2/21/16

Predicting Customer Churn

Tony Caballero

AGENDA

- Problem
- Data
- Hypothesis
- Model
- Results
- Next Steps

Problem

- Objective:
    - Leverage the data warehouse to generate insight into customer cohorts and serve as the foundation for a predictive model to predict churn

- Problem Statement:
- Churn is difficult to predict. Doing so (accurately) on a monthly basis for the next year will guide Optimizely’s efforts as well as guide how we invest our resources to service segments where we find favorability

Data

- So far, I’ve collected over 40 variables in the customer data cube
  - Some variables are time series (MRR, Traffic, # of logins)
  - others are static variables (industry, segment, country, region, AE, etc)
  - Type of data = Salesforce data, Optimizely product data, finance data
- 200,000 records generated via SQL query
- Simplified the data set by excluding customers who’ve never paid $300 in MRR since they have a different behavior profile
- Cube consists of 5 years’ worth of data – the integrity of the data improves over time as well as amount of data

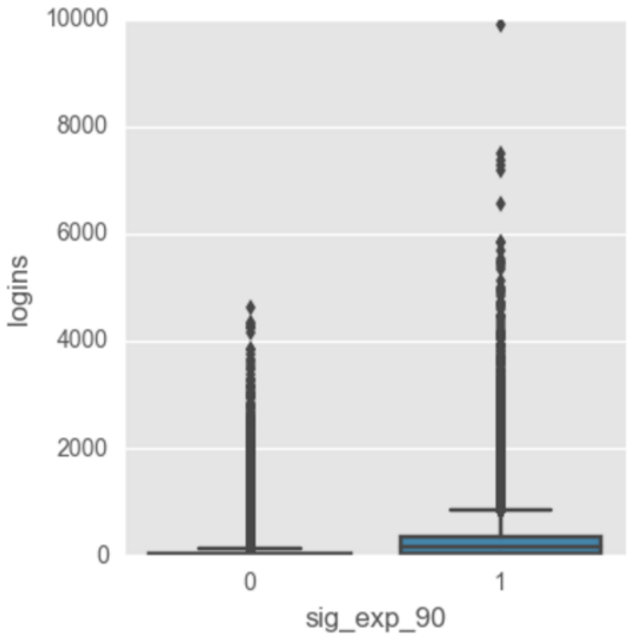
Hypothesis

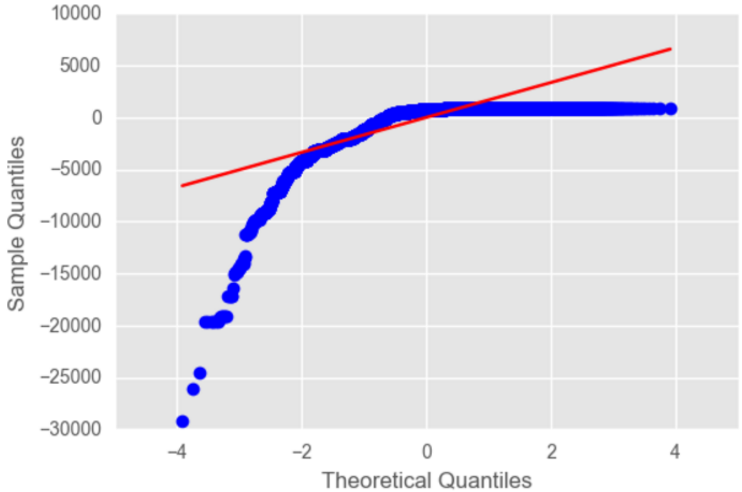
- At least one of the variables that I caputred will be a significant linear predictor of churn

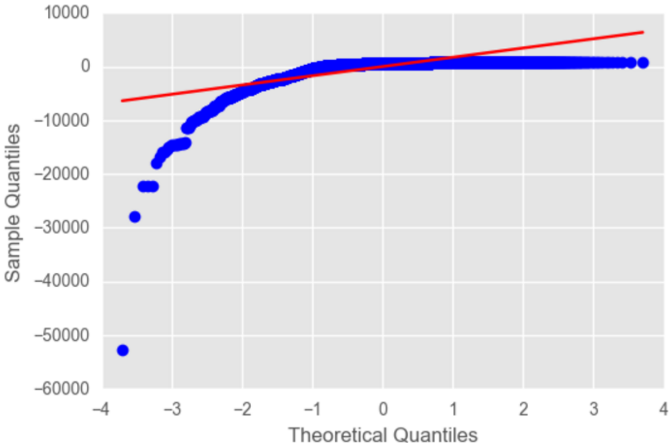
Approach

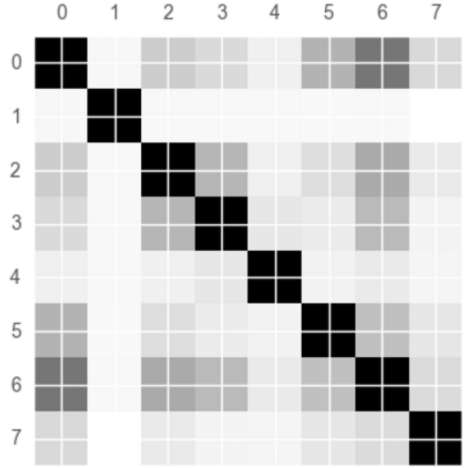
- Start with a tiny version of the data set to get a sense of which variables were interesting, then scale out to full data set
- This approach showed that my time series and product usage data yielded more interesting results
- My statis/attribute variables were just too general to yield significant results

Data Visualization









Data

	experiments_started	allocation	utilization	impressions	impression_revenue	logins	sig
experiments_started	1.000000	-0.023898	-0.001045	0.138018	0.063188	0.337929	0.1
allocation	-0.023898	1.000000	-0.000452	-0.008128	-0.004573	-0.036485	-0.
utilization	-0.001045	-0.000452	1.000000	-0.000317	-0.000160	-0.001558	-0.
impressions	0.138018	-0.008128	-0.000317	1.000000	0.120721	0.124951	0.0
impression_revenue	0.063188	-0.004573	-0.000160	0.120721	1.000000	0.066670	0.0
logins	0.337929	-0.036485	-0.001558	0.124951	0.066670	1.000000	0.3
sig_exp_90	0.152299	-0.053734	-0.002158	0.079918	0.045594	0.307443	1.0
running_experiment_days	0.529537	-0.027774	-0.001258	0.310563	0.112141	0.406427	0.2
retention	0.162483	-0.162396	0.135242	0.033289	0.025226	0.175118	0.1

	experiments_started	allocation	utilization	impressions	impression_revenue	logins	running_experiment_days
experiments_started	1.000000	0.005515	0.259833	0.193187	0.067567	0.365765	0.586635
allocation	0.005515	1.000000	-0.006589	-0.004437	-0.002307	-0.009626	-0.001938
utilization	0.259833	-0.006589	1.000000	0.356421	0.061814	0.170038	0.398714
impressions	0.193187	-0.004437	0.356421	1.000000	0.118813	0.091926	0.345466
impression_revenue	0.067567	-0.002307	0.061814	0.118813	1.000000	0.050293	0.106478
logins	0.365765	-0.009626	0.170038	0.091926	0.050293	1.000000	0.320027
running_experiment_days	0.586635	-0.001938	0.398714	0.345466	0.106478	0.320027	1.000000
retention	0.198656	-0.063559	0.105689	0.029071	0.027420	0.120760	0.188006

Model

- I decided to focus my attention on Logistic Regression due to how non-normal the data was when I attempted linear regression
- I created a binary flag indicating if the customer churned in the next month or not, which also got rid of many of my NA values
- Given my previous analysis I had already done with covariance and graphing the data, I could tell sig_exp_90 would be worth splitting data on

Results

When Sig_Exp_90 == 0:

- `training misclassification = 0.253`
- `testing misclassification = 0.255`

When Sig_Exp_90 == 1:

- `training misclassification = 0.119079071523`
- `testing misclassification = 0.113250283126`

Linear regression doesn’t seem like the way to go. More useful to use Linear Regression.

Challenges:

- Data has time series element to it.

Next Steps

- Do some time series analysis across variables that have time components. It would be good to incorporate this time series dimension into regression tree so that I can rank each variables significance and see how this ranking changes over time
- If using Time Series, it'd be good to split customers into segments

- Another useful next step is to be able to predict how many dollars we will churn in the future