Disaster Relief Project

Performance Metrics, 10x Cross-Validation:

Method	**Accuracy**	**AUC***
KNN (*K* = 3)	0.9972486	0.9939038
LDA	0.98395	0.9888009
QDA	0.9945762	0.998509
Logistic Regression	0.9952562	0.9981625
RF (mtry=1, ntree=100)	0.9970114	0.9990743
SVM (kernel= radial cost=50, gamma=10)	0.99743836	0.9936511

* AUC values are calculated with 50/50 (train/test) data split

Performance Metrics, Hold-Out Data:

Method	**Accuracy**	**AUC**
KNN (*K* = 3)	0.9924448	0.9386579
LDA	0.9817496	0.9921155
QDA	0.9959719	0.9915001
Logistic Regression	0.9897793	0.9994131
RF (mtry=1, ntree=100)	0.9946771	0.9826619
SVM (kernel= radial cost=50, gamma=10)	0.9904774	0.9238401

Conclusions:

- The algorithm works best in the cross-validation is the support vector machine, with the radial kernel, and cost =50, gamma at 10, if we use the accuracy as our criteria. According to accuracy values from 10x CV, KNN (K=3) and random forest (mtry=1, ntree=100) also performed very well, ranked as #2, and #3 best models. Logistic regression, QDA, and LDA finish as #4, #5, and #6, respectively.
- The algorithm works best on the hold-out data is QDA, with an accuracy of 0.9959719. The random forest and KNN work as the 2nd and 3rd best, followed by SVM, Logistic Regression, and LDA as #4, #5, and #6, respectively.

Conclusions:

- Overall, I would recommend QDA as the method for detection of blue tarps. The hold-out data test is a better test than the 10x CV. Although in principle cross-validation removes the dataset being tested from the dataset of model-training, it still entails the correlation between two sets of data since all of these 10 tests are correlated by a large amount of shared data. On the other hand, the hold-out data approach ensures the strict separation between training and testing data, i.e., it is a blind test. I would value the output from the hold-out data more than the 10x CV.
- Another added benefit of QDA is its relative simplicity. This model fitting needs a lot less computing power, and will be able to perform the task in a short period of time especially if we are dealing with a huge amount of data in this urgent rescue mission. In comparison, more flexible methods, such as SVM, KNN, and random forest will require a lot more overhead on computation, which may delay the critical rescue mission.

Conclusions:

- The comparisons between AUC and accuracy values are quite fascinating. In the first set of AUC with 10x CV approach, I used a 50/50 split of data, in essence, these values are on the testing data with the model fitted using the training data. However, there is a caveat, since the best parameters of more flexible models (K in KNN, mtry in RF, and kernel type, cost, gamma in SVM) were actually determined by 10x CV on the whole dataset. To get testing AUCs on the 10x CV approach, the exact models as confined by these parameters were fitted with the 50% of data (31620 observations), which in turn were used to calculate AUC for the testing data (31621 observations). For the hold-out data approach, the exact models were defined by the whole training data (63241 observations), then the test AUC is calculated on the hold-out data (2004177 observations). For both sets of AUC values, There appears to be little correlation with their corresponding accuracies. Highest accuracy does not translate to highest AUC. AUC is a measure of performance across all possible classification thresholds, whereas accuracy is defined at the default threshold 0.5. For example, in the hold-out data approach, the logistic regression has the highest AUC value of 0.9994131, this basically means logistic regression will on average perform the best if we use various classification thresholds. These results signify that accuracy and AUC are two different measures of the performance of algorithms, the highest value in one does not guarantee the best in the other.

Conclusions:

- Based on the rule of bias-variance trade-off, flexible algorithms tend to have lower bias, but higher variance errors, whereas less-flexible methods like logistic regression, LDA, and QDA methods will be more likely to have higher bias error, but less variance errors. In the 10x CV approach, flexible methods (KNN, RF, and SVM with radial kernel) have higher accuracy, whereas in the holdout data, less flexible methods (LR, LDA, and QDA) actually performed relatively better in accuracy, and totally dominated flexible methods in terms of AUC. This result indicated that holdout data have different boundaries/distributions than training data, and flexible methods may be overfitting the data in training.
- All of these models achieved accuracies of more than 98% even in the hold-out data, the need for additional improvement is probably not high. But if we have to, we may be able to use feature engineering methods, as well as boosting methods, such as XGBoost, to further increase the accuracy.

Conclusions:

- Based on these fitting results, we can interpret the type of data that we are dealing with. The data is highly nonlinear, the worst performing algorithm is LDA in both CV and the hold-out dataset. In addition, logistic regression also ranked very close to the bottom of accuracy, as the third worst in the CV, and the 2nd worst in the hold-out data set. These two algorithms share one common feature: they both have linear boundaries. In addition, logistic regression performed better than LDA, it probably indicated that the data do not have normal distribution, i.e., non-Gaussian distribution. One key difference between LDA and logistic regression is that LDA assumes that the data set has normal distribution of variances.
- The nonlinear data boundary is also confirmed by the feature parameters in more flexible algorithms. For example, the best KNN model has a K=3 as determined by CV, however such a small K value may incur higher variance error. Similarly, the SVM model performed the best in the cross validation approach, with a flexible radial kernel, and with a relatively high cost value at 50, and high gamma factor at 10. Higher cost and higher gamma values indicated that the model is highly flexible, and prone to higher variance error, i.e, may overfit the data. It is not surprising that both KNN(K=3) and SVM(radial kernel, cost=50, gamma=10) performed worse in the holdout data. They were the two top performing models in the CV approach, but dropped to 3rd and 4th in the holdout data.

Conclusions:

- The other flexible algorithm random forest usually do not suffer from overfitting as much as KNN and SVM, since the parameter mtry is designed to de-correlate among different trees. It ranked as a close 3rd place in the CV approach, and ranked 2nd best in the holdout data. For our particular case, we are dealing with a dataset of only 3 features, so the variation of random is very limited, i.e., with only 3 choices. I would expect this algorithm to outperform QDA if our dataset has more features.
- It is somewhat surprising that QDA is the best performing algorithm since this approach assumes normal distribution of variances, and I suspected earlier that data may not have normal distribution since logistic regression outperforms LDA. Of course, I also analyzed shortcomings for each algorithm. One final takeaway from this project is probably that there is no single best algorithm for all datasets. All algorithms have advantages and shortcomings. For this blue-tarp identification problem, the QDA is the best method.