From Pace to Place: Can Marathon Finishing Times Predict Regional Origins?

Isabella McClure

under the supervision of Dr. Haim Bar

and advising of Dr. Elizabeth Schifano

Honors Thesis prepared for

The University of Connecticut Honors Program

April 26, 2024

University of Connecticut, Department of Statistics

Storrs, CT

Abstract

In this research endeavor, the capabilities of neural networks are utilized to discern regional patterns within a diverse populace via marathon finishing times. Leveraging two decades of race data from the New York and Boston Marathons, the study aims to develop a predictive model capable of identifying individuals' geographic origins based on their marathon performances. This exploration holds promise in shedding light on the intersection of environmental variables and ultimate race achievements in the realm of marathon running. Preliminary results suggest that finishing times alone introduce an excess of randomness that is difficult for the algorithm to untangle, resulting in subpar prediction accuracies. Without clear explanation, certain regions are much more predictive for each of the two major marathons studied. Suggestions for improvement in future works are introducing new, personal data in addition to the finishing time to provide more information to train the models.

Keywords: Neural networks, Running predictions, Sport statistics

1 Introduction

The sport of running dates back to the early existence of homo sapiens themselves. Rooted in necessity during prehistoric times, our ancestors relied on running as a means of hunting and traversing vast areas. Its evolution occurred in ancient civilization, with the Greeks immortalizing it in the sacred precincts of Olympia during the first Olympic Games; however, the marathon, perhaps one of the most iconic manifestations of human endurance, found its genesis in the legendary tale of Pheidippides, a herald of the Battle of Marathon in 490 BCE. His fabled run from the battlefield to Athens, spanning approximately 26 miles, marked the birth of the modern marathon as we know today. Marathon racing has now grown to be an extremely popular event outside of just the Olympics. When looking solely at the six Abbott Major World Marathons – Tokyo Marathon, Boston Marathon, London Marathon, Berlin Marathon, Chicago Marathon, and New York City Marathon – over a quarter of a million people collectively took part in these events around the globe in 2023 alone. The largest of these events, the New York City Marathon (NYC Marathon), has seen a particularly impressive increase in participation since its origin, even becoming the third largest marathon in history in 2019 with 53,520 finishers. Though not the largest of the majors, the Boston Marathon being one of the oldest annual marathon races in the world has also emerged as a prestigious event. Now regarded as the "Holy Grail" of marathon racing, athletes devote years of time and thousands of dollars to qualifying for the Boston Marathon as that is now the only way to ensure participation.

With the increase in popularity came the need for entry restrictions to address excess interest. Many of the largest marathons resorted to time qualifications for guaranteed entry, forcing runners to really devote themselves to the training beforehand; this then brought about a need for optimization in performance. What is the recipe for a perfect marathon performance? While this question was already being answered by athletes qualifying, the additional obstacle created by marathon organizers led slower athletes and sports scientists to quest for the determinants of faster times. Enter the neural network, reemerging in 2010 as the foundation of what is now referred to as "deep learning" and holding the potential to unlock the answers many were looking for. Thus, much of modern research at the intersection of athletics and artificial intelligence attempts to answer what factors matter most when trying to create a realistic expectation for a marathon performance. Prior to the resurgence of neural networks, researchers relied on the tried-and-true regression methods. In 2011, the Journal of Human Sport and Exercise published a paper entitled "Prediction of marathon performance time on the basis of training indices" by Tanda (2011) interested in predicting marathon performance using various training variables with regression of different shapes (linear, exponential, polynomial, etc.). Though there were some training metrics that were deemed of "high" correlation, the rigidity of regression makes its objectively inferior to a more flexible modeling technique, a comment the authors themselves made. Despite the rigidity, regression remained one of the most popular methods to employ. A review published in the International journal of Sports Physiology and Performance analyzed thirty six different studies, finding one hundred and fourteen regression equations reported Keogh et al. (2019). The review found that around half of the equations relied on "anthropometric" variables (training volume, hydration, etc.) whereas the other half involved laboratory measurements. The paper concluded by stating the heterogeneic nature precluded them from determining an equation that was superior, and thus runners should avoid relying on a single equation. Moving closer to present day, El-Kassabi et al. (2020) published "Deep Learning Approach for Forecasting Athletes' Performance in Sports Tournaments" in September 2020 using a more complex methodology to similarly predict marathon performance based solely on finishing time. Utilizing three different types of neural networks for their analysis – Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), and Gated Recurrent Units (GRU) – they were some of the first to experiment with the results of a highly flexible and more complex statistically modeling tool. Most recently in May 2023, Lerebourg et al. (2023) investigated marathon performance with deep learning using slightly different criteria (10K time, BMI, Age, and Sex) publishing "Prediction of Marathon Performance using Artificial Intelligence" in the International Journal of Sports Medicine. Though these papers are unique in some regards, there are several features they all have in common. First, they all have a fairly small sample size with the smallest only having sixteen subjects and the largest having eight hundred and twenty. While this is standard for regression, for more complex machine learning methodologies, a dataset of a couple thousand is preferable. Second, they all focus on predicting marathon performance (partly for reasons mentioned above) mainly in the form of finishing time prediction. Finally, a common limiter contributing to the small sample size by the research to date is the requirement for collecting non-public participant data by some sort of formal recruitment.

In contrast, this study seeks to harness the extensive reservoir of historical marathon data to empower a neural network with an abundance of information. The analysis examines whether augmenting the dataset while reducing the number of variables constitutes a worthwhile tradeoff for a deep learning tool. An innovative approach is taken by incorporating participants' official finishing times as an input variable rather than an output, offering a fresh perspective on correlation—an aspect that previous research has largely overlooked. Moreover, capitalizing on the geographic data provided by race organizers presents an entirely novel angle on race prediction, posing an equally compelling research question. Certain regions of the world (particularly Kenya and Ethiopia) have produced an outsized number of top level marathon runners, justifying the theory of geography playing a role in performance. A participant's origin serves as a crucial piece of information, reflecting significant aspects of their training experience such as terrain (flat versus mountainous), climate (hot versus cold), geography (sea level versus altitude), among others. Consequently, it's plausible to consider that their marathon performance may be influenced by their geographical background, particularly considering the demonstrated correlation between training conditions and race outcomes.

This paper will first establish the datasets and variables of interest, fit those variables to first a baseline model of logistic regression and then subsequently to a sequential neural network, and conclude with the results of the modeling and discussion of potential future research in this domain.

2 Data Description

The data utilized for the modeling were taken from the official website for the Boston Marathon and the NYC Marathon from years 2000 to 2023. The justification for these specific races comes from the sheer volume of participants and historical data available. In a given year, NYC and Boston marathons have drawn a combined 80,000 people and are the largest and third largest marathons conducted in the United States, respectively. The second largest marathon, the Chicago Marathon was originally to be included; however, after inspecting the website in greater detail and attempting to collect the data it became clear the needed information was too sparse to warrant inclusion, particularly since participants origin was not a mandate to report, shrinking the dataset immensely.

The official website containing the race results for the Boston Marathon allowed for a download, in the format of a comma separated file (csv) of the information for all years of all finishers making for a relatively simple retrievable of the data. Unfortunately, requiring the participant origins of the Boston Marathon was optional prior to 2010 making around half the observations of every year from 2000 to 2009 unusable. The information contained in each csv is listed below:

- Name
- Bib number
- Age
- Gender
- City
- State (where applicable)
- Country
- 5K split
- 10K split
- 20K split
- Half
- 25K split
- 30K split
- 35K split
- 40K split
- Pace
- Official Time
- Overall (place)

For the purposes of this analysis, the only relevant information was the residency (city

Table 1: Descriptive Statistics for NYC Marathon and Boston Marathon: All years.

Variable	NYC Marathon					Boston Marathon				
	Mean	StDev	Minimum	Median	Maximum	Mean	StDev	Minimum	Median	Maximum
Participants	23,301.00	13,829.00	12,608.00	22,526.00	53520.00	13,408	5,595.00	13,387.00	9,086.00	31,926
Age	39.20	10.62	17.00	38.00	91.00	42.00	11.02	18.00	42.00	87.00
Official Time	04:40:43	00:57:01	02:09:13	04:34:31	11:44:10	03:48:48	00:37:13	02:08:41	03:43:13	07:17:47

and state) of each finisher and their respective official time. Any observation that contained a null value for these relevant columns would be removed from the dataset entirely.

A csv download was not possible regarding the results for the New York City Marathon. Thus, the only way to collect the data was through website scraping. The information scraped can be seen below:

- Name
- Gender
- Age
- State (where applicable)
- Country
- Time
- Overall (place)

As with the Boston Marathon data, a csv file was created for each year[1] with the model only being trained using state and time. The pre-processing/data cleaning for the NYC Marathon was more straightforward with the only necessary step being removing irrelevant columns. The Boston Marathon data required more avid standardization of the column names, reformatting of the inputs, and removal of incorrect or irrelevant inputs. Once the individual years for both marathons were formatted identically, a single csv file was generated with all years for each marathon.

Table 1 above contains the general statistics on both races over the past 23 years for

[1] Note: Some years were absent from the analysis from each race for various reasons. For the NYC Marathon, there was no race in 2012 due to the recent damage caused by Hurricane Sandy. For the Boston Marathon, 2013 was purposely omitted as it was the year of the Boston Marathon bombing.

United States participants, regardless of whether they indicated a city or state of origin.

Once the preprocessing was finished, the data were split into four classes – North, South, Midwest, and West. Table 2 representing the split can be found below:

Table 2: The table below indicates which states belong to which Class for the purposes of training the model to classify each region.

Region			
North (Class 0)	**South (Class 1)**	**Midwest (Class 2)**	**West (Class 3)**
Maine	Alabama	Illinois	Washington
New Hampshire	Arkansas	Indiana	Oregon
Delaware	Virginia	Iowa	California
Massachusetts	Florida	Kansas	Idaho
Rhode Island	Georgia	Michigan	Nevada
Connecticut	Kentucky	Minnesota	Utah
New York	Louisiana	Missouri	Colorado
New Jersey	West Virginia	Nebraska	Montana
Pennsylvania	Mississippi	North Dakota	Wyoming
Vermont	North Carolina	South Dakota	Alaska
Maryland	Oklahoma	Ohio	Hawaii
	South Carolina	Wisconsin	Arizona
	Tennessee		New Mexico
	Texas		

The regional classes would be added as a column to the csv files combining all years. With the state of origin, official time, and regional class all in the same file, it finally contained all necessary information to start training the model. Subsequently, the issues regarding the sample imbalances were considered. Given both races take place in the Northeast, it was suspected that there would likely be more observations coming from that region; however, the imbalance was not deemed severe enough to weight the classes in a custom manner. As explained in greater detail in Section 4, it became clear that arbitrary class balancing was needed after the initial run of fitting the neural network. K-fold Cross Validation was also performed to enhance the data splitting process as it was deemed superior than modeling on an 80/20 split. At this time it was decided two separate models would be trained, one for NYC and one for Boston, as opposed to combining the data across races.

Given the size of the data set, exploratory analysis to uncover any superficial trends was

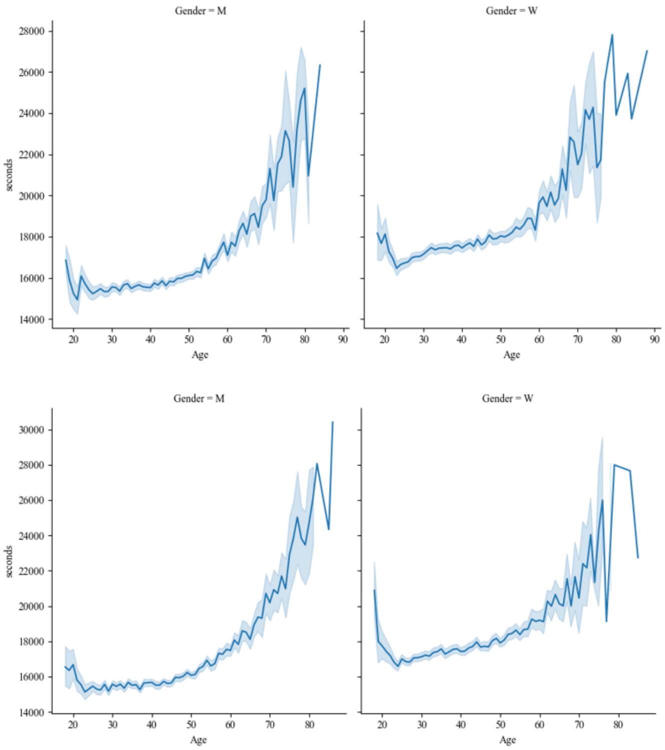

Figure 1: Finishing times by age for years 2016 (top) and 2018 (bottom) in the NYC Marathon.

deemed necessary. One preliminary analysis that ended up becoming relevant later was of finishing time (in seconds) versus the age of each participant. The figures for each respective race can be seen in Figures 1 and 2.

The trends of the two years are very similar with the volatility significantly increasing in the older age range. Though these are only two years, the non-linear trend is consistent amongst all years that were scraped. Similar to New York, Boston exhibited an exponential shape and increased volatility. As shown in Figure 2, it can be seen the trend across both races is also similar.

Understanding these trends became more important when it came time to training the model as the increase in noise in the data proved to be an issue. A more detailed explanation of the issue is provided in Section 4 of the paper.

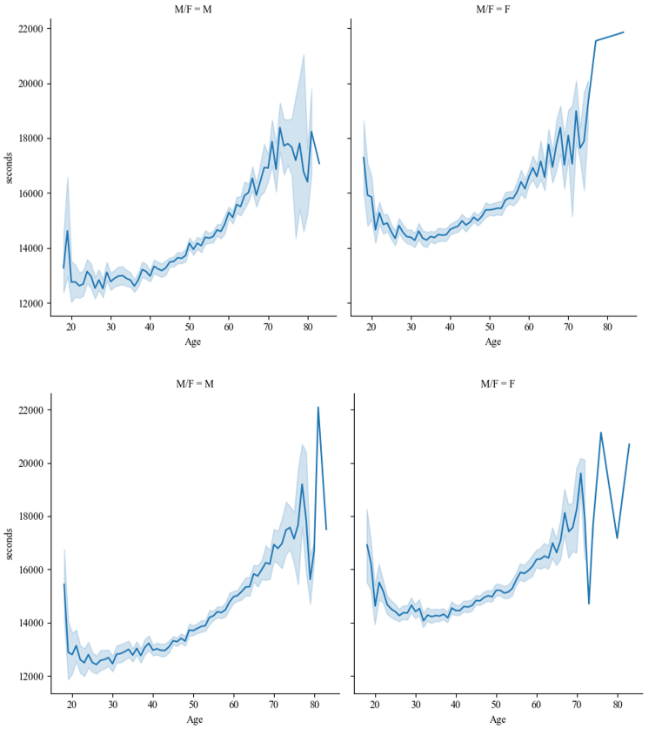

Figure 2: Finishing times by age for years 2016 (top) and 2018 (bottom) in the Boston Marathon.

3 Methods

The list of models utilized for the analysis were logistic regression and a feedforward neural network. Before the training of both models, several more preprocessing steps were needed to ensure the data were properly set up. The four regional classes were converted to numeric values (0-3) and the time variables were converted to seconds for training purposes. The final csv file contained only two columns of data, a ‘Class’ variable and a ‘Seconds’ variable that contained the transformed official times of all finishers for all years. Once all preprocessing steps were successfully completed, the modeling began.

The first step to this endeavor was fitting a logistic regression model to allow for a baseline comparison to the performance of the neural network. The justification behind the logistic was the simple nature it has in regards to interpretability and implementation, particularity when compared to the deep learning alternative of a neural network. Additionally, the thought of comparing the performance of a simple and complex methodology seemed compelling. A logistic regression model predicts the probability that Y is in a particular class, rather than modeling Y directly. A general logistic function will take the following form:

$$P(Y = 1 \mid X = x) = \frac{e^{\beta_0+\beta_1 x}}{1 + e^{\beta_0+\beta_1 x}} \tag{1}$$

where $p(X) =$ the probability of Y belonging to class 1 as a function of X. The parameters are β_0 and β_1, and $P(Y = 0 \mid X = x) = 1 - P(Y = 1 \mid X = x)$.

Through multiplication and taking the logarithm of both sides returns:

$$\log\left(\frac{p(X)}{1 - p(X)}\right) = \beta_0 + \beta_1 X. \tag{2}$$

The left-hand side of this equation is often referred to as the logit, or log odds, which is linear in X. Estimation of the parameters in the logistic model is done with the maximum likelihood method. The likelihood function can be mathematically represented for an independent sample of size n as:

$$L(\beta_0, \beta_1) = \prod_{i:y_i=1} p(x_i) \prod_{i\prime:y_{i}\prime=0} 1 - p(x_{i}\prime)$$

for $i, i\prime \in \{1, ..., n\}$.

The values of β_0 and β_1 that maximize $L(\beta_0, \beta_1)$ are the maximum likelihood estimates. This maximum likelihood method is a common approach to fitting non-linear models. These equations, however, represent the general logistic model in which there is only a binary response classification. Given there are four classes in this analysis, multinomial logistic regression was employed. Thus, (1) is replaced with:

$$P(Y = k \mid X = x) = \frac{e^{\beta_{k0}+\beta_{k1}x_1+...+\beta_{kp}x_p}}{1 + \sum_{l=1}^{K-1} e^{\beta_{l0}+\beta_{l1}x_1+...+\beta_{lp}x_p}}$$

for $k = 1, ..., K-1$ and

$$P(Y = K \mid X = x) = \frac{1}{1 + \sum_{l=1}^{K-1} e^{\beta_{l0}+\beta_{l1}x_1+...+\beta_{lp}x_p}}.$$

Similar to (2), it can be shown that the log ratio of probabilities between any class with the reference class K are linear in features:

$$\log\left(\frac{P(Y = k \mid X = x)}{P(Y = K \mid X = x)}\right) + \beta_{k0} + \beta_{k1}x_1 + \ldots + \beta_{kp}x_p$$

for $k = 1, .., K-1$.

When coding for multinomial logistic regression, softmax coding was used to indicate the presence of multiple classes ($K > 2$). This resulted in treating all K classes symmetrically rather than having a baseline class, and assumes

$$P(Y = k \mid X = x) = \frac{e^{\beta_{k0}+\beta_{k1}x_1+...+\beta_{kp}x_p}}{\sum_{l=1}^{K} e^{\beta_{l0}+\beta_{l1}x_1+...+\beta_{lp}x_p}}$$

for $k = 1, 2, ...K$.

The second and final model utilized was that of a feedforward neural network. Specifically, it's a multi-layer perceptron (MLP) model, which is a type of feedforward neural network where multiple layers of nodes (neurons) are connected sequentially. The inherent architecture of neural networks, inspired by the human brain, allows them to automatically extract hierarchical features, enabling the modeling of intricate relationships within the data.

This model was the most appropriate given the possibility of a very nuanced class division. A visual representation of the network architecture can be seen in Figure 3.

For the simple, single layer neural network, the form will take the following:

$$\begin{aligned} f(X) &= \beta_0 + \sum_{k=1}^{K} \beta_k h_k(X) \\ &= \beta_0 + \sum_{k=1}^{K} \beta_k g\left(w_{k0} + \sum_{j=1}^{p} w_{kj} X_j\right) \end{aligned}$$

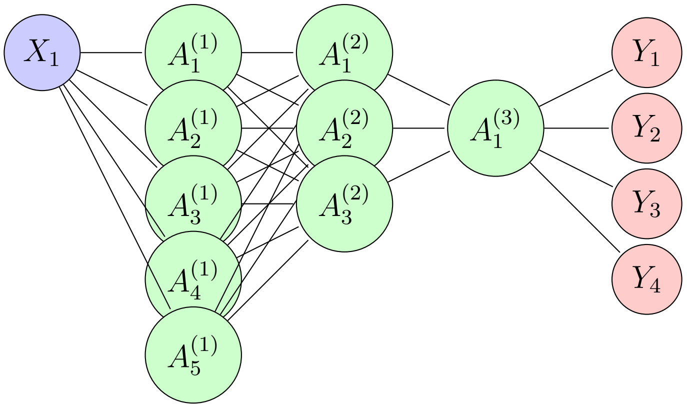

Figure 3: Neural Network Structure: This diagram depicts one of the specific neural network architecture employed here, with three hidden layers and one output. The input later had $p = 1$ unit (finishing time), the three hidden layers $K_1 = 50$, $K_2 = 30$, and $K_3 = 10$ units respectively, and an output layer of 4 units as the classes were divided into 4 regions. Thus, this neural network has a total of 1984 parameters, or weights, when including the intercept (also referred to as bias). A simple neural network was also conducted, wherein there was only one hidden layer and one output. The input layer had $p = 1$ unit, one hidden layer $K_1 = 100$, and an output layer of four units.

where p is the vector of variables $X = (X_1, X_2, ...X_p)$, K is the activation (denoted A_k, and $g(z)$ is a nonlinear activation function that is predetermined.

For the purposes of this analysis, the softmax activation function was chosen for the output as it can represent the classes in terms of probabilities. The equation

$$f_m(X) = \Pr(Y = m|X) = \frac{e^{Z_m}}{\sum_{k=0}^{4} e^{Z_k}}$$

for $m = 0, 1, ...3$, can mathematically portray the softmax activation. Utilizing this activation through to the output layer ensures the classes are treated as probabilities (in the sense that they sum to 1 and are all non-negative).

Another very popular activation known for its computational efficiency and simplicity is the Rectified Linear Unit, or ReLU, and this was utilized through the input layers. As seen in Figure 4, both activation's (scaled down for visual purposes) can be seen graphed.

This model can be constructed in two main stages. Initially, the A_k activation's for $k = 1, ..., K$, are computed in the hidden layer as functions of the input features $(X_1, X_2, ...X_p$,

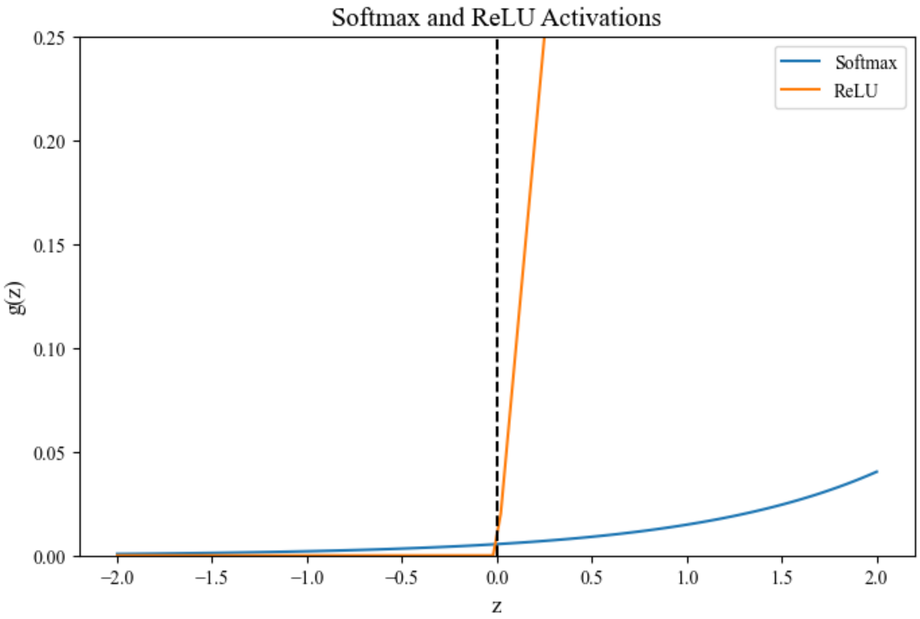

Figure 4: Softmax versus ReLU Activation

represented as

$$\begin{aligned} A_k &= h_k(X) \\ &= g\left(w_{k0} + \sum_{j=1}^{p} w_{kj} X_j\right). \end{aligned} \tag{3}$$

Each function A_k can be considered as a distinct transformation $h_k(X)$ applied to the original features. These K activation's, originating from the hidden layer, subsequently contribute to the output layer, finally resulting in

$$f(X) = \beta_0 + \sum_{k=1}^{K} \beta_k A_k.$$

Parameters $\beta_0, ...\beta_K$ and $w_0, ..., w_{Kp}$ are estimated using the data provided.

That is the basic form of a *single* neural network, meaning there is only one hidden layer; however, it is much more common to see a network consisting of multiple layers as the learning task of discovering a good solution is made much easier. In this paper, both simple and multi-layered neural networks were run. The addition of more layers was to aid the neural network in learning the potential non-linear feature, especially since that is the only input into the network.

Mathematically, the first hidden layer is the same as seen in (3). The second hidden layer then generates new activation's by treating the first hidden layer activation's, A_k, as inputs. The second layer activation's can thus be represented as

$$\begin{aligned} A_l^{(2)} &= h_1^{(2)}(X) \\ &= g\left(w_{l0}^{(2)} + \sum_{k=1}^{K_1} w_{lk}^{(2)} A_k^{(1)}\right) \end{aligned} \tag{4}$$

for $l = 1, ..., K_2$.

It can be observed that the activations in the second layer $A_l^{(2)} = h_1^{(2)}(X)$, are dependent on the input vector X. This dependence arises because although they are linked to the activation's $A_k^{(1)}$ from the first layer, they are consequently functions of X. This relationship holds true for the addition of more hidden layers as well. In turn, through a sequence of transformations the network can construct intricate representations of X that eventually contribute as features to the output layer. The superscripts added to (4) are present to indicate the referenced hidden layer.

The output layer consists of four different responses since there are four different classes, which are computed by generating four different linear models (similar to the single neural network),

$$\begin{aligned} Z_m &= \beta_{m0} + \sum_{k=1}^{K_2} \beta_{mk} h_k^{(2)}(X) \\ &= \beta_{m0} + \sum_{k=1}^{K_2} \beta_{mk} A_k^{(2)} \end{aligned}$$

for $m = 0, 1, ...3$. The forty (10 neurons $\cdot$ 4 responses) weights are then stored in a matrix.

Given the output is qualitative, training the model is done through minimizing the negative multinomial log-likelihood through the coefficient estimates. Mathematically, that can be represented by:

$$-\sum_{i=1}^{n}\sum_{m=0}^{3} y_{im}\log(f_m(x_i)),$$

which in the realm of deep learning is referred to as cross-entropy. The mean-squared error loss would be minimized if the desired output were quantitative.

Regularization, ridge or dropout, was deemed unnecessary given the extremely simple nature of this neural network and the low risk of overfitting.

The performance metrics used for all models conducted was the overall [2] accuracy. To graphically display these results, a confusion matrix was created for the neural networks. This visual typically gives a clearer representation of the spread of the accuracy. For the baseline comparison model, summary Table 5 was presented.

4 Results

In this section, an investigation into the performance of all neural networks as well as the baseline comparison, logistic regression, will take place. The results are summarized below in Tables 3 and 4.

Both tables are organized in the order in which they were run. Looking first at the neural networks for New York, they had a slightly lower accuracy compared to that of the Boston Marathon neural networks. Figure 5 depicts the confusion matrix for the initial model that was trained (Model 1).

The algorithm visibly over represented Class 1 (South) and highly underrepresented Class 3 (West). There is no clear explanation for this, as the classes were divided equally as to completely eliminate bias. What was clear, however, was that the network really struggled to classify people, rendering an accuracy that sits just below 30%. In an attempt to tune the

[2] Note: The overall accuracy was an average across the five folds.

Table 3: Neural Network Model Comparison: New York Marathon

Model Name	Architecture	Sample Size	Optimizer	Accuracy (%)
Model 1	3 hidden layers (50, 30, 10 neurons)	120,000	Adam	30.3
Model 2	1 hidden layer (100 neurons)	120,000	Adam	37.6
Model 3	3 hidden layers (50, 30, 10 neurons)	120,000	Adam	31.2
Model 4	1 hidden layer (100 neurons)	120,000	Adam	37.9

Table 4: Neural Network Model Comparison: Boston Marathon

Model Name	Architecture	Sample Size	Optimizer	Accuracy (%)
Model 1	3 hidden layers (50, 30, 10 neurons)	22,500	Adam	30.5
Model 2	1 hidden layer (100 neurons)	22,500	Adam	40.7
Model 3	3 hidden layers (50, 30, 10 neurons)	22,500	Adam	30.7
Model 4	1 hidden layer (100 neurons)	22,500	Adam	41.1

model from the data side, every participant who was over the age of sixty was removed from the training data along with the entirety of class three. The justification for these changes were that the finishing time of a participant became much more volatile starting at around age 60, making it more difficult for the model to classify those individuals using their time. This phenomenon can be seen in Figure 1.

Further, looking at Figure 5 it is clear the algorithm struggled to differentiate Class 3 in a meaningful way, and removal of this class may help increase the overall accuracy as it does not have to struggle classifying these observations. Since the increase in volatility was seen in both races, the altering of the dataset and removal of the underrepresented class was done for the NYC Marathon and the Boston Marathon. Rerunning the neural network for New

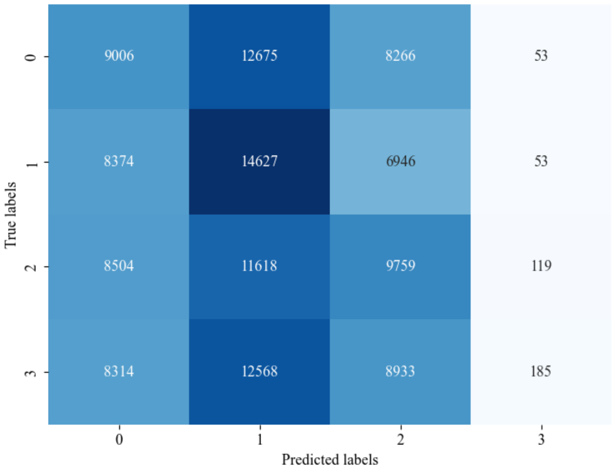

Figure 5: Confusion Matrix for Model 1: New York

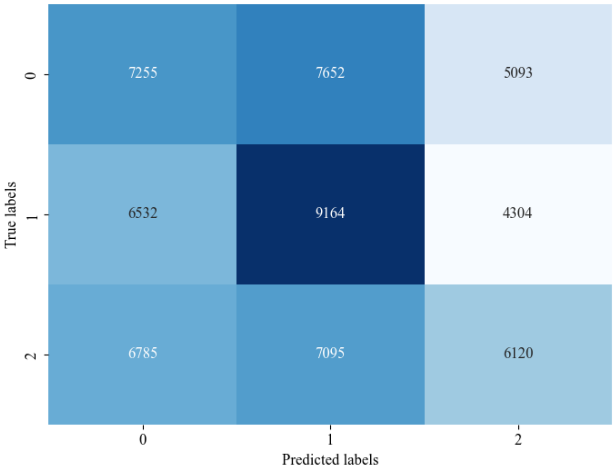

Figure 6: Confusion Matrix for Model 3: New York

York gave the confusion matrix seen in Figure 6, with an increase in performance accuracy by about 10%. The true increase in accuracy, however, was lower given the decrease in the number of classes.

Next looking at Boston, the initial model, represented by Figure 7, had a nearly identical accuracy to New York, but removing the underrepresented class (Class 1) had a slightly greater impact as the most successful run had an accuracy just below 42%, a couple percentage points higher than the best run for New York. The respective confusion matrix with the removal of Class 1 can be seen in Figure 8.

As can be seen, the tuning of the architecture did little, if anything, to improve the pre-

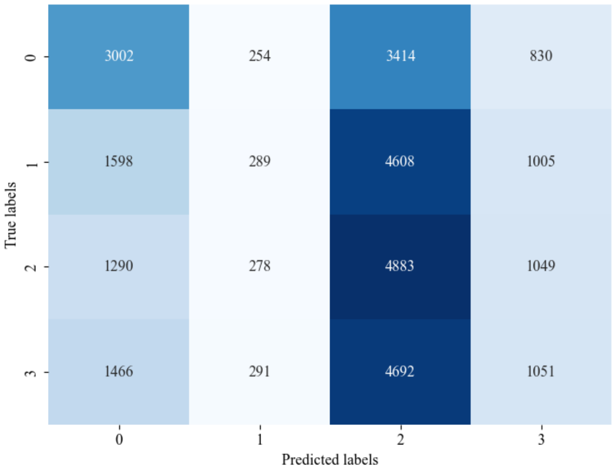

Figure 7: Confusion Matrix for Model 4: Boston

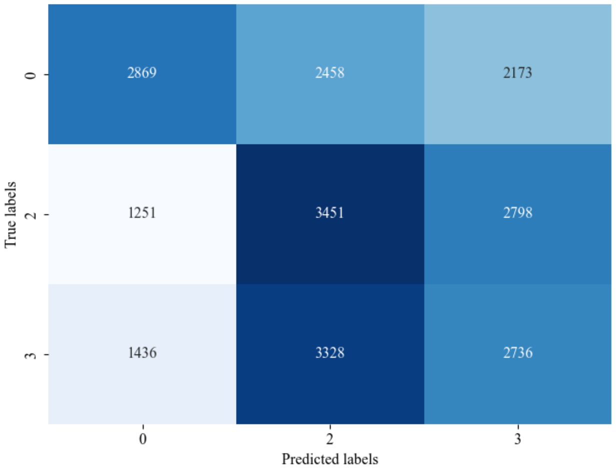

Figure 8: Confusion Matrix for Model 4: Boston

diction accuracy for either race, but was still attempted to exhaust all methods of increasing the accuracy.

When comparing the performance of the neural networks to our baseline comparison of logistic regression, it was better in one run and worse in the other. Note that logistic regression was performed only on the dataset that excluded participants over 60 and the worst performing class. Looking at Table 5 which summarizes the results, it can be seen the logistic regression run for New York outperformed the best performing neural network model. The same is not the case for Boston with the neural network having a superior accuracy, though the difference is not substantial.

Table 5: Summary of Performance of Logistic Regression and Neural Networks

Marathon	Logistic Regression Accuracy (%)	Neural Network Accuracy (%)
New York	38.3	37.9
Boston	39.6	41.1

5 Discussion

To recapitulate, this research aimed to understand the intricate relationship between geographic origins and marathon finishing times. Previous studies have applied various forms of regression to explore the predictive capabilities of machine learning; however, many of these studies relied on small sample sizes, often collected through surveys. In contrast, this research utilized finishing time as the input variable, leveraging the vast amount of publicly available data on the internet to create a larger training set for the model. Although this approach resulted in a significantly larger sample size, it was found that additional variables, such as diet and training volume, may be necessary to achieve higher accuracy.

The superior performance of the neural network over logistic regression in the case of Boston suggests its capability to capture the complex, non-linear relationship between input and output variables. However, the analysis revealed that the model for New York Marathon underrepresented the West (Class 3), possibly due to increased variability in finishing times resulting from longer travel. Similarly, the neural network for the Boston Marathon marginalized the South (Class 1), though the reasons for this remain unclear. It is interesting to note that both marathons had a different region (Class) that performed statistically worse. Some potential factors that in principle could come into play are moderate climate year round and high altitude for red blood cell production and better endurance. To address these shortcomings and provide more definitive answers, further data collection and analysis are required.

Looking ahead, it is imperative to broaden the scope of analysis by incorporating additional data sources and refining modeling techniques to enhance predictive accuracy. Factors such as genetic predispositions, socioeconomic backgrounds, and cultural influences should be considered to develop a more comprehensive understanding of regional patterns in marathon performance. Furthermore, our findings underscore the importance of exercising caution when interpreting predictive models, particularly in complex domains like athletic performance. While machine learning algorithms offer valuable insights, they should be complemented by qualitative analysis and diverse data collection methods.

In conclusion, while this study did not produce highly predictive results, it did identify some curious regional under performance that requires further investigation to better explain. By adopting a multidisciplinary approach and integrating diverse data sources, future research endeavors hold promise in unraveling the complexities of marathon performance.

References

El-Kassabi, H. T., Khalil, K., and Serhani, M. A. (2020), “Deep Learning Approach for Forecasting Athletes’ Performance in Sports Tournaments,” in *Proceedings of the 13th International Conference on Intelligent Systems: Theories and Applications*, pp. 1–6.

Keogh, A., Smyth, B., Caulfield, B., Lawlor, A., Berndsen, J., and Doherty, C. (2019), “Prediction equations for marathon performance: a systematic review,” *International journal of sports physiology and performance*, 14, 1159–1169.

Lerebourg, L., Saboul, D., Clémençon, M., and Coquart, J. B. (2023), “Prediction of Marathon Performance using Artificial Intelligence,” *International Journal of Sports Medicine*, 44, 352–360.

Tanda, G. (2011), “Prediction of marathon performance time on the basis of training indices,” *Journal of Human Sport and Exercise Volume 6(Issue 3)*, 521–520.