Ofir Iluz – 203835616

Ran Binyamini – 302940978

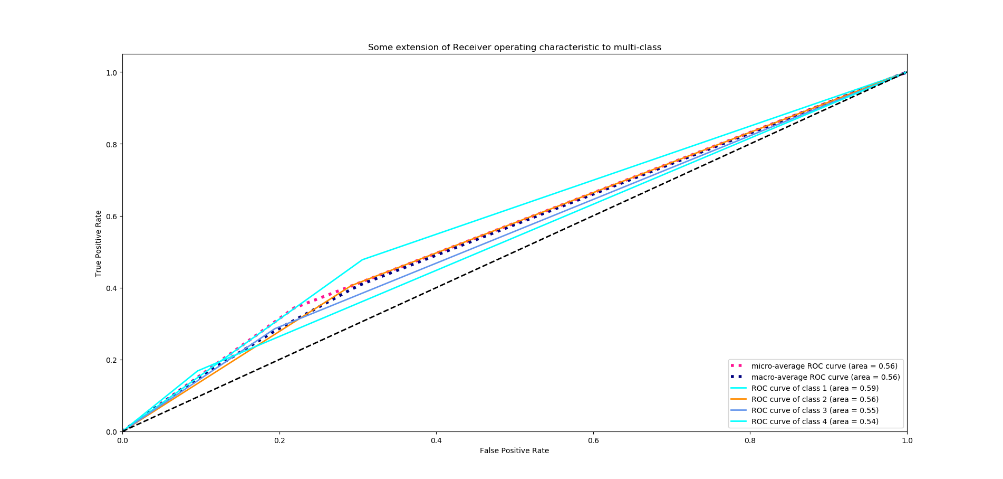
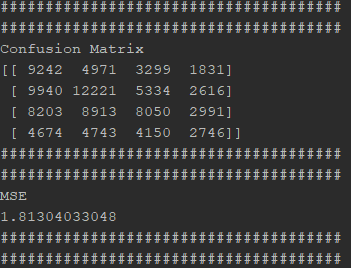
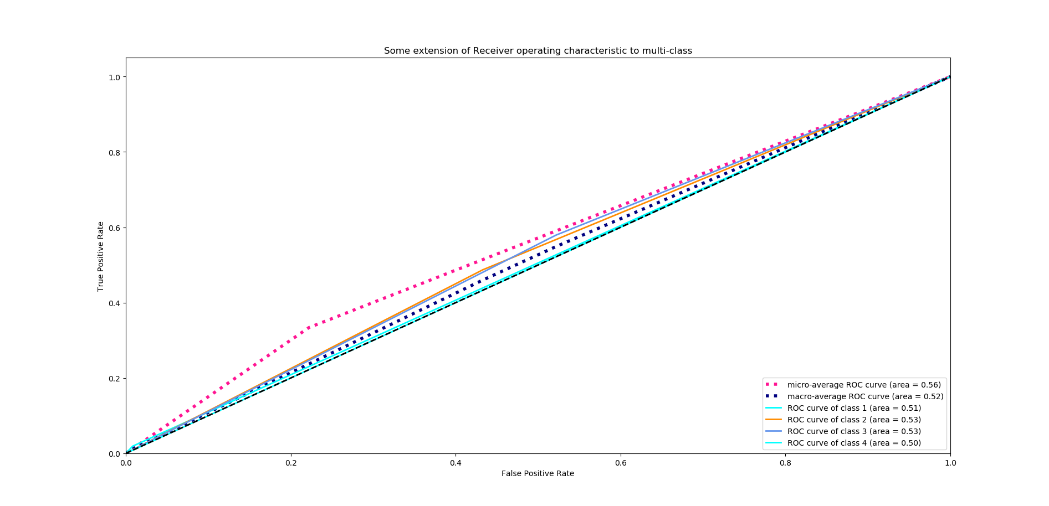
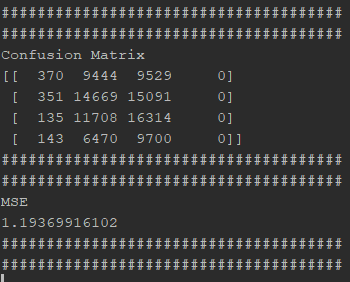
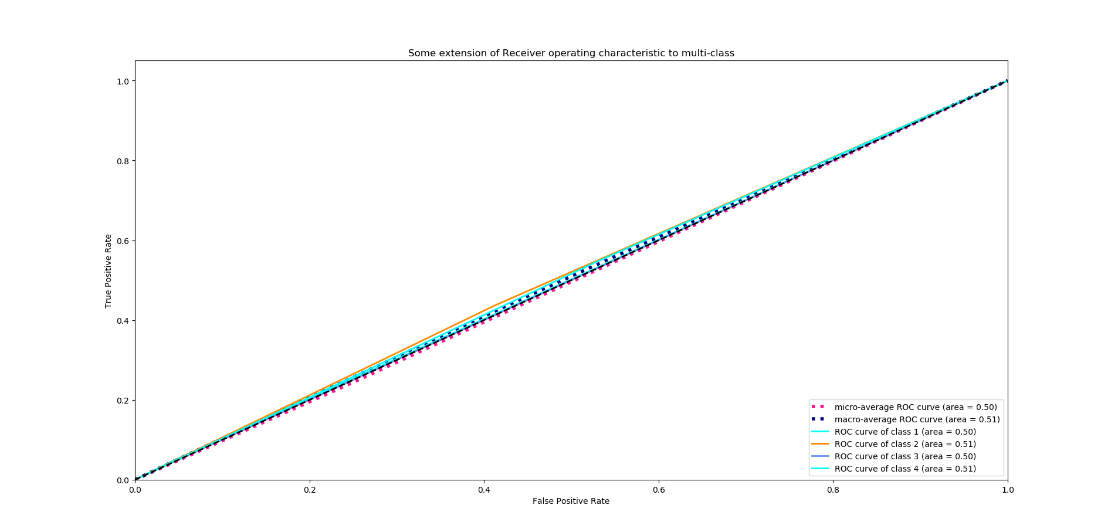
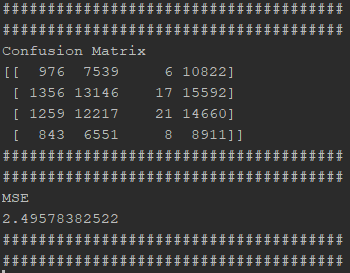
Yael Meloch - 204408645

**Data Science – Final Project**

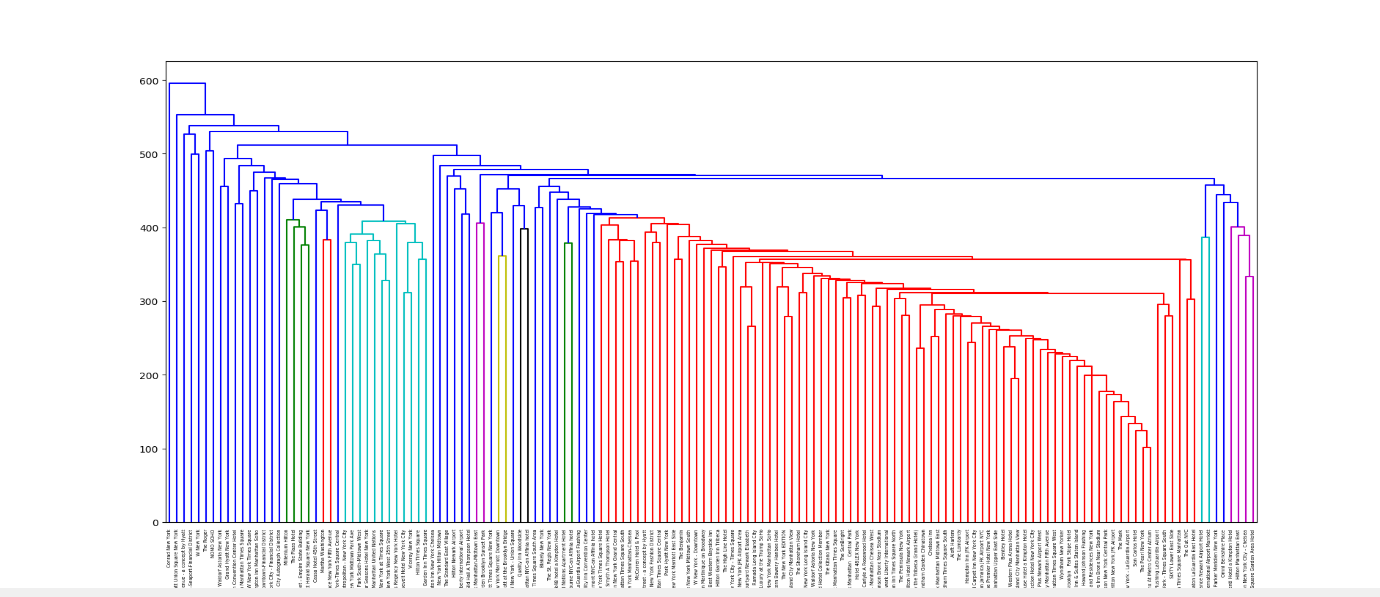
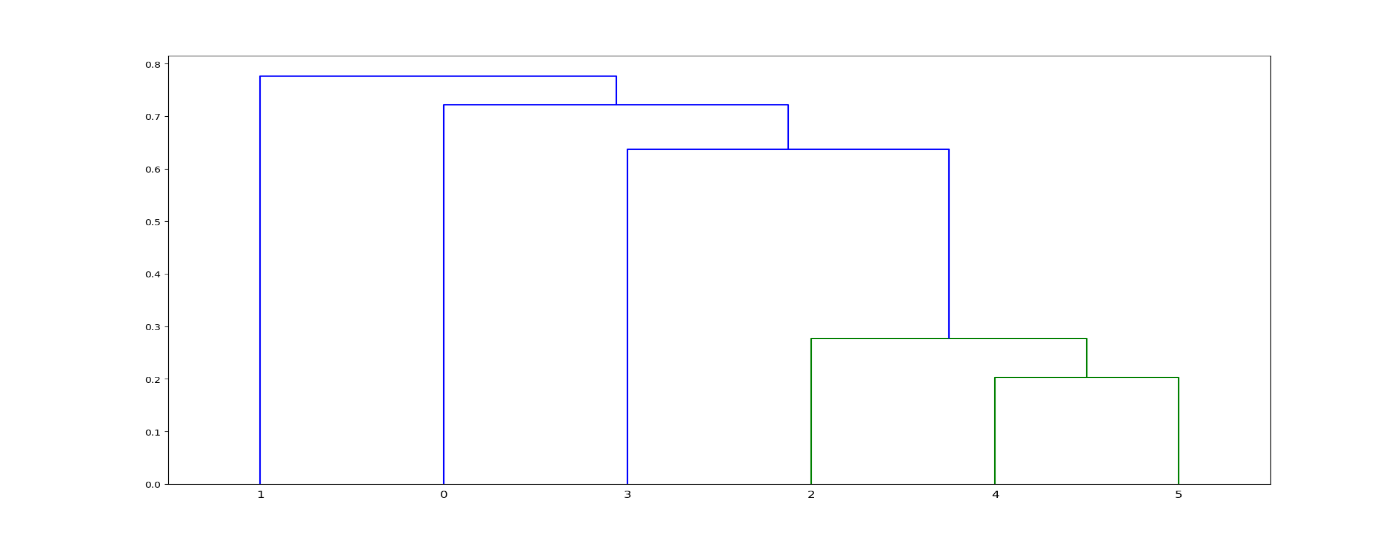
The project consisted of three steps which simulates to an extent, feature extraction, classification and clustering.

The results are as follows:

Step 2 – Classification

* This step consisted of taking the defined features and discount code as the target class and try to create a model to predict the discount code of future inputs.
* In this step, we tried using the following classifiers:
  + DecisionTree
  + NaiveBayes
  + SVM – Default params
  + OneVsRestClassifier as a wrapper to the other classifiers
* Results:
  + We have noticed that most notably, the weekday is a factor alongside the checkin date in the prediction
  + The classification raw results are low, and more precise feature extraction is required, or usage of other more concrete columns as our features.
  + When comparing DecisionTree with NaiveBayes, we have noticed that NaiveBayes acts better with the given data, and performs better, due to the nature of the classifier.
  + Note that the classification results are on the next pages
  + Classifcations results:
    - DecisionTree
      * ROC: (Multi class with avg)
      * Confusion Matrix (FP, TP etc) and MSE
    - NaiveBayes
      * ROC: (Multi class with avg)
      * Confusion Matrix (FP, TP etc) and MSE
    - SVM
      * ROC: (Multi class with avg)
      * Confusion Matrix (FP, TP etc) and MSE

Step 3 – Clustering **using pyspark**

* Step 3 consisted of taking step 1 output csv, and converting it into a new dataset, which consists of 161 columns, one for the hotel name (the index) and 160 columns each for the 40 highest checkin dates which exist within the top 150 hotels count wise, each date correlated with a discount code (1-4).
* Each row in the new dataset describes the discount price on the checkin date and discount code for the hotel, normalized between 0-100.
* After creating the data, we ran the clustering process in two different ways:
  + PDist clustering – using scipy pdist to compute the pairwise distance between all the rows and produce a distance vector and creating a hierarchy linkage with scipy that describes the hierarchal relationship between the columns.  
    Resulted dendrogram:  
      
      
      
      
      
      
      
      
      
      
      
      
    The dendrogram describes 6 hierarchal groups (for each colour), each group is a cluster of hotels that consist of distance wise relationship.
  + BisectingKMeans clustering – In this method, we tried finding relationships between cluster groups, to better understand the data using BisectingKMeans for the cluster centers and then creating the linkage dendrogram.  
    Resulted dendrogram on K=6:  
      
      
      
      
      
      
      
      
      
      
    The dendrogram descibers two groups of hierarchal clusters. Each group is a cluster of the centers from the KMeans.