Mini-Project #1: Predicting The Winner!

COMP-551: Applied Machine Learning

Aakash Nandi and **Mansha Imtiyaz** and **Malik Altakrori**
{aakash.nandi, mansha.imtiyaz, malik.altakrori}@mail.mcgill.ca
260741007, 260712985, 260605312

1 Introduction

Miami marathon[1] and half marathon races have been taking place every year since their start in 2002. In this paper, we use the data provided for the period from 2002 to 2016 to predict the possibility of participation in 2017 marathon by racers who ran in the previous marathons using Naive Bayes and Logistic Regression. In addition, we provide an estimation of the time that each past participant will take. if they choose to participate in this version using Linear Regression.

2 Problem Representation

In this section, we provide a description of the dataset, and explain the different assumptions that we have used in pre-processing the data, before using it. We start by describing the dataset in Sec. 2.1, followed by the describing the procedure that we used to exclude some of the irrelevant records (Sec. 2.2). Next, we describe the set of features that we have used to predict the participation, and estimate the time in Sec. 3.1 and Sec. 3.2 respectively.

2.1 Dataset Description

The dataset was provided as part of assignment 1, and was collected from Athlinks website [2]. Table 1 shows statistical information about the dataset in general.

Table 1: Dataset: Statistical Information

# Participants	30,417
# Marathon Records	38,805
Avg. # participants / year	2,693.5
Avg. # Races / Participant	1.41

The provided dataset has eight columns: Id, Name [of the participant], Age Category, Sex, Rank [in a race], Time [taken to finish the race in HH:MM:SS format], Pace [also in HH:MM:SS format], and Year. The dataset did not contain any missing values.

For simplicity, we created a MySQL database and inserted all the records in it. This enabled us to get information about the dataset by simply using SQL queries, instead of going over all the records manually.

2.2 Filtering Out Irrelevant Records

Before using the dataset, we had to make decision about irrelevant records that we did not want to include in our experiments. We particularly looked at three issues: (1) the year that has information for the half-marathon race; (2) runners who had multiple records for the same year (i.e. appear twice for the same year) and finally, (3) runners who did not provide their identity information, and therefore appeared as "private" in the dataset.

2.2.1 Half-Marathon Race

There were records which gave time for half marathon instead of full marathons. To identify the year, we used an SQL query[3] to calculate the average time taken by the top 10 runners in each year. The year that had the lowest average was identified

[1] http://www.themiamimarathon.com/course/

[2] https://www.athlinks.com/event/3294

[3] The query is provided in the appendix.

as the year for the half-marathon. We changed the value for half marathon to full marathon but the inclusion of these records weren't much useful when analyzing on training data. So, all the records for this specific year were excluded from our predictions.

2.2.2 Multiple Records For the Same Runner

Upon further inspection, we identified multiple runners who had more than one entry record for the same year. Because a specific runner should have only one result per year, and since we could not identify which record is the correct one for that specific year, we identified these records, and deleted all the records for these runners for the purpose of training. The outcome of this step is deleting 310 race entries. For 2017 prediction, we average the values for the multiple records for a single ID.

2.2.3 Records for Anonymous Runners

The last case we faced was for anonymous runners. Anonymous runners are those who decided to keep their information, such as the name, and sometime sex, hidden. We could not use these records because we were considering all the records for the participant, to extract one set of features, as described below in Sec 3.1. In other words, we could not tell if one set of records, despite having the same Sex and Age category, belongs to the same runner or multiple runners who happen to share the same gender and age category. Therefore, all 301 records were ignored during training.

3 Feature Representation

3.1 Y1 - Predicting Who Will Participate in 2017

In this task, we are trying to predict who, from the runners in previous years, is going to participate in the 2017 version of the Miami Marathon. To do that, we were asked to use two classification methods: Logistic Regression, and Naïve Bayes.

For this task, we tried three different feature sets. Below are the sets of features, and the motivation behind choosing each one of them.

3.1.1 Set1 - Raw Dataset columns

This set of features simply contains the columns of the dataset that was provided for us. We removed the IDs and the names of the participants, and kept the age, the gender, the rank, and the pace. The goal of using this set of features is to establish a baseline before adding our own new features.

3.1.2 Set2 - Training for 2016

This dataset contains features that describe the performance of a runner in all the past years prior to 2016. Based on that, we excluded the records of those who only participated in 2016, and never before. We added new features in addition to the previous features as per Table 2. The motivation behind using this set of features, is to train the model on predicting who will participate in 2016 based on their previous records in earlier years. This means we had two types of runners: those who participated in earlier races AND participated in 2016, and those who participated in earlier races but NOT in 2016. As mentioned earlier, those who participated only in 2016 were used as an external test set.

Table 2: List of Features for Classification

Feature Name	Description	Feature Category
TOTAL_MARA	Total number of Marathons participated in	Continuous
GENDER	Encoded as per Table 8	Binomial
AGE	Mean of Age Range	Continuous
RANK	Mean of Rank for Marathons participated in	Continuous
PACE	Mean of Pace for Marathons participated in	Continuous
2015	Participation in 2015 Miami Marathon, Yes = 1 No = 0	Binomial
2016	Participation in 2016 Miami Marathon, Yes = 1 No = 0	Binomial

3.1.3 Set3 - Incorporating the Weather Data

In features Set3, we have incorporated the weather data[4], particularly: the temperature [for the race day], the wind speed, the precipitation, and the humidity for all the years from 2002 till 2016. We used athlinks website to get the exact date on which previous races took place, and used the date to retrieve the weather informaiton for each day. For 2017, we used the average monthly expectations assuming that the participants will use these prediction to decide if they are going to participate or not. The reason behind this is that since the Marathon is an outdoor event, we assumed that the weather information could affect the participants decision.

[4] https://www.wunderground.com/

3.2 Y2 - Estimating the Finish Time

For estimating the time, we emphasize on the time taken by the participant to complete the Marathon. Table 3 describes what features have been used for regression.

Table 3: List of Features for Regression

Feature Name	Description	Feature Category
TOTAL_MARA	Total number of Marathons participated in	Continuous
GENDER	Encoded as per Table 8	Binomial
AGE	Mean of Age Range	Continuous
RANK	Mean of Rank for Marathons participated in	Continuous
PACE	Mean of Pace for Marathons participated in	Continuous
Time	Mean of Time for Marathons participated in	Continuous
2016	Participation in 2016 Miami Marathon, Yes = 1 No = 0	Binomial

4 Training Method

In this section, we describe the training methods and design decisions that we used in our experiments.

4.1 Linear Regression

We developed a real time graph plotting function which depicted cost vs iterations, using which we we were able to determine the suitable value of alpha. After several attempts to reach the approximate minima in comfortable time span, we zeroed on to an alpha value of 0.01 and its graph is depicted in Figure 1.

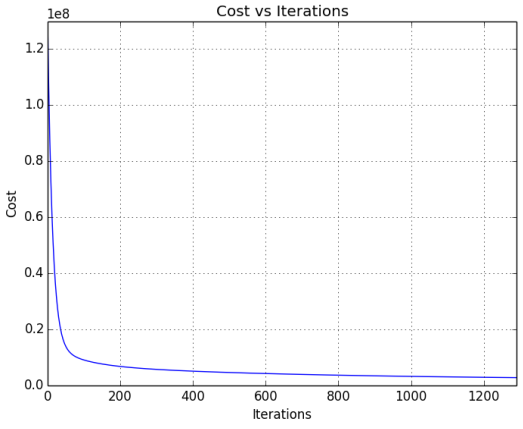

Figure 1: Cost vs Iteration for Linear Regression

4.2 Naive Bayes

For the Naive Bayes Classifier, the whole data set was divided into training, testing and validation set. Different features were tried and tested, and threshold was varied till results obtained were satisfactory. Each of the last 15 years event participation data as a single feature was tested on the Naive Bayes Classifier and the most accurate results, shown in Table 5.2 were given by last two years participation data. And thus, these features were chosen for the model. Difference in feature set of Naive Bayes and Logistic Regression is further explained in the Discussion section.

Among the features selected, half are continuous variables. In this case, basic NaiveBayes, which focuses on just binary values would not perform as expected. So, in order to account for these continuous variables, we used Gaussian distribution in the classifier to account for the features which are not binomial.

Gaussian Naive Bayes assumes that given the output, input continuous variables are independent of each other and each obey different normal distributions. The probability density function of a normal or Gaussian distribution of continuous variable x, is given by

$$f(x) = e^{-(x-\mu)^2/2\sigma^2}/\sigma\sqrt{2\pi} \quad (1)$$

where μ denotes the mean and σ is the standard deviation. These parameters can be computed from the dataset likelihood. We used log probabilities to avoid errors caused due to floating point numbers. This also helped as some probabilities were small and we didn't want the probabilities to vanish to zero.

4.3 Logistic Regression

We used our real time graph plotting function to determine the alpha and finalized it to be 1. The graph that we obtained while training it on 2016 training data set is depicted in Figure 2.

We performed a 5 fold-cross validation to pick a regularization parameter λ from a set of [0,0.1,1.0,10,100]. However we did not find significant differences in the average accuracy over training and validation set to claim that regularization was solving a high variance problem as accuracy for

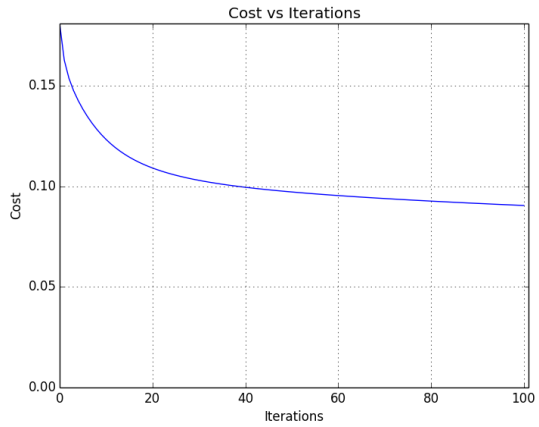

Figure 2: Cost vs Iteration for Logistic Regression

both was around 87%, which is less than our baseline accuracy of 90%.

5 Results

In this section we discuss the experimental results of our experiments[5]. To evaluate the results, 90% of samples (80% for train and 10% for testing) from the original dataset was used to train the model and the rest as a validation set.

5.1 Linear Regression

By using the first set of features, we were able to use the closed form solution to train a model. This was possible as the experiments were done on a machine with high computation power. However, upon normalizing the training data, and including more features -which had the value zero for many rows- the resulting matrix of features and records became singular, and therefore we used gradient descent.

We obtained an accuracy of 968 seconds which is approximately 16 minutes and 8 seconds on the training dataset. However our model managed to obtain an accuracy of 3086 seconds which is approximately 51 minutes and 26 seconds. The cost function along with the number of iterations has been plotted in Figure 1.

[5]The experiments were conducted using a workstation running Windows 7 (64-bit) with an Intel® Core(™) i7-4700HQ CPU @ 2.40 GHz (8 CPUs) processor and 16GB RAM.

5.2 Naive Bayes

The training set accuracy for Naive Bayes was 95.28%, and 95.05% for the validation set. The results were also compared with the scikit learn's Gaussian Naive Bayes classifier. The detailed metrics are:

Table 4: Accuracy, Precision and Recall for Naive Bayes

NaiveBayes	Training	Validation	GaussianNB
Accuracy	95.28	95.05	95.6
Precision	31.22	25.74	25.6
Recall	41.4	38.8	39.36
F1 Measure	35.6	30.9	36.45

These metrics were obtained by tuning the threshold (for setting boundary to decide between one of the two classes) on the training set and validated using the validation set. Our baseline accuracy(if we predict all 0's) is 90%, and our model did better than the baseline.

Table 5: Confusion Matrix for Naive Bayes

Training Stage		Validation Stage	
TP = 39	FP = 86	TP = 26	FP = 75
FN = 55	TN = 2629	FN = 41	TN = 2688

Due to class imbalance in the original data, we were initially getting more 0s in the data than expected, which was balanced by increasing the threshold. Table 5 shows the confusion matrix for both training and validation stage.

The ROC curve for Naive Bayes based on these results his shown in Figure 3.

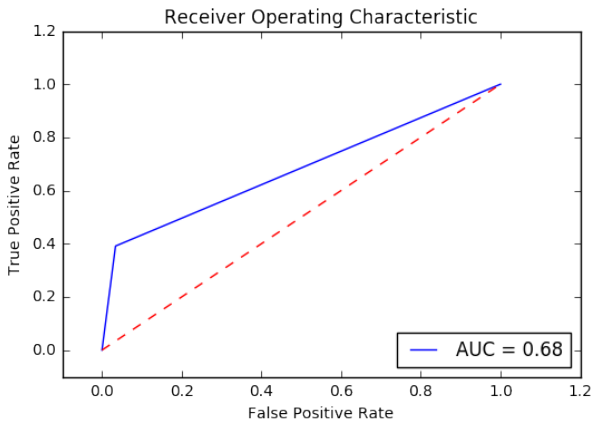

Figure 3: Naive Bayes ROC

5.3 Logistic Regression

Naive Bayes weights each feature equally as each feature is assumed to be independent and used to calculated the output probability. Therefore, if we use a lot of unimportant features, accuracy will be lowered. On the other hand, Logistic regression assigns different weights to the features. Even if the features are repetitive or highly correlated, logistic regression will compensate by assigning/varying the weights. This can (supposedly) give better accuracy for Y1, as we can use many features and by varying the weights, their correlation can be accounted for.

The training set accuracy come to 93.12% and 93.05% for validation set. Metrics for logistic regression are given in Table 5.3

Table 6: Accuracy, Precision, Recall for Logistic Regression

LogisticRegression	Training	Validation
Accuracy	93.12	93.05
Precision	64.47	60.74
Recall	20.16	24.8
F1 Measure	30.72	32.9

As seen in the case of Naive Bayes, we got more false negatives than false positives, see Table 5. This might be due to the reason of imbalance in classes in the original dataset. To evaluate the classifier, we also varied the decision boundaries and plotted the ROC curve for Logistic regression as shown in Figure 4.

Table 7: Confusion Matrix for Logistic Regression

Training Stage		Validation Stage	
TP = 49	FP = 27	TP = 44	FP = 32
FN = 194	TN = 2943	FN = 207	TN = 2930

6 Discussion

Prediction Using the given Naive Bayes algorithm, out of 30417 participants, 1290 have been predicted to participate in the 2017 Miami Marathon (4 percent turnout). See the output in the accompanied predictions.csv file. Surprisingly, the same participants have been predicted to participate using the NB classifier from the Scikit Learn library.

Limitations of our approach Due to time restriction, exhaustive search for feature sets could not be performed. As a result we were able to provide our inferences and observations just for two different feature sets.

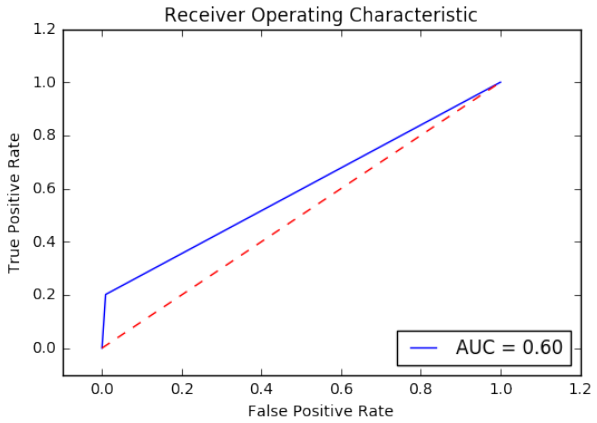

Figure 4: Logistic Regression ROC

Each feature could have been selected on the basis of how it was contributing to the overall performance metrics and then incorporated into the feature set.

We tried to incorporate the weather data into the feature set to account for performance of athletes and their likelihood to participate in that particular marathon. However we found that we were unable to train such a model and obtain fruitful results due to the range of alpha values that we were testing for. For small alpha values, the improvement in cost was really low and for higher alpha value there was an increase in the cost . Hence we had to drop the feature. Model would have performed more accurately if we had a better sense of which features directly or indirectly affect the participation in Marathon.

7 Statement of Contributions

All team members discussed and had equal contribution towards feature engineering. Data cleansing and formatting was done in SQL by Malik and using pandas by Mansha. Malik worked on coding Linear Regression, Aakash worked on Logistic Regression and Naive Bayes was coded by Mansha. Everyone wrote about their findings separately and collaborated equally on report writing. **We hereby state that all the work presented in this report is that of the authors**.

Appendix

Datasets are available via: https://drive.google.com/drive/folders/0B2abXowK4mLwbEQwSGs2N0JYYk0?usp=sharing

SQL query to sort the years based on the average time for the top 10 runners:
"Select Year, avg(Time) as av from $project1\_data$ where Rank $<= 10$ group by Year order by av;"

Table 8: Set1- Dataset columns

Feature	Type	Values
Gender	Categorical	0 (Male) / 1 (Female)
Age	Continuous	0 - 90
Rank	Ordinal	1-3913
Pace	Continuous	303-115