CS412 - Introduction to Data Mining
Final Project Report

Abhinav Sharma (sharma55)
Udit Mehrotra (umehrot2)
Suhas Hoskote Muralidhar (shmural2)

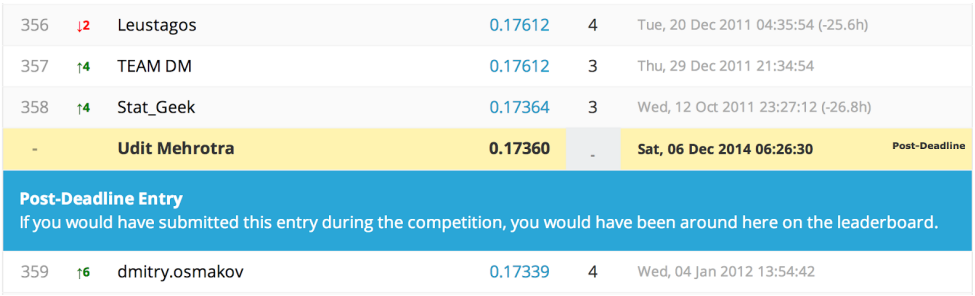
Algorithms Implemented

Pre-processing:

1. **Code:** project_preprocess.py
2. **Steps:**
    a. Keep original training.csv and test.csv in the same folder as the above code, which can be found in pre-processing folder under code directory.
    b. run python project_preprocess.py
    c. Output:
        i. x.csv - pre-processed training file
        ii. y.csv - pre-processed test file
3. We have used only above code for main pre-processing and using output of the above code, specific minor manual pre-processing is carried out for individual classifiers.

Classifiers:

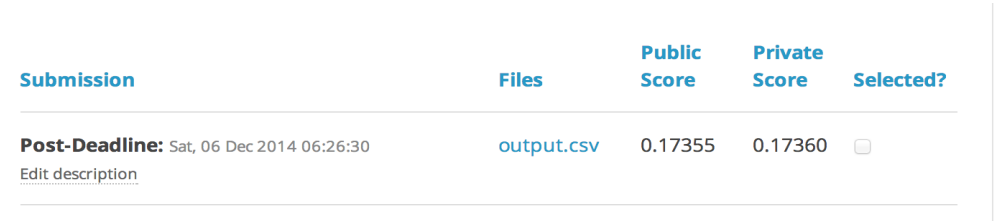
<u>Naive Bayes Classifier</u>: This algorithm provides the best score.



356	↓2	Leustagos	0.17612	4	Tue, 20 Dec 2011 04:35:54 (-25.6h)	
357	↑4	TEAM DM	0.17612	3	Thu, 29 Dec 2011 21:34:54	
358	↑4	Stat_Geek	0.17364	3	Wed, 12 Oct 2011 23:27:12 (-26.8h)	
-		**Udit Mehrotra**	**0.17360**	-	**Sat, 06 Dec 2014 06:26:30**	Post-Deadline

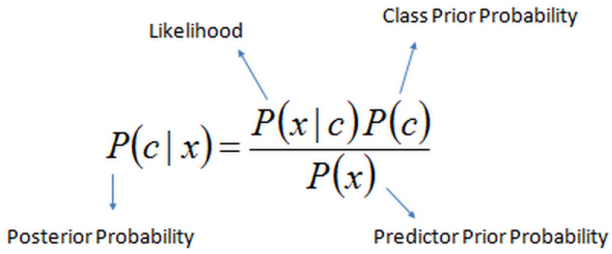
Post-Deadline Entry
If you would have submitted this entry during the competition, you would have been around here on the leaderboard.

359	↑6	dmitry.osmakov	0.17339	4	Wed, 04 Jan 2012 13:54:42	



Submission	Files	Public Score	Private Score	Selected?
Post-Deadline: Sat, 06 Dec 2014 06:26:30 Edit description	output.csv	0.17355	0.17360	

    a. **Private Score:** 0.17360 (Best Score)
    b. The Naive Bayesian classifier is based on Bayes' theorem with independence assumptions between predictors.



$$P(c \mid \mathrm{X}) = P(x_1 \mid c) \times P(x_2 \mid c) \times \cdots \times P(x_n \mid c) \times P(c)$$

c. **Parameters:** Laplace smoothing to avoid zero probability issue.

d. **Analysis for Improvement:**

   i. **Pre-processing:**

      1. In both training and test files we replaced NULL values for numeric attributes with the mean value of that attribute.
      2. For categorical variables with values like NULL, Not Available we replaced them with a new unique value in both training and test files.
      3. We converted the unique text categorical values, to unique numeric labels by assigning a unique number to each value.
      4. After careful analysis, we also created 4 new attributes based on existing MMR values as below:
         a. ProfitAcquisitionAverage= MMRAcquisitionRetailAveragePrice - MMRAcquisitionAuctionAveragePrice
         b. ProfitAcquisitionClean= MMRAcquisitonRetailCleanPrice - MMRAcquisitionAuctionCleanPrice
         c. ProfitCurrentClean= MMRCurrentRetailCleanPrice - MMRCurrentAuctionCleanPrice
         d. AverageProfit = ProfitAcquisitionAverage - ProfitCurrentAverage
         e. CleanProfit = ProfitAcquisitionClean - ProfitCurrentClean
      5. We also increased the weights of certain attributes, which enabled better accuracy with NB classification.

   ii. Gaussian distribution is used for handling numeric attributes to classify test data set.

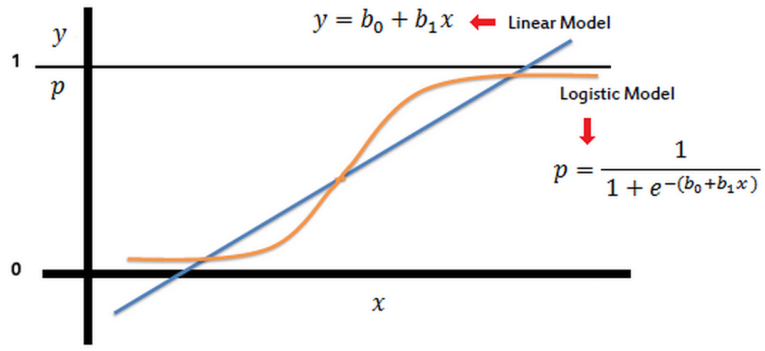
Logistic Regression:

504	↑9	PISHI	0.09113	7	Tue, 03 Jan 2012 18:52:51 (-13.9d)	
505	↓1	Rajag	0.09093	9	Tue, 03 Jan 2012 00:06:03 (-9.3d)	
-		**Outliers**	**0.09047**	-	**Sat, 06 Dec 2014 08:42:44**	Post-Deadline
Post-Deadline Entry If you would have submitted this entry during the competition, you would have been around here on the leaderboard.						
506	↑10	grigori1	0.08970	3	Thu, 05 Jan 2012 15:39:11 (-0.1h)	
507	↓6	Petko Nikolov	0.08836	3	Thu, 22 Dec 2011 12:12:49 (-2.5d)	

Submission	Files	Public Score	Private Score	Selected?
Post-Deadline: Sat, 06 Dec 2014 08:42:44 Edit description	result.csv	0.09510	0.09047	■

e. **Private Score:** 0.09047

f. Logistic regression predicts the probability of an outcome that can only have two values (i.e. a dichotomy). The prediction is based on the use of one or several predictors (numerical and categorical).



g. **Parameters**:
   i. Learning Rate - 0.015 Otherwise it was not converging and results were not good. Also, we didn't want the learning rate to be high because the data is so imbalanced it is very hard to converge or to find good weight parameters, so we kept it low.
   ii. Max iterations - 250. Just so that there are enough iterations to converge

d. Analysis for further improvement

i. Theta Parameters - We initialized weight parameters at first and then ran the algorithm 5 times to check accuracy on cross validation set and when we got the good score on kaggle we saved those parameters because otherwise it was not converging sometimes

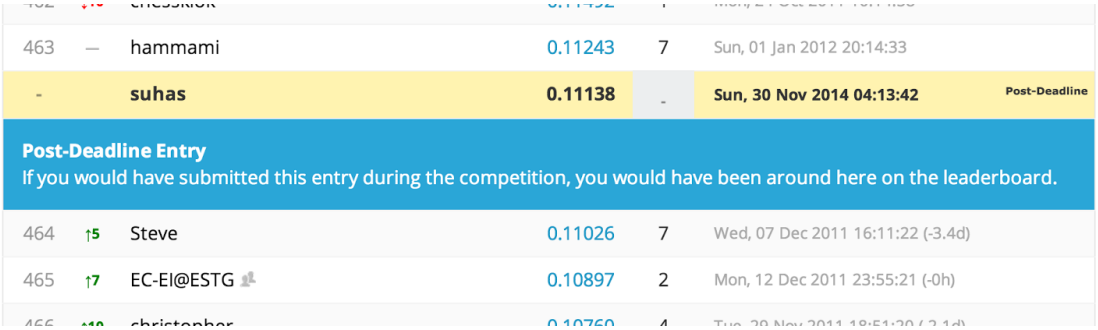
iii. Convergence - Since, normally we don't have gradient descent optimization of functions in normal in-built libraries, we checked for convergence when the cost function was not decreasing by more than 10**-3.

iv. Attributes - We performed this algorithm only on numerical columns.

v. Also along with new attributes created as per pre processing of Naive Bayes, we added 2 more attributes as below:

1. ProfitCurrentAverage= MMRCurrentRetailAveragePrice-MMRCurrentAuctionAveragePrice
2. OdoPAge = VehOdo - VehicleAge

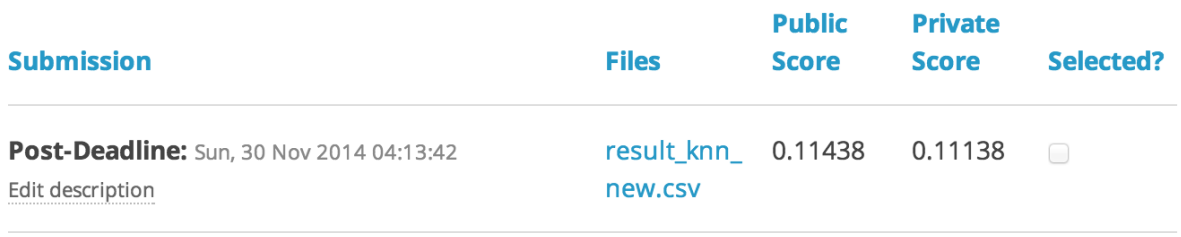
K-Nearest Neighbor Classification (K-NN):



463	—	hammami	0.11243	7	Sun, 01 Jan 2012 20:14:33
-		**suhas**	**0.11138**	-	**Sun, 30 Nov 2014 04:13:42** Post-Deadline

Post-Deadline Entry
If you would have submitted this entry during the competition, you would have been around here on the leaderboard.

464	↑5	Steve	0.11026	7	Wed, 07 Dec 2011 16:11:22 (-3.4d)
465	↑7	EC-EI@ESTG	0.10897	2	Mon, 12 Dec 2011 23:55:21 (-0h)



Submission	Files	Public Score	Private Score	Selected?
Post-Deadline: Sun, 30 Nov 2014 04:13:42 Edit description	result_knn_new.csv	0.11438	0.11138	☐

h. **Private Score:** 0.11438

i. **Pre-processed file :**

i. training file: x10.csv

ii. test file: y10.csv

j. **Code**: KNN.java and ABC.java

k. K-NN is based on lazy learner approach, where the learner waits till last minute before doing any model construction to classify the test tuple.

l. A k-nearest-neighbor classifier searches the pattern space for the k training tuples that are closest to the unknown tuple. These k training tuples are the k

"nearest neighbors" of the unknown tuple.

m. **Parameter Selected:** K = 7.

n. **Analysis for improvement:** Since the training data is unbalanced i.e. it has more 0s (87% of training data) than 1s so we are considering if more than 40% of class labels for selected K is 1, we are predicting the label as 1.

o. "Closeness" is defined in terms of a distance metric

   i. For Numeric values we considered Euclidean distance

Euclidean $\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$

   ii. For Nominal or Categorical values we considered Hamming distance, i.e. if two values are same, distance is considered as 0, else 1.

Hamming Distance

$$D_H = \sum_{i=1}^{k} |x_i - y_i|$$

$$x = y \Rightarrow D = 0$$

$$x \neq y \Rightarrow D = 1$$

   iii. Finally, the total distance is calculated by adding Euclidean distance and hamming distance for all the tuples.

p. We also applied min-max normalization on all numeric values as part of standardization process.

Work Distribution:

q. Suhas Hoskote Muralidhar (shmural2)
   i. Pre-processed both training and test files to run K-NN algorithm
   ii. Implemented K-Nearest Neighbor classifier to classify test data set.

r. Udit Mehrotra (umehrot2)
   i. Pre-processed both training and test files for Naive Bayes algorithm.
   ii. Implemented Naive Bayes classifier with Gaussian for handling numeric attributes to classify test data set.

s. Abhinav Sharma (sharma55)
   i. Implemented general preprocessing technique code which can do all the preprocessing like smoothing, transformation, word binning, and creating new parameters by passing functions on how to convert them.
   ii. Implemented Logistic Regression on the 60% training set and used rest as test set for checking accuracy and then predicted on test data set.