Predicting Customer Revenue Using Ensemble Methods

Introduction

The constant need to generate revenue has online retailers looking for ways in addition to web analytics, to identify actionable, operational changes and better use of marketing budgets. For those companies using Google Analytics, this is no exception. Competing against the likes of Amazon and Yahoo, the ability to associate Google Merchandise Store (GStore) customers to a revenue stream and predict that customer's revenue accurately, is paramount to demonstrating how companies can use machine learning techniques to increase online presence, appropriate marketing dollars, maximize revenue, retain and attract new visitors and consumers.

The goal of this project is to accurately predict, as best as possible, real world customer revenue. The solution used two of the best performing regression tree-based ensemble methods [1]:

- Random Forest (RF) or random decision tree forest – a learning method which operates by constructing a multitude of decision trees and outputting mean prediction of the individual trees.
- eXtreme gradient boosting (XGBoost) [2] - an implementation of gradient boosted decision trees designed for speed and performance. It builds on the concept of the random forest algorithm.

This project answered the question of which model was the most accurate to use and what features were most important to track for optimum prediction results. Since the data is real world data, some features of the data have been masked. The details are outlined in the dataset section. This project evaluated model accuracy by comparing their Root Mean Square Error (RMSE)[3]. RMSE being an absolute measure of fit, the lower the value, the better the fit.

Root Mean Squared Error (RMSE)

Prediction results are scored on the root mean squared error. RMSE is defined as:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2},$$

where y hat is the natural log of the predicted revenue for a customer and y is the natural log of the actual summed revenue value plus one.

Literature Review

Online retailers are using web analytic services to track visitors on their websites and increasingly, web analytics services are now adding machine learning to their toolbox.
GStore uses the real world data collected by tracking site visitors and consumers "to understand the present consumer behavior and anticipate it for the future", Zara et al., "Using Analytics For Understanding the Consumer Online" [4]. One of the most common benefits of applying machine learning resources to web analytics is generating comprehensive information used to keep complete business

picture in terms of revenue by customer category, revenue by channel, loyalty and other important measurements.

Ensemble techniques allow flexibility and assurance that the best possible scenarios have been considered. They have been widely proven to improve predictive accuracy in many applications. In the article, "Ensemble methods for uplift modeling", Soltys et al. [5] analyzes ensemble methods, stating "The gains are much larger than those achieved in case of standard classification". Random Forest (RF) and XGBoost tree base methods are used for this project because of all the above-mentioned attributes. 5-fold cross validation, and parameter tuning were used to improve RMSE model scores for better prediction. While the best solutions for revenue predictions involve ensemble machine learning techniques, feature importance plays a vital role in the performance of the model. Prof. Pedro Domingos from the University of Washington, in his paper titled, "A Few Useful Things to Know about Machine Learning" [6] discusses the importance of feature engineering, an essential part of building an intelligent system.

"At the end of the day, some machine learning projects succeed and some fail. What makes the difference? Easily the most important factor is the features used."

— Prof. Pedro Domingos

Dataset

The sample datasets comes from the competition sponsored by Kaggle, Google Cloud and Rstudio (https://www.kaggle.com/c/ga-customer-revenue-prediction). This mostly real-world dataset has masked and deleted columns for addressing privacy concerns. The data is a subset of the Google Analytics 360 data from the Google Merchandise Store, which include the following kinds of information:

- Traffic source data: information about where website visitors originate. This includes data about organic traffic, paid search traffic, display traffic, etc.
- Content data: information about the behavior of users on the site. This includes the URLs of pages that visitors look at, how they interact with content, etc.
- Transactional data: information about the transactions that occur on the Google Merchandise Store website.

All work for this project was conducted in Kaggle kernels using R language. R code for this project is in GitHub - https://github.com/vt101/CustomerRevenuePrediction.

The dataset of 903,653 observations (rows) was reduced to a sample size of 33,818 observations, approximately 35% of which were revenue producing observations. Each row in dataset sample represents one visit to the store and there are12 attributes, 4 of which contain JSON blobs of varying lengths.

- **fullVisitorId** - an unique identifier for each user of the Google Merchandise Store
- **channelGrouping** - the channel via which the user came to the Store
- **date** - the date on which the user visited the Store
- **device** - the specifications for the device used to access the Store
- **geoNetwork** - this section contains information about the geography of the user
- **sessionId** - an unique identifier for this visit to the store
- **socialEngagementType** - engagement type, either "Socially Engaged" or "Not Socially Engaged"
- **totals** - this section contains aggregate values across the session
- **trafficSource** - this section contains information about the Traffic Source from which the session originated
- **visitId** - an identifier for this session
- **visitNumber** - the session number for this user
- **visitStartTime** - the timestamp (POSIX).

Device, geoNetwork, totals and trafficSource contain the JSON blobs. “transactionRevenue” the feature to predict was embedded within the totals JSON blob. For prediction purposes and model accuracy evaluation, “transactionRevenue” will be converted to its natural log (log1P), matching the evaluation metric RMSE.

There are known limitations to this dataset as outlined in Google Analytics online help documentation (https://support.google.com/analytics/answer/7586738?hl=en). Information for some adWordsClickInfo (traffic source) and geoNetwork fields were removed and replaced with “Not available in demo dataset” for string/character values and “null” for INTEGER values. For this reason, the following features from the flattened JSON blobs were removed from the dataset:

- adwordsClickInfo.page
- adwordsClickInfo.criteriaParameters
- adwordsClickInfo.slot
- adwordsClickInfo.gclId
- adwordsClickInfo.adNetworkType
- adwordsClickInfo.isVideoAd

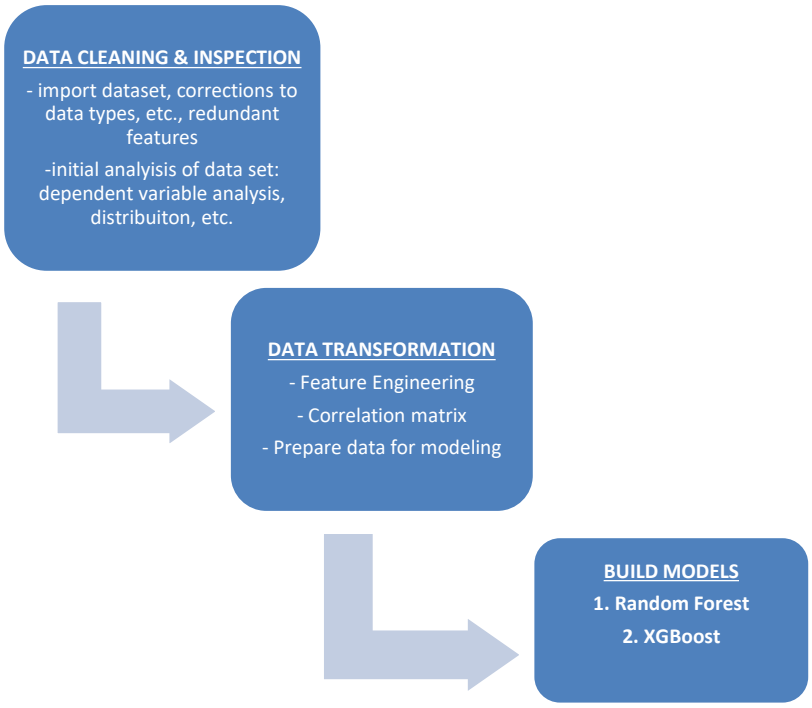
This project will not use any features missing 75% or more of their data.

Some fields were censored to remove target leakage. The major censored fields are listed below.

- hits - This row and nested fields are populated for any and all types of hits. Provides a record of all page visits.
- customDimensions - This section contains any user-level or session-level custom dimensions that are set for a session. This is a repeated field and has an entry for each dimension that is set.
- totals - Multiple sub-columns were removed from the totals field.

A check of the variance of each feature showed visits, networkLocation, latitude, longitude, cityId, screenResolution, screenColors, language, flashVersion, mobileDeviceMarketingName, mobileDeviceInfo, mobileInputSelector, mobileDeviceModel, mobileDeviceBranding, operatingSystemVersion, browserSize, browserVersion and socialEngagementType had zero variance. These features were removed from the dataset.

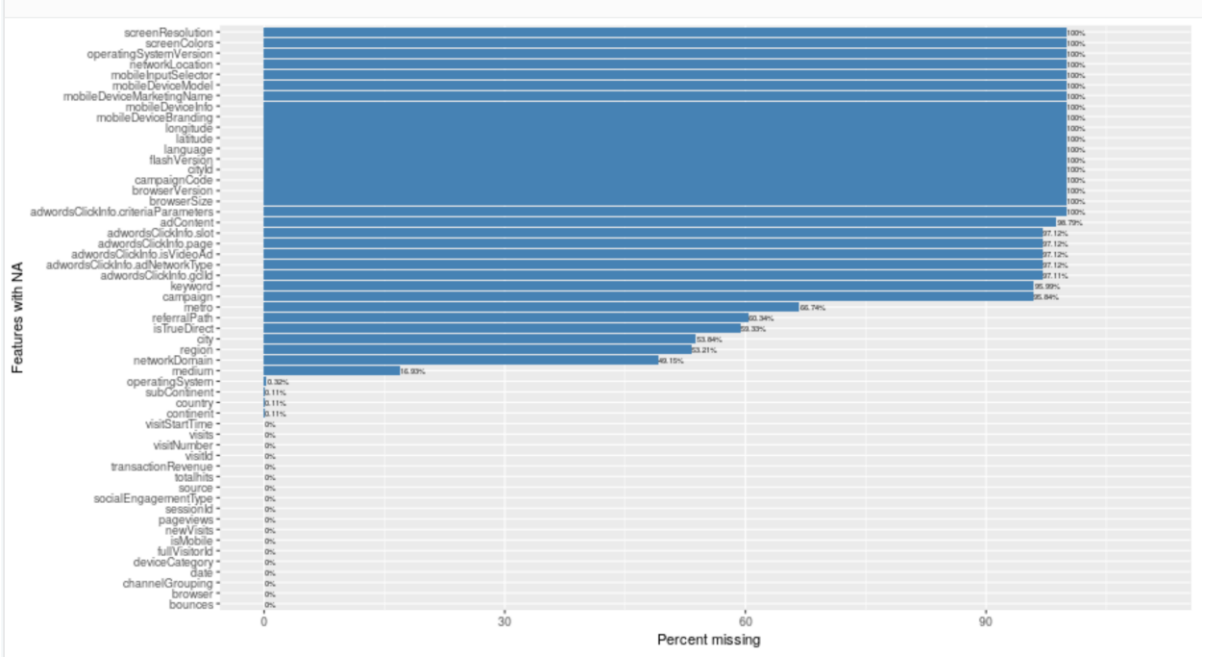
Approach



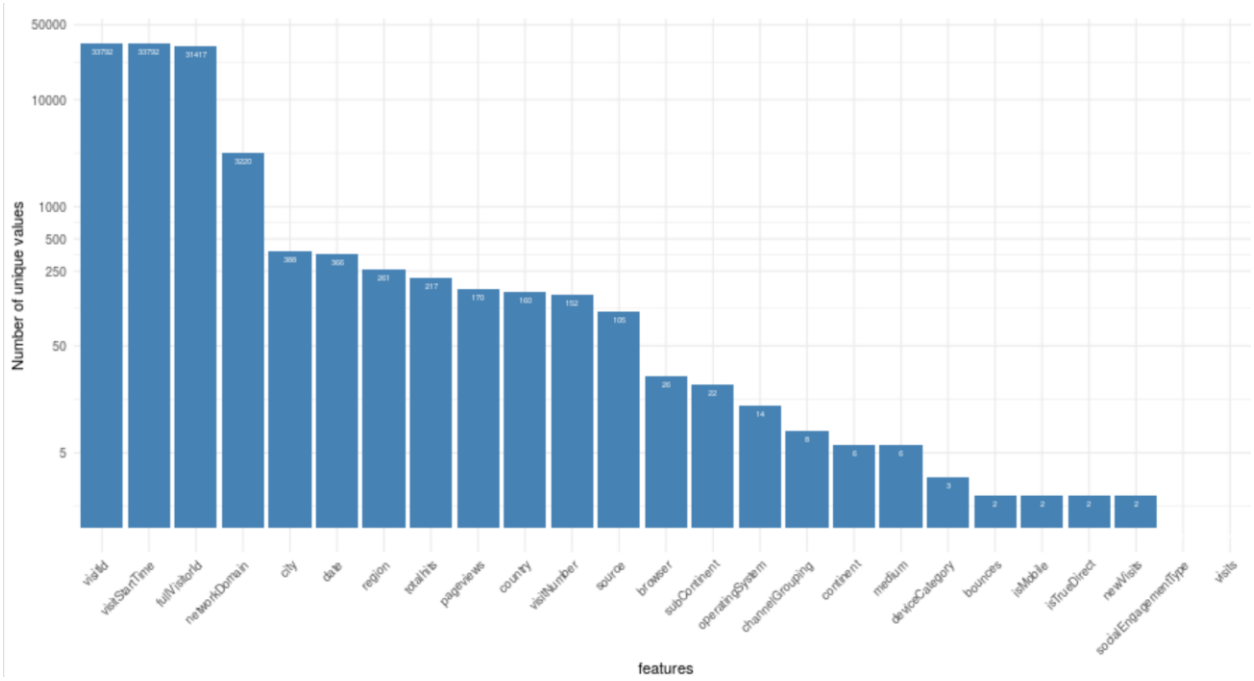
R code for this project is in GitHub - https://github.com/vt101/CustomerRevenuePrediction.

Step 1: Data cleaning and inspection

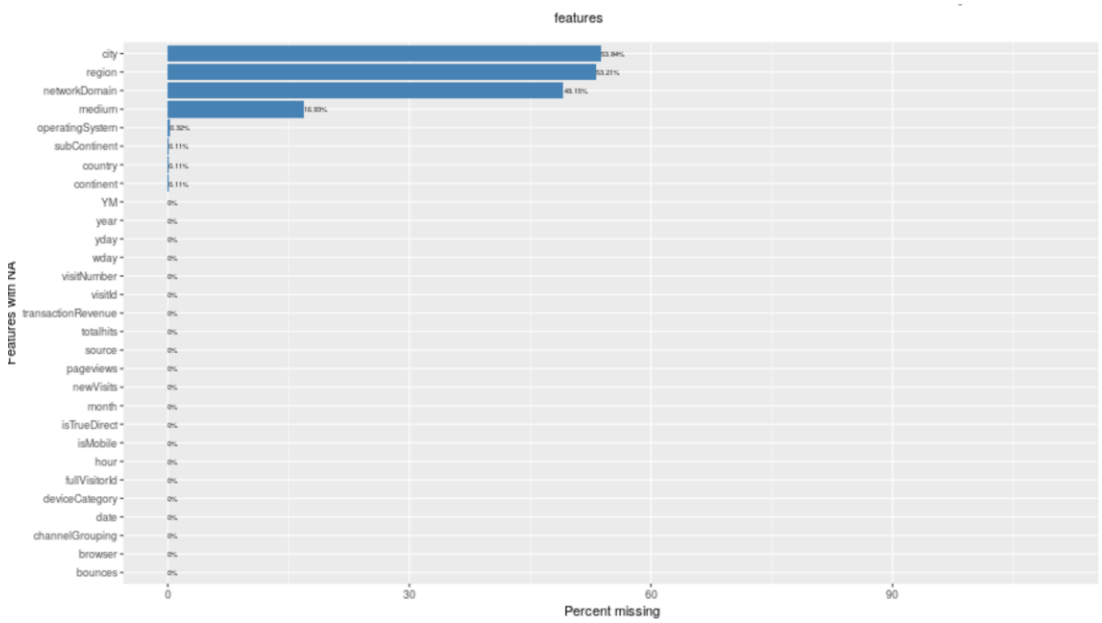
- The following values identified as not available and defined under one name (NA): 'unknown.unknown', '(not set)', 'not available in demo dataset', '(not provided)', '(none)', '<NA>'. Features with 75% or more NA values were removed from the data set.



- Features with low variance (socialEngagementType and visits) and features deemed unnecessary due to masking were removed. In summary, features with 75% of more NA values were removed.



- Data that was missing or blank was coded as “Missing”. It was determined that they were missing for a reason. Example: country, etc. This could be redirected traffic, or incognito site visitors. Categorical missing values for continent (36), subContinent (36), country (36), operating system (109), region (17996), city (18207), networkDomain (16621) and medium (5724) range from ~ 0.011% - 0.54%.



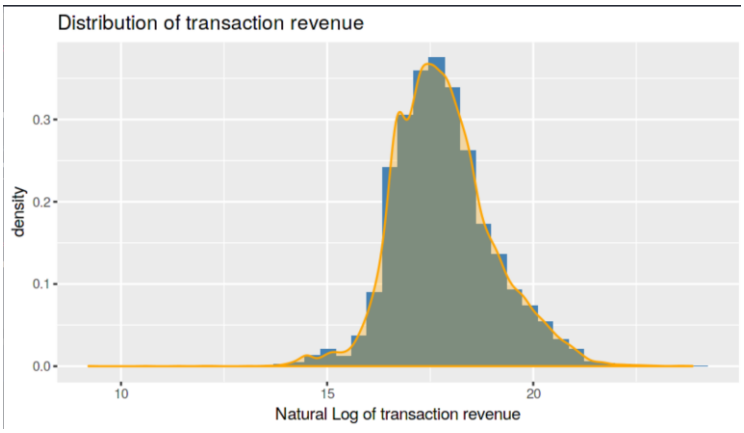
- Initial analysis showed:
    a. 33,818 observations/visits over a period of 12 months and 1 day (August 1st, 2016 to August 1st, 2017).
    b. Only 34.05% (11,515) of all observations generated revenue.
    c. There were 31,417 unique visitors and 9,996 of these visitors made a revenue transaction. Approximately 65% (21,994) of all visits were first visits.
    d. There were also 404,456 pageviews, 510,043 hits and 11,329 bounces during this period.

The table below lists visitor retention per visit. Of the 65% of first-time visitors to the site only 21% returned for another visit. The volume of these one-time only visits poses a challenge in predicting revenue for each customer to the site. As the number visits increase to the site, the accuracy of the models increases.

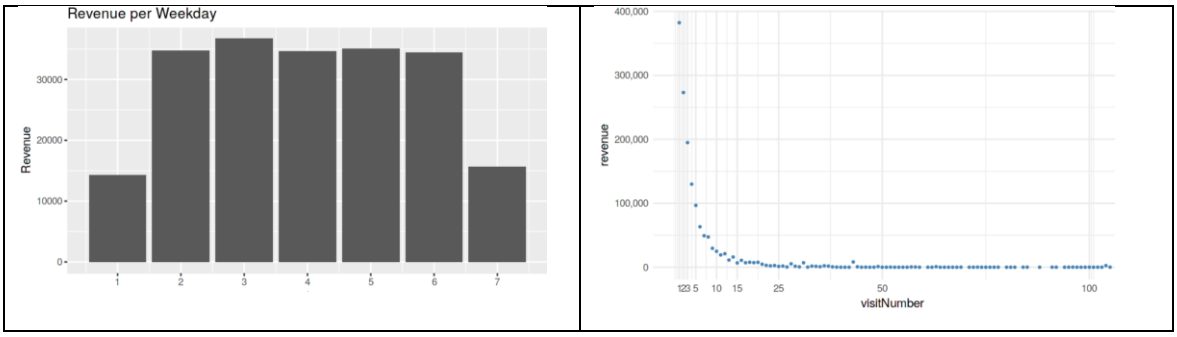
	Visits	Current_visit	Next_visit	retention
1	1 to 2	21994	4716	0.21
2	2 to 3	4716	2268	0.48
3	3 to 4	2268	1333	0.59
4	4 to 5	1333	807	0.61
5	5 to 6	807	550	0.68
6	6 to 7	550	383	0.70
7	7 to 8	383	276	0.72
8	8 to 9	276	231	0.84
9	9 to 10	231	185	0.80
10	10 to 11	185	117	0.63

The feature “transactionRevenue” in the dataset is the revenue amount generated from the visit. This dependent feature/variable is the focus of this project. Analysis of this metric showed:
~ 34% of all transactions were revenue producing transactions.
The values stored were so large they were divided by 1e+06 (1000,000) for easy reading and explanation of the metric.
The distribution of revenue is right skewed tending towards normal distribution. This is not a surprise given the one-time, high revenue transactions within the dataset.

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
0.00	0.00	0.00	45.54	25.90	23129.50



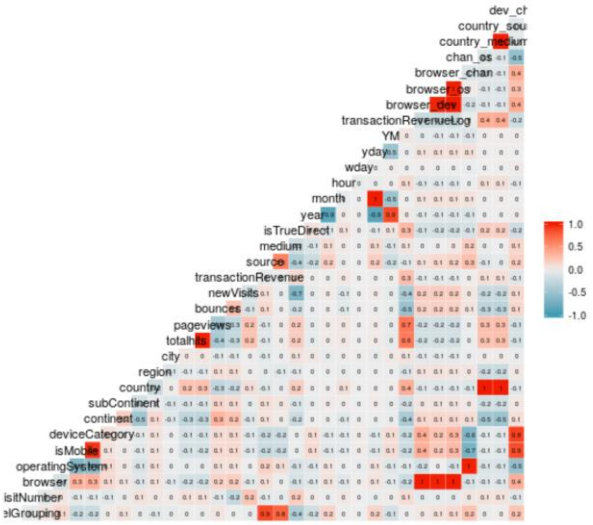
Most revenue occur during business days Monday to Friday (2-6) and the highest revenues are for visitors with five or less visits. The customer behavior of one time, extremely high-volume purchase will impact revenue prediction. The more visits the customer has to the site, the better the prediction accuracy.



Step 2: Data Transformation

- Feature engineering – the following new features were created to help identify the combination of metrics best suited for optimal prediction.
    - Year, month, hour, wday (weekday), yday (day number in the year) and YM (year month) were derived from the datetime visitStartTime variable.
    - Retention rate was created to show visitor attrition rate based on number of visits.
    - Combining/concatenating 2 categorical variables produced
        - browser_dev = str_c(browser, "_", deviceCategory)
        - browser_os = str_c(browser, "_", operatingSystem)
        - browser_chan = str_c(browser, "_", channelGrouping)
        - chan_os = str_c(operatingSystem, "_", channelGrouping)
        - country_medium = str_c(country, "_", medium)
        - country_source = str_c(country, "_", source)
        - dev_chan = str_c(deviceCategory, "_", channelGrouping)

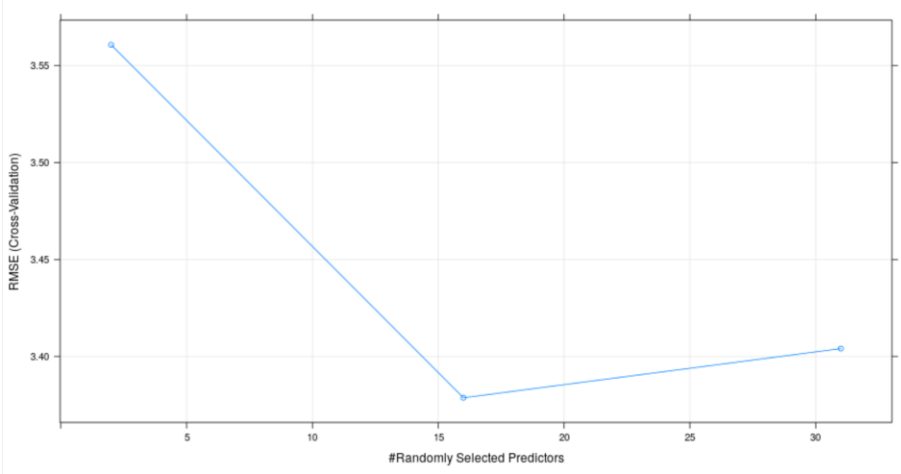
- Using a correlation matrix of the dataset features, we see that channel grouping is highly correlated to source(.87) and medium(.77); YM is highly correlated with year(.86), month(.87) and yday(.87); isMobile is highly correlated with device category(.94), etc. There were no modifications to dataset based on correlation results. The ensemble methods will address this issue.



Step 3: Build Models

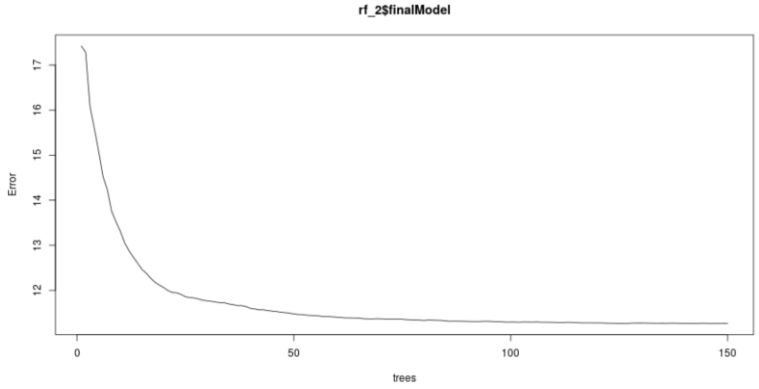
- Building RF and XGBoost models require the data be in a numerical format.
  This meant, all categorical/character/string datatype variables were first converted to factors (data structure used for fields that takes only predefined, finite number of values), then from factors to numeric values.
- The set.seed() function was used when loading the data to ensure results of the evaluation metric were repeatable predictive results.
- The dataset was split 70/30 training/test for both models. The set.seed(function was used to ensure repeatable outcomes.
- The following R libraries were used to be the models: tidyverse, randomForest, caret, Zoo, readr, jsonlite, tidyr, e1071, dplyr, lubricate, Metrics, ggplot2, scales, method and xgboost.
- Both models used grid search and k-fold cross validation. For this project, K = 5. The data is split the data into 5 partitions and run for 10 times. 1 partition is the test set and the other partitions are the training set. With each run the training and test sets are unique.
- The built-in feature importance functions within both models will be used to highlight which predictors are most useful for predicting site customer revenue.

- <u>Random Forest (RF) Model</u>
  - The base model was created with RF default values and the best value for variables available for splitting tree node (mtry) was identified by the lowest RMSE score generated. The best randomly generated mtry value was 16 with a RMSE score of 3.3786.



  - Further parameter tuning for involved generating mtry value with lower RMSE score the base model, number of trees to grow (ntree) and maximum nodes in a tree (maxnodes) to prevent overfitting.
  - The resulting optimized model had values of mtry = 8, ntree = 150, maxnodes = 203 and RMSE = 3.3628.

- The diagram below shows that the error rate is stabilized at ~150 trees.



- XGBoost
  - In addition to converting data into a numerical format, the data was converted to DMatrix (an internal data structure that used by XGBoost).much faster than the RF model.
  - Time to execute model training and prediction was
  - The following default parameters were used to create the base model:

```
param <- list(booster = "gbtree",
    objective = "reg:linear",
    eval_metric = "rmse",
    eta=0.3,
    gamma=0,
    min_child_weight=1,
    subsample=1,
    colsample_bytree=1)
```

    The RMSE score was 3.2483.

  - With parameter tuning, the best model parameters were:

```
param <- list(booster = "gbtree",
    objective = "reg:linear",
    eval_metric = "rmse",
    eta=0.01,
    nthread = 2,
    gamma=5,
    max_depth=7,
    min_child_weight=100,
    subsample=0.8,
    colsample_bytree=0.9,
    alpha = 25,
    lambda = 25)
```

  - The RMSE score was 3.2208.

Results

The XGBoost model was the faster of the two models and the better predictor of the models. The table below shows the RMSE scores and corresponding % change due model parameter tuning.

	Random Forest	XGB	% change RMSE
RMSE base model	3.3786	3.2483	-3.87%
RMSE tuned model	3.3628	3.2208	-4.22%
% change RMSE	-0.047%	-0.845%	

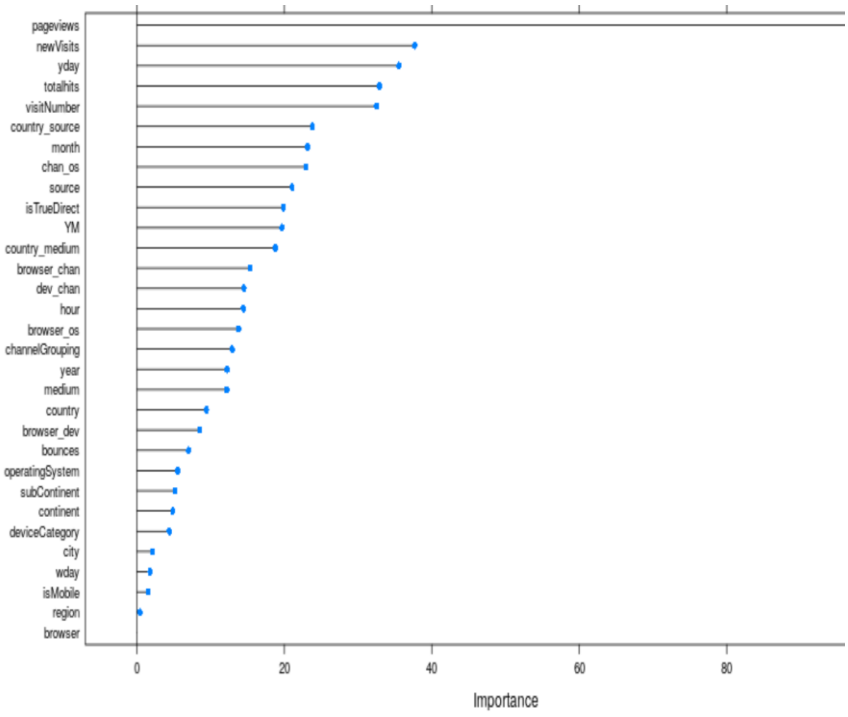
In all aspects, XGB demonstrated better performance.

The feature importance function in RF showed in order of importance, the predictors that should be tracked:

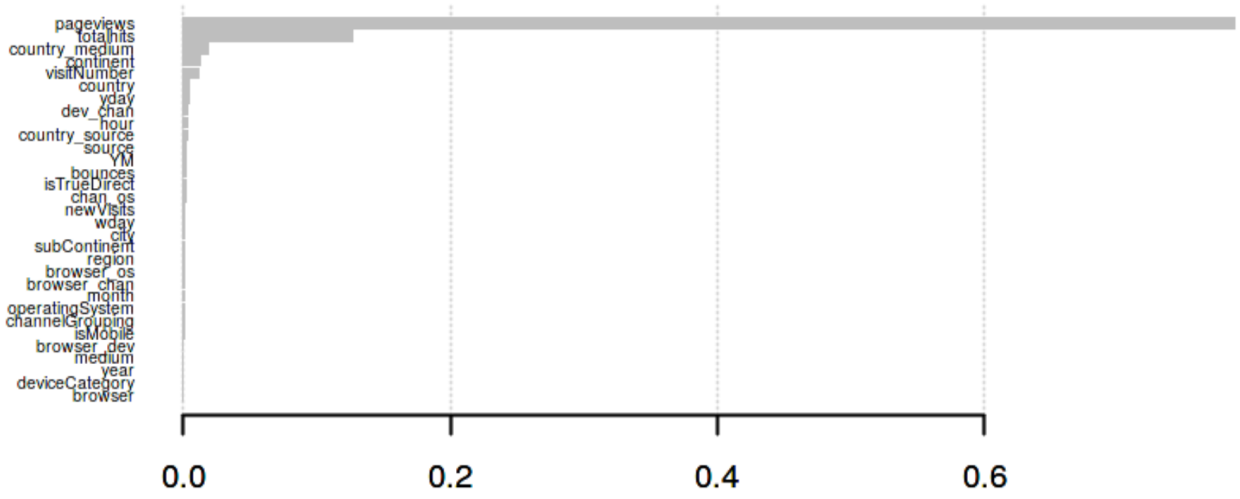
```
rf variable importance

  only 20 most important variables shown (out of

               Overall
pageviews      100.000
newVisits       37.682
yday            35.543
totalhits       32.906
visitNumber     32.499
country_source  23.809
month           23.143
chan_os         22.951
source          21.052
isTrueDirect    19.897
YM              19.679
country_medium  18.795
browser_chan    15.345
dev_chan        14.526
hour            14.439
browser_os      13.812
channelGrouping 12.938
year            12.232
medium          12.197
country          9.434
```



The feature importance function in XGB showed in order of importance, the predictors that should be tracked:



Feature	Gain	Cover	Frequency
1: pageviews	0.787221925	0.257525679	0.192774672
2: totalhits	0.126993370	0.211939365	0.157129112
3: country_medium	0.019264620	0.019971445	0.033285989
4: continent	0.013025575	0.005682721	0.004490058
5: visitNumber	0.011236716	0.075032992	0.061852836
6: country	0.004978638	0.005010851	0.007651425
7: yday	0.004510200	0.091950102	0.111358013
8: dev_chan	0.003599527	0.017811983	0.019838724
9: hour	0.003541315	0.050550932	0.074612847
10: country_source	0.003087280	0.023615846	0.027925410
11: source	0.002455600	0.021614364	0.019678365
12: YM	0.002263160	0.022549321	0.038715294
13: bounces	0.002258019	0.001271567	0.002130487
14: isTrueDirect	0.002034392	0.008192233	0.016883533
15: chan_os	0.001684950	0.025825889	0.035714286
16: newVisits	0.001610568	0.001911841	0.004100614
17: wday	0.001544374	0.025475269	0.043892605
18: city	0.001533543	0.054094573	0.041372675
19: subContinent	0.001452716	0.005101232	0.005200220
20: region	0.001231620	0.031304544	0.029849721

NOTE: Feature importance for RF and XGB models vary.

Conclusions

This project answered the following research questions:

- Which model more accurately predicts site customer revenue?
- Which predictors must be tracked to ensure the best possible accurate prediction?
- What key factors may heavily impact the accuracy of the predictions.

The eXtreme Gradient Boosting (XGB) model is fast, flexible and the better suited predictor of this real world GStore dataset. It will be possible to apply this model to other real-world e-commerce site data. The top ten most important features to track for most accurate predictions are pageviews, totalhits, country-Medium, continent, visitNumber, country, yday, dev-chan, hour and country-source.

Most revenue occurs during business days (Monday to Friday), and the highest revenues are for visitors with five or less visits. This infers transactions are Business-to-Business (B2B) transactions. One-time high revenue purchases infer yearly bulk purchases on behalf of a business. Further analysis on this subset of the data is highly recommended. It will help GStore better service and ultimately develop other methods of anticipating such revenue.

The total impact of the censored fields within the dataset (known limitation) is not fully known. Should those fields become available, the models created in this project should be re-evaluated.

References

[1] An Introduction to Ensemble Methods for Data Analysis by Berk, Richard A Sociological Methods & Research, 02/2006, Volume 34, Issue 3. https://doi.org/10.1177/0049124105283119

[2] XGBoost: A Scalable Tree Boosting System by Chen, Tianqi; Guestrin, Carlos Proceedings of the 22nd ACM SIGKDD International Conference on knowledge discovery and data mining, 08/2016 Conference Proceeding: Online Items Only

[3] Root mean square error (RMSE) or mean absolute error (MAE)? – Arguments against avoiding RMSE in the literature. Geoscientific Model Development. 2014;7(3):1247-1250 DOI 10.5194/gmd-7-1247-2014. https://doaj.org/article/6bf18970a5de4497ae44d7dd210fec3b

[4] USING ANALYTICS FOR UNDERSTANDING THE CONSUMER ONLINE. Zara, I., Velicu, B. C., Munthiu, M., & Tuta, M. (2012). USING ANALYTICS FOR UNDERSTANDING THE CONSUMER ONLINE. Paper presented at the , 18 791-796. Retrieved from http://ezproxy.lib.ryerson.ca/login?url=https://search-proquest-com.ezproxy.lib.ryerson.ca/docview/1372761032?accountid=13631

[5] A Few Useful Things to Know about Machine Learning by Domingos, Pedro Communications of the ACM, 10/2012 Magazine Article: Available Online

[6] Sołtys, M., Jaroszewicz, S. & Rzepakowski, P. Data Min Knowl Disc (2015) 29: 1531. https://doi-org.ezproxy.lib.ryerson.ca/10.1007/s10618-014-0383-9