**Team 15 - Risk Profile Prediction**

**Introduction, that provides an overview of the project and justification for your choice of this topic. What questions that you try to answer in this project?**

Our mission was to generate relevant business insights applicable to the average corporation. More specifically, our focus is to utilize data to create actionable insights which can drive sales strategy forward.

**Related work, for instance, anything that inspires you that may be some new article, a paper, a web site, a case study done by others or something that we discussed in class.**

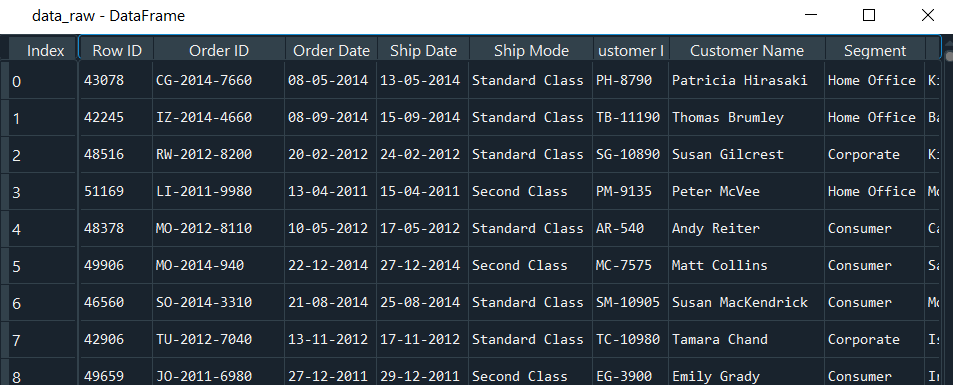
**Data that you use for this project, including details on how you get the data such as the source of the data, the scraping method and so on. You may also want to detail how you clean up the data.**

We found a dataset on Kaggle which illustrated four years (2011 - 2014) worth of transaction data from an unnamed company. Ultimately, we hoped to generate relevant and actionable insights the sales team could use to further increase next quarter’s revenue. After some exploratory data analysis (which we will explain further in the next section), our team aimed to identify current active users who were at-risk of ‘churning’.

Typically, churn is associated with users who terminate their contract and no longer use a company’s services. However, for the purposes of this analysis, we have identified any users who do not make a purchase within 2 months as individuals at-risk of churning.

To train our machine learning model, we have engineered the data to cluster our customers based on their order recency, frequency, and monetary value. In other words, we’ve performed an RFM analysis on their most recent transactions.

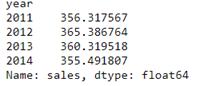
Since the data came in relatively clean, it was a simple matter of changing data types and excluding those without an adequately lengthy transaction history. Below you will see a quick snip of the raw data provided for via Kaggle.



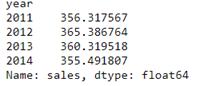
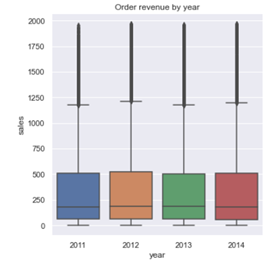
**Exploratory analysis: A description of the statistical or machine learning methods that you use in this project, using the content taught in class but you are not restricted to coursework.**

We started this project aiming to create actionable insight which could increase sales revenue. Before engineering the data, we had to ensure the data’s integrity.

After some basic statistical analysis, we realized there were various transactions which significantly skewed the data. Grouping by Order ID, we saw that while most orders averaged a few hundred dollars, there were numerous transactions which exceeded several thousand.

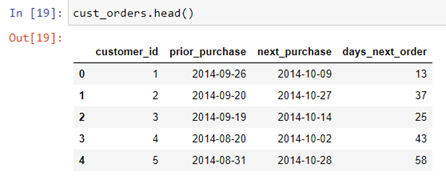


For a more representative sample, we identified the top 5% of orders as outliers and removed them from our analysis. Below is the distribution of order sale amount per order for each year. Across all years, the average order revenue is approximately $364.



We then segmented our data into two periods based on a cutoff date to start identifying customer patterns. Since we wanted to maximize our data points, from our last order date (12/31/2014) we set the cutoff date to three months prior (10/1/2014). We then identified each customer’s orders directly adjacent to this cutoff date. In a new dataframe, we stored the mentioned fields along with another attribute which marked the intervals between these dates in the attribute called ‘days\_next\_order’.

We will later reference this field when classifying our prediction variable.



Next, we began our RFM (recency, frequency, monetary value) analysis. A short description of each feature:

· Recency: Difference in days between cutoff date and last order made before said cutoff date

· Frequency: Number of orders made prior to cutoff date

· Sales: Aggregated revenue of all orders prior to cutoff date

We established each feature’s optimal number cluster by plotting each cluster’s distortion distortions (sum of squared distance between each point and cluster center). Using a Kmeans algorithm, we iterated over the outcomes of 10 cluster, each with 1000 iterations. Balancing the tradeoff between minimizing distortion and the risk of overfitting, we chose the last point with a relatively strong slope. In other words, we chose each feature based on their respective elbow plots. All three features’ elbow plot indicated 3 clusters were optimal. Using this, we identified each customer into their respective clusters. Ordering this scores in descending order, 2 being the most recent, most frequent, and highest sales, we then aggregated these scores for a more complete heuristic measure for each customer.

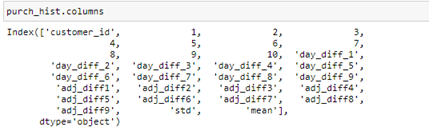
Our next goal was to identify each customer’s purchasing behavior. To achieve this, identified the 10 most recent orders (prior to cutoff date) and arranged them in descending order. In the event that there were any ‘Not A Number’ (NAN) values, we dropped that customer from our analysis. We then ran two loops to create two types of features identified as ‘day\_diff and ‘adj\_diff’. After that, we identified the standard deviation and mean between adjacent orders.

1) ‘day\_diff’: days between invoices and the invoice just prior to cutoff date

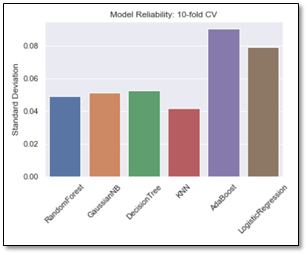
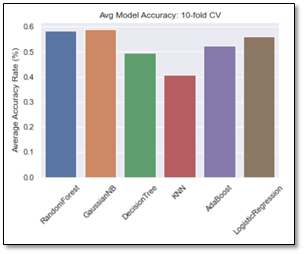
2) ’adj\_diff’: days between adjacent invoices

3) ‘std’: standard deviation between adjacent orders

4) ’mean’ average number of days between orders



We proceeded onto model selection by comparing a variety of models including: Random Forest, Gaussian Naïve Bayes, Decision Tree, K Nearest Neighbors, AdaBoost and Logistic Regression. In order to train our model and evaluate the model’s performance, we started by splitting our sample into training (80%) and testing set (20%).



After evaluating their overall accuracy and overall reliability, we decided to pursue improving 1) Random Forest, 2)Gaussian Naïve Bayes, and 3) AdaBoost. We chose Random forest and Gaussian Naïve Bayes Since they had the highest overall accuracy and relatively low standard deviation. As for our third choice, we hesitated between decision tree and AdaBoost. Ultimately, we chose to pursue AdaBoost since Random Forest had a very similar methodology to Decision tree. Should AdaBoost yield highly unreliable results, we would consider running a Decision Tree.

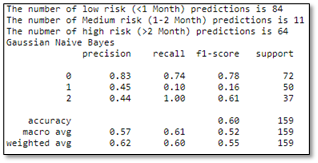
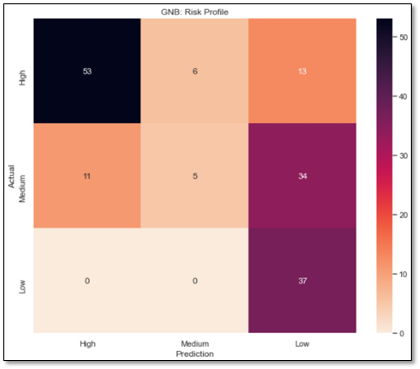
**Analyses of the data using the methods discussed above (again, you are free to bring in any relevant methods, whether covered in class or not).**

As previously mentioned, our goal was to identify individuals at risk of not many purchases within 2 months. In our model, those classified as 0 are considered customers with a ‘high-risk’ of churning.

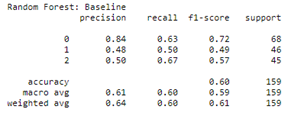
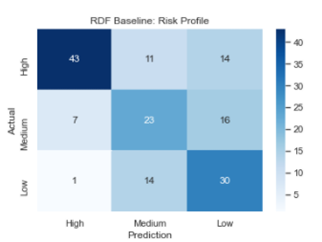
**Naïve Bayes:**

As seen above, Gaussian Naïve Bayes (GNB) has a relatively low overall accuracy of 60%. However, our target group (identified as ‘High’ risk) has a relatively accurate prediction rate. Below the confusion matrix is a classification report to better interpret the prediction result vs the test result.

The GNB model yielded a strong precision rate of 83% since 53 of its 64 predictions were in fact high-risk individuals. In terms of recall accuracy, it neglected to identify 18 individuals and this results in an accuracy of 74% (53 / 72).



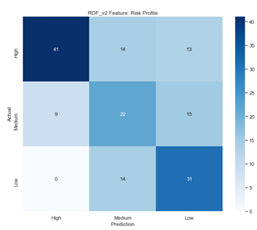
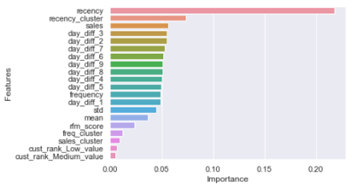
**Random Forest: Baseline**

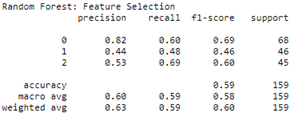


When we ran the random forest without feature selection or parameter optimization, its precision was relatively high while its recall was relatively low.

**Random Forest: Feature Selection**

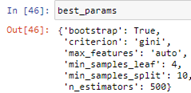
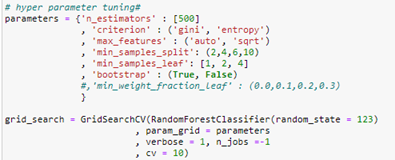
While we didn’t believe overfitting was an issue, we still ran feature selection to see whether or not it would improve the random forest’s accuracy. We used a filter method to rank each attribute’s correlation with the outcome variable. Recency (days since last purchase) had the highest correlation. Sorting feature importance in descending order, we ran another random forest model with the first features which accounted for 90% of the ‘importance’. In short, we saw no increase in accuracy. In fact, our target class (high-risk) individuals’ precision, recall scores as well as the overall accuracy actually decreased.

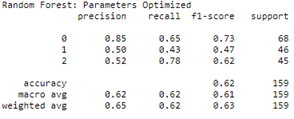
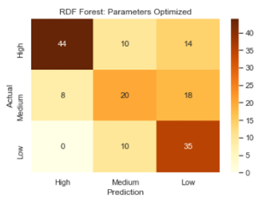




**Random Forest: (parameter optimization)**

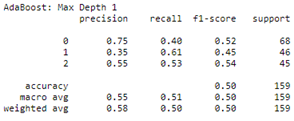
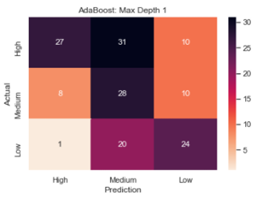
Running a 3 fold cross validation to test the optimal number of trees for our random forest, we established the optimal number was 500. Passing that into a 10 fold cross-validation grid search function seeking other parameters, we utilized the recommended optimal parameters which yielded a model whose accuracy for high-risk classifiers was relatively better than our baseline random forest model.





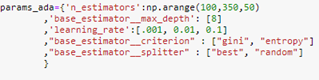
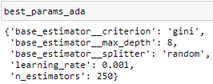
**AdaBoost: Baseline**

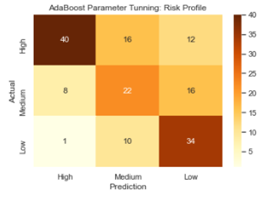
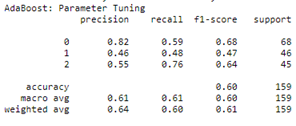
Aside from limiting the max depth of this algorithm to one, all other parameters were left on python’s default. High risk classification’s precision and recall are considerably lower than both Random Forest and Naïve Bayes.



**AdaBoost: (Parameters Optimized)**

Optimizing various parameters and running the recommended parameters yielded significant increase in high-risk precision and recall. Overall accuracy also improved significantly.





**A brief concluding section that highlights the main elements of your analysis and findings.**

Ultimately, Naïve bayes appears to be the best model for identifying high-risk customers. In terms of precision, its high-risk classifications were relatively like other algorithms. In terms of recall, it far exceeds other algorithms with a score of 75% accuracy. While we are confident in the high precision, recall can be further improved. As seen in the confusion matrix, roughly 25% of customers who will churn are not accurately classified as at risk. We believe with more time and experience; we can further increase the accuracy and reliability of these models. One such improvement would be testing other parameter ranges.

Another aspect we would consider is applying a cross validation on the entire dataset. While these figures are relatively accurate, each iteration yielded results with a variance between 2-3%. To improve the model, we should run a 10 fold cross validation over the entire dataset rather than just 1 fold over the training data.

**What can we learn from your project? What answers did you find for the questions that you proposed?**

While we are capable of reliability identifying a high number of customers at risk if churning, we can still do more. One possible improvement is to create a product recommendation engine which could generate a list of products with a high probability of enticing another purchase. We should also predict the optimal coupon rate to generate further revenue.

**Term Project Paper should be 5-10 pages in length (12 point font, 1.5 spacing) and should include a bibliography of your research (not included in the length limit of the report). You may insert figures and tables to better illustrate your findings. The paper along with the code is due in Week 11 (no class meeting in that week).**