**ORIE 5741: Learning With Big Messy Data**

# Predicting Marathon Finish Times

**Julia Allen and Hugo Mainguy**

**Date:**
5/12/2023

# ORIE 5741 Final Project - Predicting Marathon Finish Times

Julia Allen, Hugo Mainguy

May 12, 2023

## Contents

# 1 Introduction

Running a marathon is a huge accomplishment for anyone. The race is the end result of months of training, and runners want to be able to set realistic goals for themselves. In addition, the families and friends of these marathon runners often come out in droves to cheer for their loved one, and want to be able to plan their spectatorship based on a prediction for when their runner will arrive.

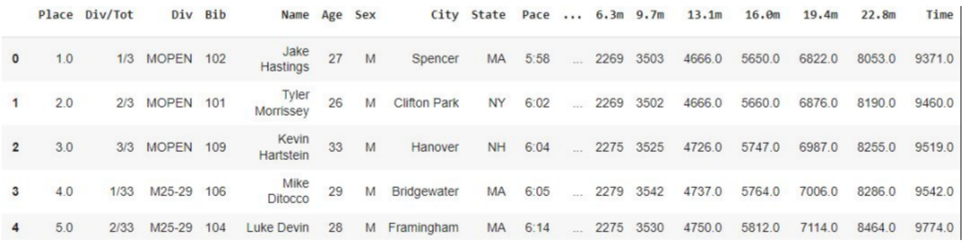
In this analysis, we first train a model to allow a spectator to receive accurate split predictions for a given runner based only on the splits they have run so far. Using Ordinary Least Squares regression, we train this model and show that it is far more effective than ”baseline” models used by races today.

Then, we use two different methods to predict if a runner will qualify for the prestigious Boston Marathon. We attempt to make these predictions using logistic regression and a decision tree to help a runner figure out how achievable this goal is.

Runners love data, and allowing them valuable data insights can be of great financial benefit to race organizers as a draw for their events. These two predictions have the potential to greatly increase the satisfaction of both runners and spectators, making participants more likely to come back to their race year after year.

# 2 Dataset

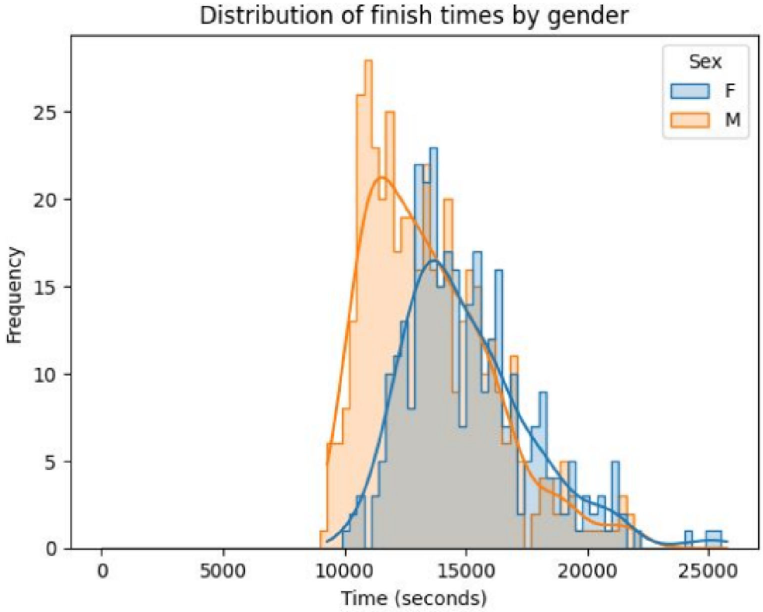
We obtained a dataset from the Cheap Marathon, a race run annually in Derry, New Hampshire. We received data from 2022 and 2023, including a total of 775 runs. We chose this dataset for a few reasons, most importantly due to one of the team members having run this marathon in 2023 and therefore knowing the race terrain well, and the fact that the data has a total of eight split times that were relatively easy to combine into a dataframe and provided a larger quantity of data in order to predict finish times based on split times. Furthermore, since the course is a double out and back (i.e. runners run on a trail, turn around, then do this again; this also means that the $n^{\text{th}}$ and $(n+4)^{\text{th}}$ split are for the same portion of the course), and is almost completely flat, which means that the analysis is less dependent on external factors. Each row has data on the runner - most importantly age, gender, and eight split times, including finish time. In order to conduct most of our analyses, we convert all the times to floats, specifically the number of seconds. This is what the first rows of the dataset look like:



| | Place | Div/Tot | Div | Bib | Name | Age | Sex | City | State | Pace | ... | 6.3m | 9.7m | 13.1m | 16.0m | 19.4m | 22.8m | Time |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1.0 | 1/3 | MOPEN | 102 | Jake Hastings | 27 | M | Spencer | MA | 5:58 | ... | 2269 | 3503 | 4666.0 | 5650.0 | 6822.0 | 8053.0 | 9371.0 |
| 1 | 2.0 | 2/3 | MOPEN | 101 | Tyler Morrissey | 26 | M | Clifton Park | NY | 6:02 | ... | 2269 | 3502 | 4666.0 | 5660.0 | 6876.0 | 8190.0 | 9460.0 |
| 2 | 3.0 | 3/3 | MOPEN | 109 | Kevin Hartstein | 33 | M | Hanover | NH | 6:04 | ... | 2275 | 3525 | 4726.0 | 5747.0 | 6987.0 | 8255.0 | 9519.0 |
| 3 | 4.0 | 1/33 | M25-29 | 106 | Mike Ditocco | 29 | M | Bridgewater | MA | 6:05 | ... | 2279 | 3542 | 4737.0 | 5764.0 | 7006.0 | 8286.0 | 9542.0 |
| 4 | 5.0 | 2/33 | M25-29 | 104 | Luke Devin | 28 | M | Framingham | MA | 6:14 | ... | 2275 | 3530 | 4750.0 | 5812.0 | 7114.0 | 8464.0 | 9774.0 |

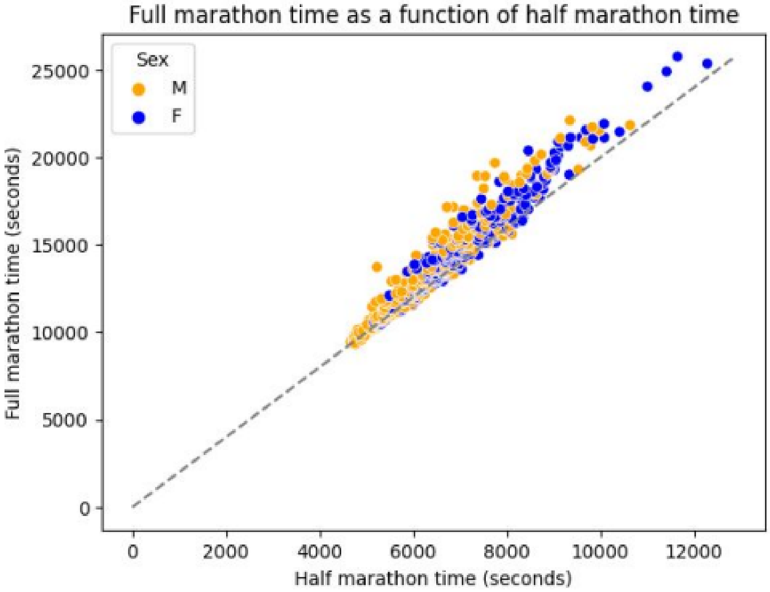
# 3 Exploratory Data Analysis

We proceed to conduct some exploratory data analysis with the dataset, in order to verify the quality of the data and draw some preliminary conclusions. We first start by creating a graph of the finish times by gender in five minute increments as a histogram, while also fitting a curve for a density function.

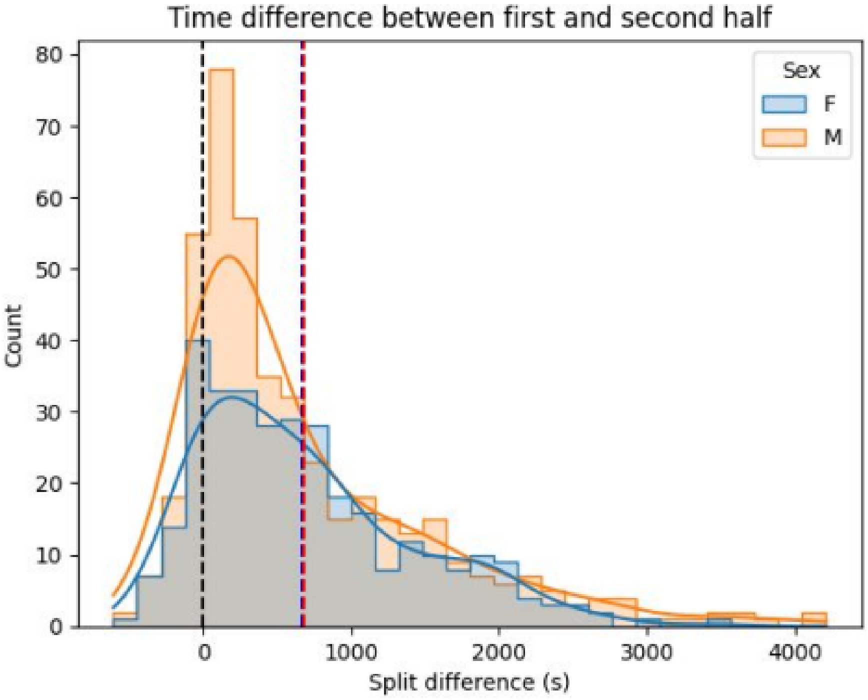


We observe that the mode of finishing times is just above 3 hours (10800 seconds) for men, whereas it is just under 4 hours (14400 seconds) for both women and overall. There are more men who finish in under three hours and a half (12600 seconds); but beyond that point, genders are relatively even. The distributions have different parameters, but both share uneven tails - specifically, the right tail is longer, meaning that there are many more runners finishing say over an hour later than the mode finish time than over an hour before. This is to be expected, and often happens in fairly competitive sport settings.

Thanks to having a lot of data from split times, we were then curious to know how much faster or slower runners are between the first half and the second half, especially since they are identical.



In this graph, we also draw the line corresponding to an even split (the first and second half being run in the same amount of time) to visually see whether runners slow down throughout the race. The answer is that most do; there are many runners who slow down a lot, but none who win more than a few minutes, and those tend to be concentrated across faster runners, who generally run more consistent splits based on this graph. There are two main trends to observe, namely that most runners slow down, and that the effect, as well as the variance, tends to grow for slower runners. We can also notice that it seems that at equivalent times, especially for slower runners, women tend to run more evenly than men, and our next graph provides some insight into this observation.



This time, we graph the number of runners with a given split difference in five minute intervals, with again a line showing where the even split is, and two more showing the mean split difference for both men and women. The two are in fact very similar, around 13 minutes (780 seconds), suggesting that the difference is statistically insignificant. The curves are very similar, and the fact that there are many men who have a very small positive split (the second half being slower by the first half by only a few minutes) is countered by a small bump of very large positive splits existing mostly for men only. Therefore, while it is clear that most people positive split, there actually is not a very significant difference by gender, though the curves do slightly differ.

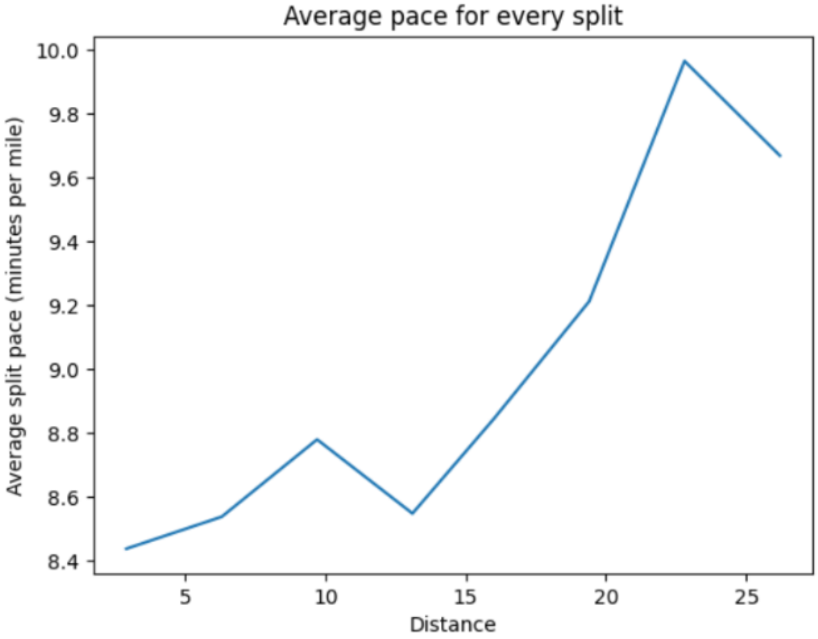
## 4 Predicting Finishing Times

Many larger marathons offer live participant tracking, where spectators can see what time their runners passed through checkpoints and see predictions for when they will arrive at future ones. Many races worldwide attempt to provide these predictions, but they are often based on naive approaches and leave runners and spectators alike frustrated with the inaccurate estimates. In this section, we attempt to train a predictor using OLS regression and compare this to these baseline approaches.

We compare our method to two "baseline" methods. In Baseline 1, the finish time is predicted based on the runner's overall average pace so far. So, if a runner has been averaging 8 minutes per mile at the halfway point, the model will predict the runner will finish the race at this 8 minute pace for a finish time of 3:30.

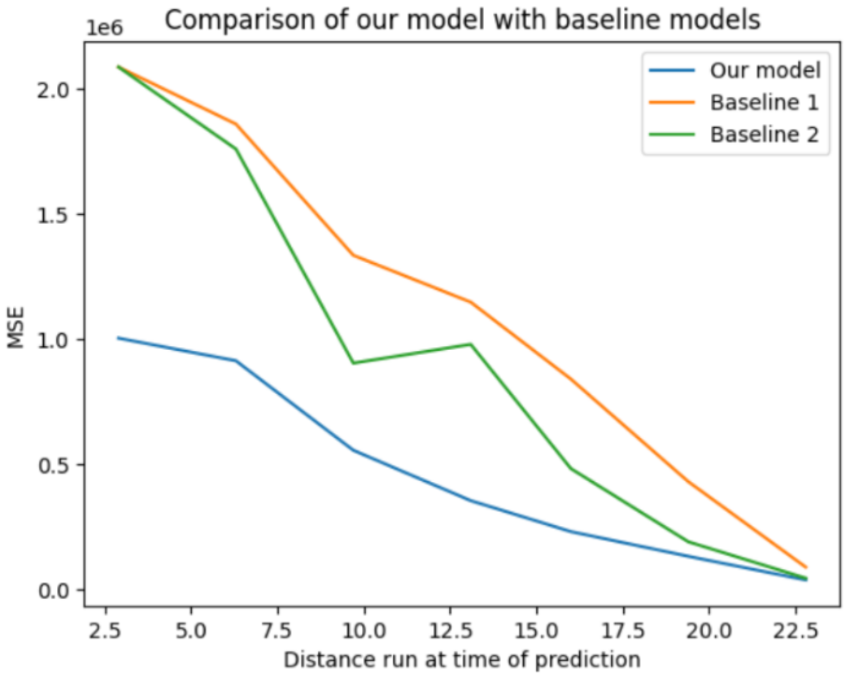
In Baseline 2, the finish time is predicted based only on the most recent split. For example, if a runner has been averaging an 8 minute pace but ran the last split at a 9 minute pace, the model will predict the runner will run the rest of the race at this 9 minute pace.

The main reason both of these methods fail is because most runners run positive splits, meaning they get slower as the race goes on. Baseline 1 assumes the end of the race will be just as fast as the beginning, and although Baseline 2 is slightly better in that it acknowledges a runner's recent pace is likely a better reflection of their future pace than their starting pace, it still does not reflect the knowledge that most runners will slow down more. The below graph shows the average pace per mile vs the distance completed so far for all runners in the race:



We observe that runners tend to get slower with every split, with the exception of the end, where they may speed up for a "final kick", and for the split ending at 13.1 miles. Based on a realization that almost every runner ran this middle split faster than the previous one, but this trend was not observed for any other race, we believe that the timing mat may have been misplaced for this split. However, since it is an error consistent across all results, it does not affect the accuracy of our model.

To do the prediction, we train an OLS model on every subset of "splits so far"- in other words, only the first split, then only the first two splits, then the first three, etc. This reflects that spectators may want to predict their runner's finish time at any point during the race based only on the splits they have run so far. We divide the data into a training set and testing set, and use Mean Squared Error to measure loss. To compare with the two baselines, we use the same test set for all three methods. However, the baselines use the methods described above, whereas our model uses insights gleaned from the training set to make more accurate predictions. In practice, the training set would be all past race results ad the "test set" would be runners currently racing. Below is a graph comparing the MSE for our model with both baselines:



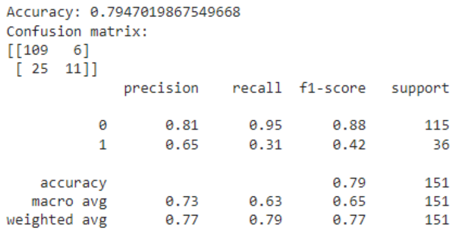
We observe that our model performs better than both baselines at every single split. For the reasons described above, Baseline 2 is better than Baseline 1 (but affected more by the misplaced timing mat since the final predictions are being based only off an inaccurate split pace). However, our model still outperforms both of them, showing that using even a fairly basic machine learning model could allow race organizers to give much more accurate finish estimates.

Future exploration could include incorporating age and gender into the model, since perhaps people in different demographic groups pace themselves differently. It would also be interesting if we had access to data on how many marathons someone had run before, since a more experienced marathoner might be less likely to see large positive splits. Finally, we want to extend this model to other races, in particular to ones with fewer split times or a hillier course, to see how well this model performs on other races.

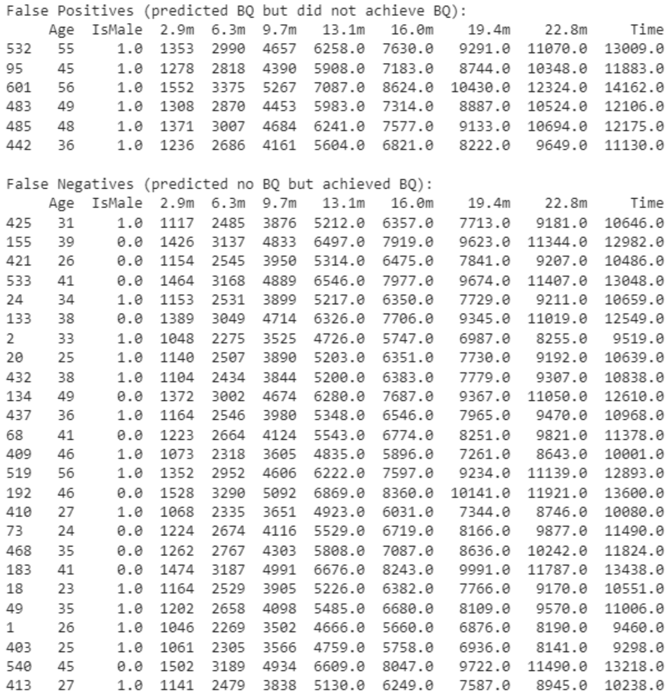
# 5 Boston Qualifier Predictor

## 5.1 Logistic Regression

Running a Boston Qualifying time, one that is required for entry into the prestigious Boston Marathon, is a big goal for many serious amateur runners. The threshold depends on the age and gender of the runner: for men and women between the ages of 18 and 34, it is set at 3:00 and 3:30, and becomes longer for older runners. A full list is available here: `https://www.baa.org/races/boston-marathon/qualify`. Here, we will be trying to predict whether someone can run a qualifying time for the Boston Marathon based on their previous split times. This can be valuable information for runners who want to know whether their goal is achievable at their current expected pace for the rest of the race. In order to predict this, we run a logistic regression model, giving as input age, gender, and a list of split times, and as outcome a 0-1 variable that is equal to 1 if and only if the runner ended up achieving a Boston Qualifier. We split the data into a training and testing set, with the training set using 80% of the data. Here are some figures obtained on the testing set:



```
Accuracy: 0.7947019867549668
Confusion matrix:
[[109   6]
 [ 25  11]]
              precision    recall  f1-score   support

           0       0.81      0.95      0.88       115
           1       0.65      0.31      0.42        36

    accuracy                           0.79       151
   macro avg       0.73      0.63      0.65       151
weighted avg       0.77      0.79      0.77       151
```



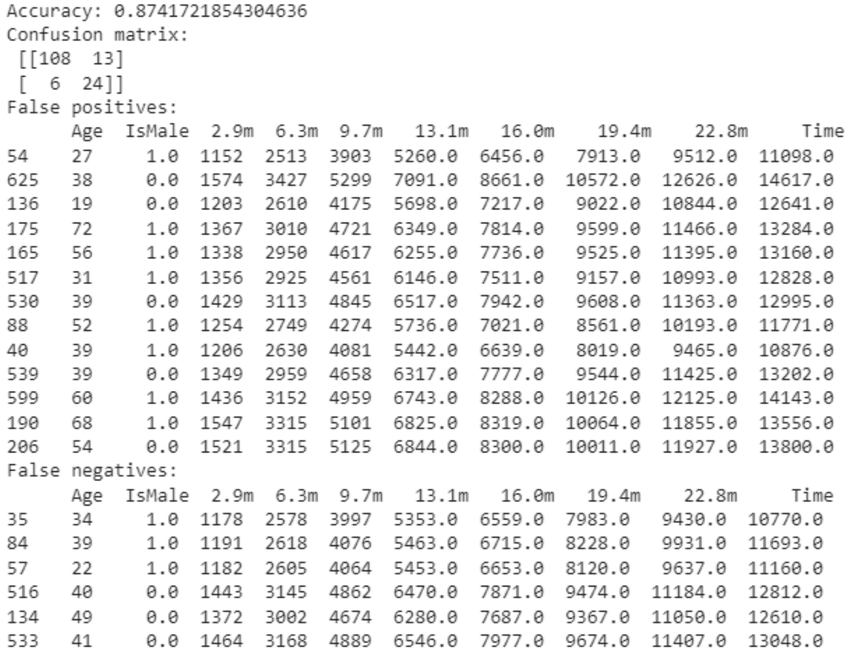
```
False Positives (predicted BQ but did not achieve BQ):
     Age  IsMale  2.9m  6.3m  9.7m   13.1m   16.0m    19.4m    22.8m     Time
532   55     1.0  1353  2990  4657  6258.0  7630.0   9291.0  11070.0  13009.0
95    45     1.0  1278  2818  4390  5908.0  7183.0   8744.0  10348.0  11883.0
601   56     1.0  1552  3375  5267  7087.0  8624.0  10430.0  12324.0  14162.0
483   49     1.0  1308  2870  4453  5983.0  7314.0   8887.0  10524.0  12106.0
485   48     1.0  1371  3007  4684  6241.0  7577.0   9133.0  10694.0  12175.0
442   36     1.0  1236  2686  4161  5604.0  6821.0   8222.0   9649.0  11130.0

False Negatives (predicted no BQ but achieved BQ):
     Age  IsMale  2.9m  6.3m  9.7m   13.1m   16.0m    19.4m    22.8m     Time
425   31     1.0  1117  2485  3876  5212.0  6357.0   7713.0   9181.0  10646.0
155   39     0.0  1426  3137  4833  6497.0  7919.0   9623.0  11344.0  12982.0
421   26     0.0  1154  2545  3950  5314.0  6475.0   7841.0   9207.0  10486.0
533   41     0.0  1464  3168  4889  6546.0  7977.0   9674.0  11407.0  13048.0
24    34     1.0  1153  2531  3899  5217.0  6350.0   7729.0   9211.0  10659.0
133   38     0.0  1389  3049  4714  6326.0  7706.0   9345.0  11019.0  12549.0
2     33     1.0  1048  2275  3525  4726.0  5747.0   6987.0   8255.0   9519.0
20    25     1.0  1140  2507  3890  5203.0  6351.0   7730.0   9192.0  10639.0
432   38     1.0  1104  2434  3844  5200.0  6383.0   7779.0   9307.0  10838.0
134   49     0.0  1372  3002  4674  6280.0  7687.0   9367.0  11050.0  12610.0
437   36     1.0  1164  2546  3980  5348.0  6546.0   7965.0   9470.0  10968.0
68    41     0.0  1223  2664  4124  5543.0  6774.0   8251.0   9821.0  11378.0
409   46     1.0  1073  2318  3605  4835.0  5896.0   7261.0   8643.0  10001.0
519   56     1.0  1352  2952  4606  6222.0  7597.0   9234.0  11139.0  12893.0
192   46     0.0  1528  3290  5092  6869.0  8360.0  10141.0  11921.0  13600.0
410   27     1.0  1068  2335  3651  4923.0  6031.0   7344.0   8746.0  10080.0
73    24     0.0  1224  2674  4116  5529.0  6719.0   8166.0   9877.0  11490.0
468   35     0.0  1262  2767  4303  5808.0  7087.0   8636.0  10242.0  11824.0
183   41     0.0  1474  3187  4991  6676.0  8243.0   9991.0  11787.0  13438.0
18    23     1.0  1164  2529  3905  5226.0  6382.0   7766.0   9170.0  10551.0
49    35     1.0  1202  2658  4098  5485.0  6680.0   8109.0   9570.0  11006.0
1     26     1.0  1046  2269  3502  4666.0  5660.0   6876.0   8190.0   9460.0
403   25     1.0  1061  2305  3566  4759.0  5758.0   6936.0   8141.0   9298.0
540   45     0.0  1502  3189  4934  6609.0  8047.0   9722.0  11490.0  13218.0
413   27     1.0  1141  2479  3838  5130.0  6249.0   7587.0   8945.0  10238.0
```

Incredibly enough, even when given the final time in the input (but no information about what a Boston Qualifier is), the algorithm only had an accuracy between 75 and 80%! We see that it is quite good at classifying runners that are quite behind the time threshold, and relatively good at classifying runners well ahead. A better glance at the false positives and false negatives reveals some more information. Specifically, the false positives tend to be older people, and often (though not always, depending on the simulation) men. While runner 442 was off by only 30 seconds, runner 601 was off by 21 minutes. The false negatives, on the other hand, tend to skew younger and slightly women, but that is less clear. Incredibly enough, in this simulation, both winners (1 and 403) were in the testing dataset and were both false negatives, despite clearing the bar by about 25 minutes!

Clearly, this model is not very reliable. Even when removing all the intermediate split times, the model is about as accurate. While the data is separable (at least for the finish times), it is not linearly separable, as age is divided into categories, and threshold increases are not consistent for every five year period. Therefore, one should probably figure out a way to let the classifier know about those thresholds, in one way or another. This could possibly be done with a decision tree, which would be able to find a better boundary - if the final time is removed, along with more of the later splits, the accuracy will decrease, but we would hope that with most split times, the accuracy could still somewhat increase.

## 5.2 Decision Tree

We train a decision tree on the data, obtaining the following results:



```
Accuracy: 0.8741721854304636
Confusion matrix:
 [[108  13]
 [  6  24]]
False positives:
      Age  IsMale  2.9m  6.3m  9.7m   13.1m   16.0m    19.4m    22.8m     Time
54    27     1.0  1152  2513  3903  5260.0  6456.0   7913.0   9512.0  11098.0
625   38     0.0  1574  3427  5299  7091.0  8661.0  10572.0  12626.0  14617.0
136   19     0.0  1203  2610  4175  5698.0  7217.0   9022.0  10844.0  12641.0
175   72     1.0  1367  3010  4721  6349.0  7814.0   9599.0  11466.0  13284.0
165   56     1.0  1338  2950  4617  6255.0  7736.0   9525.0  11395.0  13160.0
517   31     1.0  1356  2925  4561  6146.0  7511.0   9157.0  10993.0  12828.0
530   39     0.0  1429  3113  4845  6517.0  7942.0   9608.0  11363.0  12995.0
88    52     1.0  1254  2749  4274  5736.0  7021.0   8561.0  10193.0  11771.0
40    39     1.0  1206  2630  4081  5442.0  6639.0   8019.0   9465.0  10876.0
539   39     0.0  1349  2959  4658  6317.0  7777.0   9544.0  11425.0  13202.0
599   60     1.0  1436  3152  4959  6743.0  8288.0  10126.0  12125.0  14143.0
190   68     1.0  1547  3315  5101  6825.0  8319.0  10064.0  11855.0  13556.0
206   54     0.0  1521  3315  5125  6844.0  8300.0  10011.0  11927.0  13800.0
False negatives:
      Age  IsMale  2.9m  6.3m  9.7m   13.1m   16.0m   19.4m    22.8m     Time
35    34     1.0  1178  2578  3997  5353.0  6559.0  7983.0   9430.0  10770.0
84    39     1.0  1191  2618  4076  5463.0  6715.0  8228.0   9931.0  11693.0
57    22     1.0  1182  2605  4064  5453.0  6653.0  8120.0   9637.0  11160.0
516   40     0.0  1443  3145  4862  6470.0  7871.0  9474.0  11184.0  12812.0
134   49     0.0  1372  3002  4674  6280.0  7687.0  9367.0  11050.0  12610.0
533   41     0.0  1464  3168  4889  6546.0  7977.0  9674.0  11407.0  13048.0
```

This result is much better, giving us 87% accuracy. It is harder to determine something connecting all the false positives or false negatives, which are fewer and farther between. This includes the finish time - if we progressively remove split time, the accuracy drops, though not much - it remains in the low eighties and high seventies even all the way down to only the first split given, at 2.9 miles. In other words, seeing how someone runs the first eighth of the race is enough to determine with rather high accuracy whether they will hit a given threshold, and is nearly as good for our classifier as having all the data. This also confirms that for a rather complex (though piecewise linear for the finish time) decision bound, a decision tree proves better than a logistic regression.

# 6 Ethics and Equity

With any machine learning model, equity will be a concern. Since models are trained on the data available, minority groups for whom there is less data available may see less accurate predictions.

In our dataset, as shown in Section 3, the data is fairly right-skewed, meaning the slowest runners have very few other participants at their pace. This means they may not see finish times that are as accurate as those for faster runners, making it harder for their families to cheer them on and giving them a harder time figuring out if they are likely to BQ.

Similarly, there are more men in the race than women, and more younger runners than older runners. This means women and older runners are more likely to see inaccurate results.

In order to fix these issues, in the future, we would want to have a much larger dataset to train the model on, and make especially sure we had more data for slower, older, and female runners to ensure everyone has an opportunity to receive accurate race predictions.

# 7 Conclusion

Through this project, we have obtained a strong predictor of finish times based on intermediate splits, which is better than most methods used. Furthermore, since it even works well with data from multiple years, this means that one does not need to obtain the current year's data before making live predictions. We also designed a model to determine whether someone will be able to qualify for the Boston Marathon with relatively high accuracy, though we also explore the flaws of this method along with potential ways to improve it. Finally, the dataset is relatively uneven - though that is a problem with all similar datasets - in that some categories, in particular slower and older runners, and to some extent women, tend to be underrepresented, which can lead to imperfect predictions for them - accuracy could be improved if there was more data representing a broader range of people.

This analysis could be incorporated by race organizers in order to greatly improve the spectator experience and give runners a better sense of their performance. If people feel like they are able to set realistic goals and meet them, they will have a higher opinion of the race and be more likely to sign up for it again in the future (and recommend it to their friends). Additionally, if the friends and family of the runners are able to figure out their arrival times accurately, not only will they have a better experience as spectators but also the runner will have a more positive race day experience. Both of these improvements can improve the reputation of a race, helping it attract more runners and sponsors in the future.

# 8 Contributions

The two members of this project group had approximately equal contributions. Julia wrote the code and report for the ”Predicting Finishing Times” analysis, and Hugo wrote the code and report for the ”Boston Qualifier Predictor” section. The group members met together to research datasets and get and clean data- Julia cleaned the data while Hugo found the initial dataset and did the EDA. The group members then made the presentation and wrote the report together, with Julia also writing the Intro and Equity sections and Hugo also writing the Conclusion.