Personalized Race Time Prediction Using Machine Learning on Strava Running Data

Edward Feng

NCSU CSC Department

sfeng9@ncsu.edu

***Abstract*— Predicting race finish times is a common goal for recreational runners seeking to set realistic targets and monitor their progress. This study leverages machine learning techniques to develop a predictive model for race finish times using real-world training data collected from Strava. The dataset includes distance, average speed, elevation gain, and heart rate as features for detailed analysis and model development. Initial results revealed limitations in prediction accuracy due to the inclusion of recovery runs and other outliers. After implementing data filtering, the model achieved significant improvements, with a 22.9% decrease in mean absolute error (MAE) and a 45.7% decrease in mean squared error (MSE). Additionally, a decision tree classifier was trained to identify run types, further enhancing data preprocessing. These findings demonstrate the potential of machine learning for providing actionable insights into training performance and race preparation. Future work will focus on adding extraneous variables to improve accuracy and broaden the model's applicability to other activities such as cycling and swimming.**

I. Background

Running is one of the most accessible and popular forms of exercise, with millions of people participating in races ranging from local 5Ks to marathons each year. For many, race participation is more than just a test of physical endurance—it is an opportunity to set personal goals, such as achieving a sub-two-hour half marathon or a sub-four-hour full marathon. Achieving such goals requires careful planning, consistent training, and an understanding of various factors that influence race performance, including pace, distance, heart rate, and elevation gain.

Modern fitness tracking platforms like Strava have revolutionized the way runners collect and analyze data from their workouts. These platforms provide detailed metrics, such as average pace, heart rate, and elevation gain, which can be invaluable for assessing progress and tailoring training plans. However, these platforms often lack predictive features that could help runners anticipate their race finish times based on their training data.

This project aims to bridge that gap by utilizing ML to predict race finish times using real-world running data. The dataset, collected from Strava, includes variables such as distance, average speed, elevation gain, heart rate, etc. By analyzing this data and implementing regression and classification algorithms, this study seeks to provide predictions that align with individual training patterns. The findings could help recreational runners set realistic goals, improve training efficiency, and better prepare for race day.

II. Data Analysis

The analysis began with importing and organizing the dataset from Strava, which included attributes such as distance, pace, elevation gain, average heart rate, shoe, etc. The data was cleaned by handling missing values, converting attributes into numerical formats suitable for machine learning, and removing duplicates. Exploratory Data Analysis was conducted to identify patterns and trends, including summary statistics, correlation analyses, and linear regression to examine relationships between key variables.

The initial analysis using linear regression revealed that both distance and average speed showed weak positive correlations with time over the dataset as a whole. However, this finding was complicated by the presence of recovery runs and other variations in training. Filtering out recovery runs improved the linear relationships, suggesting that these runs introduced noise to the data. Additionally, it was observed that higher distances yielded more accurate predictions compared to shorter runs, indicating the need for nuanced feature selection and preprocessing

to account for run type and training specificity. For race time prediction, linear regression was chosen for its simplicity and ability to provide interpretable results. For distinguishing run types, a decision tree classifier was used due to its capacity to handle categorical variables and provide clear decision-making rules. The insights from EDA influenced the decision to exclude recovery runs and incorporate only relevant features for model training.

III. Data augmentation

Data cleaning was necessary for the reliability of the analysis. The dataset contained some missing values, which were addressed by removing activities where data was insufficient or missing when required for meaningful analysis.

The dataset also underwent several modifications to optimize it for analysis and modeling. Date values were converted into numerical ordinal representations and time features were transformed from the "hh:mm:ss" format into total minutes for numerical consistency. Categorical variables, such as run_type and shoe, were encoded for compatibility with machine learning models. During the model training, normalization was performed to ensure that features with varying scales did not dominate the models.

Data augmentations ensured the dataset was clean and structured for targeted analysis, enabling robust modeling and actionable insights.

IV. Model Selection

Two models were chosen based on the project goals. Linear regression was used for the prediction section and the decision tree classifier was used for the classification section.

Linear regression is well-suited for predicting continuous values such as race finish times. It offers simplicity and the ability to identify relationships between the input features and the target variable. Although our initial exploratory data analysis did not indicate significant linear trends between these variables, linear regression still gives the advantage of being computationally efficient and robust for datasets of moderate size.

The decision tree classifier was selected to classify run types because it works fast, is highly accurate even with small datasets, and can handle different data types and non-linear relationships. All of this aligns with the project's conditions and needs for practical implementation.

Although both models were built using standard libraries, they refer to designs from other projects. The linear regression model refers to course materials and examples of linear regressions but is tailored based on insights gained from the initial analysis. The model was implemented using the LinearRegression class from the scikit-learn library. On the other hand, the decision tree classifier refers to a project I did on airline delay prediction using several different models, and the decision tree classifier was one of them. The decision tree classifier was built using the DecisionTreeClassifer from scikit-learn with entropy as the splitting criterion and seaborn was used for visualization.

V. Training Methodology

Our training process for linear regression began by filtering the dataset to exclude recovery runs, as these types of runs tend to skew the results due to their intentionally shorter duration and slower pace. We then set up the features and intended target (time). After filtering, the dataset was split into training and testing sets using an 80/20 ratio. Normalization was applied to the features to ensure that all input variables were scaled proportionally.

Various hyperparameters were experimented with during training to optimize performance. For the decision tree, different number of folds were used for cross-validation during training to mitigate the risks of underfitting and overfitting [2]. For linear regression, different sets of features were tested to find the optimal parameters for prediction. For example, excluding maximum heartrate as a feature improved our results because it is less significant and can skew our results (maximum heartrate can be similar despite doing different types or run).

By combining these strategies, the training process produced an optimal model that provides predictions for runners.

VI. Results

The performance of the models was evaluated using different but common key metrics and validation techniques for our two models. For linear regression, the model was evaluated based on its ability to predict finish times for races. The Mean Absolute Error and Mean Squared Error were calculated to assess how closely the model's predictions align with the actual times. On the other

hand, the decision tree classifier was evaluated by using accuracy, precision, recall, and F1 score to see how well the classifier could differentiate between various run types [3].

The linear regression model outputs a predicted race finish time. It had an MAE of 4.67 and MSE of 34.87 after filtering out recovery runs. When testing a half marathon with a future date and other race information, our predicted finish time was 131 minutes. The actual time was 120 minutes. On the other hand, the decision tree outputs a classification of the run type based on features like distance, speed, and heartrate. The model can predict whether a specific run is for training or recovery.

Our initial analysis played a significant role in training the model and improving the its performance. Without checking the data types, we would not have converted the variables to a better format to process them more effectively. In addition, our analysis revealed that the presence of recovery runs skewed the results. By removing these runs from the dataset, the model was able to nake more accurate predictions, as evidenced by the decrease in MAE by 22.9% and in MSE by 45.7%.

VII. Future Works

While the models performed decently, there were some limitations that could have hindered the models' performance and could be improved:

One of the most significant limitations was the relatively small number of observations in the dartaset. A larger dataset would provide the model with more diverse and comprehensive examples, enabling it to better capture the nuances of race performance under varying conditions. The decision tree model suffered from the risk of overfitting due to using cross-validation with a small number of training, as there are fewer data points to represent the broader trends.

Another limitation includes the lack of features that influence race performance. Extraneous variables such as weather conditions, nutrition, training fatigue, or even mental state could impact a runner's performance. Including these variables in the dataset could help the model make more accurate predictions.

Finally, experimenting with more advanced learning algorithms, such as deep learning, could also improve the model's predictive accuracy. While linear regression performed decently for this data, more sophisticated techniques might capture the relationships in the data better.

There are several directions for future projects that I would like to expand on the work done here. Incorporating data from biking and swimming could make the model more versatile especially for athletes who train across multiple disciplines, such as triathletes (something I might try in the future). By analyzing multi-sport activities, the model could provide deeper insights into performance trends.

In addition, predicting injury risk is a potential application that I would like to work on. Integrating more data such as training volume, recovery periods, or heart rate variability could help identify patters associated with overtraining or injury. This predictive capability not only aid in optimizing training but also promote health and performance for athletes.

As I take more advanced coursework in machine learning and data science, designing a custom algorithm or model could be the ultimate challenge. Tailoring the algorithm to the unique characteristics and patterns could lead to breakthroughs in prediction accuracy and model interpretability.

From this project, several lessons were learned: data quality and cleaning are crucial and can significantly improve model performance, model evaluation and hyperparameter tuning can also dramatically improve the results, and that real-world data can be more challenging (incomplete and noisy). With these lessons learned, future projects can be more efficient and effective, leading to better models for real-world applications in sports performance and health.

VIII. Stakeholder Acknowledgements

The primary stakeholders for this project are recreational and competitive runners who rely on data-driven insights to enhance their training and achieve their race goals. The model provides personalized predictions of race finish times based on their training data, benefitting atheletes by allowing them to set realistic goals and plan effective training regimens. Additionally, the model could help users identify trends in their performance over time and the intensity of training, whether if recoveries are needed.

However, there are some potential negative effects. The current models inaccuracies could

misguide users in their goal-setting and training plans. In addition, over-reliance on the model's prediction without considering external factors such as health could lead to unrealistic expectations and even severe injuries.

Currently, the model is not ready for real-world testing and feedback because its predictions have not shown promising results. The next logical step would be to focus on improving the model performance, then have our stakeholders try the model with their personal feedback. Feedbacks can highlight areas for improvement, including more personalized inputs and better user experience.

IX. Conclusion

In conclusion, this project aimed to develop a predictive model for race time estimation and performance analysis for runners, leveraging historical race data and machine learning techniques. Through data analysis, model selection, and careful consideration of training methodologies, the model was able to offer valuable insights into athletic performance, assisting runners in setting realistic expectations and optimizing their training plans.

The data augmentation process ensured that the model was able to handle diverse inputs and provide predictions that are applicable to various race conditions. By selecting appropriate machine learning models and refining them, we were able to balance predictive accuracy with generalizability, ensuring that the model could work even with various running conditions.

The model is still a work in progress, with room for further improvements in areas such as the lack in observations, extraneous variables, and model performance. If we are able to get feedback from stakeholders, we will get valuable insights into the real-world applications of the model and areas where the model could be enhanced to better serve its intended audience.

Future work will focus on expanding the dataset, incorporating additional features (such as weather) and activities (such as biking and swimming), and potentially developing a customized model. Additionally, integrating the model into fitness technology platforms could provide users with real-time insights to help them improve their performance and achieve their race goals.

Overall, this project demonstrates the potential of predictive analytics in sports science and opens the door for future advancements in personalized athletic training and performance optimization.

References

[1] R. Anderson, "Running Smart with Machine Learning and Strava - Towards Data Science," *Medium*, Jan. 21, 2020. https://towardsdatascience.com/running-smart-with-machine-learning-and-strava-9ba186decde0

[2] L. A. Yates, Z. Aandahl, S. A. Richards, and B. W. Brook, "Cross validation for model selection: a review with examples from ecology," *Ecological Monographs*, vol. 93, no. 1, Nov. 2022, doi: https://doi.org/10.1002/ecm.1557.

[3] D. Powers and Ailab, "Evaluation: From precision, recall and F-measure to ROC, informedness, markedness & correlation," *Journal of Machine Learning Technologies*, vol. 2, no. 1, pp. 37–63, Jan. 2011, doi: https://doi.org/10.9735/2229-3981.