we typically end up with very scattered clear regions. To overcome this issue, we use a median filter and replace the predicted label of each pixel with the median of the predicted labels in the $5\times$ 5 square centered by that pixel. This leads to an improvement in the spatial smoothness of the final prediction, as we would hope for this type of problem (Fig 8).

The final prediction accuracy on test dataset is **92.9%** for image 2 and **87.2%** for image 3. The ROC curves for images 2 and 3 are shown in Figure 9.

5 Discussion

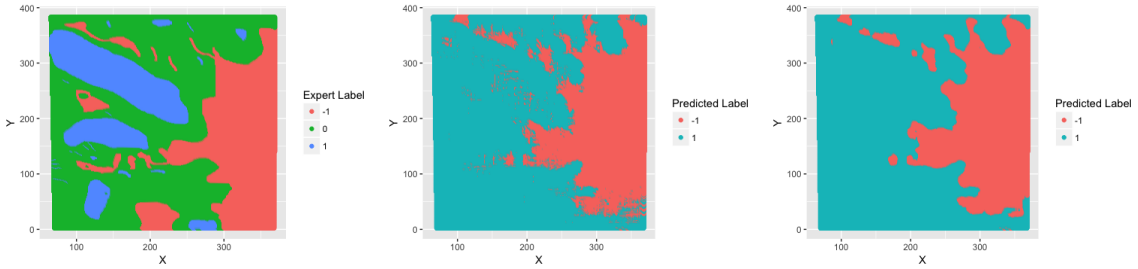
Figure 10 summarizes the performance of the different methods discussed in the previous section on the test set (images 2 and 3). As we can see, QDA performs better than others for low false positive rates (FPR $<$ .05), but LDSD begins to perform better beyond that point. If we want a reasonably high true positive rate, say, greater than .7, then LDSD would be the best choice among the different models we have tried.

Another approach we tried was preprocessing the features before training the classifiers. Two popular preprocessing methods are Principal Component Analysis (PCA) and Independent Component Analysis (ICA). PCA produces orthogonal outputs and ICA produces independent outputs, so they can be especially useful when features are strongly dependent. Motivated by the dependence between the features, we tried several preprocessing techniques (i.e. centering and scaling, power transforms, PCA, and ICA) and evaluated the effect on accuracy using a Quadratic Discriminant Analysis classifier. Using ICA, the training accuracy on image 1 increased slightly, but the testing accuracy remained constant regardless of preprocessing.



(a) Kernel density estimation (b) Spatial distribution of LDSD on image 2

Figure 7: Distribution of the new feature LDSD



(a) Truth (b) Predicted labels with LDSD (c) Smoothed predicted labels

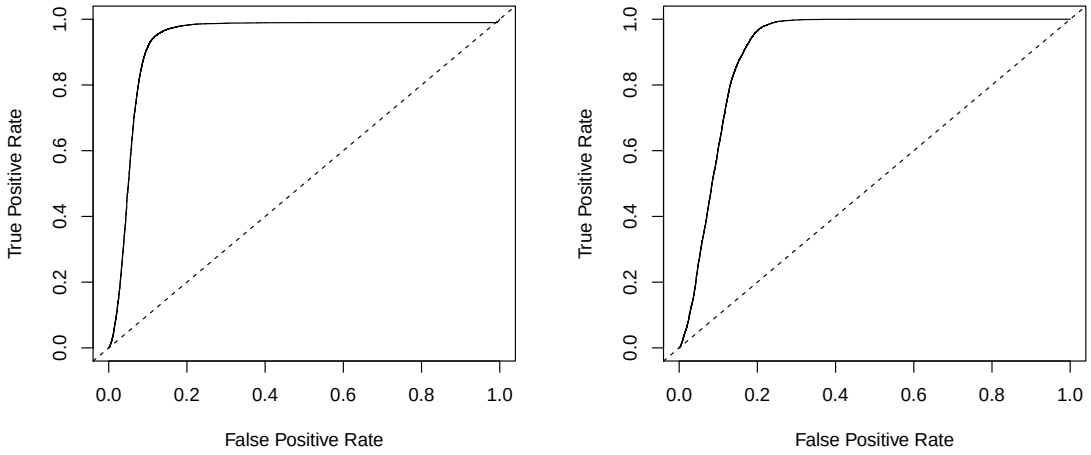
Figure 8: Prediction result for image 3, with LDSD as single feature for prediction

Since LDSD performed the best among the three models we explored, we focus on LDSD for the following discussion. We noticed certain patterns in the misclassification error. The prediction accuracy for image 3 was lower than for image 2, so we will use image 3 as an illustrative example. In general, we know from Figures 8 and 9 that LDSD successfully identified all the cloudy regions. However, it missed a clear block at the bottom of the image (with X around 200 and Y between 0 and 50). Although identified by experts as "clear," the AN plot of this region is quite non-smooth, and both NDAI and SD are quite large. As indicated in the paper [1], the fluctuation in the AN measurements is caused by incomplete terrain data registration, which usually occurs at sharp altitude changes (e.g. the coastline of Greenland). This may explain the systematic errors of our results over regions with rough topography.

We think our methods will work well for future data without expert labels. We only made use of expert labels in one image (image 1) to fit our models, and the distribution of our key features, especially LDSD, is quite homogeneous across images. Therefore we think our algorithms will provide reasonable predictions for future images from MISR.

6 Conclusion

In this report, we built various prediction models to detect cloudy regions from the MISR images of the Arctic. Having explored different types of classifiers, from parametric methods like QDA to non-parametric methods like random forest on given features, we found that designing a new feature, LDSD, performed the best, highlighting the importance of feature engineering in classification problems. And based on its performance on the test data, we predict that it would work well on future data as well.



(a) Image 2 (b) Image 3

Figure 9: ROC curves for predicting the labels of images 2 and 3 with LDSD.

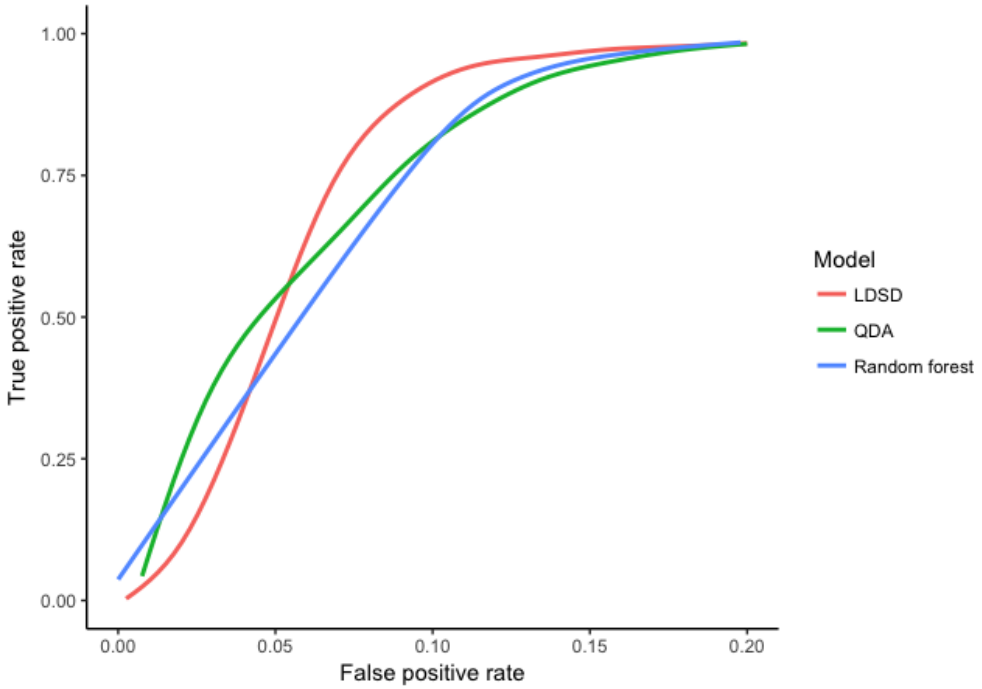


Figure 10: ROC curves for various prediction models on the test data (images 2 and 3). The results are based on non-preprocessed data.

References

[1] Shi, Tao, et al. Daytime Arctic Cloud Detection Based on Multi-Angle Satellite Data with Case Studies. Journal of the American Statistical Association, vol. 103, no. 482, 2008, pp. 584593. JSTOR, JSTOR, www.jstor.org/stable/27640081.