**Product Recommendation Case Study**

Gourav Verma

**Overview**

This case study describes the analysis approach for building a product recommendation model. Often retailers are focused on targeting the customers who will by more in the future. There can be many ways to segment customers. A company can get the voice of customers in many ways (not survey), like recordings from customer services, product reviews, ratings, etc. We can use the customer’s text review of the product to generate their sentiments, and by analyzing these sentiments accurately and analyzing the things that upset the customer, a retailer can focus more on what will make a difference. Using the available clustering method, we can segment them into different clusters. Recommendation model can be build using sentiment scores and the other user data available such as age, product ratings, gender, and demographic information.

**Business Understanding: Defining the problem**

Most of the marketing companies are able to segment the customers in order to create personal, timely and pertinent content, but how many companies know how their audience feel before they spend their scrutinized marketing budgets trying to get them to spend more? People will forget what you said, people will forget what you did, but people will never forget how you made them feel –this statement holds extremely true in the world of retail as well.

**Data understanding**

We will use the data from retailer’s database which will have consumer’s review about a product in text, product’s unique ID, and rating for product given by customer (1 to 5, being 5 the best), product features (like, department, class, type), and consumer’s demographic information such as age, gender, and location.

**Data Preparation**

We will drop rows with null values from the dataframe. Next we need to prepare the text review for generating the sentiment score. For that first we will convert all the text into lower case and we will use AFINN score calculation to generate customer’s sentiment score which can be positive as well as negative. The scores will be stored in a new variable in the dataset for each review text.

**Clustering**

K-Means clustering is a type of unsupervised machine learning that uses data that is not assigned to a specific category or group, called unlabeled data. K-Means clustering finds the most significant number of clusters and then determines the grouping of clusters based on the distance between the data points based on inputted attributes. For K-Mean we will use age and sentiment score and unrequired variables will be dropped. As the K-mean clustering requires, normally distributed variables, we will also check the skewness of the variables before applying the algorithm. We can use elbow method to determine the number of clusters. These clusters can be interpreted based on their age and sentiment scores. A marketing company can leverage this information to focus on a selected group and customize their strategy to have more satisfied customers.

**Recommendation**

Any organization looking to do business on the internet is interested in what its customers have to say. We want to know if we can predict rating based on the content of the review using recommendation model. We also want to see if we can predict how likely a customer will recommend products to their friends. A recommender system, or a recommendation system, can be thought of as a subclass of information filtering system that seeks to predict the best “rating” or “preference” a user would give to an item which is typically obtained by optimizing for objectives like total clicks, total revenue, and overall sales. The basic principle of recommendations is that there are significant dependencies between user- and item-centric activities. We will use collaborative filtering with the inbuilt recommender algorithm in the surprise package and for the evaluation cross-validation will be done with RMSE and MAE measurements.

**Modeling**

For the analysis Age, Customer sentiment score, Clothing ID, and Rating variables will be chosen, with 25% of random records as a test set. Scikit surprise package has many prediction algorithms such as SVD, KNNBasic, KNNWithMeans, KNNWithZscore, BaselineOnly, and NormalPredictor algorithms.

Mean Absolute Error (MAE) measures the average magnitude of the errors in a set of predictions, without considering their direction. It’s the average over the test sample of the absolute differences between prediction and actual observation where all individual differences have equal weight. Root mean squared error (RMSE) is a quadratic scoring rule that also measures the average magnitude of the error. Both MAE and RMSE express average model prediction error in units of the variable of interest. Both metrics can range from 0 to ∞ and are indifferent to the direction of errors. They are negatively-oriented scores, which means lower values are better.

**Deployment**

Best model will be chosen based on their accuracy and model error parameters, which will be used for predicting the product to a customer based on their age and sentiment score.