DON'T GET KICKED – MACHINE LEARNING PREDICTIONS FOR CAR BUYING

Albert Ho, Robert Romano, Xin Alice Wu – Department of Mechanical Engineering, Stanford University – CS229: Machine Learning

Introduction

When you go to an auto dealership with the intent to buy a used car, you want a good selection to choose from. Auto dealerships purchase their used cars through auto auctions and they want the same things: to buy as many cars as they can in the best condition possible. Our task was to use machine learning to help auto dealerships avoid bad car purchases, called "kicked cars", at auto auctions.

Data Preprocessing/Visualization

Data Characteristics

All of our data was obtained from the Kaggle.com challenge "Don't Get Kicked" hosted by CARVANA. It could be described as follows:

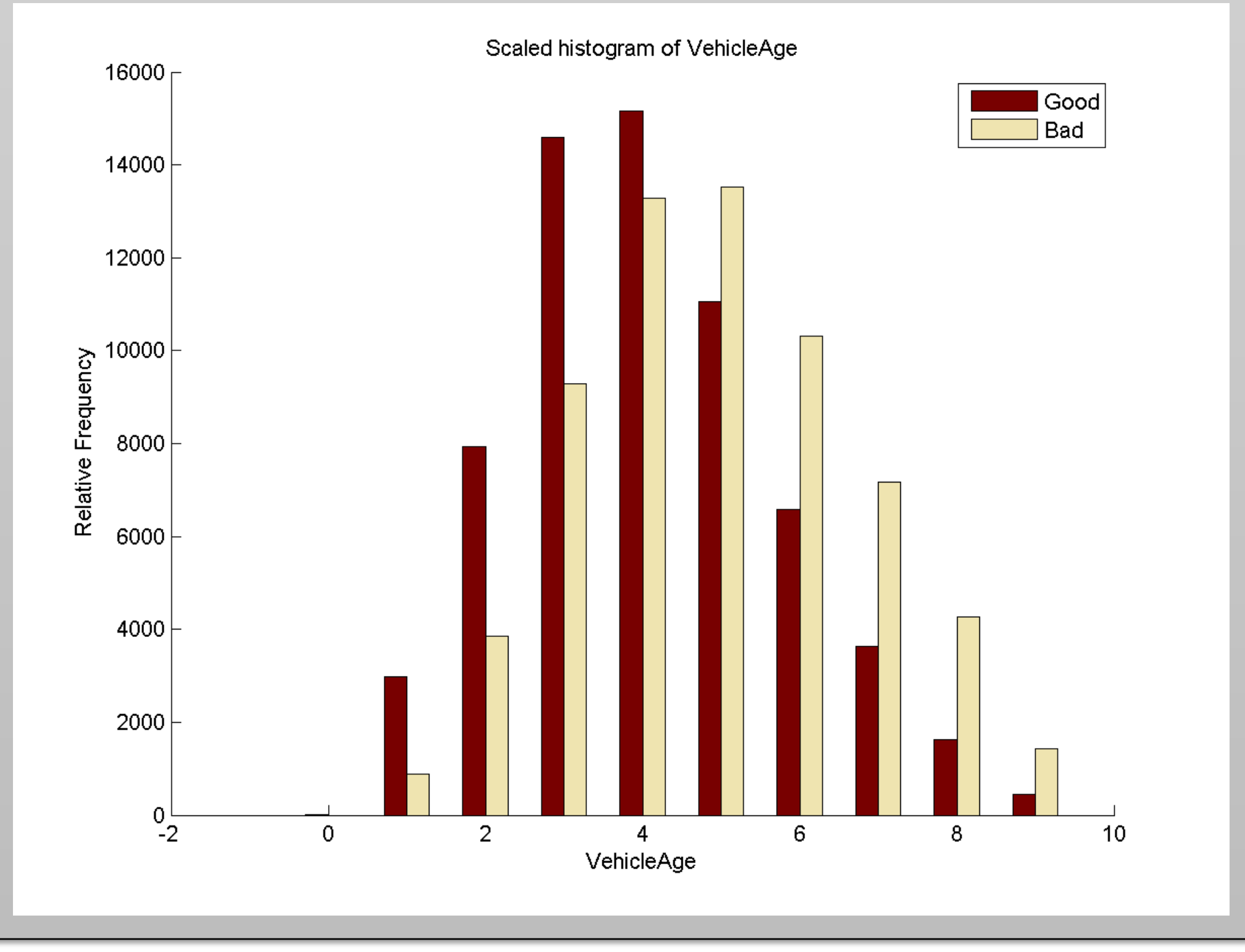
1) Contained 32 features and 73041 samples
2) Contained binary, nominal, and numeric data
3) Good cars were heavily overrepresented, constituting 87.7% of our entire data set
4) Data was highly inseparable/overlapping

Preprocessing

The steps we took to preprocess our data changed throughout the project as follows:

1) Converting nominal data to numeric and filling in missing data fields
2) Normalizing numeric data from 0 to 1
3) Balancing the data

Visualization



Algorithm Selection

MATLAB

Our initial attempts to analyze the data occurred primarily in MATLAB. Because the data was categorized into two labels, good or bad car purchases, we used logistic regression and libLINEAR[1] v.1.92. Initial attempts at classification went poorly due to heavy overlap between our good and bad training sets. We decided to follow a different approach based on the concept of boosting, which combines various weak classifiers to create a strong classifier[3].

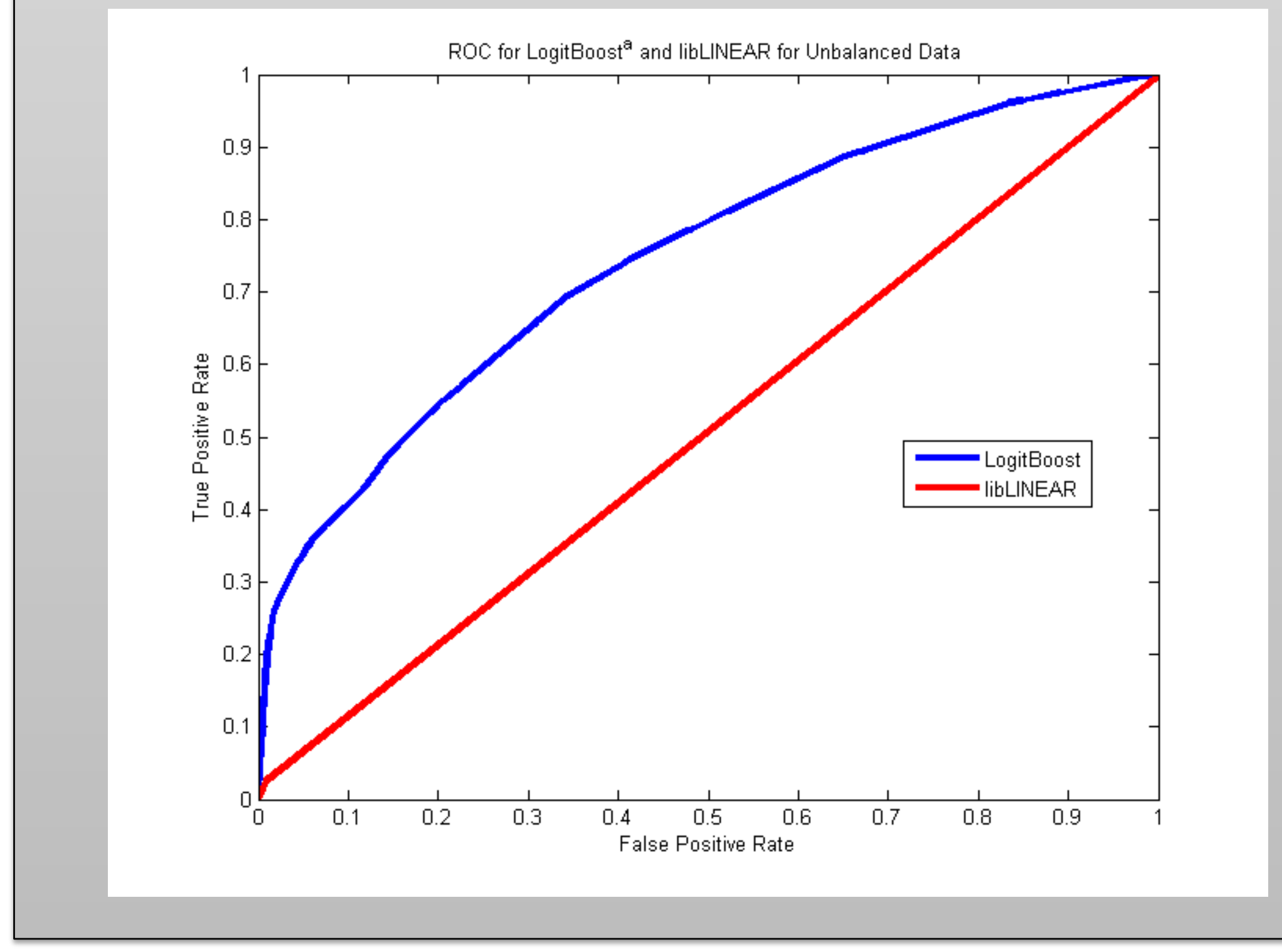
Weka

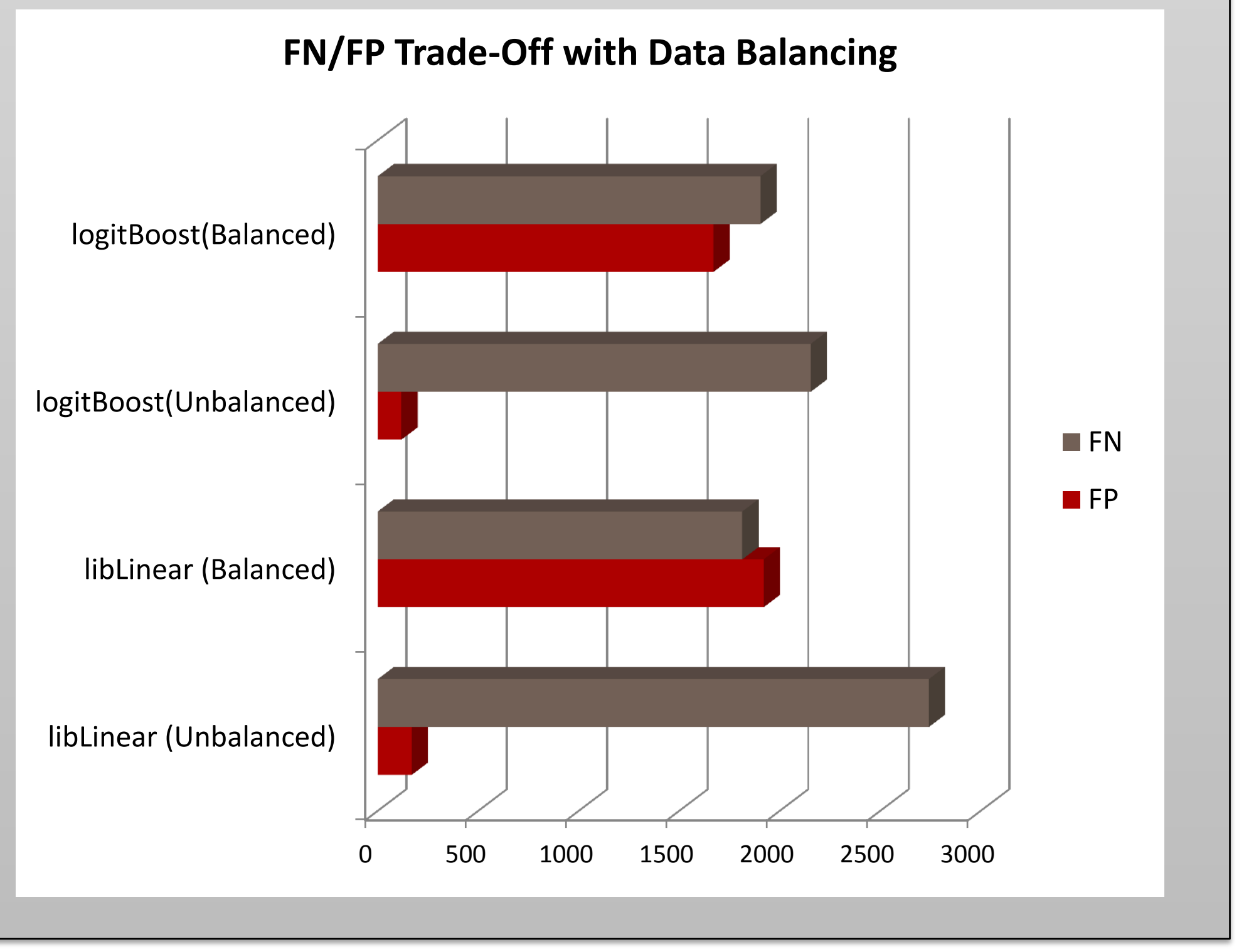
To use boosting algorithms, we used the software package called Weka[2] v. 3.7.7. Using Weka, we could apply libLINEAR and naïve bayes along with a slew of boosting algorithms such as adaBoostM1, logitBoost, and ensemble selection.

Performance Evaluation

Algorithms	Performance on Unbalanced Training Set			Performance on Balanced Training Set		
	Correctly Classified Instances (%)	AUC	F1 Score	Correctly Classified Instances (%)	AUC	F1 Score
naïve Bayes	89.41	0.746	0.351	66.46	0.745	0.332
libLinear	87.33	0.509	0.050	25.72	0.548	0.236
logistic	82.84	0.708	0.350	83.81	0.713	0.347
logitBoost[a]	89.41	0.746	0.351	66.46	0.745	0.332
logitBoost[b]	89.55	0.757	0.364	73.37	0.759	0.365
logitBoost[c]	90.11	0.758	0.368	84.45	0.686	0.338
adaBoostM1[a]	89.51	0.724	0.370	63.21	0.719	0.316
ensemble[e]	90.12	0.691	0.359	81.47	0.650	0.327
ensemble[d,e]	89.88	0.730	0.358	83.75	0.694	0.350

a. Decision Stump, b. Decision Stump 100 Iterations, c. Decision Table, d. J48 Decision Tree, e. Maximize for ROC





Discussion

Performance Metric

Initially, we evaluated the success of our algorithms based on correctly classified instances(%), but soon realized that even the null hypothesis could achieve 87.7%. We then switched our metrics to AUC, a generally accepted metric for classification performance, and F1, which accounts for the tradeoff between precision/recall. FN and FP may be more important metrics in application because has a direct impact on profit and loss for a car dealership, as illustrated below:

*Total Profit = TN*Gross Profit + FN*Loss*
*Opportunity Cost = FP*Gross Profit*

Final Result

Based on metrics of AUC and F1, LogitBoost did the best for both balanced and unbalanced data sets.

Future Work

1) Evaluate models on separated data
2) Run RUSBoost, which improves classification performance when training data is skewed
3) Purchase server farms on which to run Weka

Acknowledgement

We would like to thank Professor Andrew Ng and the TAs all their help on this project, Kaggle and CARVANA for providing the data set.

References

[1] R.-E. Fan, *et al*. LIBLINEAR: A Library for Large Linear Classification, Journal of Machine Learning Research 9(2008), 1871-1874. Software available at http://www.csie.ntu.edu.tw/~cjlin/liblinear
[2] Mark Hall, *et al*. (2009); The WEKA Data Mining Software: An Update; SIGKDD Explorations, Volume 11, Issue 1.
[3] Friedman, Jerome, *et al*."Additive logistic regression: a statistical view of boosting". *The annals of statistics* 28.2 (2000): 337-407.