SANTANDER PRODUCT RECOMMENDATION

BERKE ARAL SÖNMEZ

ALPER KOSE

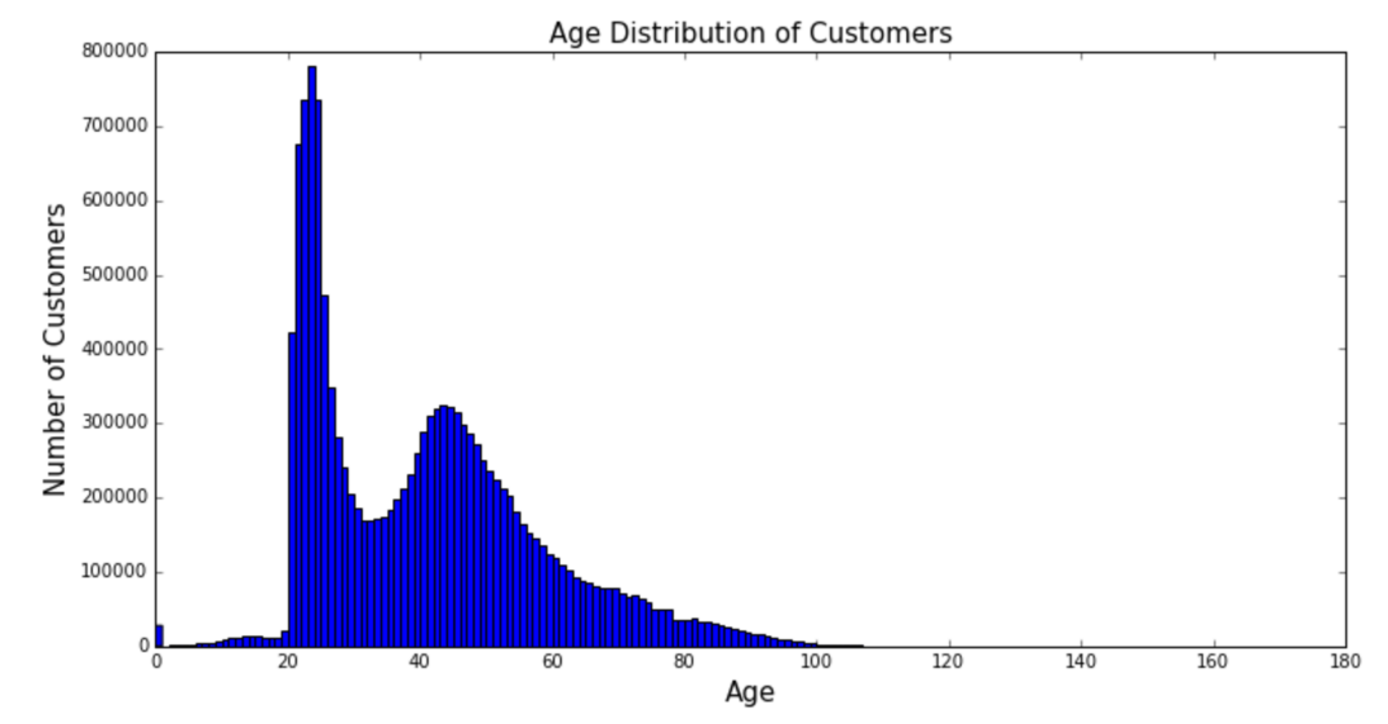
INTRODUCTION

- Training data with 13,647,309 data entries and test data with 929,615 data entries.
- Each training data entry has 24 feature columns and 24 class labels.
- Our aim is to recommend the products for the users in the test data given the features they have.
- We cleaned the data from unwanted data values and created some features based on the data we have.
- As a classification method, we used xgboost('Extreme Gradient Boosting') tool to build our model
- Finally, we recommended top seven products to the users in our test set according to the probabilistic scheme.
- Ranked among top 25% of the groups in Kaggle competition.

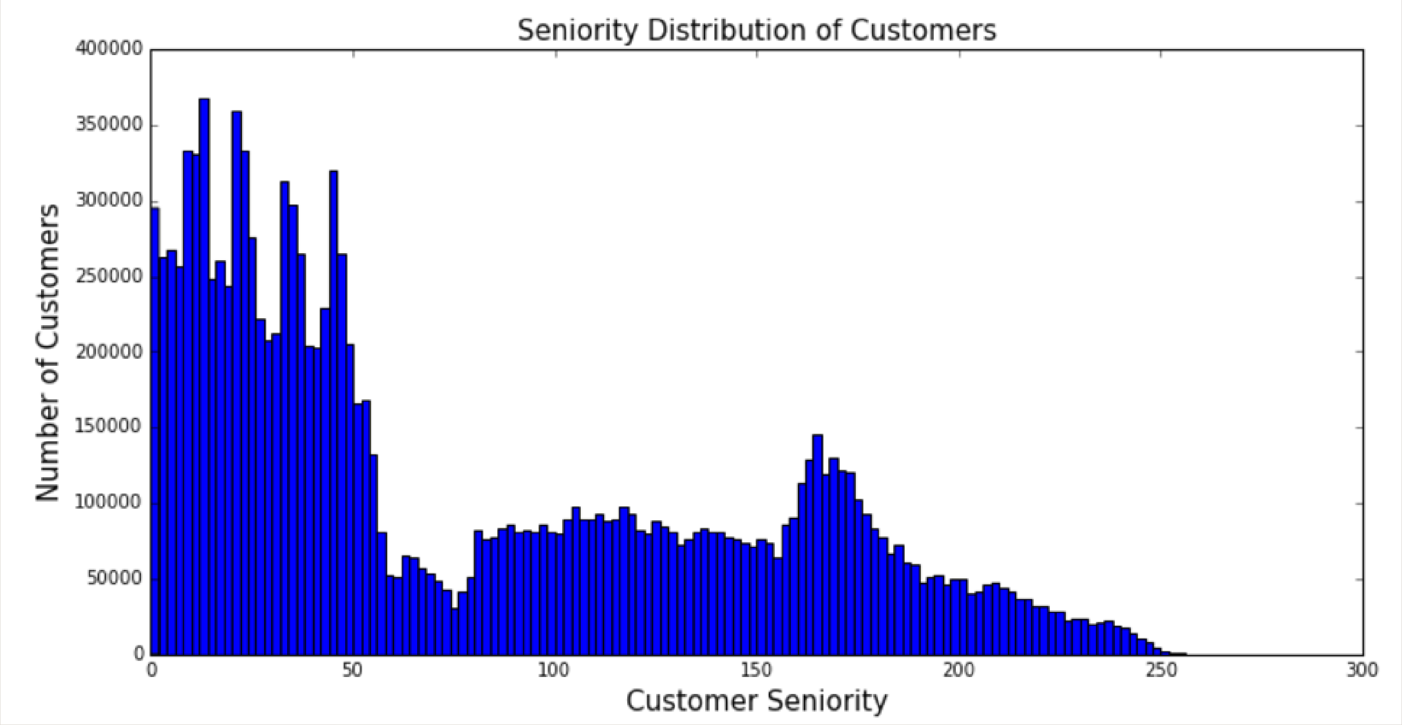
DATA CLEANING

- Purpose of the data cleaning is to remove unwanted data values.
- We need to put numerical values to our classifier therefore we transformed categorical values into numerical ones.
- Our training and test data has some nan or empty values as features and for numerical features we assigned the mean of that column while for categorical ones we assigned a number which corresponds to a nan.
- For empty class values we assigned 0 as probability of being 0 is higher than probability of being 1.

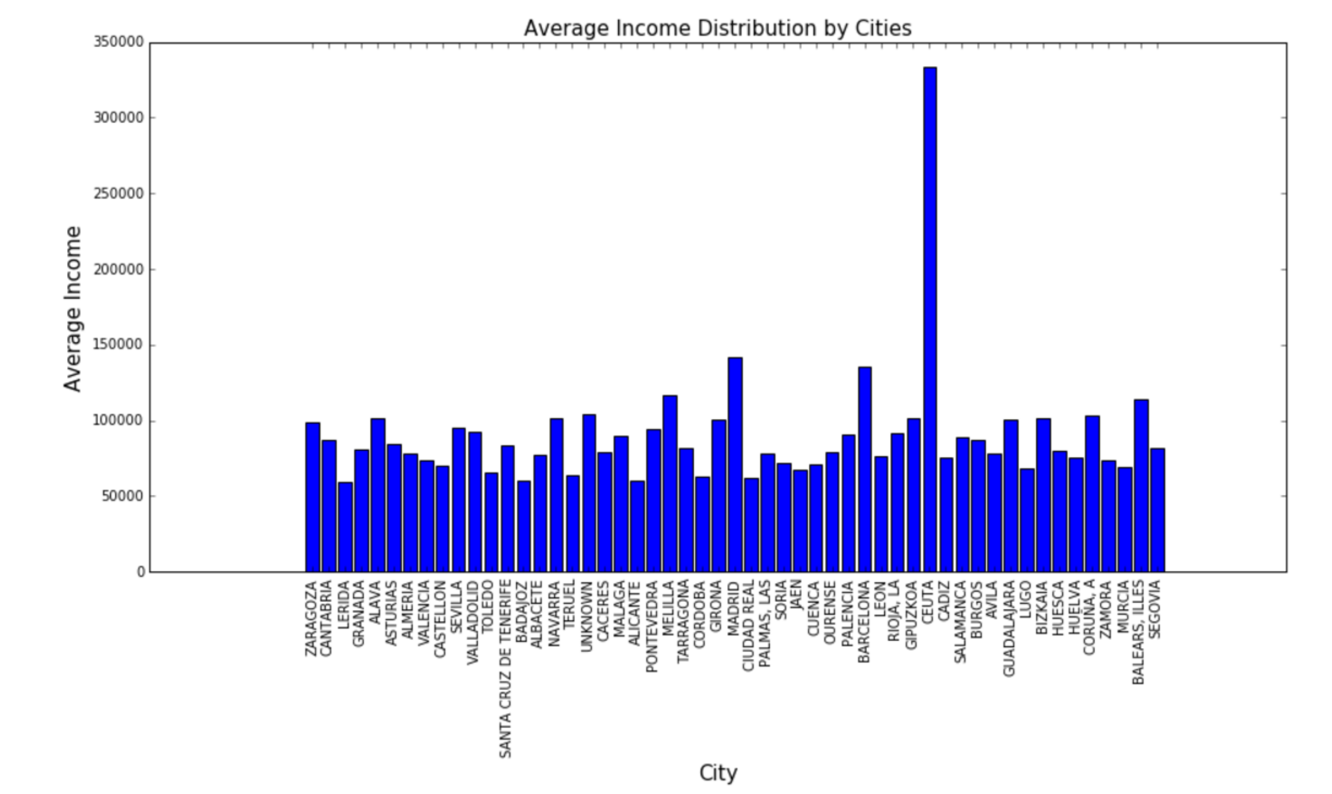
DATA CLEANING



DATA CLEANING



DATA CLEANING



FEATURE EXTRACTION

- Even though we have a training set of 13, 647,309 data entries, we only took data entries with certain dates and it corresponds to more than 25% of the training data.
- Aside from using the features we cleaned at the previous step, we also created some features based on the id of the customer.
- We created two dictionaries to record the products that a user bought in 4th and 5th month respectively.

FEATURE EXTRACTION

- If the date is 2015-06-28, we compared the bought products in this date by the same user with the bought products in the 5th month by that user. If this user didn't buy anything at that month we assign a zeros array with size 24.
- If there are not any new products bought by this user at '2015-06-28', we don't use this data.
- If there are new products bought by this user, we construct our features as the categorical and numerical features we derived previously, the products bought by this user in 5th month and 4th month respectively. Our label for this feature is the index of the new product bought by this user.

CLASSIFICATION MODEL

- In order to train our model, we used XGBoost which is an implementation of gradient boosted decision trees.
- It uses a gradient descent algorithm to minimize the loss when adding new models and commonly used on classification and regression predictive modeling tasks.
- XGBoost is used for supervised learning problems, where we use the training data (with multiple features).

CLASSIFICATION MODEL

- multi:softprob -->returns predicted probability of each data point belonging to each class.
- eta --> learning_rate
- silent = 1 --> no running messages will be printed
- num_class --> Number of classes that will be evaluated
- eval_metric = mlogloss --> Multi-class logarithmic loss is used as the evaluation metric in XGBoost
- Number of rounds is set to 100. (If we increase it, score improves as well as the computational time)

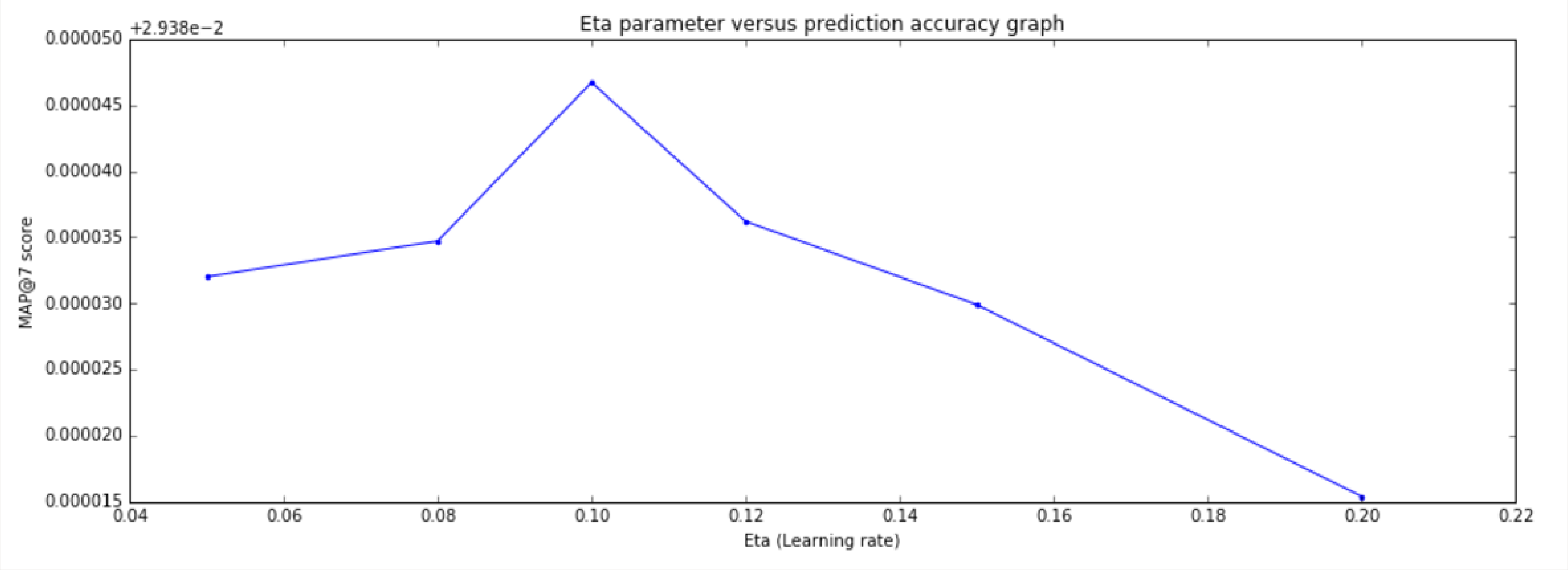
CLASSIFICATION MODEL

- In XGBoost, there are two learning methods for multiclass classification which are softmax and softprob.
- In softmax, we get the class with the maximum probability as output.
- In softprob, we get a vector with probability value of each class we have.

CLASSIFICATION MODEL

- In XGBoost, there are two evaluation metric for multiclass classification which are mlogloss, merror.
- In merror, the error objective only depends on classification error, it is calculated by number of wrong cases/number of all cases.
- In mlogloss, we try to minimize the negative of the log likelihood of the model, so it depends not only on the classification errors but how much error there is.
- We tried our model with both evaluation metrics and get similar results in Kaggle, so both of them can be used for our problem.

CLASSIFICATION MODEL



EVALUATION OF RESULTS

- We look at the prediction results from the xgboost classifier and we choose the 7 products with the highest values and decide to recommend those products for that user.

- Evaluation Formula is MAP @7:

$$MAP\ @\ 7 = \frac{1}{|U|}\sum_{u=1}^{|U|}\frac{1}{min(m,7)}\sum_{k=1}^{min(n,7)}P(k)$$

where |U| is the number of rows, P(k) is the precision at cutoff k, n is number of predicted products, and m is the number of added products for the given user at that time point.

- Ranked around 400 among 1787 participants in Kaggle.

Thank you for listening