Modeling Credit Risk

Introduction

The goal of this analysis is to predict the probability a person will have a serious delinquent loan within the next two years. The data includes 150,000 historical observation with 10 features including demographic, financial, and past historical information on someone's loan history. Serious delinquent loans accounted for about 6.7% of the observed data

Models

Seven types of Models were tried in the analysis: Random Forest, Logistic Regression, A Stochastic Linear Gradient Model using a Liner Support Vector Machine, Decision Trees, K Nearest Neighbors, Gradient Boosting and Ada Boost.

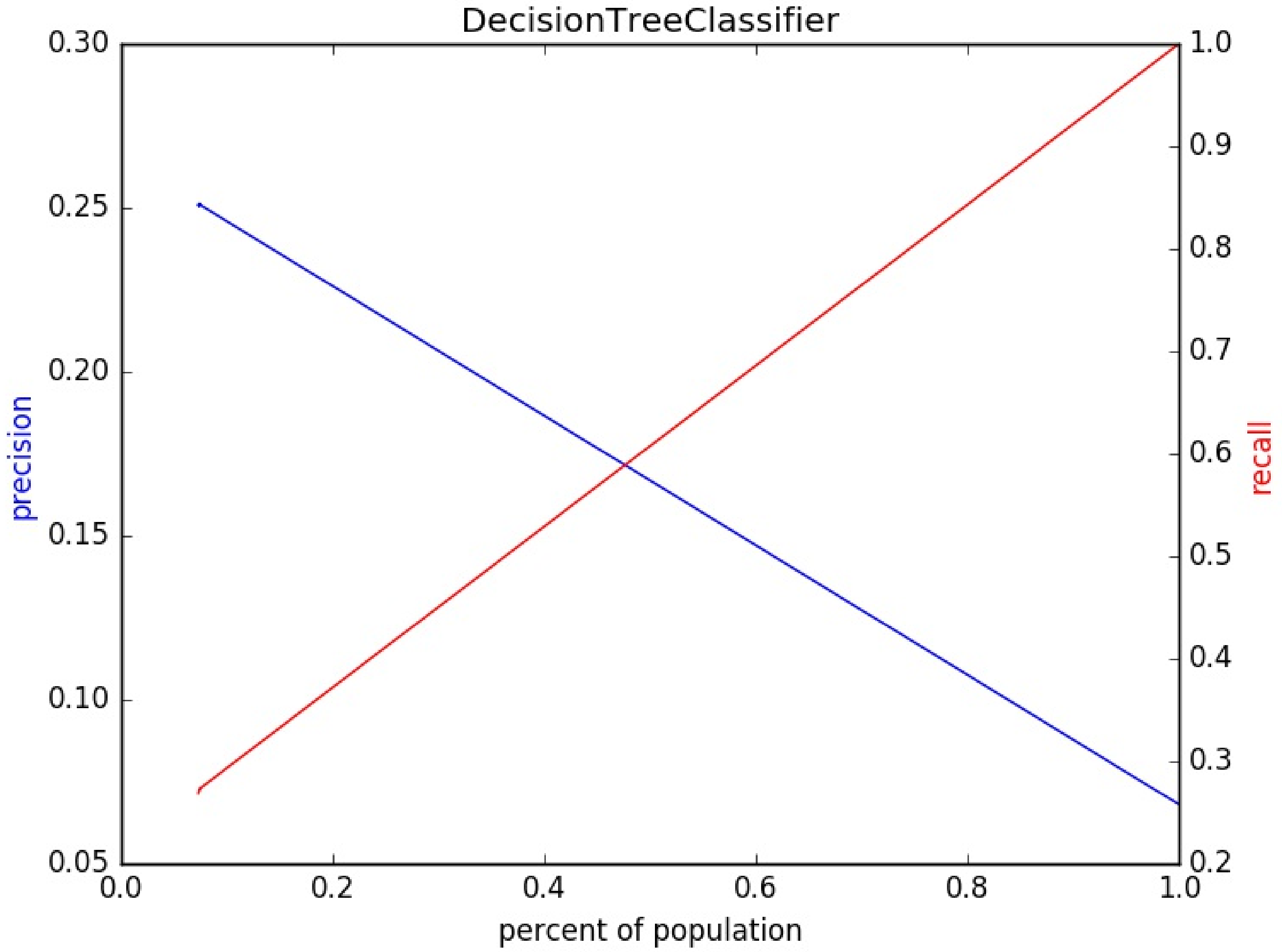
Evaluation Metrics

In the context of lending, banking institutions are likely most concerned with recall or being sure they detect as many of the true instances of people who will have serious delinquent loans in the next two years.

However, they may be concerned with turning too many customers away and have a threshold precision level they would like to meet. Since this is a binary classifier (will have a serious loan issue or will not), the area under the precision recall curve provides an appropriate and intuitive metric balancing precision and recall. The baseline, will be 0.5 which is the area that would be underneath the curve if we employed random guessing and the false positive rates were linear with one another. Alternatively, we could use an F1 Score as a weighted average of the recall and precision.

Model Performance

Overall, all the models performed well on accuracy - since serious delinquent loans are rare and the models predicted few positive results. However, this also meant our recall generally was not high. The model which performed the best overall was the Decision Tree Classifier. Although, it had lower accuracy than some of the other models - its area under the recall precision curve (pictured below) was the highest meaning it had the best balance of precision and recall.



Appendix

Results Table:

DecisionTreeClassifier(class_weight=None, criterion='gini', max_depth=None, max_features=None, max_leaf_nodes=None, min_samples_leaf=1, min_samples_split=2, min_weight_fraction_leaf=0.0, presort=False, random_state=None, splitter='best')

Accuracy: 0.9
Precicision: 0.25
Recall: 0.27
AUC: 0.61
F1 Score: 0.26
Run Time: 4.76 seconds

AdaBoostClassifier(algorithm='SAMME', base_estimator=DecisionTreeClassifier(class_weight=None, criterion='gini', max_depth=1, max_features=None, max_leaf_nodes=None, min_samples_leaf=1, min_samples_split=2, min_weight_fraction_leaf=0.0, presort=False, random_state=None, splitter='best'), learning_rate=1.0, n_estimators=200, random_state=None)

Accuracy: 0.94
Precicision: 0.57
Recall: 0.19
AUC: 0.59

```
F1 Score: 0.29
Run Time: 79.74 seconds
```

```
GradientBoostingClassifier(init=None, learning_rate=0.05, loss='deviance',
max_depth=6, max_features=None, max_leaf_nodes=None,
min_samples_leaf=1, min_samples_split=2, min_weight_fraction_leaf=0.0,
n_estimators=10, presort='auto', random_state=None, subsample=0.5,
verbose=0, warm_start=False)

Accuracy: 0.93
Precicision: 0.0
Recall: 0.0
AUC: 0.5
F1 Score: 0.0
Run Time: 15.47 seconds
```

```
KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minkowski',
metric_params=None, n_jobs=1, n_neighbors=3, p=2, weights='uniform')

Accuracy: 0.93
Precicision: 0.26
Recall: 0.05
AUC: 0.52
F1 Score: 0.09
Run Time: 12.14 secondss
```

```
LogisticRegression(C=100000.0, class_weight=None, dual=False,
fit_intercept=True, intercept_scaling=1, max_iter=100, multi_class='ovr',
n_jobs=1, penalty='l1', random_state=None, solver='liblinear', tol=0.0001,
verbose=0, warm_start=False)
```

```
Accuracy: 0.93
Precicision: 0.52
Recall: 0.1
AUC: 0.55
F1 Score: 0.17
Run Time: 8.94 seconds
```

```
RandomForestClassifier(bootstrap=True, class_weight=None,
criterion='gini', max_depth=None, max_features='auto',
max_leaf_nodes=None, min_samples_leaf=1, min_samples_split=2,
min_weight_fraction_leaf=0.0, n_estimators=50, n_jobs=-1,
oob_score=False, random_state=None, verbose=0, warm_start=False)

Accuracy: 0.93
Precicision: 0.53
Recall: 0.18
AUC: 0.58
F1 Score: 0.26
Run Time: 59.01 seconds
```

```
SGDClassifier(alpha=0.0001, average=False, class_weight=None,
epsilon=0.1, eta0=0.0, fit_intercept=True, l1_ratio=0.15,
learning_rate='optimal', loss='log', n_iter=5, n_jobs=1, penalty='l2',
power_t=0.5, random_state=None, shuffle=True, verbose=0,
warm_start=False)

Accuracy: 0.93
Precicision: 0.58
Recall: 0.01
```

AUC: 0.5

F1 Score: 0.02

Run Time: 0.57 seconds