Predicting Shoppers Loyalty Trough Transaction Behavior

Jingwen Sang, Shivayogi Biradar, Matthew A. Lanham

Purdue University Krannert School of Management

sangj@purdue.edu; sbiradar@purdue.edu; lanhamm@purdue.edu

Abstract

Coupon offering is one of the traditional and prevalent sales tools to both attract potential customers and increase the satisfaction of existing customers. With enough purchase history, it is possible to predict which shoppers, when presented an offer, will buy a new item. However, identifying the shopper who will become a loyal buyer prior to the initial purchase is a more challenging task.

Acquired Valued Shoppers Data asks participants to predict which shoppers are most likely to repeat purchase. The challenge provides almost **350 million rows** of completely anonymized transactional data from over **300,000 shoppers**. It is one of the largest problems run on Kaggle to date.

During the course of this project we worked on various machine learning libraries such as H2o, Xgboost and Vowpal Wabbit to optimize our solution and move up on the leaderboard of this data challenge.

Our final solution is in top **10%** of this data challenge.

Introduction

The **Acquire Valued Shoppers Challenge** on Kaggle, which asks participants to predict which shoppers are most likely to repeat purchase. To aid with algorithmic development, we have been provided the complete, basket-level, pre-offer shopping history for a large set of shoppers who were targeted for an acquisition campaign. The incentive offered to each shopper and their post-incentive behavior was also provided.

transactions.csv - contains transaction history for all customers for a period of at least 1 year prior to their offered incentive → ~350 Million Rows ~21Gb

trainHistory.csv - contains the incentive offered to each customer and information about the behavioral response to the offer → ~160057 Rows ~1Mb

testHistory.csv - contains the incentive offered to each customer but does not include their response (you are predicting the repeater column for each id in this file) → ~151844 Rows ~ 1Mb

offers.csv - contains information about the offers → ~37 Rows ~431 Bytes

Data

Dataset used in this study was obtained from "Acquire Valued Shoppers Competition" on Kaggle Variables contained in the dataset are shown in the following table:

HISTORY	
id	A unique is representing a customer
Chain	An integer representing a store chain
Offer	An id representing a certain offer
Market	An id representing a geographical region
Repeattrips	The number of times the customer made a repeat purchase
Repeater	A Boolean, equal to repeattrips > 0
Offerdate	The date a customer received the offer
TRANSACTIONS	
Id	A unique is representing a customer
Chain	An integer representing a store chain
Dept.	An aggregated grouping of the category
Category	The product category
Company	An id of the company that sells the item
Brand	An id of the brand to which the item belongs
Date	The date of purchase
Productsize	The amount of the product purchase
Productmeasure	The units of the product purchase
Purchasequantity	The number of units purchased
Purchaseamount	The dollar amount of the purchase
OFFERS	
Offer	An id representing a certain offer
Category	The product category
Quantity	The number of units one must purchase to get the discount
Company	An id of the company that sells the item
Offervalue	The dollar value of the offer
Brand	An id of the brand to which the item belongs

Feature Engineering

"Offer", "Company", "Category and "Brand" are the three most important variables in *transactions*.

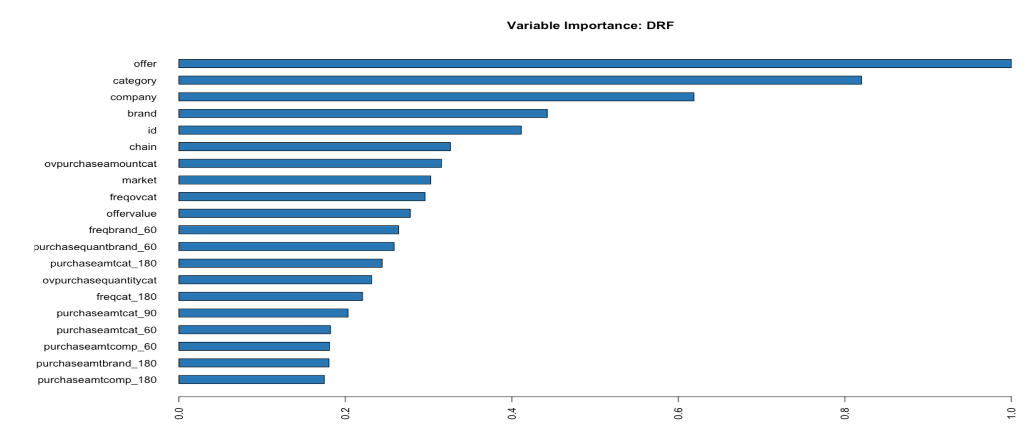
In order to gain a clearer view of the noisy data, we further aggregated the data on several periods prior to offer issuing date. Take company as an example:

SECONDARY FEATURE(COMPANY)	
Xc	The number of times a shopper has bought from the company on offer
Xc(a)	The total amount a shopper has bought from the company on offer
Xc(q)	The quantity of items a shopper has bought from the company in offer
Xc(30)	The number of times a shopper has bought from the company on offer 30 days before the coupon was offered
Xc(60)	The number of times a shopper has bought from the company on offer 60 days before the coupon was offered
Xc(90)	The number of times a shopper has bought from the company on offer 90 days before the coupon was offered
Xc(180)	The number of times a shopper has bought from the company on offer 180 days before the coupon was offered
Xc(n)	a negative feature indicating a shopper has never bought from the company before

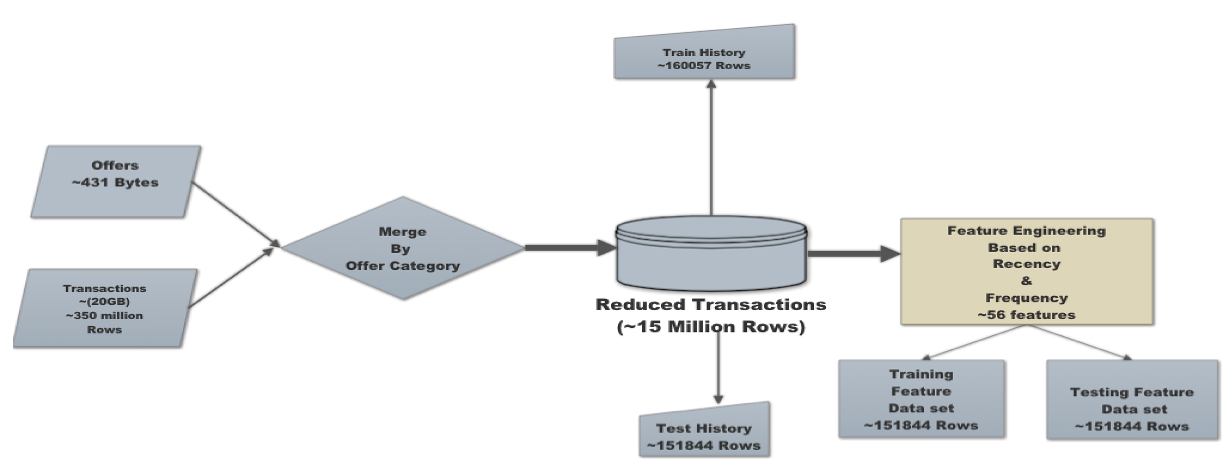
The above aggregation method was employed on company, category and brand and 45 features were obtained which combined with original features make upto 56 features in total.

N	Company	N	Category	N	Brand	N	
1	Xc	16	Xca	31	Xb	46	Id
2	Xc(a)	17	Xca(a)	32	Xb(a)	47	Chain
3	Xc(q)	18	Xca(q)	33	Xb(q)	48	Dept.
4	Xc(30)	19	Xca(30)	34	Xb(30)	49	Category
5	Xc(a, 30)	20	Xca(a, 30)	35	Xb(a, 30)	50	Company
6	Xc(q, 30)	21	Xca(q, 30)	36	Xb(q, 30)	51	Brand
7	Xc(60)	22	Xca(60)	37	Xb(60)	52	Date
8	Xc(a, 60)	23	Xca(a, 60)	38	Xb(a, 60)	53	Productsize
9	Xc(q, 60)	24	Xca(q, 60)	39	Xb(q, 60)	54	Productmeasure
10	Xc(90)	25	Xca(90)	40	Xb(90)	55	Purchasequantity
11	Xc(a, 90)	26	Xca(a, 90)	41	Xb(a, 90)	56	Purchaseamount
12	Xc(q, 90)	27	Xca(q, 90)	42	Xb(q, 90)		
13	Xc(180)	28	Xca(180)	43	Xb(180)		
14	Xc(a, 180)	29	Xca(a, 180)	44	Xb(a, 180)		
15	Xc(q, 180)	30	Xca(q, 180)	45	Xb(q, 180)		

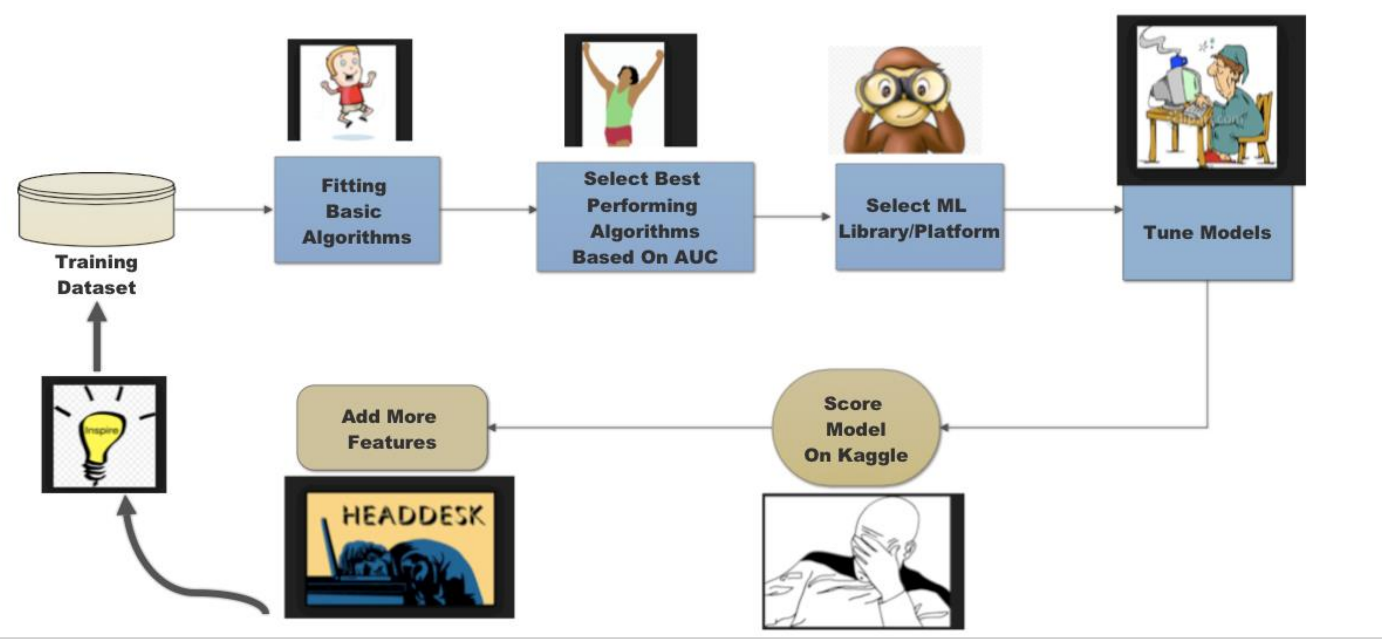
Importance rating for the top 20 variables based on Random Forest:



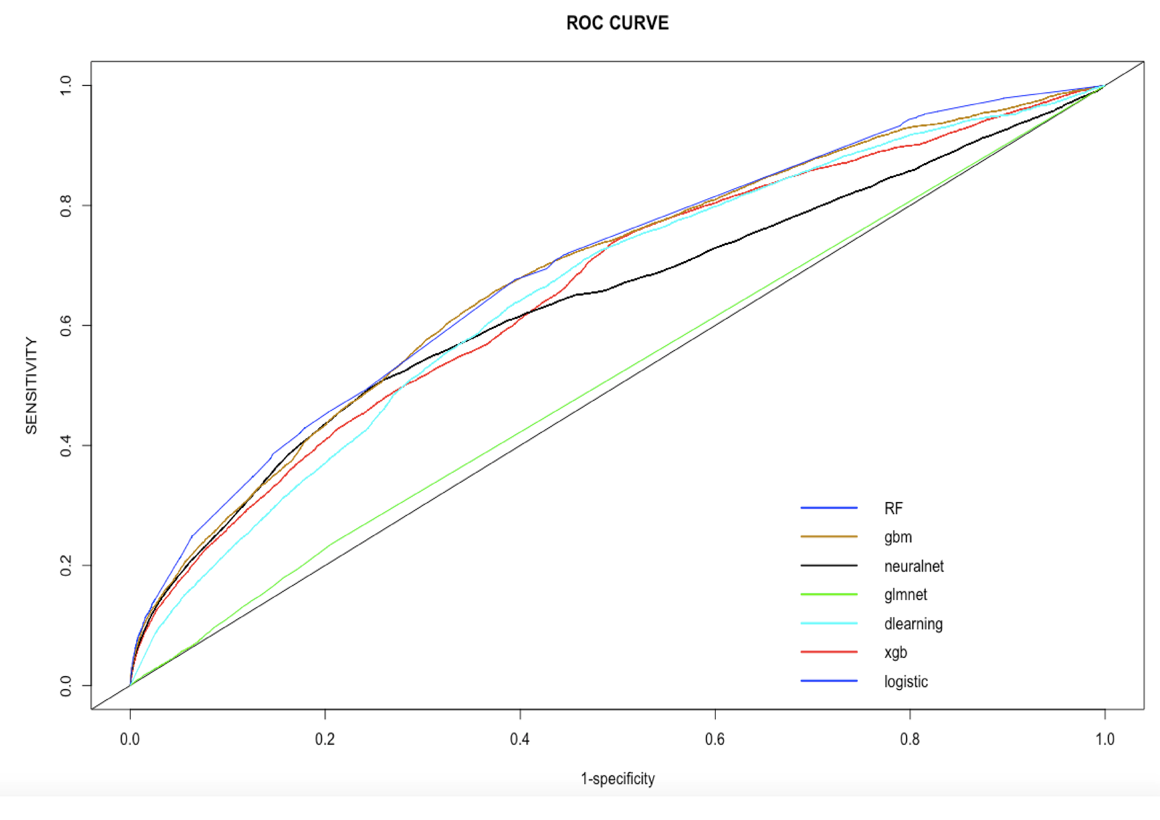
Methodology



Modelling Process



Models & Tuning Parameters



Basic algorithms were fitted on the given dataset and the top 4 algorithm were selected to be tuned for further improvement in the model scored. Based on the algorithm a suitable ML platform/library was selected to optimize tuning, performance and computing time.

Model	Platform	Number of models build	Computation Time
Random Forest	H2o in R	100	~ 45 min
GBM	H20 in r	175	~ 80 min
XGboost	Python	70	~ 40 min
Logistic Regression	Vowpal Wabbit	1	~ 9 sec
Quantile Regression	Vowpal Wabbit	1	~ 9 sec

Models	Training AUC Standard	Training AUC Tuned	Testing AUC Standard	Testing AUC Tuned	Final Score
Random Forest	0.83	0.75	0.75	0.72	0.59697
GBM	0.75	0.78	0.72	0.75	0.59459
Xgboost	0.71342	0.7287	0.70342	0.71452	0.5849
Logistic Regression	------	------	------	--------	0.5749
Quantile Regression	-----	-----	-----	-----	0.5812

Conclusions

- Feature engineering was one of the major factors in improvement of score in this competition.
- A special emphasis was placed on model tuning and based on model a suitable platform/library was selected to tune the algorithm.
- Worked efficiently with datasets larger than the memory of computer and employed techniques for economical usage of Cpu.
- Achieved a score in top 10 % of the competition.

This model could help business to design coupon offering more efficiently. In reality, business could use the algorithm included in the study to know better of their customers based on repurchase possibility and apply specific marketing strategies accordingly.

Acknowledgement

The poster is partially funded by Business Information and Analytics Center (BIAC), Purdue University. We really appreciate their support and the opportunity that they provided.