STA521 Project2: Cloud Image

Shiqi Liu (sl801@duke.edu), Tianyu Wu (t.wu@duke.edu)

1 Data Collection and Exploration

(a)

The key purpose of the paper *Daytime Arctic Cloud Detection Based on Multi-Angle Satellite Data With Case Studies* is to build an algorithm in order to ascertain whether a pixel of satellite images of Arctic is covered by cloud or not. Since the surfaces of cloud in Arctic are similar to those of the Arctic surface, it is challenging to tell the difference between cloud surface and Arctic land surface. The satellite images or the data used by the authors are shot by NASA Terra Satellite. The satellite comprises nine cameras with nine different angles in four spectral bands.

The data are collected from 10 MISR (Multiangle Imaging Spectro Radiometer) orbits of path 26 over the Arctic, northern Greenland, and Baffin Bay. This path is selected due to its richness of surface features. Six data units from each orbit are included in this study, and three of the total 60 units are excluded since the sea ice melts in the summer and affect the MISR operational algorithm. To evaluate the performance of the study, one of the authors hand-labels the image pixels as either clear or cloudy. Around 71.5% (5086002) of the pixels are labeled in total; the others are left unlabeled due to ambiguity.

The authors develop a classifying algorithm using enhances linear correlation matching (ELCM) and quadratic discriminant analysis (QDA) with three features (the linear correlation of radiation measurements from different MISR view directions, the standard deviation of MISR red radiation measurements, and the normalized difference angular index). As a result, the algorithm performs much better than other existing MISR operational algorithms. The study itself is also significant since the whole study only included three features and relatively simple classifying methods to separate clear and cloudy regions. Potentially, a more efficient and accurate classifying algorithm for Arctic cloud will eventually enable the scientific community to have more accurate global climate model simulations.

(b)

Percentages of pixels for different classes of the three images are shown in Figure 1. In Image1, the percentages of the three classes are relatively even. In Image2, the distribution on the three classes is imbalanced. 43.78% of pixels in Image2 are not cloud, while only 17.77% pixels are cloud. In image3, though it has a more imbalanced distribution compared with Image3, most of the imbalance comes from unlabeled class, which is irrelevant to the training process.

image	not cloud	unlabeled	cloud	Total
Image1	37.25%	28.64%	34.11%	100%
Image2	43.78%	38.46%	17.77%	100%
Image3	29.29%	52.27%	18.44%	100%

Figure 1: Percentage of Percentage of not cloud, unlabeled and cloud for each image.

Well-labeled maps are shown in Figure 2. Based on the labeled map, the classification of data points shows a sticky pattern in all three images. Meanwhile, cloudy regions and clear regions are separated by unlabeled

data points. Therefore, i.i.d. assumption can't be justified for this data set.

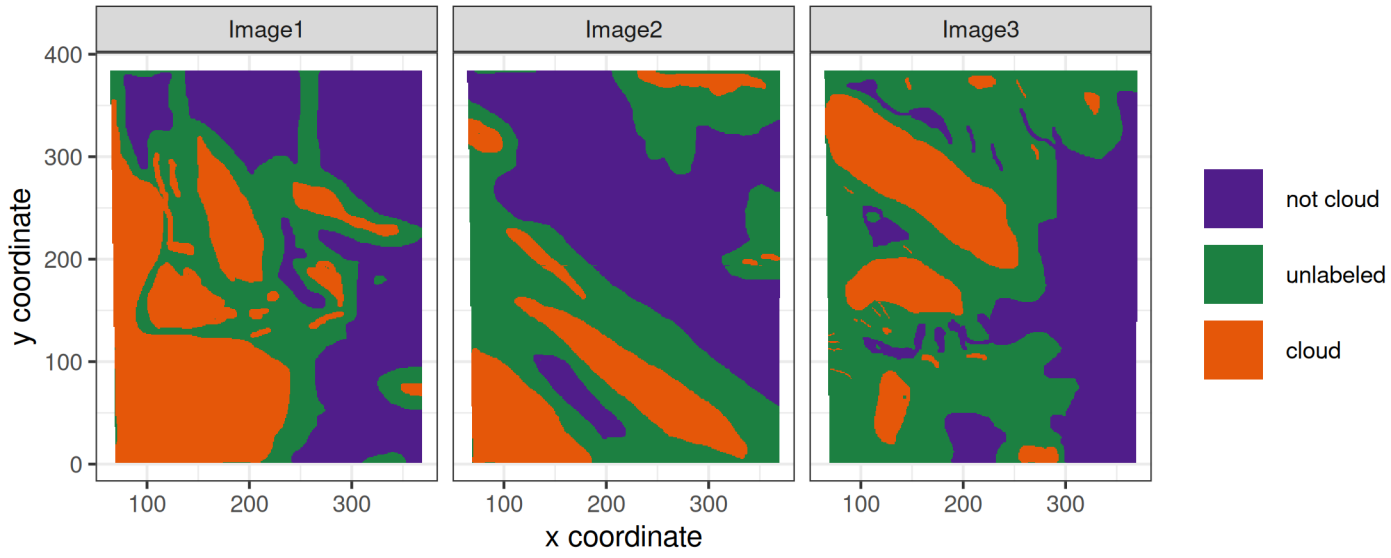

Figure 2: Image1-3 with expert labelled.

(c)

To perform a visual and quantitative EDA on the data set, we first combine the three data sets and remove all unlabeled data points. Then, we generate a plot (in Figure 3) with pairwise relationships between all eight features and a group of box plots (in Figure 4) to show the relationship between each feature and the expert labels.

In the upper part of the Figure 3, three different correlations are presented for each pair of the eight features. The first of the three correlation is computed with the whole data set. The second one is computed with data points labelled as not cloud and the third one is computed with data points labelled as cloud. We observe that the correlations between pairs of the five radiation values are relatively higher than the correlations between pairs of NDAI, SD and CORR. We also find that the correlations between pairs of NDAI, SD and CORR are smaller, when the data set are divided into two groups. The reason for this phenomenon may be that the ranges for the two groups on NDAI, SD and CORR are different.

From the first plot in Figure 4 and the first diagonal plot in Figure3, we find that the distribution of NDAI for the two groups are quite different with each others, so NDAI is supposed to be a good feature to separate data set. For CORR and SD, the differences are relatively smaller. For the five radiation values, DF, CF,BF,AF, and AN, their distributions are similar with each others for both classes. Also, with unlabeled data points removed, the correlations between Label and each of the eight features increase by varying degrees.

2 Preparation

(a)

For the three images in Figure 2, each of them may have a unique pattern. In case of over-fitting to one particular image, the combination of the three images is necessary before training. We also notice that there are some strong dependencies between one pixel and its neighbors. Therefore, we first choose to split the data set by block splitting, to avoid breaking its spatial structure. To be specific, we first divide the data set into numerous 8×8 pixel blocks and for pixels in the same 8×8 block, they are all assigned to one single fold randomly chosen from K folds, and the default value of K in our splitting function is 6, and unlabeled points are removed. Also, we do not desire that blocks at same location of different images are in the same

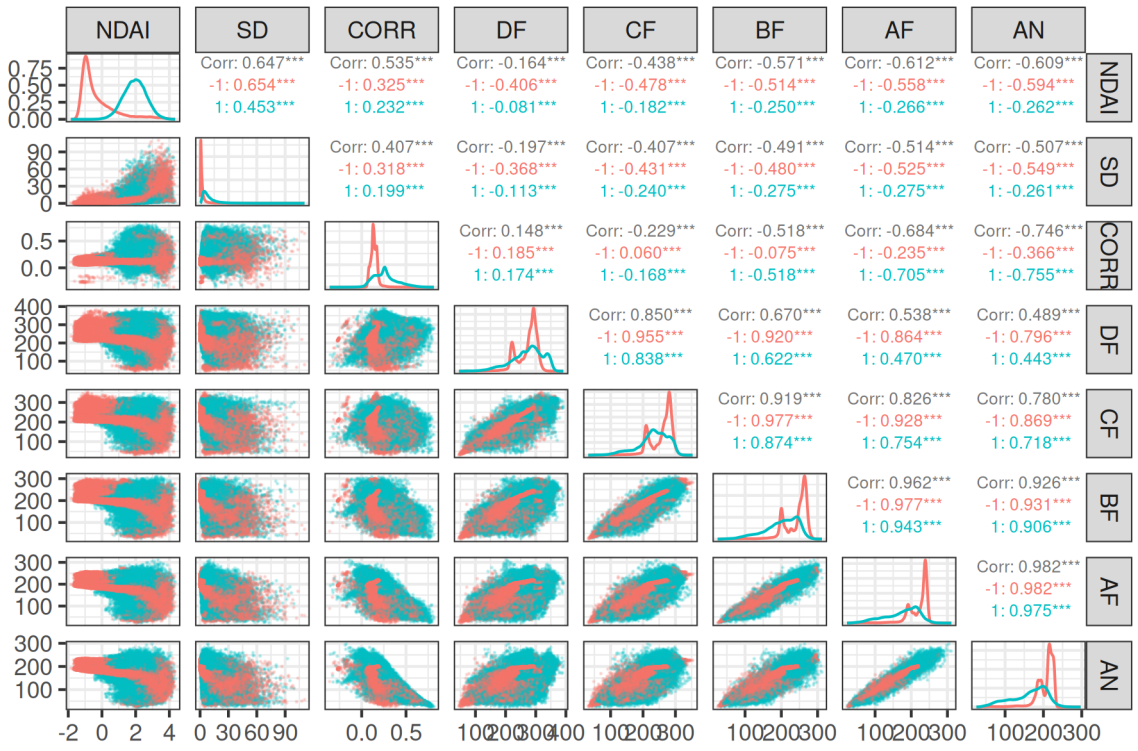

Figure 3: Pairwise relationships between the three features

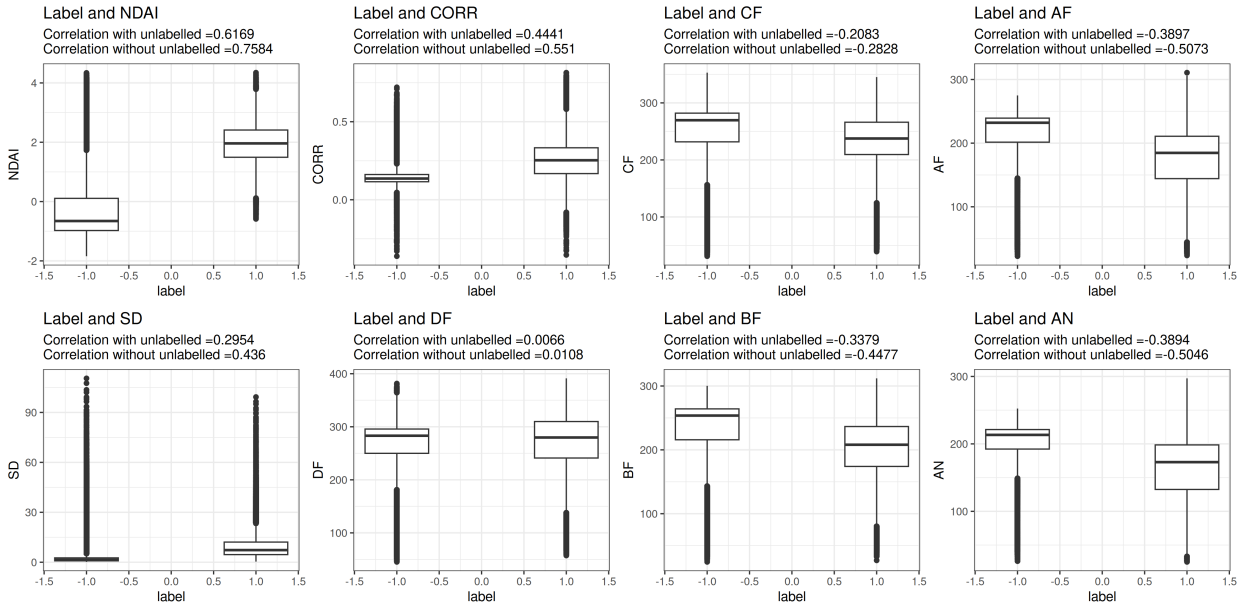

Figure 4: Boxplots of label verse each feature, with unlabelled data removed. Correlations of label and each feature with and without unlabeled data.

fold, so the three data sets are splitted separately before combination. Splitted images are shown in Figure 5. However, as a result of block splitting, ratios of the two labels are not even among different folds.

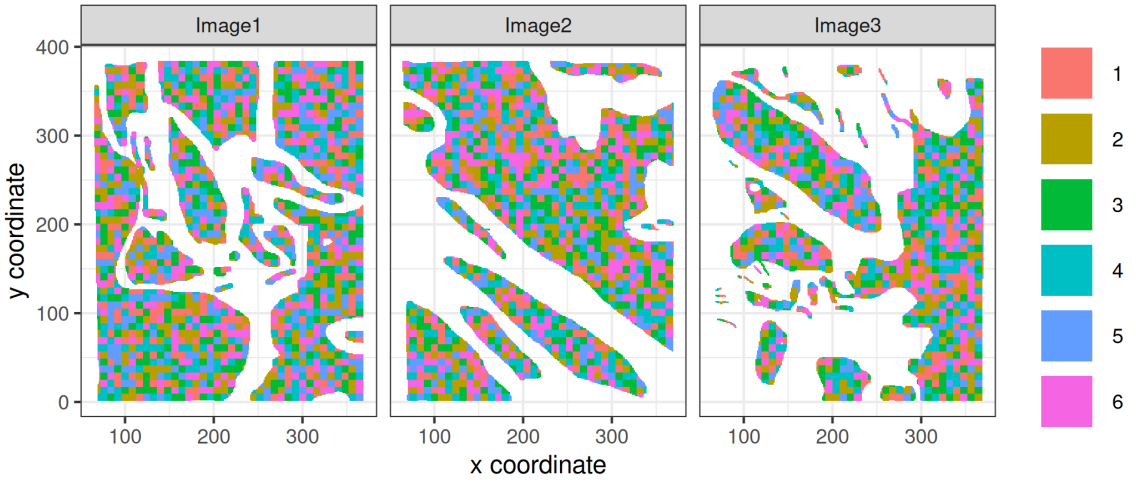

Figure 5: Splitting result of block splitting.

For the second splitting method, we expect that each fold has the same ratio of cloud and not cloud data points as the whole data set, which is 61.08 for not cloud and 38.92 for cloud. To achieve that goal, the second method first divides the data set into cloud and not cloud data sets. Then, each point in the two data sets is evenly and randomly assigned to one out of K folds, and the default value of K is 6. The percentages of cloud and not cloud for all six folds are shown in Figure 6. All folds have 61.08 for not cloud and 38.92 for cloud.

After all, for both splitting methods, all data points in fold K are used for testing data, all data points in fold $K-1$ are used for validation, and the rest are used for training.

fold	not cloud	cloud	Total
1	61.08%	38.92%	100%
2	61.08%	38.92%	100%
3	61.08%	38.92%	100%
4	61.08%	38.92%	100%
5	61.08%	38.92%	100%
6	61.08%	38.92%	100%

Figure 6: Percentages of cloud and not cloud for each fold generated by the second split method.

(b)

For test and validation data sets using either one of the two splitting methods, the percentage of not cloud is all around 60 (in Figure 7). Therefore, the accuracy of a trivial classifier, which sets all labels to -1, would also be around 60 for validation sets and test sets, regardless of splitting methods. The accuracy of a trivial classifier would be high, if the data set is highly imbalanced. For example, if 99 of the data set are labelled as not cloud, the accuracy of the trivial classifier would be 99. Concretely, the trivial classifier is not well-performed for our data set, so more fancier classifiers are necessary to achieve a higher accuracy.

data	split method	not cloud	cloud	Total
test	block split	60.61%	39.39%	100%
valid	block split	58.24%	41.76%	100%
test	even split	61.08%	38.92%	100%
valid	even split	61.08%	38.92%	100%

Figure 7: Percentage of not cloud and cloud for validation set and test set of two split method.

(c)

By performing the logistic regression forward selection on all features (in Figure 8), NDAI, CORR, and SD are three of the "best" features. With adding NDAI, CORR, and SD to the training model, AIC and deviance drops significantly. According to *Daytime Arctic Cloud Detection Based on Multi-Angle Satellite Data With Case Studies*, these three features are computed with all radiations of nine angels, so these three features include some information of the other features, DF, BF, AN, CF and AF. Also, these three features of each pixel are based on the radiation values of pixels around itself, so these three features also include some information of its neighbors. Choosing these three features, our model would be more likely to classify closep data points into the same category. In the following questions, only these three features are used to train classification models.

Step	Df	Deviance	Resid. Df	Resid. Dev	AIC
	NA	NA	208060	278134.6	278136.6
+ NDAI	-1	147904.4698	208059	130230.2	130234.2
+ CORR	-1	9675.0264	208058	120555.1	120561.1
+ SD	-1	4225.1203	208057	116330.0	116338.0
+ DF	-1	2551.9709	208056	113778.0	113788.0
+ BF	-1	1703.2342	208055	112074.8	112086.8
+ AN	-1	853.0864	208054	111221.7	111235.7
+ CF	-1	153.4786	208053	111068.2	111084.2
+ AF	-1	172.2860	208052	110895.9	110913.9

Figure 8: Result of logistic regression forward selection on all features.

(d)

CVmaster function is in CVmaster.R.

3 Modeling

(a)

Selected classification models are Logistic Regression(LR), Linear Discriminant Analysis(LDA), Quadratic Discriminant Analysis(QDA), Naive Bayes(NB) and Classification and Regression Tree(CART). Logistic Regression is a classification model with mainly two assumptions. The first is that the response variable (class) follows a Bernoulli distribution. The second is that the log-odds of this Bernoulli variable is linear with respect to covariates. This linear assumption, as indicated by part4(a) in the next section, is reasonable but might not be accurate. Assumptions for LDA and QDA are the same: there is a data generating process behind the scenes. First, generate a class. Second, generate instances for the chosen class. Within each class,

instances are normal distributed with different means. The difference of LDA and QDA lies in the fact that the covariance for different classes are the same for LDA while the covariance are different across classes for QDA. There is also a data generation process behind NB. The assumption here is that coviarates are generated independently within each class. As can be seen from Figure 10, distributions of covariates are not normal within each class, nor are they independent in general. Some of the distributions are bimodal, which are far from Gaussian distribution while the distributions of NDAI and SD are quite close to Gaussian. The dependence of covariates can also be seen from GGPair plot as in Figure 3. It is interesting to observe that within each class the correlation between covaraites are actually smaller. The independence assumption of Naive Bayes is not perfectly accurate, but reasonable. For accuracy results, we can see that cross-validation mean accuracy is very close to test accuracy in general. This indicates that CV is a robust way of evaluating model performance. The top model is Naive Bayes according to test accuracy.

classifier	split method	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Average	Test
Logistic Regression	block split	0.9136859	0.8903206	0.8951867	0.8812245	0.8930896	0.8947015	0.8862240
Logistic Regression	even split	0.8925801	0.8938201	0.8939931	0.8910517	0.8925513	0.8927993	0.8920867
QDA	block split	0.9114822	0.9058705	0.8725846	0.8904351	0.8857954	0.8932336	0.9120353
QDA	even split	0.8938489	0.8972518	0.8971653	0.8971653	0.8988090	0.8968481	0.8964702
LDA	block split	0.8999821	0.9148861	0.8899852	0.8874817	0.8969356	0.8978541	0.8951726
LDA	even split	0.8958099	0.8961560	0.8973095	0.8966750	0.8989820	0.8969865	0.9000173
NB	block split	0.9074828	0.9144763	0.8837777	0.9244102	0.9186532	0.9097600	0.9138384
NB	even split	0.9104594	0.9115840	0.9149869	0.9091905	0.9108919	0.9114226	0.9104280
CART	block split	0.8854271	0.9094795	0.9070102	0.9053283	0.8984155	0.9011321	0.8933681
CART	even split	0.9020965	0.8987225	0.8985495	0.8989820	0.9003662	0.8997433	0.8980851

Figure 9: The accuracies across folds, the average accuracies across folds, and the test accurary for each classification model with two different splitting methods.

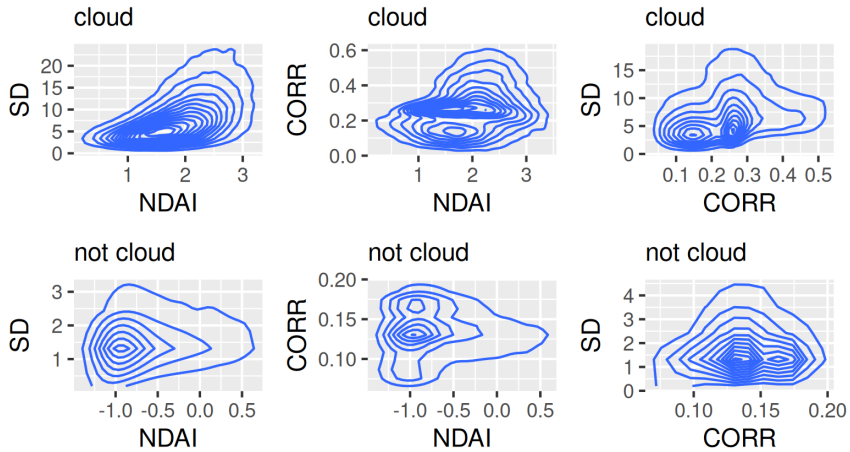

Figure 10: Distribution of covariates

(b)

Cutoff values are selected based on maximum Youden's J Statistic (Youden's Index), which is defined as True Positive Rate(TPR) - False Positive Rate(FPR). Geometrically it indicates the vertical distance right above the 45 degree line which represents ROC curve for a basic random model. As can be seen from the Figure 11, the selected cut-off points locate almost at the top-left corner of each ROC curve. This is preferred as we

want to maintain relatively high TPR and low FPR at the same time.

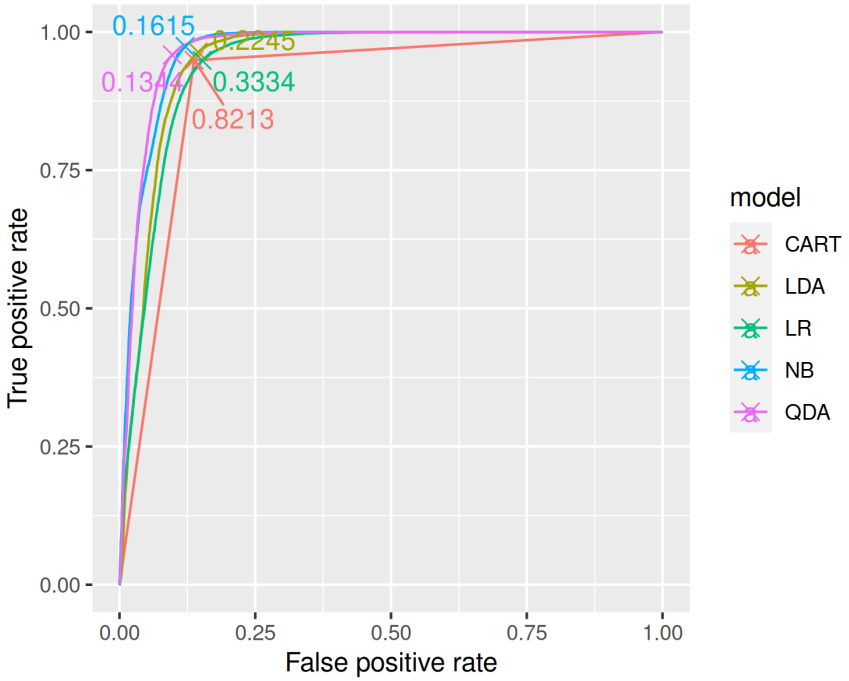

Figure 11: ROC curve with cut-off values

(c)

Other selected relevant metrics are precision, recall, f1 score and area under curve(AUC). Precision measures the ratio of correct prediction out of all positive predictions. In other words, precision measures quality of positive prediction. Recall measures the ratio of successfully predicted positive instances out of all positive instances. In other words, recall measures the ability to identify positive instances. F1 score is the harmonic mean of precision and recall, which can be seen as a comprehensive measurement regarding positive predictions combining both precision and recall. AUC measures area under ROC curve, which can be seen as a comprehensive measurement regarding the accuracy of positive predictions. Best model is again Naive Bayes, according to f1 score and AUC (Figure 12).

classifier	split method	precision	recall	f1	auc
Logistic Regression	block split	0.8269481	0.8850876	0.8550306	0.9427083
Logistic Regression	even split	0.8528944	0.8733699	0.8630107	0.9500998
QDA	block split	0.8936816	0.8698647	0.8816123	0.9674895
QDA	even split	0.8614273	0.8747036	0.8680147	0.9526778
LDA	block split	0.8469186	0.8835607	0.8648517	0.9491892
LDA	even split	0.8514262	0.9001926	0.8751306	0.9497112
NB	block split	0.8859248	0.9210526	0.9031473	0.9656913
NB	even split	0.8584736	0.9218287	0.8890239	0.9655821
CART	block split	0.7931010	0.9488631	0.8640181	0.9057084
CART	even split	0.8145365	0.9557647	0.8795173	0.9085481

Figure 12: Distribution of covariates

4 Diagnostics

(a)

From optimization perspective, the objective function of logistic regression loss is convex. According to Figure 13, parameters converged after 6 steps and achieves global minimum. From the effectiveness perspective, the model works effectively as there is a huge drop from null deviance to residual deviance, indicating tremendous gain from logistic regression model compared to a plain constant model. From validity perspective, all the coefficients are statistically significant. The p-values are too small that they become numerically 0 (Figure 14. To test the linear assumption, scatter plots of predicted log-odds and covariates (Figure 15) are plotted. Strict linearity should not be expected as the linearity is with respect to all covariates while each scatter plots only demonstrate the relationship between one log-odds and one variable. Therefore, a smooth estimation is made to demonstrate the trend. As can be seen, there is linear relationship with respect to NDAI and CORR, but not for SD. Training data does not align with linear assumption perfectly. Besides linearity assumption, collinearity is also crucial for linear models. We adopted variance inflation factors to test collinearity among covariates (Figure 16). Since as a rule of thumb VIF value that exceeds 5 or 10 indicates a problematic amount of collinearity, there is not collinearity issue in training data. In the end, we try to visualize decision boundary in two-dimension space using PCA, but the projected data indicates that there does not exists such boundary in lower dimensional space (Figure 17). The evidence is that predicted positive and negative labels mix together. We cannot determine the predicted label of a given lower-space point. This makes sense if we notice that the decision boundary in 3-dimensional space is 2-dimensional, the projection of which should also be 2-dimensional.

No.Iteration	Convergence	Null.deviance	Residual.deviance
6	1	235341.440363	97235.4429295864

Figure 13: Summary of logistic regression model

Variable	Estimate	z value	Pr(>|z|)
(Intercept)	-3.32046932	-161.99346	0
NDAI	1.90668626	185.31368	0
CORR	8.82033299	83.79446	0
SD	-0.07086621	-59.37456	0

Figure 14: Coefficients of logistic regression model

(b)

The best model is Naive Bayes. Hence,this part will be discussed based on result of Naive Bayes. There are two types of misclassification errors: false positive(FP) and false negative(FN). To visualize the pattern of misclassified data points, we plot misclassified data points against all 3 selected features and xy coordinates for all 3 images. In spatial visualization (Figure 19), there are several significant trends of misclassified data points. Initial guess was those misclassified might be close to unlabeled boundary. But it turns out that false negative and true positive are mixed quite well far away from boundary in image3. There is also a tendency for false positive in a small area enclosed by boundary, as in image2 and image3. Furthermore, misclassified points are located close to each other. This makes sense as weather condition and readings are similar in nearby regions. In feature visualization (Figure 18), it is noticed that false positive and true positive are

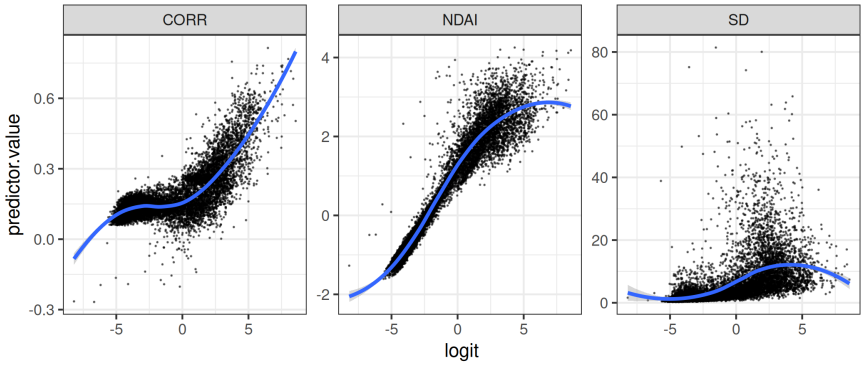

Figure 15: Linearity test for logistic regression model

NDAI	CORR	SD
1.51787583371424	1.07550050449298	1.56273852955056

Figure 16: Collinearity test for logistic regression model

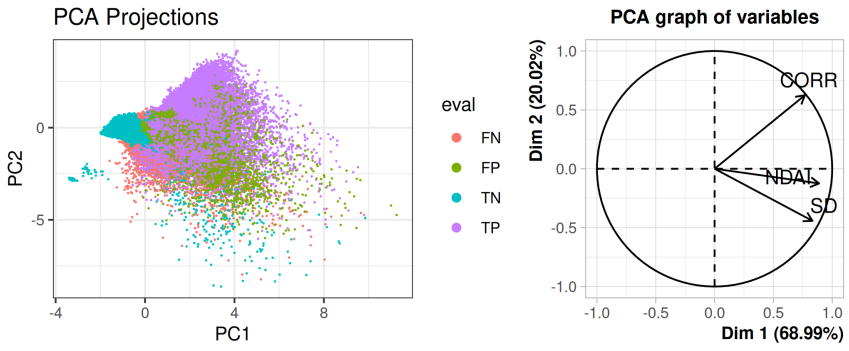

Figure 17: PCA for logistic regression model

cluttered together in the same feature range. We can check this in more details based on Figure 20. False positive and true positive share almost identical feature range for NDAI and SD. False negative and true negative share almost identical feature range for CORR and SD. This is exactly the reason models make wrong predictions. After all, similar features are supposed to match to the same classes. Moreover, in these projected 2-dimensional feature space, the decision boundary is quite clear.

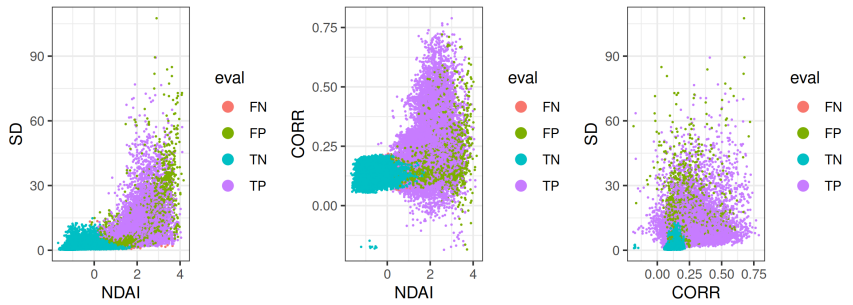

Figure 18: misclassified data points against 3 features

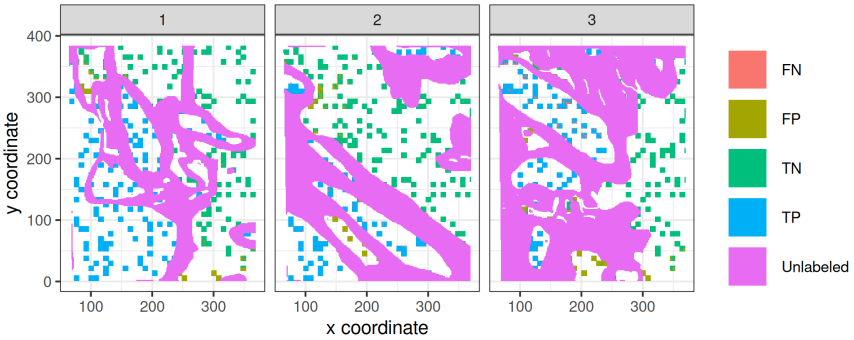

Figure 19: misclassified data points against xy coordiantes

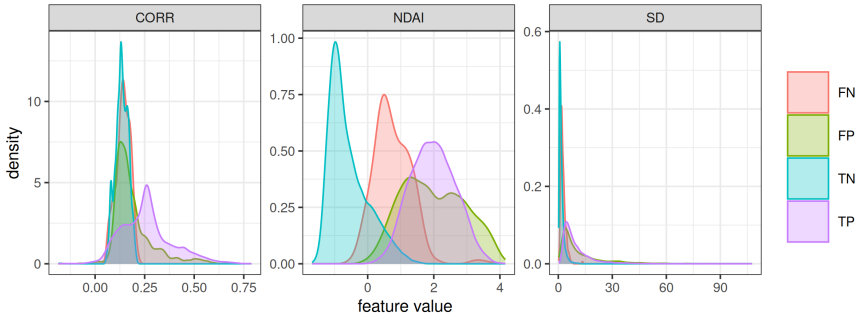

Figure 20: feature range of misclassified data points

(c)

The analysis in part4(b) indicates that most of the misclassfication errors are due to the fact that similar features correspond to different labels. Therefore, it is very difficult in general to make further improvement. However, one idea is to use KNN to make use of the fact that data points with similar labels are located next to each other. Advantage of KNN is that it has a much more complicated decision boundary. Simple decision

boundary cannot resolve those tricky cases as indicated by analysis in part4(b). The second idea is to use ensemble models to incorporate result from different base models. From Figure 21, we can see that there are some minor prediction difference between NB and QDA, such as the bottom-left corner of second plot. This is due to different data generation assumption made by two generative models. Also, taking into account distribution of labels might be helpful in a sense that it is very difficult to distinguish labels solely based on covariates (discriminative model). The hope is that by combining KNN together with 2 best generative models (QDA and NB), the overall result will improve. The final decision is made by majority vote. The third idea is to use random forest to create some randomness in the process of making use of features. Besides that, decision boundary of random forest might be much more complicated as well. The final idea is to do some feature engineering. A log transformation is applied to SD since it does not satisfy linear assumption. However, feature engineering does not make any difference in this scenario as similar features are transformed similar new features as well and no more power of differentiation is introduced. Unfortunately, all 4 ideas fail to deliver a better classification model, as shown in Figure 22. All of the models should be able to perform well on future data without expert labels and an overall 90% accuracy should be expected. This is justified by the test accuracy, which is essentially performance on future unseen data.

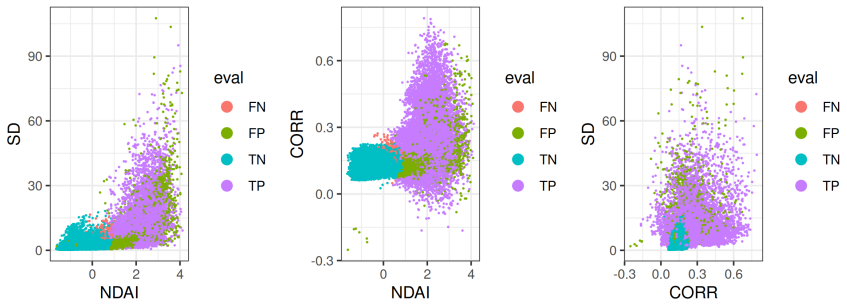

Figure 21: misclassified data points against 3 features, QDA

classifier	split method	test accuracy
Random Forest	block split	0.9018926
Random Forest	even split	0.9164552
Ensemble Model	block split	0.9022153
Ensemble Model	even split	0.9142346
LOGLR	block split	0.8887484
LOGLR	even split	0.8955185
KNN	block split	0.9088052
KNN	even split	0.9080082

Figure 22: Test accuracy of 4 new models

(d)

Results are similar. Number of iterations are the same. Estimated coefficients are similar. The diagnostic plot for linearity share the same smooth trend estimation. Scatter plots in PCA feature space are similar as well. Variance inflation factors are alike. Misclassfication error for Naive Bayes also exhibits similar feature distributions and spatial distributions. This is expected as the class distribution within training dataset and

test dataset turns out to be close to each other. The large scale of dataset also reduces the effect of different splitting method. Similar plots are not listed in the report to save space.

(e)

The conclusion is that all of the models built perform reasonably well. Naive Bayes is the best among them. Logistic regression model is analyzed in depth and the model turns out be good and robust. The difficulty of making prediction lies in the fact that similar feature range correspond to different labels, or, from another perspective, true positive and false positive/true negative and false negative share similar feature distribution. This data limitation makes the prediction inherently challenging regardless of the model selected. Several attempts are made to improve classification model, including KNN, ensemble model, random forest and feature engineering. The reason ensemble model does not improve overall prediction is that the prediction difference between models are very limited. The gain from combining almost homogeneous models is very limited. We also see improvement from CART to random forest, but the improvement is not significant enough to beat Naive Bayes.

5 Reproducibility

All the code is under repository https://github.com/TianyuTerry/CloudImage. Follow the steps described in README.md to reproduce results.

6 Acknowledgement

Shiqi Liu is in charge of part 1 and 2. Tianyu Wu is in charge of part 3 and 4. Both of them contributed to the other half of the content a lot. Resources are mainly materials in STA521 and Stack Overflow for R code.