STAT 154: Project 2 Cloud Data

Shrey Samdani: 3032000414

Man Chong Chan(Aaron): 3033281903

1 Data Collection and Exploration (30 pts)

(a) **Write a half-page summary** of the paper, including at least the purpose of the study, the data, the collection method, its conclusions and potential impact.

This article proposes two new operational Arctic cloud detection algorithms: ELCM and ELCM-QDA using Multiangle Imaging SpectroRadiometer (MISR) imagery. The key idea is to identify cloud-free surface pixels in the imagery instead of cloudy pixels as in the existing MISR operational algorithms. The data used in this study were collected from 10 MISR orbits of path 26 over the Arctic, northern Greenland, and Baffin Bay. Path 26 was chosen for the study because it includes permanent sea ice in the Arctic Ocean, snow-covered and snow-free coastal mountains in Greenland, permanent glacial snow and ice, and sea ice that melted across Baffin Bay over the study period. Through extensive exploratory data analysis and using domain knowledge, three physically useful features: CORR, SD, NDAI have been identified, which are vital to the new algorithm. The ELCM algorithm based on the three features is more accurate and provides better spatial coverage than the existing MISR operational algorithms for cloud detection in the Arctic. Moreover, results from the ELCM algorithm can be used to train QDA to provide probability labels for partly cloudy scenes.

In this study, unlike the past, statisticians are directly involved in the data processing. The success of the study was achieved by the collaborations between atmospheric scientists, statisticians, and the MISR science and instrument teams at the Jet Propulsion Laboratory. This demonstrates the contribution of statisticians in the analysis of the study. The second significant aspect of this research is that it demonstrates the power of statistical thinking, and also the ability of statistics to contribute solutions to modern scientific problems.

(b) **Summarize** the data, i.e., % of pixels for the different classes. **Plot well-labeled beautiful maps** using x, y coordinates the expert labels with color of the region based on the expert labels.

We can see a summary of the data below:

```
##       x                y              label            NDAI
## Min.   :  2.0   Min.   : 65.0   Min.   :-1.0000   Min.   :-1.8420
## 1st Qu.: 98.0   1st Qu.:143.0   1st Qu.:-1.0000   1st Qu.:-0.4286
## Median :193.0   Median :218.0   Median : 0.0000   Median : 1.3476
```

```
##   Mean    :193.1   Mean    :218.1   Mean    :-0.1334   Mean    : 1.0847
##   3rd Qu.:289.0   3rd Qu.:294.0   3rd Qu.: 0.0000   3rd Qu.: 2.3142
##   Max.    :383.0   Max.    :369.0   Max.    : 1.0000   Max.    : 4.5639
##        SD                CORR               DF                CF
##   Min.    :  0.1987   Min.    :-0.3872   Min.    : 45.28   Min.    : 31.19
##   1st Qu.:  1.6376   1st Qu.: 0.1253   1st Qu.:244.56   1st Qu.:219.27
##   Median :  4.3095   Median : 0.1603   Median :281.91   Median :259.31
##   Mean    :  8.0633   Mean    : 0.1860   Mean    :271.36   Mean    :246.37
##   3rd Qu.: 10.2264   3rd Qu.: 0.2231   3rd Qu.:300.39   3rd Qu.:279.59
##   Max.    :117.5810   Max.    : 0.8144   Max.    :410.53   Max.    :360.68
##        BF                AF                AN
##   Min.    : 24.49   Min.    : 21.07   Min.    : 20.57
##   1st Qu.:200.79   1st Qu.:185.16   1st Qu.:174.88
##   Median :236.17   Median :211.54   Median :197.58
##   Mean    :224.20   Mean    :201.71   Mean    :188.29
##   3rd Qu.:258.62   3rd Qu.:235.15   3rd Qu.:216.80
##   Max.    :335.08   Max.    :318.70   Max.    :306.93
```

There are 11 features for each data point. They are x, y, label, NDAI, SD, CORR, DF, CF, BF, AF and AN. x and y are just the coordinate of the pixel in its corresponding image; label is the true label provided by expert; NDAI, SD and CORR are features that had been identified through extensive exploratory data analysis; DF, CF, BF, AF and AN represent different radiance angles. The radiance angles also use units that are much greater in magnitdue than the first three features.

For image1, there are 0.44 of pixels labeled as -1 (not cloud), 0.38 as 0 (unlabeled) and 0.18 as 1 (cloud). For image2, there are 0.37 of pixels labeled as -1 (not cloud), 0.29 as 0 (unlabeled) and 0.34 as 1 (cloud). Finally for image3, there are 0.29 of pixels labeled as -1 (not cloud), 0.52 as 0 (unlabeled) and 0.18 as 1 (cloud).

Do you observe some trend/pattern? Is an i.i.d. assumption for the samples justified for this dataset?

Region Plot Based on Expert Label
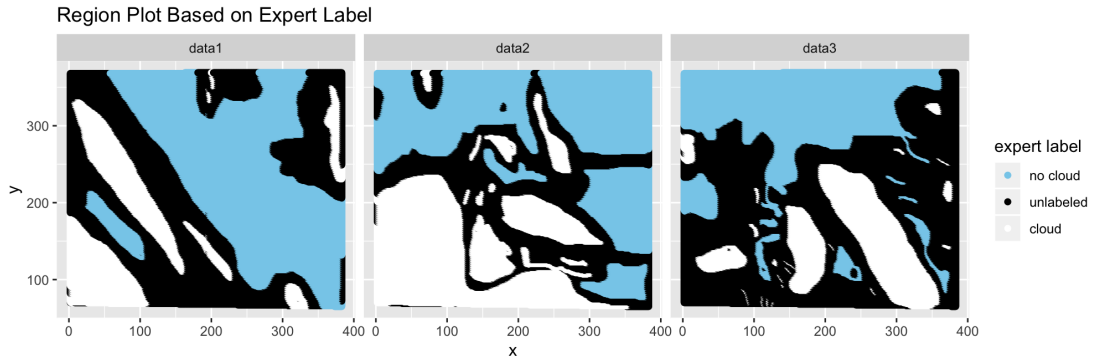
Looking at the plot based on the expert labels, we can definitely see that there are clusters for each of the classification labels, meaning that data are not iid. For example, if you know the label of one data point, the labels of the nearby data points are more likely to have the same label. In other words, there exists spatial dependence between

2

data points.

(c) **Perform a visual and quantitative EDA** of the dataset, e.g., summarizing (i) pairwise relationship between the features themselves and (ii) the relationship between the expert labels with the individual features.
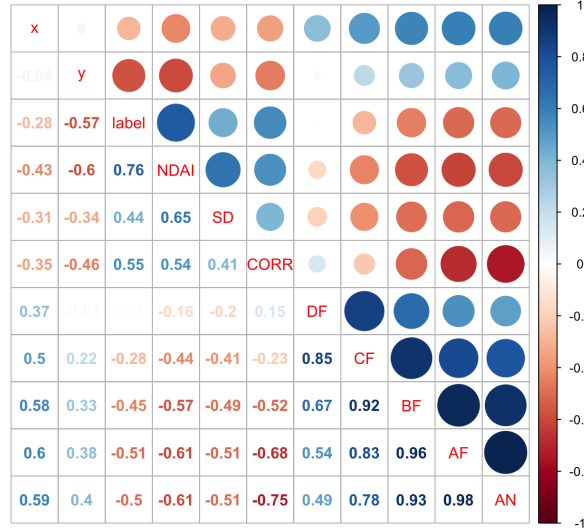
Many of the radiance readings(DF, CF, BF, AF, AN) have high correlation, which is reasonable as they are simply different angles of the same picture. There is a high linear correlation between NDAI/CORR and label. This suggests that a linear method of classification might be useful in predicting the labels. The y coordinate has a relatively high correlation with the label, although given its definition, this may be merely a coincidence.

Do you notice differences between the two classes (cloud, no cloud) based on the radiance or other features (CORR, NDAI, SD)?

NDAI of no cloud   **SD of no cloud**   **CORR of no cloud**
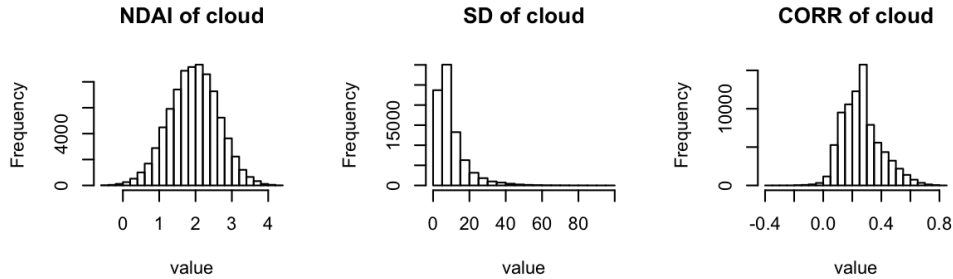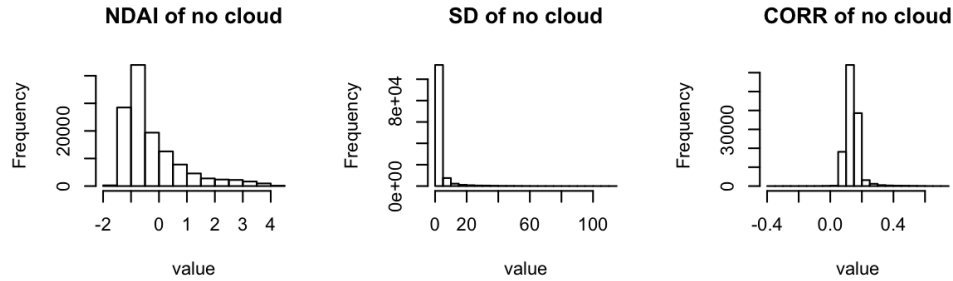
Based on the distribution of NDAI given different labels, we found that the NDAI of cloudy pixels tend to be higher with mean = 1.95, while NDAI of non-cloudy pixels have mean = -0.26.

Similarly for SD, we found that the SD of cloudy pixels also tend to be higher with mean = 9.84, while SD of non-cloudy pixels have mean = 2.98

For CORR, there is no obvious difference between different labels; their means are not significantly different; the two means are 0.26 and 0.14 respectively.

2   Preparation (40 pts)

Now that we have done EDA with the data, we now prepare to train our model.

(a) (Data Split) **Split the entire data** (image1.txt, image2.txt, image3.txt) into three sets: training, validation and test. Think carefully about how to split the data. **Suggest at least two non-trivial different ways** of splitting the data which takes into account that the data is not i.i.d.

As we concluded in 1b: there exists spatial dependence between data points. However, we want the data to be independent. If spatial dependence exists between data points in the training set and test set, this might result in a model with a high accuracy. Such models might perform well on the data that we have because they take advantages of the spatial dependence but they might not be a good model for future data. Future data will come in as a chunk of pixels and they will be likely to have different level of spatial dependence between pixels. If our model predicts the future data based on the spatial dependence of the training data, the prediction will probably be inaccurate. To tackle this issue, we have come up with two different methods.

Method 1:
Since future data will be likely be formatted as a chunk of pixels, we decided to divide each of the pictures into 10x10 blocks. This would give us approximately 3596 blocks of data. Then, we randomly chose 80% of the blocks as our training set; 20% as our test set; 20% of the training set as our validation set. By dividing the data into blocks, we hope that this would eliminate the spatial dependence between blocks, as pixels in different blocks will not be dependent on each other.

Method 2:

4

Since future data might come in as another picture and we happen to have 3 pictures, we decided to assign the three pictures as our training set, validation set and test set respectively. Since the three pictures are completely different images, they are spatially independent of each other. This would solve the problem of spatial dependence between training set, validation set and test set.

(b) (Baseline) **Report the accuracy of a trivial classifier** which sets all labels to -1 (cloud-free) on the validation set and on the test set. In what scenarios will such a classifier have high average accuracy? *Hint: Such a step provides a baseline to ensure that the classification problems at hand is not trivial.*

Such a trivial classifier gives accuracy 71% and 61% for the validation set and test set respectively. This classifier would have a high average accuracy if the new data given were mostly non-cloudy area.

(c) (First order importance) Assuming the expert labels as the truth, and without using fancy classification methods, suggest three of the "best" features, **using quantitative and visual justification**. Define your "best" feature criteria clearly. Only the relevant plots are necessary. Be sure to give this careful consideration, as it relates to subsequent problems.

```
## [1] "LASSO REGRESSION COEFFICIENTS (lambda = 0.1)"
## 9 x 1 sparse Matrix of class "dgCMatrix"
##                     s0
## (Intercept) -0.6908522
## NDAI         0.3995550
## SD               .
## CORR         1.2251421
## DF               .
## CF               .
## BF               .
## AF               .
## AN               .
## [1] "OLS REGRESSION COEFFICIENT: 0.805471348428594"
```

Looking at the lasso regression, we can see that the important features are NDAI and CORR. This is somewhat reliable since we can see that the regression coefficient of the OLS estimate is at 0.8, suggesting a somewhat strong linear correlation between the featues and the label. This inference is also supported by the high correlations between NDAI/label and CORR/label (shown above). We also know from the above correlations that the one of the radiance readings should be sufficient to represent all of the readings since they are all highly correlated. Looking at the correlations, we see that BF has the highest sum of correlations between the radiance readings, so we choose the features NDAI, CORR, and BF to represent our data.

(d) Write a generic cross validation (CV) function **CVgeneric** in R that takes a generic classifier, training features, training labels, number of folds K and a loss function (at

least classification accuracy should be there) as inputs and outputs the K-fold CV loss on the training set. Please remember to put it in your github folder in Section 5.

3 Modeling (40 pts)

We now try to fit different classification models and assess the fitted models using different criterion. For the next three parts, we expect you to try *logistic regression and at least three other methods.*

(a) **Try several classification methods and assess their fit using cross-validation (CV). Provide a commentary on the assumptions for the methods you tried and if they are satisfied in this case.** Since CV does not have a validation set, you can merge your training and validation set to fit your CV model. **Report** the accuracies across folds (and not just the average across folds) and the test accuracy. CV-results for both the ways of creating folds (as answered in part 2(a)) should be reported. Provide a brief commentary on the results. Make sure you honestly mention all the classification methods you have tried.

We ran tests on 5 models (QDA, LDA, Logistic Regression, Decision Tree, Kernel SVM - RBF) using 4 fold CV for method 1 and 2 fold CV for method 2. For method 2, we are bounded by 2 folds since our training set is only 2 images.

```
## [1] "METHOD 1:"
##       model      1      2      3      4 Average
## 1       QDA 0.8820 0.8856 0.8971 0.8920  0.8892
## 2       LDA 0.8865 0.8975 0.9073 0.8827  0.8935
## 3  logistic 0.8841 0.8878 0.9034 0.8768  0.8880
## 4     dtree 0.8969 0.9098 0.9131 0.8965  0.9041
## 5 kernelSVM 0.9481 0.9487 0.9505 0.9474  0.9487
```

```
## [1] "METHOD 2:"
##       model      1      2 Average
## 1       QDA 0.9544 0.8386  0.8965
## 2       LDA 0.8210 0.7801  0.8005
## 3  logistic 0.7188 0.7686  0.7437
## 4     dtree 0.9002 0.7786  0.8394
## 5 kernelSVM 0.8626 0.8118  0.8372
```

```
## [1] "TEST ACCURACIES:"
##       model Test.Accuracy
## 1       QDA     0.8919758
## 2       LDA     0.8981869
## 3  logistic     0.8932897
## 4     dtree     0.8979957
## 5 kernelSVM     0.9542773
```

QDA: Performed with about 90% accuracy.
Assumptions: The features are all normally distributed.
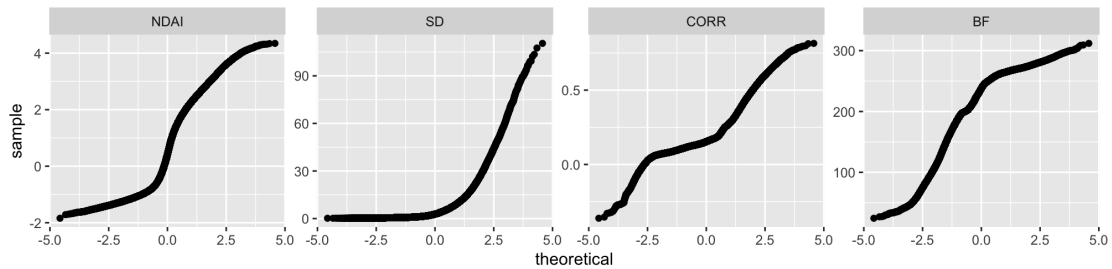
Looking at the qqplots of the features, we can see that all the features except SD have some resemblance to a normal distribution, meaning that this assumption is satisfied. The number of features must also be less than the number of data points, which is satisfied.

LDA: Performed with about 90% accuracy, puts emphasis on NDAI and CORR.
Assumptions: Same as above (normal distribution of features - satisfied). There is another assumption that the covariance matricies of each class are equivalent. Here we see the norm of the difference between the covariance matricies:

```
## [1] 4864.022
```

It can be seen that there is a significant difference between the matricies, meaning that QDA might be a better approach over LDA.

Decision Tree: We can see here that the tree formed is quite simple. The main feature used is NDAI, and the AN feature seems to have such an insignificant impact that it might be included in the model due to overfitting.
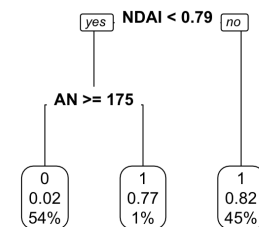
Assumptions: Since decision trees are nonparametric, there are no assumptions about the underlying distribution.

Logistic Regression: AN was arbitrarily chosen as the radiance reading to use in the model.
Assumptions: One of the assumptions is that there should be a linear relationship between the logit of the outcome and the features. We observe below that this is the case.

```
## [1] "CORRELATION OF LOGIT OF OUTCOME/FEATURES: 1"
```

Another assumption is that features should not be highly correlated with each other, which is satisfied since only AN was used from the radiance readings. We also satisfy the assumption that the output is a binary classification label.

Kernel SVM: This model performed the best out of the models tested here. The kernel used is the RBF kernel, with arbitrary parameters $C = 1, \sigma = 1$. The value of C is set by default and the value of sigma was chosen using the sigest function (part of the kernlab package). Because of the computational complexity of this model, it was infeasible to tune the hyperparameters during the scope of this project.
Assumptions: No assumptions, as the kernel svm does not make any assumptions about the input data.

We can see from the accuracies that using method 1, each of the models provides a similar accuracy of around 90%. Method 2 has a little lower accuracies, likely due to the fact that there is less data within each of the folds as opposed to the first method. We also see that NDAI, CORR, and AN seem to be features that are important. The AN feature might be seen as important due to possible overfitting of the data.

(b) **Use ROC curves to compare the different methods.** Choose a cutoff value and highlight it on the ROC curve. Explain your choice of the cutoff value.
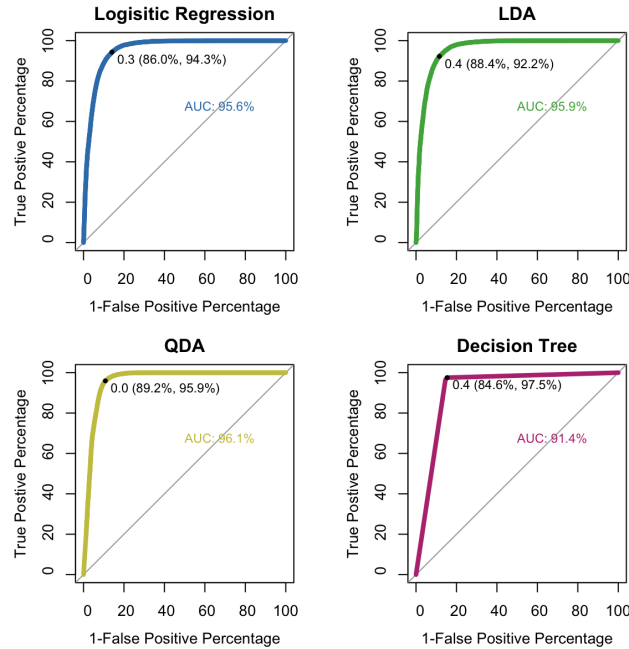
All the of ROC curves of the models have a very close AUC. The threshold labeled in each graph is the point where the graph has the highest sum of the sensitivity (true positive rate) and specificity (1-false positive rate). For QDA and LDA, the boundary

is defined as $Q_A(x) - Q_B(x) > threshold$ (class A).

QDA has the highest AUC among the 4 models, making it appear to be the best model. It also matches that the sum of its sensitivity and specificity is the highest among the 4. Looking at the test accracies for method 2, we can also see that QDA has the highest accuracy, which justifies the ROC observations.

There is no ROC curve for the kernel SVM method because this method doesn't output any probabilities, just class values.

(c) (Bonus) Assess the fit using other relevant metrics.

We look at the log loss for each of the models (KSVM is not included since the model does not output probabilities.)

```
##       model          1          2          3          4    Average
## 1       QDA 0.20326811 0.18424889 0.16688352 0.18568073 0.18502031
## 2       LDA 0.07839672 0.07400595 0.06171107 0.08181213 0.07398147
## 3 logistic 0.07216740 0.07122619 0.06274870 0.07574471 0.07047175
## 4    dtree 0.06964748 0.06687156 0.06497634 0.07217004 0.06841635
```

We can see from the log loss table above that logisitc regression performs has a significantly lower loss as opposed to QDA and Decision trees. The log loss penalizes more for confident misclassificatoins. This means that the logistic regression is not as confident in misclassifications as the other models. This is, however, a trade off with the low CV accuracy obtained using splitting method 2.

4 Diagnostics (50 pts)

(a) Do an in-depth analysis of a good classification model of your choice by showing some diagnostic plots or information related to convergence or parameter estimation.

Based on the ROC plots and accuracies, most of the models had similar performance, with QDA doing marginally better (KSVM is not considered because the only metric we have to compare it is the test accuracy). QDA also has a high accuracy rate with both splitting methods. Therefore, we choose to diagnose the QDA model.

Parameter convergence of QDA
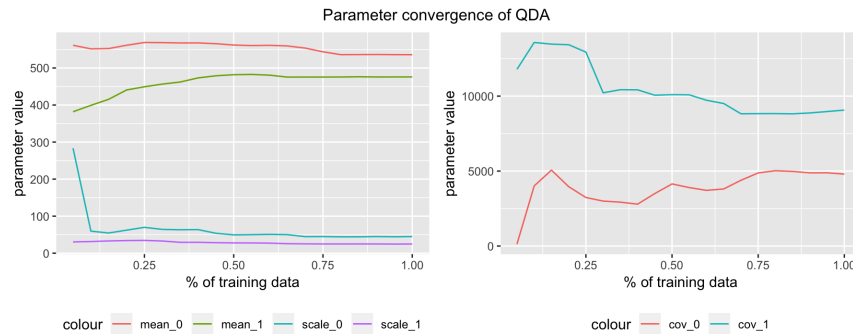
The parameters of QDA are the mean, the covariance and the scaling matrix of each class. Here we have graphed the frobenius norm of each of these parameters. Looking

at Parameter Convergence Graph, each of the parameter converges to the optimal value at about 80% of the training data. It means that it only takes about 80% of the training data to stabilize our classifier.

We also see that the covariance of each of the classes is quite different, which means that LDA would not be a feasible option.


Test Accuracy of QDA
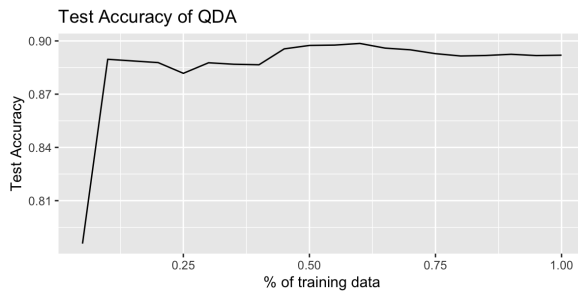
Looking at this plot, we can see that the model might actually be overfitting once it passes 60% of the data.

(b) For your best classification model(s), do you notice any patterns in the misclassification errors? Again, use quantitative and visual methods of analysis. Do you notice problems in particular regions, or in specific ranges of feature values?

Accuracy graph for KSVM (x,y)
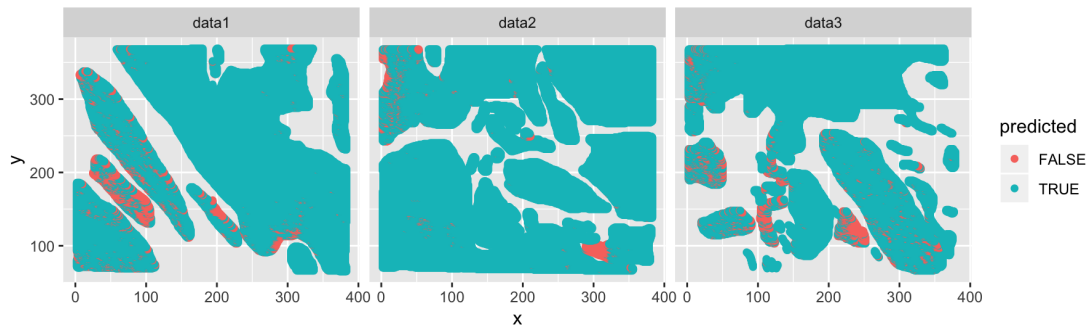
Accuracy graph for KSVM (NDAI,CORR)

Accuracy graph for QDA (x,y)


Accuracy graph for QDA (NDAI,CORR)
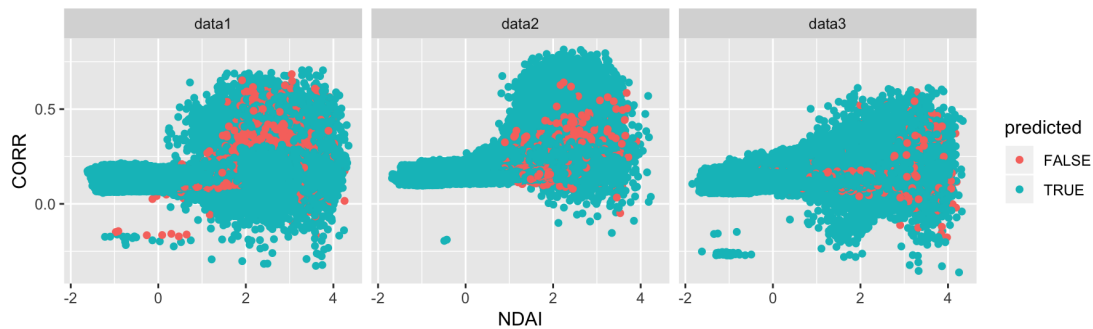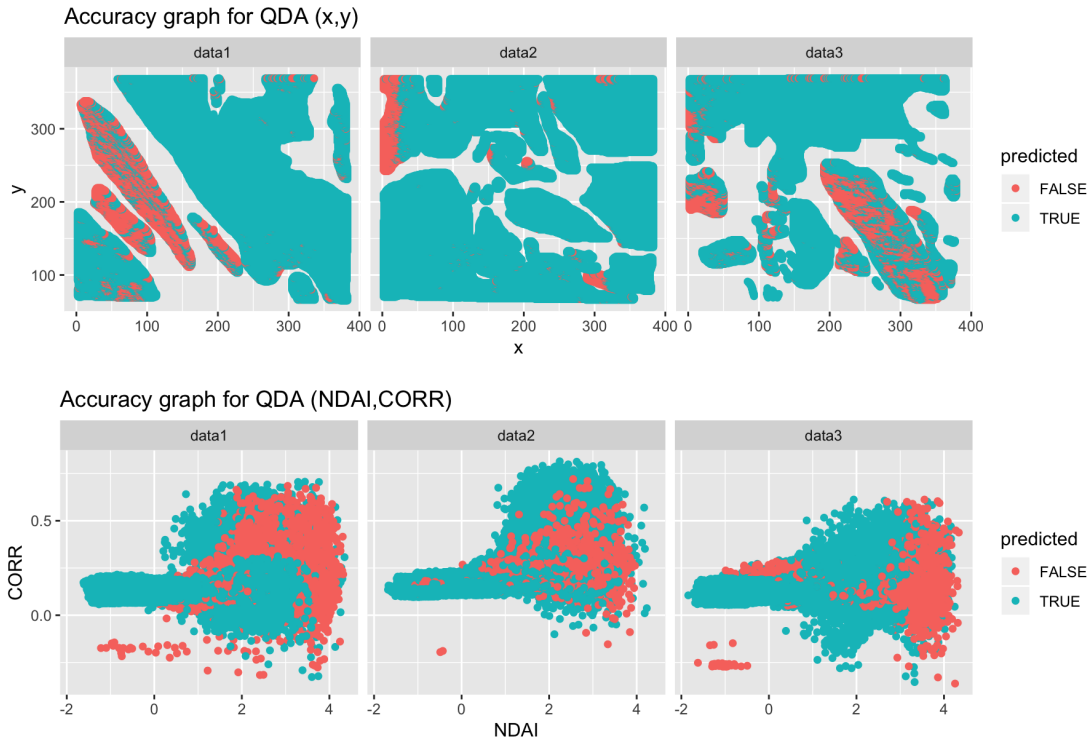
From the accuracy graphs based on location, we can see that there are definitely clusters in both KSVM and QDA regression that are entirely misclassified (moreso in QDA). Looking at the specific clusters, it appears that some of the misclassifications look to be more underfitting in certain areas (left side of data2 for QDA) and overfitting in others. For QDA, it seems that lower values of CORR also have more misclassifications. This again might be do to some over/underfitting of the model.

From the accuracy graphs based on the NDAI/CORR, we can see that for the KSVM model, most of the misclassifications are are values of NDAI over 1. For logistic regression, we can also see misclassifications in higher NDAI values. The misclassifications for CORR seem to be relatively well spread our over the range of CORR.

(c) Based on parts 4(a) and 4(b), can you think of a better classifier? How well do you think your model will work on future data without expert labels?

Since overfitting is related to having a high variance, we can use a model that uses ensemble methods to reduce the variance. An example of such a model is a random forest. Since a random forest is an ensemble method of a decision tree, it will have the same expectation but a much lower variance (depending on the number of trees used). The results are shown below.

```
## [1] "TEST ACCURACY: 0.951960058288144"
```

11

Accuracy graph for Random Forest (x,y)
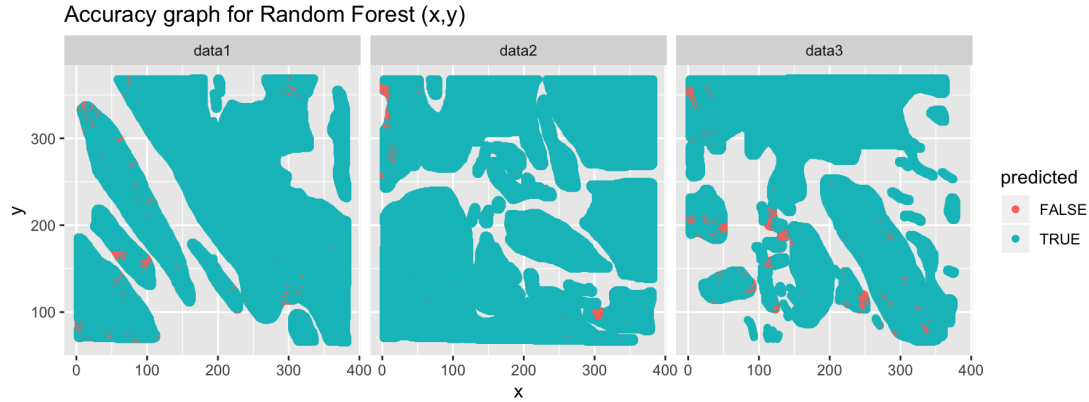
The results suffer less from overfitting than the models specified in parts (a) and (b). This model should work well better with future data without expert labels since it will be less likely to overfit the data and more likely to balance the bias and variance. We can also see that both the test accuracies are relatively high as compared to the other methods.

(d) Do your results in parts 4(a) and 4(b) change as you modify the way of splitting the data? Yes, the answers change based on the splitting method. the answers above were generated using splitting method 1, since it provides access to more training data points. However, even though the data is split up by blocks, there will inherently still be some dependence between the training and test/validation sets. In this regard, method 2 will model a real world situation better, since there is no dependence in pixels in separate images. Since method 2 gives us less training points, the observed testing points will likely have a lower accuracy rate, as we saw in 3(a). Empirically, it would be better to use method 2 with multiple images, where each image is a separate fold.

(e) Write a paragraph for your conclusion. We have tried 5 models for this project: QDA, LDA, Logistic Regression, Decision Tree and Kernel SVM. All of the 5 models did quite well on the CV accuracy using splitting method 1, with Kernel SVM as the highest; QDA stands out and reported the highest CV accuracy when splitting method 2 was used. Since splitting method 2 model a more realistic world situation as there is completely no dependence in pixels in separate images, we believe this might indicate that QDA is a good model on future data. Moreover, the ROC of QDA reports the best AUC and suggests a cutoff of 0; the assumptions of QDA is also met as we discussed above. The model converges at 80% of the training data. We believe that QDA is the best model of the five. Since QDA might have the problem of overfitting, we also tried to use a more advanced model: Random Forest. We hope to use this model to lower to variance of the prediction. It turned out this model did exceptionally well, reporting a 0.952 test accuracy. It correctly classified many of the mistakes QDA or other models made. We believe that Random Forest will perform outstandingly on the future data without expert labels since it will be less likely to overfit the data and more likely to balance the bias and variance.

12

5 Acknowledgement

Both members had equal contribution to the project (met up to work on the project).
Resources:

1. https://www.r-bloggers.com/how-to-implement-random-forests-in-r/

2. https://berkeleyml.github.io/static/notes/n25.pdf

3. https://berkeleyml.github.io/static/notes/n18.pdf

4. https://statquest.org/2018/12/17/roc-and-auc-in-r/

5. Yuansi Chen, Raaz Dwivedi, Bin Yu provided help during Office Hours for this project

Procedure: We followed the parts in order:

1. Took 1 week for data cleaning, processing, and EDA

2. Spent 1 weeks testing different models and went to OH for opinions

3. Spent the remainder of the time diagnosing the models and performing analysis

4. Compiled everything into RNW file at the end using latex

6 Github

the project is hosted here: https://github.com/shreysamdani/Cloud-Data-Classification