Performance Prediction of runners in 2017 Miami marathon

Gavin McCracken
McGill ID: 260518506
Email:gavin.mccracken@mail.mcgill.ca

Shruti Bhanderi
McGill ID : 260724575
Email: shruti.bhanderi@mail.mcgill.ca

Zachary Warnes
McGill ID:260581958
Email:zachary.warnes@mail.mcgill.ca

I. Introduction

We analyze the use of some machine learning algorithms to predict the performance of probable participants of 2017 Miami marathon based on their previous performance results.We are provided the record of an athlete's performance at a Miami marathon event from previous years. The previous performance data of athlete was summarized in a standardized vector in order to feed to machine learning algorithms.An analysis has been carried out using different sets of athletes and their performance.The results show that, for the studied datasets and proposed set of attributes, Naive Bayes classifier provides rather good estimates of the presence of athletes in a challenging and unpredictable race such as the marathon.

II. Problem Representation

In this machine learning system, we will be using athletes characteristics (age, gender) and their previous performance history to determine if they will be participating in the 2017 Miami Marathon(Y1). We will also use these features to estimate a finishing time for all athletes(Y2), regardless of expected participation in 2017 marathon.

A. Features

Our learning data set for Y1 and Y2 includes all those who have participated in the previous years and their results. For deciding the weights of each feature provided we have deduced the minimum cross validation errors and accordingly decided the importance of each feature. The detailed description for feature selection in Y1 and Y2 prediction will follow.

1) Y1 Prediction features: For our prediction of Y1 (presence in the marathon) we have considered the features mentioned in the Table 1.

Feature Name	Type	Description
Number of races	continuous	number of races given a person
Age	continuous	ages 0 - 98
Pace	continuous	303-1155
Rank	continuous	rank 1-3913

TABLE I. Y1 Prediction

In addition to this variable, our computation of a participation ratio helps to model how many continuous Miami marathons an individual has competed in, with the assumption that a general trend of continuity may very well continue this year. Specifically, for Logistic regression, we have used the portion of the years which the runner participated and try to weight years closer to the current year higher, using distribution method. We also considered gender and age to be features but it is not providing great meaning to results.

2) Y2 Prediction features: For our prediction of Y2 (the finish time of the participants) we have considered features four features named :age, gender, years and "improvementRatio".Out of the four features, age, gender and years are provided in our data set, and "improvementRatio" is derived by taking the average of the finish time of runners' till the year 2015 and taking ration of this average and 2016 finish time, which will in turn provides the better information about their improvement in terms of recent results. Table 2 indicates the better view of all the features considered for predicting Y2. Running is a dynamic activity but our hope is that this ratio accurate models improvement or decline of a runner.

Feature Name	Type	Description
Age	continuous	ages 10 - 98
Gender	categorical - boolean	women(0) or men(1)
Years	continuous	2003-2016
Improvement Ratio	continuous	time from most recent year : average time

TABLE II. Y2 Prediction

An important design choice we made was to completely disregard all finish times for this method, seeing as we found it only minorly relevant how fast someone is compared to how frequently he or she runs marathons and we did not want it to skew our results. For our prediction of Y2 (the finish time of the participants) we used almost exactly the same features with a few slight variations. We also thought of adding a z-score of an athletes marathon time and considered extrapolating this to include z-scores of previous years' data, but it may skew the results so dropped the idea.

B. Representation

For results representation, we are displaying the predicted finish time in a continuous output as it will be rounded to the nearest second. While predicting about whether the athlete will participate this year will be a categorical representation(0 or 1). Using logistic regression for predicting Y1 will separate the data by line.

C. Design Choices

In our initial approach to data we used regular expression to identify entries with abnormal values. At first, we compared the data with the time for the world record for a full marathon. If someone had a time/pace which is faster than the world record, it can be deduced to be incorrect. We found 3700

such entries but on further analysis we come to the point that it could be corresponded to half marathon times.Since pace is always time per mile, we can deduce whether an athlete has participated in a half or full marathon. As we are only predicting participation and finish times for a full marathon we can ignore these extra entries.

D. Additional data

We considered about adding the weather, race routes etc for the better results. But later point our intuition denied to add any other data as it might skew the results.

III. Training Method

There are 38805 athletes' data in the csv file provided, out of which the name field having 'private' value was removed and are left with 34721 entries.

We will be using two methods to predict whether they will run in this years Miami Marathon or not : Method 1 will use a Logistic optimized with gradient descent and regularization and Method 2 will use Naive Bayes.

A. Predict Y1 using logistic regression

We used Logistic Regression with L2 regularization using gradient descent. We varied our regularization parameter from λ = [0.1,0.3,1.0,3.0,10] and evaluated the performance using 5-Fold cross validation to choose the best value. We additionally optimized alpha at 1.0. After experimenting with other stationary alpha values(0.001,0.003,0.01,0.3,1.0,3.0,10), we found that, while the code was slightly slower to finish running, this dynamic alpha resulted in a more stronger local optimum. We had the added option of varying our threshold for our sigmoid function, to adjust the values we obtained for recall and precision. Which can be altered to favour negative or positive results.

B. Predict Y1 using a Naive Bayes classifier

Feature Manipulation: All features were used in the Gaussian Naive bayes classifier which can be used for obtaining a 'participation likelihood'. For Training Method, we used a Naive Bayes classifier to calculate the respective ratio for each candidate, in a given year. In our model, there is no possibility that one of the probabilities will equal zero. This avoids divide by zero errors, and remains true as long as floating point underflow does now occur. Floating point underflow could've been avoided altogether by using natural logarithms to compute the probabilities, however there was no need in all computed examples because there was no risk of this happening.

C. Predict Y2 using linear regression

1) Data Selection: We trained our model using data from all previous year data, to predict the finish time for 2016. When predicting for 2017, we took the entire data from 2003 to 2016 and recreated our features. This limited us to choose features which were year independent. When training we only considered the participants, who at least participated and had a finish time in any of the years before 2017. This was done to avoid the noise in the feature 'improvementRatio' but also cost us a large amount of data. All the features were normalized according to their Z-Valuethis was chosen intentionally so that our feature 'improvementRatio' differentiates between a good runner and a bad one. As the Z-Value represents the number of standard deviations away from the mean a value lies, a higher positive value will indicate a bad runner and vice versus.

2) Model and Parameters: We used a Linear Regression with regularization in closed form since the matrix $X^TX + \lambda I$ was invertible. Thus, closed form solution will give us the best results as we dont have time constraint. Different values of the compression coefficient was tested using 5-fold cross validation.

D. Cross validation and Prediction

We used 5-Fold cross validation technique to validate our model. For 2017 prediction,we incorporated the data from previous years and recalculated the features, and to increase the odds for finding a lower local optimum this was run multiple times with randomized initial weights.

IV. Results

For both logistic and linear regression we validated all results using k-fold cross validation with k=5, outputting a confusion matrix for each iteration as well as randomization of our initial weights.

λ	Percent Error	Mean Train error	Mean Cross Validation error
0.1	14.6	0.32	0.32
0.3	14.7	0.32	0.33
1.0	14.7	0.32	0.33
3.0	14.7	0.33	0.34
10.0	14.7	0.33	0.34

TABLE III. λ SELECTION

λ	Accuracy	Precision	Recall
0.1	72	73	93
0.3	72	73	93
1.0	72	73	93
3.0	72	73	93
10.0	72	73	93

TABLE IV. Logistic regression λ selection

Regarding prediction of Y2 and the the choice of lambda, our results remain steady. Considering the large dataset with limited variety amongst the features, once normalization was applied, each feature only varied slightly.Another technique which was utilized was varying the degree of each feature, which provided limited change in results. On smaller sets of data this change in degree can result in severe overfitting, but due to many data points any high degree polynomials would exhibit less of this effect. Overall we saw Training error and Validation error extremely close to each other and it may suggest overfitting of our model but repeated modification concluded in similar results.

Now considering our prediction of Y1, we see similar patterns in how change of lambda affected our results. This model was more difficult to implement largely due to the fact that no negative examples were given in the data. We went on to infer the features of those years which a runner did not attend the marathon. The amount of these entries that we insert would have an immediate result skewing our data to favour a

negative or positive prediction. As a consequence, the model favours predicting a positive output for Y1. Modifications of threshold values for the sigmoid function as well as the makeup of entries added could help to sway the preference of the predictions however what would be most beneficial would be to have a better distribution of positive and negative examples directly sampled from the dataset.

Naive Bayes:

Considering all the features mentioned in Features section, Naive Bayes classifier was used to predict the Y1. It was trained by splitting the data up into the aforementioned features, and then calculating the guassian for each respective feature. Table 5 shows the accuracy results for the Naive Bayes classifier.

λ	Error	Precision
Gaussian	8.36381	91.63618

TABLE V. NAIVE BAYES DISTRIBUTION

1) Prediction Result:: The prediction for finishing time was done on the entire data after recalculating all the features

V. DISCUSSION

One of the major errors that may arise from our technique is if we update our features for 2017 and the values of some of the features change drastically, then our model wont be able to predict with high accuracy. The reason is because during learning enough examples of features with new values were not present. The most ideal situation would be if the subset of individuals who ran in 2016 and also had data from previous years was large enough for us to create a good model. The problem is that we dont really have enough data to predict their attendance in 2016, so we are actually overfitting to this group when creating the model, which throws off our predictions. Another problem we had was with the messy data. It was very difficult for us to make the most of this dataset, as the labeling system was so varied. As a result, we are not using a large portion of the data that would be useful for our algorithman example being half marathon times, which could be very useful but we simply did not have the time to parse the many variations of labels indicating half marathon races.

Ultimately, the decision for someone to run a marathon and how they perform are complex questions and although our dataset was relatively large there is always room for uncertainty. In the case for predicting Y2 with more frequent data for any particular runner such as daily/weekly/monthly running would serve to greatly help the accuracy of our models. Nevertheless, our models offer some insight on these challenging questions

Logistic Regression performs moderately with respect to performance in terms of error, precision and recall. But changing the regularization factor has little effect on the decisions and hence little effect on the weight. This means that features have very small say in the outcome. More complex features would be beneficial in achieving conclusive results

For Naive Bayes Multinomial distribution works the best in both the cases.

VI. STATEMENT OF CONTRIBUTIONS

- **Defining the problem:** This was done together in the group through discussion.
- **Developing the Methodology:**This was done together in the group through discussion.
- **Performing the Data Analysis:** Data cleaning was done by primarily by Zachary and was helped in mapping the data by both Shruti and Gavin.
- **Coding the Solution:** Zachary has done a and c part of the code and was helped by Shruti for predictions of Y1 and Y2, Gavin had coded b part. All the stuff was merged by Shruti
- **Writing the Report:** Shruti has made the report in latex, Zachary added the Logistic regression and Linear regression training methods and results.

"We hereby state that all the work presented in this report is that of the authors."

REFERENCES

[1] *https://alexn.org/blog/2012/02/09/howto-build-naive-bayes-classified.html*

[2] *http://machinelearningmastery.com/naive-bayes-classifier-scratch-python/*