Lab 4 - Cloud Data Stat 215A, Fall 2018

Aummul Baneen Manasawala, Yuchen Zhang, Lei Zhang

11/08/2018

1 Introduction

The goal of this group project is the exploration and modeling of cloud detection in the polar regions based on radiances recorded automatically by the MISR sensor aboard the NASA satellite Terra. We attempt to build several prediction models to distinguish cloud from non-cloud using the available signals. The dataset we have are three image data with expert label (+1 = cloud, -1 = not cloud, 0 unlabeled), three features based on subject matter knowledge (NDAI, SD, CORR), and five multi-angel sensor data (Radiance angle DF, CF, BF, AF, AN). Finally our best fit model will be used to distinguish clouds from non-clouds on a large number of images that won't have these expert labels.

2 EDA

In order to explore the given data, we start with visulaizing the images and areas which have been labeled as having clouds or not by the experts as can be seen in figure 1. In order to see the relationship between the preence of clouds and the features, we plot the values of features on the same grid too. We find that the strong connection between the presence/absence of cloud in figure 1 and the distribution of NDAI values in figure 2 is visually apparent. Therefore from these alone we can conclude that NDAI is a prominent feature that could strongly predict the clouds in our model.

We further explore the relationships between the radiances of different angles visually which could be summarized with the boxplots in figure 3. Along with the relationships of different angles with is strong and positive, for our model it is important to know how are all the features related to the label as well as each other. From figure 4, we can see that only NDAI, SD and CORR features are positively and significantly correlated to the expert label of whether the clouds are present (numeric 0 or 1). For the five angles we can deduce that higher are those angles, lower is the probability of clouds presence. This substantiates the fact that the radiances from MISR detection have larger variance due to the physical presence of cloud. Absence of cloud make the standard deviation of the each of the randiance angles quite small. This could corroborated visually in figure 5.

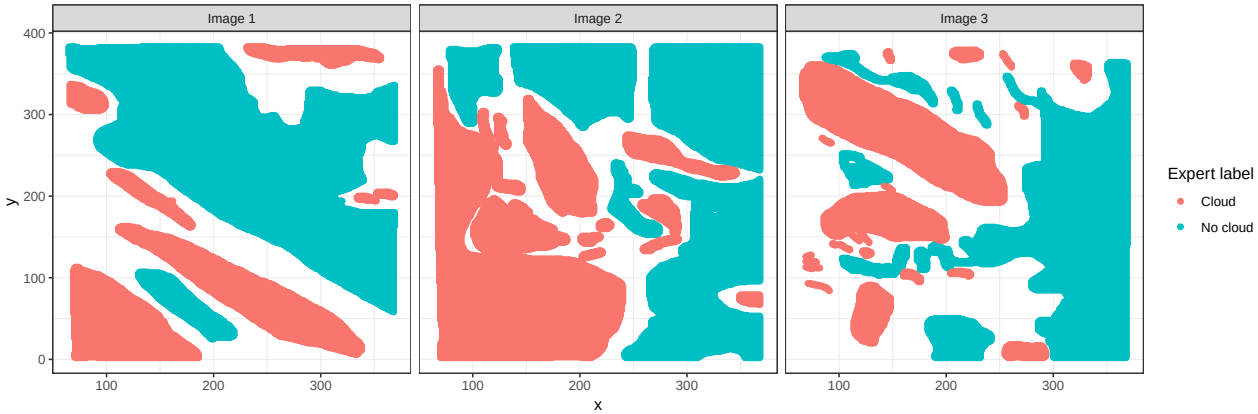


Figure 1: Visualizing the pattern of clouds in the three images using expert labels.

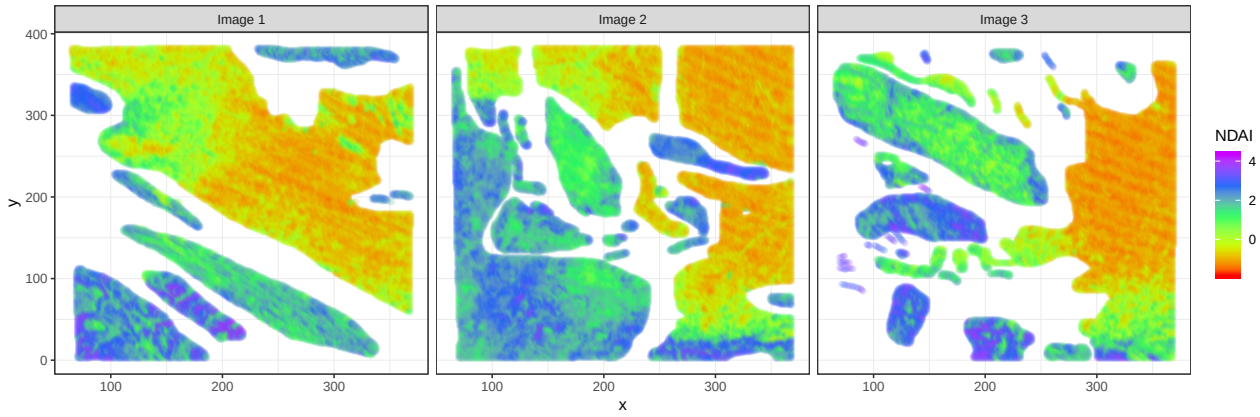


Figure 2: Visulaizing the distribution of NDAI values in the images

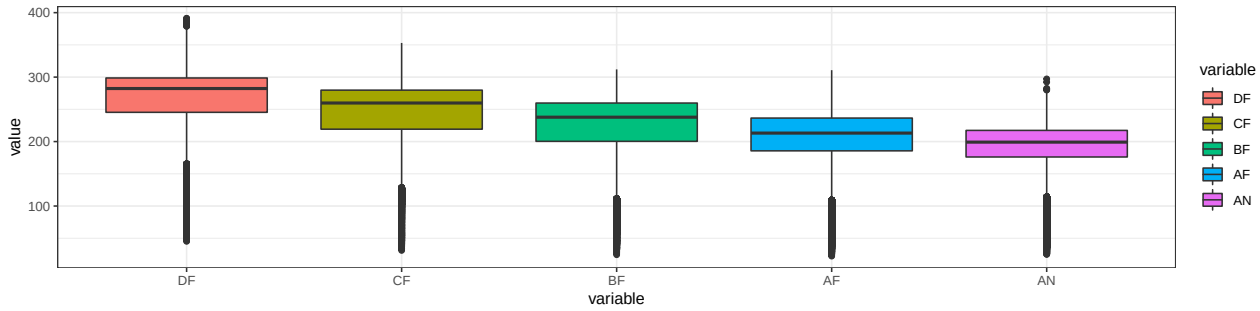


Figure 3: Relationship between the radiances of different angles

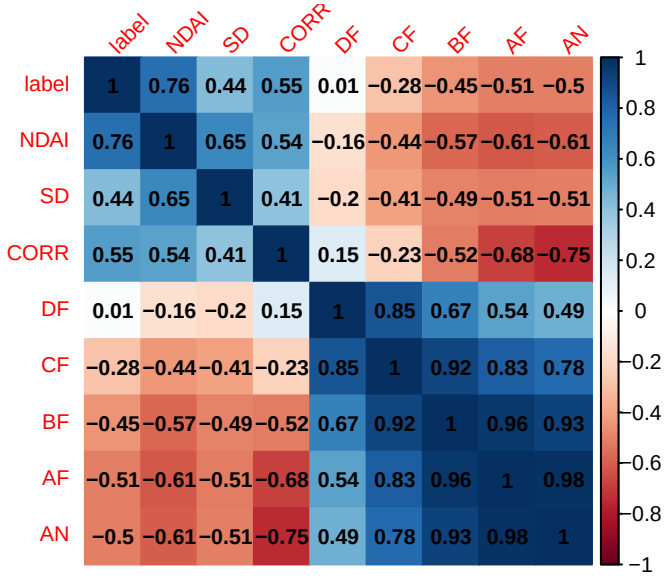


Figure 4: Quantitative relationship between the features

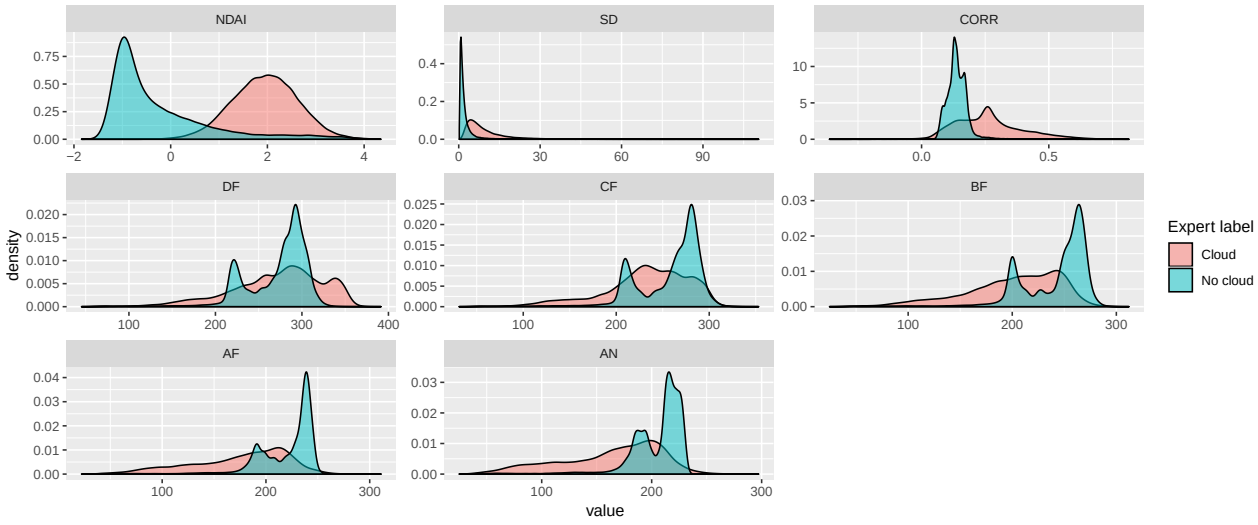


Figure 5: Difference between the feature distribution due to presence of clouds

3 Feature Selection

We estimate the minimum number of features that can make a good model to avoid the curse of dimensionality along with enhancing the interpretability of our models. To do so, we take two approaches. First we choose among a set of models of different sizes using Cp, BIC, and adjustedR2. Second, we consider doing it using the validation set and cross-validation approaches.

In the first approach, we use regression and compare the R square value of models with various number of features. We can safely use linear regression because for binary 0-1, linear regression is a pretty good approximation for logistic regression and gives an adequate classification. In the second, we split the observations in training and test set and obtain the best subset selection for each candidate model with particular number of prediction variables. Then, we extract the coefficients for the best model of that size, calculate the predictions, and compute the test mean square error(MSE). In both cases as seen in figure 6, we get an elbow at 3 which means that major variability is explained by 3 variables and adding additional variables in the model donot add significantly much to improve the model.

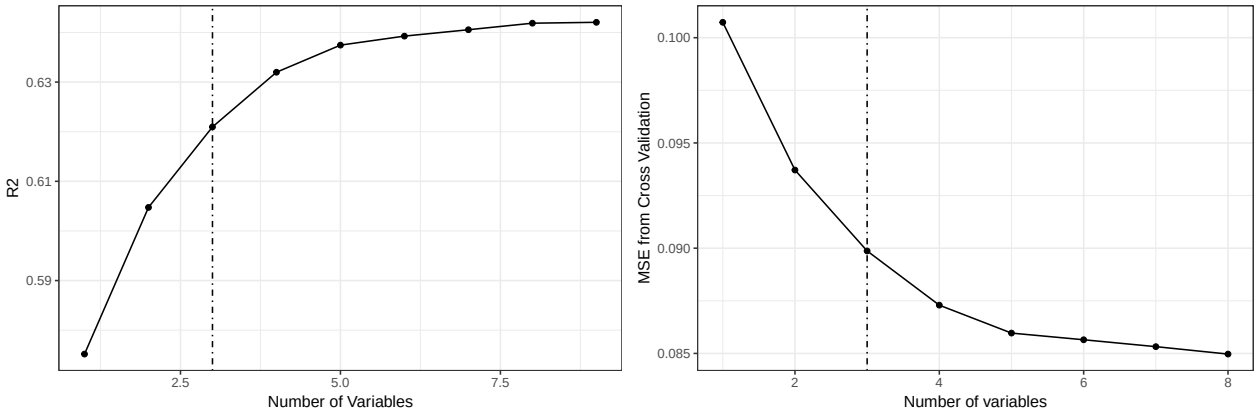


Figure 6: Selecting the number of features in the model using regression R square (left) and cross validation MSE (right)

For the purpose of selecting the features we employ two methods, best subset method in regression using BIC and Cp and random forest method to select features. We find that NDAI, CORR and SD are the best features to be incorporated in the models. We donot strictly model with only these three until and unless we have computational limitation or interpretability issue.

4 Classification Models

Before we do modeling, we first check the data quality. There is no missing value for the three image datasets. First, we combine the three image data together, remove the unlabeled data (about 40% of the data are unlabeled) and convert the label to categorical data (+1 = cloud, 0 = not cloud). Then we split the total image data into training (80%) and test (20%) set. We use the 80% training data to train our models, the six classification models we choose are Logistic regression, KNN, Decision Tree, Random Forest, Support Vector Machines, and Neural Networks. We apply cross-validation for hyperparameter optimization, then use one specific model for each kind on 20% testing set and get the prediction results. Finally we compare the six models' confusion matrix and ROC curve to select the best fit model.

4.1 Logistic Regression

We apply logistic regresion to the two models : first with all the features and second with only the three best features obtained which contribute the most to the deviations from the ANOVA test. We denote their AIC scores as AIC_1 and AIC_2 respectively. The second model($AIC_2 = 92788$) is exp((88437-92788)/2) = 0 times as probable as the first model($AIC_1 = 88437$) to minimize the information loss. Therefore, we drop the second model from consideration.

To evaluate the performance of the model, we calculate the accuracy and plot the ROC curve. The logistic model gives 89% accuracy and the AUC for the ROC curve turns out to be 0.9544.

```
## [1] "Accuracy of logistic regression: 0.890731261865283"
```

Table 1: Confusion Matrix of Logistic Regression

	Reference	0	1
Prediction			
0		23073	2259
1		2288	13993

4.2 KNN

k nearest neighbors is a simple algorithm that stores all available cases and classifies new cases by a majority vote of its k neighbors. Since the independent variables in training data are measured in different units, it is important to standardize variables in order to have the distances that are comparable for each variable. To obtain the best value for k, 10-fold validations is used as our data is in the order of 100,000 and 10 fold validation would have 10,000 data points for the test case which is sufficient for reliably test our model. As can be seen from the result, the best number of neigbours to use for our knn should be 5 to give us the most accurate results.

We find that application of knn model with k as '5' on the training data set gives us a model that is accurate 95.7%. This is quite good accuracy and the ROC curve yeilds a AUC of 0.955 makes knn as our serious prospect.

```
## [1] "Accuracy of knn: 0.932569149063994"
```

Table 2: Confusion Matrix of KNN

	Reference	0	1
Prediction			
0		24060	1505
1		1301	14747

4.3 Decision Tree

Decision tree is a supervised graph based algorithm to represent choices and the results of the choices in the form of a tree. The nodes in the graph represent an event or choice and it is referred to as a leaf and the set of decisions made at the node is reffered to as branches. Decision Tree is robust to noisy data, useful in data exploration, and its non parametric quality means it does not have any assumptions about the distribution of the variables. However one common disadvantage of decision tree is overfitting, and it is taken care of partially by constraining the model parameter and by prunning. There are many ways to measure the impurity and decide the split points, here I use Information Gain and Gini Index.

4.3.1 DT - Criterion as Information Gain

I use criterion as Information Gain for this decision tree model. I apply 10 fold corss-validation and repeated 3 times on the training set. Table 3 is the resampling results across tuning parameters. The complexity parameter (cp) is used to control the size of the decision tree and to select the optimal tree size. If the cost of adding another variable to the decision tree from the current node is above the value of cp, then tree building does not continue. Accuracy is used to select the optimal model using the largest value, therefore the final value used for the model was cp = 0.0010196.

Table 3: Trained Decision Tree classifier results

cp	Accuracy	Kappa	AccuracySD	KappaSD
0.0010196	0.9271925	0.8494600	0.0032405	0.0062783
0.0010389	0.9268801	0.8488894	0.0032290	0.0062209
0.0013286	0.9236038	0.8427275	0.0023552	0.0047459
0.0019157	0.9194643	0.8338930	0.0025032	0.0049985
0.0021242	0.9182347	0.8316037	0.0023962	0.0047182
0.0028426	0.9164163	0.8283180	0.0023272	0.0046151
0.0048510	0.9134124	0.8218261	0.0022796	0.0045218
0.0079331	0.9113797	0.8180234	0.0022325	0.0043970
0.0126528	0.9101861	0.8149914	0.0035115	0.0077842
0.7413524	0.6111158	0.0000000	0.0000139	0.0000000

Figure 7 is the visualization of my final decision tree after cross-validation. The color of the leafs shows the impurity of split and the class, the % shows the percent of data falls in this branch. We can see that the first split is NDAI < 0.66. The more dark red and more white of our leafs indicates our splits are more pure.

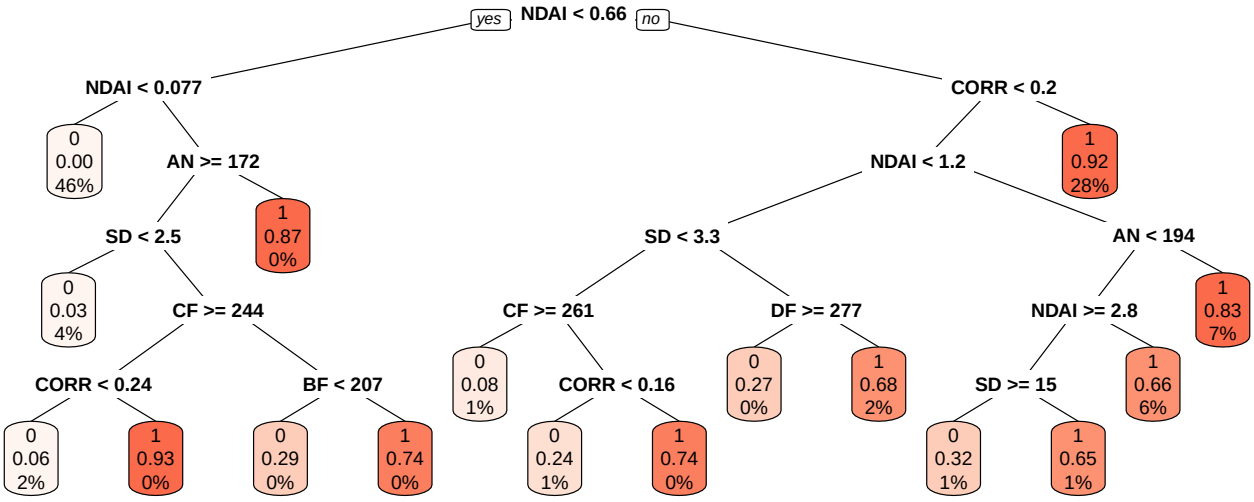


Figure 7: Plot of Decision Tree

Now I predict target variable for the test set using my final trained moded with cp = 0.0010196. Table 4 is the confusion matrix used to describe the performance of this classifier, including actual and predicted value, for example 15685 cloudy and 22708 non-cloudy points are predicted correcly. More detailed valuation (accuracy, precision, recall, ROC) will be shown in the comparison part.

Table 4: Confusion Matrix of Decision Tree (Infomation Gain)

	Reference	0	1
Prediction			
0		22708	567
1		2653	15685

4.3.2 DT - Criterion as Gini Index

I use criterion as Gini Index for this decision tree model, the other parameters remain the same. I use cross-validated (10 fold, repeated 3 times) resampling method, the final model with largest accuracy is when cp = 0.0010196, which is exactly the same as Information Gain method. Table 5 shows the confusion mateix of Gini Index Decision Tree.

Table 5: Confusion Matrix of Decision Tree (Gini Index)

	Reference	0	1
Prediction			
0		23282	986
1		2079	15266

4.4 Random Forest

Random Forest is a ensemble learning method that combines multiple trees as opposed to a single decision tree to form a powerful model. In the process it reduces dimensionality, removes outliers and treats missing values. Here we build 500 decision trees using Random Forest. Figure 8 shows the plot of error rate across decision trees. The plot seems to indicate that after 200 decision trees, there is not a significant reduction in

error rate.

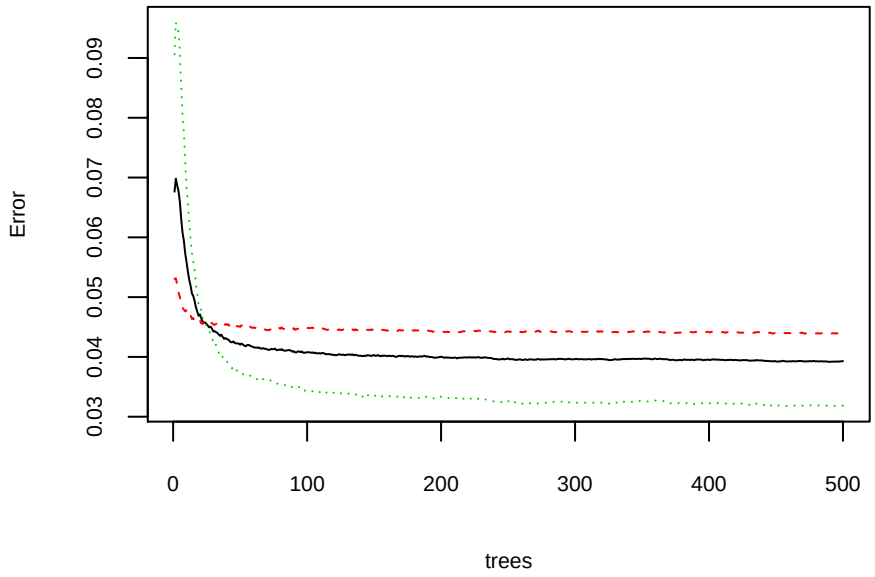


Figure 8: Plot of Random Forest

Figure 9 is variable importance plot. Top 5 variables are selected and plotted based on Model Accuracy and Gini value (node impurity). We find that first three features are fixed for two methods, which are NDAI, SD, CORR in feature importance descending order. Table 6 shows the confusion matrix of random forest algorithm on test data. The corrected classified region points increase a little compared with two decision tree models.

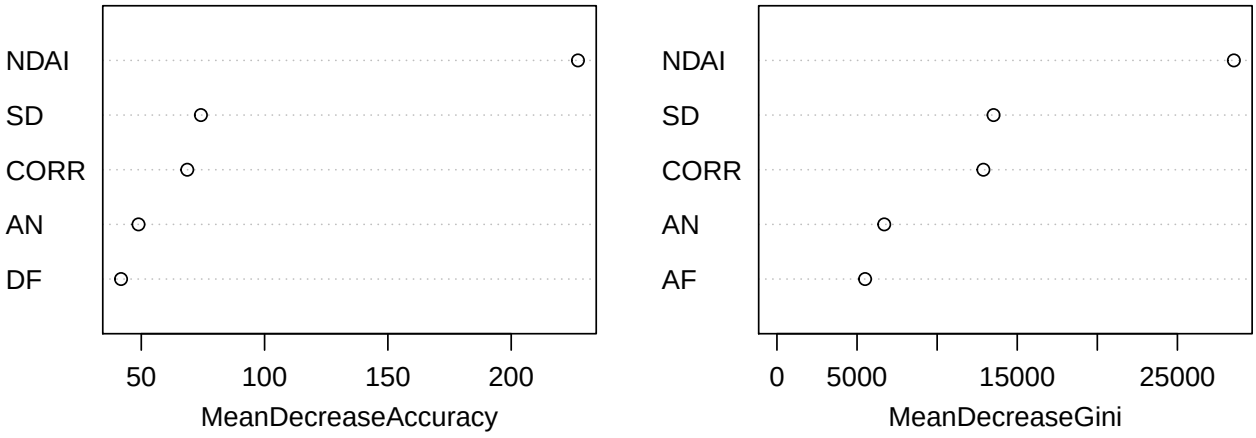


Figure 9: Variable Importance Plot for Random Forest

Table 6: Confusion Matrix of Random Forest

	Reference	0	1
Prediction			
0		24223	484
1		1138	15768

4.5 Support Vector Machine

Support Vector Machines (SVMs) is a data classification method that separates data using hyperplanes. One observation about classification is that in the end, if we only care about assigning each data point a class, all we really need to know do is find a "good" decision boundary, and we can skip thinking about the distributions. Support Vector Machines (SVMs) are an attempt to model decision boundaries directly in this spirit.

If we have labeled data with dimension d, SVMs can be used to generate $d-1$ dimensional hyperplanes such that the data space is divided into segments and each segment contains only one kind of data.

Due to the very high computational power requirement, it is not realistic to run cross validation or using the whole dataset. Here I randomly sample 10000 data points as the training set and another 2000 points as the test set to explore the performance of SVMs.

Table 7: Confusion Matrix of SVM

	Reference	0	1
Prediction			
0		1102	35
1		125	738

```
## [1] "Accuracy of SVM: 0.92"
```

4.6 Neural Networks

The basic idea of a neural network is that we are going to Combine input information in a complex & flexible neural net "model". Model "coefficients"" are continually tweaked in an iterative process. The network??s interim performance in classification and prediction informs successive tweaks.

A neural network consisits of three layers: 1. Input layers: Layers that take inputs based on exising data 2. Hidden layers: Layers that use backpropagation to optimize the weights of the input variables in order to improve the predictive power of the model. 3. Output layers: Output of predictions based on the data from the input and hidden layers.

There is no model assumption for neural network. But a neural network has a high requirement for computaion power. Again, it is not realistic to run cross validation or using the whole dataset. So I used the same dataset in the SVMs part to run the neural network and evaluate its performance.

One of the most impotant procdures when forming a neural network is data normalization. This has been done in the very first step of data restructuring.

```
## [1] "AIC of Neural Networks: 673.585988441974"
```

```
## [1] "BIC of Neural Networks: 788.951434393593"
```

Table 8: Confusion Matrix of Neural Network

	Reference	0	1
Prediction			
0		1093	43
1		134	730

```
## [1] "Accuracy of Neural Networks: 0.9115"
```

4.7 Model Comparison

After applying different classification algorithms, we find that some of the features might be better predictors of the presence of clouds than others. Table 7 shows the feature importance for different models, DT is scaled to 100 and RF is not. We find that the three of the best features are NDAI, SD and CORR for all of the algorithm models, followed by two radiance angel AN and AF, the rest three angel are far less important.

Table 9: Comparision of Feature Importance

Vars	DT_Info	DT_Gini	RF
NDAI	100.0000000000	100.0000000000	28523.692607
SD	70.2185453117	70.6493537701	13522.260760
CORR	68.0230992355	72.7075497645	12887.827170
AN	36.7467756161	40.3628065862	6698.125456
AF	36.6215224724	34.4166597205	5501.974657
DF	2.6963007119	3.3800405857	4488.176728
BF	0.1621103783	0.1595103507	4298.863615
CF	0.0000000000	0.0000000000	3189.948673

In order to choose the best fit model, model evaluation is done based on testing dataset. Table 8 shows the detailed measures in confusion matrix, eg. $accuracy = \frac{TP+TN}{P+N}$, $sensitivity = racall = \frac{TP}{P}$, $specificity = \frac{TN}{N}$, $precision = \frac{TP}{TP+FP}$. Random forest has higher accuracy compared with single decision tree model. Figure 10 shows the ROC curve for different models, the larger the area (AUC), the better the performance. The balanced accuracy for DT-Info, DT-Gini, RF, logistic, Knn, SVM and Neural Networks are 0.930, 0.929, 0.963, 0.885, 0.928, 0.926 and 0.918 respectively. In conclusion, we choose random forest as our best classification model.

Table 10: Comparision of Confusion Matrix

	DT_Info	DT_Gini	RF	Logit	Knn	SVM	NN
Sensitivity	0.965	0.939	0.970	0.861	0.907	0.951	0.944
Specificity	0.895	0.918	0.955	0.910	0.949	0.905	0.891
Pos Pred Value	0.855	0.880	0.933	0.859	0.919	0.863	0.845
Neg Pred Value	0.976	0.959	0.980	0.911	0.941	0.967	0.962
Precision	0.855	0.880	0.933	0.859	0.919	0.863	0.845
Recall	0.965	0.939	0.970	0.861	0.907	0.951	0.944
F1	0.907	0.909	0.951	0.860	0.913	0.905	0.892
Prevalence	0.391	0.391	0.391	0.391	0.391	0.387	0.387
Detection Rate	0.377	0.367	0.379	0.336	0.354	0.367	0.365
Detection Prevalence	0.441	0.417	0.406	0.391	0.386	0.426	0.432
Balanced Accuracy	0.930	0.929	0.963	0.885	0.928	0.928	0.918

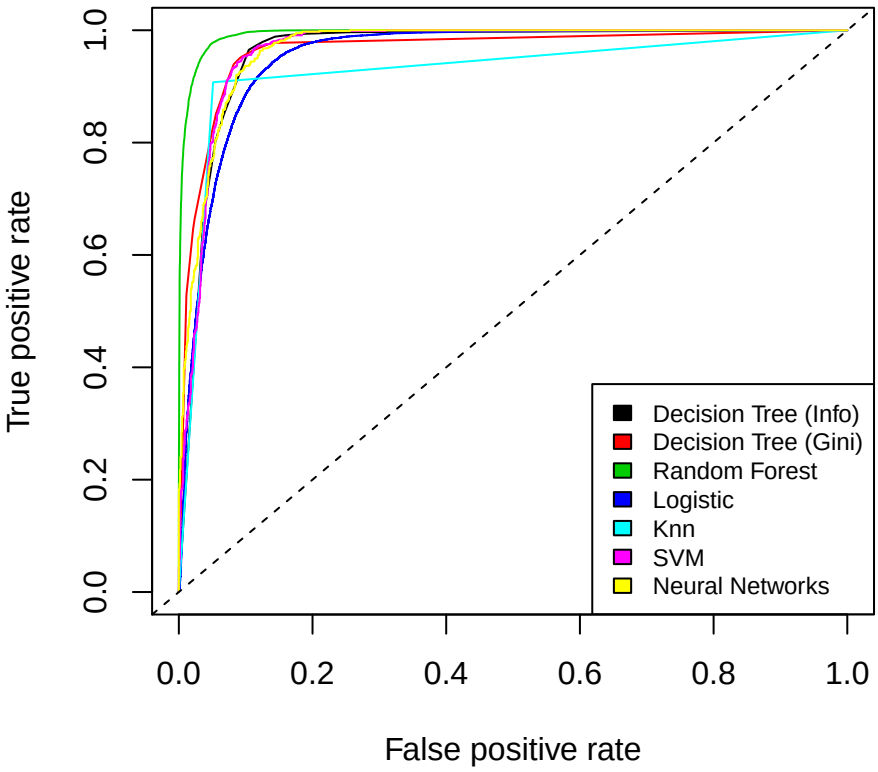


Figure 10: Test Set ROC Curves

5 Performance of Best Model

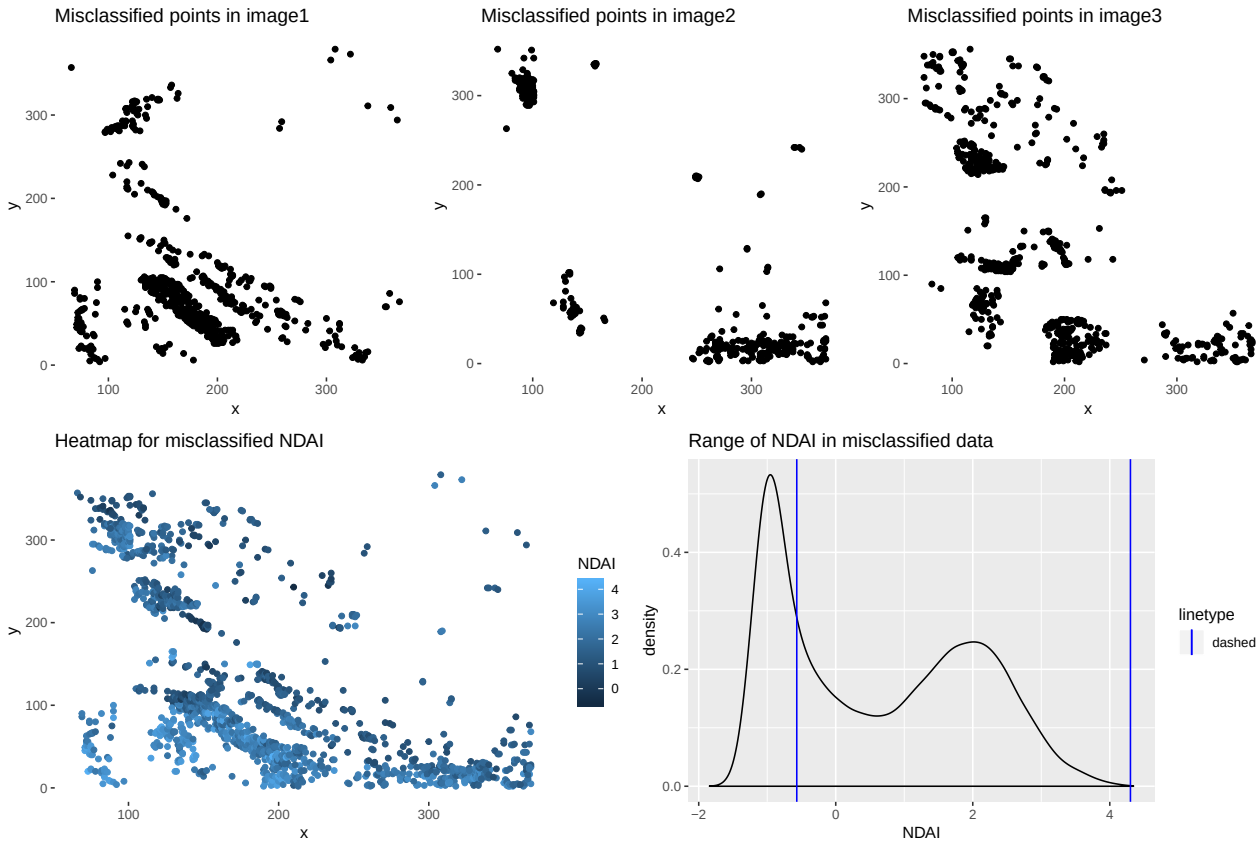
5.1 Misclassification Patterns

In this part, we choose our best-performance model "Random Forest" to do the misclassification Analysis. Based on the predicted labels from Random Forest, we figure out there are 1599 observations in the test dataset are misclassified. And 1128 observations are false positive, while 471 observations are false negative. This indicates a higher probability for misclassifying ice area to cloud than misclassifying cloud to ice area.

We then take out all the covariates of these misclassified observations and compare them with the correctly-labeled ones.

Based on two sample t-tests, when compared with correctly-labeled observations, the misclassified observations have higher NDAI, SD and lower DF, CF, BF, AF, AN with p-value less than e-16. CORR doesn't seem to have relationship with the misclassification.

We also include the visualization of misclassified points in the analysis.



5.2 Prediction on Unlabeled Data

Here I use our best fit model (Random Forest) to predict the unlabeled observation data and then combine the predicted data with the labeled ones and plot the pixel-level three image data to distinguish clouds from non-clouds. Figure 11 shows the results of classification, compared with Figure 1, we can see that the white part in Figure 1 are classificied as either cloud or no cloud in Figure 11. We can see clear patterns in our figure, which means our classification for unlabeled area are reasonable.

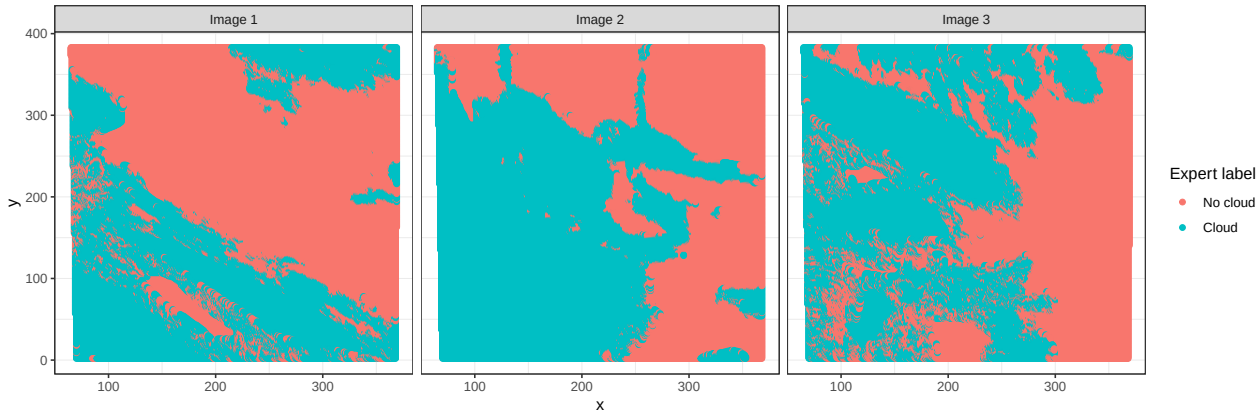


Figure 11: Plot of Expert Pixel-level Classification

6 Conclusion

Running six models to classify the data, we found that each model had some pros and cons. Some were simpler to compute but this came with the loss in prediction accuracy like in case of logistic regression. Some models were accurate but missed interpretability as in the case of neural networks. Some like logistic regression were best measured by goodness of fit parameters in classical statistics, others like random forest and neural networks used prediction accuracy measures that are followed in the modern statistics or machine learning. Most models quite accurately predicted the areas of cloud from the non cloud. Hence, we are confident about our final model to predict the clouds on new images satisfactorily.