Paulina Delabra Serrano, Jane Warren, and Donald Ye
Prof. Kyle Dent
CISC 4631-L01
December 5th, 2025

Mining The Marathon

I. Introduction

In finance, we often hear the saying, "past performance does not guarantee future results." In running, this principle also applies. Marathon training is an endeavor that takes months of commitment, lifestyle changes, and effort from athletes, yet outcomes are never guaranteed. An athlete's performance at the beginning of the race might start strong, but a demanding elevation profile, a poor fueling strategy, difficult weather conditions, or an unexpected muscle pull might change the trajectory of the competition. During a marathon training cycle, after 6 to 8 weeks of consistent running, runners are recommended to run more than half a marathon every weekend, building up to 22-mile long runs. This way, runners have practice with at least 50% of the total distance and are familiar enough with the physical and mental demands that they can prepare for the last four to six miles. If runners have practiced so much with the first part of the race, a natural question emerges: would it be possible to make a prediction of their final marathon time and argue that "past performance can guarantee future results?"

II. Problem Definition and Motivation

This project aims to predict a runner's final marathon time based on demographic attributes (gender, age, and nationality) and split paces at key distance markers (5K, 10K, 15K, 20K, and Half Marathon). Understanding marathon performance dynamics is valuable for multiple parties. For runners, accurate mid-race predictions can inform real-time pacing adjustments and help set realistic goals for future races. For coaches, identifying performance patterns across demographics enables more personalized training strategies. For race organizers, predictive models can improve resource allocation, medical support positioning, and spectator engagement. All around the board, those connected to running seem to love predicting things.

The problem is particularly well-suited for data mining approaches for several reasons. First, the dataset from the 2025 NYC Marathon contains over 56,000 runners with detailed split times and paces, creating a rich opportunity for pattern discovery that would be impossible through manual analysis. Second, the relationship between early-race performance and final outcomes is non-linear and influenced by complex interactions between demographic factors, pacing strategies, and physiological limits, which are precisely the type of relationship that nonlinear regression models excel at capturing.

In an age where runners don't start a run before starting their Strava, and race organizers track runners' every step with timing mats and GPS devices, understanding the dynamics of marathon performance is more accessible than ever. The proliferation of data-collecting devices means sports science can benefit tremendously from rigorous data analysis. This project leverages data mining techniques to extract actionable insights from one of the world's largest marathons, contributing to our understanding of human endurance performance.

III. Data Source and Collection

NYRR's database is one of the most comprehensive marathon databases in the world. Various individuals have explored this rich mine of data and made portions available online (Rock, 2025). However, existing publicly available files only account for runners' demographic information and their finish times, not the split times at different key distances. To obtain the desired granular data, we developed a custom web scraper to extract detailed split information directly from the NYRR results API for the 2025 TCS NYC Marathon.

IV. Custom Data Collection

We developed a data scarper to obtain marathon performance data from the NYRR API. The challenge was that the NYRR API only allowed retrieving 500 records per request which was restrictive given our dataset of roughly 59,600 finishers. To work around we had to generate and iterate through many parameter combinations (e.g. bib numbers, age groups, common surnames) to collect the full dataset. This significantly increased the complexity of the scarper and called for careful handing to avoid missing records or exceeding rate limits.

The general runner data was shown in a different HTML page that he split data. The general running information data took around 1-2 hours to scrape while the split collection data took approximately 7-8 hours to scrape. Our final dataset represents 94.6% coverage of all marathon finishers, based on NYRR's official count of approximately 59,600 finishers. The 5.4% gap is attributable to runners who had incomplete timing data (missed multiple timing mats).

V. Dataset Description

Our raw dataset contained 1,881,791 total entries across three participant categories: 1,878,889 runners (traditional foot race participants), 1,565 handcycle participants, and 1,337 wheelchair participants.The dataset included the following features for each participant.

Demographic attributes: RunnerID (unique identifier), RunnerName, Gender (M/F), Age, City, Country, Bib number

Performance metrics: OverallTime (final marathon completion time), OverallPlace (finishing position), Split times at 34 distance markers (3M, 5K, 4M, 5M, 6M, 10K, 7M, 8M, 9M, 15K, 10M, 11M, 12M, 13M, HALF, 14M, 15M, 16M, 17M, 18M, 19M, 30K, 20M, 21M, 35K, 22M, 23M, 24M, 40K, 25M, 26M, MAR), Pace (per mile) at each split, Speed at each split, Distance covered at each split

The original format stored each split as a separate row, resulting in 34 rows per runner. This long-format structure totaled 1,881,791 entries but represented 56,391 unique runners who completed the race with recorded splits. The complete dataset and data collection scripts are publicly available in our project repository (Ye, 2025).

VI. Data Cleaning and Preprocessing

Given the focus of this study on running performance, we first filtered the dataset to include only foot race participants. Wheelchair and handcycle participants were removed because their performance profiles differ fundamentally. Next, we convert all the time measurements in the dataset to seconds, rather than HH:MM:SS or MM:SS format. We then restructured the data so that we had one row per runner, with each split as a column.

After the restructuring, we removed runners with missing OverallTime values to ensure our analysis focused on completed marathons. Other runners had missing intermediate split paces (particularly at mile markers like 13M), which occurred when runners were not captured by every timing

mat due to crowding or technical issues. Our missing value strategy differed between preprocessing and modeling. During preprocessing, missing intermediate splits were retained as NaN rather than imputed at the data cleaning stage. This is because we use a couple predictive models that natively handle missing values. Additionally, imputing paces is a difficult task because the mean pace could be very low for some runners and very high for others, thus creating an anomalous feature. In the modeling stage for the GradientBoostingRegressor model, numeric features (Split paces and Age) were imputed using median values, while categorical features (Gender and Country) were imputed using mode.

For the regression section of the project, we engineer a couple statistical features to improve model performance. These include average pace, fade and ratio between splits, and the coefficient of variation (CV). For the clustering portion, we normalized each runner's split paces by dividing by their starting pace, yielding factors around 1.0 where (values > 1.0 = slowing, values <1.0 = accelerating). We filtered splits with 40% missing data, such was done for missing data on mile 13. We also add minimum pace, maximum pace, and pace range onto the existing features.

VII. Quality Issues and Biases

We want to acknowledge that any findings obtained are limited. All the data collected comes from a single race on a single day (Sunday, November 2nd, 2025) under specific conditions: season, temperature, humidity, NYC's specific elevation profile, pavement type, among others. We recognize that all of these factors have an impact on athlete performance. A longer discussion of biases is done later in Section X.

VIII. Data Mining Methods

This project employed two complementary data mining techniques: supervised regression for predictive modeling and unsupervised k-means clustering for pattern discovery.

a. Regression with Gradient Boosting

For the first task, we chose regression over classification due to the continuous distribution of our target variable, marathon finish time. We choose gradient boosting models as our desired subset of regression models because they work well for data that is noisy, heterogeneous, and nonlinearly dependent. Gradient boosting works by iteratively combining small and weak predictive models to produce a larger, stronger one.

We could have discretized finish times into bins such as "sub-3 hours," "3-4 hours," "4-5 hours," or ""5+ hours," and frame this as a classification problem. However, this approach has significant drawbacks. For example, a runner finishing in 2:59:30 and another in 3:00:30 differ by only 1 minute but would be placed in different classes, while runners finishing at 3:01 and 3:59 (58 minutes apart) would be treated identically. This would represent a massive information loss problem. Runners care about specific time predictions, especially when setting a personal record is in question; therefore, probabilistic class membership (i.e. a 73% chance of finishing between 3-4 hours) would be insufficient. In contrast with classification, regression preserves the full information, provides interpretable point estimates with confidence intervals, and uses evaluation metrics (RMSE, MAE, R^2) that directly measure prediction error in meaningful units (minutes).

We implemented the following models for regression and compared performance.

I. **Gradient Boosting Regression:** This model served as our baseline. Its main limitation is its inability to handle missing values natively, so as stated above, we had to perform some imputation.
II. **Histogram-based Gradient Boosting Regression:** This is a modern version of the Gradient Boosting Regressor that possesses native missing value support and is designed for datasets with >10K samples, which matches our scale. It uses histograms, as implied in the name.
III. **Categorical Boosting Regression:** While this model is not available in Scikit-Learn, it specializes in working with categorical features. We selected CatBoost because it does not require one-hot encoding for categorical variables. It also provides native missing-value handling and ordered boosting, which uses permutations to prevent target leakage that is present in other gradient boosting algorithms (Prokhorenkova et al., 2018).

b. K-Means Clustering

For the second task, we chose clustering over supervised learning approaches because we did not have predefined pacing strategy labels, and our goal was exploratory to natural groupings in runner behavior rather than predict a known outcome. We selected K-Means as our algorithm because it performs well on a large dataset ($n > 50{,}000$), scales efficiently with sample size, and produces interpretable cluster centroids that represent prototypical pacing patterns (MacQueen, 1967).

K-Means works by iteratively assigning data points to the nearest cluster centroid and updating centroids based on the mean of the assigned points, minimizing cluster variance (MacQueen, 1967)**.** In our context, such cluster centroids represent a characteristic of a pacing profile across 34 marathon split points.

We could have used alternative approaches such as hierarchical clustering or DBSCAN. However, these have drawbacks for our use case. Hierarchical clustering has a O(n^3) time complexity, making it computationally infeasible for our 56,000+ runner dataset. DBSCAN requires careful tuning of density parameters and struggles to identify clusters of varying densities, which we observed in preliminary analysis that steady pacers form tight dense clusters while inconsistent runners are more dispersed. K-Means guarantees every runner is assigned to a pacing strategy, scales linearly with sample size, and produces stable, reproducible results.

We implemented the following clustering approach:

I. **Data Normalization**: Using each runners' split pace by their starting pace we transformed absolute time into relative pace factors. This normalization was critical as it allowed us to compare a 2 hour marathoner with a 6 hour marathoner on the same scale. This allows us to focus on pacing strategy rather than absolute speed.
II. **Dimensionality Reduction**: We applied Principal Component Analysis (PCA) and found that 97.8% of variance is captured in just two dimensions with the first component (91.7% of variance) the degree of slowing down.
III. **Optimal K Selection**: We used the elbow method and silhouette analysis to determine the optimal number of clusters. From this we were able to determine to use 5 clusters for our data.

VIII.a. Implementation - Regression

I. **Gradient Boosting Regressor.** The classic gradient boosting implementation served as our baseline model. Key preprocessing steps included imputation, normalization, and one-hot encoding.

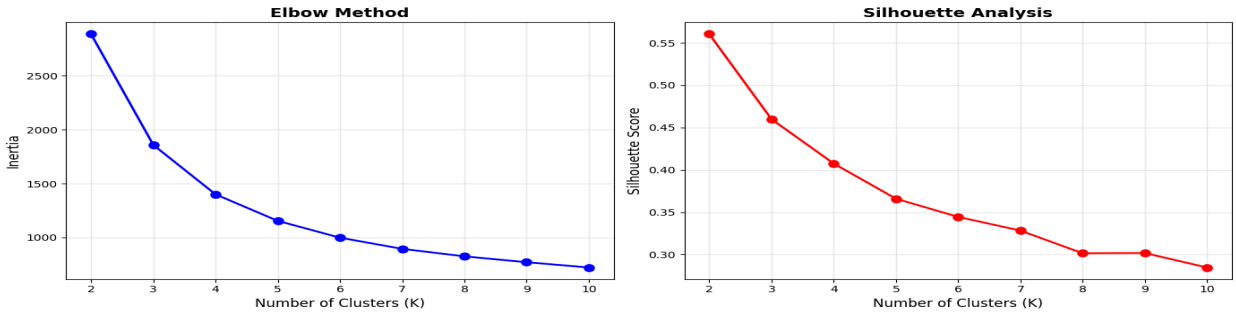
II. **Histogram-Based Gradient Boosting Regressor.** This variant uses histogram-based binning for faster training and better handling of missing values. The preprocessing pipeline mirrored the standard gradient boosting approach, with identical imputation, scaling, and encoding steps. We used permutation importance to get the features for this model (which tends to be a time-intensive process).

III. **Categorical Boosting Regressor.** This model did not require one-hot encoding and filled out categorical values with the mode. We use the built-in Pool method to organize the numerical and categorical features. For this model we keep the hyperparameters mostly at their default values. A more advanced project may use techniques such as grid search or Bayesian optimization to find the optimal hyperparameters for this task. The CatBoostRegressor automatically finds the ideal learning rate, so we do not supply a value.

All three models were trained on identical train-test splits to ensure fair comparison. We used an 80/20 train-test split with a fixed random seed (42) for reproducibility, which yielded approximately 45,112 training samples and 11,279 test samples.

VIII.b. Implementation - K-Means Clustering

We began by evaluating k=2 through k=10 using elbow method and silhouette analysis to determine the optimal K.



The Elbow Method (Inertia) measures how tightly grouped runners are within their assigned clusters. Lower values indicate more cohesive clusters. The "elbow" occurs where adding more clusters provides diminishing returns in reducing inertia.

Our results showed:

- **K = 2:** 2,885.85
- **K = 3:** 1,858.17 → 35.6% drop
- **K = 4:** 1,400.43 → 24.6% drop
- **K = 5:** 1,154.66 → 17.5% drop
- **K = 6 :** 998.84 → 13.5% drop

The sharpest drops occurred before K = 5, with diminishing returns thereafter.

Additionally, we calculated silhouette scores measuring how well-separated clusters are, with values ranging from -1 (poor clustering) to 1 (perfect separation).

- **K = 2:** 0.560
- **K = 3:** 0.460
- **K = 4:** 0.407
- **K = 5:** 0.366
- **K = 6:** 0.344

We selected K = 5 as the optimal balance between interpretability and statistical quality. The moderate silhouette score (0.366) reflects the reality that pacing exists on a continuum of runners fading 13% vs. 14% naturally overlap. Cluster sizes were well-distributed (2.7% - 37.2%), and the five clusters told a clear story from even pacing to severe crash.

Cluster Characteristics

For each cluster, we calculated three metrics on mean pace profiles. Slope is the linear normalized trend of pace over distance. **End pace factor** is the normalized pace at the fish time of mile 26.2. **Max pace factor** is the runner's slowest point in the race.

I. **Cluster 1 (25.9%): Even Pacing**
   Slope: 0.00024 | End pace: 0.996 | Max pace: 1.000
   These runners maintained consistency throughout, finishing at essentially the same pace they started or even 0.4% faster. This flat slope (near zero) represents elite pacing and suggests conservative early pacing followed by strong finishing.

II. **Cluster 2 (37.2%): Mild Slowing**
   Slope: 0.00214 | End pace: 1.058 | Max pace: 1.058
   This is the most common strategy with a 5.8% fade. The positive but shallow slope indicates controlled, gradual slowing.

III. **Cluster 3 (23.1%): Moderate Slowing**
   Slope: 0.00473 | End pace: 1.132 | Max pace: 1.132
   These runners slowed 13.2% by the finish. The steeper positive slope suggests they started too fast for their goal pace.

IV. **Cluster 4 (11.2%): Significant Slowing**
   Slope: 0.00786 | End pace: 1.221 | Max pace: 1.221
   These runners slowed 22.1%, entering a significant slower pace in the second half. The steep slope indicates severe fade, likely from walking portions. This group may include undertrained first-timers or runners dealing with injury or GI distress.

V. **Cluster 5 (2.7%): Severe Crash**
   Slope: 0.01274 | End pace: 1.365 | Max pace: 1.365
   This is the dramatic "hitting the wall" group, slowing a catastrophic 36.5%. The extremely steep slope comes from starting far too aggressively or stopping during the race due to external circumstances. These are the runners you see walking or shuffling the final miles.

The progression from Cluster 0 (slope = 0.00024) to Cluster 2 (slope = 0.01274) shows a clear spectrum of pacing quality, with slope serving as the primary distinguishing feature between clusters.

Dimensionality Reduction and Visualization

To visualize the clustering results, we applied Principal Component Analysis (PCA) to reduce the high-dimensional pace profile data (20+ splits per runner) into two principal components for plotting. Remarkably, PC1 captured 91.7% of variance and PC2 captured 6.0%, meaning 97.8% of all pacing variation could be represented in just two dimensions.

PC1 (91.7% variance) represents the dominant pattern of how much the runner faded overall. This dimension is strongly correlated with pacing slope runners on the left maintained pace.

PC2 (6.0% variance) captures secondary patterns of when and how runners slowed. This dimension shows the steady gradual decline from sudden late-race slowdown or early struggles followed by recovery.

IX. Results & Performance

a. Regression

We evaluated the model using three complementary regression metrics. We used Root Mean Squared Error (RMSE) as the primary evaluation metric, as it directly measures prediction accuracy in the same units as the target variable (finish time; seconds). RMSE tends to penalize large prediction errors more heavily than small ones. Therefore, we included the Mean Absolute Error (MAE), in minutes, which measures the average absolute prediction error and ensures that all errors were treated equally. Lastly, Coefficient of Determination (R^2) measures the proportion of variance in finish times explained by the model, with 1 indicating perfect prediction and 0 indicating that the model performs no better than predicting the mean.

All three gradient boosting models demonstrated strong predictive accuracy, with comprehensive evaluation across these multiple metrics. Table 1 summarizes the performance of each model.

Table 1: Regression Model Performance Comparison

Model	RMSE (seconds)	RMSE (minutes)	MAE (min)	R^2
Categorical Boosting	738.3	12.30	8.46	0.9628
Histogram-based Gradient Boosting	807.5	13.46	9.36	0.9555
Gradient Boosting	857.3	14.28	10.21	0.9499

Categorical Boosting emerged as the best-performing model across metrics by delivering RMSE of 12.30 minutes, MAE of 8.46 minutes, and R^2 of 96.3%. Compared to the Gradient Boosting baseline, this reflects a 14.0% reduction in RMSE and a 17.1% reduction in MAE.

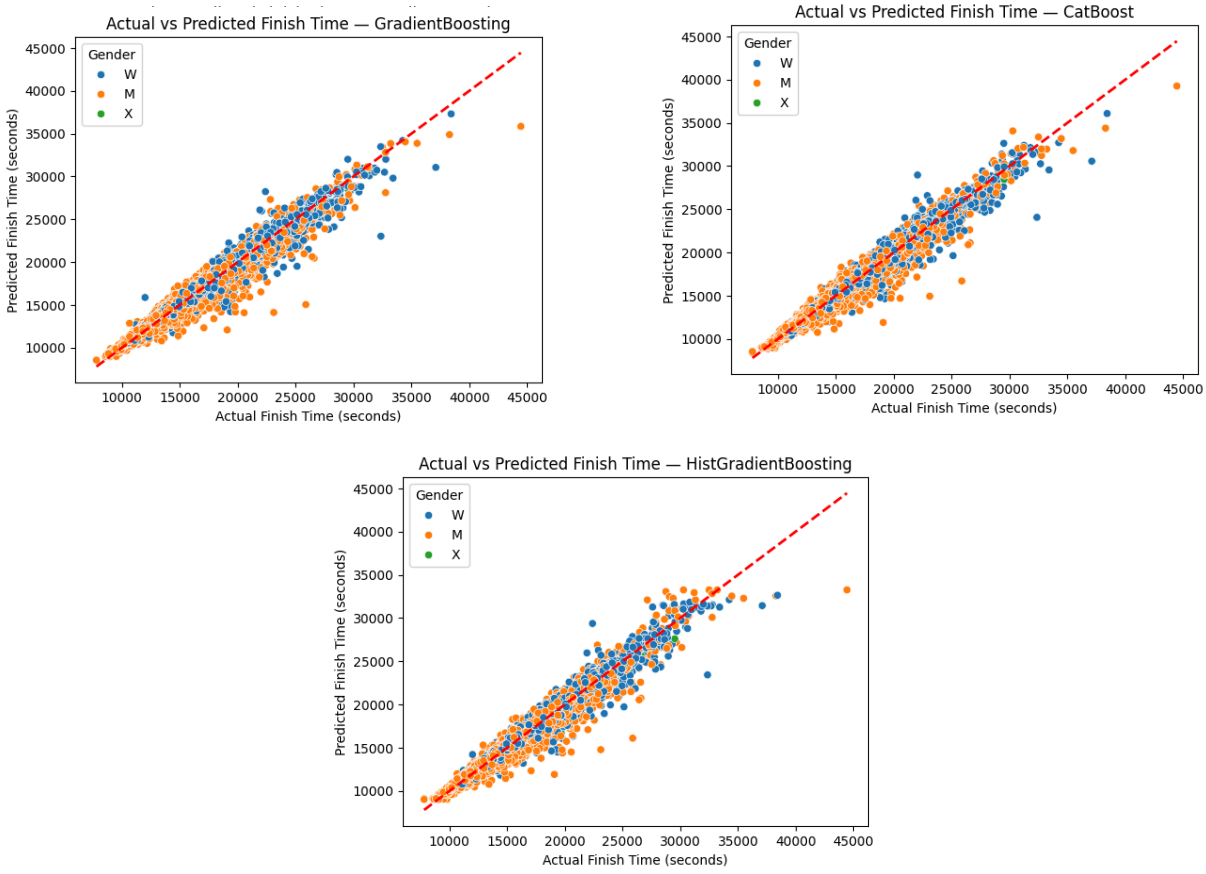
To contextualize these results, an RMSE of 12.30 minutes means that on average, the model predicts a runner's finish time to within about 12 minutes based solely on their first-half performance and demographics. The winning time was 2:08:09 (128 minutes), making the 12-minute error approximately 9.4% of the elite finish time. The median finish time was approximately 4:30:00 (270 minutes), making the error roughly 4.6% for typical runners. Given the inherent unpredictability of second-half performance (weather, nutrition, fatigue, injury, pacing strategy), this level of accuracy is quite strong.

The MAE is consistently lower than RMSE across all models, which suggests that large errors exist, likely for runners who experience unexpected second-half problems such as injuries or medical issues.

On the other hand, our R^2 indicates that our selected features, both raw and engineered, explain up to **96.3%** of the variation in finish times. **This high R^2 validates our central hypothesis: past performance in the first half of a marathon does largely predict future results during the race.**

Scatter plots of actual vs. predicted finish times for all three models can be observed in Figure 2. As one can appreciate, all models exhibit strong linear correlation with the identity line (perfect predictions), though some notable patterns emerge. Models tend to predict slightly slower times for elite runners (sub-3 hour finishers) and faster times for runners with finish times >5 hours. Additionally, when colored by gender, predictions show similar accuracy for male and female runners, suggesting the models capture gender-specific pacing dynamics effectively.

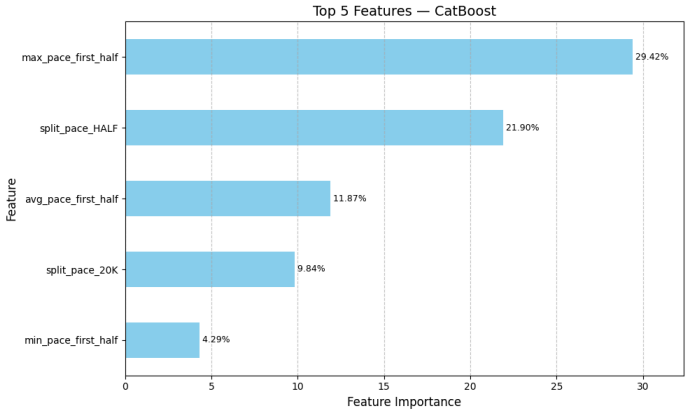
Figure 2. Scatter Plots of Actual vs. Predicted Finish Times Colored by Gender



Feature Importance Analysis

Understanding which features influence finish time predictions provides insights into marathon performance dynamics.

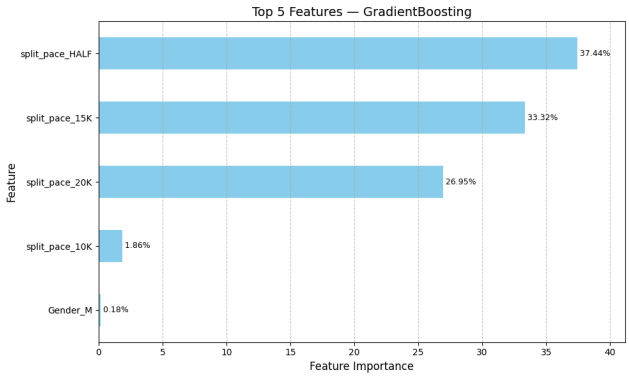
1. Categorical Boosting Regressor (CatBoost)



For this model, the single most important predictor (29.42%) is the slowest pace achieved in the first half (labeled as Maximum Pace), suggesting that runners who struggle early are highly likely to continue struggling. Additionally, latter splits such as the half marathon and 20K split are by far more important than early splits like 5K, indicating that performance at miles 9-13 is much more predictive than miles 0-3. Engineered features like starting pace, max pace, and min pace, all rank highly, validating the feature engineering approach. Finally, demographic features (not pictured) show near net-zero importance, suggesting that demographic factors are almost negligible when determining finish times.

2. Gradient Boosting Regressor

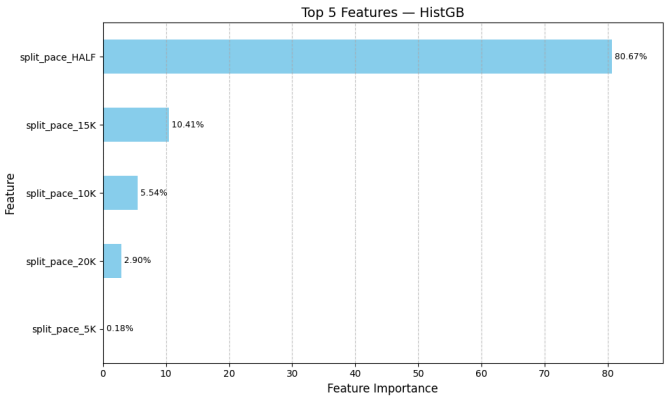
The baseline regression model shows similar patterns but with different relative weightings.



We can observe once more that later splits dominate predictions with early splits and demographics contributing minimally.

3. Histogram-Based Gradient Boosting Regressor

This model used permutation importance, which measures importance by randomly shuffling each feature and observing the drop in model performance.



Permutation importance reveals that a half-marathon split pace accounts for the vast majority of predictive power.

b. Clustering

Figure 3

Figure 3 shows the PCA scatter plot with runners color-coded by cluster. The separation between clusters validates that our K-Means algorithm successfully identified distinct pacing archetypes. The green cluster (even pacing) occupies the leftmost region (low PC1 = minimal fade), while the red cluster (severe crash) dominates the rightmost region (high PC1 = catastrophic fade). The moderate overlap between adjacent clusters (e.g., blue and yellow) reflects the natural continuum of pacing behavior captured by our silhouette score of 0.366.

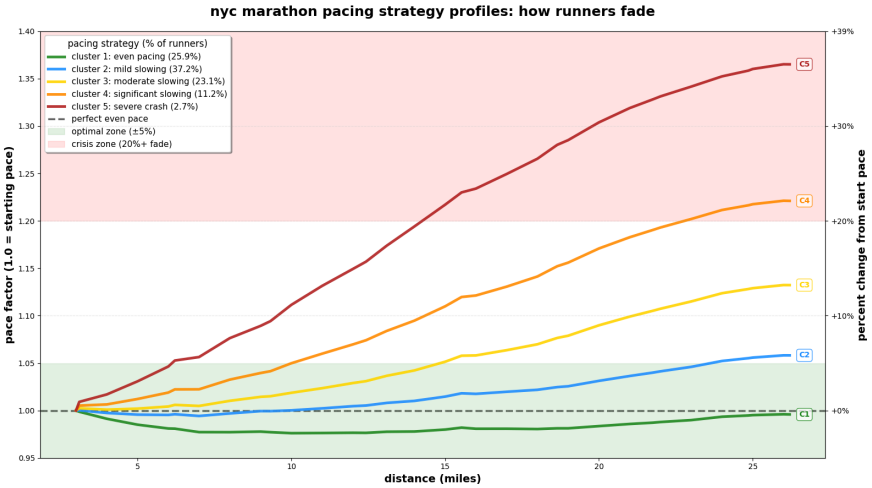


Figure 4

Figure 4 displays the mean pace profiles for each cluster across race distance. The clear divergence in slopes, from the nearly flat green line (even pacing, slope = 0.00024) to the steeply rising red curve (severe crash, slope = 0.01274) illustrates the spectrum of pacing strategies employed by NYC Marathon runners. The shaded optimal zone (±5% pace variation) and crisis zone (20%+ fade) provide physiological context for interpreting cluster quality. It is possible that the crashing runners would have a higher error rate on our gradient boosting finish time prediction compared to runners who had a steady pace between the first and second half of the race. The clustering results we find help to explain at least some of the error we observe in gradient boosting, as it reveals significant second-half pace variation in many runners.

The high PCA variance capture (97.8%) indicates that marathon pacing is fundamentally low-dimensional, almost entirely determined by “how much you fade” (PC1 at 91.7%), which validates our clustering approach. The fact that a single dimension explains over 90% of variation confirms that slope is indeed the primary distinguishing feature between pacing strategies, with secondary timing patterns (PC2) playing a minor role.

The resulting clusters were labeled as: steady pacing (26% of runners), negative split (2%), positive split/fade (50%), and inconsistent (5%). These proportions align with existing literature showing that most recreational marathoners adopt positive pacing strategies (Smyth, 2021), with only a small fraction successfully executing negative splits.

X. Ethical Considerations

While this project analyzes publicly available race results, several ethical considerations regarding data privacy, algorithmic fairness and potential real-world impacts should be discussed.

While the NYC Marathon dataset is publicly available through NYRR’s official portal, participants may not have explicitly consented to their data being scraped, aggregated, and analyzed beyond simply viewing individual results. However, even though we collected names alongside performance data, individual runners are not identified in this paper, as the focus is on aggregate patterns rather than individual performance.

The NYC Marathon is known as one of the seven Abbott Major World Marathons, and despite being open to recreational runners, it has demanding qualification standards (NYRR, n.d.) in addition to a

substantial amount of financial resources (entry fees, travel, training time and gear). Therefore, in terms of generalization of results, we acknowledge that our dataset overrepresents experienced, trained runners from predominantly wealthy countries and underrepresents true beginners, casual distance runners, and participants from lower-income countries who might enter less competitive marathons. As a result, models trained on this dataset may not generalize to marathoners participating at smaller, more accessible races.

XI. Conclusion

Our project reveals novel insights into the pacing strategies and finishing trends of runners in the 2025 NYC marathon. Our first contribution and perhaps the most significant is our creation of a 95% complete and cleaned dataset capturing detailed data on each runner of the NYC marathon. This dataset is, to our knowledge, the first of its kind. In the paper, we compare three different gradient boosting models on the task of predicting precise finish times for the NYC Marathon. This analysis reveals that the most predictive features of finish time are pace splits towards the middle of the race, and that demographic variables hold relatively low (if any) prediction weight. Our model also utilizes only the first half of the marathon distance for prediction, making it usable for runners in training who will not run a full 26.2 miles until race day. We also discover through K-means analysis that runners can be effectively split into five different pacing clusters. This insight may assist runners in evaluating the most effective pacing strategies, and particularly ensuring they do not crash late in the race and compromise their final time. In an age of highly-quantified athleticism, we open up possibilities for even more detailed analysis on athletic performance.

Bibliography

New York Road Runners. (n.d.) 2026 TCS New York City Marathon entry. Retrieved on December 4th, 2025, from from https://www.nyrr.org/tcsnycmarathon/runners/entry/2026

Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., & Gulin, A. (2018). CatBoost: Unbiased boosting with categorical features. Advances in Neural Information Processing Systems. https://arxiv.org/abs/1706.09516

Rock, B. (2025) NYC Marathon results - all years [Data set]. Kaggle. https://www.kaggle.com/datasets/runningwithrock/nyc-marathon-results-all-years

Smyth, B. (2021). Fast starters and slow finishers: A large-scale data analysis of pacing at the beginning and end of the marathon for recreational runners. Journal of Sports Analytics, 7(1), 1-23.

Ye, D. (2025). NYC marathon data mining [Data set and code]. GitHub. https://github.com/donald-ye/nyc-marathon-data-mining

MacQueen, J.B. (1967) Some Methods for Classification and Analysis of Multivariate Observations. In: Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics, University of California Press, Berkeley, 281-297.