Daytime Arctic Cloud Identification Based on Multi-Angle Satellite Data*

Xiaoyang Wu (吴旖杨)
16342182

Jul 2, 2020

1 Data Collection and Exploration

1.1 Summary about data paper

The paper *Daytime Arctic Cloud Detection Based on Multi-Angle Satellite Data With Case Study* written by Shi et al. aims to provide a systematic method of accurate cloud detection in the Arctic which would do great favor to modulate the sensitivity of the Arctic and further, promotes the study on the dependence of surface air temperatures on increasing atmospheric carbon dioxide levels in the Arctic. The paper successfully identify three physically useful features–the linear correlation of radiation measurements from different MISR view directions (CORR), the standard deviation of MISR nadir red radiation measurements (SD_An) and a normalized difference angular index (NADI) and proposed two efficient operational Arctic cloud detection algorithms,**enhanced linear correlation algorithm (ELCA)** and **ELCM-QDA algorithm**, using **Multiangle Imaging SpectroRadiometer (MISR) images** provided by NASA, which yield satisfying outcome with high accuracy and coverage.

In achieving this, the vast amount of data provided by MISR is of vital importance. The MISR sensors comprise nine cameras viewing the Earth scene at nine distinct angles in four spectral bands and cover 233 paths extending across the daylight side of the Earth from the Arctic down to Antarcitica, collecting data from all paths on a repeat cycle of 16 days. The data used in the paper were collected form 10 MISR orbits of path 26, whose surface features are considerably rich, including the permanent sea ice, snow-covered and snow-free coastal mountains, permanent glacial snow and ice, and sea ice that melted during study period. These obits spans 144 days from April 28 to September 19 in 2002, which is exactly the daylight season in the Arctic. Among the 180 blocks in path 26, six data units 11-13, 14-16,17-19,20-22,23-25 and 26-28 are included, and 275-m red radiation measurement in the 26 measurements are chosen to investigate.

By comparing the performance of the two new algorithms to the expert labels, the researchers find out that the three physical features contain sufficient information to separate clouds from ice- and snow-covered surfaces and the algorithms are quite efficient.

*https://github.com/Gofinge/Daytime-Arctic-Cloud-Detection-Practice
Code and paper of data can be found here.

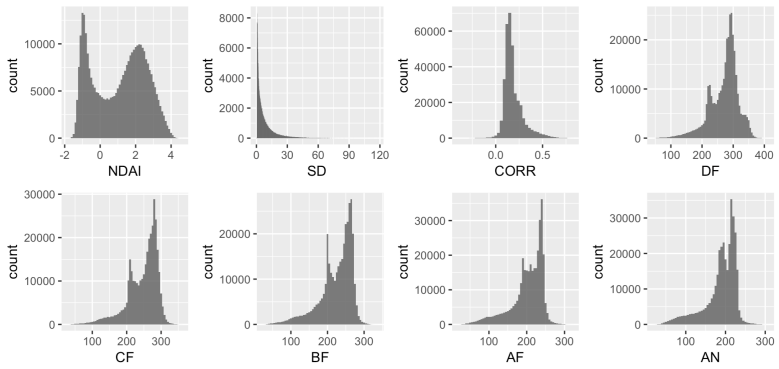


Figure 1.1: Histograms of 8 features

The potential impact of this study is significant. First, the statistic methods used and the new algorithms proposed in this study provide inspiration for statisticians to think of appropriate and efficient methods in analyzing the abundant Earth science data that support weather and climate studies and also study fields beyond. Second, in this study, statisticians are directly involved in data processing instead of joining after, which encourages statisticians to work directly together with other scientists and possible instrument team to achieve great success through close collaboration. Also, the result itself is of great influence, which provides strong statistic support to further study on dependences of surface air temperatures on increasing atmospheric carbon dioxide levels in the Arctic.

1.2 Summary of data

Combined three pictures together, the data in this project has **345556** observations and **11** features, including ten numerical variables and one class attribute variable. The ranges of these ten variables, y coordinate, x coordinate, NDAI, SD, CORR, DF, CF,BF, AF, and AN are shown in table 1. As for the class attribute variable **label**, we have 23.43% observations labeled as **cloud**, 36.78% observations as **not cloud**, and 39.79% unlabeled. Also, in order to have a clear look at the distribution of different features, we draw histograms in figure 1.1, which shows that the five radiance angle all have negative skew. The relationship between the expert labels with the individual features will be discussed in the next subsection.

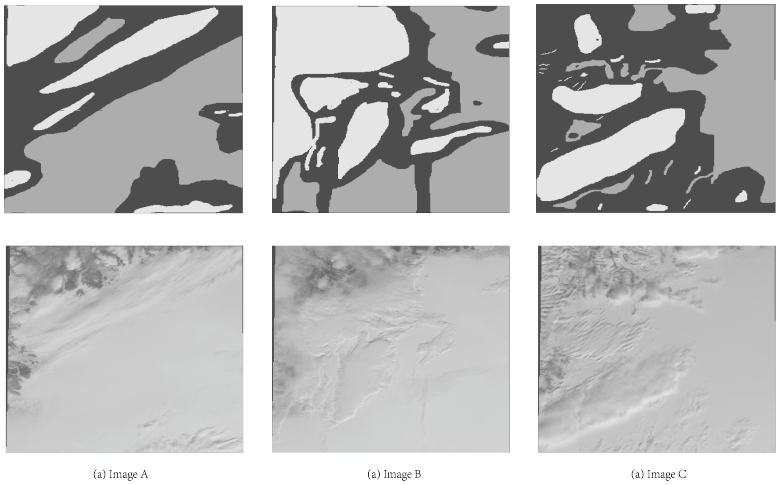


Figure 1.2: Pixel images of data

In the labeled maps generating by x,y coordinates, as shown in Figure 1.2, I find out that pixels labeled as cloud and not cloud are well separated, and unlabeled pixels always occur between these two groups, which is natural since the pixels in boundary are always harder to classified. In this dataset, **the i.i.d assumption should not hold** since there seems to have strong correlation between neighbored pixels instead of existing

independently.

1.3 EDA of dataset

In exploration of the dataset, I did some pairwise comparison between the features themselves by scatterplot and discussed the differences between two labeled classes in terms of the individual features by boxplots. Without loss of generality, I drawn 0.1% of the total data randomly in order to better present the trends.

From figure 1.3 (boxplot), it is clear that there are some differences between two classes based on individual features. Namely, variables **y,x,DF,CF,BF,AF,AN** all yield higher values in pixels labeled as **not cloud** than those labeled as **cloud**, while variables **NDAI,SD,CORR** have lower values in **not cloud** group.
From figure 1.4 (pairwise), we can see that, ranking from strongest to weakest, y positively correlated with AF, AN, BF, CF, DF, negatively CORR; x positively correlated with AN,AF,BF,CF, negatively NDAI, CORR; NADI positively correlated with CORR, negatively AF,BF,AN; SD negatively correlated with AF,BF,AN,CF; CORR negatively correlated with AN,AF,BF; DF positively correlated with CF,BF,AF,AN.

Beside, I also have a rough look at the importance of these features by PCA. As showed in figure 1.5 (PCA), first three components contain almost 90% of the variability of the data. I would discuss it in details in section 2.3.

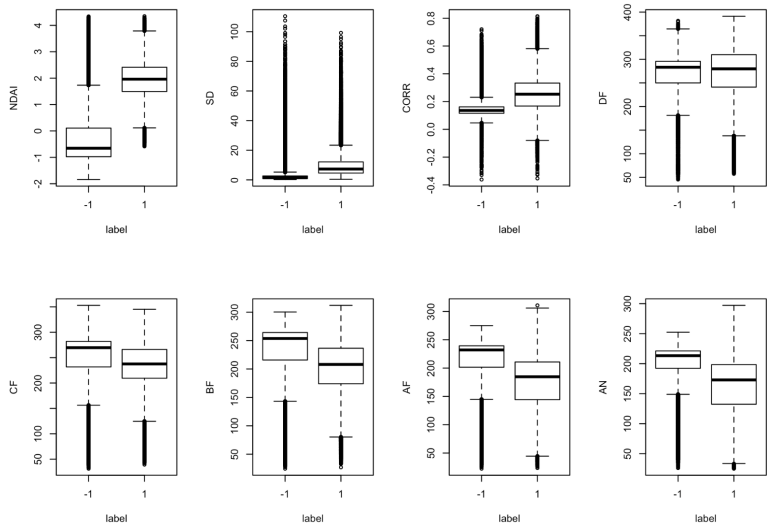


Figure 1.3: Boxplot of features

2 PREPARATION

2.1 Splitting method

Taking the invalid i.i.d assumption into consideration, I use two methods to split the data. The first method is **Average Grid Method(AGM)**. In this split, I take 20% of the total data as test set, 20% as validation set and the rest as training set by picking pixels in every small grids which is divided by $x \cdot \sqrt{0.2} \times y \cdot \sqrt{0.2}$.

What's worth mentioning is that in order to make algorithm applied to all clouds picture instead of the specific three ones provided, I combine the data from the three pictures together instead of doing separately. With the same idea, I come up with the second split method, **Image Based Method (IBM)**, in which I take the second and third pictures as training set and then split first one to half-half randomly as test and validation set respectively. In this way, future data can be added by pictures in the training set and maintain

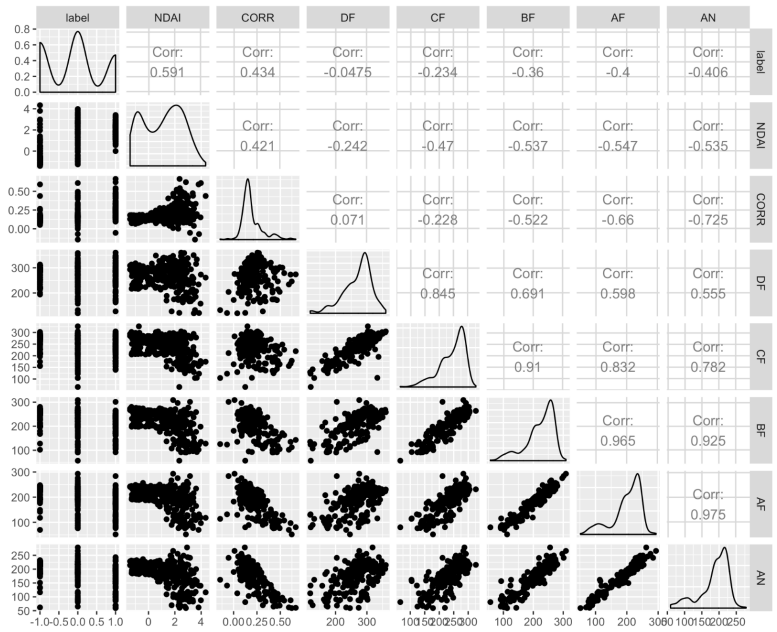
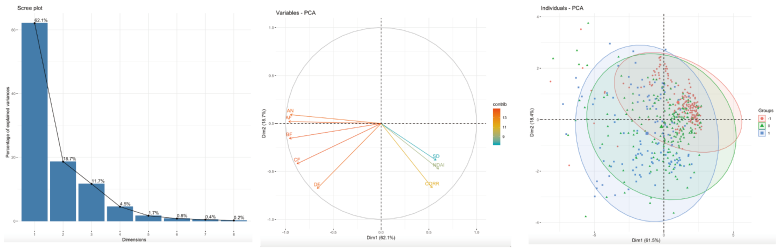


Figure 1.4: Pair plot of data



(a) Explanation rates of PCs (b) Variables directions and contribution (c) Projection of sampled data

Figure 1.5: PCA

their completeness.

As you can see, these two methods are of strong contradictory, since the former consider pixels individually and the latter consider pictures as a whole. I would discuss the performance of these two split methods in different classification ways and no surprisingly the former one yields higher accuracy. But in pragmatic situation, the latter one is more realistic, so I would take a further step to optimize the latter method in section 4.

2.2 Accuracy of trivial classifier

The trivial classifier which sets all labels to -1 on the validation set and the test set yields an accuracy of **0.847** using QDA and accuracy of **0.864** using Logistic regression. The trivial classification seems unreasonable, but actually it would obtain relatively high average accuracy in some extreme scenario where the rate of pixels labeled as cloud is very small.

2.3 Best three Features

In order to choose the "best" three features before modelling, I first apply PCA to the data. As showed in figure 2.1, the radiance angle features could classify the three groups clearly. But since the variance of five radiance angle features are much higher than CORR, NADI and SD, i.e $var[NDAI] = 2.02$ v.s. $var[AN] = 1627.80$. Thus the peripheral components ignored by PCA may also contain important information about the differences of data. Taking this into consideration, I turn to model based method to pick up features.

As we can see from table 3, the top three features in terms of importance yield by random forest, which perform best in classification in the next section are **NDAI,SD,CORR**.

2.4 CVgeneric

The generic cross validation function **CVgeneric** takes training features x, training labels y, number of folds k and loss function, which would return error rate, and default classifiers including QDA and Logistic regression as inputs, and K-fold CV loss on training set as well as the model as outputs.

3 MODELING

In this section, I fit four different classification methods, including QDA, LDA, Logistic Regression (LR),and Random Forest into the data and assess their performance using cross-validation as well as ROC curves.

I have also tried KNN and SVM method but didn't fail to yield results efficiently due to the large computational cost while RF could be quickly performed in Python. The results show that random forest is ranked top both in accuracy and in ROC curves using the first splitting method and QDA obtain the highest accuracy using the second splitting method.

3.1 Classification methods: assumption & accuracy

For different classification methods, I at first have a look at their assumptions:

- **LDA** assumes that the fitted data is Gaussian, and that each attribute has the same variance.
- **QDA** assumes that the fitted data is Gaussian.
- **Logistic regression** assumes
    1. The outcome is a binary or dichotomous variable.
    2. There is a linear relationship between the logit of the outcome and each predictor variables.
    3. There is no influential values (extreme values or outliers) in the continuous predictors.
    4. There is no high intercorrelations (i.e. multicollinearity) among the predictors.
- **Random forest** assumes i.i.d

As we can see, all methods are based on the assumption of i.i.d. Since there do exist high correlations among predictors, as indicated in section 1, the fourth assumption of logistic regression is invalid. Also, since we can not guarantee the gaussian distribution of the data, the LDA and QDA would also have invalid assumption.

The accuracies across folds and the test accuracy for different methods using two splitting methods are shown in figure 3.1. It shows that random forest is ranked top with accuracy 96.12% and while the other three obtain similar accuracy using the first splitting method and in terms of the second split way, QDA obtain the highest accuracy of 88.28% while Logistic regression yields the lowest 78.48%.

3.2 ROC curves

In order to compare the performance of different classification methods directly, I use ROC curves in figure 3.2 to visualize the differences. Since the sensitivity (TPR) and miss rate (1 - FPR) should be weighed equally in this study, in order to balance the most top and most left, I perform the numeric differentiation on the values of each curve and choose the minimum of their absolute values after log transformation. In this way, I successfully identify the most left top ones in each curve, and it is clearly shown that the using both splitting way yield the same winner, random forest, whose curve is the nearest one to point (0,1), indicating that random forest perform best.

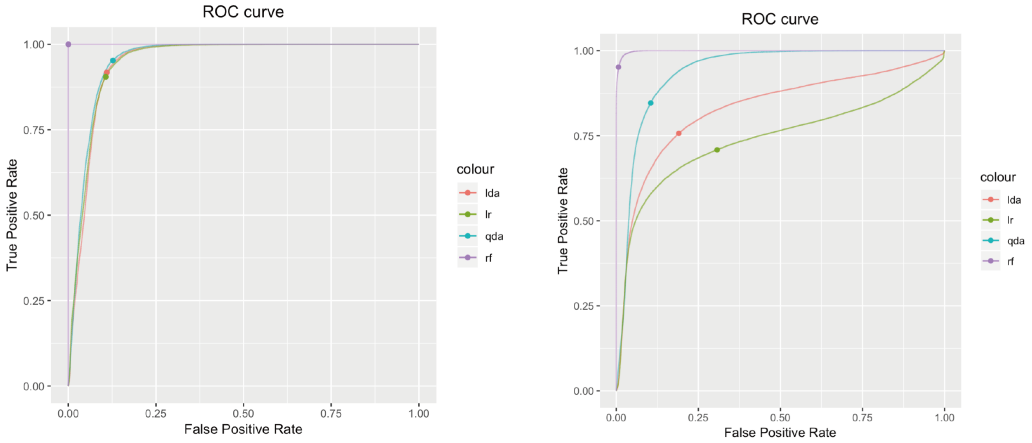
Accuracy under data splitted by Average Grid Method

	Fold1	Fold2	Fold3	Fold4	Fold5	Fold6	Fold7	Fold8	Fold9	Fold10	TEST
LDA	89.83%	89.99%	89.46%	89.69%	89.49%	90.30%	89.58%	89.76%	89.67%	89.76%	**89.77%**
QDA	89.40%	89.98%	89.64%	89.69%	89.61%	89.73%	89.60%	89.63%	89.66%	89.59%	**89.71%**
LR	89.14%	89.31%	89.20%	88.97%	89.51%	89.26%	89.01%	89.46%	89.18%	89.49%	**89.32%**
RF	96.12%	95.73%	95.88%	95.93%	95.71%	95.90%	95.85%	96.23%	96.23%	96.15%	**96.12%**

Accuracy under data splitted by Image Based Method

	Fold1	Fold2	Fold3	Fold4	Fold5	Fold6	Fold7	Fold8	Fold9	Fold10	TEST
LDA	97.58%	97.70%	97.46%	97.71%	97.64%	97.85%	97.53%	97.63%	97.63%	97.81%	**80.54%**
QDA	97.69%	97.94%	97.87%	97.53%	97.52%	97.93%	97.78%	97.67%	97.60%	98.06%	**88.29%**
LR	98.16%	98.02%	98.47%	98.36%	98.00%	98.45%	97.85%	98.03%	98.31%	98.34%	**78.48%**
RF	98.68%	98.75%	98.82%	98.63%	99.16%	98.55%	98.88%	98.74%	98.71%	99.06%	**85.17%**

Figure 3.1: Accuracy table of different classification under data splinted by two methods



(a) ROC under data splitted by AGM

(b) ROC under data splitted by IBM

Figure 3.2: ROC

3.3 Another way to assess the fit

Before discussing what kind of assessment should be claimed suitable to this problem, I design a figure to explain our consideration for the the assessment design.

As showed in figure 3.3, subfigure (c) is the label map for this designed case, (a) and (b) are two classification results with same error rate for this case. While the error rates for two classification are the same, the errors in result A occur randomly in all four regions and the errors in result B are all in the right bottom region, which corresponding to the two kinds of errors I would discuss in section 4.2. Since the second kind of errors are much harder to cope with and the first can be optimized by smoothing algorithm, I claim that result A is better than result B.

Keeping this in mind, I design a assessment of the fit to distinguish this two kind of errors even with same error rate, which first turn the classification results to matrix form via x-y axis and then use 2-norm values as criteria, the lower the value, the better the performance.

To put it mathematically, let $G(X)$ be the classification result matrix and Y be the label map, the criteria value S is obtained by:

$$S = \|G(X) - Y\|_2$$

As showed in table 4.1, I can see that by this new assessment criteria, random forest with errors randomly occuring are viewed better than QDA, whose errors mainly exis in a whole region, which verifies the effectiveness of our new assessment.

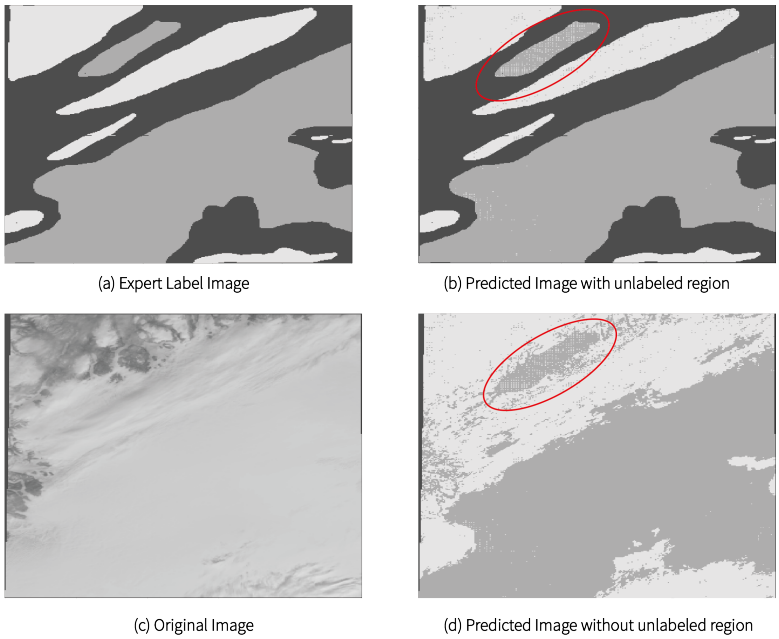
(a) Classification Result A (b) Image Classification Result B (c) Export Label Map

Figure 3.3: Example for two kinds of error

4 DIAGNOSTICS

4.1 In-depth analysis of a good classification model

As I can see from sections above, random forest method perform extraordinarily in both accuracy (with test error as small as 0.03) and ROC curves compared with other classification method, which verify again the powerful strength that random forest have when applied to large dataset. In order to have a better understanding of its performance compared with the expert labels, I present the classification outcome in figure 4.1, where subfigure (a) shows the original expert labels, subfigure (b) shows classification performed by random forest without considering the unlabeled pixels, subfigure (d) predicts the unlabeled pixels as well and subfigure (c) is the real photo . As we can see clearly from figure 4.1, the classification results by random forest are almost the same as the expert labels. At the same time, the classification for unlabeled pixels are relatively reasonable to some extent compared with the real picture.



(a) Expert Label Image (b) Predicted Image with unlabeled region

(c) Original Image (d) Predicted Image without unlabeled region

Figure 4.1: Performance of Random Forest in data splitted by AGM

4.2 Mis-classification errors

Although the classification results using random forest algorithm are almost the same as the expert labels, there still exit two kinds of mis-classification errors. The first kind of errors occur mainly near the boundaries, which can be identified and revised by the optimized algorithm later in section 4.4, and the second

kind of errors occur intensively in the a specific region, which is marked by red circle in Figure 4.1. In the region, the white small grids which should be gray in corresponding to the expert labels are exactly the pixels I picked as validation and test set according to our first splitting method —— average grid sample method. It would be more obvious in Figure 4.2 (a) using data splitted by IBM. What's worth mentioning is that, all the other classification fails in this region,which confirm our choice of random forest again, appearing white in the whole region which are labeled as gray by experts. The inconsistency between expert labels and classification results would be discussed later.

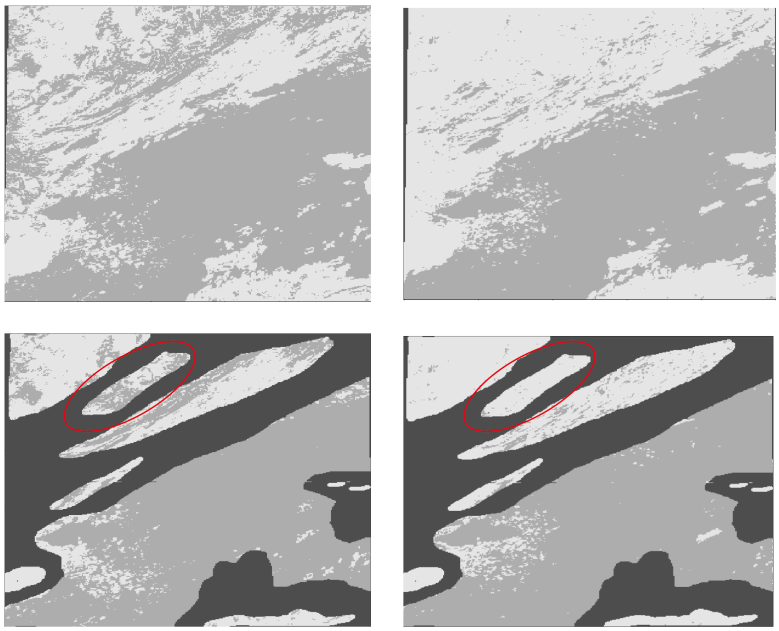
4.3 Modify splitting way (answer of -d-)

For a better reading experience, I move subsection (d) before (c) and discuss the splitting way first before analyzing the optimization of classifiers.

As I mentioned in section 2, the second splitting way – Image Based Method – should be a better choice considering the future application on more pictures and the non-i.i.d situation. The initial results of it, however, it's far from satisfactory, yielding accuracy of only 54.7%. To investigate the reasons of the poor performance, I plot the histograms of features in three pictures provided and find there exist some linear deviation among pictures, which may be resulted by the difference of light intensification or other factors. Thus I normalized the features in three pictures and then obtain accuracy 85.2%, still poorer than the first splitting method but closer to real life.

In this way, the comparison between this two splitting methods is of nonsense, since the performance of first splitting method is "fake" and the data used by these two are totally different.

Note: *since I turn to a different splitting methods, it's necessary to reconsider the choice made in section 4.1 above using new training, test and validation data. Given QDA with accuracy of 88% as a baseline for assessment of classification performance, I still choose random forest even though its accuracy is 78% for the following reason. As figure 4.2 indicates, QDA method cannot identify the specific region mentioned in section 4.1 while random forest successfully identify some as gray correctly, and the misclassification errors near the boundaries can be justified by smoothing which would be introduced in next subsection.*



(a) Predictor image of Random Forest (b) Predictor image of QDA

Figure 4.2: Performance of RF and QDA in data splitted by IBM

4.4 Better classification (answer of -c-)

In terms of the misclassification errors near the boundaries, I introduce a new indicator **score** to justify whether or not the pixel should be classified to the other group considering the correlation between neighbored pixels. The value of score S_{ij} in pixel with $x = i, y = j$ is given by

$$S_{ij} = \frac{\sum_{m=1-l}^{i+l} \sum_{n=j-l}^{j+l} \hat{P}(y = 1|x)_{mn}}{l^2} \hat{y}_{ij}$$

where l refers to the window length.
If $S_{ij} < Threshold$, we take the classification of pixel (i, j) $\hat{y}_{ij}$ to be wrong and turn it into $-\hat{y}_{ij}$. The threshold is obtained through cross validation and control the intensification of aggregation of pixels in same group.

	Random Forest	QDA	Remark
Old Accuracy Rate	85.17%	88.29%	
Accuracy Rate	92.17%	94.08%	QDA > RF
Norm	102	124	RF > QDA

Figure 4.3: Evaluation of RF and QDA after optimization

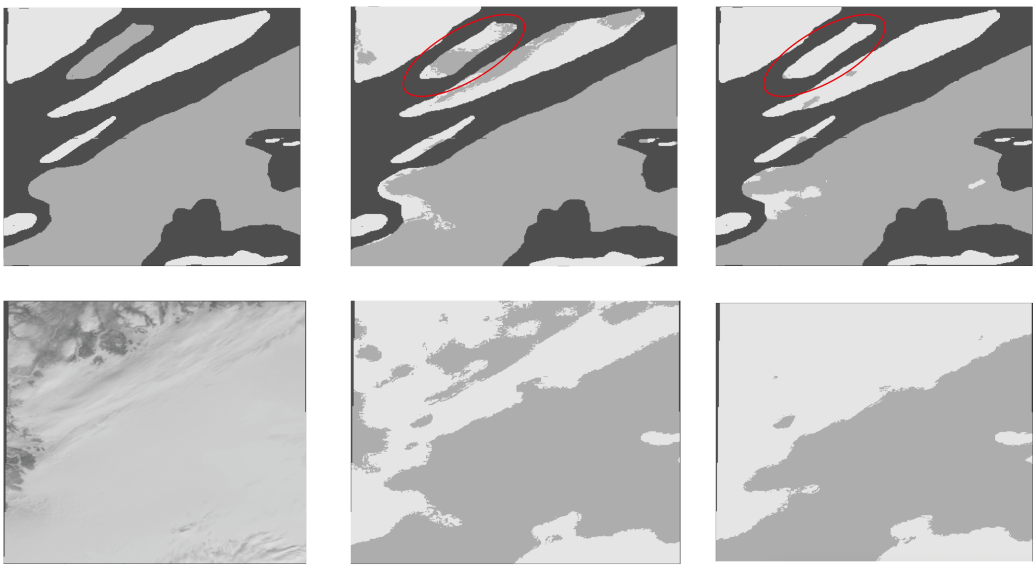


Figure 4.4: Performance of RF and QDA after optimization

As indicated in figure 4.3, after optimized, the accurary of random forest is increased to 92.1% , almost the same as the accuracy attained by QDA. But according to our new assessment method in section 3.3, the norm of random forest is 102 while that of QDA is 124, which indicates correctly that random forest perform better. The performance of this optimized classification method is showed in figure 4.4. As we can see, after optimization, the mis-classification errors near boundaries are greatly reduced while the regions marked by red circle which can not be correctly classified by QDA could be successfully identified mostly.

In consideration for future data, our classification would fit well since I've already taken whole pictures as training set in our second splitting method and yield good results after normalizing features and optimizing the classification method in terms of the errors near boundaries.

4.5 Conclusion

1. The second splitting method **Image Based Method** is of more practical meaning than the first method, though with lower accuracy, since it is applied to data by picture as a whole instead of by picked pixel grid and thus more applicable for future data.
2. Under the second splitting method, the Random forest classification perform very well in some specific regions where all other methods fail, and though its accuracy is not the highest (which is obtained by QDA method), the errors mainly occur near boundaries which can be smoothed and thus the accuracy would increase.
3. The optimization method applied to random forest classification is to increase its accuracy based on the expert labels. But the smoothing may be not that pragmatic since the experts tend to give labels to entire regions while in fact there are always cloud and not cloud small areas in an entire region.

5 POSTSCRIPT

主观感受方面的问题还是用中文表述了，我最喜欢的算法是Random Forest算法，后面简称为RF。RF算法的使用面之广是无容置疑的，可以看到很多跨领域处理大数据的研究喜欢来一个RF算法，因为它作为集成算法有着相对不错的精度表现，同时随机性也使其不容易过拟合。但是这样一个经典而又强大的算法，它的base只是一个极其简单的树形算法，二维平面上来看的话，只是一条线划分出两块区域。然而就是这样一个极其简单的分类算法，加上统计学精妙的集成理论以及剪枝等技巧，使得原本简单的算法，越拥有了很多复杂统计理论背景的算法没有的鲁棒性，这是我喜欢这套算法的原因。

这里在说一些题外话，由于现在是主要做一些计算机视觉相关的实习，所以选择了这个数据集，但是在做这份大作业的时候，我主要还是使用了课程相关的那些经典分类算法，来一一比较。然而这样的处理方式对于图像来讲是很粗糙的，对于单一像素点，并未使用周边像素点的信息（在这个报告里，我虽然对最终结果做了一些优化，但也只是很朴素点处理方式），实际上更优的做法是使用一个经典小型卷积网络backbone作为特征提取器，而后执行分割任务，最终效果会有很大提升，当然这个已经不属于我们这个课程的内容了，所以并没有去实际训练。

至于附录部分，我这里就不把代码之类的复制入报告了，老师可以通过最前面的github链接，进入这个project的repo，所有的代码都在其中。