Dr. Dicy Saylor **706-718-4724**
Data Challenge Answers **dicyann.adams@gmail.com**

Introduction

- The purpose of this challenge is to let you demonstrate the way you think and work.
- You shouldn't spend more than 8 hours to complete the exercise.
- The dataset we are providing contains the orders made by customers in one of our applications. Here's the description of each column:

  1. customer code: unique id of a customer;
  2. branch id: the branch id where this order was made;
  3. sales channel: the sales channel this order was made;
  4. seller code: seller that made this order;
  5. register date: date of the order;
  6. total price: total price of the order (sum of all items);
  7. order id: id of this order;
  8. quantity: quantity of items, given by item code, were bought;
  9. item code
  10. item total price: total price of items, i.e., quantity* price;
  11. unit price: unit price of this item;
  12. group code: which group this customer belongs;
  13. segment code: segment this client belongs;
  14. is churn: if this client is set as a churn.

Question 1 (10 Points)

- List as many use cases for the dataset as possible.

  1. Compare sales volume across branches to identify which branches perform best/worst. Try to pin point reasons for under performance. Strategize with business ops for ways to improve these branches.
  2. Identify "normal" business volume for a seller. Build an anomaly detector to alert when a a seller is performing outside the norm. Include seasonal changes by using the register date as a feature.
  3. Analysis of sellers with a high level of customer churn. Is the seller underperforming or is definition of churn too strict. A seller's velocity could be dependent on region and time of year.
  4. Create a learning algorithm that classifies customers based on churn status. Target the false positives as customers likely to churn in the near future.
  5. Identify which items are best sellers at particular branches and times of year. Offer sales based on this analysis to drive more units purchased.

Question 2 (10 Points)

- Pick one of the use cases you listed in question 1 and describe how building a statistical model based on the dataset could best be used to improve the business this data comes from.
- I will create a binary classifier for customers churn status. If the classifier misclassifies a customer having churned when they haven't, i.e. a false positive then they are at risk for churning soon. We would want to reach out to these customers to offer an incentive such as a sale to do business with us again. A service like this would not need to be run often, e.g. quarterly, and our business partners would be the sales department.

Question 3
(20 Points)

- Implement the model you described in question 2, preferably in Python. The code has to retrieve the data, train and test a statistical model, and report relevant performance criteria. Ideally, we should be able to replicate your analysis from your submitted source-code, so please explicit the versions of the tools and packages you are using.
- The python notebook I used to create the churn classifier can be found at `https://github.com/dr-adams/data challenge.`
- `Packages versions are as follows:`
  1. `python3`
  2. `sklearn-0.20.3`
  3. `pandas-0.24.2`
  4. `seaborn-0.9.0`
  5. `numpy-1.16.3`
  6. `json-2.0.9`

Question 4
(60 Points)

- Explain each and every of your design choices, you can use jupyter notebooks. (e.g., preprocessing, model selection, hyper parameters, evaluation criteria). Compare and contrast your choices with alternative methodologies. Describe how you would improve the model in Question 3 if you had more time.
- Design choices are explained in the notebook as I worked.
- A binary classifier is fairly easy to train and test once you engineer some clever features, which is why I chose this idea.
- The most difficult step is feature engineering itself. I noticed three categorical features have high cardinality (channel, seller, and item) and I decided to bin these data rather than use an encoding method. If I had more time, I would like to understand what these data represent to ensure that binning is the best method.
- I would also like to turn the binning steps into a class in order to scale with different data sets. If this step were a class I could stick it into the transform pipeline along with the simple imputer step.
- I would take more time to find the best classifier with a gridsearch, but the gradient booster performed well so I didn't explore further.