Project 2

Jonathan Stuart, Nikhil Sakhamuri

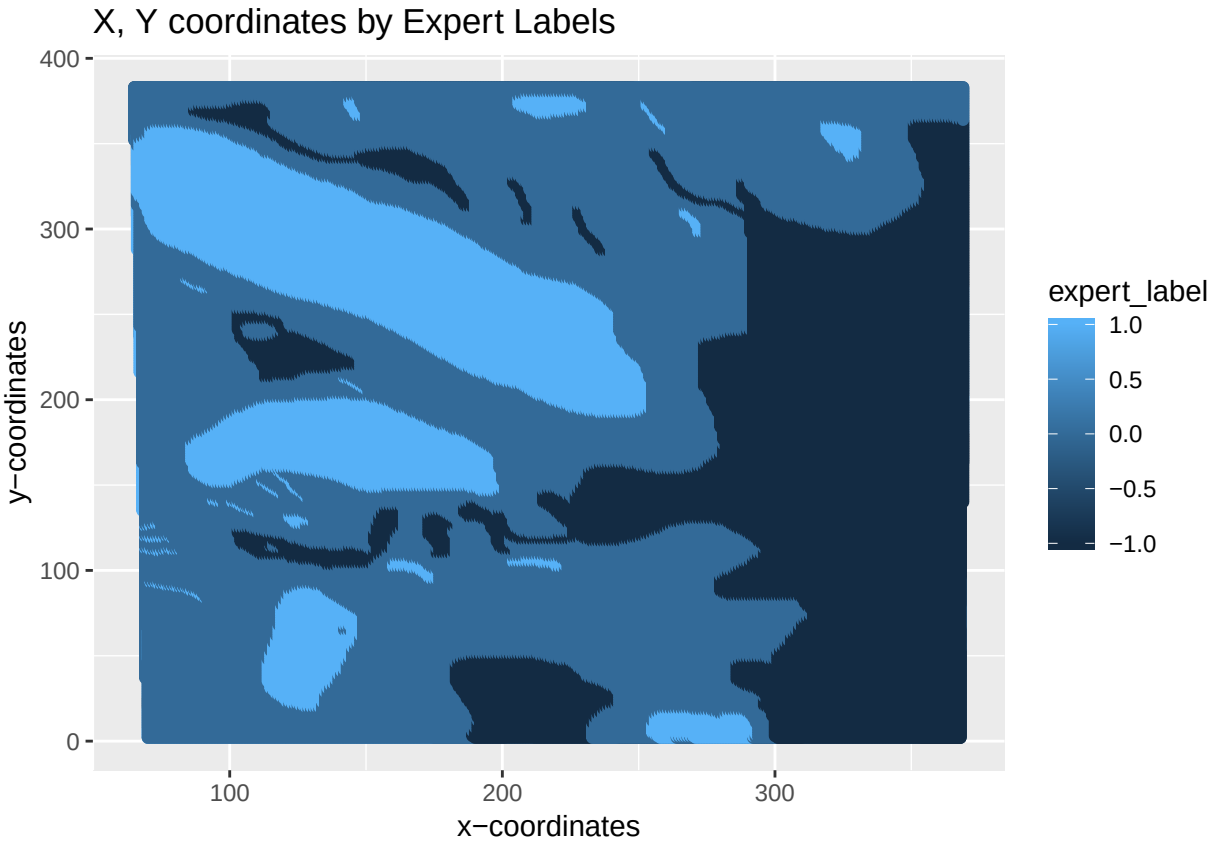
For at least the past 40 years, climate change and its potential impact on the Earth and the species that inhabit it have been hotly debated. Over the past decade, scientists have proven beyond a reasonable doubt that large scale, epoch making changes are underway with respect to the Earth and its climate. As an extremely dynamic system, the changes experienced by Earth's climate rest heavily on a number of interactive dependencies, the presence or absence of which could have measurable effects on the rate of increase of surface temperatures. Cloud coverage in the Arctic regions has been shown to be one such dependency, contributing significantly to rising sea levels through the role they play in helping to buffer the impact of rising temperatures the Arctic. Thus, directly, the ability to identify patterns of cloud coverage in the arctic via satellite imagery bears directly on the climate change debate.

Because of the similarities in how snow and clouds interact with electromagnetic radiation, identifying areas of cloud coverage in arctic regions poses an interesting problem. To solve this classification problem, Shi, Yu, et al. set out to build "cloud detection algorithms that can efficiently process the massive [Multiangle Imaging SectroRadiometer] dataset... without requiring human intervention." The MISR has 9 cameras at 9 different angles each taking images over four regions of the EM spectrum. The MISR collects an average of 3.3 megabits of data per second over 233 distinct but overlapping 360km wide geographic paths around the Earth. Dealing with such large amounts of data requires that images of some spectra be transmitted at full resolution while others are transmitted at a lower resolution. For their study, Shi, Yu, et al. used a collection of 10 MISR orbits of path 26 over the Arctic region.

Shi, Yu, et al. used three features selected through EDA and domain knowledge on which to build an enhanced linear correlation matching (ELCM) algorithm. They then predicted the probability of cloudiness by training Fisher's QDA on the labels outputted by the ELCM algorithm. With 100% coverage of the pixels for which a label is provided and 91.8% agreement with the expert labels of cloudiness classification, the ELCM method developed by Shi, Yu, et al. far outperformed the other classification algorithms under consideration. Further, the ELCM-QDA regime went beyond the binary labels of the ELCM algorithm by providing probability labels. Ultimately concluding that the three selected variables contained sufficient information to correctly classify cloud cover in arctic images, the methods employed by Shi, Yu, et al. proved impactful. In addition to contributing to the growing body of Earth science data with implications for long-standing problems like disaster forecasting and global food supply, Shi, Yu, et al. also demonstrated the ability of statistical thinking to help solve humanity's most pressing problems.

Exploratory Data Analysis

Through examining a series of summary tables we found that, overall, considering all three images, approximately 40% of pixels were unlabelled, and more pixels than not were labelled non-cloudy vs. cloudy at 37% and 23% respectively. Digging deeper, variations in the distribution of labels for each image. In image 3, for example, approximately 52% of pixels were unlabelled while image 2 had slightly more than half as many as unlabelled, 28% indicating that image 2 had significantly more labelled pixels. Among all three images considered individually, there were more not cloudy pixels than cloudy, the biggest difference being found in image 1 at 44% and 18%, respectively.



Plotting the X and Y coordinates and filling regions based on the expert labels, we do see a pattern emerge. Clearly visible are three distinct regions of cloud cover along with a distinctly not cloudy region.

Next, a series of pairwise scatterplots, boxplots and density plots helped us to vizualize the relationship between each of the variables, the radiance angles and the expert labels. Through the scatterplots, most notably we see that the not cloudy label is more prevalent for pixels with negative ndai values and low corr values, indicating predictive power. Between all the radiances, the plots were similar. Two are included for illustraion.

The variable ndai and the radiance angle df provided the most and least indicative boxplots with respect to the expert labels. As we can see from the plots, again we notice that negative ndai values indicate not cloudy, while radiance angle df alone provides no distinction between lables.

Through the density plots, we recognize a multi-modal character of the radiance angle not cloudy label and again see the clear distinctions of cloudy vs. not cloudy in the ndai and corr plots.

Data Preparation

Recognizing that the data are not i.i.d., we came up with the following two ways of splitting the data into training, validation and test sets. The first method of splitting the data is to take validation and test sets from one image file, using the rest of the data to train. The second method involved randomly selecting the test data, using the x and y coordinates to select the validation data and using the rest of the data to train. We took this approach because, during the exploratory data exploration phase of the project, we noticed patterns between the labels and ranges of coordinates.

After running a trivial classification, we achieved 17.67% classification accuracy.

Referring back to the data exploration section of this report, because of the interactions between the ndai an corr variables, we believe there to be great predictive power in them. Clearly stated, our criteria for deciding on those two as the best features is based on the fact that negative ndai values are very much correlated with

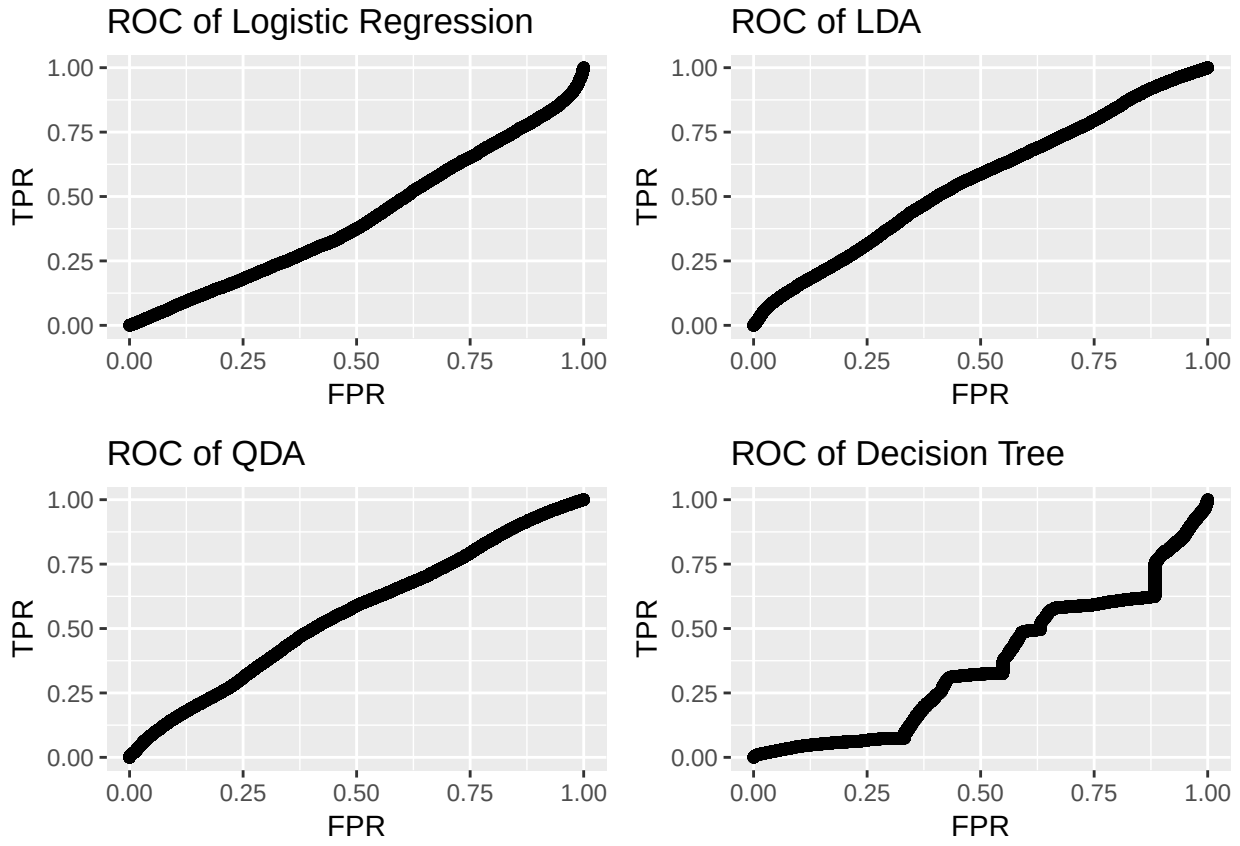


Figure 1: ROC Curves

the not cloudy label, and on the fact that corr values closer to zero are similarly coordinated. Beyond that, we chose also the radiance angle cf as a best feature because the peak of the cloudy label corresponds with a valley of the not cloudy label and vice versa.

Modelling - First Data Split

After running logistic regression, LDA, QDA and decision tree models, we obtained the following results: The training error for logistic regression across all 10 folds was 0.1099933, 0.1075707, 0.1149731, 0.1086474, 0.108782, 0.1097914, 0.1107335, 0.1097241, 0.110498, 0.1126851, and this model had an average loss of 0.1103398. The training error for LDA across all 10 folds was 0.1066285, 0.1057201, 0.1113055, 0.1052153, 0.1048116, 0.106965, 0.1072678, 0.105821, 0.1074024, 0.1100269, and this model had an average loss of 0.1071164. The training error for QDA across all 10 folds was 0.1084791, 0.1060565, 0.1116083, 0.1059892, 0.1070323, 0.109825, 0.1088829, 0.106965, 0.1093876, 0.1105653, and this model had an average loss of 0.1084791. The training error for the Decision Tree Model across all 10 folds was 0.1089838, 0.112214, 0.1142328, 0.1120458, 0.1127187, 0.1140983, 0.1136272, 0.1113728, 0.1116756, 0.115074, and this model had an average loss of 0.1126043.

After training models on all training data then predicting on test values, we obtained the following results: The test error for logistic regression was 0.1195128, The test error for lda was 0.0993675, the test error for qda was 0.0843757 and the test error for the decision tree was 0.1089717

```
grid.arrange(plot1, plot2, plot3, plot4, ncol=2)
```

Between both splits of data, the QDA model on the first split has the most optimal point on the ROC curve, when maximing for Sensitivity of Specifiticity -1. The point at FPR ~ 47 is this optimal point with Sensitvity

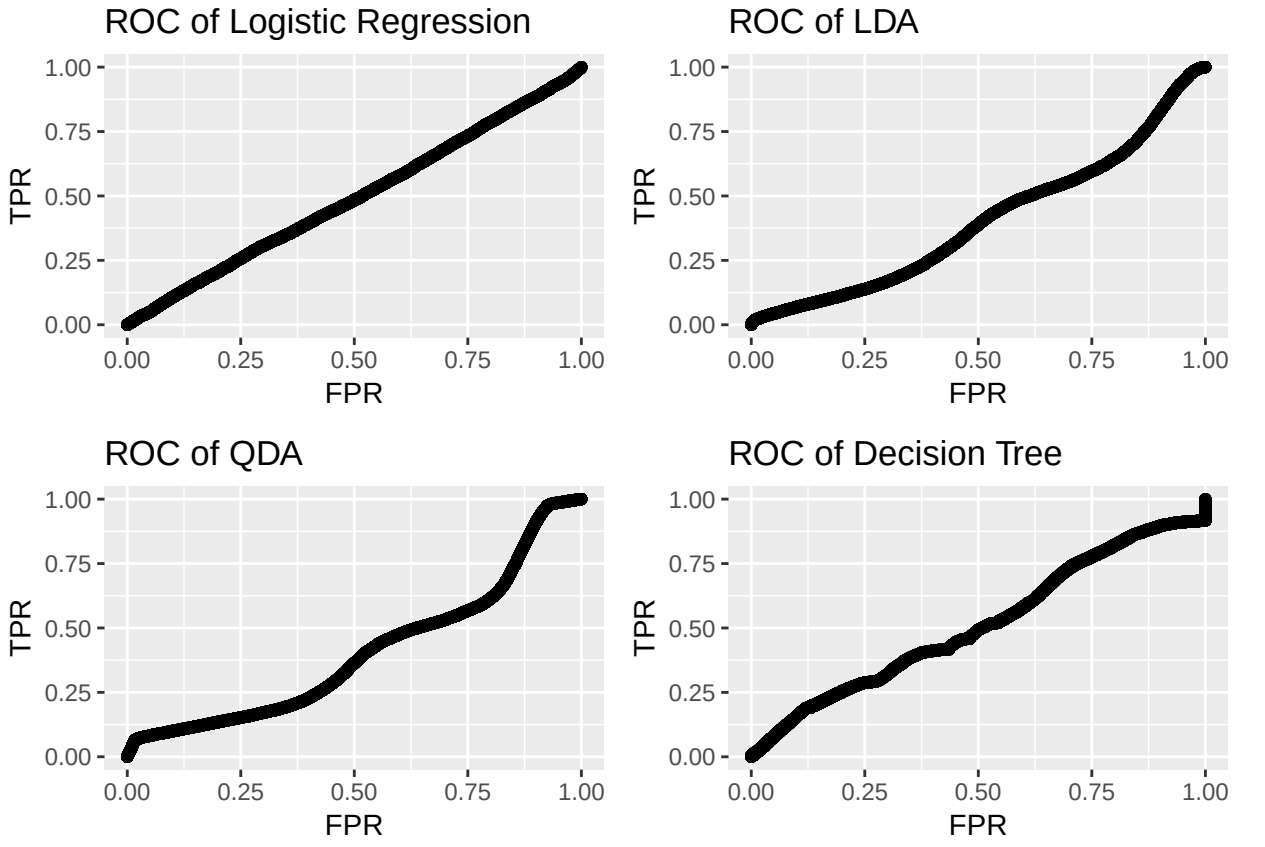
~56 and Specificity = ~56.

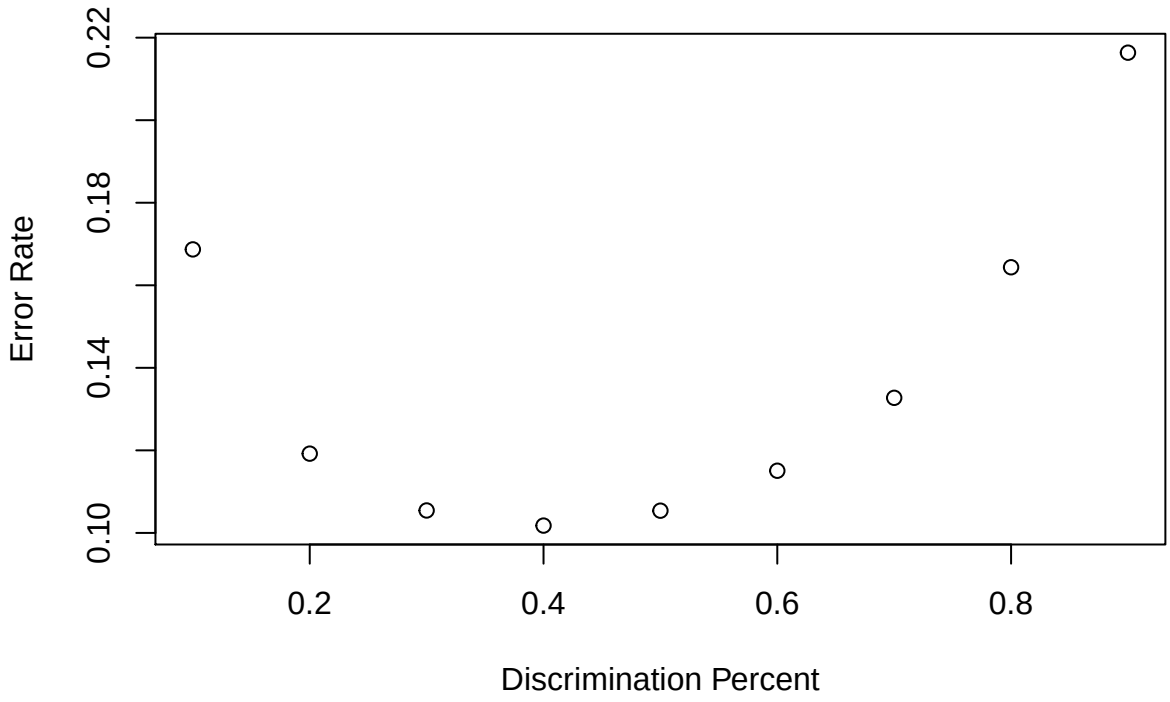
Modelling - Second Data Split

After running logistic regression, LDA, QDA and decision tree models, we obtained the following results: The training error for logistic regression across all 10 folds was 0.0964543, 0.0978569, 0.0989334, 0.0965522, 0.0969436, 0.0960955, 0.0957041, 0.0971719, 0.0982157, 0.1003034. This model had an average loss of 0.0974231. The training error for LDA across all 10 folds was 0.0909417, 0.0934208, 0.0931272, 0.0897022, 0.0913005, 0.0898327, 0.0902567, 0.0926379, 0.0929641, 0.0930293. This model had an average loss of 0.0917213. The training error for QDA across all 10 folds was 0.0919529, 0.0945624, 0.0963891, 0.0903219, 0.0942362, 0.0920181, 0.0919529, 0.0943667, 0.0962912, 0.0950191. This model had an average loss of 0.0937111. The training error for the Decision Tree Model across all 10 folds was 0.0857227, 0.0878429, 0.0874189, 0.0826239, 0.0867991, 0.084842, 0.0840917, 0.0862446, 0.0878755, 0.0864729. This model had an average loss of 0.0859934.

After training models on all training data then predicting on test values, we obtained the following results: The test error for logistic regression was 0.1053873, the test error for lda is 0.0985027, the test error for qda is 0.0985509 and the test error for the decision tree is 0.0927254

```
library(gridExtra)
grid.arrange(plot1, plot2, plot3, plot4, ncol=2)
```






```
##b.
confusionMatrix(as.factor(log_pred[,1]), as.factor(test$expert_label))
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction     0     1
##          0 11405   996
##          1  1305  7065
##
##                Accuracy : 0.8892
##                  95% CI : (0.8849, 0.8935)
##     No Information Rate : 0.6119
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.7684
##
##  Mcnemar's Test P-Value : 1.355e-10
##
##             Sensitivity : 0.8973
##             Specificity : 0.8764
##          Pos Pred Value : 0.9197
##          Neg Pred Value : 0.8441
##              Prevalence : 0.6119
##          Detection Rate : 0.5491
##    Detection Prevalence : 0.5970
##       Balanced Accuracy : 0.8869
##
##        'Positive' Class : 0
##
```

Diagnostics - First Data Split
A good classification model found on the first split of data was logistic regression. Overall, this first split was less effective than the second split; below I will analyze the logistic regression model on this split. First we attempted to optimize logistic regression by seeking a more optimal point than 0.5 to use as the decision boundary for the two classes. The optimal discriminative point was determined to be 0.4 rather than 0.5. However, the margin was close enough that the difference was trivial so we decided to keep 0.5 as the discriminative point.

After adding an L1-regularization to the data we graphically analyzed the different lambda values and visually determined a good lambda value. Even with the L1-regularization, however, the error rate was worse than before, without it. It seems that neither L1-regularization nor changing the discrimination percentage helped much with lowering the test error rate and in fact, L1-regularization was counterproductive with this first split of data.

Finally, by analyzing the Confusion Matrix, it is clear that this model classified most points as class 0. This turned out to be the correct decision as it was correct 1.2483314 times. This model was also much more prone to making false positives than false negatives, which ties into the propensity of this model to classify points into class 0.

Diagnostics - Second Data Split
A good classification model found was logistic regression on the second split of data. Again, we attempted to optimize logistic regression by seeking a more optimal point than 0.5 to use as the decision boundary for the two classes. As with the first split of data, it appeared as if a Discrimination Percent of 0.4 is slightly better than 0.5. However, for the same reason, we ignored this anomaly. We again attempted to reduce test_error with L1-regularization.

Again, after adding an L1-regularization to the data we graphically analyzed the different lambda values and visually determined a good lambda value.

We found that even with the L1-regularization, we still got approximately the same error rate as before, without it. It seems that neither L1-regularization nor changing the discrimination percentage helped with lowering the test error rate.

The Confusion Matrix and summary statistics gave an analysis of the misclassification error of the logistic regression model. There seems to be an even split of false negatives and false positives indicating no tendency of the model towards either type of error. This logisitic regression model on the second split of data has a much higher specificity (about double) that of the model created by the first split of data while only having .06 less sensitivity. It is clear that this second model is the superior one.

In conclusion, our analysis on parts a and b was largely fruitless for both splits of data. Although we made several interesting findings regarding our model (that a discriminative value of 0.4 was optimal to 0.5, and the optimal regularization paramter for L1-regularization), we failed to find anything that largely decreased my model's test error rate. Conversly, adding L1-regression largely increased test error for the first data split. On the positive side, we believe the logistic regression model will work well on future data. For the model made on the second split of data, on 95% of sets of test data, the prediction accuracy will be between 88.45& and 89.31%. This is a strong accuracy value. Overall the second split of data proved to be a much better split in terms of predictive power. All models realized a lower test error rate in the second split than the first.