Credit Card Fraud Detection

By Steven Lacy and Amanda Abbott

Motivation & Summary

Credit card fraud is a modern problem that affects everyone to some degree.

- Costs time
- Costs money
- Costs human resources to track and resolve
- Shakes our sense of security.

Machine learning models to detect fraud can be built using code and data science libraries, and they can also be built using tools such as Amazon Web Services Autopilot program.

Which approach to building machine learning models to detect credit card fraud is best - an automated approach or building a model?

Which type of model from the myriad choices available works the best?

Model Summary

We used Amazon Autopilot to build two models optimized for AUC score and F1 score. Both are XGBoost models. In addition, we built five models ourselves, including an XGBoost model.

Name	Value
SageMaker.ImageUri	683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-xgboost:1....
SageMaker.InstanceCount	1
SageMaker.InstanceType	ml.m5.4xlarge
SageMaker.VolumeSizeInGB	50
_tuning_objective_metric	validation:auc
alpha	0.000342442465138095
colsample_bytree	0.804529120945972
eta	0.010080463642658579
eval_metric	accuracy,f1_binary,auc
gamma	7.874928457844001e-05
lambda	0.3243854551471722
max_depth	4
min_child_weight	3.5224880265502856
num_round	664
objective	binary:logistic
save_model_on_termination	true
scale_pos_weight	577.2893401015228
subsample	0.9230368693045823

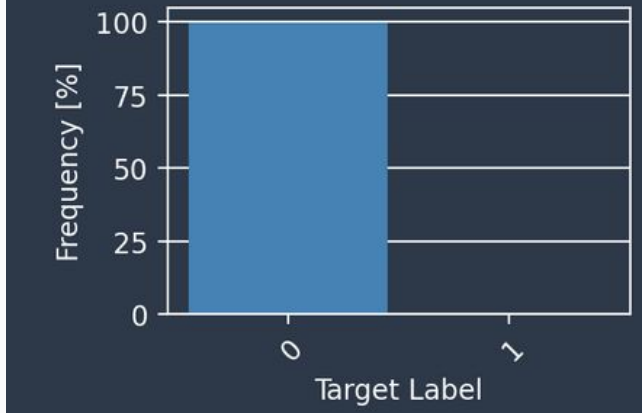
Data Cleanup & Model Training

The Credit Card Fraud Detection dataset is a very imbalanced dataset, with 0.172% of target values being 1 (fraudulent) and the rest 0 (not fraudulent).

Autopilot used RobustStandardScaler from Scikit-learn to scale the features. RobustStandardScaler removes the median and then scales the data between the first and third quartiles (interquartile range).

We used RobustStandardScaler in our “home grown” models too.

Target Label	Frequency Percentage	Label Count
0	99.83%	227452
1	0.17%	394



Histogram of the target column labels.

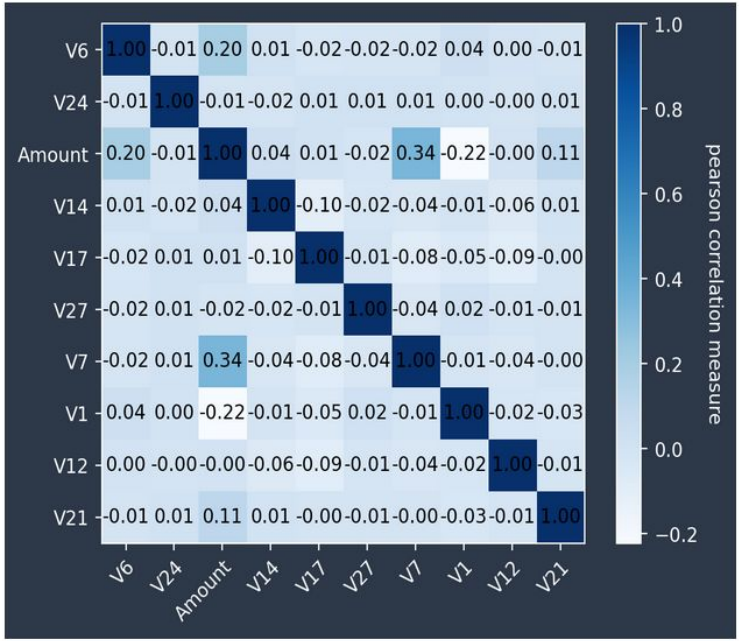
Data Cleanup & Model Training

Most of the features of the dataset are anonymized due to privacy concerns.

We spotted correlations between some of the features and would love to know what the data represents.

Autopilot split the data into training and testing sets, and we did the same with our models.

Autopilot took care of the hyperparameter tuning and model training. We didn't do a lot of tuning with our models and ran the training as we've done throughout the course.



Cross column correlation for numeric features

Autopilot Model Evaluation

Autopilot XGBoost Model Optimized for AUC

Name	Minimum	Maximum	Standard Deviation	Final value
ObjectiveMetric	0	0.97974	0	0.9926300048828125
validation:auc	0	0.97974	0	0.9926300048828125
train:auc	0	0.99533	0	0.9998800158500671
validation:accuracy	0	0.99133	0	0.9966800212860107
validation:f1_binary	0	0.24197	0	0.4532400071620941
train:f1_binary	0	0.27881	0	0.5352100133895874
train:accuracy	0	0.9913	0	0.9969199895858765

Autopilot Model Evaluation

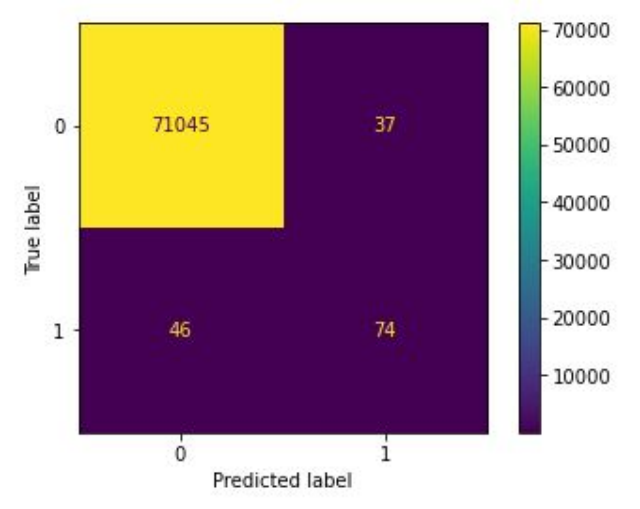
Autopilot XGBoost Model Optimized for F1 Score

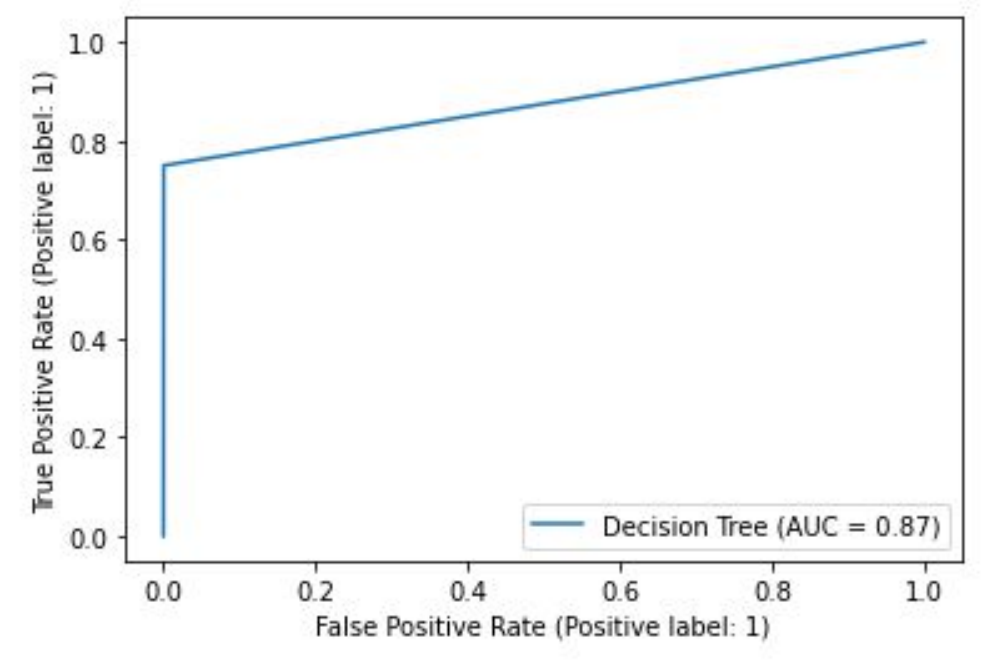
METRICS

Name	Minimum	Maximum	Standard Deviation	Final value
ObjectiveMetric	0	0.41081	0	0.8474599719047546
train:auc	0	0.99927	0	1
validation:auc	0	0.96884	0	0.9645599722862244
validation:accuracy	0	0.99615	0	0.9995200037956238
validation:f1_binary	0	0.41081	0	0.8474599719047546
train:f1_binary	0	0.45879	0	1
train:accuracy	0	0.99629	0	1

Decision Tree Model Evaluation

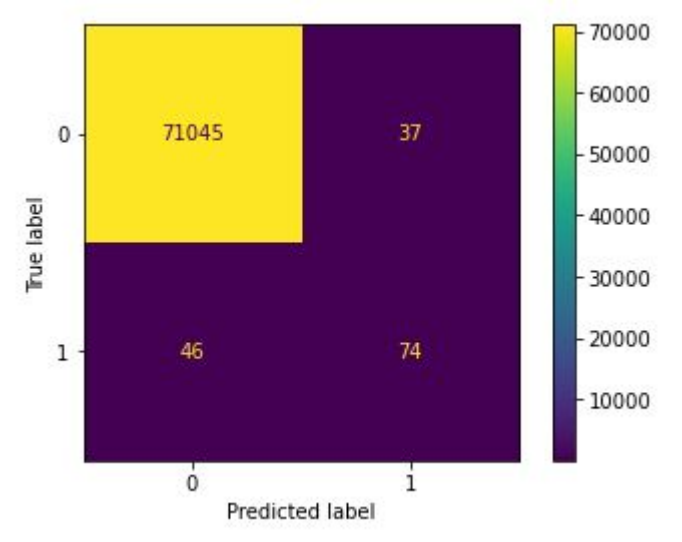
AUC Score: 0.8707
F1 Score: 0.7574
Accuracy: 0.9992

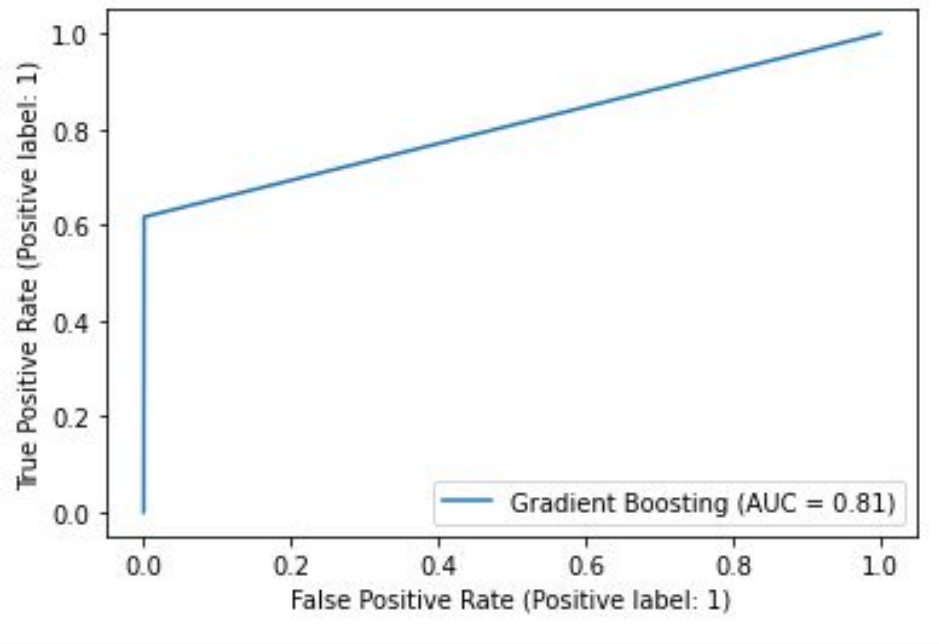




Gradient Boosting Model Evaluation

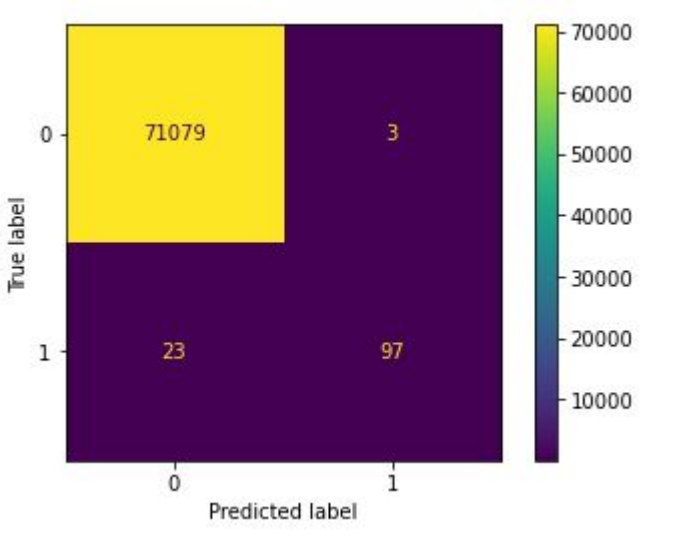
AUC Score: 0.8081
F1 Score: 0.6407
Accuracy: 0.9988

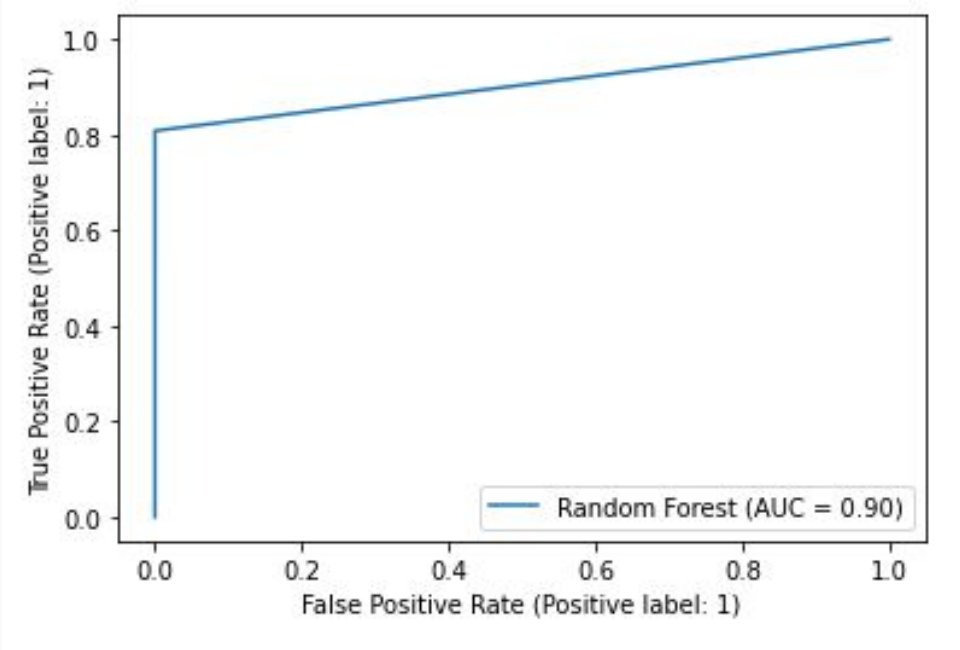




Random Forest Model Evaluation

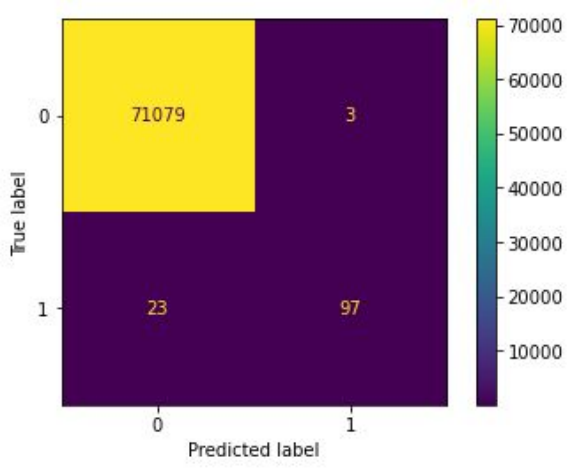
AUC Score: 0.9041
F1 Score: 0.8818
Accuracy: 0.9996

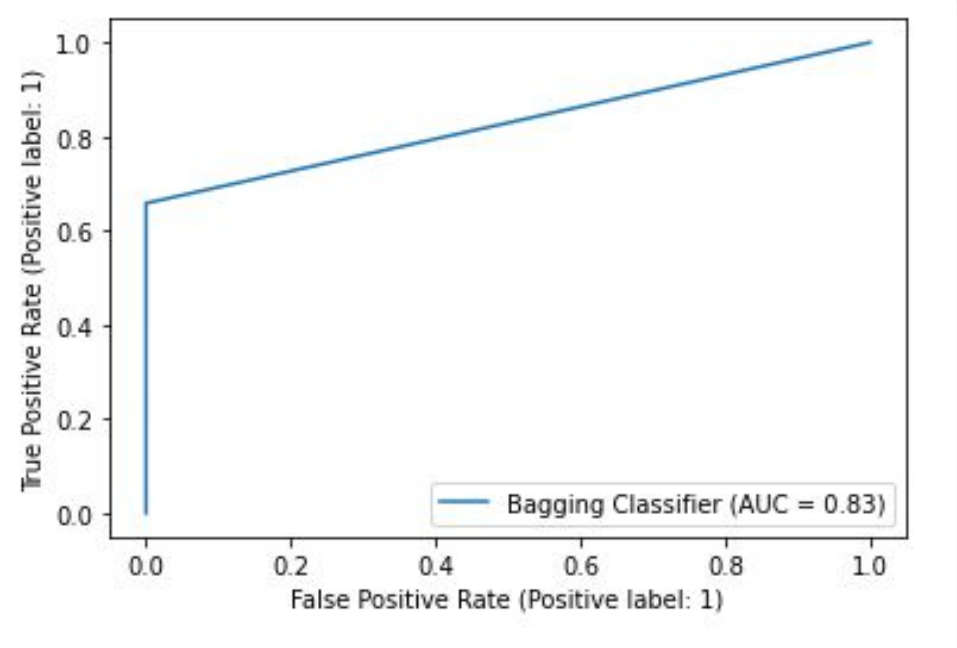




Bagging Classifier Model Evaluation

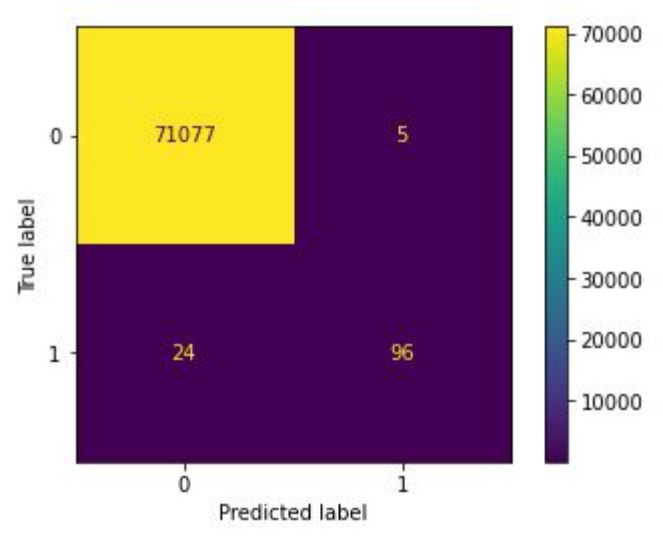
AUC Score: 0.8291
F1 Score: 0.7822
Accuracy: 0.9994

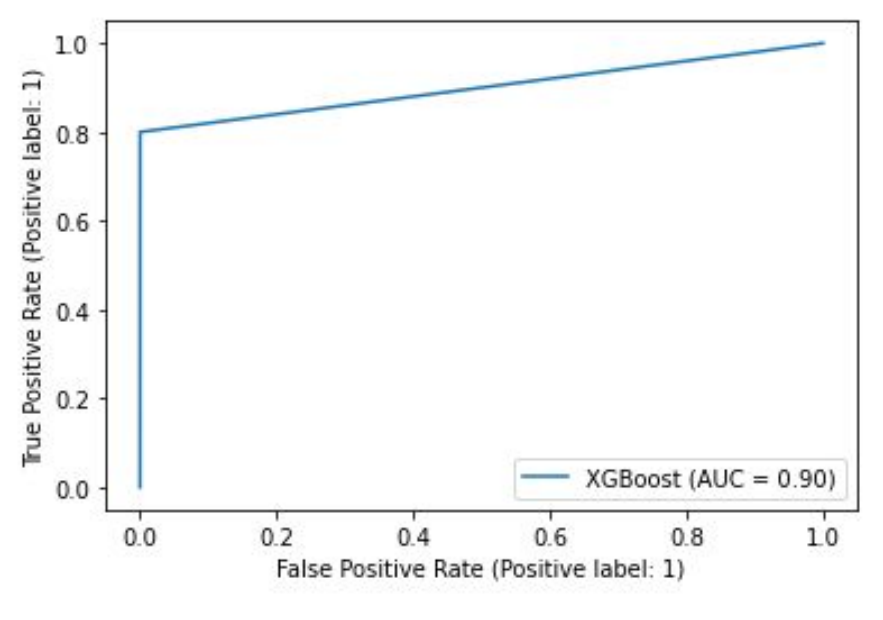




XGBoost Model Evaluation

AUC Score: 0.9000
F1 Score: 0.8688
Accuracy: 0.9996





Discussion

- Overall top performing model: Autopilot-generated XGBoost model optimized for AUC score
- Top performing “home grown” model: Random Forest

Were we surprised that the machine beat the humans?

No! The Autopilot program is very robust - we were impressed.

Detecting credit card fraud is “just” a binary classification problem, but it’s complex. A program like Autopilot is well-suited to the task.

Model	AUC Score	F1 Score	Accuracy
AWS AUC Optimzed	0.9926	0.4532	0.9967
AWS F1 Score Optimized	0.9688	0.4108	0.9962
Decision Tree	0.8707	0.7574	0.9992
Gardient Boosting	0.8081	0.6407	0.9988
Random Forest	0.9041	0.8818	0.9996
Bagging Classifier	0.8291	0.7822	0.9994
XGBoost	0.9000	0.8688	0.9996

Postmortem

We struggled with obtaining additional metrics from Autopilot. If we had additional time, we would have built the code from scratch, instead of using another programmer's code, to truly understand all the variables and parameters involved. Or we'd just use the GUI!

We'd also like to revisit the models we built and test resampling techniques with them. And of course, there are always more models to build...

Q & A