A Data Driven Analytical Framework for Improving Endurance Runner's Performance

A Major Qualifying Project submitted to the faculty of Worcester Polytechnic Institute in partial fulfillment of the requirements for the Degrees of Bachelor of Science in Computer Science and Data Science

Authored By:

Ethan Rudometkin (DS)

Mason Perham (DS)

Advised By:

Chun-Kit Ngan

Abstract

The paper introduces a comprehensive analytical framework for predicting runners' endurance performance, addressing limitations in existing research. It includes a broad range of data including running gait, physiological, and anthropometric features. The feature selection method employed combines filter, wrapper, and embedded methods. Specifically, the Maximum Relevance Minimum Redundancy (MRMR) filter method, Recursive Feature Elimination (RFE) with a Random Forest model, and Ridge Regression were used to optimize feature selection. This multi-faceted approach ensures the selection of the most relevant and non-redundant features. The paper evaluates a variety of models, including SVM, ANN, AdaBoost, XGBoost, and deep learning models, for predicting runners' performance. The approach of testing multiple machine and deep learning models was intended to overcome the limitations of prior research that often restricted their analysis to a few models. By employing a more comprehensive range of models, this research aimed to identify the most accurate predictive model for runners' endurance performance. The evaluation metrics suggest that the XGBoost model yielded the most favorable results based on the error metrics provided: Mean Squared Error (MSE): 2.07, Root Mean Squared Error (RMSE): 1.44, and Mean Absolute Error (MAE): 1.12. The XGBoost model achieved the lowest values across these metrics, indicating its superior performance in predicting the outcome variable with the least amount of error.

ACKNOWLEDGEMENT

The team would like to express our sincere gratitude to our esteemed project advisor, Professor Chun-Kit Ngan, for his unwavering support and invaluable guidance throughout this project. His expertise and mentorship have been instrumental in shaping our work. Additionally, we extend our heartfelt appreciation to Dr. Eric Poon, Assistant Professor in the Department of Sports Science and Physical Education at Chinese University of Hong Kong, for his profound knowledge in High-Intensity Interval Training and his significant contributions to cardio-metabolic health research. His insights have greatly enriched our understanding in this area. We also thank Mr. Indy Ho, Assistant Professor in the Department of Management and Hospitality at Technological and Higher Education Institute of Hong Kong, for his extensive expertise in functional anatomy, sports injury management, and data science. His contributions have been invaluable in enhancing the depth and breadth of our project. Special recognition is also extended to RunLap Research Center for their support and collaboration throughout the duration of this project. We are truly grateful for the support, expertise, and dedication of all those mentioned, which have played a pivotal role in the success of this endeavor.

Contents

List of Figures

List of Tables

Chapter 1: Introduction

Endurance performance in running reflects an athlete's capacity to sustain a rapid pace over an extended period, such as during a marathon. It serves as a pivotal determinant of success in competitions, with superior endurance often correlating with better race outcomes. Elite marathon runners such as Eliud Kipchoge and Bashir Abdi often exhibit significantly higher VO2 max values, which is a key indicator of their superior endurance capabilities. The VO2 max of these athletes typically ranges between 70 and 85 ml/kg/min, compared to about 43.5 ml/kg/min in untrained individuals. These athletes consistently register among the top in various prestigious marathons worldwide clocking in at times near or below 2:03:00 in their races, illustrating their high endurance levels. Thus, understanding and investigating endurance performance is highly desirable [1].

To optimize endurance performance in runners, coaches employ a multifaceted approach focusing on structured training, nutrition, rest, and strength conditioning. This strategy involves crafting periodized training plans that progressively enhance endurance through varied workouts, including long runs, tempo runs, interval training, and recovery sessions. Such programs are designed to boost cardiovascular fitness, increase aerobic capacity, and elevated lactate threshold. Additionally, coaches offer dietary advice to ensure athletes are well-fueled for their endeavors, supporting both workout efficiency and recovery. Emphasizing the significance of rest and recovery, these plans incorporate sufficient sleep, active recovery, and scheduled rest days to prevent overtraining, facilitate physiological adaptation, and minimize injury risks. Strength and conditioning exercises are integrated to target key muscle groups relevant to running, improving strength, neuromuscular coordination, running economy, and delaying the onset of fatigue in prolonged activities. This comprehensive approach is essential for enhancing athletes' endurance capabilities and achieving superior performance in competitions [2].

While structured training programs and coaching strategies are instrumental in enhancing endurance performance in runners, they sometimes face limitations. Human coaches might struggle with the integration and interpretation of complex data from physiological, biomechanical, and psychological sources, which can limit the effectiveness of training programs. Artificial Intelligence (AI), on the other hand, is adept at processing this multifaceted data to extract deep insights into an athlete's performance, fatigue levels, and injury risks capabilities that extend beyond human analytical capacity. Additionally, coaches often face challenges in monitoring and analyzing long-term performance trends, particularly with multiple athletes. AI excels in managing and analyzing vast datasets over extended periods, helping to identify trends and patterns that are crucial for developing effective, long-term athletic development plans. This advanced data handling enables a more nuanced and scientifically informed approach to training that can significantly enhance athletic performance.

The integration of AI and Data Science methodologies in running has revolutionized the way that endurance performance is enhanced, offering a structured and analytical complement to the traditional, multifaceted strategies employed by coaches. At the forefront of this scientific approach is the collection of empirical data, encompassing both anaerobic and aerobic capacities through treadmill tests, alongside the gathering of biomechanical parameters and demographic data. With this data the application of various machine learning models and statistical methods designed to decode the complexities of endurance running. Machine learning models, such as Convolutional Neural Networks (CNNs) and Multilayer Perceptrons (MLPs), delve into biomechanical analysis, while regression analysis and supervised learning algorithms like Random Forest and Support Vector Regression tackle the prediction of athletes' anaerobic thresholds and performances in events like half-marathons [5,6,7]. Additionally, Recurrent Neural Networks (RNNs) are utilized for estimating lactate thresholds, and advanced algorithms, including Principal Component Analysis (PCA) and XGBoost regression trees, contribute to the understanding of musculoskeletal modeling and key predictors of running velocity [3,9]. By leveraging the predictive power of these models, coaches can design periodized training plans that not only progressively enhance an athlete's endurance but also optimize their overall performance in competitions. This blend of empirical data analysis and sophisticated modeling techniques with traditional coaching wisdom underscores the transformative impact of technology in sports training. It presents a data-driven pathway to improving endurance performance in runners, combining the experiential knowledge of coaches with the precision of modern science to achieve superior competition outcomes.

However, common limitations in these studies include small sample sizes, reducing generalizability, and reliance on self-reported data, which can introduce inaccuracies. Additionally, the model development is static and does not allow for updates with new data, limiting long-term applicability. Many papers either focus on a narrow range of features or when considering a wide range do not employ feature selection methods to identify the most impactful features, thereby limiting the performance and effectiveness of their models. Prior research also restricts themselves to a small set of machine learning or deep learning models, not exploring a broader spectrum to identify the most suitable model for their data. Lastly, no proper tool can support and help coaches understand and train their running athletes.

To mitigate the above shortcomings, our research work proposes the development of a data-driven analytical framework to support the improvement of runners' endurance performance.

Specifically, our contributions are four-fold: (1) devise a data-driven, track-and-field domain-guided pipeline to select the best set of predictors, among running gaits (Vertical Force, Contact Time, Running Economy), physiology (Vo2 Max, Lactate Threshold, Heart Rate), and anthropometry (Height, Weight, Body Fat%) factors, to predict runners' endurance performance; (2) use the state-of-the-art ensemble-and-ranking-based technique to select the best set of relevant and non-redundant features for the downstream Machine Learning (ML) regressor constructions. The approaches involve in this technique include the filter method (Maximum Relevance Minimum Redundancy), the wrapper method (Recursive Feature Elimination with Random Forest), and the embedded method (Ridge Regression) that respectively provide the rank for each feature and then select the final list of features among them based upon their average ranks; (3) develop and evaluate three different branches of ML regressors, including single-based (SVM and ANN), ensemble-based (AdaBoost and XGBoost), and deep learning-based (Feedforward Neural Networks) models to conduct an experimental analysis on our existing study data (131 professional runners) collected by our investigators in a sport research organization in Hong Kong. In this study, we demonstrate that our ensemble-based XGBoost model outperforms the other two ML methods by 98.7?% on average in terms Mean Squared Error, Root Mean Squared Error, and Mean Absolute Error; and (4) design and implement a user-friendly dynamic dashboard to support professional coaches to illustrate the interrelations among the features within and across the above three factors, as well as to provide the insights to runners based upon their predicted endurance performance by our XGBoost model.

Following the introduction, the paper is organized as follows: Chapter 2 conducts a detailed literature review focusing on prior research and methodologies in the realm of endurance performance analytics. We examine various studies that have utilized physiological, biomechanical, and data-driven approaches to predict and enhance runner performance. Chapter 3 describes our data-driven analytical framework, detailing the processes of data collection, cleaning, and preprocessing which are crucial for the integrity and usability of the data used in our analyses. In Chapter 4, we delve into the methodologies for data cleaning and preprocessing in-depth, ensuring that the data set is robust and reliable for further analysis. Chapter 5 presents a descriptive analysis of the runners' characteristics, providing insights into the diverse variables captured in our study and their preliminary relationships. Chapter 6 explores the interrelation and selection of features, employing advanced statistical and machine learning techniques to refine the predictive model. Chapter 7 discusses the development of our predictive models, where we evaluate various machine learning and deep learning approaches to find the most accurate predictive model for endurance performance. Chapter 8 is dedicated to the evaluation of these models, comparing their performance and discussing their implications. In Chapter 9, we introduce a dynamic and interactive Tableau dashboard that illustrates the complex interrelationships among the analyzed features and provides actionable insights based on the predictive models developed. Finally, Chapter 10 concludes the paper by summarizing our findings, discussing the implications of our research, and suggesting directions for future studies in this evolving field of sports analytics.

Chapter 2: Literature Review

During the research phase, we reviewed many different papers relating to predicting running performance. We grouped these papers into three groups. The first group consists of papers that focus on anaerobic and aerobic data. This data is similar to our physiological data. The second group of papers focus on biomechanical analysis. This data is similar to our running gait data. The third group of papers focuses on testing different machine learning models to make predictions. From this group of papers, we created the list of models that we tested.

2.1 Anaerobic and Aerobic Power Measurement

The studies in this group, which concentrates on anaerobic and aerobic power measurement, bases their assumptions on the empirical data obtained using treadmill tests to assess the maximum speeds that are supported by anaerobic and aerobic systems. The direct approach of measuring physiological capacity within a controlled environment is one notable benefit of this method.

Directly measuring anaerobic and aerobic powers is an advantage in these studies, as it provides physical evidence about the key abilities for endurance running. Therefore, the empirical data collected from treadmill tests gives a simplistic and immediate impression of how well-trained athlete's bodies can cope with the pressure exerted during physical exercises

Conversely, the small sample size used in these studies limits the generalizability of results. A major drawback to only using traditional mathematics-based modeling together with empirical knowledge is that it overlooks potential benefits from advanced machine learning approaches. Consequently, such studies may be deprived of any machine learning or deep learning approaches hence they might fail to capture nonlinear and complex relationships that could exist between different variables when dealing with a bigger number of datasets; thus hindering thorough understanding of factors associated with endurance performance [3,4,5].

2.2 Biomechanical Analysis with IMUs

The studies in this group use Inertial Measurement Units (IMUs) to capture detailed biomechanical data via sensors attached to runners. This method involves the use of accelerometer, gyroscope and magnetometer readings to examine minute constituents of running mechanics.

The richness of data from IMUs is one merit they have over other approaches that makes it possible to analyze things such as average vertical loading rate and stride length more subtly. With IMUs, one can obtain details that aid in getting an overall picture on how athletes move when running; this may help uncover important facts about performance optimization and injury prevention.

However, some machine learning models especially Convolutional Neural Networks (CNNs) and Multilayer Perceptrons (MLPs) are black boxes making them difficult to interpret. As well as this, using a relatively small number of samples and only considering limited variables may prevent comprehensive insights from being gained through biomechanical analysis [6,7,8].

2.3 Machine Learning for Performance Prediction

The studies in this group focused on applying various machine learning techniques to predict endurance performance. It involves models like RunNet-CNN, RunNet-MLP, GBDT, ANN, KNN, SVR and others with emphasis being placed on the use of modern algorithms for precise predictions.

The key benefit of using machine learning is its potential to make accurate predictions and uncover complex relationships in data. For instance, CNNs and MLPs can detect intricate patterns that conventional methods would miss thereby providing a better understanding of the determinants of performance.

Nevertheless, some machine learning models are “black box” meaning their interpretability is hindered hence raising questions about transparency into decision-making processes. Furthermore, because they belong to a broad range of athletes, including runners everywhere else within this population will be compromised for overfitting risk when dealing with small sample sizes. There must then be diverse datasets that capture individual physiology variations with regards to training and performance as central to success of these approaches by machine learning [9,10,11].

Chapter 3: Data Driven Analytical Framework

Our data-driven analytical framework is a comprehensive approach that begins with data collection and cleaning processes, ensuring the dataset's accuracy and completeness. Through a series of assessments and tests, you gather detailed information about runners' characteristics, including body composition, running gait parameters, and cardiovascular fitness. Data preprocessing steps, such as handling missing values, correcting errors, and transforming categorical variables, prepare the dataset for analysis. We then explored key parameters and distributions to gain insights into gender differences, running styles, and physiological indicators among the runners. Visualization techniques, such as pie charts, histograms, and scatter plots, aid in presenting the data effectively, highlighting patterns and trends.

Feature interrelation and selection play crucial roles in refining the dataset for predictive modeling. Utilizing techniques like Pearson correlation tests, Maximum Relevance Minimum Redundancy (MRMR), Recursive Feature Elimination (RFE), and Ridge Regression, you identify highly correlated features and select the most relevant ones for further analysis. Model development encompasses various machine learning and deep learning techniques, including Support Vector Regression (SVR), Artificial Neural Networks (ANN), AdaBoost, XGBoost, and Feed Forward Neural Networks. Each model is carefully trained and validated using appropriate methodologies to ensure robust performance in predicting runners' maximum speed.

The development of Tableau dashboards adds an interactive layer to the analytical framework, allowing users to explore the data dynamically. Features such as prediction sheets, min/max overviews, variable relationship analyses, and sheets that show interrelationship between features provide users with comprehensive tools for understanding the dataset, making informed decisions, and predicting performance outcomes.

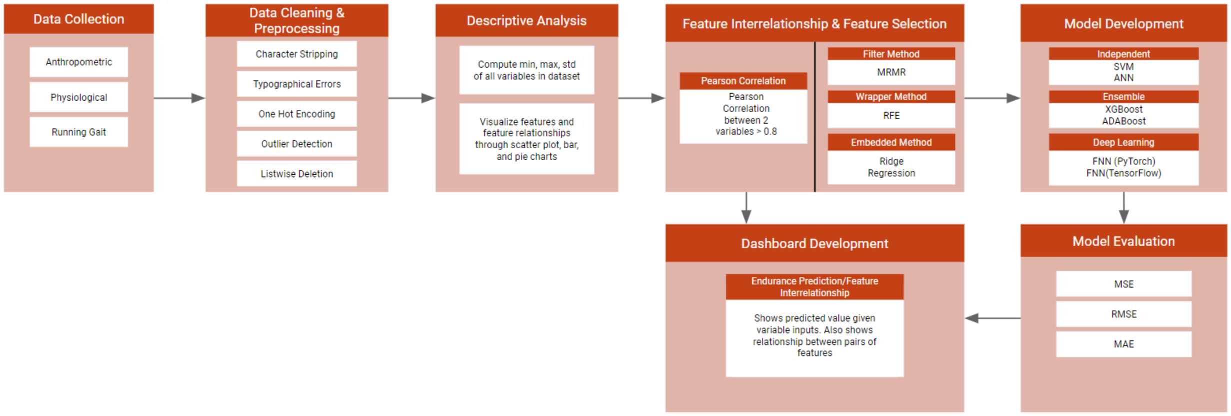

Data Collection
Anthropometric
Physiological
Running Gait
Data Cleaning & Preprocessing
Character Stripping
Typographical Errors
One Hot Encoding
Outlier Detection
Listwise Deletion
Descriptive Analysis
Compute min, max, std of all variables in dataset
Visualize features and feature relationships through scatter plot, bar, and pie charts
Feature Interrelationship & Feature Selection
Pearson Correlation
Pearson Correlation between 2 variables > 0.8
Filter Method
MRMR
Wrapper Method
RFE
Embedded Method
Ridge Regression
Model Development
Independent
SVM
ANN
Ensemble
XGBoost
ADABoost
Deep Learning
FNN (PyTorch)
FNN(TensorFlow)
Dashboard Development
Endurance Prediction/Feature Interrelationship
Shows predicted value given variable inputs. Also shows relationship between pairs of features
Model Evaluation
MSE
RMSE
MAE

Figure 1. Data Driven Analytical Framework Process

Chapter 4: Data Collection/Data Cleaning & Preprocessing

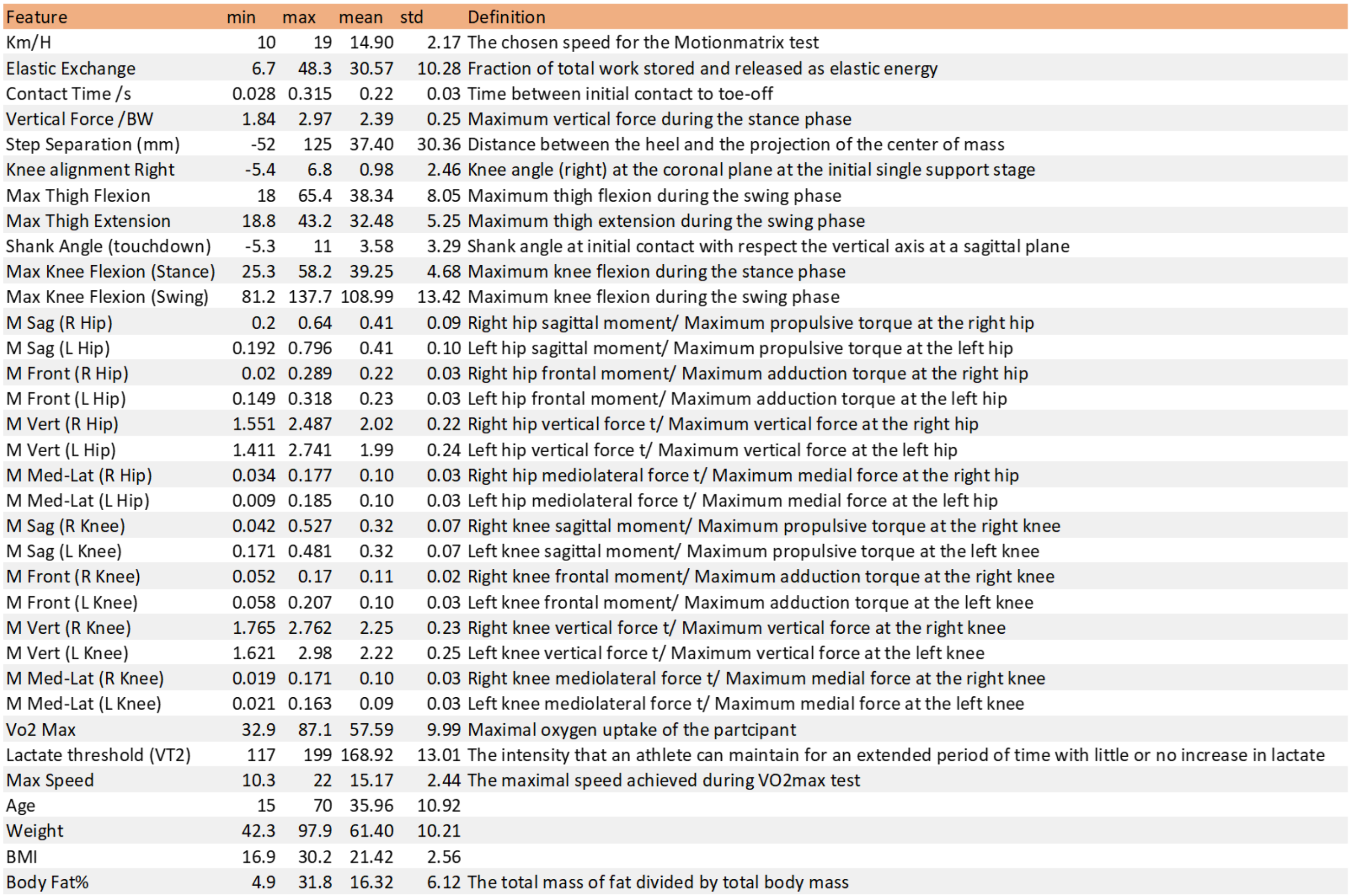
The data collection process for this project involved a single laboratory session for each participant, during which they were instructed to abstain from strenuous activities for at least 24 hours prior to data collection and wore their usual running attire, including shoes. Three distinct assessments were conducted on the participants.

First, the Body Composition and Anthropometry assessment utilized a Bioelectrical Impedance Analysis (BIA) scale (InBody 270, Inbody, Korea) to measure body weight, body mass index (BMI), and body fat percentage. Measurements were taken with participants wearing lightweight clothing and bare feet, following the manufacturer's instructions for data input. The Running Gait Parameters assessment employed the MotionMetrix system in conjunction with two Kinect sensors (Microsoft, Washington, DC, USA) positioned as recommended by the manufacturer. Subjects had a brief accommodation period on a treadmill at a self-selected pace, followed by a one-minute run at their usual 10 km/h pace, with data collection occurring during the final 30 seconds. This assessment provided analyses of various kinetic and kinematic variables. Participants underwent a Cardiovascular Fitness Test (VO2max Test), involving a standardized ramp treadmill test to evaluate cardiovascular fitness (VO2max) and HRmax. The treadmill speed began at 4.8 km/h on a 2% incline and increased every 2 minutes until volitional exhaustion. Criteria for VO2max achievement included a respiratory exchange ratio >= 1.15, VCO2 line crossing over VO2 line, and HR exceeding 220 minus the participant's age. Gas analysis data (VO2 and VCO2) were continuously recorded using a metabolic analyzer (PNOĒ metabolic analysis system, USA), and HR data were captured through HR telemetry (H10 Sensor, Polar, Finland), with HRmax determined as the highest recorded value at test completion. Data accuracy was ensured by adhering to the manufacturer's guidelines for calibration, clothing, and lighting conditions.

Several actions were taken to clean and analyze the dataset effectively. Initially, data entries missing Body Fat Percentage were excluded to ensure data completeness. Typographical errors, such as misplaced commas, were corrected, and unnecessary characters, like the percent sign in Elastic Exchange, were removed. Categorical variables like strike type (Rear Foot, Mid Foot, Fore Foot) and Gender (Male, Female) were transformed using one-hot encoding, a method to convert categorical data into binary vectors. Scatter plots were created for each variable to visualize their distribution. These plots were generated by plotting each variable against a single value axis, providing a clear view of the data spread and any potential anomalies. To handle these outliers, we created a function to calculate the interquartile range (IQR) for a given column and identify outliers based on the IQR method. Outliers were detected for each numerical column in the dataset, and the results were compiled into a single Data Frame, which was then exported to a CSV file. The CSV file containing the identified outliers was sent to the RunLap team for a thorough review. Their objective was to verify the accuracy of the outlier results and to examine if any data had been incorrectly entered. After a detailed analysis, the RunLap team returned the file with necessary corrections, ensuring that the data was precise and reliable for further analysis and use.

Chapter 5: Descriptive Analysis Of Runners Characteristics

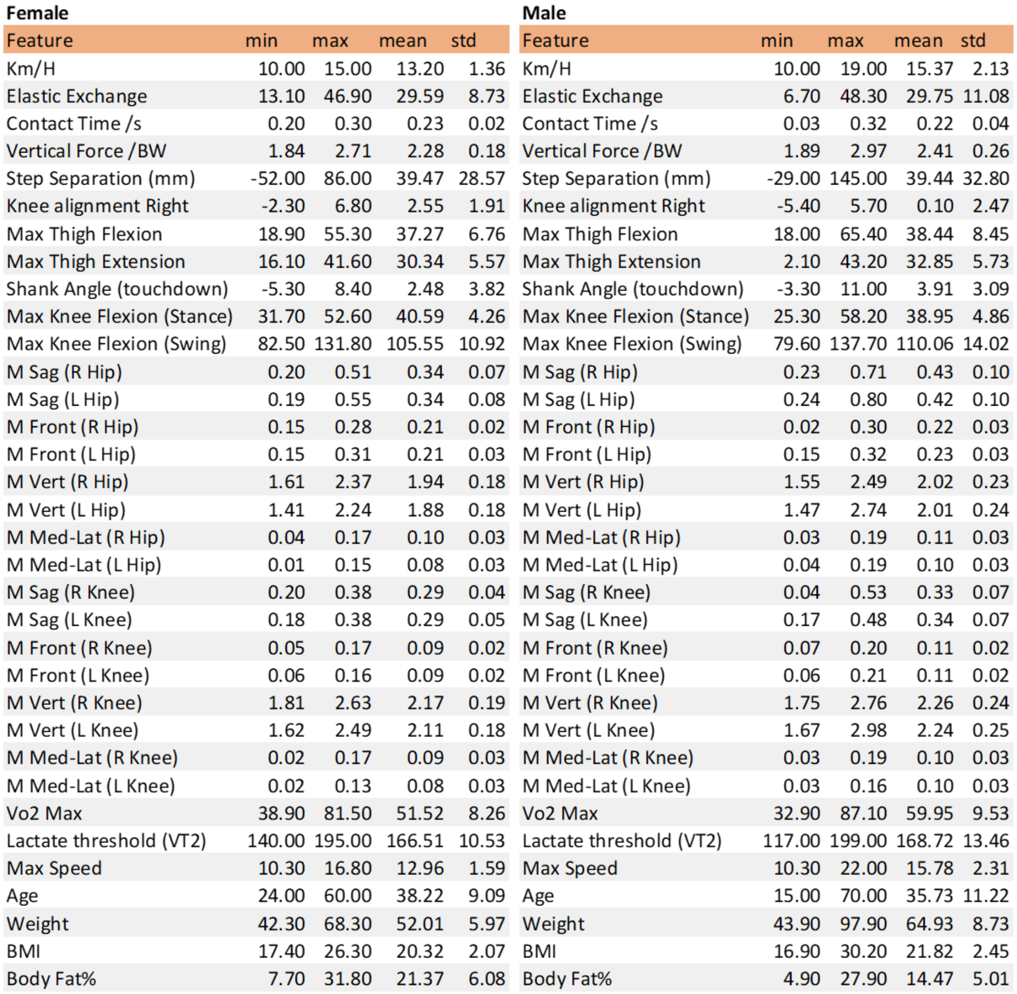
The dataset we used contained 55 features that describe the running style and abilities of 167 runners examined. Figure below shows a table displaying the numerical features used in the model. The table also shows the mean value, minimum value, maximum value, standard deviation, and description of each of the features. The table is able to capture the summary of the numerical data and make it easy to read and interpret.



Feature	min	max	mean	std	Definition
Km/H	10	19	14.90	2.17	The chosen speed for the Motionmatrix test
Elastic Exchange	6.7	48.3	30.57	10.28	Fraction of total work stored and released as elastic energy
Contact Time /s	0.028	0.315	0.22	0.03	Time between initial contact to toe-off
Vertical Force /BW	1.84	2.97	2.39	0.25	Maximum vertical force during the stance phase
Step Separation (mm)	-52	125	37.40	30.36	Distance between the heel and the projection of the center of mass
Knee alignment Right	-5.4	6.8	0.98	2.46	Knee angle (right) at the coronal plane at the initial single support stage
Max Thigh Flexion	18	65.4	38.34	8.05	Maximum thigh flexion during the swing phase
Max Thigh Extension	18.8	43.2	32.48	5.25	Maximum thigh extension during the swing phase
Shank Angle (touchdown)	-5.3	11	3.58	3.29	Shank angle at initial contact with respect the vertical axis at a sagittal plane
Max Knee Flexion (Stance)	25.3	58.2	39.25	4.68	Maximum knee flexion during the stance phase
Max Knee Flexion (Swing)	81.2	137.7	108.99	13.42	Maximum knee flexion during the swing phase
M Sag (R Hip)	0.2	0.64	0.41	0.09	Right hip sagittal moment/ Maximum propulsive torque at the right hip
M Sag (L Hip)	0.192	0.796	0.41	0.10	Left hip sagittal moment/ Maximum propulsive torque at the left hip
M Front (R Hip)	0.02	0.289	0.22	0.03	Right hip frontal moment/ Maximum adduction torque at the right hip
M Front (L Hip)	0.149	0.318	0.23	0.03	Left hip frontal moment/ Maximum adduction torque at the left hip
M Vert (R Hip)	1.551	2.487	2.02	0.22	Right hip vertical force t/ Maximum vertical force at the right hip
M Vert (L Hip)	1.411	2.741	1.99	0.24	Left hip vertical force t/ Maximum vertical force at the left hip
M Med-Lat (R Hip)	0.034	0.177	0.10	0.03	Right hip mediolateral force t/ Maximum medial force at the right hip
M Med-Lat (L Hip)	0.009	0.185	0.10	0.03	Left hip mediolateral force t/ Maximum medial force at the left hip
M Sag (R Knee)	0.042	0.527	0.32	0.07	Right knee sagittal moment/ Maximum propulsive torque at the right knee
M Sag (L Knee)	0.171	0.481	0.32	0.07	Left knee sagittal moment/ Maximum propulsive torque at the left knee
M Front (R Knee)	0.052	0.17	0.11	0.02	Right knee frontal moment/ Maximum adduction torque at the right knee
M Front (L Knee)	0.058	0.207	0.10	0.03	Left knee frontal moment/ Maximum adduction torque at the left knee
M Vert (R Knee)	1.765	2.762	2.25	0.23	Right knee vertical force t/ Maximum vertical force at the right knee
M Vert (L Knee)	1.621	2.98	2.22	0.25	Left knee vertical force t/ Maximum vertical force at the left knee
M Med-Lat (R Knee)	0.019	0.171	0.10	0.03	Right knee mediolateral force t/ Maximum medial force at the right knee
M Med-Lat (L Knee)	0.021	0.163	0.09	0.03	Left knee mediolateral force t/ Maximum medial force at the left knee
Vo2 Max	32.9	87.1	57.59	9.99	Maximal oxygen uptake of the partcipant
Lactate threshold (VT2)	117	199	168.92	13.01	The intensity that an athlete can maintain for an extended period of time with little or no increase in lactate
Max Speed	10.3	22	15.17	2.44	The maximal speed achieved during VO2max test
Age	15	70	35.96	10.92	
Weight	42.3	97.9	61.40	10.21	
BMI	16.9	30.2	21.42	2.56	
Body Fat%	4.9	31.8	16.32	6.12	The total mass of fat divided by total body mass

Table 1. Descriptive Analysis of Numerical Features

The provided tables below summarize data on female and male runners, offering comparative insights into their running styles and abilities. Key parameters such as Km/H, Elastic Exchange, and Contact Time highlight differences in speed, energy efficiency, and stride mechanics between genders. While the mean running speeds are slightly higher for males, both genders exhibit comparable energy utilization efficiency. Joint angles and forces during running show similar ranges for both genders, though individual values suggest the need for personalized training approaches. Physiological indicators like VO2 Max and Lactate Threshold are higher for males, aligning with typical gender differences in aerobic capacity. Physical fitness measures, including Weight, BMI, and Body Fat%, reflect expected gender variations, with males presenting higher weight and BMI but lower Body Fat%.



Female					Male				
Feature	min	max	mean	std	Feature	min	max	mean	std
Km/H	10.00	15.00	13.20	1.36	Km/H	10.00	19.00	15.37	2.13
Elastic Exchange	13.10	46.90	29.59	8.73	Elastic Exchange	6.70	48.30	29.75	11.08
Contact Time /s	0.20	0.30	0.23	0.02	Contact Time /s	0.03	0.32	0.22	0.04
Vertical Force /BW	1.84	2.71	2.28	0.18	Vertical Force /BW	1.89	2.97	2.41	0.26
Step Separation (mm)	-52.00	86.00	39.47	28.57	Step Separation (mm)	-29.00	145.00	39.44	32.80
Knee alignment Right	-2.30	6.80	2.55	1.91	Knee alignment Right	-5.40	5.70	0.10	2.47
Max Thigh Flexion	18.90	55.30	37.27	6.76	Max Thigh Flexion	18.00	65.40	38.44	8.45
Max Thigh Extension	16.10	41.60	30.34	5.57	Max Thigh Extension	2.10	43.20	32.85	5.73
Shank Angle (touchdown)	-5.30	8.40	2.48	3.82	Shank Angle (touchdown)	-3.30	11.00	3.91	3.09
Max Knee Flexion (Stance)	31.70	52.60	40.59	4.26	Max Knee Flexion (Stance)	25.30	58.20	38.95	4.86
Max Knee Flexion (Swing)	82.50	131.80	105.55	10.92	Max Knee Flexion (Swing)	79.60	137.70	110.06	14.02
M Sag (R Hip)	0.20	0.51	0.34	0.07	M Sag (R Hip)	0.23	0.71	0.43	0.10
M Sag (L Hip)	0.19	0.55	0.34	0.08	M Sag (L Hip)	0.24	0.80	0.42	0.10
M Front (R Hip)	0.15	0.28	0.21	0.02	M Front (R Hip)	0.02	0.30	0.22	0.03
M Front (L Hip)	0.15	0.31	0.21	0.03	M Front (L Hip)	0.15	0.32	0.23	0.03
M Vert (R Hip)	1.61	2.37	1.94	0.18	M Vert (R Hip)	1.55	2.49	2.02	0.23
M Vert (L Hip)	1.41	2.24	1.88	0.18	M Vert (L Hip)	1.47	2.74	2.01	0.24
M Med-Lat (R Hip)	0.04	0.17	0.10	0.03	M Med-Lat (R Hip)	0.03	0.19	0.11	0.03
M Med-Lat (L Hip)	0.01	0.15	0.08	0.03	M Med-Lat (L Hip)	0.04	0.19	0.10	0.03
M Sag (R Knee)	0.20	0.38	0.29	0.04	M Sag (R Knee)	0.04	0.53	0.33	0.07
M Sag (L Knee)	0.18	0.38	0.29	0.05	M Sag (L Knee)	0.17	0.48	0.34	0.07
M Front (R Knee)	0.05	0.17	0.09	0.02	M Front (R Knee)	0.07	0.20	0.11	0.02
M Front (L Knee)	0.06	0.16	0.09	0.02	M Front (L Knee)	0.06	0.21	0.11	0.02
M Vert (R Knee)	1.81	2.63	2.17	0.19	M Vert (R Knee)	1.75	2.76	2.26	0.24
M Vert (L Knee)	1.62	2.49	2.11	0.18	M Vert (L Knee)	1.67	2.98	2.24	0.25
M Med-Lat (R Knee)	0.02	0.17	0.09	0.03	M Med-Lat (R Knee)	0.03	0.19	0.10	0.03
M Med-Lat (L Knee)	0.02	0.13	0.08	0.03	M Med-Lat (L Knee)	0.03	0.16	0.10	0.03
Vo2 Max	38.90	81.50	51.52	8.26	Vo2 Max	32.90	87.10	59.95	9.53
Lactate threshold (VT2)	140.00	195.00	166.51	10.53	Lactate threshold (VT2)	117.00	199.00	168.72	13.46
Max Speed	10.30	16.80	12.96	1.59	Max Speed	10.30	22.00	15.78	2.31
Age	24.00	60.00	38.22	9.09	Age	15.00	70.00	35.73	11.22
Weight	42.30	68.30	52.01	5.97	Weight	43.90	97.90	64.93	8.73
BMI	17.40	26.30	20.32	2.07	BMI	16.90	30.20	21.82	2.45
Body Fat%	7.70	31.80	21.37	6.08	Body Fat%	4.90	27.90	14.47	5.01

Table 2. Descriptive Analysis of Numerical Features by Gender

The "Strike Type Distribution" graph presents an analysis of foot strike patterns among runners, segregated by gender as well as a combined overview. It reveals that Mid-Foot striking is predominant across the general running population at 38.52%, closely followed by Rear-Foot striking at 41.52%, with Fore-Foot striking being relatively less common at 18.96%. When disaggregated by gender, female runners show a preference for Mid-Foot striking (44.4%), with Rear-Foot (31.1%) and Fore-Foot (24.4%) striking following respectively. Conversely, male runners exhibit a higher incidence of Rear-Foot striking (45.6%), with Mid-Foot striking at 37.70%, and Fore-Foot striking at 16.7%. This data suggests a notable difference in striking preference between genders, with females favoring Mid-Foot and males favoring Rear-Foot striking. Fore-Foot striking is the least favored among both, albeit more common in females than males.

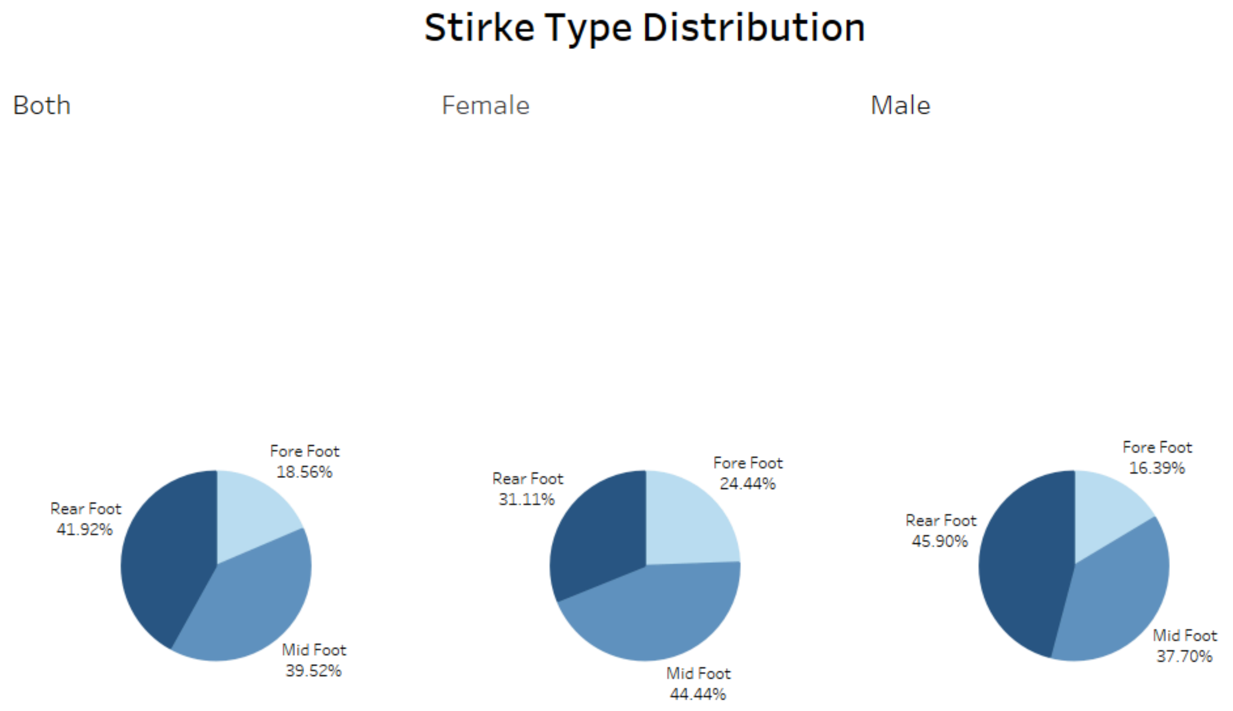


Figure 2. Strike Type Distribution Pie Charts

The "Max Speed Distribution" graph is a dual histogram displaying the distribution of maximum speeds achieved by male and female runners. The top histogram represents the total count of all runners, while the bottom histogram provides a gender breakdown for each speed interval. From the total distribution, we observe that the most common maximum speed interval among all runners is 15-16 km/h, with 60 individuals falling into this category. The distribution appears right-skewed, indicating that fewer runners achieve higher maximum speeds. Very few runners reach speeds above 20 km/h. When examining the gender-specific histogram, it's noticeable that for males, the most frequent maximum speed interval is also 15-16 km/h, with 46 males falling in this range. For females, the highest count is at a lower speed interval of 13-14 km/h, with 14 females reaching this maximum speed. The histograms suggest that on average, male runners tend to reach higher maximum speeds compared to female runners. In both genders, there's a significant drop in the number of individuals with maximum speeds over 16 km/h.

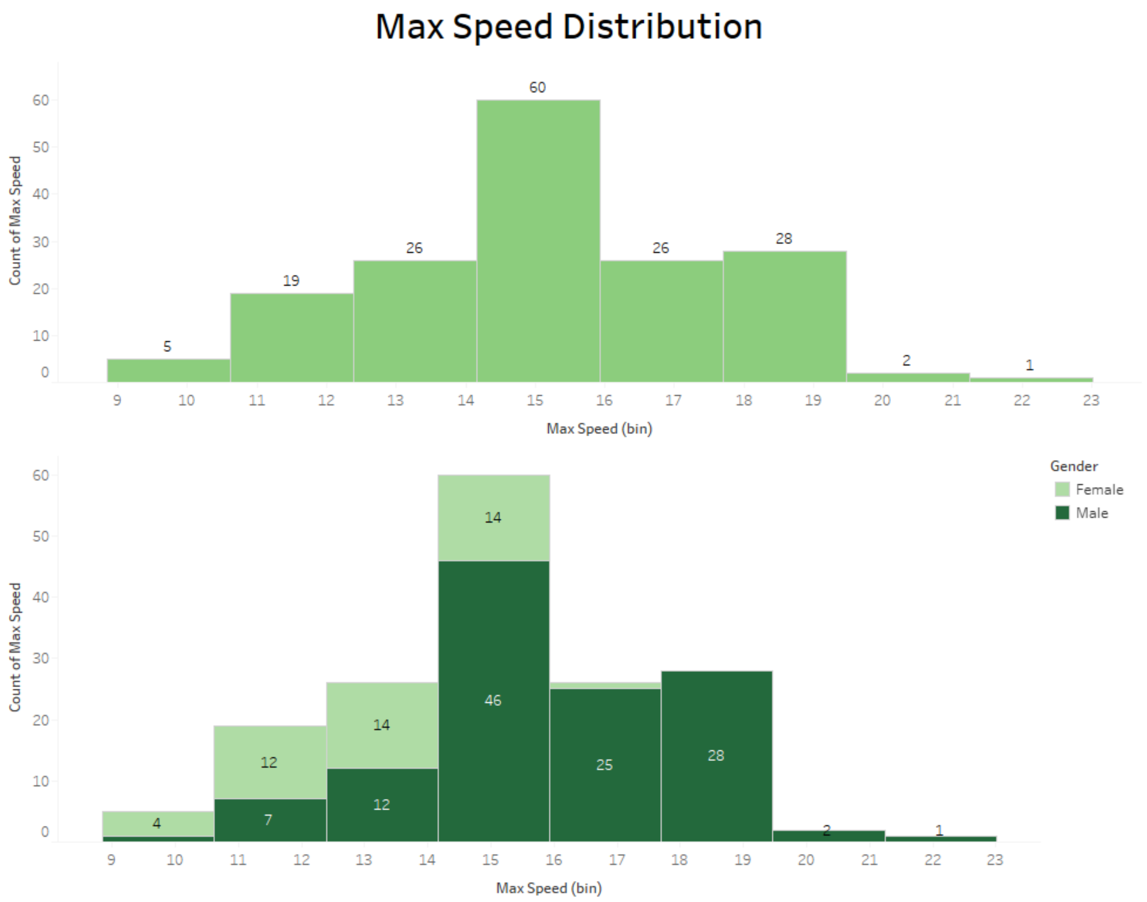
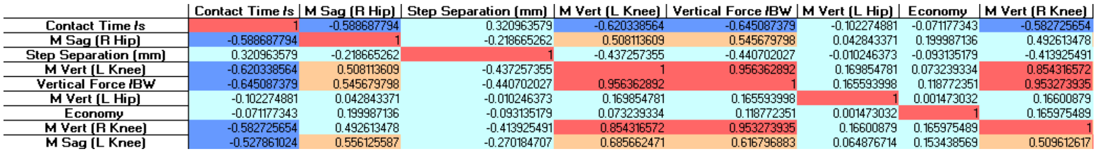


Figure 3. Max Speed Prediction Bar Plots

Chapter 6: Feature Interrelation and Selection Approach

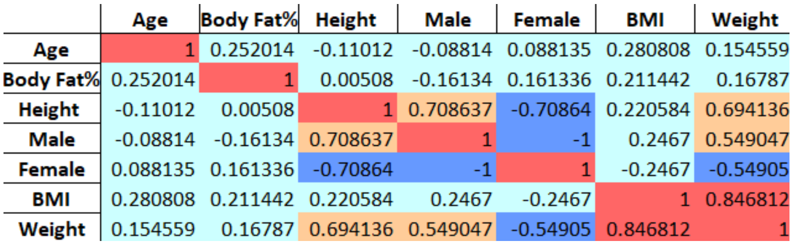
6.1 Feature Interrelation

To determine the interrelationships between the features, Pearson correlation tests were employed. Initially, the Pearson correlation tests were run between all features in the dataset. Each feature was compared to every other feature one at a time to determine how correlated they were. The correlation test results in a number between -1 and 1, with -1 being a perfect negative linear relationship and 1 being a perfect positive linear relationship. A score of 0 represents no linear relationship between the two features. The result of the Pearson correlation tests was stored in a large matrix and converted to the heatmap. The heatmap displays negative relationships as dark blue and positive relationships as dark red. Light blue represents minimal correlation. In addition to calculating the Pearson correlation coefficients of all features, we calculated the correlation coefficients within each of the three feature categories (anthropometric, physiological, and running gait) separately and their combinations. The features with correlations greater than |0.8| were selected and deemed highly correlated. These highly correlated features were used in the Tableau Dashboard - Interrelationship Section.



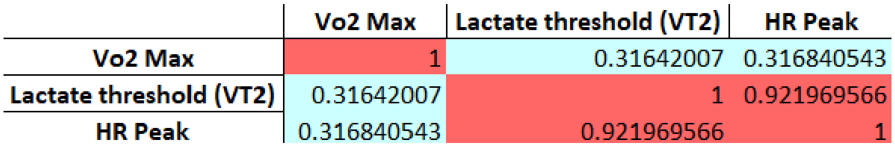
	Contact Time /s	M Sag (R Hip)	Step Separation (mm)	M Vert (L Knee)	Vertical Force /BW	M Vert (L Hip)	Economy	M Vert (R Knee)
Contact Time /s	1	-0.588687794	0.320963579	-0.620338564	-0.645087379	-0.102274881	-0.071177343	-0.582725654
M Sag (R Hip)	-0.588687794	1	-0.218665262	0.508113609	0.545679798	0.042843371	0.199987136	0.492613478
Step Separation (mm)	0.320963579	-0.218665262	1	-0.437257355	-0.440702027	-0.010246373	-0.093135179	-0.413925491
M Vert (L Knee)	-0.620338564	0.508113609	-0.437257355	1	0.956362892	0.169854781	0.073239334	0.854316572
Vertical Force /BW	-0.645087379	0.545679798	-0.440702027	0.956362892	1	0.165593998	0.118772351	0.953273935
M Vert (L Hip)	-0.102274881	0.042843371	-0.010246373	0.169854781	0.165593998	1	0.001473032	0.16600879
Economy	-0.071177343	0.199987136	-0.093135179	0.073239334	0.118772351	0.001473032	1	0.165975489
M Vert (R Knee)	-0.582725654	0.492613478	-0.413925491	0.854316572	0.953273935	0.16600879	0.165975489	1
M Sag (L Knee)	-0.527861024	0.556125587	-0.270184707	0.685662471	0.616796883	0.064876714	0.153438569	0.509612617

Figure 4. Joint Loading Feature Heatmap



	Age	Body Fat%	Height	Male	Female	BMI	Weight
Age	1	0.252014	-0.11012	-0.08814	0.088135	0.280808	0.154559
Body Fat%	0.252014	1	0.00508	-0.16134	0.161336	0.211442	0.16787
Height	-0.11012	0.00508	1	0.708637	-0.70864	0.220584	0.694136
Male	-0.08814	-0.16134	0.708637	1	-1	0.2467	0.549047
Female	0.088135	0.161336	-0.70864	-1	1	-0.2467	-0.54905
BMI	0.280808	0.211442	0.220584	0.2467	-0.2467	1	0.846812
Weight	0.154559	0.16787	0.694136	0.549047	-0.54905	0.846812	1

Figure 5. In Body Feature Heatmap



	Vo2 Max	Lactate threshold (VT2)	HR Peak
Vo2 Max	1	0.31642007	0.316840543
Lactate threshold (VT2)	0.31642007	1	0.921969566
HR Peak	0.316840543	0.921969566	1

Figure 6. VO2 Max Feature Heatmap

6.2 Feature Selection

Our feature selection process is a two-stage approach: Ensemble and Ranking. An ensemble refers to a technique where multiple methods are combined to make a prediction or decision. The idea is to leverage the strengths of individual methods to improve overall performance. Our proposed feature selection approach incorporates a combination of filter, wrapper, and embedded methods to identify and select the most relevant and non-redundant features for machine learning models. It begins with the Maximum Relevance Minimum Redundancy (MRMR) filter method, which ranks features based on their relevance to the target variable while minimizing redundancy between other features in the dataset. Recursive Feature Elimination (RFE), is applied with a Random Forest model iteratively removing less important features based on performance scores until desired number of features is reached. An embedded feature selection method using Ridge Regression fine-tunes the selected features by shrinking the feature coefficients towards zero to perform dimension reduction. In the ranking stage of our feature selection process we combine the results obtained from the ensemble of methods (filter, wrapper, and embedded methods) and provide a ranking of features based on their importance to the target variable.

Maximum Relevance And Minimum Redundancy

The Maximum Relevance and Minimum Redundancy (MRMR) criterion is designed to optimize feature selection for machine learning models. MRMR focuses on selecting features that are highly relevant to the target variable while ensuring minimal redundancy among them. In MRMR, 'Maximum Relevance' is identified by selecting features with the highest mutual information with the target variable, indicating their significant informational contribution. Simultaneously, 'Minimum Redundancy' is maintained by minimizing the correlation between features, ensuring diversity in the information they provide [12].

The script created proceeds by computing Mutual Information (MI) scores between each feature and the target variable using the mutual_info_regression() function from the scikit-learn library. These MI scores quantify the degree of association between individual features and the target variable, providing a measure of their relevance in predictive modeling. The script calculates pairwise Mutual Information among all feature pairs using the normalized_mutual_info_score() function. This step aids in evaluating the redundancy or similarity between features, crucial for ensuring the selected features offer unique predictive information. The MRMR score for each feature is computed. This score represents a balance between the feature's relevance to the target variable and its redundancy with other features. The MRMR score is calculated using a formula that considers the MI of the feature with the target variable relative to its MI with other features. Once MRMR scores are determined for all features, the script sorts the features in descending order based on their MRMR scores. It also generates the feature alongside their corresponding ranks, providing a clear overview of the importance of each feature according to the MRMR method.

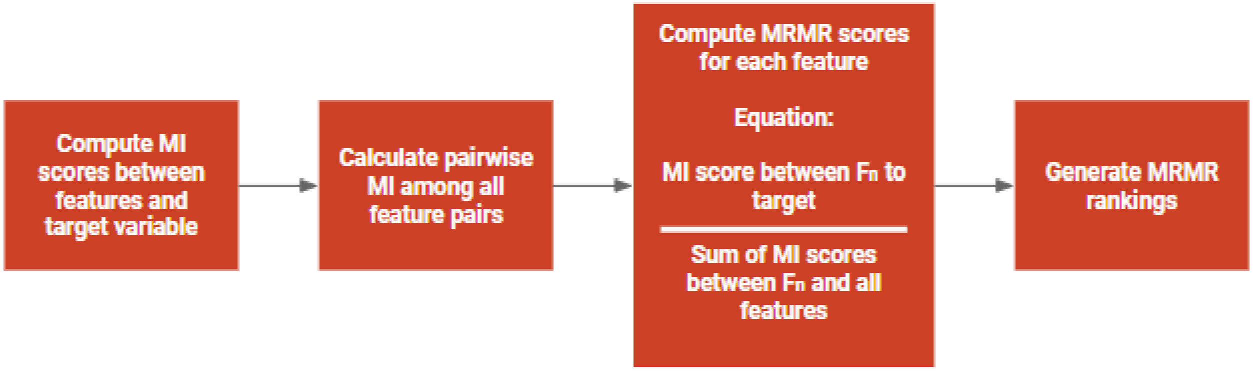

Compute MI scores between features and target variable
Calculate pairwise MI among all feature pairs
Compute MRMR scores for each feature
Equation:
MI score between Fn to target
Sum of MI scores between Fn and all features
Generate MRMR rankings

Figure 7. MRMR Calculation Process

Recursive Feature Elimination

The Recursive Feature Elimination (RFE) method with a Random Forest model is a robust technique aimed at enhancing the feature selection process. Just as Maximum Relevance and Minimum Redundancy (MRMR) seeks an optimal subset of features by maximizing relevance to the target while minimizing redundancy amongst them, RFE iteratively refines the feature set to identify the most predictive features [13].

This code we created starts by dividing the dataset into training and testing sets using the `train_test_split` function, ensuring that 80% of the data is reserved for training and 20% for testing. Following this, a Random Forest Regressor model is initialized with default settings, setting the random state for reproducibility. Sequential Feature Selector (SFS), a technique for feature selection, is then initialized. SFS aims to identify the optimal subset of features by sequentially adding or removing them based on a specified criterion, in this case, the negative mean squared error. The goal is to select the top 20 features from the dataset while performing backward elimination. After fitting SFS to the training data, the indices and names of the selected features are obtained for further analysis. The Random Forest model is trained using only these selected features. Subsequently, the test set is transformed to include only the chosen features, and predictions are made using the trained model. The code generates a DataFrame to rank the selected features based on their importance.

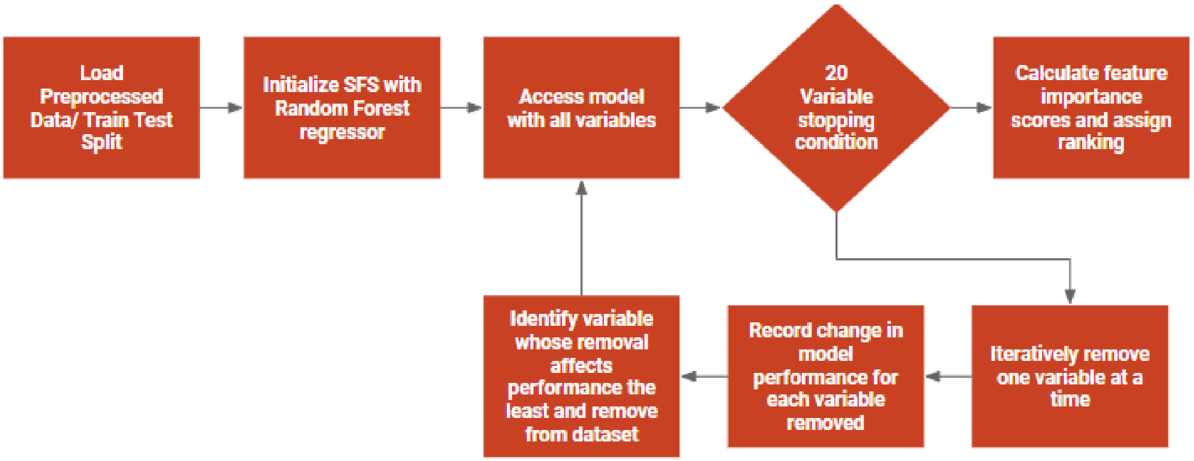

Load Preprocessed Data/ Train Test Split
Initialize SFS with Random Forest regressor
Access model with all variables
20 Variable stopping condition
Calculate feature importance scores and assign ranking
Iteratively remove one variable at a time
Record change in model performance for each variable removed
Identify variable whose removal affects performance the least and remove from dataset

Figure 8. Recursive Feature Elimination Process

Ridge Regression

Ridge regression is a linear regression technique used for dealing with highly correlated features in multiple regression data. Highly correlated features can cause problems in the model such as inflating standard errors and making it difficult to interpret the coefficients in the model. Ridge regression combats highly correlated features by adding a minimization constraint to the objective function. The ridge regression constraint tells the program to try and minimize the square of the sum of the errors of the coefficients. This constraint tends to shrink the size of the coefficients towards zero. Ridge regression is typically good at dealing with datasets where the number of features is similar to the number of predictive data points.

Our team implemented Ridge Regression using the Scikit-Learn library in python. We began by prepping the dataset, which included filtering out features that were not needed in the model. We then split the data into two groups, our target variable ‘Max Speed’ named ‘Y’ and the rest of our data named ‘X’. After, we used the train/test split function to split our data into training and validation groups. The validation set was 20% of the total dataset. To begin training the model, we initialized the Ridge Regressor and fit the model on the training data. Once the model was trained, we were able to export a csv file containing the features ranked from most to least important.

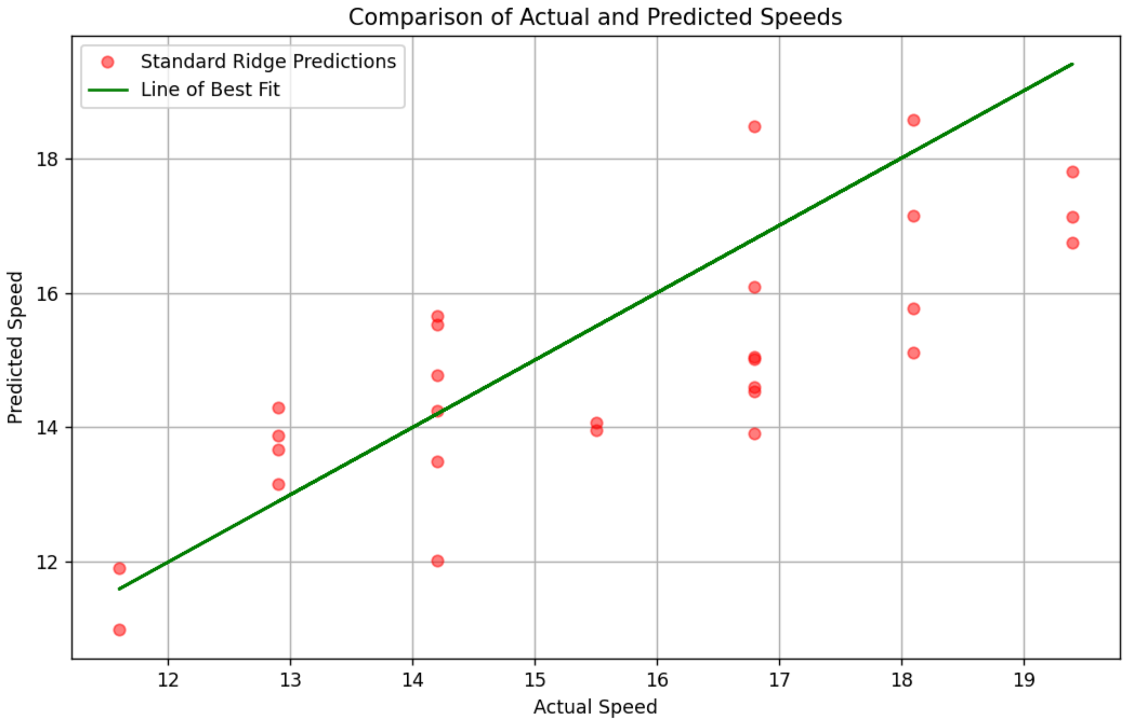
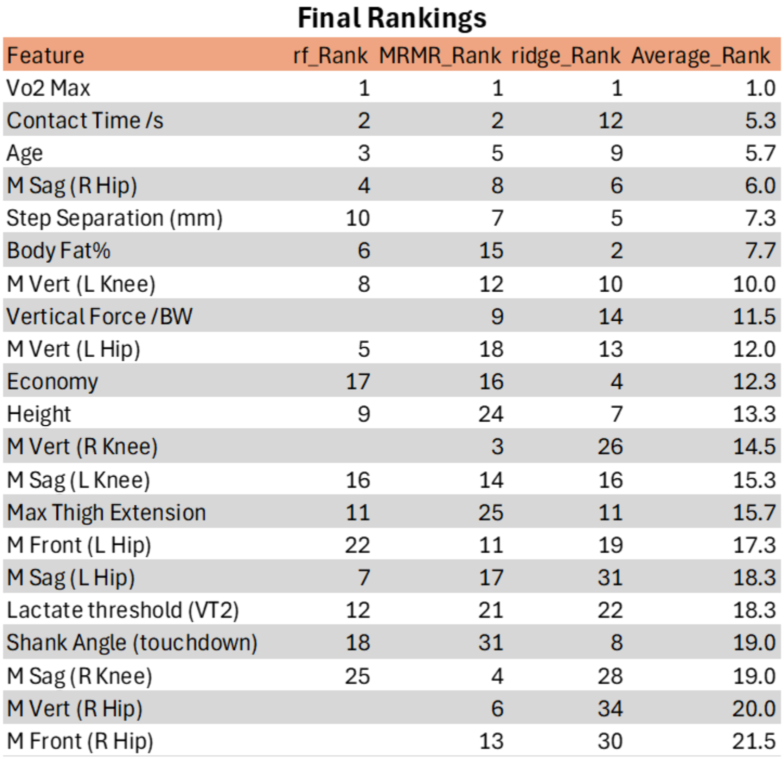


Figure 9. Ridge Regression Predicted vs. Actual Speed

Final Rankings

To determine the final feature rankings, we averaged the ranking scores obtained from the three feature selection methods: Maximum Relevance Minimum Redundancy (MRMR), Recursive Feature Elimination (RFE) with Random Forest, and Ridge Regression. This approach ensured a balanced consideration of each feature's relevance, non-redundancy, and predictive power. After averaging these scores, we selected the top 20 features as the most significant for our model, providing a comprehensive set of variables that are likely to impact and predict runners' performance effectively.



Final Rankings

Feature	rf_Rank	MRMR_Rank	ridge_Rank	Average_Rank
Vo2 Max	1	1	1	1.0
Contact Time /s	2	2	12	5.3
Age	3	5	9	5.7
M Sag (R Hip)	4	8	6	6.0
Step Separation (mm)	10	7	5	7.3
Body Fat%	6	15	2	7.7
M Vert (L Knee)	8	12	10	10.0
Vertical Force /BW		9	14	11.5
M Vert (L Hip)	5	18	13	12.0
Economy	17	16	4	12.3
Height	9	24	7	13.3
M Vert (R Knee)		3	26	14.5
M Sag (L Knee)	16	14	16	15.3
Max Thigh Extension	11	25	11	15.7
M Front (L Hip)	22	11	19	17.3
M Sag (L Hip)	7	17	31	18.3
Lactate threshold (VT2)	12	21	22	18.3
Shank Angle (touchdown)	18	31	8	19.0
M Sag (R Knee)	25	4	28	19.0
M Vert (R Hip)		6	34	20.0
M Front (R Hip)		13	30	21.5

Table 3. Final Feature Rankings

Chapter 7: Model Development

Training and Validation Sets

To prevent overfitting, we split our dataset into two groups. The first group, the training set, contained a random sample of 80% of the data. The second group, the validation set, contained the remaining 20% of the data. While training our models, only the training set was used. This allowed us to use the predictions on the validation set to test their accuracy. Additionally, we performed K-Fold cross validation on the training set. K-Fold cross validation entails splitting the training set into K groups and running the models K times, each time using a different group within the training set as a miniature validation set. This method allows for better generalization within smaller datasets.

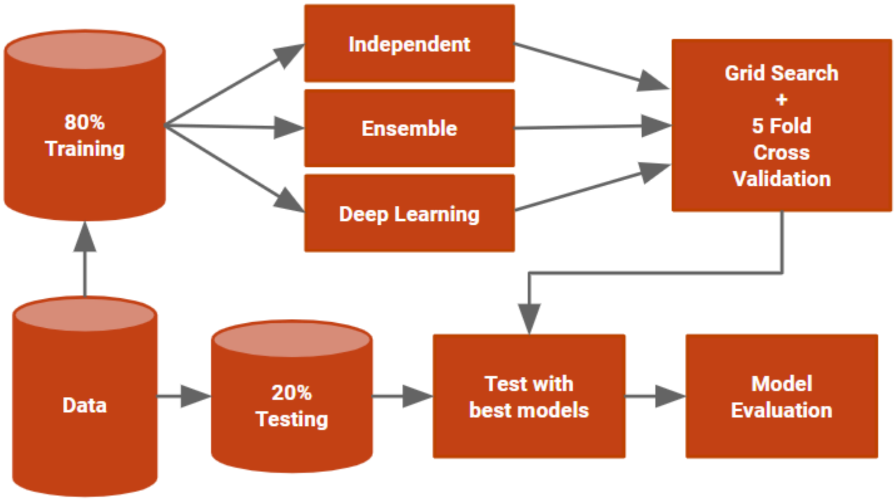


Figure 10. Model Training Flow Chart

Ensemble Models

XGBoost

XGBoost builds a model in stages, each new stage correcting errors made by the previous stage, effectively refining the accuracy of predictions. XGBoost uses decision trees as its base models and optimizes them through a technique called gradient boosting, which involves calculating errors and using them to improve the model in subsequent iterations. This approach makes XGBoost powerful for handling complex datasets and delivering precise predictions.

Our code starts by defining a dataset into features and targets, followed by a split into training and testing sets. The code uses GridSearchCV from scikit-learn, which integrates grid search with k-fold cross-validation to fine-tune the model's hyperparameters. In this process, the dataset is divided into five folds, facilitating the evaluation of each parameter combination across all folds to mitigate overfitting and ensure that the model generalizes well to unseen data. The `param_grid` outlines various values for `n_estimators`, `learning_rate`, and `max_depth`, allowing `GridSearchCV` to explore a range of parameter settings. The grid search, configured to optimize for the lowest negative mean squared error, systematically evaluates each combination of parameters. It identifies the best settings (`best_params_`), which are used to retrain the model (`best_estimator_`) on the entire training dataset. This rigorous approach ensures the selection of the most effective model configuration. The model is evaluated on a separate test set, calculating the mean squared error, mean absolute error, and root mean squared error to quantify its performance.

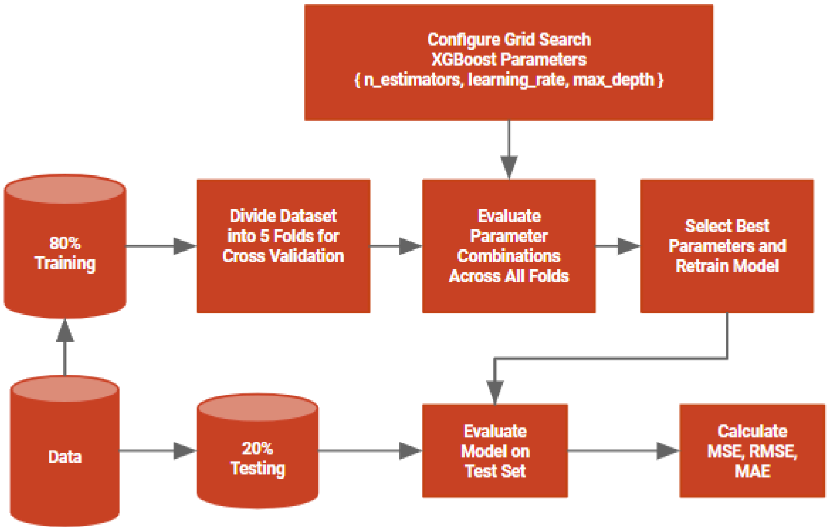

Configure Grid Search
XGBoost Parameters
{ n_estimators, learning_rate, max_depth }
80% Training
Divide Dataset into 5 Folds for Cross Validation
Evaluate Parameter Combinations Across All Folds
Select Best Parameters and Retrain Model
Data
20% Testing
Evaluate Model on Test Set
Calculate MSE, RMSE, MAE

Figure 11. XGBoost Flow Chart

Ada Boost

Adaptive boosting, otherwise known as AdaBoost, is a machine learning method used for classification and regression. It is effective in improving the accuracy of a typical decision tree. The idea of AdaBoost is to combine the predictions of a group of weak learning models to create a stronger learning model. AdaBoost trains a decision tree and computes the error. The model then assigns weights to the tree based on the error. The model reruns the tree with the new weights and recalculates the error. This process repeats for a specified number of iterations or until a predetermined accuracy score is reached. After every step, the weights of the tree are evaluated and updated. AdaBoost focuses on the features that are misclassified by the current tree, allowing these misclassifications to be corrected. To begin training the model, we initialized the AdaBoost Regressor and fit the model on the training data. Once the model was trained, we were able to export a csv file containing the features ranked from most to least important.

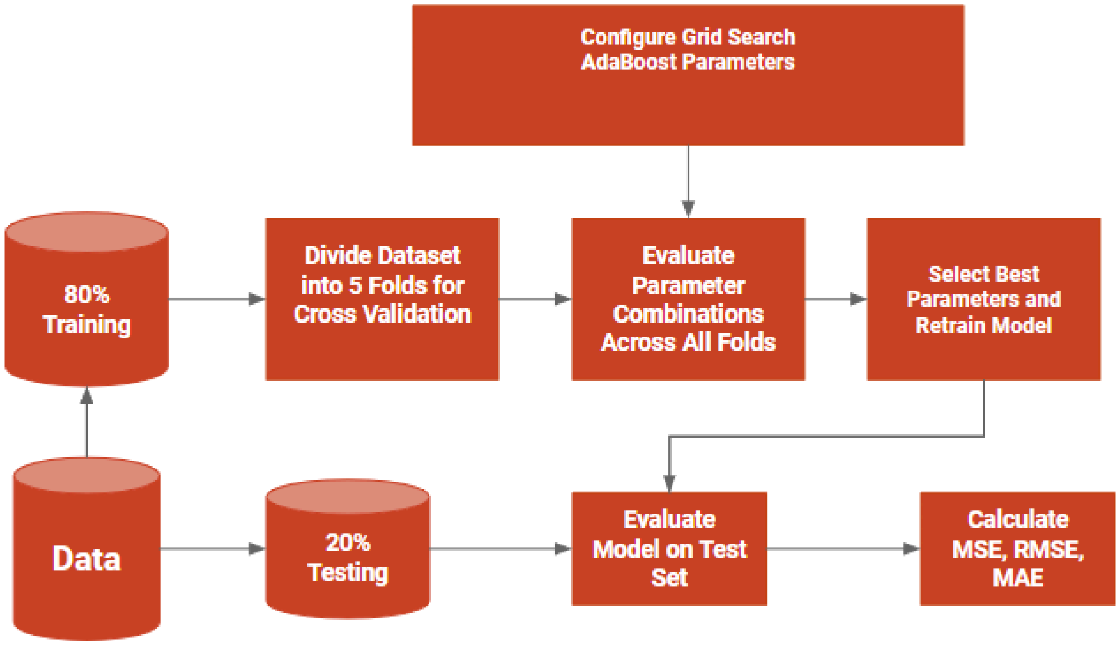

Configure Grid Search AdaBoost Parameters
80% Training
Divide Dataset into 5 Folds for Cross Validation
Evaluate Parameter Combinations Across All Folds
Select Best Parameters and Retrain Model
Data
20% Testing
Evaluate Model on Test Set
Calculate MSE, RMSE, MAE

Figure 12. AdaBoost Flow Chart

Independent Models

Support Vector Regression (SVR)

Support vector regression is a form of regression analysis that utilizes support vector machines to model and predict continuous outcomes. The objective of the SVR is to determine an optimal hyperplane that best represents the data. The model sets equidistant decision boundaries on both sides of the plane and aims to keep as much data within these boundary lines as possible. SVR tries to minimize the error margins of the points outside of the boundary by adjusting the plane and the weight of the decision boundary, while also capturing the essence of the dataset. Our SVR model performed a grid search to test all of the combinations of the model parameters, in order to find the optimal combination. The best parameters for our SVR model were a linear kernel and a C value of 30. The linear kernel sets the hyperplane to be linear. The high C value makes the fit well to the training data. We also implemented 5-fold cross validation of the data. Figure 13 below shows the line of best fit of the model. The points closer to the line were more accurate than those further away.

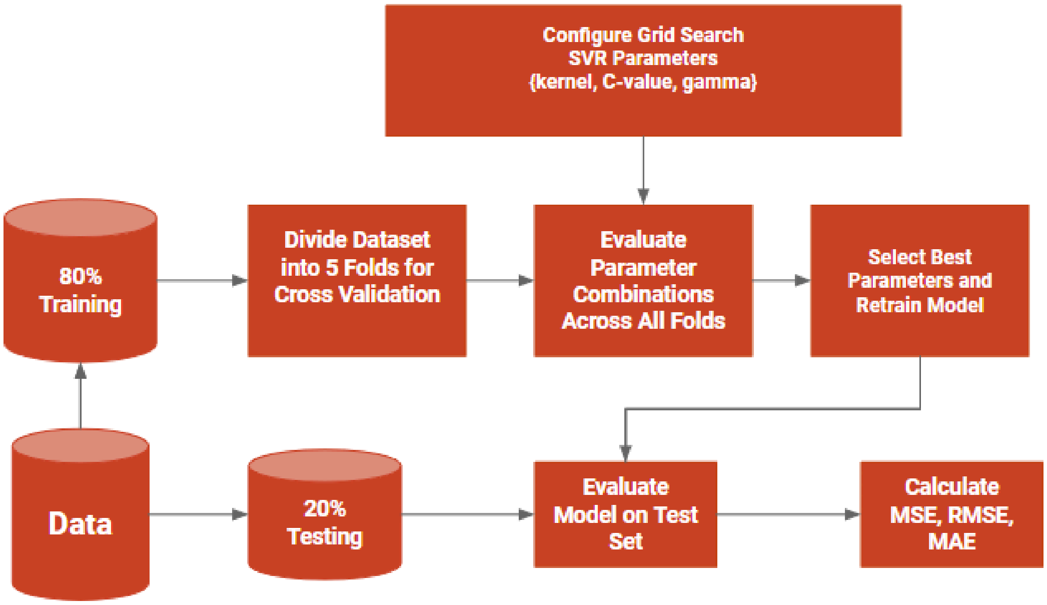

Configure Grid Search
SVR Parameters
{kernel, C-value, gamma}
80% Training
Divide Dataset into 5 Folds for Cross Validation
Evaluate Parameter Combinations Across All Folds
Select Best Parameters and Retrain Model
Data
20% Testing
Evaluate Model on Test Set
Calculate MSE, RMSE, MAE

Figure 13. SVR Flow Chart

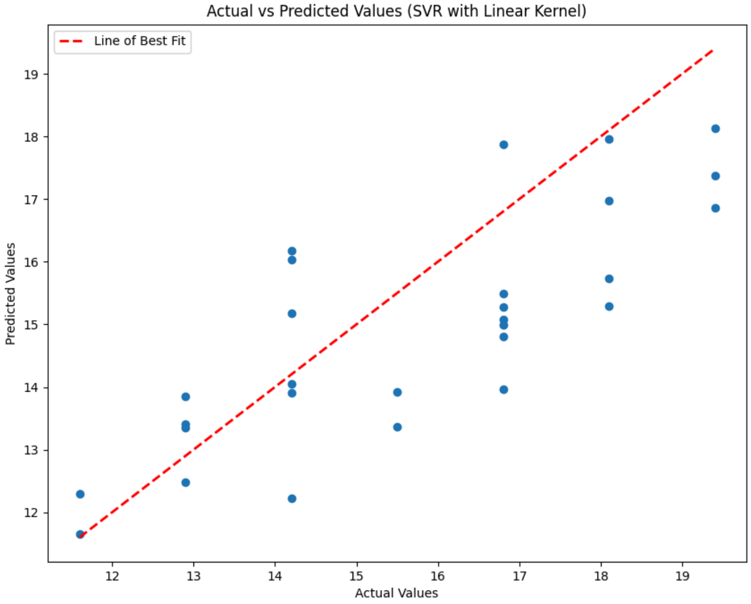


Figure 14. SVR Actual vs Predicted Values

Artificial Neural Networks (ANN)

Artificial neural networks are machine learning models designed to function like the human brain. ANNs are made up of three layers. The first layer is the input layer, it is where the data is put into the model. The last layer is the output layer, it is where the final output/decision of the model is made. The middle layer is called the hidden layer. This is where the meat of the model is. Each layer consists of nodes connected by edges that have different weights. The data is input into the input layer, then it is manipulated by the hidden layer, and output by the output layer. Once the error of the output layer is determined, the model goes back and adjusts the weight of the edges connecting the nodes. This process is known as an epoch. It is done to minimize the error of the model. This process of outputting scores and adjusting weights repeats until the model is sufficiently trained. Our ANN training model contained 1400 epochs. The model determined the optimal number of iterations was approximately 1100. Figure 16 shows a comparison between the training and testing error scores. The blue dot shows the point where the validation loss score stopped decreasing and began to increase.

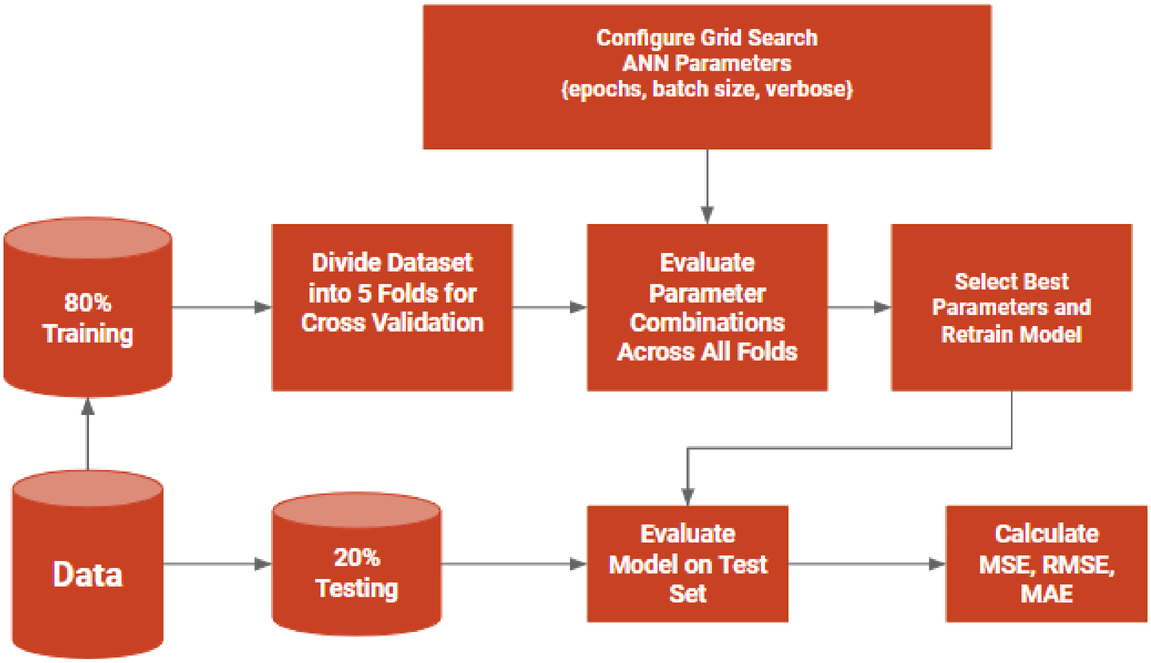

Configure Grid Search
ANN Parameters
{epochs, batch size, verbose}
80% Training
Divide Dataset into 5 Folds for Cross Validation
Evaluate Parameter Combinations Across All Folds
Select Best Parameters and Retrain Model
Data
20% Testing
Evaluate Model on Test Set
Calculate MSE, RMSE, MAE

Figure 16. ANN FlowChart

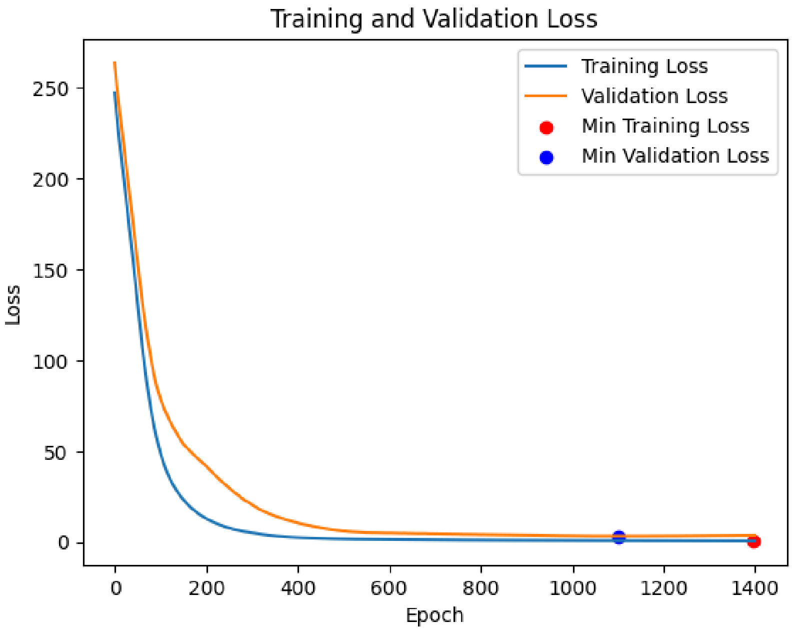


Figure 17. ANN Training and Validation Loss

Deep Learning Models

Feed Forward Neural Network (Sklearn)

The process of developing a feed forward neural network starts with splitting the data into training and testing subsets using the `train_test_split` function from `sklearn`. This function allocates 80% of the data to training and the remaining 20% to testing. Hyperparameter tuning is conducted through `GridSearchCV`, which integrates seamlessly with a pipeline comprising preprocessing (standard scaling) and a neural network (MLPRegressor). The grid search explores various configurations of neural network parameters, including different sizes of hidden layers, activation functions, and regularization strengths. By evaluating combinations across these parameters, `GridSearchCV` automates the selection of the best settings for the model. Integral to the grid search is the use of k-fold cross-validation, specified here as 5-fold cross-validation through `KFold(n_splits=5)`. This method enhances the model's validation process by ensuring that each data subset serves both as a training and a validation set across different iterations. This approach not only helps in mitigating model overfitting but also boosts the reliability of the performance metrics, providing a more accurate reflection of the model's effectiveness on unseen data. The model's performance is evaluated on the test set using critical metrics such as Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE).

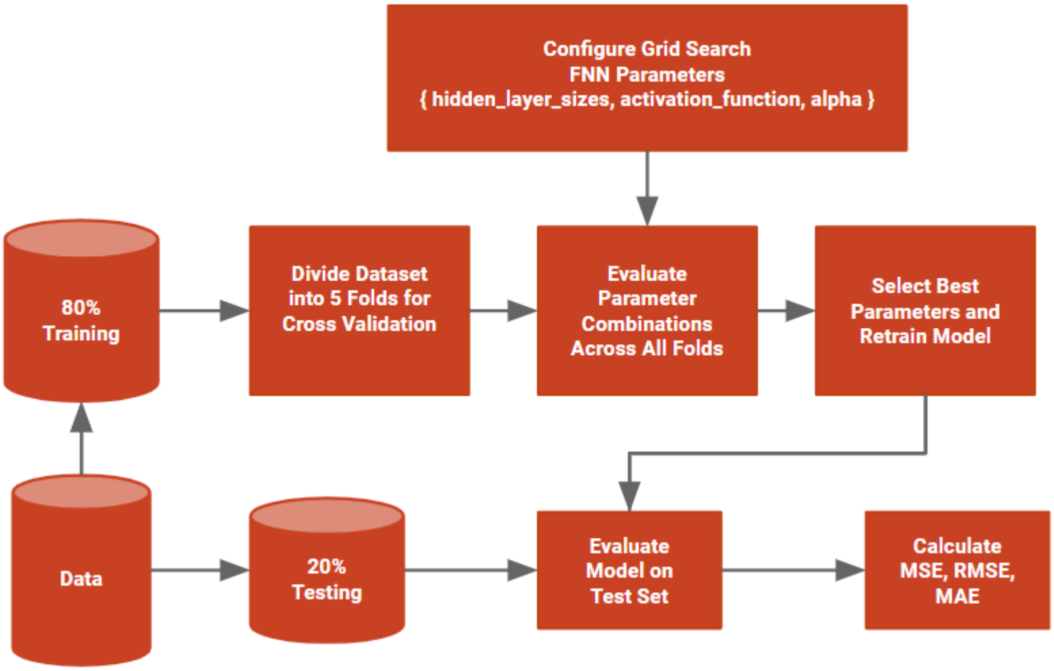


Figure 18. FNN (Sklearn) FlowChart

Feed Forward Neural Network (PyTorch)

We also utilized a deep learning architect developed with PyTorch. The model consists of an input layer, a hidden layer with a ReLU activation function, dropout regularization, and an output layer. The hidden layer for the PyTorch model contains 32 neurons, all of which employ ReLU activation. Model training was conducted over 1000 epochs, with a learning rate of 0.01, a batch size of 16, and a dropout rate of 0.8. The model parameters were decided using a grid search over a range of values for each parameter. The same 20% validation split as the TensorFlow model was used. The training was optimized using mean squared error and the Adam optimizer.

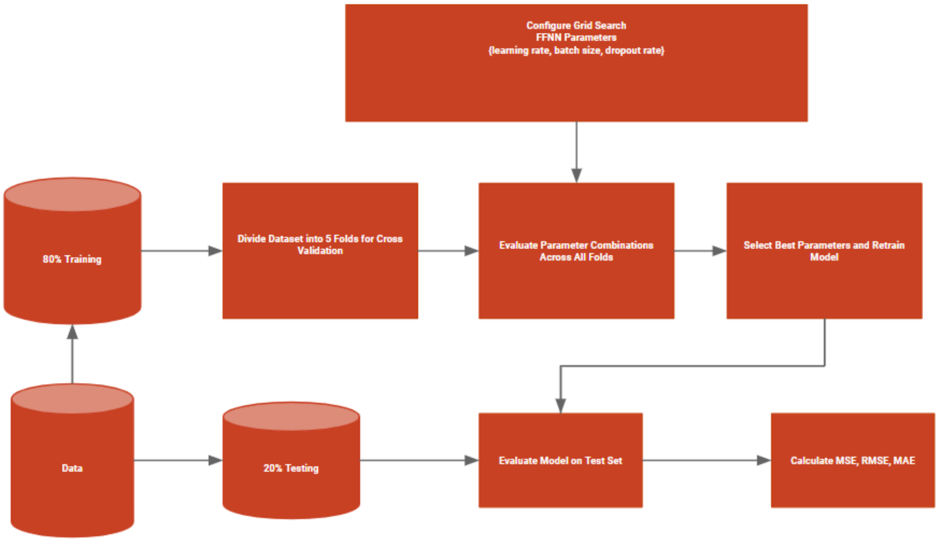

Configure Grid Search
FFNN Parameters
{learning rate, batch size, dropout rate}
80% Training
Divide Dataset into 5 Folds for Cross Validation
Evaluate Parameter Combinations Across All Folds
Select Best Parameters and Retrain Model
Data
20% Testing
Evaluate Model on Test Set
Calculate MSE, RMSE, MAE

Figure 19. FFNN (PyTorch) Flow Chart

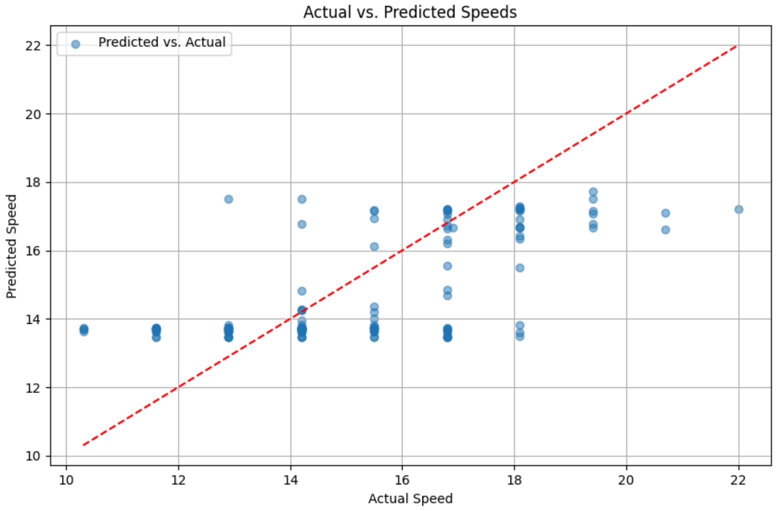


Figure 20. PyTorch FFNN Actual vs. Predicted Speeds

Chapter 8: Model Evaluation

In predictive modeling, we evaluate the performance of models by using error metrics such as Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE). Lower values in these metrics typically indicate better predictive accuracy. The models compared in this evaluation include Artificial Neural Networks (ANN), Feedforward Neural Networks implemented with scikit-learn (FNN (sklearn)) and PyTorch (FNN (PyTorch)), Support Vector Regression (SVR), AdaBoost, and XGBoost.

MSE measures the average of the squares of the errors—that is, the average squared difference between the estimated values and the actual value. In this metric, ANN reported the highest error at 22.15, indicating that on average, the square of its prediction error is greater compared to the other models. XGBoost performed the best with the lowest MSE value of 2.07, suggesting that it has the most precise predictions among the evaluated models.

RMSE is the square root of the MSE and provides a measure of the average magnitude of the error. It is more sensitive to outliers than MSE. The results are consistent with MSE, with ANN having the highest RMSE of 4.420 and XGBoost the lowest at 1.440. The RMSE values for all models suggest that XGBoost tends to have the lowest variance in its predictions.

MAE measures the average magnitude of the errors in a set of predictions, without considering their direction. It's a linear score which means all individual differences are weighted equally. Again, XGBoost demonstrates the lowest error with a MAE of 1.120, indicating it has a better average absolute deviation from the true values. In contrast, ANN has the highest MAE at 2.800, which suggests its predictions are less accurate on average.

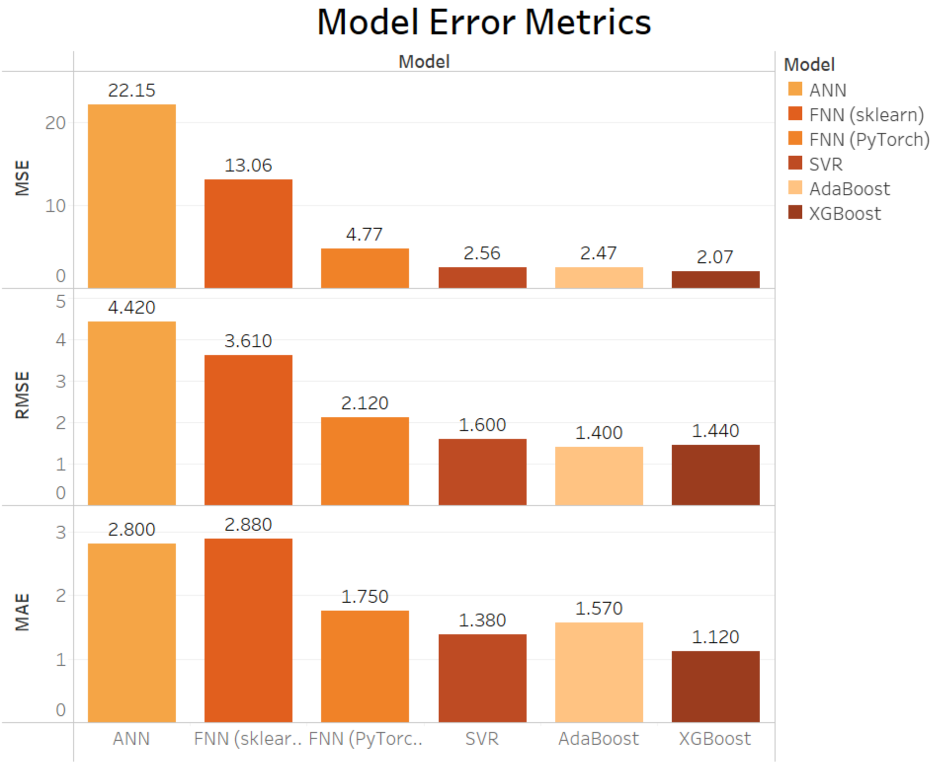


Figure 21. Final Error Metrics

Chapter 9: Tableau Dashboard

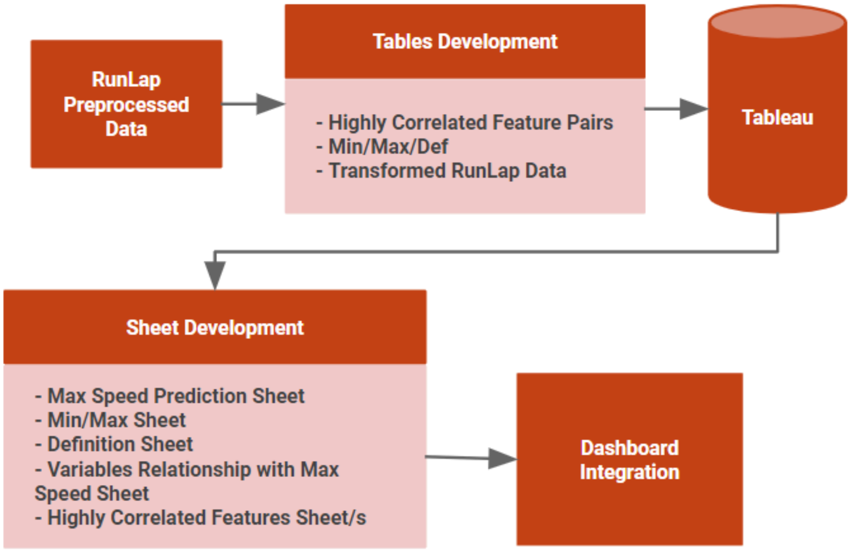

RunLap Preprocessed Data
Tables Development
- Highly Correlated Feature Pairs
- Min/Max/Def
- Transformed RunLap Data
Tableau
Sheet Development
- Max Speed Prediction Sheet
- Min/Max Sheet
- Definition Sheet
- Variables Relationship with Max Speed Sheet
- Highly Correlated Features Sheet/s
Dashboard Integration

Figure 22. Tableau Dashboard Flow Chart

Table Development

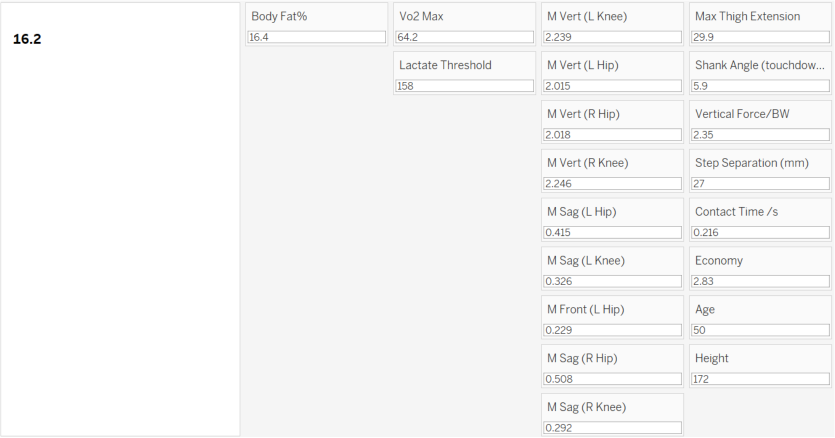
In order to construct the predictive dashboard in Tableau we first had to transform our runners dataset. The process involved creating lists for each selected feature in the top 20, including 'Max Speed', which was appended to the list of variables. This step involved iterating through the dataset to collect values for each feature across the filtered records, effectively capturing the essence of each variable in a separate list. These lists were then meticulously organized into a dictionary, where each key represented a feature name, and its corresponding value was the list containing the data for that feature.

To enhance the comprehensiveness and utility of the predictive dashboard being developed for Tableau we began preparing and analyzing the top 20 features. The process began with reading data from three CSV files: feature rankings, runner data, and a list of data fields. An empty dataframe was created to store minimum and maximum values for each of the selected features. By iterating through each feature, minimum and maximum values were calculated and stored. This step was crucial for understanding the range of data for each feature, facilitating better data visualization and analysis in Tableau. The min-max dataframe was merged with the dataframe containing descriptions of each feature. This merging process enriched the feature data with descriptive information, making the dataset not only more informative but also more accessible for users of the predictive dashboard. The dataset was then exported to an Excel file, laying the groundwork for its integration into Tableau.

Sheet Development

Prediction Sheet

To create the sheet that predicts the value for integration into the dashboard, a calculated field was established in Tableau that leverages Python scripting through the TabPy server. The script begins by defining a DataFrame from the input arguments which are passed from Tableau. These arguments represent the input features used in our model. The script then proceeds to separate the features into a matrix X and a target vector y, with 'Max Speed' being the target. Then implemented the same XGboost model that was developed previously that had the highest accuracy out of all the models we developed. For the prediction phase, the script prepares the input by reshaping the array of parameters received from Tableau and then uses the trained model to make a prediction of 'Max Speed'. This predicted value is then rounded to one decimal place and returned as a string to Tableau, where it is displayed as a floating-point number in the calculated field.



16.2

Body Fat%
16.4

Vo2 Max
64.2

Lactate Threshold
158

M Vert (L Knee)
2.239

M Vert (L Hip)
2.015

M Vert (R Hip)
2.018

M Vert (R Knee)
2.246

M Sag (L Hip)
0.415

M Sag (L Knee)
0.326

M Front (L Hip)
0.229

M Sag (R Hip)
0.508

M Sag (R Knee)
0.292

Max Thigh Extension
29.9

Shank Angle (touchdow...
5.9

Vertical Force/BW
2.35

Step Separation (mm)
27

Contact Time /s
0.216

Economy
2.83

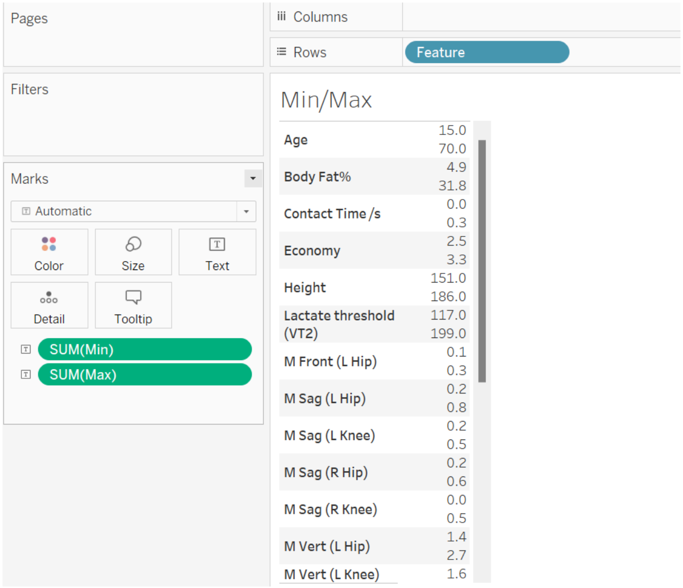
Age
50

Height
172

Figure 23. Tableau Prediction Sheet

Min/Max Overview Sheet

The Min/Max Overview Sheet was constructed from a dataset, specifically focusing on selected features to highlight their minimum, maximum, and descriptive values. Each feature was arranged in rows, allowing for an organized display. Adjacent to each feature, minimum and maximum values were prominently presented as text labels, ensuring a clear and concise representation of the data range for each feature.



Rows: Feature

SUM(Min)
SUM(Max)

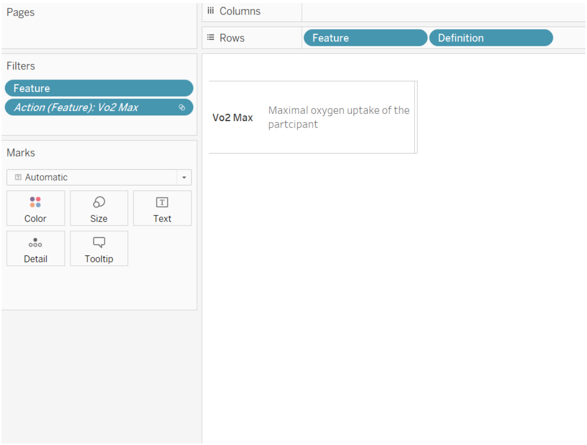
Min/Max

Age	15.0 70.0
Body Fat%	4.9 31.8
Contact Time /s	0.0 0.3
Economy	2.5 3.3
Height	151.0 186.0
Lactate threshold (VT2)	117.0 199.0
M Front (L Hip)	0.1 0.3
M Sag (L Hip)	0.2 0.8
M Sag (L Knee)	0.2 0.5
M Sag (R Hip)	0.2 0.6
M Sag (R Knee)	0.0 0.5
M Vert (L Hip)	1.4 2.7
M Vert (L Knee)	1.6

Figure 24. Tableau Min/Max Overview Sheet

Definition Reference Sheet

The Definition Reference Sheet followed a similar creation process, emphasizing the provision of detailed explanations for each feature. This approach facilitated a deeper understanding of the dataset's components, making the information accessible and informative for users.



Rows: Feature Definition

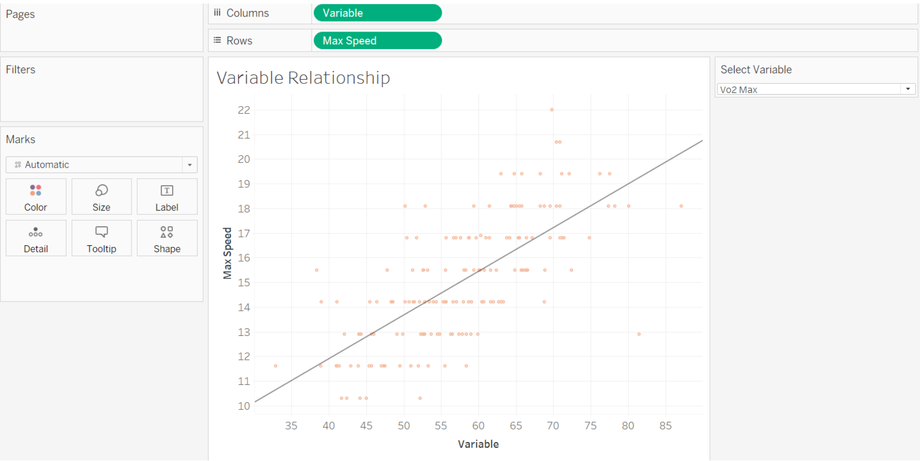
Filters:
Feature
Action (Feature): Vo2 Max

Vo2 Max	Maximal oxygen uptake of the partcipant

Figure 25. Definition Reference Sheet

Variable Relationship to Max Speed Analysis

To explore the dynamic relationship between variables and max speed, a specialized sheet was developed. Initially, an empty variable was positioned in the rows section to serve as a placeholder. A user-selectable parameter, named "Selected Variable," was introduced, granting users the flexibility to choose the variable of interest. The inclusion of a trend line illustrates the correlation whether positive or negative between the selected variable and max speed, offering valuable insights into how different factors influence speed.



Columns: Variable

Rows: Max Speed

Select Variable: Vo2 Max

Figure 26. Variable Relationship to Max Speed Sheet

Interrelationship Sheets

The dynamic dashboard we created also contains plots of the highly interrelated features. Within each of the three feature categories (anthropometric, physiological, and running gait) there is a menu that allows the user to select from a list of features. When a feature is selected, the dashboard displays a scatterplot of the feature compared with the other features it is highly correlated to. We defined a high correlation as having a Pearson correlation value greater than 0.8. For example, if ‘Vertical Force /BW’ is selected, the dashboard below in Figure 27 is shown.

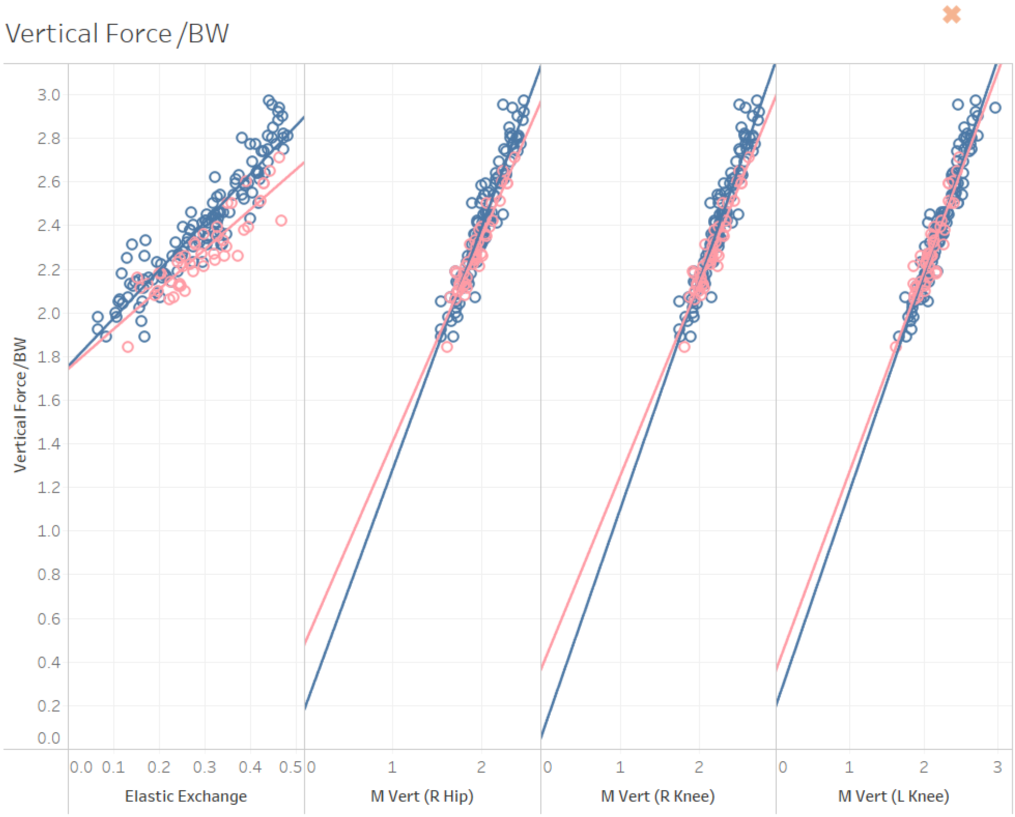


Figure 27. Sample of the interrelationship dashboard, Vertical Force /BW

The dashboard displays the relationship between 'Elastic Exchange', 'M Vert (R Hip)', 'M Vert (R Knee)', 'M Vert (L Knee)' and the selected feature 'Vertical Force /BW'. The dashboard is split between blue and pink points, representing male and female runners respectfully. Additionally, there are two lines of best fit for each plot, one for each gender represented. The dashboard also allows users to filter by age and gender, which allows a user to get a better understanding of how they compare to similar data points.

Users also have the ability to mouse over the plots to view more information. When mousing over data points, the exact values for each feature on the dashboard is displayed. When mousing over the regression lines, the r-squared value, p-value, and formula are displayed.

Max Speed Boxplot Comparison

To explore the max speed prediction, there is a menu that links to the dashboard shown below in Figure 28. The dashboard allows the user to filter by age and gender, and compare their own max speed to the box plots displayed. Users can mouse over the box plot to display the values of the upper whisker, upper hinge, median, lower hinge, and lower whisker. Users can also mouse over the data points to display the max speed of those runners.

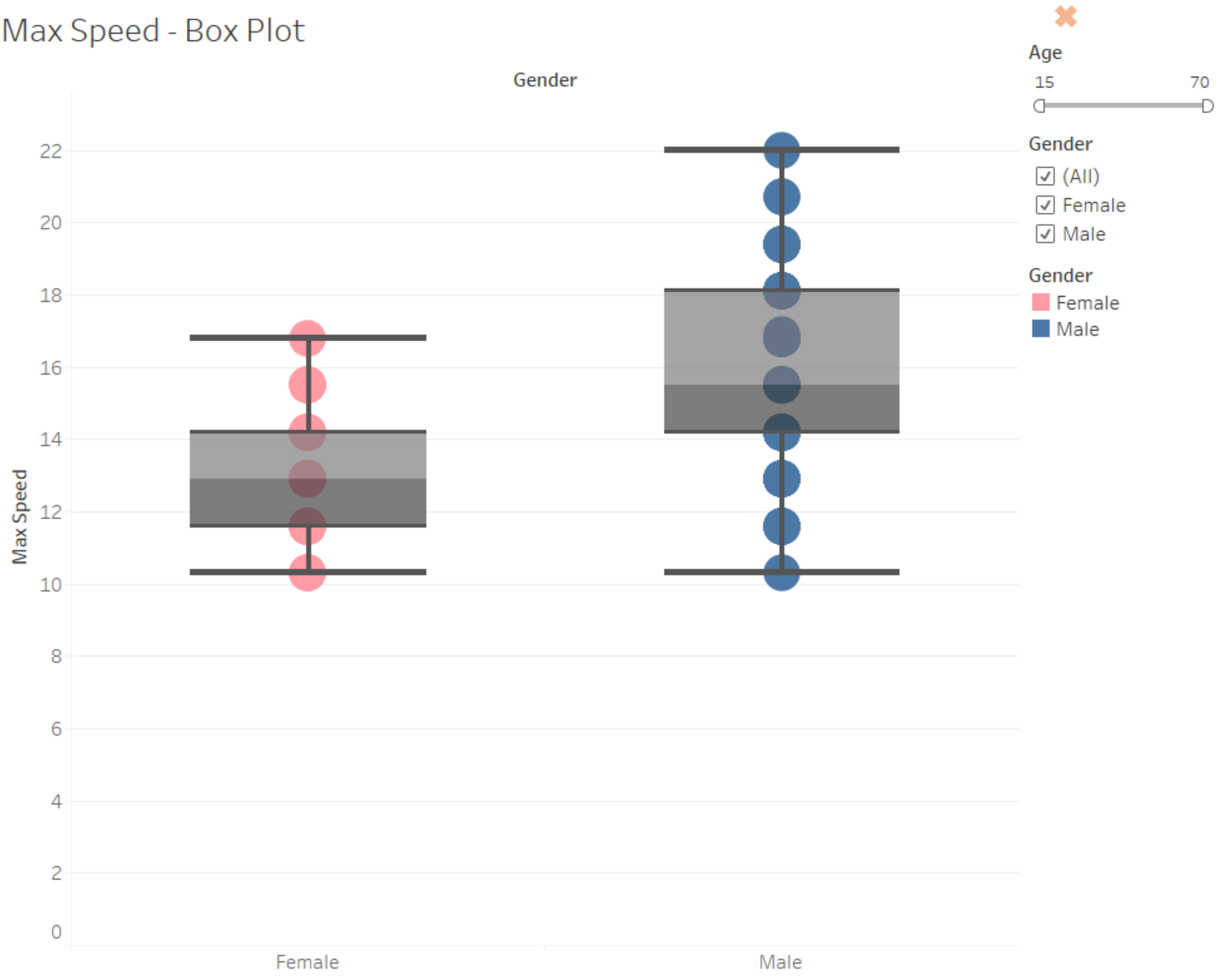
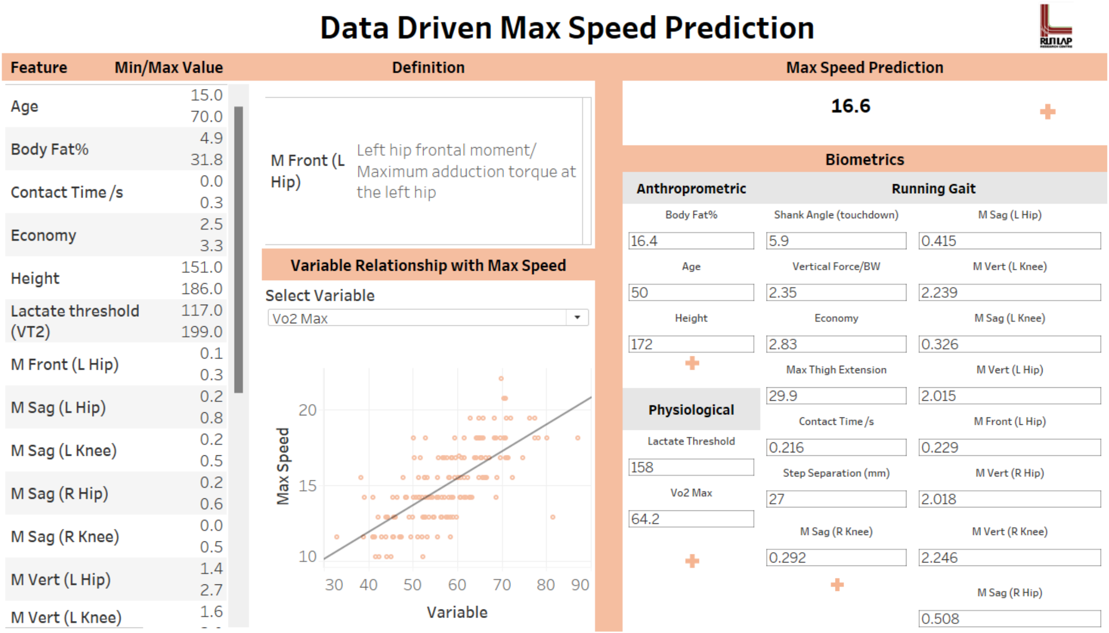


Figure 28. Max Speed Boxplot

Dashboard Integration

The culmination of this process was the assembly of the final dashboard, integrating all individual sheets into a unified display. An interactive feature was incorporated, allowing user interactions within the Min/Max Overview Sheet to dynamically update the Definition Reference Sheet with information about the selected variable. The Variable Relationship to Max Speed Analysis section provided users with the capability to examine the impact of various factors on max speed through their selections. Positioned prominently on the dashboard, the prediction sheet displayed the calculated max speed based on user-manipulated input features, offering a predictive insight into performance outcomes. This cohesive dashboard delivers a user-centric analytical tool, enabling interactive exploration, comprehensive understanding, and predictive modeling of key performance indicators within the dataset.



Data Driven Max Speed Prediction

Feature	Min/Max Value
Age	15.0 70.0
Body Fat%	4.9 31.8
Contact Time /s	0.0 0.3
Economy	2.5 3.3
Height	151.0 186.0
Lactate threshold (VT2)	117.0 199.0
M Front (L Hip)	0.1 0.3
M Sag (L Hip)	0.2 0.8
M Sag (L Knee)	0.2 0.5
M Sag (R Hip)	0.2 0.6
M Sag (R Knee)	0.0 0.5
M Vert (L Hip)	1.4 2.7
M Vert (L Knee)	1.6

Definition

M Front (L Hip) — Left hip frontal moment/ Maximum adduction torque at the left hip

Variable Relationship with Max Speed

Select Variable: Vo2 Max

Max Speed Prediction

16.6

Biometrics

Anthropometric

Body Fat%: 16.4
Age: 50
Height: 172

Physiological

Lactate Threshold: 158
Vo2 Max: 64.2

Running Gait

Shank Angle (touchdown): 5.9
Vertical Force/BW: 2.35
Economy: 2.83
Max Thigh Extension: 29.9
Contact Time /s: 0.216
Step Separation (mm): 27
M Sag (R Knee): 0.292

M Sag (L Hip): 0.415
M Vert (L Knee): 2.239
M Sag (L Knee): 0.326
M Vert (L Hip): 2.015
M Front (L Hip): 0.229
M Vert (R Hip): 2.018
M Vert (R Knee): 2.246
M Sag (R Hip): 0.508

Figure 29. Final Dashboard

Chapter 10: Conclusion

This research paper has delved deeply into the multifaceted aspects of enhancing athletic performance through advanced data analytics, providing a comprehensive examination of several predictive models aimed at forecasting endurance metrics. The integration of machine learning techniques such as Support Vector Machines (SVM), Artificial Neural Networks (ANN), AdaBoost, and XGBoost has been pivotal in understanding and predicting the performance outcomes of endurance athletes. Notably, the XGBoost model emerged as the most effective, offering the highest accuracy and the least prediction errors among the models tested.

This study not only validates the robustness of using sophisticated analytical methods in sports science but also sets the stage for future explorations into this exciting field. It opens up possibilities for integrating more complex data sets, including physiological, psychological, and environmental factors, to create a holistic model of athlete performance. Additionally, further research could explore the impact of real-time data analytics and its application in dynamic training environments.

Moreover, the implications of this work extend beyond the realm of sports science. The methodologies and insights gained here could be applied to other fields where performance optimization is crucial, such as rehabilitation medicine and human resource management. By continuing to refine these models and expand their applicability, future research can provide more tailored and effective solutions to a wide range of performance-related challenges.

In conclusion, this paper marks a significant contribution to the field of sports analytics, highlighting the potential of machine learning to transform traditional training methods and enhance athletic performance through data-driven insights. As this field evolves, it will undoubtedly continue to provide valuable tools and methodologies for coaches, athletes, and researchers aiming to push the boundaries of what is possible in sports and beyond.

References

[1] Venturini, Elio, and Francesco Giallauria. “Factors Influencing Running Performance during a Marathon: Breaking the 2-H Barrier.” Frontiers in cardiovascular medicine, March 2, 2022. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8924290/.

[2] Lorenz, Daniel, and Scot Morrison. “Current Concepts in Periodization of Strength and Conditioning for the Sports Physical Therapist.” International journal of sports physical therapy, November 2015. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4637911/.

[3] Rüst, Christoph Alexander, Beat Knechtle, Patrizia Knechtle, Ursula Barandun, Romuald Lepers, and Thomas Rosemann. “Predictor Variables for a Half Marathon Race Time in Recreational Male Runners.” Open Access Journal of Sports Medicine 2 (2011): 113–19. doi:10.2147/OAJSM.S23027.

[4] Bundle, Matthew W. “High-Speed Running Performance: A New Approach to Assessment and Prediction | Journal of Applied Physiology.” Journal of Applied Physiology, November 1, 2003. https://journals.physiology.org/doi/full/10.1152/japplphysiol.00921.2002.

[5] Gómez-Molina, Josué, Ana Ogueta-Alday, Jesus Camara, Christoper Stickley, José A Rodríguez-Marroyo, and Juan García-López. “Predictive Variables of Half-Marathon Performance for Male Runners.” Journal of sports science & medicine, June 1, 2017. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5465980/.

[6] Bundle, Matthew W. “High-Speed Running Performance: A New Approach to Assessment and Prediction | Journal of Applied Physiology.” Journal of Applied Physiology, November 1, 2003. https://journals.physiology.org/doi/full/10.1152/japplphysiol.00921.2002.

[7] Coquart, Jeremy B. “Prediction of Performance in a 100-Km Run from a Simple Equation.” PLOS ONE, 2023. https://journals.plos.org/plosone/article?id=10.1371%2Fjournal.pone.0279662.

[8] Etxegarai, Urtats, Eva Portillo, and Jon Irazusta. “Estimation of Lactate Threshold with Machine Learning Techniques in Recreational Runners.” Shibboleth authentication request, 2018. https://www-sciencedirect-com.ezpv7-web-p-u01.wpi.edu/science/article/pii/S1568494617306993?via%3Dihub.

[9] Wiecha, Szczepan. “Modeling Physiological Predictors of Running Velocity for Endurance Athletes.” Shibboleth authentication request, 2022. https://www-scopus-com.ezpv7-web-p-u01.wpi.edu/record/display.uri?eid=2-s2.0-85142333952&origin=resultslist&sort=plf-f&src=s&sid=2767e8c184f70060359d1a9908e155b6&sot=b&sdt=b&s=TITLE-ABS-KEY%28%22running%2Bperformance%22%2B%22machine%2Blearning%22%29&sl=55&sessionSearchId=2767e8c184f70060359d1a9908e155b6.

[10] Liu, Qi, W.R. Johnson, Shiwei Mo, and Vincent C.K. Cheung. “Classification of Runners’ Performance Levels with Concurrent Prediction of Biomechanical Parameters Using Data from Inertial Measurement Units.” Journal of Biomechanics, October 8, 2020. https://www.sciencedirect.com/science/article/abs/pii/S0021929020304966.

[11] Hasegawa, Hiroshi, Takeshi Yamauchi, and William J Kram=emer. “Foot Strike Patterns of Runners at the 15-Km Point during... : The Journal of Strength & Conditioning Research.” LWW, 2007. https://journals.lww.com/nsca-jscr/abstract/2007/08000/FOOT_STRIKE_PATTERNS_OF_RUNNERS_AT_THE_15_KM_POINT.40.aspx.

[12] Zhao, Z., Anand, R., & Wang, M. (2019, August 15). Maximum relevance and minimum redundancy feature selection methods for a marketing machine learning platform. arXiv.org. https://arxiv.org/abs/1908.05376

[13] Jeon, H., & Oh, S. (2020, May 4). *Hybrid-recursive feature elimination for efficient feature selection*. MDPI. https://www.mdpi.com/2076-3417/10/9/3211

[14] Li, Guohui, and Peide Niu. "An Enhanced Extreme Learning Machine Based on Ridge Regression for Regression." Neural Computing and Applications 22, no. 3 (2013): 803-810. https://doi.org/10.1007/s00521-011-0771-7.