Predicting Runner Participation and Finishing Time for the 2016 Montreal Marathon

Eitan Bulka, eitan.bulka@mail.mcgill.ca, 260457154
Xiliang Zhu, xiliang.zhu@mail.mcgill.ca, 260666980
Orestes Manzanilla, orestes.manzanilla@polymtl.ca, 1863009 (Polytechnique ID)

I. Introduction

For the upcoming (2016) Montreal Marathon, our task is to predict who will race in the full marathon, as well as the time they take to finish. Our algorithms are trained using data from previous Montreal Marathons. The prediction methods and results can be broken into three parts: a participation prediction using a logistic regression approach, a participation prediction using a Naïve Bayes classifier, and a finishing time prediction using a linear regression approach.

II. Predicting Participation: Logistic Regression

A. *Problem Representation*

For the Logistic Regression, we extracted data from the previous 4 year's marathons held in Montreal, as well as age and gender information. Attendance to each year was represented as a binary variable with value of 1 when the participant attended to the marathon, and 0 otherwise. Year 2015's data is considered the target of the training and the data from the years '12, '13 and '14 are considered the explicative features. It is assumed that relationship to attendance in different years can be related. In order to consider these relationships, and lower the sparsity of the data, the features used are the sums of each pair of year's attendance. Specifically, the total attendance of '14 and '13 was considered a feature, as well as the total of '14 and '12, and the total of '13 and '12. The sum of these three features was considered as an additional feature, as well. The choices made regarding the rest of the features produced to representations.

In the first representation (the simplest one) age was represented initially as a continuous variable, and gender was represented as a binary variable with values of 1 for male competitors, and -1 for female competitors. A 0 was used in those cases where gender and/or age were missing.

In the second representation, age as a feature was substituted by 6 binary variables, whose value was 1 if the participant belonged to a specific age range, or 0 otherwise. The ranges were defined as follows: "from 10 to 20", "from 21 to 30", "from 31 to 40", "from 41 to 50", "from 51 to 60", and "from 61 to 70". Those whose age was missing, were assumed to be in the range where lies the mean of the age, which is the range from 31 to 40. One additional feature was created to cope with possible non-linearities, which is the sum of the squares of the features.

B. *Training Method*

For both representations the coefficients were fitted using Gradient Descent without regularization term. Various hyper-parameters were fitted (in this order):

1. "α" (Learning rate or Step-size)
2. Values for vector of parameters "θ" (with α fixed in the best value found)
3. Number of iterations (with the α and θ values fixed in the best values found)

For simplicity, the values of the initial values for the vector of parameters "θ" were considered identical. Specifically, the accuracy of the results were compared for the cases where the initial solution is a vector of zeros, a vector of ones, etc.

Hyper.-parameters were chosen so to minimize the mean validation error estimated via 10-fold cross-validation (total of misclassified individuals / total of individuals). In order to implement th k-fold cross-validation, a vector of folder labels was created, inspired by ("Im so confused, 2012).

C. *Results*

The best results for the first representation (considering age a continuous variable), a mean error rate of 20.62% was achieved in both training and validation sets, with α=0.01, θ =1 for all parameters, and Maximum No. of Iterations = 1000. For the second representation (considering age as a set of binary variables, adding as well the sum of the squares of the features), was achieved a mean error of 20.68% both in training and validation sets. To make predictions on 2016, the data of the previous 3 years was used, in the first representation, without holding a validation set. The resulting discriminating model has the form:

$$P(X=1)=\frac{1}{1+e^{-X*\theta}}$$

where $\theta=[\theta_0,\theta_1,\ldots\theta_6]$ and

- θ_0 =-0.604692, is the coefficient of the independent term
- θ_1 =-0.169379, is the coefficient of "15+14"
- θ_2 =-0.216503, is the coefficient of "15+13"
- θ_3 =-0.288030, is the coefficient of "14+13"
- θ_4 =-0.679027, is the coefficient of "sum"
- θ_5 =0.186354, is the coefficient of "gender"
- θ_6 =0.659913, is the coefficient of "age".

We predict x=1, if P(x=1)>0.5, and x=0 otherwise (threshold is 0.5).

D. Discussion

Results for Logistic Regression were less promising than those of Naîve Bayes. In particular, around 30% of the individuals participate in 2015, so a broad guess of not attending to the competition would give already 30% of error, which is very close to the results in both of the representations used to fit this model. Even though the curves of the Error and the 2-norm of the gradient show a good behavior along the iterations of the algorithm (see Figure 1 and Figure 2), the fact that the Error is so high, and that the difference between training and validation errors is so small, suggests that the models are a case of High Bias or under-fitting. We trained the logistic regression with samples of the dataset of different size, and noticed that as the size increments, training and validation errors converge to each other, indicating that the model is too simple (see Figure 3). Two strategies seem plausible to get better results in the future: increasing the number of features by processing data that has been omitted in this study (in particular the text data), or creating additional variables (for example 2 by 2 interactions between the features).

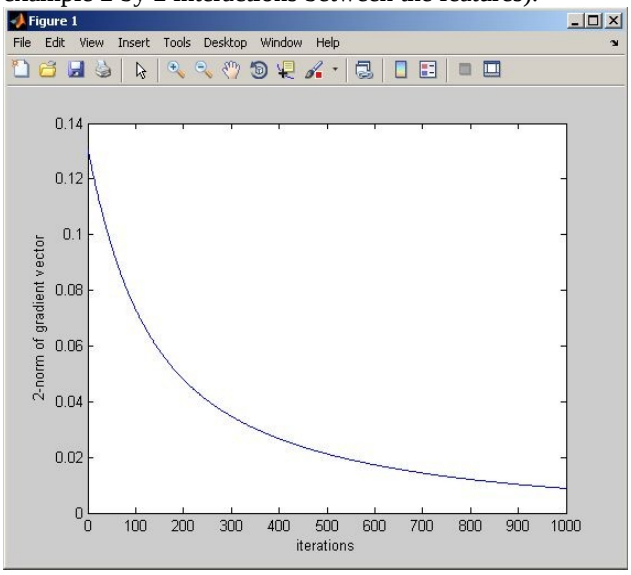

Figure 1: Binary Age Gradient vs. Iterations

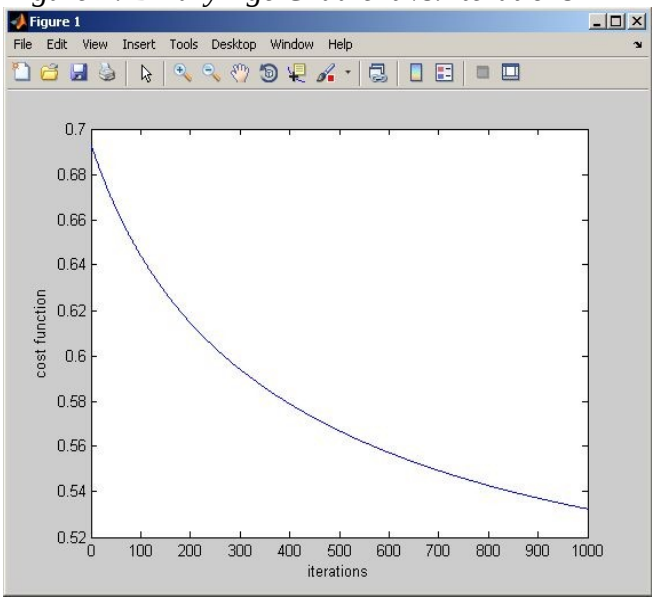

Figure 2: Binary Age Error vs. Iterations

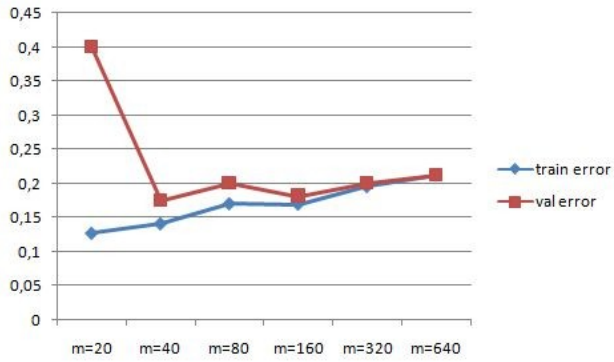

Figure 3: Data Size vs. Accuracy

III. PREDICTING PARTICIPATION: NAIVE BAYES

A. Problem Representation

For the naïve bayes part, we first extract data from the previous 4 years' marathons held in Montreal. Those data are most representative for the participation prediction. Then, to change the information we need from words to numerical value, we simply set 1 for the attendance and 0 for the absence for the previous events. Our main methodology is using year '15 data as our Y and year '14, '13, '12 data as X. This is because we can assume the consecutive years' attendance will follow the same pattern. So after training from these data, we can get one certain set of weights that can be applied to year '15, '14, '13 to predict year '16.

Due to the sparsity of this particular dataset and to illustrate the connection for certain years, first three features used are the results of '14+'13, '14+'12, '13+'12. This approach can smooth the influence of large amount of 0s in our data. After that, we sum these three features up as the 4th feature which can represent the total attendance in the previous years. Finally, we extracted gender and age information as our 5th and 6th features. Male are set as 1 while female are set as -1.

B. Training Method

For the training part, we split our X and Y data into 8:2 to represent training and testing set respectively. Then we calculate each P(xi|y) for every feature in our training X set. After getting all the probabilities, we can apply them to the Bayes formula to calculate the training and testing error rate, along with our prediction for year 2016.

C. Results

The results reached an error rate of 0.179 for testing set and 0.174 for training set. So those rates reflect a good training model on Naïve Bayes.

Finally, we apply the same feature extraction method on dataset of year ’15, ’14, ’13, that is, use ‘15+’14, ‘15+’13, ‘14+’13 as first three features and the same sum, gender, age data as 4th-6th features. This will be our final dataset for predicting the attendance of 2016. So use Naïve Bayes formula and apply the probabilities and X array, thus, we obtained the final prediction.

D. Discussion

From the prediction output, we noticed that the classification results are quite reasonable. However, we think there is still some work to be done on the feature selection since the age feature seems to play too big a role in our prediction, which is not that intuitive. So we believe the future improvement should be the part that how to set a proper representation to increase the importance of attendance feature.

IV. PREDICTING FINISHING TIME: LINEAR REGRESSION

A. Problem Representation

For predicting the finishing time, only data from full Montreal Marathons is used. Any data pertaining to a race in another city, or a different type of race (i.e half marathon, 10k etc..) is thrown out. The reason is as follows: full marathons in other cities have many additional variables that could affect a runner's finishing time, such as how hilly the race is or the altitude of city. Other types of races are also not considered because they are substantially different than a full marathon. Only two features are used to predict a participant's finishing time: their average finishing time, x1, and their most recent finishing time, x2. Both of these features are represented as continuous variables and have a unit of seconds.

B. Training Method

In order to obtain weights in the linear regression model, we need to obtain a set of features (average finishing time and most recent finishing time or x1 and x2) and a set of outputs to go along with those features (finishing time predictions or y). The finishing times from 2015 are used as outputs (y), and data from 2012-2014 is used to generate the features. In order to obtain this set of training data, only participants who ran the full Montreal Marathon in 2015, and had run at least one previous full Montreal Marathon, could be used for training. It should be noted that if a participant did not finish the race, they were treated as if they had never ran that race (for predicting finishing time ONLY). After sifting through the data, a set of 211 runners were obtained for training, who had all ran the full montreal marathon in 2015 and at least one from 2012-2014.

C. Results

A cross-validation study is performed to evaluate two potential linear regression models. The first model includes a bias term and makes predictions using the following model:

$$y=w0+w1x1+w2x2$$

The second model does not a have a bias term, is described by the following equation:

$$y=w1x1+w2x2$$

Where once again, x1is the average finishing time and x2 is the most recent finishing time. A 4-fold cross-validation is performed, where the set of training data (the 211 participants who ran full Montreal Marathons in 2015 and at least one between 2012 and 2014) is divided into four groups. For experiment 1, the first group of data is withheld from the data set to obtain the weights. These weights are used in the linear model to predict the finishing times on the first group. The absolute value of the error for each prediction is averaged to obtain an average validation error for experiment 1. Predictions are also made on the other 3 groups, which give rise to an average error on training data. This experiment is repeated 3 more times. The results from each experiment are averaged to obtain 1 validation error and 1 training set error. This entire process is performed twice, one with the model including the bias term, and once without.

The results are as follows:

Partition	Training Error	Validation Error
1	9.02%	9.42%
2	8.91%	10.46%
3	9.27%	8.90%
4	9.16%	8.91%
Average	9.09%	9.42%

Cross-Validation With Bias Term

Partition #	Training Error	Validation Error
1	8.84%	9.23%
2	8.78%	9.61%
3	9.00%	8.82%
4	9.05%	9.05%
Average	8.92%	9.18%

Cross-Validation Without Bias Term

Although the results are very similar, the results without a bias term are slightly better. In addition, it intuitively makes more sense that there wouldn't be a bias term, and the prediction would be a percentage of the average time summed with a percentage of the most recent time.

When re-training on the full data set using the no-bias model, the following weights are obtained: w1=.335, w2=.658. Based on intuition one would think they would add to 1, which is almost the case. Ultimately, the data demonstrates that a person's finishing time is about ⅓ correlated to their average finishing time, and ⅔ correlated to their most recent finishing time.

D. Discussion

Ultimately the following model was created, y=.335x1+.658x2, where y is the predicted finishing time in the 2016 Montreal Marathon, and x1 and x2 are the participants average and most recent finishing times in Montreal Marathons from 2012-2015. Their is one main drawback to this approach, which occurs a when a participant has never ran a full Montreal Marathon from 2012-2015. The model fails for this type of person, and a time of 4:30:00 is predicted as a guess. I would expect this model to perform well for runners who have ran a previous Montreal Marathon, but perform poorly for runners who haven't. The model could potentially be improved by formulating a regression between other types of races and full marathons, as well marathons in other cities to the Montreal Marathon, to obtain better predictions for participants who have never ran a full Montreal Marathon.

STATEMENT OF CONTRIBUTIONS

Orestes wrote the code, performed the data analysis, and wrote the part of the report pertaining to the Logistic Regression method. Xiliang came up with the methodology for Y1 (predicting participation), and wrote the code, performed the data analysis, and wrote the part of the report pertaining to the Naive Bayes method. Eitan came up with the methodology for Y2 (predicting finishing time), and wrote the code, performed the data analysis, wrote the part of the report pertaining to the Linear Regression method, and assembled the report, code and predictions for subsmission.

REFERENCES

[1] "im so confused" (2012, September 27). Stack Overflow: MATLAB: 10 fold cross Validation without using existing functions [Online forum comment]. Message posted to http://stackoverflow.com/questions/12630293/matlab-10-fold-cross-validation-without-using-existing-functions