

A conceptual image featuring a robotic arm with a metallic, articulated structure. The arm is holding a glowing incandescent lightbulb. The arm's gripper is positioned near a white electrical outlet on a wall with a textured, wavy pattern. A white power cord is plugged into the outlet, and another white cord is connected to the base of the lightbulb. The scene is dimly lit, with the primary light source being the glowing bulb, which casts a warm, yellowish glow. The background is dark and out of focus.

Detecting Electricity Fraud in a Spanish Utility

Data Science for Business #AA – Team 8

Sophie Alderman, Lindsay Axford,
Guillermo de Miguel, James McGoldrick,
Nambaya Ouattara, Tamur Yusifzai

Combating Losses Due to Electricity Fraud

Following the great recession and resulting economic crisis in Spain, a Spanish utility company observed a surge in energy losses.

In 2013, the company completed over **28'000** inspections, confirming **9'000** cases of fraud

Can the company **reduce inspection costs** yet still **detect fraud**?



Potential for Using Data Science

- “Real” data for the cost of fraud not available; therefore, assumed values have been used.

Assumed Values:

Cost per inspection	168 €¹
Average cost of fraud	740 €²

Assuming we achieve

70% Accuracy:

Detect 75% of frauds and
75% reduction of
fraudless inspections

15% reduction in costs giving
savings of **800 thousand €**

¹ Based on expert assessment

² Based on average electricity losses/fraud case in 2018 and the average cost of electricity in the same period

Dataset: Spanish Electricity Customers

- Dataset contained 28,734 inspections on customers during 2013
- Inspections were conducted on suspicious customers
- Attached to each customer were 49 independent variables related to:
 - Socio Demographic Status Region, city, etc.
 - Tariff Rates HV or LV, # of supply cut offs, # of claims
 - Installation Type Supply Voltage
 - Measurement Equipment Direct or remote measure
 - Consumption Patterns Consumption (avg, max, cum)
- The data is biased towards detecting fraud
 - Not necessarily possible to detect fraud with greater accuracy on general population of customers

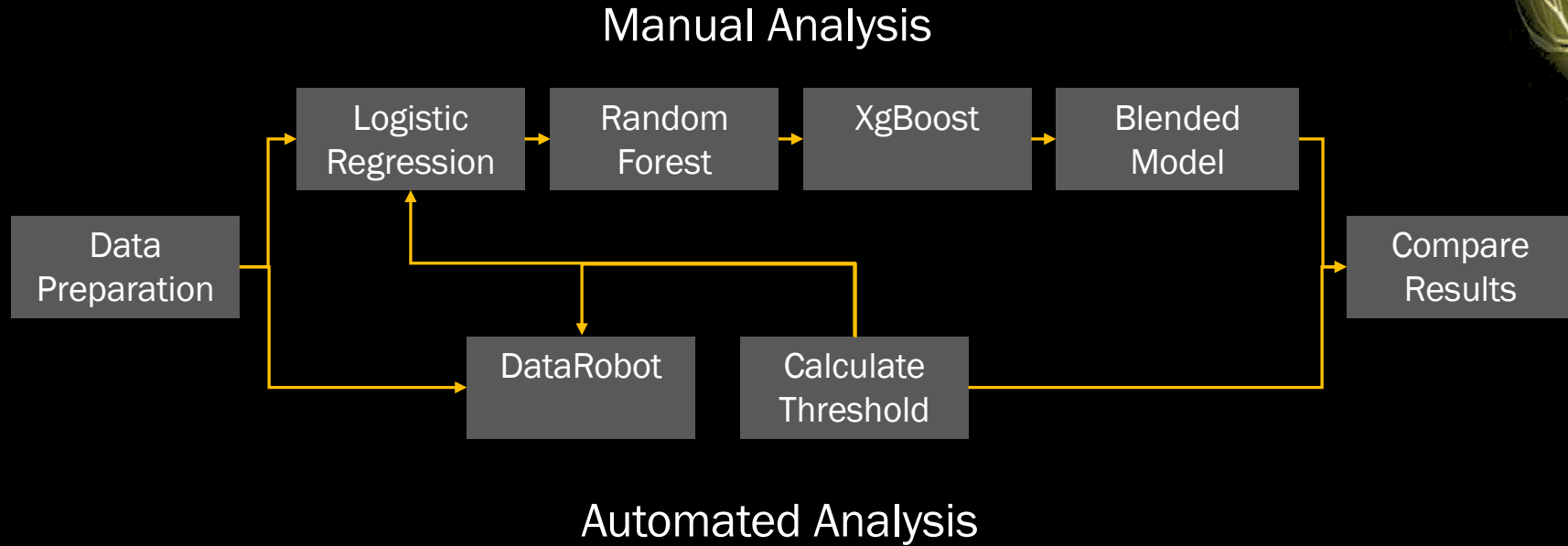
Framework for interpreting Model Results

Reduce Costs by
Maximizing Negative
Predictive Value



Predicted	Actual		Maximize Neg. Pred Value
	-	+	
	-	+	
-	Inspect: ✗ Fraud: ✗ Cost savings: 168 €	Inspect: ✗ Fraud: ✓ Cost savings: -554 €	
	Inspect: ✓ Fraud: ✗ Cost savings: 0 €	Inspect: ✓ Fraud: ✓ Cost savings: 0 €	
Specificity		Sensitivity	

Classification Model Prediction Process



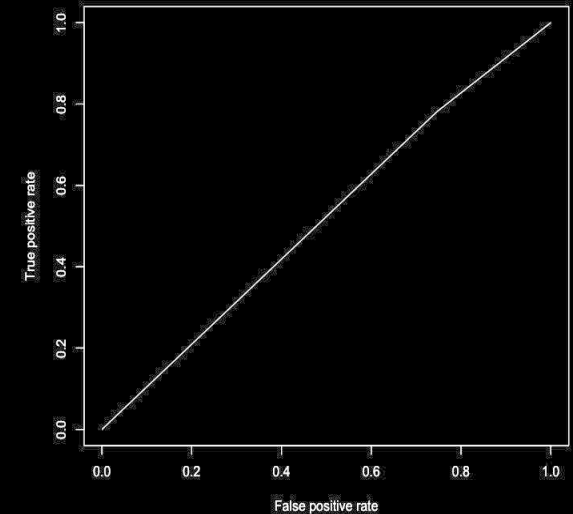
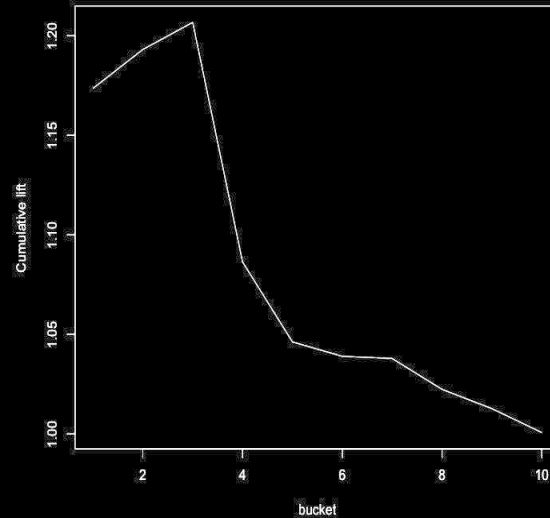
Results: Logistic Regression

Data Conditioning

- Remove variables with very large numbers of categories (e.g. municipality: >1000, postal code: >2000) as RStudio cannot handle this many.
- Tested combining these categories to reduce number down to 50. Final decision was to remove these variables completely from the regression.
- Remove categories with only 1 factor (e.g. sector code, sector description)

Confusion Matrix

	0	1
0	1465	613
1	4339	2202



Accuracy = 0.4255

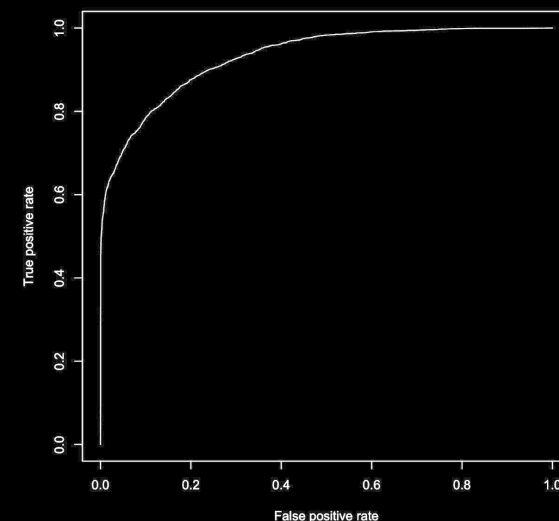
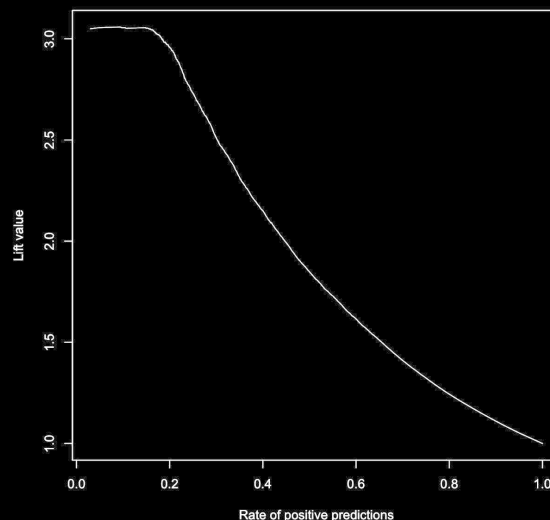
Results: Random Forest

Data Conditioning

- Remove variables with very large numbers of categories (e.g. municipality: >1000, postal code: >2000). Random Forest will not run with more than 54 categories per variable.
- Tested combining categories to reduce number of variables down to 50. Final decision was to remove these variables completely from the regression.
- Remove categories with only 1 factor (e.g. sector code, sector description)

Confusion Matrix

	0	1
0	5560	880
1	244	1935



Accuracy = 0.8696

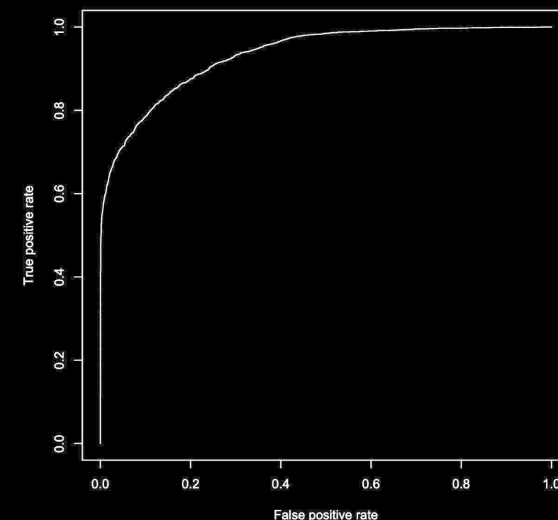
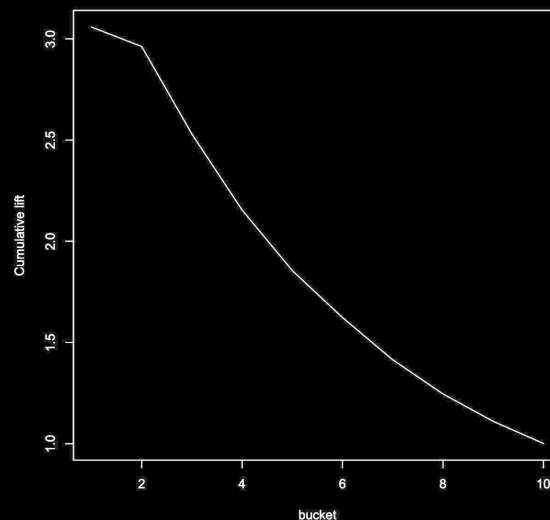
Results: Xgboost

Data Conditioning

- Remove categories with only 1 factor (e.g. sector code, sector description)
- No need to remove variables with high number of categories.

Confusion Matrix

	0	1
0	5668	968
1	136	1847



Accuracy = 0.8708

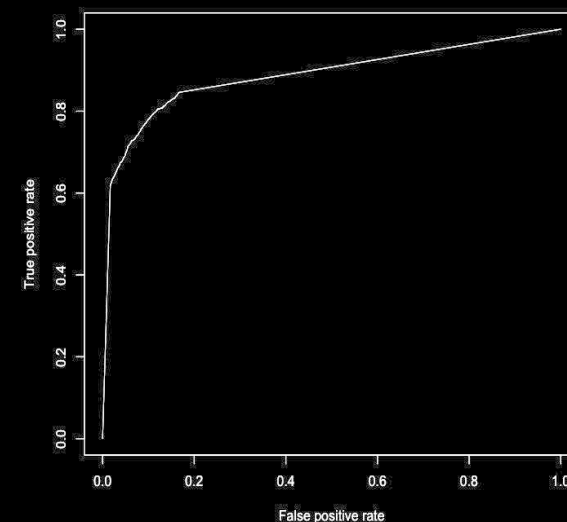
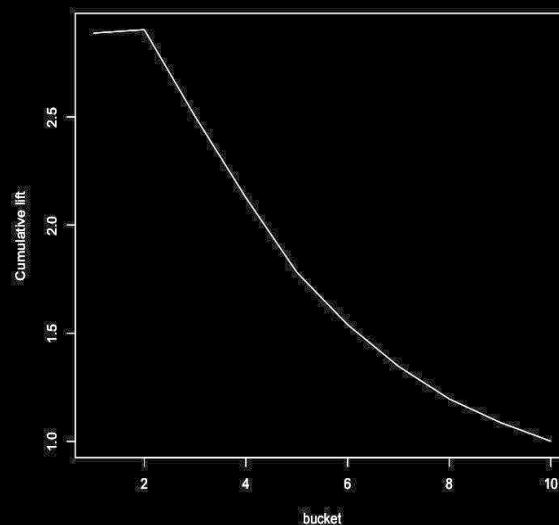
Results: Blended Model

Data Conditioning

- Remove categories with only 1 factor (e.g. sector code, sector description)
- No need to remove variables with high number of categories.

Confusion Matrix

	0	1
0	5463	794
1	341	2021



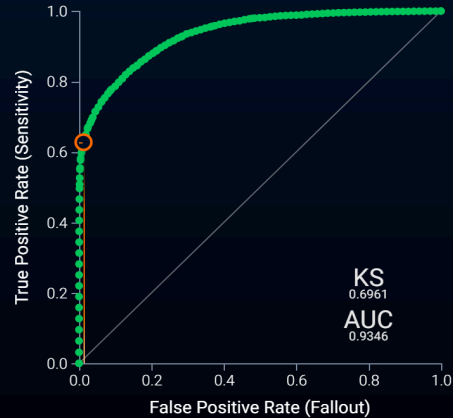
Accuracy = 0.8683

Results: DataRobot Recommended Model

	DataRobot Xgboost	RStudio Xgboost
AUC	0.9346	0.9341
Sensitivity	0.6324	0.6561
Specificity	0.9907	0.9766
Pos Pred Value	0.9706	0.9314
Neg Pred Value	0.8475	0.8541

ROC Curve

Data Selection: Cross Validation



Model Name & Description

Feature List & Sample Size

Validation

Cross Validation

Holdout

XG Boost

eXtreme Gradient Boosted Trees Classifier with Early Stopping and Unsupervised Learning Features

Tree-based Algorithm Preprocessing v22 with Unsupervised Learning Features

M111BP69

RECOMMENDED FOR DEPLOYMENT

Informative Features

80.0 %

0.9378 *

0.9346 *

0.9415

Evaluate

Understand

Describe

Predict

Lift Chart

ROC Curve

Feature Fit

Advanced Tuning

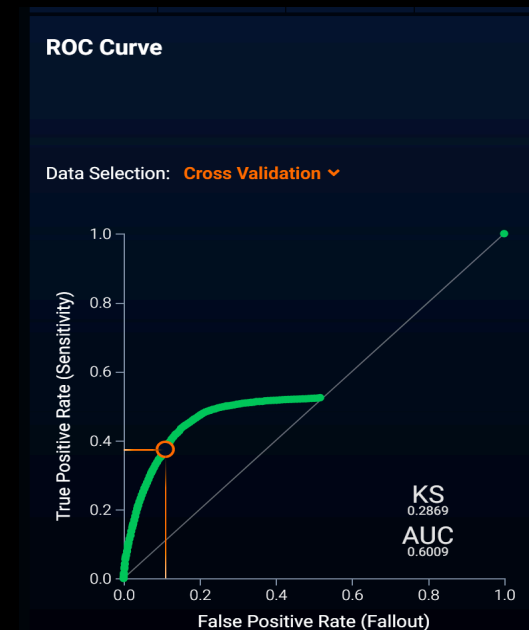
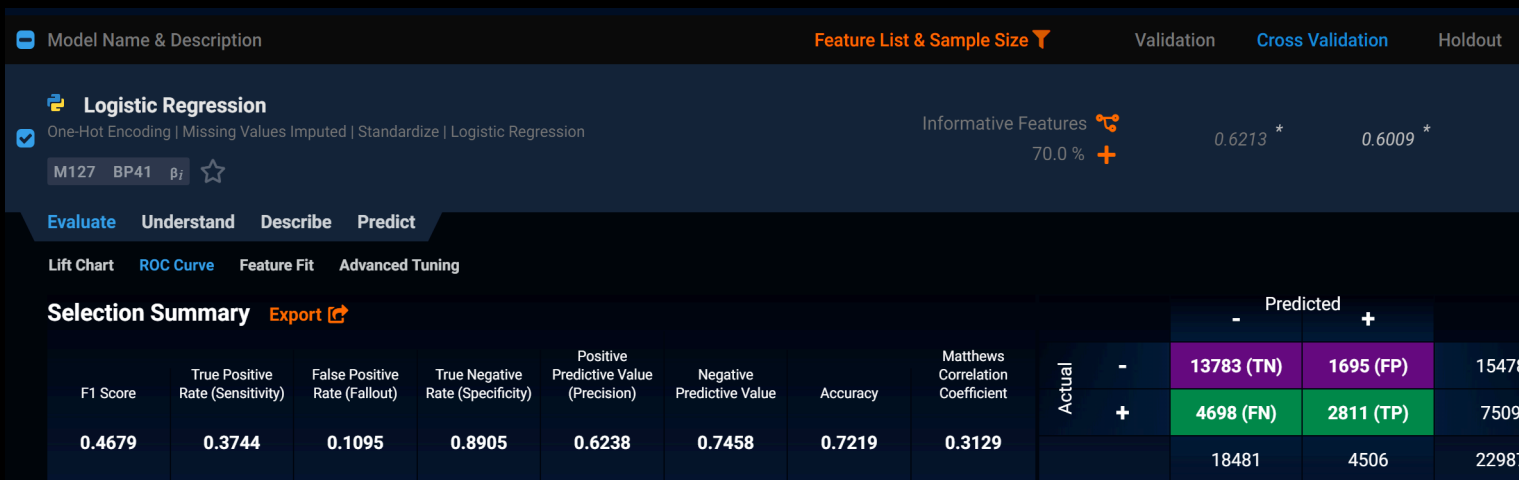
Selection Summary

Export

								Predicted			
								-	+		
F1 Score	True Positive Rate (Sensitivity)	False Positive Rate (Fallout)	True Negative Rate (Specificity)	Positive Predictive Value (Precision)	Negative Predictive Value	Accuracy	Matthews Correlation Coefficient	Actual -	15286 (TN)	192 (FP)	15478
0.7593	0.6276	0.0124	0.9876	0.9609	0.8454	0.87	0.7043	Actual +	2796 (FN)	4713 (TP)	7509
									18082	4905	22987

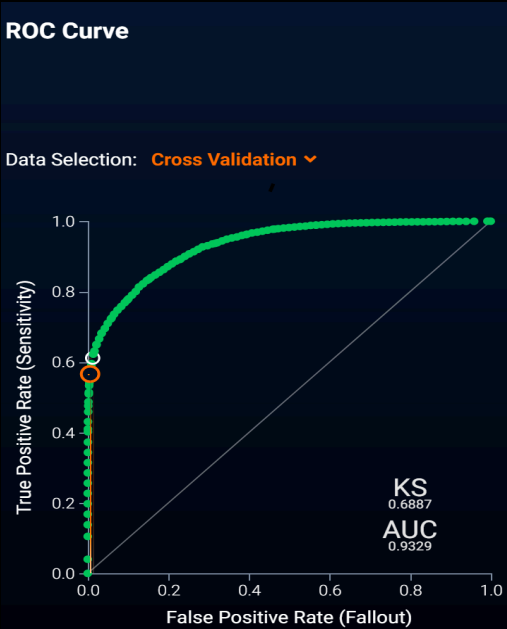
Results: DataRobot vs Manual Logit Model

	DataRobot Logit Regression	RStudio Logit Regression
AUC	0.6213	0.5173
Sensitivity	0.3744	0.7822
Specificity	0.8905	0.2524
Pos Pred Value	0.6238	0.3366
Neg Pred Value	0.7458	0.7050



Results: DataRobot vs Random Forest

	DataRobot Random Forest	RStudio Random Forest
AUC	0.9329	0.9320
Sensitivity	0.5661	0.6874
Specificity	0.9941	0.9580
Pos Pred Value	0.9790	0.8880
Neg Pred Value	0.8253	0.8634



Results: Comparison of all models

Model	Logit Regression R	Logit Regression DataRobot	Random Forest R	Random Forest DataRobot	Xgboost R	Xgboost DataRobot	Blended Model R
Sensitivity	0.7822	0.3744	0.6874	0.5661	0.6561	0.6324	0.7179
Specificity	0.2524	0.8905	0.9580	0.9941	0.9766	0.9907	0.9412
Pos Pred Value	0.3366	0.6238	0.8880	0.9790	0.9314	0.9706	0.8556
Neg Pred Value	0.7050	0.7458	0.8634	0.8253	0.8541	0.8475	0.8731
AUC	0.5173	0.6213	0.9320	0.9329	0.9341	0.9346	0.8874
Accuracy	0.4255	0.7219	0.8696	0.8543	0.8708	0.8700	0.8683
Cost Savings	-6%	-7%	31%	20%	29%	26%	33%

Conclusions

- **Cost Savings** of up to **33%** predicted with best model (Blended model generated in R)
- Our results **confirm** that **maximizing the negative predicted values** is **key to saving money** for the utility company.
- Each model has its own strengths – no one model consistently outperforms all the others.
- DataRobot is certainly quicker and easier than running in R but the results are close!



Next Steps

- Profit thresholds tailored to each algorithm
- Additional feature engineering
- Run Stepwise AIC (currently taking too long to complete)
- Profiling of customers– get more demographic information on those likely to commit fraud in order to create a segment profile
- Access the original data (before the suspicious profiles were selected) to test and further improve our model
- As fraud is more expensive than inspection, the company may want to embark on a program of random inspections in order to create a better fraud detecting model





Thank You

Results: Comparison of the Models

Model	Logistic Regression	Random Forest	XGBoost	Blended Model
Sensitivity	0.7822	0.6874	0.6561	0.7179
Specificity	0.2524	0.9580	0.9766	0.9412
Pos Pred Value	0.3366	0.8880	0.9314	0.8556
Neg Pred Value	0.7050	0.8634	0.8541	0.8731
Cost Savings	-6.0%	31.0%	29.0%	33.0%

The Blended Model gives