Sports Analytics for statistical analysis and predictions in Formula 1

Myrotheou Sofia

SID: 3308230014

SCHOOL OF SCIENCE & TECHNOLOGY

A thesis submitted for the degree of

Master of Science (MSc) in Data Science

17/01/2024

THESSALONIKI – GREECE

Sports Analytics for statistical analysis and predictions in Formula 1

Myrotheou Sofia

SID: 3308230014

Supervisor:	Prof. Christos Tjortjis
Supervising Committee Members:	Dr. Paraskevas Koukaras
	Dr. Christos Berberidis

SCHOOL OF SCIENCE & TECHNOLOGY

A thesis submitted for the degree of

Master of Science (MSc) in Data Science

17/01/2024

THESSALONIKI – GREECE

Abstract

Formula 1 has developed over the years into a data-intensive sport with hundreds of sensors collecting data and capturing metrics for every car and driver during each race. Nowadays, teams are leveraging advanced analytics to make informed strategic decisions and to gain a competitive edge. Furthermore, predictive analytics has emerged as a valuable tool, benefiting teams, athletes, fans, and investors alike. Despite this, the integration of environmental factors, such as weather conditions, into predictive models remains underexplored.

This dissertation investigated various predictive modelling techniques, focusing on traditional statistical methods and advanced data mining methods. The models were evaluated using historical and weather data from season 2014 to season 2023, the hybrid era of Formula 1. The results show that ensembles methods, particularly Gradient Boosting consistently outperformed others in predictive accuracy, reaching an R^2 score higher than 0.98 when incorporating weather data.

Additionally, this study offers actionable insights into the features that affect Formula 1 performance predictions, by identifying and quantifying their effect. The research highlights the critical influence of features like qualifying position, grid position, and points, offering actionable insights. Moreover, key findings reveal that integrating weather data significantly enhances model accuracy, with environmental factors such as temperature, humidity, and rainfall playing a crucial role in performance.

This study aims to help the stakeholders who seek a deeper understanding about the predictive process and the relative merits of different modelling techniques in the context of Formula 1. By bridging the gap between traditional statistical methods and modern machine learning approaches, this study contributes to the growing field of predictive analytics in motorsport, paving the way for more precise and adaptive strategies, interesting for teams, investors, and analysts.

Myrotheou Sofia

17/01/2024

Contents

TABLE OF FIGURES

TABLE OF TABLES

1 Introduction

Formula 1 (F1), often considered the pinnacle of motorsport, is a global phenomenon renowned for its high-speed competition, use of state-of-the-art technology, and elite drivers. Formula 1 has changed significantly. Since its humble beginnings in 1950 with the Formula One World Championship, F1 has evolved significantly. Everyone is aware of the progress. The evolution came not only in terms of technical innovation and safety but also as being a global sporting brand and business. Formula 1 is a phenomenon known nowadays to all. The sport's dynamic nature and complex data at hand present a unique blend of predictability and unpredictability.

In recent years, especially the last decade, the intersection of data science and motorsports has accelerated. Many data scientists are attracted to the field of Formula 1, since it stands out as a sport that collects a wealth of real-time data. The availability of increased volumes of data, additionally with historical metadata, has spurred major advancements with regards to sports analysis, enabling teams to predict race outcomes, track positions, and optimize strategies (Heilmeier, Graf, et al., 2020). Nevertheless, the area remains still underexplored.

Beyond the team themselves, accurate predictions significantly impact a wide variety of stakeholders, including investors, team managers, and owners (Heilmeier, Graf, et al., 2020). In addition to that, the sports betting industry with accurate predictive models can hold substantial financial value, with the beneficiaries to be bookmarked, bettors, fans, or commentators (R. P. Bunker & Thabtah, 2019).

Predicting Formula 1 race results accurately is still a difficult task, even with the wealth of data available today. This issue of accurate predictions comes from the convergence of multiple and complex variables, such as car design, driver skill, and external factors like weather conditions. These variables additionally depict usually non-linear relationships of the data.

Despite the implementation of traditional statistical models for sports outcome predictions, their accuracy frequently suffers due to their inability to accurately represent the complexities of the data. Emerging techniques, such as artificial neural networks (ANNs), demonstrate promise because they can process large datasets and adapt to complex patterns. However, their potential in Formula 1 remains underexplored, as the whole area does (Van Kesteren & Bergkamp, 2023). The effect of the weather data is still to be fully explored as the whole area.

Among the findings of the research is the power of ensemble models. Among the four models used, the two ensemble models were the most accurate ones. Additionally, the new finding this research offered is that the effect of the weather is not as great as it thought. Even though the addition of weather data improves the accuracy of all four models, and along with it the generalization since feature selection was applied, as a measurable effect it is not as important as the driver's and the team's performance. Its overall effect comes secondary when compared to driver and team performance.

1.1 Problem Statement

The complex dynamics and non-linear interactions of Formula One races are frequently not well captured by traditional statistical models used to forecast race results. Advances in machine learning, particularly with the neural networks, offer new opportunities for improved accuracy. In addition to ANNs the ensemble models gain ground substantially. Nonetheless, application on predictive modelling in motorsports and especially F1 remains underexplored. An issue noticed is the limited number of studies addressing the integration of weather data and measuring their impact on predictions.

The goal of this research is to methodically review existing predictive models in Formula 1. Next, it aims to develop a strong predictive algorithm for forecasting the drivers positions in each race. Traditional statistical models are useful, yet they frequently lack precision in capturing the complex, nonlinear dynamics of motorsports and, in the extent of the F1 races. Since the importance of all factors is stated, this study aims to determine the most significant factors influencing race outcomes by measuring the influence of weather on the accuracy of predictions made by various models.

Moreover, this study, by leveraging historical data (2014–2023) and environmental variables, such as temperature, humidity, and rainfall, aims to identify key predictors of race outcomes and evaluate the performance of various predictive models. The findings aim to provide useful information for teams, investors, and other stakeholders while contributing to the broader field of predictive analytics in motorsport.

1.2 Research Questions

First and foremost, the question of this research to focus on Formula 1 stems from the personal liking of the author. More specifically, the following Research Questions guide this study:

1. Review existing predictive modelling approaches in sports analytics in general and specifically in motorsports. This question was raised in order to identify the models used in the area, their strengths and weaknesses and find the most appropriate ones to use in this research

2. Analyse the significance of various features in influencing prediction accuracy and the extent to which they affect the results. This next question was raised because the most important question when it comes to predicting the outcome of any sport race or match is the factors that influenced the outcome. Discovering the features that contribute to the outcome is crucial to understanding the sport in general. Moreover, there is the forever question in Formula 1 of how much the weather influences the outcomes, a question trying to be measured in this research.

3. Develop and test models that can reliably predict the finishing positions in Formula 1 races. This question is the last step in this research. The conclusion of the analysis is to model a predictor able to rank accurately the final positions in a race outcome.

1.3 Scope

This research uses a dataset spanning 2014–2023. For the seasons 2018-2023, there are additional weather data. The scope of this research revolves around investigating the efficacy of various machine learning models. These models are trained to accurately predict race outcomes.

Moreover, by incorporating weather data (2018-2023), the study searched for the role of weather in improving the model's performance. Another part of the research concerning the weather is the extent to which these variables affect the predictions and the extent to which weather plays a role in the final ranking of the drivers.

Lastly, it is reported that the findings provide useful insights for F1 teams and data scientists. Moreover, contributing to the broader field of predictive analytics in sport affects other stakeholders such as investors.

2 Background

In this chapter of the dissertation, all the methods and algorithms used to come to results and conclusions will be explained. Additionally, the most relevant works from the existing literature will be summarized.

In general, the research showed a variety of papers concerning sports prediction methods, with basketball being the most "famous" sport choice for research. However, it was lacking when it came to motorsports, especially in Formula 1. For the purposes of this dissertation, existing research concerning NASCAR analytics was used to extract the information needed for Formula 1, in addition to the few papers researching specifically Formula 1.

2.1 Data Mining and Machine Learning

By combining techniques from database systems administration, Machine Learning (ML), and statistics, Data Mining (DM) aims to identify and extract significant and meaningful patterns within data. Additionally, its aim is to identify relationships from large datasets (Han et al., 2011). Classification, regression, clustering, and association rule learning are the methods used in Data Mining (Müller & Guido, 2017).

Data Mining and Machine Learning use mostly the same algorithms. However, their end use is different. Compared to Machine Learning, Data Mining uses the same algorithms to find patterns, trends, and relationships in data. Its primary goal is the discovery of hidden knowledge (Han et al., 2011). Among its methods mentioned before, the most famous data mining techniques, used in sports analytics are clustering, classification, and regression, and thus these methods are explained in more detail.

On the other hand, Machine Learning uses the same algorithms to learn from the data. The models are trained and then can make predictions or decisions without necessarily being explicitly programmed to do so. Using the same methods, classification, and regression, its primary goal is predictive analysis (Watt et al., 2016). Additionally, Machine Learning can use reinforcement learning (Andriy Burkov, n.d.) but it was not used in this dissertation, thus not mentioned further in this Chapter.

To sum up, the models explained and used in this dissertation are Classification and Regression. Both methods are used for knowledge discovery and predictive analysis. Meanwhile the Clustering methods is used mainly in preprocessing steps, in knowledge discovery. With the mentioned techniques, Data Mining enables the discovery of trends and insights within data (Van Kesteren & Bergkamp, 2023) and it categorized into three stages. First the data preparation, then model building, and finally the results interpretation. These stages are also used in this research.

The mentioned methods mainly use statistical models. These statistical models are mathematical representations that describe relationships between variables in given data(Watt et al., 2016). There are multiple reasons, why these 'old' statistical models are still in use. Firstly, their evolution has influenced the data-driven modelling, using historical data to find the relationships. Moreover, their effectiveness has widened their use. Lastly, they are used as the foundation for more complex algorithms(Grover & Mehra, 2008).

Now that Data Mining and Machine Learning as concepts and as methods used are clearly stated, a clear distinction between classification and regression is needed. The reason that both are modelling techniques in use is that they are used for different problems. Both are the two primary supervised learning techniques, used in predictive modelling.

Supervised learning is a paradigm, used to mention models that is trained on labelled dataset, where each input X corresponds to output Y (Watt et al., 2016). This technique allows the algorithm to train and then predict the target value for new unseen data(Han et al., 2011). Supervised learning has great application in business environments (Shmueli, 2010). Supervised learning comes in contrast with unsupervised learning, the method using unlabelled data. Unsupervised learning is further analysed in the clustering section.

Classification and Clustering, being supervised learning methods, is now clear rely on labelled data. So, what is their difference? Their output differs, in classification we have discrete values as outputs, while in regression, continuous (James et al., 2017). By discrete values, the values representing separate items are included. Discrete values can be counted individually. These values are typically represented by integers. On the other hand, with continuous values, measurements that can take any values in any range are included. While continuous values are not countable, they are measurable and are usually represent with fractions or decimals.

2.1.1 Classification

Classification is a method, as it was mentioned above, of supervised learning, thus predicting the categorical label of new data, based on patterns learned from already labelled datasets. The trained model assigns the class label to previously unknown- unseen data (Black et al., 2023). Among the many classification modelling techniques, the most common methods for classification include decision trees, k-nearest neighbour, Support Vector Machines (SVM) and naive bayes (Andriy Burkov, n.d.).

When it comes to classification, the models distribute the inputs into distinct groups named classes, thus the name classification: the separation of data into classes. A widely mentioned example among data scientists is identifying whether an email is 'spam' or 'not spam' (Han et al., 2011). In the area of sports prediction, there are studies, that are also presented in this dissertation, where the classification model is used. In these cases the models can have outcomes, such as win or lose. The classification between two options is called binary classification. However, classification can have more than just two outcomes, for instance the options in win, tie, or lose.

Support Vector Machines (SVM)

Support Vector Machines (SVMs), traditionally used for classification, are supervised learning models that can however be used for both classification and regression tasks (Andriy Burkov, n.d.). Concerning the regression use of it, they will be analysed later in this section.

With regards to the classification task only, the primary goal of SVM is to find the optimal hyperplane that separates data points of different classes in a high-dimensional space while maximizing the margin between the classes (Müller & Guido, 2017). The output of the model is different classes for the data.

Maximizing the margin is one of the SVMs' primary characteristics, one of its key factors. SVMs aim is establishing a decision boundary that has the maximal possible distance. This boundary is the so-called margin, between the closest data points from each class. Support Vector Machine get their name for the fact that the algorithm utilizes the data points, that are the closest to the said margin. These points are also known as the support vectors (Black et al., 2023; Watt et al., 2016). By focusing on these points, the support vector, this algorithm, can thus set the maximal possible margin and ensuring that this is the optimal one.

Another important key feature is that they can also handle non-linear data. The separating boundary does not need to be a line. Even though they were first use for linear separation it is not necessary. Using kernels, for example the polynomial of the famous RBF (Radial Basis Function) SVM can handle and separate nonlinearly mapped data (Watt et al., 2016). The kernel is nothing more than mathematical formulations used to transform the data.

SVMs even though as a model is old, they are still widely use. The reason for this is that they can very effectively work on high-dimensional spaces and finding hyperplanes. Additionally, they are considered to be robust to overfitting (Watt et al., 2016). The use of SVMs is less common compared to decision trees and ensemble methods (R. Bunker & Susnjak, 2022).

Bayesian

Another model that fits under the category of classification is the Bayesian model. The Bayesian model is using the probabilistic framework of bayes to predict / determine the category or rank based on the features (data) provided (Andriy Burkov, n.d.). The Bayesian models are built on the bayes theorem and assume that the features are conditionally independent. Using this theorem the Bayesian models estimate the likelihood of a data belonging to a particular class.

Decision Trees

Another famous classification methods are the decision trees. It is a common method and powerful. Decision trees pose questions at each node and then partition data into subsets based on feature values, mimicking a tree-like model structure (Han et al., 2011). The nodes represent features or questions, the branches the outcomes and the leaves the final prediction, decision, the class the data belong to.

Among the advantages of these models are their interpretability. Decision Trees offer simple yet powerful insights into the data. They are versatile and act like a white box, a model that is easily interpreted (Han et al., 2011). Additionally, they are fast (R. Bunker & Susnjak, 2022). Moreover, they are frequently chosen by analysts because they can handle non-linear relationships effectively and they require minimum preprocessing since they can work with raw data, without normalization or scaling. Another important advantage is that they can handle both categorical and numerical data (Watt et al., 2016) (Han et al., 2011).

Despite their significant advantages, decision trees tend to be prone to overfitting. This happens if the model becomes too complex and deep. Also, they are biased towards small chances in the dataset. Lastly, they can become computationally expensive when dealing with large dataset (Han et al., 2011).

All these disadvantages of Decision Trees led the scientist in search of a better solution, but while keeping of the important advantages of the said structure. This need led to the use ensemble methods such as random forests (Han et al., 2011). Ensemble models, combine multiple models to improve prediction accuracy and improve generalization. The accuracy of these ensemble models, the committees of models, is higher.

Random Forests

From the various Ensemble Methods The Random Forests specifically combines all the advantages of decision trees, without carrying the load of its disadvantages (Han et al., 2011). With the ensemble techniques one can retain the interpretability while at the same time boosting accuracy (R. Bunker & Susnjak, 2022)**.**

More specifically, Random forests are an ensemble learning method that aggregates predictions from multiple decision trees. As an aggregated method, an ensemble method, they combine outputs and generalize better than their cousins, decision tree models. The model can generalize through majority voting (for classification) or averaging (for regression). Consequently, random forests enhance predictive accuracy and reduce overfitting (Han et al., 2011).

In sports predictions, random forests can combine features like the player’s skill, team strategy and historical data to predict outcomes with relative high accuracy. Additionally, these algorithms including Random Forests and Boosting algorithms, are frequently used due to their robustness against overfitting. Moreover, in (R. Bunker & Susnjak, 2022) variants like CART (Classification and Regression Trees) and Logistic Model Trees (LMT) are noted for high accuracy in sports like American Football and Basketball, respectively.

2.1.2 Regression

Regression is another old statistical method, used to examine the relationship between a dependent variable and one of more independent variables. It enables the user to identify relationships among multiple factors and predicting continuous outcomes based on the input (Schneider et al., 2010). Regression is of many types, the ones discussed in this paper include - Linear Regression, Lasso Regression, Support Vector Regression, Ridge Regression, Polynomial Regression (Tatachar, 2021). As stated above, in regression, the model predicts continuous numerical outcomes, such as forecasting house prices based on features like size and location(Han et al., 2011). In the context of sports predictions, continuous values can be predicted times gaps or ranked positions.

Linear Regression

There are many forms of regression, for example the linear regression. Linear regression is a technique used to predict a continuous target variable. its main goal is to model the relationship between the predictor and the target fitting them on a straight line (Watt et al., 2016). The model finds the optimal values, by minimizing the sum of squared errors, which measure the differences between the predicted and the actual values.

Among its strengths is the simplicity, the speed, how the coefficient provides insights to the direction of their relationship and the strength of the relationships. However, the model has its weakness. Primarily, it assumes linearity and normality and it is extremely sensitive to outliers (Watt et al., 2016).

A regularized version of it, is the **Ridge Regression**. Ridge Regression, also known as L_2 regularized regression, is a linear regression technique that addresses multicollinearity in datasets by introducing a penalty term to the ordinary least squares cost function. (Müller & Guido, 2017), (James et al., 2017). It is particularly effective when the predictor variables are highly correlated or when the number of predictors exceeds the number of observations. It ensures robust predictions in high-dimensional datasets. In sports prediction it can be useful to predict and athletes race times.

Logistic Regression

The logistic regression is another method. This kind of regression is often considered a classification method under a different name, however for the text's continuity it is presented here. The logistic regression is valued for its reliability and computational simplicity. Logistic Regression, by contrast to the rest of the regression n methods, is widely used mainly for binary classifications. As mentioned above, binary classification is a type of assigning data into just two classes- groups. For instance, it could be used in win/loss scenarios in sports.

Its simple implementation and its ease in understanding both the process and the results, make it a valuable tool. Logistic regression is deemed as a valuable method for understanding the influence of different factors on match outcomes (R. Bunker & Susnjak, 2022) . These characteristics of the model, have solidified Logistic Regression as a foundation in predictive modelling in sports area, especially when the analysing data call for binary outcomes

Support Vector Machines

SVMs, are not only used in classification scenarios. These models can also be adapted for regression tasks too. through Support Vector Regression (SVR) the model identifies a hyperplane that minimizes the error. The aim is again maintaining a defined margin. SVR is preferred for being particularly effective in high-dimensional spaces. Additionally, when the number of features exceeds the number of samples again, they are extremely effective.

The model in general, like in the classification scenario, balances well the trade-off of bias and variance and thus providing robust predictions.

Decision Trees

Decision Trees as algorithm can be used for regression tasks too. They can successfully support both classification and regression tasks (Han et al., 2011; James et al., 2017). They are used to predict a continuous target variable, by recursively splitting the data. The goal is to model the relationship between the input and the continuous output by partition the data to regions, where each region is as similar as possible.

Among the advantages of using Decision Trees for regression is that they are easy to interpret, since the tree structure is easy to visualize and humanly understand it.

Additionally, it supports nonlinear relationships and lastly, as mentioned in the classification part the data can be used raw. There is no need for scaling and normalization.

Among the disadvantages, as mentioned above, is the potential risk of overfitting with trees too complex and deep and the sensibility to minor changes, since they can change the tree a lot.

As an extension of the Decision Trees, the ensemble model of random forests can apply the same principles they did on the classification task to continuous data, making it a versatile tool for regression tasks.

2.1.3 Clustering

While classification and regression are supervised techniques, clustering is not. As an unsupervised method, the model is trained on unlabelled data. In other words, the model is given data without instructions about the output and the target, and the goal of the algorithm is to find patterns, structures, and relationships (Watt et al., 2016).

Clustering's goal is to group data objects into clusters, meaning groups based on their similarity. The model's goal is to ensure that the data points within each cluster are similar to each other and dissimilar to the rest of the data. This method is useful to discover patters, and in this dissertation, it was used like so.

K-Means

K-Means is a widely used unsupervised learning algorithm designed to partition a dataset into k distinct clusters. It is an iterative algorithm that groups data points based on their similarity and assigns them to clusters, minimizing the intra-cluster variance(Han et al., 2011).

As a process K-Means first select the k initial cluster centroids randomly or using predefined heuristics. Then, the model assigns each data point to the nearest cluster centroid based on a distance metric (e.g. Euclidean distance). Then come the update. It recalculates the cluster centroids as the mean of all data points assigned to each cluster. Finally, it repeats the assignment and update steps until centroids stabilize or a predefined number of iterations is reached(Han et al., 2011). This repetition of the steps makes K-Means an iterative method.

K-means are used due to their simplicity and computational efficiency. Additionally, it is a method easy to implement. These reasons make K means a preferable choice for exploratory data analysis. Moreover, it is scalable and flexible.

On the disadvantages of the method, one can include the need for predefined k, number of clusters. Additionally, it is sensitive to a poor initialization, which can lead to suboptimal clustering results. Moreover, K-Means is sensitive to outliers, as they can significantly distort cluster centroids.

K-Means clustering has significant applications in sports analytics, particularly in segmenting players, teams, or events based on performance or other metrics. Clustering players based on attributes such as speed, stamina, and skill levels can help coaches devise training strategies tailored to specific groups. Teams can be grouped based on performance metrics like win rates, goal differentials, or possession statistics to identify patterns and competitive advantages. In sports like football or basketball, clustering game events (e.g., shots, passes, turnovers) can uncover tactical insights and improve strategic planning. By analysing fan behaviour data, K-Means can cluster fan segments for targeted marketing campaigns or stadium seating arrangements.

2.1.4 Data Prepossessing

Data preprocessing is an essential and pretreatment step in the data analysis and machine learning method. It is usually called a pipeline. It involves transforming raw, sometimes unstructured, data into a clean and structured format. This happens because models need a correct format of data that suitable for analytical tasks. This process includes several key operations.

Data preprocessing includes the step of **data cleaning**. This is usually the initial step. This step addresses the problems of missing values, correcting errors, and removing inconsistencies, when they exist in the data. **Data integration** is another critical step of data preprocessing. The analysts, in this step of the preprocessing, combines data from multiple sources. At the same time, the focus is also in resolving discrepancies in naming and additionally in eliminating redundancies. These two preprocessing steps' goals are to improve data quality. This happens to ensure the reliability and accuracy of subsequent analytical and predictive models.

In addition to the first two steps, another preprocessing task is the **data transformation**. Data transformation converts raw data into suitable formats for analysis. In this step often normalization and discretization are included. The first process of the two is the scaling of data attributes to a specific range (e.g., [0, 1]). On the other hand, discretization, separates into groups continuous values into categorical ranges. For example, discretization can convert the variable "age" into labels like youth, adult, and senior (Han et al., 2011). These transformations make multi-level abstraction easier, which in its turns makes data mining and interpretation of the output more effective.

Finally, **feature extraction** plays a crucial role in choosing the most pertinent attributes for each problem, or each analysis. There are various techniques that can be used to select these "most important feature" and to determine the most useful variables. It is important to note that **data reduction** methods aim to minimize the dataset size without compromising the quality of the outcomes. When successfully achieving this, it is ensured all the information needed is retained.

All the steps mentioned above ensure that raw data is prepared in a way that optimally supports complex analysis and any algorithm the analysts wish to implement.

Correlation analysis and Mutual Information

When building predictive models for sports, such as Formula 1, both **correlation analysis** and **mutual information** are crucial in the feature selection process. Additionally, they play a very important role in understanding the relationships between variables. These two techniques complement each other in identifying critical features, both linear and nonlinear and improving model accuracy. For this research they were part of the feature extraction step.

Correlation Analysis

Correlation analysis is a statistical method that was used to get the most important feature for making accurate prediction. It evaluates the strength of the linear relationship between the variables. Simply put, correlation analysis measures the statistical relationship between two variables, the change in one variable due to the change of another.

This analysis finds if there is an existing relationship between two variables. It indicates both the strength, how strong two variables are related and direction, positively or negatively of the association of two variables. By positive correlation it means that both are moving in the same direction, one increasing the other does too, and by negative it means that when the one increases the other decreases.

The two most common measures used in Correlation Analysis is Pearson's correlation and the Spearman’s rank correlation. The Pearson’s coefficient for linear relationships measures the linear relationship between two continuous variables. The Spearman's rank correlation measures any monotonic relationships and is used mainly when the data do not follow a gaussian distribution(Rovetta, 2020) .

Mutual Information

Mutual information was also used in the feature extraction part. Mutual information was additionally picked in this research to identify non-linear relationships. Since sports predictions includes many complex relationships, correlation analysis was not enough on its own.

Mutual information measures the quantity of information obtained about one random variable through another. It measures the reduction in uncertainty of one variable given knowledge of the other. With this method, it manages to capture both linear and nonlinear dependencies. This metric is valuable in the process of feature selection as mentioned above. The importance hides in helping to identify variables that share significant information with the target variable(Belghazi et al., 2018).

To compare these two and discuss as to why we need them both it is important to note that while correlation analysis is efficient for identifying linear relationships it is not enough. Mutual Information can identify nonlinear patterns that Correlation analysis might have overlooks, making both essentials. Thus, both simple-linear and complex-nonlinear relationships are successfully considered when these methods are combined during the feature extraction step. The inclusion of both the above measures leads to reliable predictions since the variables affecting the outcome are successfully identified. Finally, the use of those two measures can reduce overfitting and thus improving the model's interpretability, enhancing its predictive accuracy.

2.1.5 Evaluation Metrics

It should be highlighted that performance metrics are different from loss functions. Loss functions show a measure of model performance, they are used to train a machine learning model (using some kind of optimization like Gradient Descent), and they are usually differentiable in the model's parameters. Model performance metrics on the other hand, are used to monitor and measure the performance of a model usually after training, and don't need to be differentiable (Plevris et al., 2022).

Evaluation metrics are essential to assess the performance and the effectiveness of statistical or machine learning models. It is important to note that rarely one measure is enough. By using different metrics for performance evaluation, one can improve the overall predictive power of the mod (Plevris et al., 2022). These metrics provide valuable insights to the reliability of the predictions. For the regression tasks such as the one in this research the most important metrics to compare the performance of the models are the Mean Absolute Error, the Mean Squared Error, and the R squared.

Mean Absolute Error

Mean Absolute Error (MAE) is a metric used to evaluate the accuracy of a regression model. It corresponds to the average (mean) absolute differences (error) between the predicted values against the actual values. It uses the l_1-norm loss. The metric provides a simple measure of model prediction error in regression tasks. MAE is less sensitive to outliers compared to metrics like Mean Squared Error. The MAE is estimated of n number of observations and corresponds to the difference in the actual values and the predicted as the following formula explains.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

Where:

y_i: Actual values

$\hat{y}_i$: Predicted values.

n: number of observations

In general, it is used to provides a clear and intuitive measure of the average error. It is easy to understand and additionally, it is a metric robust, like mentioned above, a measure less sensitive to outliers, especially compared to Mean Squared Error

Mean Squared Error

Mean Squared Error (MSE) is another metrics used in evaluation regression tasks. It measures the average difference of the squared values between the predicted and actual values. This way this metric gives an emphasis on larger error, since it is squaring them. The formula below explains the metric (Tatachar, 2021).

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

Where:

y_i: Actual values

$\hat{y}_i$: Predicted values.

n: number of observations

Even though the measure squares the error is widely used, because it magnifies even the significant deviations. This exaggeration of the errors the models makes, it makes the metric ideal to point out the larger errors, which are not desirable in any predictive model.

R squared

The last metric used in this research is the R-Squared (R^2) or the coefficient of determination. This metric measures the proportion of variance in the dependent variable that is explained by the independent variables. In these cases, if R 2 is calculated as the ratio of the variance explained by the model to the total variance used mainly in nonlinear regression (Plevris et al., 2022)

$$R^2 = 1 - \frac{SS_{residual}}{SS_{total}}$$

Where:

$SS_{residual}$: Sum of squared residuals

SS_{total}: total sum of squared

The used of R^2 is to measure the goodness of fit of the model to the data. It evaluates how well any model fits and captures the variability of the target variable. However, R^2 alone cannot stand as a metric, that's why the other two metrics were used as well.

In conclusion, the value of evaluation metrics is very important in assessing any model's performance. Without being able to evaluate the performance of the trained model it is impossible to value its worth (Tatachar, 2021).

Concerning the sports analysis domain and its complexities, evaluation is particularly needed. In this researcher's regression task, each of the chosen metric provides useful insights into the model's predictive accuracy, error behaviour, and overall reliability. The combination of these metrics ensures a comprehensive understanding of each model's strengths and weaknesses, leading to better predictions and clearer for the user insights.

2.2 Data Science for Sports Prediction: Applications and Challenges

Sports Analytics is a recently risen area of statistical studies and rapidly growing. It involves all the processes of data collection and data analysis in order to extract useful information concerning the sports industry (Sarlis & Tjortjis, 2020). The useful information derived from the sports analysis can be the outcome. This is called sports predictions, the area this dissertation is interested in. However, there are more areas of interests in sports analytic. Statistical analysis can also be leveraged to interpret and optimise team strategies, as well as financial aspects of the sports. Additionally, Sports Analytics can be used to predict injuries and their impact (Papageorgiou et al., 2024b)(Sarlis & Tjortjis, 2024).

In sport analysis Data Mining (DM) and Machine Learning (ML) techniques are to process historical performance data and players statistics to produce predictions and outcomes that are in the interest of the analyst. In addition, they can be used to analyse contextual data, such as the weather. In general, they can be used to make predictions and forecast and additionally to extract useful insights

Machine Learning and Data Mining both use many of the same algorithms. For the purpose of this study, for Machine Learning, Multilayer Perceptron (MLP) and Artificial Neural Networks (ANNs) will be deemed as its algorithm while Classification, Clustering and Regressionn are deemed as data mining methods.

Concerning making predictions about the final outcomes, in general, sports analytics can be categorized into three primary approaches (R. P. Bunker & Thabtah, 2017):

1. **Classification**: These algorithms labels/assigned outcomes into different distinct categories based on the available historical data. In the context of sports predictions, classification is usually used to make predictions of two outcomes, the so-called binary classification, for instance like win or lose. Classification can also be used for more than two classes that mentioned

before, for example in win -tie-lose scenarios in match outcome predictions.

2. **Regression**: These algorithms model continuous variables and finally predict the outcome. The outcomes are continuous values in comparison to the distinct classes of classification. These methods can be used to forecast time gaps or position ranks. Generally, regression delivers detailed outcome predictions.
3. **Clustering**: In contrast with the two previous methods, clustering is an unsupervised method of modeling. Without being trained on labeled data, the algorithms try to find similarities. With clustering methods the sport analysts' groups races, matches or athletes into groups based on similar performance metrics. This way the models aid the analysts in understanding broader patterns.

Concerning these three major categories of DM, which were analyzed in detail in previous sectors of the dissertation, are deemed on the more traditional statistical side, they thus rely mainly on historical data. Research indicates that models relying solely on historical data typically exhibit moderate predictive accuracy (Haghighat et al., 2013). Nevertheless, early statistical methods, for example the famous regression analysis, have played a crucial role in uncovering patterns and identifying hidden knowledge.

Their relatively poor performances lie in their inherent limitations. As statistical models, it is known that they provide just a simplified version of the hidden relationships. Consequently, there is a risk of misleading results if assumptions are not correctly set (Matsui et al., 2021). However, their adaptability to different sports, keeps them still relevant (R. P. Bunker & Thabtah, 2017). Another drawback is their sensitivity to minor dataset variations. Due to the influence of data inconsistencies, the models can result in drastically different outcomes (R. P. Bunker & Thabtah, 2017).

Nonetheless, the majority of the research on the statistical models already indicates the need to improve these models. In order for the improvement to happen, research emphasizes the importance of capturing suitable prediction models. These models should be capable of integrating both historical data and predetermined features to capture non-linear relationships and improve their accuracy (Haghighat et al., 2013). It is advocated that the hybrid models that combine various methodologies can maximize the predictive potential of diverse datasets.

However, even when using just historical data, artificial neural networks (ANNs) have demonstrated improvements in accuracy. Because of their capacity to capture and represent complex nonlinear relationships, ANNs have been extensively used in predicting match results. In many cases, due to their nature, they outperform traditional models (Haghighat et al., 2013). However, their primary drawback is their lack interpretability compared to other modelling approaches (R. Bunker & Susnjak, 2022).

2.2.1 Relative Work – Model Examples

In this section of the dissertation, the evolution of models, results and challenges of recent and older papers on predictive modelling are presented. It begins with the foundational DM methods and their evolution is explored in sports prediction, particularly motorsports like Formula 1. A comprehensive table, Table 1 (see page 33), is presented with the summarization of the relative work for prediction. Additionally, there is the table for the features important in making prediction in motorsport, Table 2 (see page 39).

Making Predictions

Research conducted by (Graves et al., 2003), one of the earliest found, uses **a probability model**, a Bayesian model, with a hierarchical framework. The model was used to predict outcomes in the finishing driver positions of a NASCAR race. Since it is a probabilistic model hypothesis testing took place. The null hypothesis was set as "random finish positions". The model tested, concluded that there is strong evidence against the null hypothesis making the alternative true, that there are not random final placements.

In this research it was noted that both the past races and additionally the driver's abilities, can assist in successfully predicting the outcome of future races, revealing that historical data can be leveraged. Additionally, these finding gives us information about features that can be used in Formula 1 predictions. Additionally, the study finds that the drivers have different abilities in different tracks, extracting the “track specialists”, a point helpful in F1 too.

Among the challenges faced in this research is trying to capture complex relationships, a challenge that can also be applied in Formula 1 as well as all the sports, since the data are usually both quantitative and qualitive. One other challenge faced here is the possible estimation biases that may occur when drivers participate in races with higher winning probabilities. Furthermore, the data scarcity challenge, which can limit the testing and thus the results is present, leading to limited generalization of the findings, a challenge encountered by many papers.

Moving on to the next paper, even as early as early 2000s the famous **Multi-Layer Perceptron** was used. This is not a traditional statistical model, but rather an artificial network. The infamous MLP model is used in one of the earliest research projects on predicting sport outcomes. Kahn (Kahn, 2003) applied a MLP model with a 10-3-2 structure to classify home team win/loss outcomes using just five features. The model achieved an accuracy of 75%.

Among the challenges of the study is again the data scarcity and quality problem. Additionally, this study came across other problems to, mainly with the nature of the MLP model. The model’s complexity makes it prone to overfitting and the challenge in determining the optimal hidden layers as well as the choice of the correct hyperparameters. Lastly the computation cost and the training time it takes for the MLP was a challenge to the author.

Another example of using **MLP** for predictions by McCabe and Trevathan (Mccabe & Trevathan, 2008)), who analysed data across football and rugby. An overall accuracy of 67.5% was attained by McCabe and Trevathan, who used an MLP neural network. The model came with a 20-10-2 structure. The model showed transferability across sports. This was a substantial contribution to the field of sport analytics. A major issue that the study brough to light however was a high level of dependency among features, complicating the modelling process.

In the next study examined (Davoodi & Khanteymoori, 2010) the authors employed a **multilayer feed-forward neural network** to predict finishing times of horse races. The prediction of finishing times leads to race standings. Davoodi & Khanteymoori (Davoodi & Khanteymoori, 2010) applied their ANN and achieved an average accuracy of 77%. The optimal model structure had eight input nodes, a hidden layer of 5-7 nodes, and a single output node and was trained with Backpropagation (BP). In their research they concluded that BP required longer training times and more parameter tuning, in contrast with the Levenberg-Marquardt (LM). Although the latter was the fastest in terms of training time it did not match BP's accuracy.

Concerning the area of Formula 1 This study is especially relevant for, since similar features, such as track conditions and race distances, play a significant role in the final race outcomes.

Moving forward and in more recent years the research conducted by Hucaljuk & Rakipovic will be examined. (Hucaljuk & Rakipovic, 2011) tested models including **ANN, Naïve Bayes, and LogitBoost,** both traditional and the ANNs for Champions League football matches predictions. The researchers obtained a 60% accuracy, by utilizing domain-specific predictors. The best performing model to be the ANN, specifically trained with backpropagation. They used domain knowledge as a dimension reduction technique, however, they noted that was a challenge of their research, because using this method for dimension reduction could be time-consuming and too tailored to specific datasets.

The next research examined is the one by (Blaikie et al., 2011). The paper, as part of conference proceeding investigates the effectiveness of **artificial neural networks** (ANNs) in predicting outcomes in two different leagues. The NFL and the college football games.

Among the most important revelations of the study is that the model performed well for the NFL, however it did not yield similarly accurate predictions for college football. The dataset used consisted of over 200 games from both leagues. The implemented ANN model achieved an average absolute error of 10–12 points per game for the NFL and 12–14 points for NCAA Bowl Championship games.

A "committee of committees" method was employed for the modelling part of the research. By combining different predictions from multiple committees, the authors managed to enhance accuracy. This way they contributed adding to the knowledge that the committees and the ensemble methods wield better results than training just one model.

The study underlined that in order to model effectively, there is the need of reducing the number of statistics. This reduction can be achieved either by simplifying the dataset or by utilizing larger neural networks, that are in turn capable of identifying patterns in larger datasets. The researchers noted that using too many statistics can lead to data redundancy, with certain features making only a small contribution to predictions. Reducing the number of statistics to a manageable level while retaining critical information is another challenge that affects model accuracy.

Finally, the models suggested in this research faced the risk of overfitting. The model seems to capture the noise in historical data rather than meaningful patterns, resulting in poor generalization to new data.

The next study examined, was on javelin throwers by (Maszczyk et al., 2014). They found neural models to be significantly more accurate than their nonlinear regression models counterparts. The authors used data from 70 javelin throwers to predict throw distances. After identifying four significant predictive features using a correlation matrix and regression analysis, then applied both **a nonlinear regression** model and **a multi-layer perceptron (MLP) neural network** with a 4-3-1 structure. The ANNs finally achieving an absolute error of 16.77 meters compared to 29.45 meters for regression.

The results of this study, demonstrates the potential of ANNs in producing high-quality predictions. This way they contribute to optimizing sports-related decision-making, such as athlete recruitment and selection processes. Nonetheless, their comparison was in contrast to traditional nonlinear models. Given the inherent nature of sports data, with their complex, nonlinear relationship, it should be assumed that the linear models should not be able to fully capture all the intricate relationships.

The final research examined concerning making predictions is by (Papageorgiou et al., 2024a). This research is about predicting the NBA's player performance. Using data from seasons 2011-2021 the authors compared multiple models. Apart from high prediction accuracy on each players performance the research showed the optimal line up of the player contributing also to the evaluation of sports strategy area.

In this research multiple models were trained on historical data of performance of the athletes and them compared with a comprehensive evaluation. This study concluded that the **Voting Meta-Model** was a robust choice, often ranking as the best or among the top performers, particularly due to its ability to combine the strengths of multiple models. Additionally, Random Forest also excelled in certain scenarios, particularly when handling bigger datasets, which had complex interactions. These results also support the fact that ensemble methods are more effective.

One of the challenges the authors faced was the NBA players' performances, which can be deemed as highly volatile. This high volatility can be linked to several factors relevant to sports like player injuries, team dynamics, or even player fatigue. Moreover, another challenge was that despite the strengths of models like Gradient Boosting and Random Forests, they might overfitting in not properly tunes

Table 1*:*

Relative Work comparison table

Authors	**Dataset**	**Models**	**Results**	**Sport**
Graves et al (2003)	91 races between 1996-2000	Tony Stewart	3.84 wins (predicted) vs 6 (actual)	NASCAR
		Jeff Burton	4.48 wins (predicted) vs 4 (actual)	

Kahn (2003)	208 games, 2003 season	MLP	75% accuracy	NFL
Davoodi & Khanteymoori (2010)	100 races in January 2010	ANNs	77%	Horse Racing
Hucaljuk & Rakipovic, 2011	96 matches	ANNs with BP	68%	European Champions League.
Blaikie et al (2011)	Over 200 games	ANNs	Absolute Error of 10-12 points	NFL
Maszczyk et al. (2014)	70 throwers	Nonlinear regression model MLP	Absolute Error: 29.45 meters Absolute Error: 16.77 meters	Javelin Throwing
Papageorgiou et all (2024)	Seasons 2011-2021	Multiple models compared		NBA

Evaluating strategies

Moving on with the review of existing literature, another area of sports prediction is reviewed. This part is dedicated to the models for evaluating race strategies. Compared to the predicting outcomes, evaluation of strategies has more information on Formula 1. In recent years there were some important researched done to evaluate tire strategies for Formula 1 racing. However, due to lack of data concerning tire use and pit stops, tire strategy falls out of the scope of the dissertation, however the most interesting findings are listed below.

First, Heilmeier, Graf, et al' research is reviewed. In this research (Heilmeier, Graf, et al., 2020) the authors use Monte Carlo simulations to robustly evaluate the race strategies in Formula 1. This evaluation is conducted by accounting for probabilistic events such as accidents and full course yellow phases. Their model is used to identify the optimal strategy. Moreover, the study identifies and models critical probabilistic factors, including starting performance, lap time variability, and pit stop durations.

Another example of machine learning, specifically ANNs methods being used in sports predictions, are in the research by (Heilmeier, Thomaser, et al., 2020) . The research evaluates the strategy using the Virtual Strategy Engineer (VSE) and is specialized in Formula 1. Among the most important findings of this paper are that the VSE, is able to make reasonable decisions and react to the particular race situation.

The VSE method uses two ANNs to make the race strategy decisions, one for pitstops and the other for tire compound. The networks are trained on timing data from the six seasons from 2014 to 2019, with various filters applied to the data to remove irrelevant or noisy data. The VSE improves a race simulation's realism and can support a real strategy engineer in his decisions (Heilmeier, Thomaser, et al., 2020).

The most recent study found was by (Aguad & Thraves, 2024). In order to optimize the F1 pitstop strategy, they employed a model. Their model considers two drivers at a time competing in a race. Each driver deciding at each lap whether to continue on-track or to pit and switch tires. The tire option is one of the three available compounds. Their model allows for different features of uncertain event to be included for example yellow flags or randomness in lap times.

The key findings of the study show that drivers' distinct objective functions conclude different race strategies. Players are more likely to take risks when they aim to maximize their chances of winning rather than the time difference with their opponent. Additionally, the study also finds that the probability of winning is increased by 15% when a strategic driver is up against an opponent who does not care about competition. Finally, yellow flags tend to increase the winning chances of the driver with the worst performance

2.2.2 Formula 1 Important Features

Numerous studies stress the significance of feature selection in predictive modeling, especially when it comes to sports analytics. While not a standalone algorithm, feature selection, selecting only a subset of the features, is used for improving model accuracy, its robustness and its generalization. (Blaikie et al., 2011). In predictive analysis there are significant variables/ features like teams and drive performance statistics, weather, or track data that often outweigh the sheer volume of the data (R. Bunker & Susnjak, 2022). To identify the critical factors influencing race outcomes in Formula 1, the following research where examined. The choice of researched were mainly researched concerning Motorsports.

In the recent research from (Patil et al., 2023) several important determinants for points accumulation at the end of races were identified. To come to these results Patil et al. (2023) analysed data from the 2015–2019 seasons. Their work emphasizes how important is the need for detailed feature analysis when forecasting race outcomes.

One clear indicator is the **number of Laps Completed.** This is a measure of race performance and reliability of the driver and the car. Secondly, there is the feature of **Laps Spent in Second and Third Positions.** This feature indicates the consistency of competitiveness during the race. A third important feature is the **Average Pole Position,** that reflects the starting grid advantage of the driver and its correlation with race results. Lastly, **Tire Choices.** This feature has made a more important impact especially when the intermediate tires under wet conditions were worn. This feature has been critical in strategy outcomes.

The next study is by Pfitzner and Rishel (Pfitzner & Rishel, 2008). Several features were identified as important predictors of NASCAR race. Even though it is on NASCAR and not Formula 1, the findings of the papers are still relevant, thus mentioned here. Features like car speed, driver characteristics, team attributes, and performance in related races demonstrated strong relationships with the finishing ranking of the races. Specifically, factors including qualifying speed, pole position, practice times, points scored in the prior year, laps completed in the prior year, and the number of cars or drivers on a team were positively correlated with race results. Simply put, the driver ranking, the performance streaks play this role into result prediction.

The conclusions are indeed significant for Formula 1 too; however, their research was based on a small sample size (only 14 races), however the paper provides a robust framework to analyse performance across different tracks.

Silva and Silva (Silva & Silva, 2010) draw on the previous model by Pfitzner and Rishel and improved on it. They employed **Spearman's rank correlation coefficients and chi-square tests to test** the relationship of qualifying performance and previous race results with the final finishing positions. These tests were applied in F1 datasets and not in NASCAR. The results showed that the driver qualifying position, and past race performance has a positive correlation with finishing position for the F1 data. Silva and Silva (2010) further highlighted that for F1, qualifying performance was the most reliable predictor, whereas NASCAR's reliance on cumulative season performance

The next study that was analysed presents an innovative ranking of Formula 1 drivers and teams to rank and to identify the best driver and find if their abilities affect the teams (Bell et al., 2016). The model suggested here, really depicts how drivers perform against their teammates. The **cross-classified multilevel model** has potential to assess the importance of a wide range of determinants and highlights that team effects were more significant than driver effects, with this disparity increasing over time. Additionally, the importance of team effects was reduced in wet weather and on street tracks. Comparing driver contributions to team contributions, they have concluded that the car is more important than the driver when it comes to race results. These results can be applied to this Formula 1 prediction models, keeping in mind that the features representing the driver are not important.

Based on the model mentioned just before, by Bell et al., a **Bayesian multilevel rank-ordered logit regression model** was constructed by (Van Kesteren & Bergkamp, 2023). Their research focused on the present F1 era, the hybrid era, with data from 2014-2021 in contrast with the previous research, which focused on the whole the history of the sport like (Bell et al., 2016). In their work they model individual race finishing positions, and the results were estimated using Hamiltonian Monte Carlo sampling with 8 chains of 1250 samples each after 1000 burn-in iterations.

According to this study the suggested model captures the data accurately, allowing for precise inferences about driver skill and constructor advantage. The results show that Hamilton and Verstappen are the best drivers in the hybrid era, and the top three teams (Mercedes, Ferrari, and Red Bull) clearly outperform other constructors before the 2021 season (Van Kesteren & Bergkamp, 2023). Approximately 88% of the variance in race results was explained by the constructors

Apart from the factors concerning the performance of the athletes and the teams, there are the external factors that the research notes as important. In this last part of the literature review finally, the impact of weather on the car’s performance has been reviewed.

In the research conducted by Saleh Mousavi-Bafrouyi et al. ((Saleh Mousavi-Bafrouyi et al., 2021) it was highlighted that **Air Temperature** has the biggest impact on the wheel force distribution. In the research it is explained that with higher temperatures the front-wheel force increases while the rear-wheel force decreases. This shift can lead to a shift in the balance of the car and alter its grip on the track. This fact is deemed extremely relevant for F1 predictions. Moreover, **Humidity and Wind Direction** influence the wheel performance, but their effect is to smaller extent. When looked at as a whole, these findings demonstrate the complex relationships between environmental conditions any cars performance (Saleh Mousavi-Bafrouyi et al., 2021). Motorsports and especially F1 outcomes, as any car would get affected by these environmental issues too.

In addition to the previous environmental characterises mentioned above, the temperature of the track plays an important role, especially when it comes to tire choices. According to the official Red Bull Racing website, the optimal conditions of the track for the tires are around 100°C. In hotter weather, tires functions well, however they might overheat fast and thus wear easily. On the other hand, in colder weather conditions, tires last longer but provide less than the ideal track traction. In conclusion, **Track Temperature is** another important environmental feature to consider.

Finally, there is another 'feature' in sports that is nearly impossible to quantify. This factor is luck. The study by (Haque et al., 2022) signifies the presence of luck in competitive championship. This presence of luck highlights that the complex, nonlinear feature-based models outperform simpler ones in sports forecasting. The research introduces the skill coefficient phi φ. This statistical measure is a metric that quantifies the deviation between observed scores and those expected under luck, incorporating contextual variables, for instance home and away games. Using a Bayesian model with a Poisson distribution and random effects, the study tries to identify the balance between luck and skill. However interesting this research is luck is not taken into consideration here.

Table 2:

Relative Work important feature listing

Research	**Sport**	**Important Features**
Patil et al, 2023	F1	#laps completed #laps spend in 2nd and 3rd position Average pole position Tire choice
Pfitzner & Rishel, 2008	NASCAR	qualifying speed pole position practice times points scored laps completed
Silva & Silva, 2010	F1	qualifying position, past races performances (e.g. points and wins)
Bell et al, 2016	F1	Teams affect more than drivers
Van Kesteren & Bergkamp, 2023	F1	Teams affect more than drivers 88% of variance is explained by constructors

Saleh Mousavi-Bafrouyi et al, 2021	Car Aerodynamics	Air Temperature Humidity Wind Direction
Haque et al., 2022	Various	luck

2.2.3 Research Gaps and Contributions

In conclusion, both statistical models and ANNs are used in sports analytics. There are many papers examining sports strategies, many of which were not included in this research since they fell out of the scope of this dissertation.

Concerning making predictions there is an extended use of traditional statistical models, however ANNs and especially the famous MLP wield better results in most cases. However, in both cases the use of more complex data, for example to include weather, is noted, since sports prediction is not always about the skill of one player or the team.

The first gap that was noticed in the research conducted was concerning Formula 1 and motorsports in general. First and foremost, there were not enough papers to review especially when it came to F1. There does not seem to be an area where sports analysis has been done extensively. There were quite a few concerning real-time data strategy predictions, but the volume of papers on important features and predictions on performance was not enough. Most sport analytics focuses on team sports like basketball of football, with minimal attention given to motorsports and individual sports in general.

Moreover, the prepares doing predictions, already noted the problem of the underutilization of weather data. The existing studies rarely incorporate environmental factors despite their known influence. Lastly, there is significant limited application of predictive modeming in formula 1.

3 Methodology

In this chapter of the dissertation, first there is going to be a brief overview of the methodology used. Next, there is going to be the data preprocessing part. At this subsection, the first part is going to be about the data cleaning. Handling missing values and removing possible outliers. Next there is the section of data integration. Merging the tables, resolving inconsistencies, and eliminating redundancies. Finaly there is the data transformation part. It includes normalization, encoding and aggregation. The last section of the methodology is featuring engineering. It could be incorporated in the data transformation part, but since it is an important part of the dissertation, which features play an important role in formula 1 prediction, it was described separately.

3.1 Methodology Overview

As derived from the relative work review in previous sectors of the dissertation the outcome of a Formula 1 race is influenced by a wide variety of factors. These factors include driver and team performance, as well as weather conditions. In addition to the factors just mentioned there are others that are not taking into account in this dissertation. For example, there is the tire use, the team budget, luck and the athlete's phycology or injuries. This dissertation follows the framework proposed by (Bunker & Thabtah, 2017).

For this research, the data was sourced from the Ergast API, covering for the years 2014 to 2024. The specific period was selected due to the significant changes in race regulations and technology and to keep the car's and teams' regulation the same.

This era is called 'hybrid era' and it starts with the introduction of the V6 hybrid engine in 2014. The previous years were the V8 era (2006–2008) and KERS period (2009–2013). Apart from the engine difference, 2014 came with major regulatory adjustments. For example, there were important pit stop modifications (refuelling and safety measures), and an increase in the number of races per season that further influenced race dynamics. The choice of the recent hybrid era was made for the research to be able to ensure data consistency and captures the modern evolution of Formula 1 racing.

3.2 Data Collection

For this study, data was extracted from multiple sources, primarily the Ergast API and supplementary datasets from Kaggle. The analysis focuses on the years 2018 to 2023, as reliable historical weather data was only available for these years. Pit stop data, sourced from "racefans.net," was available for 2018–2021, leading to two separate analyses: one incorporating pit stop information and one excluding it to evaluate feature importance.

The datasets used include:

- **results.csv:** Contains race results (driver, team, position, points).
- **circuits.csv:** Includes details about race circuits (ID, name, country).
- **driver_standings.csv:** Tracks driver standings across seasons.
- **constructor_standings.csv:** Tracks constructors' standings across seasons.
- **qualifying.csv:** Details qualifying session results (driver, grid position).
- **races.csv:** Information on individual races (race ID, year, round).
- **drivers.csv:** Information on drivers (ID, reference).
- **constructors.csv:** Data on constructors (ID, reference).
- **status.csv:** Status of race outcomes (e.g., finished, retired).

Additionally, weather data for the season 2018-2023 was available and retrieved from Fast API. The data include:

- **Time:** The timestamp for each observation.
- **AirTemp:** The air temperature is in degrees Celsius.
- **Humidity:** The relative humidity percentage.
- **Pressure:** Atmospheric pressure in millibars.
- **Rainfall:** A Boolean value indicating whether it was raining.
- **TrackTemp:** The track surface temperature is degrees Celsius.
- **WindDirection:** The compass direction of the wind.
- **WindSpeed:** The wind speed in km/h or another appropriate unit.
- **Round Number:** The corresponding race round in the F1 calendar.

- **Year:** The year of observation

3.3 Data Preprocessing

As mentioned before, preprocessing is a crucial step to get valuable results. In this dissertation, the first step was to integrate from the existing tables. The data was already in a format, appropriate for use. Secondly, data cleaning took place, handling missing values and eliminating redundancies. Lastly, feature extraction was applied.

3.3.1 Data Integration

The first step on Data Integration was the construction of the results dataset. The **results** dataset was first turn from a csv file to a data frame. After keeping only the important information, features; thus columns, the final dataset consisted of 12 columns and 3,947 for the 2014-2023 seasons. The columns were labelled as follows: ‘season’, ‘round’, ‘circuitRef’, ‘country’, ‘driverRef’, ‘grid’, ‘position’, ‘points’, ‘number_y’, ‘constructorRef’, ‘status’, and ‘time’. The dataset contained both numerical and categorical data, as almost all the data frame used in this dissertation.

To clarify the features, the suffix ‘Ref’ indicates a reference using a standardized abbreviation for names. For instance, ‘circuitRef’ provides an informal name for each circuit (e.g., ‘monza’ instead of the full name Autodromo Nazionale di Monza, or ‘baku’ instead of Baku City Circuit). Additionally, the feature ‘number_y’ represents the driver’s unique number assigned to each driver at the start of their career.

Moving on, the **circuits** dataset was constructed and turned the information from a csv file to a data frame. The first addition to this dataset was the classification of circuits as either "permanent" or "street." This mapping was manually constructed and sourced from the official Formula 1 website. To be able to aid in this analysis, the classification was converted into numerical values, explained in the transformation part.

The dataset consisted of 125 entries, for the data after 2018, and 204, for the data from 2014 to 2023 season with the following columns: ‘season’ (numerical), ‘round’ (numerical), ‘circuitRef’ (categorical), ‘circuitId’ (numerical), and ‘circuit_type’ (numerical). This combination of numerical and categorical remains for all datasets.

The next data frame analysed was the **drivers** dataset. This dataset consisted of 2,357 entries for the seasons 2018-2023, while for the expanded 2014 to 2023 period there were 4,131 entries. The dataset had the following columns: ‘season’ (numerical), ‘round’ (numerical), ‘driver_name’ (categorical), ‘driverRef’ (categorical), ‘code’ (categorical), ‘points_after_race’ (numerical), ‘pos_after_race’ (numerical), and ‘wins_after_race’ (numerical). This combination of numerical and categorical features enabled a detailed examination of driver-related data.

The next dataset analysed was the **qualification** data frame. This data frame had only 9 columns and 9,815 entries in total. A preprocessing procedure step involved converting the Q1, Q2, and Q3 times from object format to numerical values. A custom function was deployed to convert the times of each driver into seconds. Then the best qualifying time for each driver was calculated and then selected to be added to a new column containing only the best times of the driver. This column was named "qualies_best_secs.".

The final qualification dataset consisted of 2,297 entries for the 2018-2023 season, and 3,931 for the 2014-2023 seasons. Both datasets consisted of the following columns: ‘season’ (numerical), ‘round’ (numerical), ‘driverRef’ (categorical), ‘quali_pos’ (numerical), and ‘qualies_best_secs’ (numerical). Here it important to note that it was observed that there was a slight difference in the entries of drivers and their corresponding best times. The handling of this difference is described on the following part.

The **constructor** standings dataset was following. Minimal preprocessing was required for this dataset. It consisted of 1,150 entries for the 2018-2023 seasons and 1,979 for the 2014-2023 season. It consists of the following 7 columns: ‘season’ (numerical), ‘round’ (numerical), ‘con_name’ (categorical), ‘constructorRef’ (categorical), ‘con_points_after_race’ (numerical), ‘con_pos_after_race’ (numerical), and ‘con_wins_after_race’ (numerical). Again, there is the combination of both numerical and categorical data.

This dataset provided a straightforward structure for analysing the performance of constructors throughout the racing season, as the dataset contain information on the performance of each constructor.

Following up next is the **races** data frame. The final races data frame consists of 4 columns and 125 entries for the 2018-2023 and 204 races for the season from 2014 to 2023. The entries' summation number was double checked. The races from 2014 to 2023 were counted individually and then their sum was indeed verified to be 125 for the 2014-2023 and 204 for the 2014 to 2023 season. The four columns were, raceId, circuitId, season and round. The verification was additionally checked from the official formula 1 site and additionally the sum of races for each season were calculated and then summarized to get to 125 races. The races for each season are as presented in the Table 3.

Table 3:

Races count per season

Season – year	**Races-rounds**
2014	19
2015	19
2016	21
2017	20
2018	21
2019	21
2020* covid season	17
2021	22
2022	22
2023	22

A comprehensive table with the all the features, columns, used in for the basic dataset is presented in Table 4.

Table 4:

Description of data variables

Variable	Description
Season	The year of the season
Round	The number of the round
circuitRef_x	The unique reference name of the circuit of the race
Country	The country that the race takes place
draiverRef	The unique reference name of each driver
Grid	The starting position of the driver
Position	The finishing position of the driver
Points	The points for won at each race
Number_y	The number of the driver on the car
constructorRef	The unique reference name of each constructor
status	The finishing status of the race
circuitId	The unique ID number for each circuit
Circuit_type	The type (street or permanent) of each circuit
Driver name	The name (first name and surname) of each driver
Code	The three-letter code that each driver has
Points_after_race	The cumulative point of each driver after each race
Pos_after_race	The final position on the standings of each driver after the race
Wins_after_race	The cumulative number of wins for each driver after the race
Quali_pos	The position of each driver after qualification
Qualies_best_secs	The best time of the qualification for each driver in seconds
Con_pos_after_race	The constructors' position on the standings for each constructor
Con_wins_after_race	The cumulative wins for each constructor after the race

Moving on with the data for the weather information were analysed. The **weather** data frame consisted of 11 columns. To be more specific, these columns were 'season', 'round', 'Time', 'AirTemp', 'Humidity', 'Pressure', 'Rainfall', 'TrackTemp', 'WindDirection', 'WindSpeed', 'raceId'. This data frame contained 18,219 entries and included Formula 1-

specific data, such as track temperature instead of the temperature in general. It was joined with other data frames using common keys like 'season' and 'round'. Additionally, there was the ‘Rainfall’ column which was a Boolean column. The issue is addressed later, on the transformation part of the dissertation.

The "Time" column, representing minute-by-minute race data, accounted for the large number of entries in this dataset. The table describing the feature of the weather dataset is presented in Table 5.

Table 5:

Description of weather data

Variable	Description
Time	The minute-by-minute weather data for each race
AirTemp	The air temperature on the race day
Humidity	The humidity on the race day
Pressure	The pressure of the atmosphere on the race day
Rainfall	The existence or not of rain on the race day
TrackTemp	The temperature of the track on the race day
WindDirection	The direction of the wind on the race day
WindSpeed	The speed of the wind on the race day
Round	The number of the round
Season	The year of the season
raceId	The unique ID of each race

3.3.2 Data Cleansing

For this part of preprocessing, the following decisions were made. Firstly, duplicate entries for the "circuitRef" column were removed. After handling missing values in the "position" column (removing 321 non-finishers) and filling missing qualification times with 0.0 (for the 23 missing values, approximately 1.16% of the dataset), the data frame was ready for further analysis. With regards to the qualification times, the column with the best qualification time in seconds ‘qualies_bes_secs’, instead of 2297 rows it had only 2268 rows. Approximately 1% of the data were missing. This was due to some drivers failing to complete their qualifying sessions, either because of, crash of mechanic failure, or due to penalties, or personal issues. Since there were a few missing values only and verifying that there were all due to not completing the lap, it was set 0.0.

Concerning the weather dataset, when it was merged with the rest of the data, a small discrepancy of the data was noticed. It seemed that some rounds lacked weather data, and these missing entries (4.4% of the dataset) were excluded, since they could not be randomly filled not with the average values since the weather conditions are unique.

With regards to columns giving the same data and thus eliminating redundancies the columns, ‘country.’ ‘driver_name’, ‘code’, ‘con_name', 'time’, ‘Time’ were eliminated. Since every circuit has a unique name and particularly its unique ID the column country was not useful. Keeping the circuitRef give the same information. The driver_name and con_name were dropped since their ref names, driverRef and construcotrRef were kept. The ‘code’ column was also dropped for the same reason. Finally, the columns times were dropped since they do not give information to the predictive model but more to the data collection process.

3.3.3 Data Transformation

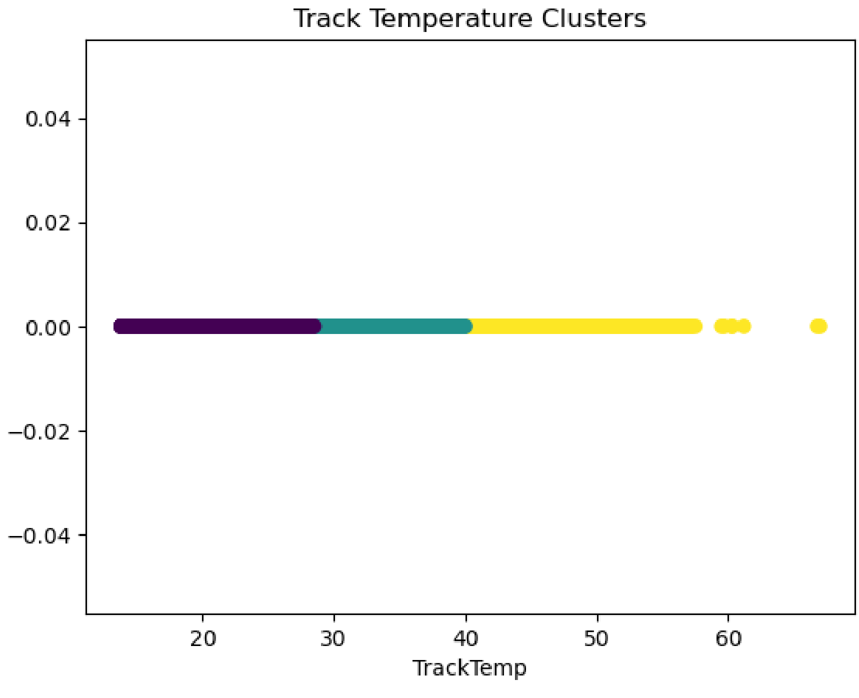
In this part, the data were transformed into a machine-useful way. The first transformation that took place was in the circuits data frame. To use the knowledge whether a circuit is permanent or street it weas needed to transform them into numerical values. Since previous papers suggested that there tracks expertise, in this dissertation it was deemed wise to check whether the status of the circuit played a role. The permanent circuits were assigned the value 0, while street circuits were assigned the value 1.

Next, on the drivers dataset, a new column, "driver_name," was created by combining the previously separate "forename" and "surname" columns. This provided a more comprehensive representation of each driver’s identity for analysis.

A K-means clustering algorithm was applied to classify track temperatures into cold, warm, or hot categories. The track temperature was selected since it was deemed more important than the atmosphere temperature. The ideal conditions were classified as "warm", as both cold and hot temperatures can negatively affect tire performance. These clusters were then labelled as follows: 2 for warm, 1 for cold, and 0 for hot. Following up there is the graph showcasing the three clusters that the track temperature was separated. It is clear from the figure that the ideal conditions were the smallest clusters, meanwhile the hot cluster seems to be the most populated and to have an extreme value too. The results are pictorially summarized in Figure 1. The clusters ere then named, 2 for the warm conditions, 1 for the cold and 0 for the hot conditions.

Fig 1:

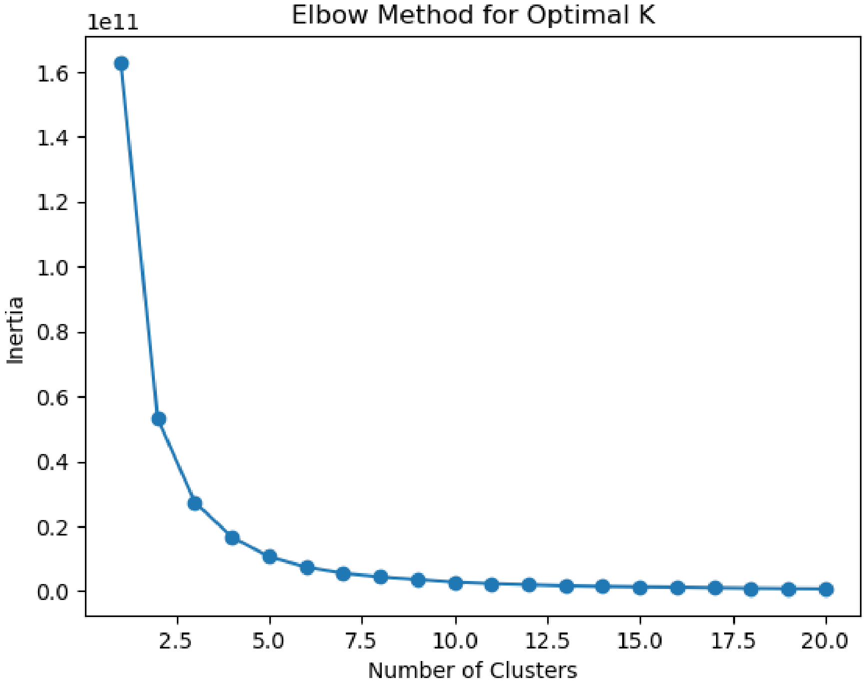
Clustering Track Temperature



K-means method was additionally employed in this dissertation to find the optimal number of timestamps to use for the weather information for each race. Th elbow method was used to determine the optimal number of time stamps. The depiction of the results is showed in Figure 2.

Fig 2:

Elbow method to determine the optimal number of clusters.



The number of timestamps used in each race for adding the weather is between 2 and 3. Adding more timestamps lead to redundancy. Moving on there had come categorical columns that needed to be transformed into a more useful way. The columns 'circuit-Ref_x', 'driverRef', 'constructorRef', 'status', were one hot encoded.

3.3.4 Feature Engineering

Before developing the models, in order to determine the feature that actually influenced the final position, the process of feature extraction was essential. This process helps to avoid overfitting and to improve accuracy of the model in forecasting the final rankings of F1 races. To determine the most important features, two methods were employed the correlation analysis and the mutual information analysis, to ensure that both linear and nonlinear dependencies will be revealed. The results of these two methods were compared to highlight the most important features.

At this point it is deemed important to note that two different analyses were conducted. The first analysis, called modeling with basic dataset, was the one excluding the weather data. On the other had the enhanced dataset was used and the analysis included the weather information. The choice of making two distinct analysis was made so the actual impact of these external factors needs to be highlighted.

A correlation analysis was conducted first for the data without weather for the season 2014-2023. The results of the analysis will be presented in a later part of the discussion. Concerning the methodology of implementing the method, the columns, 'country', 'driver_name', 'code', 'con_name' to avoid redundancies. Additionally, the categorical columns, 'circuitRef_x', 'driverRef', 'constructorRef', 'status' were encoded. The analysis was plotted using ‘matplotlib’ and ‘seaborn’ libraries. With the help of these libraries the results were visually prominent. The darker color highlighted the most important features. Specifically, the red color depicts the positive correlation while blue depicts the negative relationship between the two variables. The plots are presented in the Results Chapter.

Moving forward with the analysis, the mutual information analysis took place. For this analysis the library ‘sklear’ was used. Specifically, the feature_selection part of it. The results of the relationship of each feature with the final points were sorted in descending order and then the top ten features were picked as the most influential features.

Regarding the enhanced dataset, including weather and concerning seaosns 2018-2023, the same methods were followed. The columns to drop were the 'driver_name', 'code', 'con_name', 'time', 'Time', 'TempLabel' to avoid redundancies, or using columns that are helpful only to the analysts. Then the same categorical columns were encoded and the same algorithms used.

The only point to note is that there were some weather features, like Air Pressure and Temperature Cluster that were identified from the literature review but appeared less impactful than expected. In general, the weather features appeared less impactful than expected, challenging some prior assumptions about the weight of weather data in race forecasting.

3.4 Models and Evaluation Metrics

In this section of the methodology chapter of the dissertation, the models used for predictions are outlined. Apart for the predictive models used the evaluation metrics are also briefly presented here.

The selection of four models mentioned below was based on their abilities to handle the specific nonlinear characteristics of complex dataset, as well as the research goals in predicting racing position. Additionally, the research on the existing literature mentioned above as well as some additional reviews for the specific models. Each of the following model offers unique strengths Finally, the evaluation metrics that were chosen to ensure a whole assessment of each of the models' predictive performance and additionally aid in their comparison.

3.4.1 Data Models

First and foremost, it is crucial to remember that the problem at hand in this research is regarded as a regression problem. Predicting the final ranking of all 20 drivers at each rank is not a problem of assigning data into distinct groups, since there should be 20 groups, each group having one driver at each race.

Since the problem at hand is a regression problem, the **Random Forest Regressor** was the first choice. This model is an ensemble method, which combines multiple decision trees, offering a more robust performance for non-linear relationships (Smith et al., 2013), the relationships hidden in the data for F1 races. This power of capturing non-linear relationships, its accuracy, and interpretability were the main reasons for choosing this model, and it exists even without explicit feature engineering (Rodriguez-Galiano et al., 2015). Additionally, the model was encountered in many different papers for forecasting. Moreover, the RFR seems also to be able to outperform regression trees and support vector machine (Rodriguez-Galiano et al., 2015)

The next model chosen was the **Ridge Regression** model. The traditional-statistical regression is the process of a function that has the least deviation between predicted and actual values for the data (Basak et al., 2007). Ridge Regression is used as a model that is a traditional linear, with the ridge regularization to prevent overfitting. It was chosen for its simplicity and ability to generalize well in unseen data. Moreover, Ridge Regression was chosen for a linear model comparison with the rest of the models.

Moving on, the **Gradient Boosting Machine (GBM)** was chosen. This model is known for iteratively optimizing its performance by correcting the loss function from the previous iteration, making it a very effective modelling method (Natekin & Knoll, 2013). GBM was chosen for its wide range of advantages. Primarily, it is capable of handling complex relationships, needed in sports analysis, and additionally it is capable of handling features that are not as significant as others (Natekin & Knoll, 2013). Apart from these advantages, the GBM was chosen due to its effectiveness. It is an ensemble method; thus, it is robust and has high accuracy in the performed tasks (Konstantinov & Utkin, 2021).

Lastly, **Support Vector Regression (SVR)** was chosen. One of the main characteristics of Support Vector Regression (SVR) is that instead of minimizing the error between observed and actual values, SVR attempts to minimize the generalized error. These machines are ideal for small datasets, thus chosen for this problem. The SVR is meant to achieve good, generalized performance (Basak et al., 2007). Additionally, the Radial Basis Function (RBF) kernel is used. This kernel was chosen for leveraging and capturing nonlinear relationships. The RBF maps the inputs into higher dimensions and thus is able to capture those complex relationships (Kuo et al., 2014).

3.4.2 Evaluation Metrics

The evaluation of the models used in this dissertation will be made with three evaluation metrics. The R^2 metrics the Mean Absolute Error and the Mean Squared Error. Each metric was chosen for a different reason all three providing a robust comparison and evaluation of the performance models.

R^2, as mentioned before is the coefficient of determination. This metric measures how well a statistical model predicts an outcome and how well it fits the data, in other word the R^2 measures the goodness of fit. R^2 will be used to evaluate whether the extracted knowledge from the models is useful and

meaningful. The best possible score for $R^2$is 1.0, when the model perfectly predicts the outcome and in contrast the R^2 is valued at 0 when the model does not predict the outcome. As it is understood the coefficient of determination normally ranges between 0 and 1.

Now, moving forward with the next two evaluation metric the Mean Absolute Error (MAE) and the Mean Squared Error (MSE). As it was explained previously both of these metrics measure the errors with MSE giving more value to the larger error, which are generally unwanted in any prediction task. However, it is sensitive to outliers and thus the MAE is used to balance this drawback. The latter was additionally chosen for its ease of interpretation and of course the straightforward depiction of the error.

3.5 Software tools and libraries

The following software's and tools were utilized to meet the ends need of this dissertation, for the modelling and the preprocessing needed.

- Python

Python was selected as the primary programming language for this study. The choice of python was due to its versatility and its extensive system of libraries, tools and resources for data mining and machine learning. Additionally, it is valued for it easy use, and it is also the writers programming language preference.

- Jupyter Notebooks

Jupyter notebooks, was the author's choice of the integrated development environment IDE. This IDE was used for implementation, testing and documenting the code. Jupyter notebooks provided a user-friendly interface that integrates code, easily outputs visualization. Moreover, it provides textual explanations in a single document. It was deemed particularly useful for data exploration- visualization and model evaluation.

Concerning the libraries that was used:

1. Pandas

This library was used for the purpose of data manipulation and data analysis. Is makes handling and structuring e data, with tables, rows, and columns and easy task. It was also used for data cleaning, filtering, and merging.

2. Numpy

This library was used when numerical computations were needed. It provided effective operations on large data. Also, it includes mathematical functions such as statistics and linear algebra. This library was essential in working of implementing the models for quick computing and feature scaling and numerical transformation.

3. Scikit-Learn

The purpose of this library is for building, training, and evaluation predictive models. It incorporates a wide range of machine learning algorithms, including those used in this dissertation, for example the Random Forest Regressor, the Ridge Regression, the SVR and the Gradient Boosting Machine. It was additionally used for preprocessing, scaling, and encoding, model selection and performance evaluation, with ready functions.

4. Matplotlib

The purpose of this library is to provide comprehensive, detailed and customizable plots. These plots were used in this dissertation to effectively visualize the results for the readers and the author. Additionally, it provides control over the plot components.

5. Seaborn

This library's purpose was for a quick statistical visualization for pandas data frames.

In summary the combination of the above tools and libraries provided a strong framework for data analysis and machine learning. All the approached and choice were made in order for this dissertation to be reproducible and scalable.

3.6 Data Workflow

In this section the work process of the dissertation will be explained step by step. Frist the process of successfully answering the Research Questions and model a predictive algorithm begins with domain understanding. Understanding the key factors and dynamics influencing the problem area and Formula 1 race outcomes—are firstly identified. This step informs the subsequent stages by clarifying objectives and highlighting relevant variables.

Following this step, the data collection steps begin. This step involves gathering reliable data with good quality from verified sources, here this step was made from sources like the Ergast API and other supplementary websites.

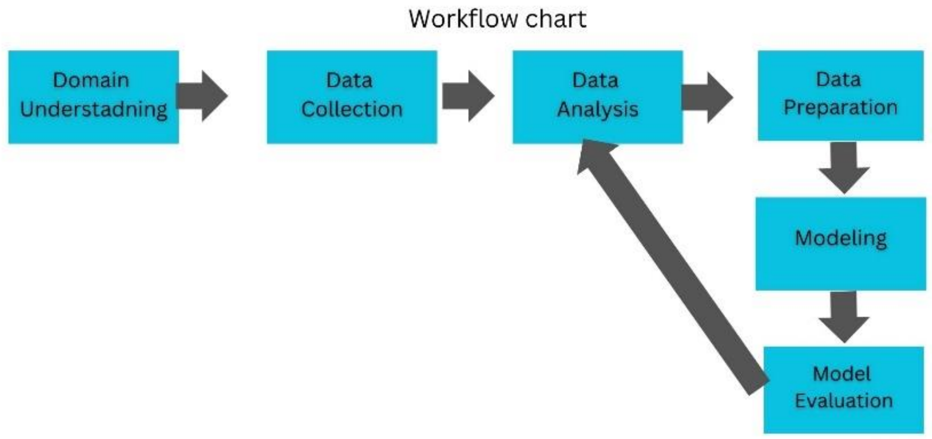
Once the data is collected, data analysis, the third step begins. This step is performed to explore existing patterns, or relationships, and anomalies, providing critical insights that guide the next steps. Then comes the fourth step, the data preparation. This step is mentioned in this dissertation as Data Prepossessing. In this phase the work focuses on cleaning and transforming the dataset, addressing missing values, encoding categorical variables, and selecting the most predictive features to ensure the data is ready for modeling.

After the preprocessing step, the process moves to modeling and evaluation. At these last steps machine learning algorithms are applied, and metrics such as Mean Absolute Error and R-squared are used to assess the algorithms' performance.

Finally, the process often loops back to data analysis to refine the model further, iterating as needed to optimize accuracy and robustness. This iterative approach ensures a comprehensive and adaptive strategy for achieving predictive success.

Fig 3:

Data Workflow figure



3.6.1 Domain Understanding

Formula 1 (F1) is a worldwide loved sport. It stands out in motorsports due to its unique structure. In these sports the teams independently design and develop their cars, following the FIA regulation. This unique feature of sport allows for significant variation in performance among teams. For this reason, F1 is functioning both as an individual and as a team sport (Judde et al., 2013). Since it is individual and team sport, the competition consists of two championships: the Drivers' World Championship (WDC) for the drivers and the Constructors' World Championship (WCC) for the teams. These two championships corresponding to the dual nature of the sport.

Since the start of the first F1 season in 1950, the championship has evolved into one of the most prestigious and widely followed global sports. The sport attracts millions of fans each season and thousands at each circuit at each race. F1 races typically last from Friday to Sunday and thus called *race weekends*. These weekends begin with practice sessions on Friday, followed by qualifying on Saturday, and finish with the main race on Sunday. The qualifying event is divided into three knockout rounds, with drivers battling for the best starting positions. The fastest of all gains the “pole position" the first position on the start grid of the race (Bell et al., 2016).

The F1 season spans from March to December, with two breaks, the summer and the winter break. Each season features a series of Grand Prix across the globe, ‘chasing’ the ideal weather conditions for racing.

Teams and Historical Context

Over 100 teams have participated in the F1 championship since 1950. However, only a select few constructors’ teams consistently achieved championship success in F1 history. These successful teams include Ferrari, Mercedes, Lotus, McLaren, Renault, Williams, and Red Bull.

Car Evolution and Regulations

According to the FIA's 2022 technical regulations, an F1 car is an open-wheel, single-seater vehicle designed exclusively for high-speed circuit racing. The regulations have changed multiple times over the years with. The regulations have changed multiple times over the years with a major shift in 2014. This is the year of this dissertation analysis too. In 2014 F1 transitioned from V8 to V6-hybrid engines, marking the beginning of a new era(Bell et al., 2016). Another regulatory update is set for the 2025 season. From 2014 to 2021 big changes were made with the exception of the 2020 season, disrupted by the COVID-19 pandemic, saw a reduction in the number of races, with only 17 Grand Prix events held instead of the typical 21.

Circuits

The circuits used in the F1 events, referred to as Grand Prix (GP), must be officially approved by the FIA and are specifically designed for F1 racing. These circuits usually start with a straight, with few exception and feature various corners and chicanes. The corners and chicanes are aimed in reducing speed and enhancing driver safety. Circuit lengths vary, and each race must cover a minimum of 305 km, with a typical race duration of around two hours.

F1 circuits also differ in type; some are traditional, mapped as permanent race circuits. Meanwhile there are some circuits, like Monaco and Baku, where the circuits are street circuits. The street circuits mean that the track takes place on the city roads. The direction of circuits is mainly clockwise, with a few exceptions. Additionally, the races take place at night, with few exceptions. The night racing was introduced in 2008, in Singapore. Then concept was adopted by other venues the Abu Dhabi and Bahrain.

Point System

The current points system was implemented in 2010. It gets 25 points for 1st place, 18 for 2nd, and 15 for 3rd. The rest places, up to the 10th get points with a decreasing scale down to 1 point.

Pit Stops

Pit stops play crucial in Formula 1. The pit stops are important for the race strategy each team will choose. In general, in all circuit motorsport, teams aim to optimize pit stop timing to secure the best competitive advantage. Pit stop strategy may involve changing tires, refueling, or making car adjustments (Heilmeier, Thomaser, et al., 2020), however in F1 tire change and specific car fixes are only allowed. Mathematical optimization models can be used to calculate the optimal tire strategy when only tire degradation is considered (Heilmeier, Thomaser, et al., 2020). However, pit stops can also be influenced by external factors like weather conditions, and the competition from the other teams and drivers. An efficient pit stop strategy can significantly impact a team's final position.

4 Experimentation

In this chapter of the dissertation, the experimentation within the modelling phase is presented. For the models that were decided to be used and finally compared in the Results chapter, their implementation and tuning is presented here.

4.1 Modelling with Basic Dataset

The models were first evaluated using a dataset excluding weather-related features, that is the basic dataset. As mentioned on too many times, feature selection is a very important step in modelling. For every model, there was an implementation without feature selection, and then the post-feature selection. For all models the same features were used. The features used in modelling are the following:

['points', 'pos_after_race', 'con_pos_after_race',
'grid', 'points_after_race', 'quali_pos', 'qualies_best_secs', 'number_y',
'circuit_type', 'con_wins_after_race']

Since the weather data is excluded the main features that are used are features representing the drivers performance and the constructors performance. These features were extracted with the process of the Correlation Analysis and Mutual Information. Additionally, they are in harmony with the feature suggested by the existing literature that highlighted the performance of the athletes and the cars-teams performance as the most important factors when predicting race outcomes.

4.1.1 Random Forest Regressor

The first model implemented was the Random Forest Regressor. During the implementation of the model, many hyperparameters were tested. The testing was done, apart from the need to achieve high accuracy for the avoidance of overfitting.

The hyperparameters were chosen using the grid search method, the Cross-Validation grid search (GridSearchCV). The parameters set in the grid for the method to search in, were multiple and stated in the following. For the number of estimators, the model searched between 100, 200, 300, 500 and 1000. For the max depth the search was between 2, 5, 8, 10, 20, 30 and 40. Finally the minimum samples split was search among 2, 5, 10, 15, and 20. The optimal hyperparameters for the models pre and post feature selection is depicted in table 6.

Table 6:

Hyperparameters of Random Forest Regressor (RFR)

Model	**max_depth**	**Min_samples_split**	**n_estimators**
RFR pre feature selection	10	2	300
RFR post feature selection	5	10	300

4.1.2 Ridge Regression

During the implementation of this linear model, the hyperparameters were chosen using the Cross-Validation grid search (GridSearchCV). This method is a systematic method for tuning that search through specified set of hyperparameters combinations. The alpha parameter was set, after searching through 0.1, 1, 10 and 100. The optimal alpha is set to 1.

The final hyperparameters chosen for the Ridge regression model is presented in Table 7. The alpha parameter the only parameter that was tune in this model is the parameter that gives the regularization strength of the model and balances between the fit and the complexity of the model.

Table 7:

Hyperparameters of Ridge Regression (Ridge)

Model	alpha
Ridge pre feature selection	1
Ridge post feature selection	1

4.1.3 Gradient Boosting

Gradient Boosting was the next model of choice. It is an ensemble model, and this played a key role in picking it, since the literature work reviewed highlighted their efficiency and accuracy. It is expected that as an ensemble model it will generally wield better results in general.

Again, the hyperparameters were set using the Cross-Validation grid search (GridSearchCV). The algorithm searched through multiple estimators. For the parameter of n_estimators, the number of estimators the method searched through 100, 200 and 300. For the learning rate the method searched among the values of 0.01, 0.1 and 0.2. Finally, the maximum depth was set after searching among the values of 3, 4, and 5. The optimal hyperparameters were set at. In Table 8, the hyperparameters chosen for the model pre- and post-feature selection are showcased.

Table 8:

Hyperparameters of Gradient Boosting Machine

Model	Learning Rate	Max_depth	n_estimators
GBM pre feature selection	0.1	3	100
GBM post feature selection	0.01	4	300

4.1.4 Support Vector Regressor

For the Support Vector Regression (SVR) model, again the optimal hyperparameters were chosen using the Cross-Validation grid search (GridSearchCV). GridSearchCV searched through multiple estimators to conclude the optimal choices. The hyperparameters were tuned automatically and were set as shown in the Table 9 for the pre and post feature selection models.

The search was for the values of C between 0.1, 1, 10 and 100. For the epsilon values the search was between 0.01, 0.1 and 0.2 and the kernel was set as ‘rbf’ since the data were complex hiding non-linear relationships.

Table 9:

Hyperparameters of Support Vector Regressor

Model	C	epsilon	kernel
SVR pre feature selection	10	0.2	‘rbf’
SVR post feature selection	100	0.3	‘rbf’

The parameter C, is the one that controls the complexity of the model, improving it generalization. Higher values of C give emphasis on minimizing the error and potentially overfitting, while low values of C results in simpler models that aids generalization. The parameter epsilon defines the margin tolerance around the predicted values. A smaller epsilon makes the model more sensitive to small deviation, while a larger epsilon makes the model focus on larger patterns.

4.2 Enhanced modelling with weather data

On the next part of the dissertation the model trained using the same dataset, but it was enhanced using the weather information. The enhanced dataset including the weather information were concerning seasons the 2018-2023. For the season 2014 to 2018 the weather information was not publicly available. Even though the races conducted are fewer, for each race three timestamps were added with the weather at three points of the race, resulting in bigger dataset.

According to the papers reviewed already in previous chapters, adding weather data improves the accuracy of the models; thus, an overall improvement is expected. But there is not a comparative study between same models, using the same data, added the weather information. This improvement is the result of the inclusion of weather data, that provides an additional dimension to the analysis. This fact allows for a deeper exploration of how environmental factors influence Formula 1 race outcomes. This section details the implementation and evaluation of various predictive models using the dataset with integrated weather data.

The features chosen for the models using the weather data are set as follows:

['points', 'pos_after_race', 'con_pos_after_race', 'con_points_after_race', 'grid', 'points_after_race', 'quali_pos', 'qualies_best_secs', 'number_y', 'circuit_type', 'Pressure', 'TempCluster', and 'Humidity', ‘WindDirection’]

These features were the result of the feature engineering process, using mutual information and correlation analysis. In general, the weather feature was not deemed as important as expected, as analysed in the results chapter of the dissertation. Additionally, the literature review viewed the Air Direction as an important environmental factor affecting the cars aerodynamics. Nonetheless it was not revealed as important in the feature extraction process, but was added either way, as an important factor for cars.

4.2.1 Random Forest Regressor

The first model implemented was the Random Forest Regressor. During the implementation of the model, many hyperparameters were tested. The testing was done, apart from the need to achieve high accuracy for the avoidance of overfitting.

The hyperparameters were chosen using the grid search method, the Cross-Validation grid search (GridSearchCV). The parameters set in grid for the method to search in, were multiple. For the number of estimators, the model searched between 100, 200, 300, 500 and 1000. For the max depth the search was between 2, 5, 8, 10, 20, 30 and 40. Finally the minimum samples split was search among 2, 5, 10, 15, and 20. The optimal hyperparameters for the models pre and post feature selection is depicted in table 10.

Table 10:

Hyperparameters of enhanced Random Forest Regressor with enhanced dataset

Model	max_depth	Min_samples_split	n_estimators
RFR pre feature selection	30	2	200
RFR post feature selection	30	2	200

4.2.2 Ridge Regression

During the implementation of this linear model, the hyperparameters were chosen using the Cross-Validation grid search (GridSearchCV). This method is a systematic method for tuning that search through specified set of hyperparameters combinations. The alpha parameter was set, after searching through 0.1, 1, 10 and 100. The optimal alpha is set to 1. The hyperparameter tuned are depicted for the models pre and post feature selection in Table 11.

Table 11:

Hyperparameters of Ridge Regression (Ridge) with enhanced dataset

Model	alpha
Ridge pre feature selection	1
Ridge post feature selection	1

4.2.3 Gradient Boosting

Gradient Boosting was the next model of choice. It is an ensemble model, and this played a key role in picking it, since the literature work reviewed highlighted their efficiency and accuracy. It is expected that as an ensemble model it will generally wield better results in general.

Again, the hyperparameters were set using the Cross-Validation grid search (GridSearchCV). The method was able to search through multiple estimators. For the parameter of n_estimators, the number of estimators the method searched through 100, 200 and 300. For the learning rate the method searched among the values of 0.01, 0.1 and 0.2. Finally, the maximum depth was set after searching among the values of 3, 4, and 5. The optimal hyperparameters were set at. In Table 12. The hyperparameters chosen for the model pre and post feature selection are showcased.

Table 12:

Hyperparameters of Gradient Boosting Machine with enhanced dataset

Model	Learning Rate	Max_depth	n_estimators
GBM pre feature selection	0.2	5	300
GBM post feature selection	0.2	5	300

4.2.4 Support Vector Regressor

For the Support Vector Regression (SVR) model, again the optimal hyperparameters were chosen using the Cross-Validation grid search (GridSearchCV). The algorithm of GridSearchCV searched through multiple estimators, in order to conclude the optimal choices. The hyperparameters were tuned automatically and were set as shown in the Table 13, for the pre and post feature selection models.

Table 13:

Hyperparameters of SVR

Model	C	epsilon	kernel
SVR pre feature selection	100	0.01	‘rbf’
SVR post feature selection	100	0.2	‘rbf’

The parameter C, is the one that controls the complexity of the model, improving it generalization. Higher values of C give emphasis on minimizing the error and potentially overfitting, while low values of C results in simpler models that aids generalization. The parameter epsilon defines the margin tolerance around the predicted values. A smaller epsilon makes the model more sensitive to small deviation, while a larger epsilon makes the model focus on larger patterns.

5 Results

In this chapter of the dissertation, the results of the implementation of the models will be presented. As mentioned already, first the modelling with the basic dataset will be presented, the dataset without any weather information and then the modelling with the enhanced dataset. The enhanced dataset includes the weather information for each race. This way, the findings at the end can be compared to try and measure the effect of the weather on forecasting the ranking of positions on F1 races.

Each section mentioned above, consists of two basic sections. The feature engineering part and the models' performance part. On the feature engineering part, the results of the correlation analysis and the mutual information will be presented, concluding to which feature were selected and why. On the next part, the models' performance, first the performances of the models without feature engineering will be presented, followed by the models with their important features selected. The two processes are compared. In the end the best models from the basic and the enhanced dataset are compared.

5.1 Modelling with Basic Dataset

First the results of the models trained on the basic dataset will be presented. Concerning the feature selection part, first a correlation analysis took place to identify the most import features that linearly correlate to the target value, the 'position' and the mutual information. The results, especially for correlation analysis are advised to be studied in colour by the author.

After the feature engineering part, the analysis of the results of modelling on this data pre and post features selection will be presented. Finally, it is concluded on which modelling methods wielded better results based on the selected evaluation metrics.

5.1.1 Feature Engineering

As mentioned in earlier parts of the dissertation, feature selection as a part of data pre-processing plays an especially key role. The number of features before feature selection is 23 and the target value is one, the value ‘position’. The need for feature selection stems for the possibility of some of these features creating redundancies, additionally they seem slightly a lot, compared to the dataset size. Thus, in this research feature selection could not be left out. Since this result analysis contains no weather information, from the literature review conducted before, it is expected that the main feature that affects the final ranking position is the metrics on the drivers and the team performance.

Correlation analysis

Correlation analysis as statistical technique measures the strength and the direction of the relationships between variables. It is primarily designed to identify linear relationships. The following graph depicts these relationships found by the analysis between the target value, position, and the rest of the features. Figure 4, presented below, depicts the correlation analysis conducted without incorporating weather data, allowing a focused assessment of performance-related variables. The bolder the colour the higher the relation. The high positive correlation corresponds to more red colours and the higher negative correlation to blue. The non correlated feature is depicted with a colour close to grey.

In Figure 4, the most important features are as those with the higher correlation, negative or positive. As mentioned before the more intense the colours, the higher the correlation. Additionally, in each square the value of the Correlation analysis between the two variables is displayed. A deep red colour with correlation set to 1 indicates the perfect relationships, typically when the variable is compared with itself. On the Table 14 the top scoring features and their values are depicted.

Fig 4:

Correlation Analysis without weather data

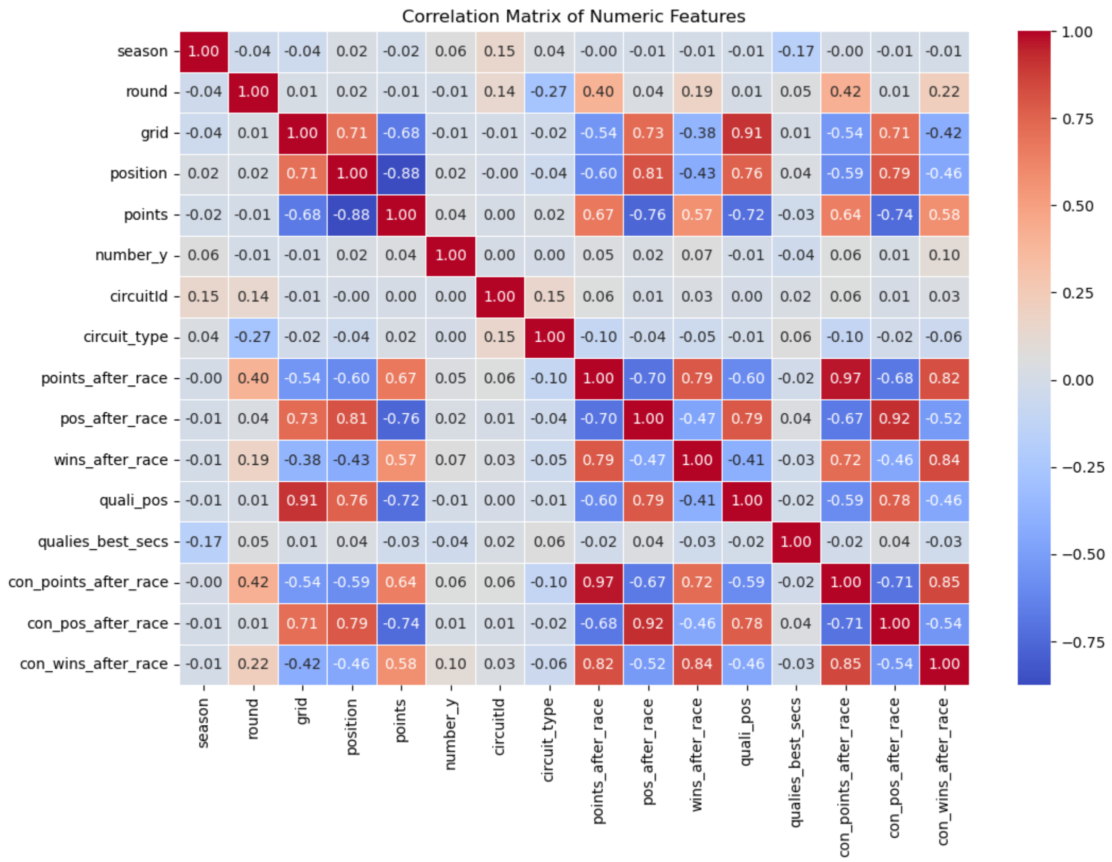


Table 14*:*

Correlation Analysis most important features

Feature	Value	Description
Points	-0.876	The cumulative points gathered after the race for the driver
Position after race	0.811	The final standing position of the driver
Constructor's Position after race	0.788	The final standing position of the constructors
Qualifying Position	0.765	The position achieved in qualifications
Grid Position	0.706	The starting position
Points after the race	-0.604	The pointes accumulated by the driver after the race
Wins after the race	-0.426	The total wins for the driver after the race
Constructor wins after the race	-0.457	The total wins for the teams after the race

As expected, the most important features are the ones that measures the performance of the athlete and the car, depicted by the final position after races, and the points gathered, as well as the qualification performance of the driver. The qualification performance was mentioned in the literature as an important factor and this dissertation's findings aligns with it.

Moving on with the results of the correlation analysis, there are some features with low positive or negative relationships worth mentioning. A weak correlation with the final position is noted with the features **Best Qualifying Lap Time,** with a value of **0.036** and additionally, with the **Driver/Car Number** valued at **0.020.** However, the driver should influence the result, since the performance is based on the skill of the driver. One possible reason for the Driver number to not have high correlation is because this specific analysis struggles depicting complexities.

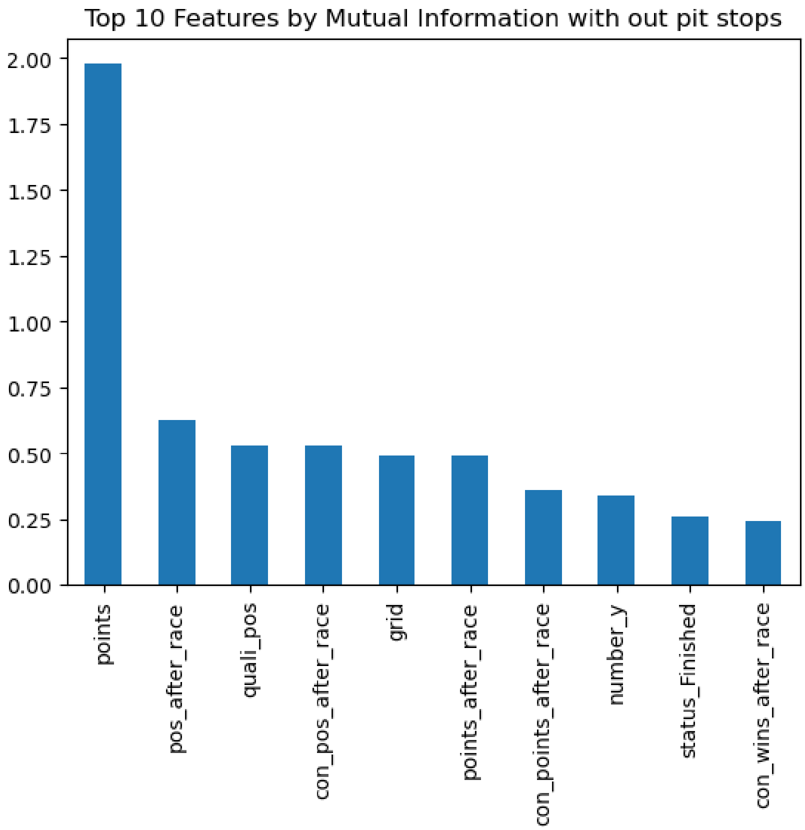
Mutual Information

Moving forward with the analysis, the mutual information (MI) analysis took place to identify any non-linear correlation missed in the correlation analysis, since this analysis manages to capture both both linear and non-linear dependencies, providing a more comprehensive understanding of feature relevance. This section explores the relationships between features and the final race position using mutual information analysis.

The different scores each feature has in the MI analysis are presented in Figure 5 below. The plot highlights the MI scores and offers valuable and useful insights into their significance. The predictive importance of the top 10 most relevant features and their precise MI score and their description are presented in Table 15. Using Figure 5 and Table 15, the reader can fully understand the features that contribute the most to forecasting race outcomes. The MI analysis added to the Correlation highlighted the non-linear relationships between variables and the final race position, complementing the findings from the previous correlation analysis.

Fig 5:

Mutual Information analysis without weather data



The features revealed with mutual information are not very different from the ones revealed in correlation analysis. The most important feature that affect the final position of the drivers are as follows. The basic difference with the correlation analysis, is the extent to which the number_y , thus the driver, affect the results. Additionally, the status feature was added to the most important features. Both, of the mentioned feature depict more complex relationship, that probably why the relationship wasn't captured with Correlation Analysis.

Table 15*:*

Mutual Information top 10 features

Feature	**Score**	**Description**
Points	1.979	cumulative driver performance and overall competitiveness
Position after Race	0.626	The standing position of the driver
Qualifying Position	0.532	The final qualification position
Constructor's Position After Race	0.531	The final position of the constructors in the standings

Grid Position	0.492	The starting position of the driver
Point After Race	0.363	The accumulated points of the driver after the race
Constructor's Point After Race	0.363	The accumulated points of the teams after the race
Driver/Car Number	0.340	The number of the driver
Statues (finished)	0.261	The status of the driver finishing the race
Constructor Wins after Race	0.242	The cumulative wins of the teams after the race

By combining these two analyses both linear and nonlinear dependencies were revealed. First and more most, the points are the most informative feature, as highlighted by their high MI score and correlation value. The points accumulated in the season is a strong indicator of the performance of each driver. As showed in the literature review the drivers and the team performance as very influential at the final positions.

In addition to points, both drivers and the constructors, there other feature showcasing the skill of the drivers and the performance of the team were deemed and important and thus selected in the models. More specifically, a driver's and constructors' position after a race is highly predictive of their final or future positions. Moreover, the win counts are deemed as important feature. In contrast with the literature review showcasing that the teams affect the result more than the driver in this analysis the team's performance were calculated as important, but it has a less direct impact than individual driver performance.

However, the race performance isn't the only indicator. As anticipated from the literature review done, the performance in the qualifications influence the final positioning. Qualifying position has a strong influence on the final race result and additionally the grid, the starting position won at the qualification plays an important role. Usually, a driver starting at the front enjoys a competitive advantage.

5.1.2 Basic Models' Performance

In this part of the dissertation the performance of the modeling methods selected previously will be presented and compared. For each modeling method, first the results of the model's pre-feature selection process will be presented and then the results post feature selection, comparing the performance and the generalization of the models tested in this dissertation.

Pre-Feature Selection process

The results of the performance of each model before selecting the most relevant features are shown in Table 16 Random Forest Regressor (RFR) appears as the highest performing model. This model with R^2-score of 0.9424, was able to explain approximately 94.24% of the variance in the race position predictions. The high score of the R^2 indicates high accuracy of the model. The other two metrics used in this analysis showed also exceptionally result. The Mean Squared Error (MSE) were valued at 1.4847 and Mean Absolute Error (MAE) at 0.6499, both being low. These low errors suggest that the model produces predictions with minimal errors.

Gradient Boosting (GBM) also performed exceptionally well. Having a high R^2 score of 0.9416, and thus explain approximately 94.1% of the variance, it lies just slightly below the RFR model. The MSE of the GBM was calculated at 1.5057 and its MAE at 0.6958. Both metrics are only marginally higher than those of RFR. Having metric so close indicates a comparable level of accuracy and robustness between the two said models.

On the last two places came the Support Vector Regressor (SVR) and the Ridge Regression model (Ridge). Even though being ranked low the SVR model demonstrated moderate performance, with an R^2 score of 0.9052, suggesting that it explains 90.52% of the variance in the predictions. In contrast to RFR and GBM it had a higher MSE (2.4421) and MAE (1.0200), highlighting larger prediction errors.

Finally, out of all the models came the Ridge model, which exhibited the lowest performance. With a value for the R^2 metric at 0.8752, the models explain 87.5% of the variance. The other two error metrics, MSE and MSE were valued at 3.2158 and 1.2862 respectively. These metrics show that Ridge’s accuracy and error rates are notably lower than the other models. However, it still provides reasonable predictions.

Table 16:

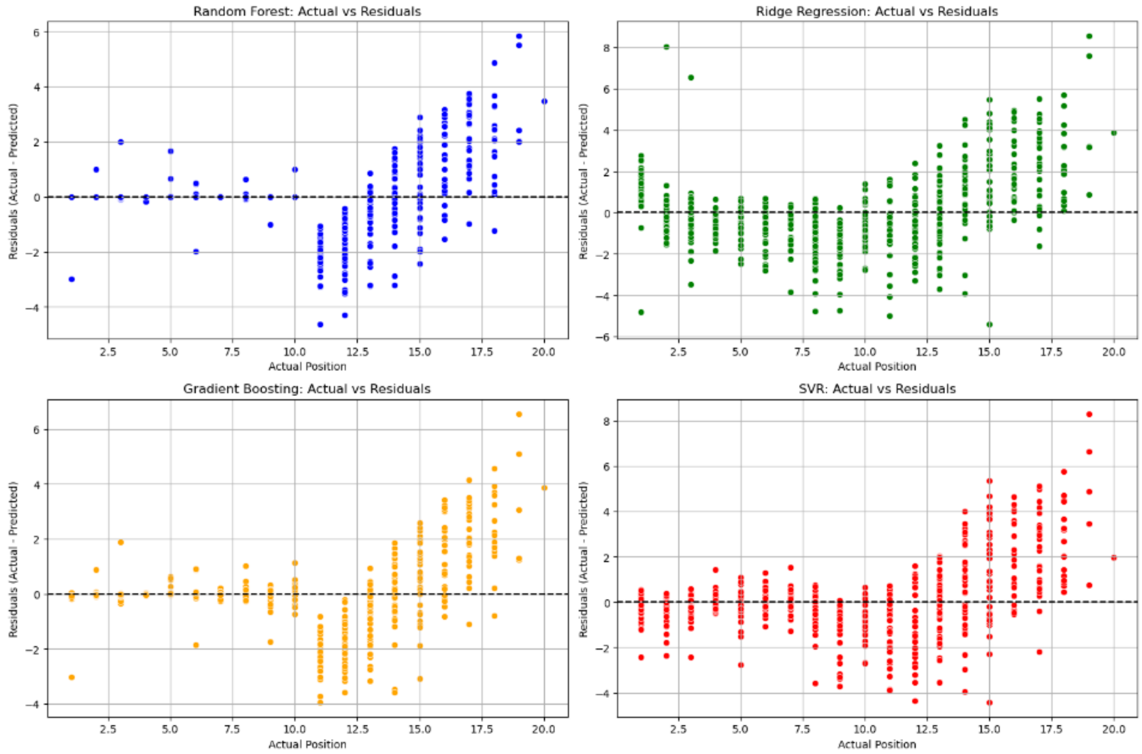
Comparison of evaluation metrics for dataset without weather data pre feature selection

Model	**Mean Squared Error**	**Mean Absolute Error**	**R-squared**
Random Forest	1.484704	0.649863	0.942389
Ridge	3.215776	1.286213	0.875218
Gradient Boosting	1.505700	0.695839	0.941574
SVR	2.442051	1.019976	0.905241

The three chosen evaluation metrics are not the only measure to select the best performing model. Following up next, there is a graph comparing the actual values of each model and the residuals. Figure 6 shows the residuals the error for each prediction. It is now visually clear too that the two best performing model are the RFR and the GBM. The residuals at both cases grow more scatter after the tenth place. For the ridge regression and the SVR, the residuals plot highlights their bad performances and their bad fit to the data.

Fig 6:

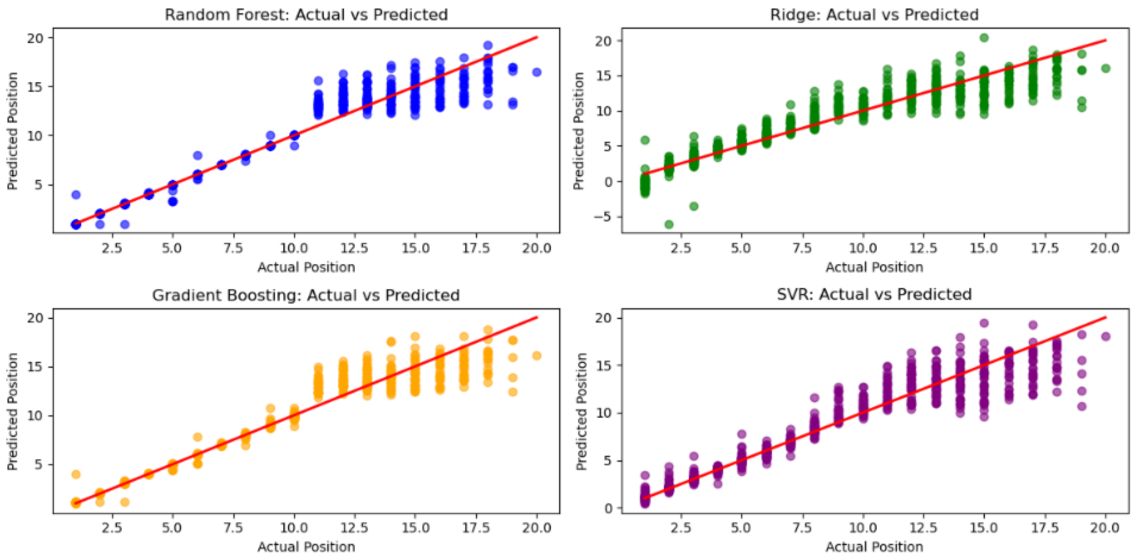
Comparison of actual values and residuals of data without weather pre feature selection.



Adding to the visualization is Figure 7 plotting the actual versus the predicted values. Even though it is clear that the RFR is the best performing model, followed closely from GBM, there is a graph comparing the actual versus the predicted values for each value at figure 18. The red line is the actual values, and the predicted values scatter around in colourful dots. Again, it is visually inspectable that the two best performing models are predicting way worse after the 10^{th} place.

Fig 7:

Comparison of actual values and predicted values of data without weather pre feature selection



In conclusion, the Random Forest Regressor emerged as the most accurate and reliable model, followed closely by Gradient Boosting. Both models significantly outperformed SVR and Ridge Regression, demonstrating their robustness and suitability for this analysis.

Post Feature Selection process

Moving forward with the analysis, the results for each of the models, after feature selection process are presented in Table 17 with their relevant metrics. The results are not as different as one might expect. The highest performing model in this case is Gradient Boosting (GBM), with a high R^2 score of 0.9278. The high R^2 score indicated that the model explains approximately 92.78% of the variance in the final positioning while making predictions. The two errors, the Mean Squared Error (MSE), and the Mean Absolute Error (MAE) are valued at 1.8606 and 0.8335. These low values suggest minimal errors in the predictions of the GBM, thus indicating a high level of accuracy.

The Random Forest (RFR) model also performed relatively competitively. The RFR model achieved an also impressive R^2 score of 0.9245. This means it explains about 92.45% of the variance in predictions. The difference with GBM came in errors. The MSE of the RFR was calculated at 1.9455 and the MAE at 0.9245. Despite these marginal differences, RFR had a strong predictive power, close to GBM's performance.

In comparison, with the top two performing models, the Support Vector Regressor (SVR) model showed only a moderate performance. Its R^2 score was calculated at 0.8841, while the errors were at 2.9881 for the MSE, and 1.0626 for the MAE. These results indicate larger errors and lower explanatory power compared to the first two models. Meanwhile, Ridge Regression had the lowest performance among the models evaluated. It had an R^2 score of 0.8330, in addition to the highest errors of all four models.

Table 17:

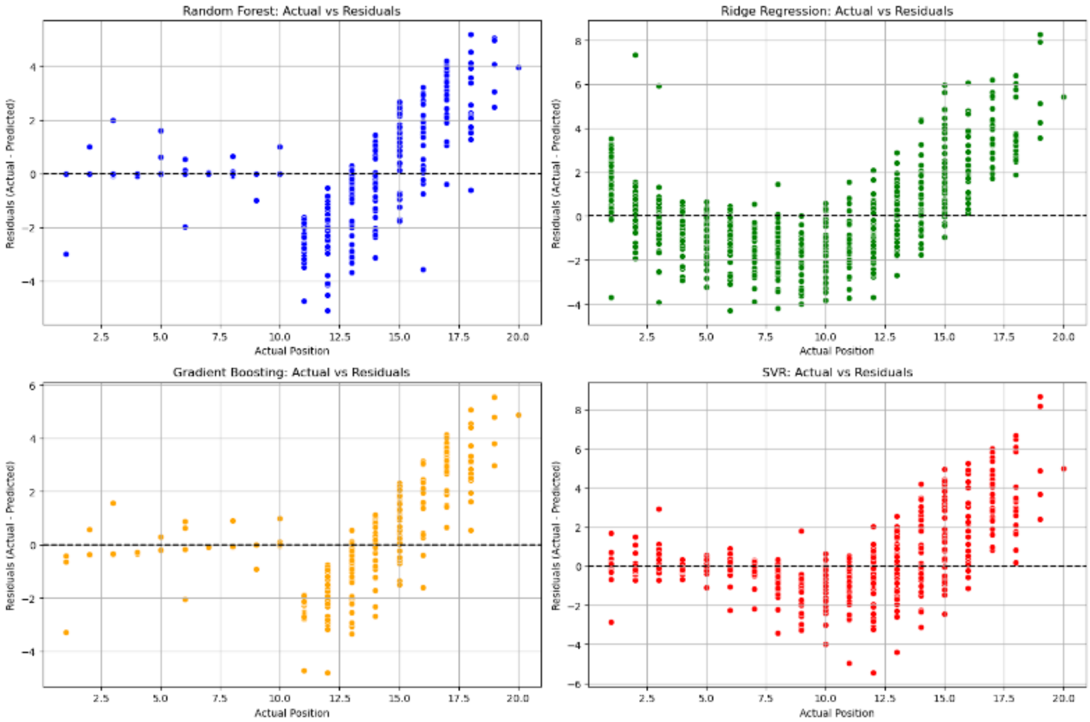
Comparison of evaluation metrics for dataset without weather data post feature analysis

Model	**Mean Squared Error**	**Mean Absolute Error**	**R-squared**
Random Forest	1.945487	0.759734	0.924509
Ridge	4.304462	1.585733	0.832974
Gradient Boosting	1.860585	0.833471	0.927804
SVR	2.988149	1.062571	0.88405

Moving forward with the analysis a visualization of the residuals, the error of each model is depicted in Figure 8. With this plot it is clear to the reader that the two best performing models are again very close and again making noticeable more mistakes after the 10th place.

Fig 8:

Comparison of actual values and residuals of data without weather post feature selection.

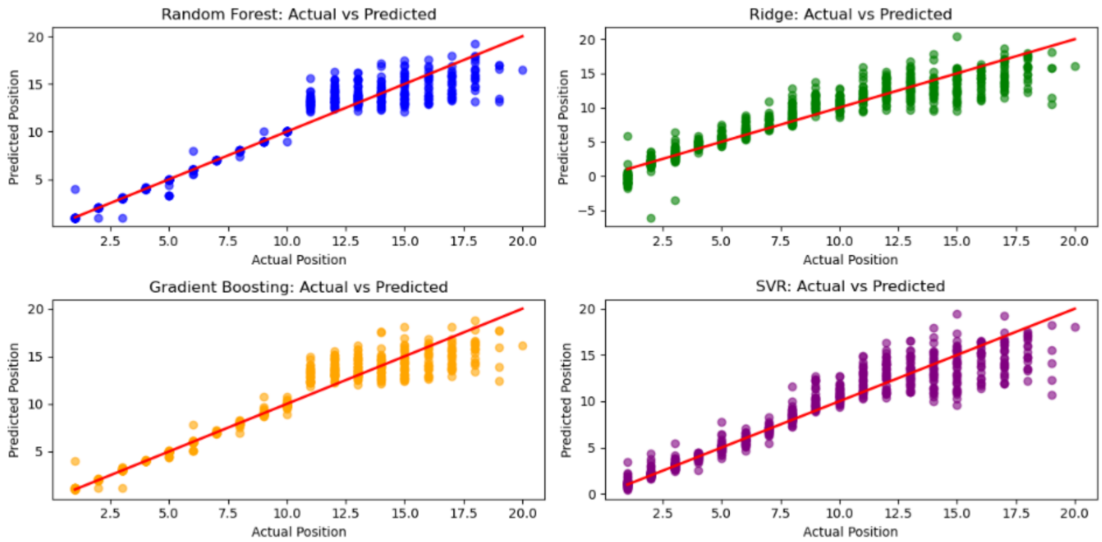


Expanding on this fact, it is clear from Figure 8 the plotting of the residuals, that the RFR model and the GBM's, residuals, cluster tightly around zero for positions 1 to 10, indicating their strong predictive accuracy within this range. However, for positions beyond 10, the residuals become more dispersed, suggesting reduced accuracy for lower-ranked positions.

Moving forward with the results analysis the Figure 9 is presented for the comparison of actual values of each model, presented on the red versus the predicted ones, scattered around the red line in colourful dots. The results are like the pre-features selection analysis.

Fig 9:

Comparison of actual values and predicted values of data without weather post feature selection.



The RF model and GBM produce predicted positions that fall closely with the diagonal red line of actual positions, representing perfect predictions, for positions 1 to 10, signifying high accuracy in this range. However, for positions beyond 10, the predictions deviated significantly from the actual values.

Ridge Regression and SVR's predictions deviate for the actual values from the beginning. It is clear that after the 10th position these models struggle more too, however they struggle for the first 10 more than the two best performing models. These indicates poor alignment for Ridge and SCR between predictions and actual values, suggesting that these two models might not be well-suited for this task.

5.1.3 Comparison

In conclusion, GBM and RFR performed very closely. Both models significantly outperformed SVR and Ridge Regression, emphasizing their robustness in both pre and post feature selection scenarios. The choice between the two seems impossible. To aid this process, the need for statistical comparison of the performance between them arose. The Analysis of Variance (ANOVA) test was conducted using 5-fold cross-validation using both the R-squared (R^2) and Mean Squared Error (MSE) metrics to assess if there is any statistically significant difference.

Concerning the results for the R^2 measure. The results of the ANOVA test for R^2 indicated that there were no significant differences. More specifically, the F-statistic for R^2 was calculated at 0.2469, and the p-value at 0.6327. Both values are above the commonly used threshold of 0.05 for statistical significance.

Similarly, when the ANOVA test was applied to the MSE, the results showed no significant differences between the two models. The F-statistics for MSE was 0.2563, and the p-value was 0.6263, further confirming the lack of statistically significant performance differences between RFR and GBM.

In conclusion, the ANOVA test revealed that the performance of the RFR and GBM models is no different and the choice between the two models, in terms of accuracy, is minimal. The analysts can choose based on needs like computational costs, speed, and interpretability.

Concerning the choice of pre- and post-feature selection, the accuracy of all models was better in the dataset pre feature selection. Concerning RFR and GBM achieved better performance metrics in the pre-feature selection dataset, indicating that they benefited from the full set. Additionally, SVR and the Rdige model, models more sensitive to scaling and the quality of the feature also performed better pre feature selection. The full dataset. enables higher predictive accuracy and smaller errors across all tested models, indicating that the full feature set provides critical information that enhances the models' performance.

5.2 Modelling with Enhanced Dataset

Moving forward with the result analysis, the section concerning modelling on a dataset including weather data is analysed. Concerning the feature selection part, first a correlation analysis took place to identify the most import features that linearly correlate to the target value, the ‘position’ and then the mutual information. The results, especially for correlation analysis are advised to be studied in colour by the author.

After the feature engineering part, the analysis of the results of modelling on this data pre and post features selection will be presented. Finally, it is concluded on which modelling methods wielded better results based on the selected evaluation metrics.

5.2.1 Feature Engineering

Concerning the identification of the most important features, concerning the feature selection part, first the results of the correlation analysis are presented to identify the most import features that linearly correlate to the target value, the ‘position’. It is advice to be studies in colour. Like the analysis done for the dataset without weather, it importance is crucial. Specifically, the dataset containing weather information consists of 38 columns (features). Since this result analysis contains weather information, from the literature review conducted before, it is expected that the main features that affect the final ranking position are the metrics on the drivers and the team performance, as before and additionally, some weather information like Humidity, Air temperature and Wind Direction.

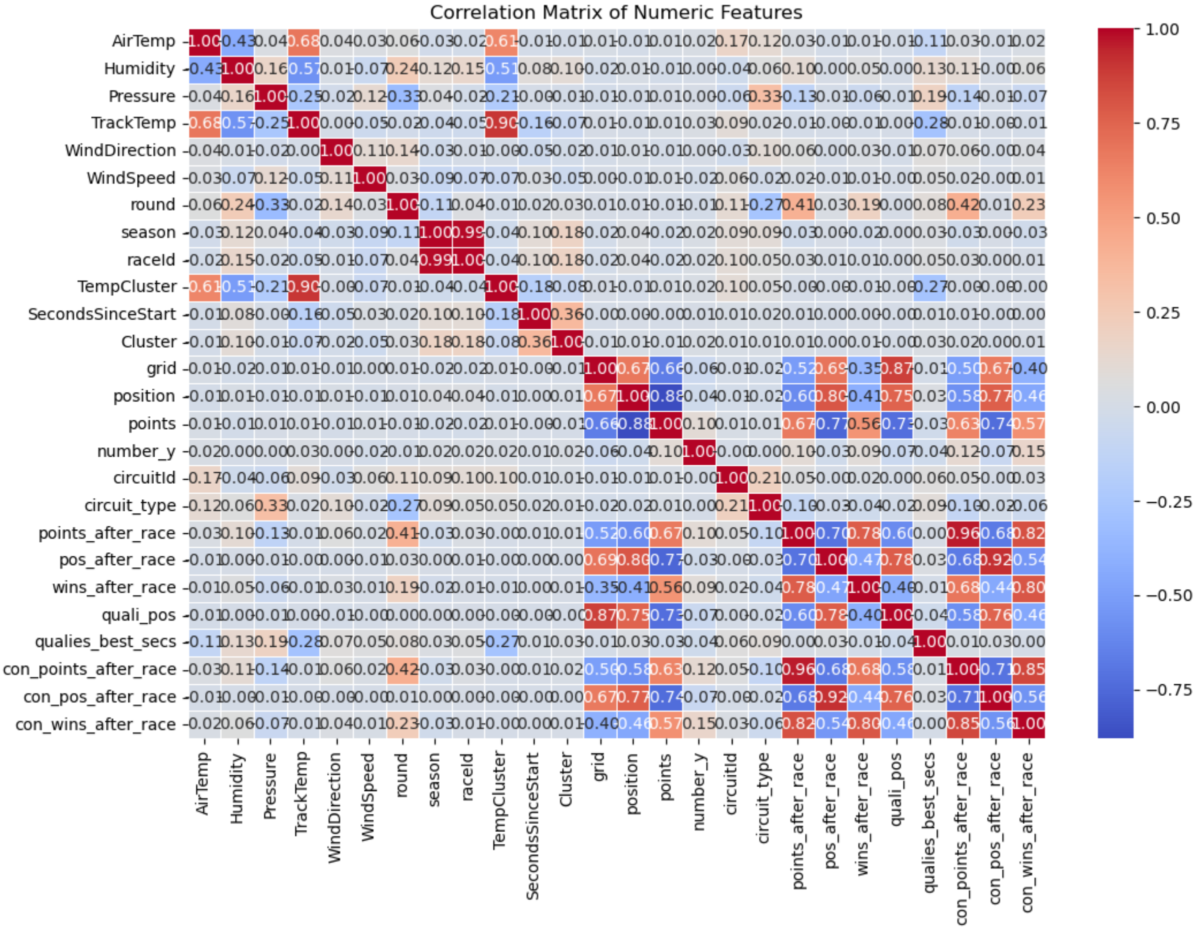
Correlation Analysis

The following graph, Figure 10 depicts the relationship between the target value, position, and the rest of the features, depicts the correlation analysis. The bolder the colour the higher the relation. The high positive correlation corresponds to more red colours and the higher negative correlation to blue. The non correlated feature is depicted with a colour close to grey. In comparison with the same analysis but without weather data, there are some visually easy to spot changes when it comes to the features playing part.

More specifically, Figure 10 presents the correlation analysis with weather data, providing insights into the relationship between various features and the final race position. From a visual inspection the top features correlated with the final position include position after the race, qualifying position, and grid position, the same features encountered in the analysis without the weather data.

Fig 10:

Correlation analysis with weather data.



In Table 18, the important features are presented in ranked order. The value of the Correlation Analysis, indication the degree of the relationship between the two variables is presented clearly.

Table 18:

Correlation Analysis's most important features

Feature	Value	Description
Points	-0.879	The cumulative points gathered after the race for the driver
Position after race	0.798	The final standing position of the driver
Constructor's Position after race	0.768	The final standing position of the constructors

Qualifying Position	0.754	The position achieved in qualifications
Grid Position	0.670	The starting position
Points after the race	-0.595	The pointes accumulated by the driver after the race
Constructors point after the race	-0.576	The points accumulated by the constructors after the race
Constructor wins after the race	-0.460	The total wins for the tams after the race
Wins after the race	-0.412	The total number of wins for each driver

Despite adding the weather data, the most influential features revealed with correlation analysis do not include any weather feature. Again, the driver's and the team's performance are the key feature influencing the final positioning in F1 races.

If the result's list is widened, there are some other features included that were not deemed relevant in the analysis with the basic dataset. More specifically the **raceId** with a value of **0.037200** has a very weak positive correlation suggesting that the specific race does not significantly affect the final positioning. The same is true for features like **season (0.036258, qualies_best_secs (0.034654)** and **Humidity (0.010995).**

Moreover, there are the features with weakly, negatively correlation with position, for example the feature of **Pressure** with a value of **-0.011556**, **TempCluster,** the feature corresponding the cluster the track temperature belongs having a value of **-0.011995.**

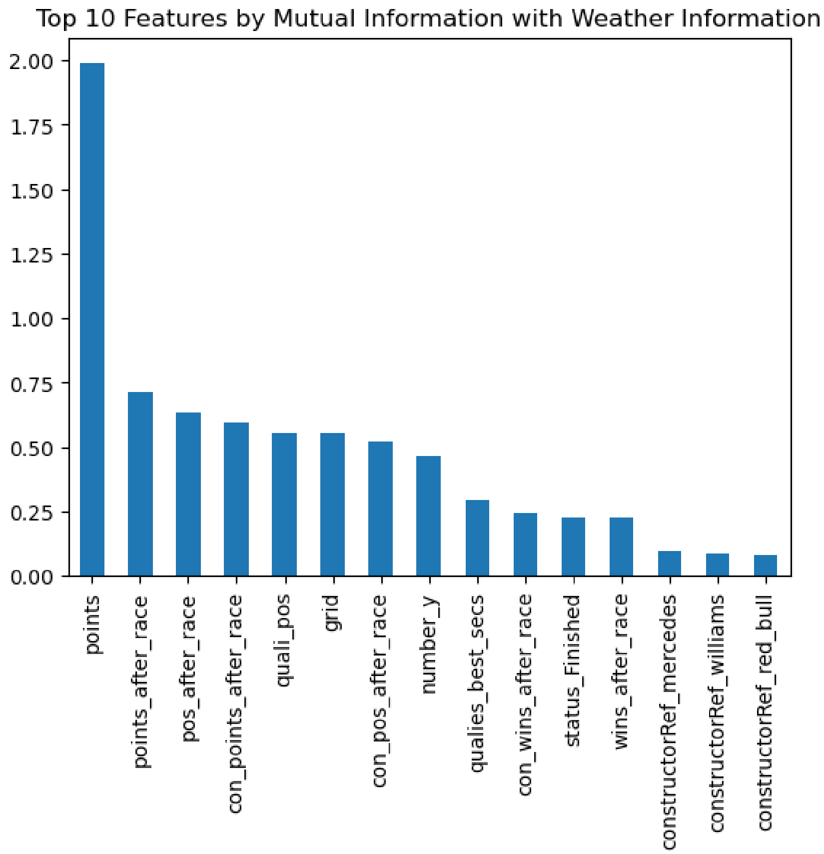
Mutual Information

Moving forward, the mutual information analysis will further delve into identifying non-linear relationships among the features, offering a more comprehensive view of how these variables interact with the final race position. Moving forward with the analysis, the mutual information took place to identify any non-linear correlation

The top 15 Features revealed by Mutual Information with the weather information are graphically depicted in Figure 11. The predictive importance of the top 15 most relevant features and their precise MI score and their description is presented in Table 19, excluding the last 3 features that were specific constructors, and their score was less than 0.01. Instead of top 10, these times the top 15 features were chosen, since the dataset gets more columns and probably needs more features to depict all the intrici relationships. Using the following plot and table the reader can fully understand the features that contribute the most in forecasting race outcomes.

Fig 11:

Mutual Information analysis with weather data



The features revealed with mutual information are not very different from the ones revealed in correlation analysis. The basic difference with the correlation analysis, is the extent to which the qualification position and the grid position affect the results. Additionally, the extend of the influence of the number_y , thus the driver, has greater significance in the MI analysis for the position forecasting. More specifically, compared to the correlation analysis in MI analysis the number_y is the 8th most important feature.

Additionally, the qualies_best_secs feature was added to the most important features. Both, of the mentioned feature depict more complex relationship, that probably why the relationship wasn't captured with Correlation Analysis.

Table 19:

Mutual Information top 10 features and scores

Feature	Score	Description
Points	1.988	cumulative driver performance and overall competitiveness
Points After Race	0.710	The accumulated points of the driver after the race
Position after race	0.635	The final standings position of the driver after the race
Constructor's Points After Race	0.593	The accumulated points of the teams after the race
Qualification Position	0.556	The position achieved after the qualification
Grid Position	0.552	The starting position of the driver
Constructor Position after the race	0.521	The final standing position of the constructor after the race
Driver/Car Number	0.461	The number of the driver
Quailes best secs	0.296	The best qualifying lap time achieved
Constructor Wins after Race	0.243	The cumulative wins of the teams after the race
Statues (finished)	0.228	The status of the driver finishing the race
Wins after the race	0.226	The total number of wins for the driver after the race

Based on **correlation** and **mutual information**, the most impactful features for predictive modelling are the features that indicate the performance of the driver first and then the performance of the team, as it was revealed in the basic dataset. Following the steps of the basic dataset, the performance on the qualifications seems to affect the final ranking.

Concerning the addition of the weather, the most important weather feature seems to be, Humidity, Air Pressure, and the track temperature cluster and not the Air temperature and the wind direction suggested by the existing literature that affects the aerodynamic. However, the literature review done suggested that the weather improves accuracy, a point left to be examined.

5.2.2 Enhanced Models' Performance

In this part of the dissertation the performance of the modeling methods selected previously will be presented and compared. For each modeling method, first the results of the model's pre-feature selection process will be presented and then the results post feature selection, comparing the performance and the generalization of the models tested in this dissertation. It is important to note that here the data set is enhanced with extra weather information.

Pre-Feature Selection process

The results of the performances of each model, before selecting the most appropriate features are shown in Table 20 . The Gradient Bosting machine achieves the highest R-squared (R^2) values. The model explains approximately 98.9% of the variance of the data. Additionally, it resulted in the lowest errors rates with MSE and MAE at the values of 0.27 and 0.23 respectively making it the highest performing model. These low errors suggest that the model produces predictions with minimal errors.

Random Forest regressor doesn't fall far behind. RFR offers competitive performance and strong predictive power. Its R^2 values reaches almost 0.98. The error metric falls marginally behind the GBM at the values of 0.34 and 0.27 for the MSE and the MAE respectively. This model also indicates robust predictions and is almost as good as the GBM.

Like the results encountered in the models trained with the basic dataset, Ridge regression and SVR falls slightly behind. SVR, in this dataset explains almost 0.98 % of the variance. Even though the R-squared measure is very high, an indicator of a good fit, the error fall short in comparison with the two models mentioned right above More specifically, the Mean Squared Error is calculated at 0.42 and the Mean Absolute Error at 0.39

Lastly, the Ridge Regressor, as excepted from a linear model, fails to capture all the complex relationships and this fact is depicted at it metrics. The R-squared is calculated at 0.886 , a relative high value, but lower than the rest of the models. The errors have the highest values among the four models. The MSE is valued at 0.423 and the MAE at 0.298.

Table 20:

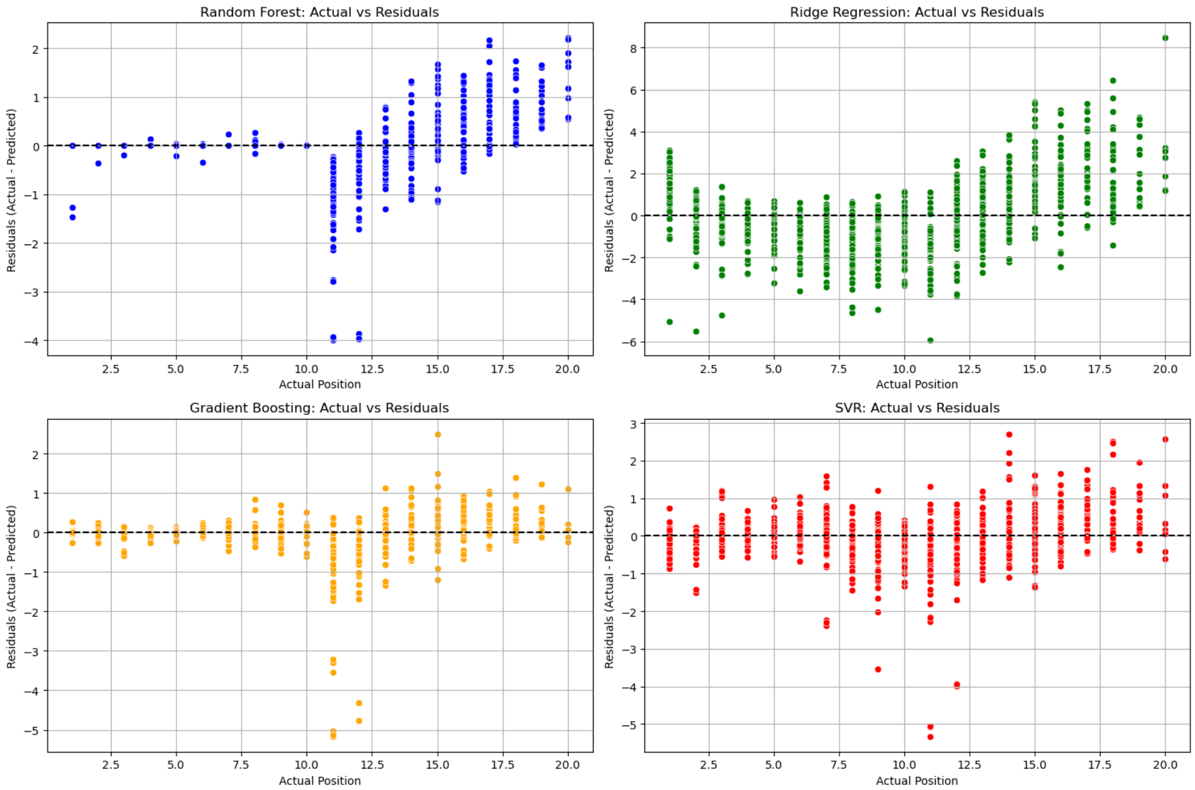
Comparison of evaluation metrics pre feature selection for data with weather.

Model	**MSE**	**R-squared**	**MAE**
Random Forest	0.338785	0.986814	0.270925
Ridge	2.919749	0.886361	1.287183
Gradient Boosting	0.270314	0.989479	0.231240
SVR	0.423436	0.983520	0.390843

Like it is already mentioned the three metrics don does not provide the full picture to understand the performance of the models. Following up, at Figure 12 the residuals, the errors of each model are plotted against the actual positions of the drivers. With this plots it is visually clear to the reader that the GBM and then the RFR provide the most accurate predictions minimizing the error. The two weaker models, the SVR and especially the Ridge regression model show the greatest variance among the four models, with their residuals scattered more.

Fig 12:

Comparison of actual values and residuals of data with weather pre feature selection.



Adding to the visualization, Table 21 and the Figure 13 that together provide a clear picture of the predictions made by the models. Both from the table and more visually clear from the figure the reader can see that the two best performing model have significantly better predictions. This is especially true regarding the first ten places.

Table 21:

Actual values and the predictions of each model

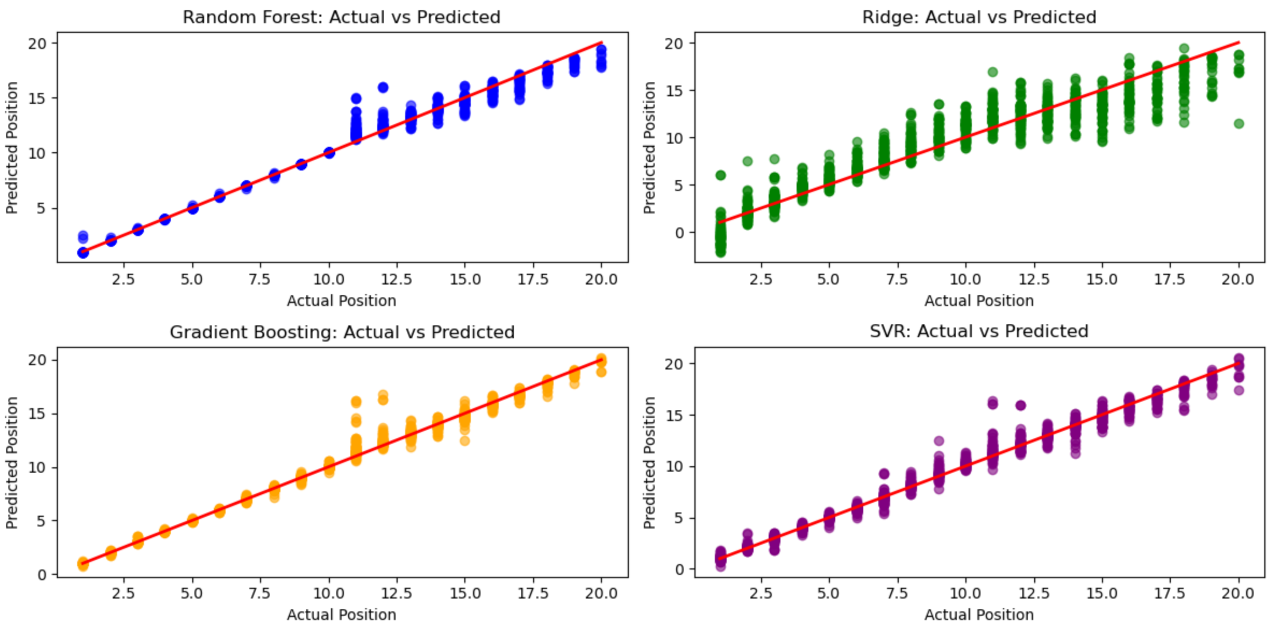
Entry	Actual Position	Predicted RF	Predicted Ridge	Predicted GMB	Predicted SVR
4716	4	4.000000	4.925572	4.017349	4.044813
4328	15	14.256667	10.738822	14.450962	14.257764
4954	14	12.940000	11.290785	12.450962	12.428548

4484	12	12.460000	11.560542	12.435563	11.854542
4066	7	7.000000	9.764231	7.043180	6.737210

In Figure 13 it is clear that the prediction of the GBM and the RFR up until the tenth place is almost identical to the actual position. The colourful dots, depicting the predictions almost fall on the red line. The red line is the actual values of the points. Again, it is visually impeccable that the models do perform better in predicting the first ten places and fall short when making predictions for the places above the 10th.

Fig 13:

Comparison of actual values and predictions with weather pre feature selection.



More specifically, for RFR and GBM, the actual versus predicted positions are highly aligned. There is minimal scatter, indicating a strong correlation between actual and predicted values, and little deviation from the actual positions.

Meanwhile for the cases of cases of Ridge Regression and SVR the data points show more scatter around the red diagonal line compared to other two. This means that the predictions of these two models are less precise, with noticeable deviations from the actual values

Overall, Random Forest and Gradient Boosting both show excellent performance, with minimal scatter and strong alignment with the diagonal line. If computational efficiency is not a major concern, Gradient Boosting is preferred due to its ability to manage complex relationships and overfitting better. However, if computational resources are a consideration, Random Forest provides a strong alternative with similar performance.

Post Feature Selection process

Moving forward with the analysis the results after performing feature selection are presented. The evaluation metrics for each model were calculated to assess their performance in predicting outcomes based on the weather data and are shown in Table 22. Like it was mentioned feature selection is a crucial part of the modelling process, and additional can help counter overfitting a possible problem considering the small dataset.

The performance of each model was measured using three key evaluation metrics, the same with the rest of the dissertation, the Mean Squared Error (MSE), R-squared (R^2), and Mean Absolute Error (MAE). These metrics provide insights into how well the models fit the data, the proportion of variance explained, and the average magnitude of errors in their predictions and are shown in Table 22.

Table 22:

Evaluation metrics for each model with weather data post feature selection

Model	**MSE**	**R-squared**	**MAE**
Random Forest	0.466286	0.981852	0.328840
Ridge	4.614031	0.820418	1.693496
Gradient Boosting	0.261956	0.989804	0.259894
SVR	1.874507	0.827043	0.801347

From the table we can extract that the lowest MSE was found in Gradient Boosting (GBM). At the same time GMB achieved the highest R-squared value of 0.989. This high R-squared error suggests that the model explains 98.9% of the variance in the data, making it a highly accurate and reliable model. Concerning the other two evaluation metrics, the GBM had MSE and MAE as low as 0.26 and 0.25 respectively.

At the same time the RFR models doesn't fall far behind. Its r-squared value reaches as high as 0.98, showcasing that the model explains approximately 98% of the variance in the data. RFR also has low error metrics. The Mean Absolute Error is calculated at 0.32 8and the Mean Squared Error at 0.466. these metrics show that the model can make robust predictions, even if it comes second.

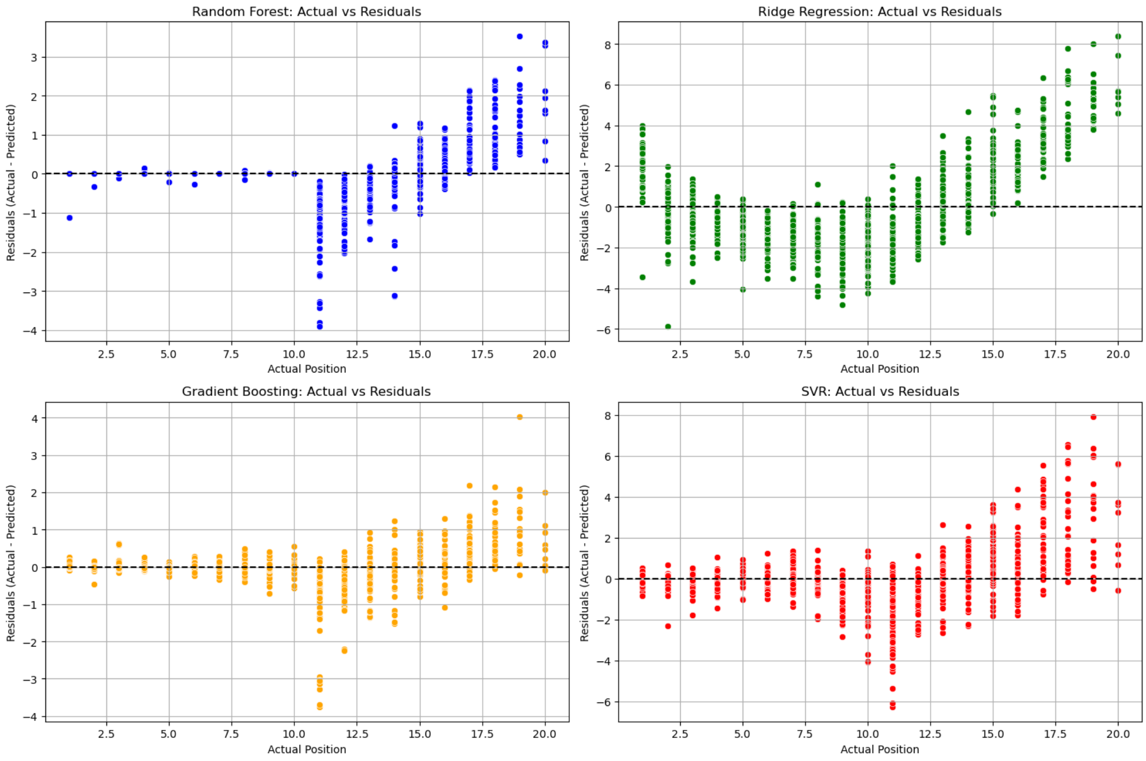
Ridge Regression and the Support Vector Regression come at the last two places at the prediction race once more.

Again, as mentioned in the previous sectors, the metrics aren't enough to fully capture the capacities of the models. Figure 14 graphically represents the errors in each models prediction. The residuals of each model are depicted as colourful dots scatter around the actual values. This graph helps the reader to visually inspect the errors. The Ridge model and the SVR makes the most errors, since their residuals are scattered more around the line of the actual values.

On the contrary, RFR and GBM have their residuals close to the fine black line. If a dot lies exactly on the line, it means that the error rate is 0, meaning that the prediction is exactly the actual value. It is interesting to note here, that the RFR makes closer prediction than the GBM, up until the 10th place. Then the residuals scatter more. On the other hand, GBM has it residual constantly close to the black like. GMB, like the RFR also makes larger error after the 10th place.

Fig 14:

Comparison of actual values and residuals of data with weather post feature selection.



However, the residuals are not the only measure that it can visualized to help in the better evaluation of the models. Figure 15 and table plot the actual values of versus the predicted values for each model.

When looking at Table 23, the reader can compare the actual values to the predicted ones. The first columns are the data point, the second the actual position and each column is the prediction of each model. In this numerical comparison it is clear that all models can make good enough prediction. However, when looking at the Figure 15, the performance of the models is clearer.

Table 23:

Actual and Predicted values for all models with weather post feature selection.

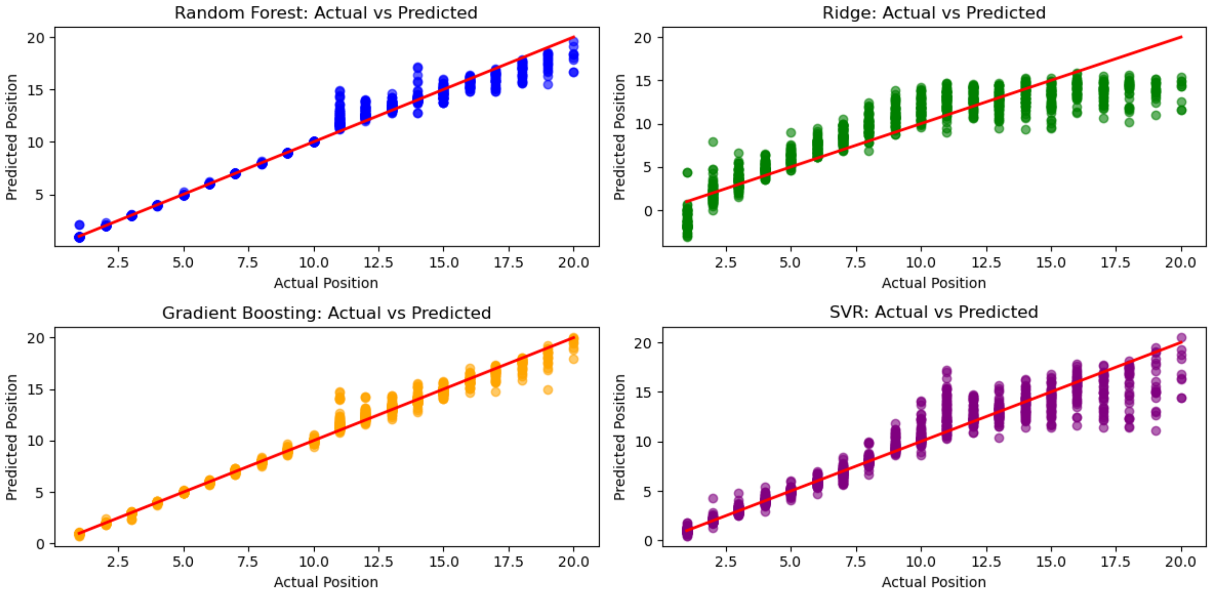
Entry	Actual Position	Predicted RF	Predicted Ridge	Predicted GMB	Predicted SVR

4716	4	3.910	5.523738	4.121769	3.724995
4328	15	14.230	11.608470	14.69659	13.757339
4954	14	12.765	12.469659	12.988475	12.706861
4484	12	12.700	12.779344	12.730169	14.152659
4066	7	7.000	9.938104	7.1683306	6.357057

In Figure 15 the actual versus the predicted values of the models are compared. With this plot it is easily visualized the performance of each model. Taking a look, it is noticeable that the GBM and the RFR produce very accurate predictions concerning at least the 10 first place, while both remaining strong with their performance. Ridge regression on the other hand and additionally SVR seem to no be able to predict racing outcomes accurately.

Fig 15:

Comparison of actual values and their predicted of data with weather post feature selection.



Keeping in mind that the Rdige model tends to smooth over extreme values it makes sense that it leads to larger deviations in this scenario.

In conclusion the Gradient Boosting machine offers the best generalization ability among all the models. This is noticed since the model comes with the lowest MSE and the highest R-squared. Its predictions, as noticed in Figure 15 are consistently close to actual values, with only minor deviations in edge cases.

5.2.3 Comparison

In conclusion, the best model in the enhanced dataset, based on the performance metrics is the Gradient Boosting. The model achieved the lowest MSE and the highest R-squared, thus being the optimal choice. Random Forest falls not far behind. RFR provides a strong alternative. The choice between the two models is hard since they both perform extremely well. To aid in the decision of the analysis, the need of comparison of the two performances for statistically important difference is risen. This comparison is made using the Analysis of Variance, the ANOVA test. This test was conducted using a 5-fold-crosvlaidation using both the R-squared (R^2) and Mean Squared Error (MSE) metrics.

Concerning the results for the R^2 measure. The results of the ANOVA test for the R^2 indicated that there was no statistically significant difference between the two models. More specifically, the F-statistic for R-squared was calculated at 0.0135, and the associated p-value was 0.9104. The p-value is above the threshold of 0.05 of statistical significance, while the F-statistic is very small. The F-statistic shows the variance between the models, which in this case in minimal. These results indicated that the models performed similarly in explaining the variance of the target variable.

Additionally, to the R^2 test, the same ANOVA test with 5-fold validation was applied to the Mean Squared Error (MSE) values. This extra step was made to further compare the models' predictive performance. The test for MSE also showed no significant difference between the models. The F-statistic was measured and had a value of 0.0024 while the p-value was calculated at 0.9625. These results suggest that, with respect to MSE, both models performed equivalently since the p-value remains significantly higher than the threshold of 0.05 and the variance of the models, the F-statistic remains minimal.

In conclusion, the ANOVA test showed that two models are performing almost identically, and their differences are minimal. The choice between them falls on the preference of the analysts. The training time of RFR is faster than the GBM as well as the prediction time. Additionally, RFR is easier to implement and more robust. In general, it is cheaper to implement, so even though GBM yields numerically better results, since their difference is not significant, this author choice would probably be the RFR.

Concerning the choice of pre- and post-feature selection process, the accuracy of all models was better on the dataset pre feature selection, if just the number of the evaluation metrics are taken into account. However, the process of feature selection is a process sacrificing a bit of accuracy to train models that are generalized better. Even though the accuracy of the models drops post feature selection, the models generalize better, thus it makes the choice of the post feature selection models the decision of the analysis.

Among the advantages of picking the model post feature selection, giving ground to better generalizations, are the reduced complexity of the models and the lower dimensionality. These facts make the training and the prediction process cheaper and faster too.

6 Discussion

In this part of the dissertation, the discussion of the results presented earlier takes place and the performance of the predictive models is evaluated. The models were analysed with and without weather data. In this section the key findings are highlighted, their alignment with prior research and the boarder implication for Formula 1 stakeholders.

The findings of this study highlight the value of predictive analytics, particularly the statistical models of regression, in understanding Formula 1 race outcomes, with targeting the final position. Using historical data, qualitative and quantitative, as well as incorporating weather data, a part lacking in previous research, this research demonstrated the effectiveness of various methods in forecasting driver positions. Additionally, it is discussed where the importance of incorporating weather data affects the final predictions.

In this study, four different regression models were trained and then evaluated in their accuracy of predicting race outcomes. These models were the Random Forest Regressor (RF), the Ridge Regression, the Gradient Boosting Machine (GBM) and Support Vector Regressor (SVR). Each model was evaluated using three distinct evaluation metrics. The metrics were Mean Squared Error (MSE), Mean Absolute Error (MAE), and R-squared (R^2) values.

6.1 Model Performance

To summarize the performance of these models, in both cases, Random Forest and Gradient Boosting significantly outperformed the two others, Support Vector Regression and Ridge Regression. The two cases mentioned here, being in the data with data and without weather information. Their R^2 values always exceeded 0.94. The models showed great capability in handling many non-linear relationships between the variables, making them the best-suited models for the complex dynamics of Formula 1 and probably more sports, since the sports data hides nonlinear, complex relationships.

Between the two superior models, Random Forest Regressor demonstrated impressive performance on accuracy with an R^2 higher than 0.94 in the preferred case of pre feature selection for the basic dataset indicating that approximately 94% of the variance in the target variable (race position) could be explained by the model. Additionally, the model performed well even after feature selection, with the R^2 score slightly decreasing to 0.9245. When the dataset was enhanced with the weather information the model reached a R-squared metric of higher of 0.98 both pre and post feature selection.

In comparison, Gradient Boosting Machine indicated a R squared of 0.94 and 0.92 when trained in the basic dataset pre and post feature selection respectively. These scores are almost identical to the RFR scores. Particularly in the case of weather data, GBM showed higher results than RFR with both times almost reaching a 0.99 R squared score.

Support Vector Regression performed moderately good in all cases, while the Ridge model had a harder time making accurate predictions, but this was due to its linear nature. The choice of the model after all, was to verify the complex nature of sport prediction modelling.

To conclude, the best models were GBM and RFR. These findings align with previous research, which highlights the robustness of ensemble methods in sports analytics (Haghighat et al., 2013), (Van Kesteren & Bergkamp, 2023). Concerning the model performance and which models, between the GBM of the RFR was better, the ANOVA test showed that there is not any significant difference between the two, so it is entirely based on the choice of the analysist. Additionally concerning the pre or post after feature selection, all models lost some of their accuracy only to gain in generalization, as expected from the process.

Lastly, one point to note on the performance of the models is that all four models in all cases had significantly better predictions in the first 10 places, and not so much on the last ten.

6.2 Basic and Enhanced Dataset Comparison

In this part we discuss whether the addition of the weather data had the expected effects. Witch the enhanced dataset included, the effect was an improvement in accuracy for all the models. The results of this addition showed that all the models examined significantly improved their accuracy when trained on the enhanced dataset with the weather data.

This conclusion verifies the existing literature, confirming that the inclusion of weather improves accuracy, however the extent of its effect was not nearly as important as other features like the ones concerning the drivers and team performances.

6.2.1 Key factors

These facts, suggest that other factors such as the team and the driver's performance play a more dominant role. The top key predictors for the target value of 'position' were the features presenting the performance of the driver and the team, such as 'points,' 'grid position,' qualifying position,' and 'constructor points after the race. These features were identified through mutual information and correlation analysis. These results come in harmony findings of many researchers mentioned on the related work section (Patil et al., 2023)(Bell et al., 2016; Van Kesteren & Bergkamp, 2023).

More specifically on the most important factors, the first consideration is on the individual skill of the driver. Apart from the feature depicting the personal performance of the driver, like the ones mentioned in the previous paragraph, there is the specific feature number_y. This feature is the number of the driver, and its effect can measure the personal influence of each driver, thus their skill. This feature is not as nearly as important as other performance measures. These finding comes to verify the findings of previous research that the team and the car plays an important role in F1 results (Bell et al., 2016; Van Kesteren & Bergkamp, 2023).

Additionally, the final position is influenced by the qualification performance. This feature seems to play an important role. In accordance with previous research (Pfitzner & Rishel, 2008; Silva & Silva, 2010), claiming that the qualifying and previous races play an important role in F1 results, this dissertation's feature analysis verifies it too.

Moreover, this research interestingly found weather-related features, only moderately impactful, thus revealing that features like air temperature, rainfall and humidity were identified as only moderately impactful variables when the Mutual Information and the correlation analysis took place, supporting observations by (Heilmeier, Thomaser, et al., 2020). Concerning the weather data, it was the Wind Direction was added to the feature even though it was not revealed in the correlation analysis nor the mutual information analysis, since it is a very important factor in the aerodynamics of the car and its grip on the track (Saleh Mousavi-Bafrouyi et al., 2021).

6.2.2 Feature Engineering

For feature engineering part, after the feature selection, simplifying the model and reducing its complexity, led to a slight trade-off in performance. The model's R^2 decreased slightly, and the MSE increased compared to the pre-feature selection version. However, the post-feature selection model demonstrated improved generalization, as evidenced by a lower variance between training and test performance. This reduction in variance indicates that the model is less likely to overfit the training data, thus improving its ability to generalize to unseen data.

The post-feature selection process slightly reduced performance for both Random Forest and Gradient Boosting, but the trade-off was minimal and led to better generalization. Thus, for predicting F1 race positions with weather data, Gradient Boosting stands as the most effective and optimal model, while Random Forest offers a strong alternative if computational efficiency is prioritized. The feature selection played its role in generalization, but not remarkably notable. Ridge Regression and SVR should be excluded due to their subpar predictive accuracy and higher error rates.

The features that are deemed important emphasize the performance history of each driver, one might argue that it is their skill that plays an important role in the final position. However, the driver's performance is not the only factor playing an important role. The constructor's performance seems to play as much an important role as the driver, possibly a more important role, highlighting that Formula 1 is not only about that driver, but also about the car, too.

To conclude, the models seem to make more accurate predictions when trained with the enhanced dataset. However, there is a chance that the models developed grew too complex and while gaining accuracy the models ended up losing in generalization.

6.3 Implications for stakeholders

The first group of stakeholders are the Formula 1 teams. The teams can allocate resources and optimize qualifying performance; thus, guiding team strategy over prioritizing weather data collection. Weather is deemed important for the use of tires, however a good team and a good driver will always be good, no matter if the small temperature changes and a good car will remain good no matter what the weather conditions are.

The second group affected by these findings are those who care about its predictive applications, like investors. For this group this research provides a robust framework for an easy and understandable way the deployment of statistical models. The modelling of this research balances accuracy and generalizability, crucial for live race strategies and long-term planning. 1.

6.4 Assumptions, Limitations and Threats

In this study there were several assumptions made to ensure the quality, validity and applicability of the findings. The first one is that the study assumes that the data collected from Ergstat API and complementary resources like Kaggle are accurate and that they accurately represent the Formula 1 races from the season of 2014 to season 2023. This assumption is the most crucial as the quality of the data and the reliability depend on it.

Secondly the analysis presumes that the selection of features during feature extraction was made correctly. It assumes that the selected features are significant determinants of each race and that the whole process was a success. These features are considered to represent the most influential variables in the final ranking of each race in Formula 1. Lastly, the predictive models employed are assumed to generalize effectively to new unseen data.

Additionally, to the assumption, this dissertation comes with its limitation. First, the datasets study is relatively small. Even though considering the fact that the dataset is larger than older studies, the data included in this study is deemed relatively small. The small dataset may influence the model's performance leading to high accuracies and its lower its generalization. Moreover, there could be data imbalances not caught in the analysis. Specific conditions, such as the rain, might have been underrepresented. For example, the amount of rain could not be represented, thus potentially limiting robustness of the findings regarding the weather impact.

Another limitation lies in the reliance of correlation analysis and the mutual information for feature selection. Even though some of the features selected are confirmed by existing research, still they might have overlooked more complex relations.

7 Conclusions

This dissertation aims to answer three questions. Considering the first research question, on revieing existing modeling in Formula 1. The comparison of traditional statistical models showed that sports prediction is not covered by linear models, such as the Ridge model. The findings of the research showed that for Formula 1, like the rest of the sport analysis industry, the effectiveness of ensemble methods is undeniable. The effectiveness of the ensemble models is particularly true when considering complex datasets like those encountered in F1. The Random Forest Regressor consistently emerged as one of the best-performing models alongside Gradient Boosting.

Moving to the next research question on the importance of features and the effect of weather on the accuracy of the models. The importance of weather data was not assessed to be as important as initially thought. From this research, adding to the small research done on Formula 1 it seems that the performance metrics, both from the driver and the team (constructors) are significantly more important than the effect of the weather. Simply put, a good driver remains good no matter the weather, the same goes for the car.

However, even though the weather was not as important as it was initially thought, its addition resulted in a noticeable improvement in model performance. All models saw increases in R^2 values and a reduction in their errors. These facts indicate that the weather features played a significant role in enhancing predictive accuracy.

The implications of this study are significant for the use of predictive analytics in Formula 1, an underexplored area. Given its popularity Formula 1 is yet to be explored in sports analytics. This research provided some basic insights for teams, which can use the knowledge to build on it, for better understanding team competitiveness and allocating their resources. Additionally, it can help investors and analysts to better forecast race outcomes and understand the sport.

7.1 Future Work

In terms of recommendations and suggestions for future research, first, the research can address some of the already identified limitations. For example, it can incorporate larger and more diverse datasets. For example, in future work real-time telemetry data can be added or expanded weather variables measuring more features or quantifying feature like 'rain'.

Furthermore, the exploration of advanced ML techniques is recommended. For instance, Recurrent Neural Networks could be deployed (RNNs) or Long-Short-Term Models (LSTMs) can be employed for deeper understanding of the complexities of the feature from the models. Models like the one mentioned above can have better results, since they have the ability to capture temporal dependencies. However, even a Multilayer Perceptron would be suggested to yield better results.

Additionally, another suggestion is to include driver-specific metrics. For example, the driver's psychology is a metric that can hardly be quantified but can affect performance or team dynamics. Apart from the driver's psychology, one can aim to include features like driver's injuries. Another promising feature to add to is the influence of social media on drivers and teams' performance. Social media might affect performance in ways that have yet to be determined. These complex features are hard to quantify but can help train models and improve our understanding of the sport.

8 References

Aguad, F., & Thraves, C. (2024). Optimizing pit stop strategies in Formula 1 with dynamic programming and game theory. *European Journal of Operational Research*, *319*(3), 908–919. https://doi.org/10.1016/j.ejor.2024.07.011

Allender, M. (2008). Predicting The Outcome Of NASCAR Races: The Role Of Driver Experience. In *Journal of Business & Economics Research-March* (Vol. 6).

Basak, D., Pal, S., & Patranabis, D. C. (2007). Support Vector Regression. In *Neural Information Processing-Letters and Reviews* (Vol. 11, Issue 10).

Belghazi, M. I., Baratin, A., Rajeswar, S., Ozair, S., Bengio, Y., Courville, A., & Hjelm, D. (2018). *Mutual Information Neural Estimation*.

Bell, A., Smith, J., Sabel, C. E., & Jones, K. (2016). Formula for success: Multilevel modelling of Formula One Driver and Constructor performance, 1950-2014. In *Journal of Quantitative Analysis in Sports* (Vol. 12, Issue 2, pp. 99–112). Walter de Gruyter GmbH. https://doi.org/10.1515/jqas-2015-0050

Black, J., Kueper, J., & Williamson, T. (2023). An introduction to machine learning for classification and prediction. *Family Practice*, *40*(1), 200–204.

Blaikie, A. D., Abud, G. J., David, J. A., & Drew Pasteur, R. (2011). *NFL & NCAA Football Prediction using Artificial Neural Networks*.

Bunker, R. P., & Thabtah, F. (2017). A machine learning framework for sport result prediction. *Applied Computing and Informatics*, *15*(1), 27–33. https://doi.org/10.1016/j.aci.2017.09.005

Bunker, R. P., & Thabtah, F. (2019). A machine learning framework for sport result prediction. *Applied Computing and Informatics*, *15*(1), 27–33. https://doi.org/10.1016/j.aci.2017.09.005

Bunker, R., & Susnjak, T. (2022). The Application of Machine Learning Techniques for Predicting Match Results in Team Sport: A Review. In *Journal of Artificial Intelligence Research* (Vol. 73).

Davoodi, E., & Khanteymoori, A. R. (2010). Horse Racing Prediction Using Artificial Neural Networks. *RECENT ADVANCES in NEURAL NETWORKS, FUZZY SYSTEMS & EVOLUTIONARY COMPUTING*, 306.

Graves, T., Reese, S., & Fitzgerald, M. (2003). *Hierarchical Models for Permutations: Analysis of Auto Racing Results*.

Grover, L. K., & Mehra, R. (2008). The lure of statistics in data mining. *Journal of Statistics Education, 16*(1). https://doi.org/10.1080/10691898.2008.11889552

Haghighat, M., Rastegari, H., & Nourafza, N. (2013). *A Review of Data Mining Techniques for Result Prediction in Sports*. www.Databasebasketball.com,

Han, J., Kamber, M., & Pei, J. (2011). *Data Mining. Concepts and Techniques, 3rd Edition (The Morgan Kaufmann Series in Data Management Systems)*.

Haque, S., Eberhart, Z., Bansal, A., & McMillan, C. (2022). Luck is Hard to Beat: The Difficulty of Sports Prediction. *IEEE International Conference on Program Comprehension, 2022-March,* 36–47. https://doi.org/10.1145/nnnnnnn.nnnnnnn

Heilmeier, A., Graf, M., Betz, J., & Lienkamp, M. (2020). Application of Monte Carlo methods to consider probabilistic effects in a race simulation for circuit motorsport. *Applied Sciences (Switzerland), 10*(12). https://doi.org/10.3390/app10124229

Heilmeier, A., Thomaser, A., Graf, M., & Betz, J. (2020). Virtual strategy engineer: Using artificial neural networks for making race strategy decisions in circuit motorsport. *Applied Sciences (Switzerland), 10*(21), 1–32. https://doi.org/10.3390/app10217805

Hucaljuk, J., & Rakipović, A. (2011). *Predicting football scores using machine learning techniques*.

Hucaljuk, J., & Rakipovic, A. (2011). Predicting football scores using machine learning techniques. *2011 Proceedings of the 34th International Convention MIPRO,* 1623–1627. https://api.semanticscholar.org/CorpusID:30846617

James, Gareth., Witten, Daniela., Hastie, Trevor., & Tibshirani, Robert. (2017). *An introduction to statistical learning: with applications in R*. Springer : Springer Science+Business Media.

Judde, C., Booth, R., & Brooks, R. (2013). Second Place Is First of the Losers: An Analysis of Competitive Balance in Formula One. *Journal of Sports Economics, 14*(4), 411–439. https://doi.org/10.1177/1527002513496009

Kahn, J. (2003). Neural Network Prediction of NFL Football Games. *World Wide Web Electronic Publication*.

Konstantinov, A. v., & Utkin, L. v. (2021). Interpretable machine learning with an ensemble of gradient boosting machines. *Knowledge-Based Systems, 222*. https://doi.org/10.1016/j.knosys.2021.106993

Kuo, B. C., Ho, H. H., Li, C. H., Hung, C. C., & Taur, J. S. (2014). A kernel-based feature selection method for SVM with RBF kernel for hyperspectral image classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *7*(1), 317–326. https://doi.org/10.1109/JSTARS.2013.2262926

Lecun, Y., Bottou, L., Orr, G. B., & Mm, K.-R. (1998). Efficient BackProp. In *Neural Networks: tricks of the trade*. Springer.

Ledesma, C., Choo, W., & Hale, P. (2007). *Real-Time Decision Making in Motorsports: Analytics for Improving Professional Car Race Strategy Author's signature ………………………………………………………………………………………………… System Design and Management Program*.

Maszczyk, A., Gołaś, A., Pietraszewski, P., Roczniok, R., Zając, A., & Stanula, A. (2014). Application of Neural and Regression Models in Sports Results Prediction. *Procedia - Social and Behavioral Sciences*, *117*, 482–487. https://doi.org/10.1016/j.sbspro.2014.02.249

Matsui, S., Le-Rademacher, J., & Mandrekar, S. J. (2021). Statistical Models in Clinical Studies. *Journal of Thoracic Oncology*, *16*(5), 734–739. https://doi.org/10.1016/j.jtho.2021.02.021

McCabe, A., & Trevathan, J. (2008). Artificial intelligence in sports prediction. *Proceedings - International Conference on Information Technology: New Generations, ITNG 2008*, 1194–1197. https://doi.org/10.1109/ITNG.2008.203

Mccabe, A., & Trevathan, J. (2008). *Artificial Intelligence in Sports Prediction*. 1194–1197. https://doi.org/10.1109/ITNG.2008.203

Mitchell, J. (n.d.). *Data Mining Methods for Sports Prediction*.

Müller, A. C., & Guido, S. (2017). *Introduction to Machine Learning with Python A GUIDE FOR DATA SCIENTISTS Introduction to Machine Learning with Python*.

Natekin, A., & Knoll, A. (2013). Gradient boosting machines, a tutorial. *Frontiers in Neurorobotics*, *7*(DEC). https://doi.org/10.3389/fnbot.2013.00021

Papageorgiou, G., Sarlis, V., & Tjortjis, C. (2024). Unsupervised Learning in NBA Injury Recovery: Advanced Data Mining to Decode Recovery Durations and Economic Impacts. *Information (Switzerland)*, *15*(1). https://doi.org/10.3390/info15010061

Patil, A., Jain, N., Agrahari, R., Hossari, M., Orlandi, F., & Dev, S. (2023). A Data-Driven Analysis of Formula 1 Car Races Outcome. *Communications in Computer and Information Science*, *1662 CCIS*, 134–146. https://doi.org/10.1007/978-3-031-26438-2_11

Pfitzner, B. C., & Rishel, T. D. (2008). *Do Reliable Predictors Exist for the Outcomes of NASCAR Races? thesportjournal.org/article/do-reliable-predictors-exist-for-the-outcomes-of-nascar-races*. http://jayski.thatsracin.com/

Plevris, V., Solorzano, G., Bakas, N. P., & ben Seghier, M. E. A. (2022). Investigation of Performance Metrics in Regression Analysis and Machine Learning-Based Prediction Models. *World Congress in Computational Mechanics and ECCOMAS Congress*. https://doi.org/10.23967/eccomas.2022.155

Rodriguez-Galiano, V., Sanchez-Castillo, M., Chica-Olmo, M., & Chica-Rivas, M. (2015). Machine learning predictive models for mineral prospectivity: An evaluation of neural networks, random forest, regression trees and support vector machines. *Ore Geology Reviews*, *71*, 804–818. https://doi.org/10.1016/j.oregeorev.2015.01.001

Rovetta, A. (2020). Raiders of the Lost Correlation: A Guide on Using Pearson and Spearman Coefficients to Detect Hidden Correlations in Medical Sciences. *Cureus*. https://doi.org/10.7759/cureus.11794

Saleh Mousavi-Bafrouyi, S. M., Reza Kashyzadeh, K., Khorsandijou, S. M., & Saleh Mousavi, S. M. (2021). Effects of Road Roughness, Aerodynamics, and Weather Conditions on Automotive Wheel Force. In *International Journal of Engineering* (Vol. 34, Issue 02).

Samuel, M. B., Felix, P., Miguel, Y. N., Cyrille, T. S., & Talla, P. K. (2020). Study and Simulation of the Fuel Consumption of a Vehicle with Respect to Ambient Temperature and Weather Conditions. *International Journal of Engineering Technologies and Management Research*, *7*(1), 24–35. https://doi.org/10.29121/ijetmr.v7.i1.2020.480

Sarlis, V., Papageorgiou, G., & Tjortjis, C. (2023). Sports Analytics and Text Mining NBA Data to Assess Recovery from Injuries and Their Economic Impact. *Computers*, *12*(12). https://doi.org/10.3390/computers12120261

Sarlis, V., Papageorgiou, G., & Tjortjis, C. (2024a). Leveraging Sports Analytics and Association Rule Mining to Uncover Recovery and Economic Impacts in NBA Basketball. *Data*, *9*(7). https://doi.org/10.3390/data9070083

Sarlis, V., Papageorgiou, G., & Tjortjis, C. (2024b). Leveraging Sports Analytics and Association Rule Mining to Uncover Recovery and Economic Impacts in NBA Basketball. *Data*, *9*(7). https://doi.org/10.3390/data9070083

Sarlis, V., & Tjortjis, C. (2020). Sports analytics — Evaluation of basketball players and team performance. *Information Systems*, *93*. https://doi.org/10.1016/j.is.2020.101562

Sarlis, V., & Tjortjis, C. (2024). Sports Analytics: Data Mining to Uncover NBA Player Position, Age, and Injury Impact on Performance and Economics. *Information (Switzerland)*, *15*(4). https://doi.org/10.3390/info15040242

Schumaker, R. P. (2013). Machine learning the harness track: Crowdsourcing and varying race history. *Decision Support Systems*, *54*(3), 1370–1379. https://doi.org/10.1016/j.dss.2012.12.013

Shmueli, G. P. N. B. P. (2010). *Data Mining for Business Intelligence* (Second). John Wiley & Sons.

Silva, K. M., & Silva, F. J. (2010). *A tale of two motorsports: A graphical-statistical analysis of how practice, qualifying, and past success relate to finish position in NASCAR and Formula One racing,* Retrieved from http://newton.uor.edu/FacultyFolder/Silva/NASCARvF1.pdf

Smith, P. F., Ganesh, S., & Liu, P. (2013). A comparison of random forest regression and multiple linear regression for prediction in neuroscience. *Journal of Neuroscience Methods*, *220*(1), 85–91. https://doi.org/10.1016/j.jneumeth.2013.08.024

Tajana, M. (2022). *Critical analysis of competitive balance in Formula 1 motor racing. Does budget concentration affect teams' performance?*

Tatachar, A. v. (2021). Comparative Assessment of Regression Models Based On Model Evaluation Metrics. *International Research Journal of Engineering and Technology*. www.irjet.net

van Kesteren, E. J., & Bergkamp, T. (2023). Bayesian analysis of Formula One race results: Disentangling driver skill and constructor advantage. *Journal of Quantitative Analysis in Sports*, *19*(4), 273–293. https://doi.org/10.1515/jqas-2022-0021

Watt, J., Borhani, R., & Katsaggelos, A. K. (2016). Machine learning refined: Foundations, algorithms, and applications. In *Machine Learning Refined: Foundations, Algorithms, and Applications*. Cambridge University Press. https://doi.org/10.1017/CBO9781316402276

9 Appendix

In the appendix the reader can find the source code, as well as a data sample

9.1 Source code

Circuits

```
# Convert the "Street" and "Permanent" to numeric values
numeric_circuit_types = {name: 1 if circuit_type == "Street" else 0 for name, circuit_type in circuit_types.items()}
# Map the circuit names in `circuits_final` to the new numeric values
df_circuits['circuit_type'] = df_circuits['name'].map(numeric_circuit_types)


df_circuits.head()
```

drivers

```
df_drivers['driver_name'] = df_drivers['forename'] + ' ' + df_drivers['surname']
```

qualifications

```
# Function to convert 'minutes:seconds.milliseconds' to total seconds (float)
def convert_time_to_seconds(time_str):
    if in instance(time_str, str) and ':' in time_str:
        minutes, seconds = time_str.split(':')
        total_seconds = int(minutes) * 60 + float(seconds)
        return total_seconds
    return pd.to_numeric(time_str, errors='coerce')  # Handles NaN
```

df_qualies['q1'] = df_qualies['q1'].apply(convert_time_to_seconds)

df_qualies['q2'] = df_qualies['q2'].apply(convert_time_to_seconds)

df_qualies['q3'] = df_qualies['q3'].apply(convert_time_to_seconds)

Check result

df_qualies.info()

df_qualies['qualies_best'] = df_qualies[['q1', 'q2', 'q3']].min(axis=1)

weather

temperatures classification

k_means for temp classification

from sklearn.cluster import KMeans

import numpy as np

track_temps = weather_df[['TrackTemp']].values

kmeans = KMeans(n_clusters=3, random_state=42) # 3 clusters - cold, warm, hot

weather_df['TempCluster'] = kmeans.fit_predict(track_temps)

Label clusters based on temperature

cluster_means = weather_df.groupby('TempCluster')['TrackTemp'].mean().sort_values()

Map cluster labels to "cold", "warm", "hot"

cluster_label_mapping = {cluster: label for cluster, label in zip(cluster_means.index, ['cold', 'warm', 'hot'])}

weather_df['TempLabel'] = weather_df['TempCluster'].map(cluster_label_mapping)

weather_df[['season', 'round', 'TrackTemp', 'TempLabel', 'TempCluster']].head()

Rainfall

weather_df['Rainfall'] = weather_df['Rainfall'].astype(bool)

Define the rain status for each round and season

```
round_rain_status = weather_df.groupby(['season', 'round'])['Rainfall'].apply(
    lambda x: 2 if x.nunique() > 1 else (1 if x.any() else 0)
).reset_index(name='rain')

round_rain_status
# 125 races = correct
```

Optimal Timestamps

```
weather_df['Time'] = weather_df['Time'].astype(str)
# Remove 'days' prefix
weather_df['Time'] = pd.to_timedelta(weather_df['Time'].str.split(' ').str[-1])
# Calculate seconds since start
weather_df['SecondsSinceStart'] = weather_df['Time'].dt.total_seconds()
weather_df.head(10)
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt

# use elbow method to decide optimal number
X = weather_df[['SecondsSinceStart']]

inertia = [] #stores the squared distances
for n in range(1, 21):  # Try different numbers of clusters (1 to 20)
    kmeans = KMeans(n_clusters=n, random_state=42)
    kmeans.fit(X)
    inertia.append(kmeans.inertia_)

# Plot to observe the elbow point
plt.plot(range(1, 21), inertia, marker='o')
plt.xlabel('Number of Clusters')
plt.ylabel('Inertia')
```

```
plt.title('Elbow Method for Optimal K')
plt.show()

optimal_clusters = 3 #elbow around!
# Kmeans again
X = weather_df[['SecondsSinceStart']]
kmeans = KMeans(n_clusters=optimal_clusters, random_state=42)
weather_df['Cluster'] = kmeans.fit_predict(X)
# cluster centers
cluster_centers = kmeans.cluster_centers_.flatten()
# select the closest row to each cluster center
def get_closest_rows(group, cluster_centers):
    #  distance of each timestamp to the cluster centers
    distances = np.abs(group['SecondsSinceStart'].values[:, None] - cluster_centers)

    # get the closest timestamp
    closest_rows = []
    for center in cluster_centers:
        closest_idx = np.argmin(np.abs(group['SecondsSinceStart'].values - center))
        closest_row = group.iloc[closest_idx]
        # closest_rows.append(group.iloc[closest_idx])

        # avoid duplicates, bcs there was
        if closest_row['SecondsSinceStart'] not in [row['SecondsSinceStart'] for row in clos-
est_rows]:
            closest_rows.append(closest_row)

    return pd.DataFrame(closest_rows)
# Group by 'season' and 'round', and apply the function
```

```
final_weather_df = weather_df.groupby(['season', 'round']).apply(get_closest_rows, cluster_centers=cluster_centers).reset_index(drop=True)
final_weather_df.head(15)
```

Models

```
columns_to_drop = ['country', 'driver_name', 'code', 'con_name', 'time', 'Time', 'TempLabel']


df = df_final_data.copy()
df = df.drop(columns=columns_to_drop)
df.info()
```

user manual, data samples, etc.

All features

```
target = 'position'

# Include all columns except the target 'position'
features = [col for col in df.columns if col != target]

# Separate the features and target
X = df[features]
y = df[target]

# Identify numerical and categorical features
numerical_features = X.select_dtypes(include=['float64', 'int64']).columns.tolist()
categorical_features = X.select_dtypes(include=['object']).columns.tolist()

# Print the identified features
print("Numerical features:", numerical_features)
print("Categorical features:", categorical_features)
```

Feature selection without weather

```
target = 'position'
features = ['points', 'pos_after_race', 'con_pos_after_race', 'grid', 'points_after_race',
            'quali_pos', 'qualies_best_secs', 'number_y', 'circuit_type', 'con_wins_after_race']

# Separate the features and target
X = df[features]
y = df[target]

numerical_features = X.select_dtypes(include=['float64', 'int64']).columns.tolist()
categorical_features = X.select_dtypes(include=['object']).columns.tolist()

print("Numerical features:", numerical_features)
print("Categorical features:", categorical_features)
```

Feature selection with weather

```
target = 'position'
features = ['points', 'pos_after_race', 'con_pos_after_race', 'con_points_after_race',
            'grid', 'points_after_race', 'quali_pos', 'qualies_best_secs', 'number_y',
            'circuit_type', 'Pressure', 'TempCluster', 'Humidity', 'AirTemp' ]
# add air pressure

# Separate the features and target
X = df[features]
y = df[target]

numerical_features = X.select_dtypes(include=['float64', 'int64']).columns.tolist()
categorical_features = X.select_dtypes(include=['object']).columns.tolist()

print("Numerical features:", numerical_features)
```

```
print("Categorical features:", categorical_features)
```

Random Forest Regressor

```
best_pipeline = Pipeline(steps=[
    ('preprocessor', preprocessor),
    ('regressor', RandomForestRegressor(n_estimators=200, max_depth=30, min_samples_split=2, random_state=42))
])

# Fit the pipeline on the training data
best_pipeline.fit(X_train, y_train)
# Make predictions on the test set
y_pred_rf = best_pipeline.predict(X_test)
# Compare the predicted positions with the actual positions
comparison_df = pd.DataFrame({
    'Actual Position': y_test,
    'Predicted Position_RF': y_pred_rf
})
# Display the first few rows of the comparison
print(comparison_df.head())

# Evaluate the model's performance
mae = mean_absolute_error(y_test, y_pred_rf)
mse = mean_squared_error(y_test, y_pred_rf)
r2 = r2_score(y_test, y_pred_rf)
print(f'Mean Absolute Error: {mae}')
print(f"Mean Squared Error (MSE): {mse}")
print(f"R-squared: {r2}")
```

Ridge

```
from sklearn.linear_model import Ridge
from sklearn.model_selection import GridSearchCV
from sklearn.metrics import mean_squared_error, r2_score

ridge_model = Ridge()



param_grid_ridge = {
    'alpha': [0.1, 1, 10, 100]  # Regularization strength, larger values mean more regulari-
zation
}
# GridSearchCV with Ridge Regression
grid_search_ridge = GridSearchCV(ridge_model, param_grid_ridge, cv=5, n_jobs=-1,
scoring='neg_mean_squared_error')
# Create the pipeline with preprocessor and Ridge model
pipeline_ridge = Pipeline(steps=[
    ('preprocessor', preprocessor),
    ('regressor', grid_search_ridge)
])
# Fit the grid search to the data
pipeline_ridge.fit(X_train, y_train)
# Get the best model from grid search
best_ridge_model = pipeline_ridge.named_steps['regressor'].best_estimator_
y_pred_ridge = pipeline_ridge.predict(X_test)

# Evaluate the model
mse_ridge = mean_squared_error(y_test, y_pred_ridge)
mae_ridge = mean_absolute_error(y_test, y_pred_ridge)
r2_ridge = r2_score(y_test, y_pred_ridge)
print(f"Best Hyperparameters for Ridge: {grid_search_ridge.best_params_}")
```

```
print(f"Mean Squared Error (MSE) for Ridge: {mse_ridge}")
print(f"Mean Absolute Error (MAE) for Ridge: {mae_ridge}")
print(f"R-squared for Ridge: {r2_ridge}")

from sklearn.ensemble import GradientBoostingRegressor
from sklearn.model_selection import GridSearchCV
from sklearn.metrics import mean_squared_error, r2_score
```

Gradient Boosting

```
# Define the Gradient Boosting model
gbm_model = GradientBoostingRegressor()

# Define the hyperparameters to tune
param_grid_gbm = {
    'n_estimators': [100, 200, 300],  # Number of boosting stages (trees)
    'learning_rate': [0.01, 0.1, 0.2],  # Learning rate
    'max_depth': [3, 4, 5],  # Max depth of each tree
}

# GridSearchCV with Gradient Boosting
grid_search_gbm = GridSearchCV(gbm_model, param_grid_gbm, cv=5, n_jobs=-1, scoring='neg_mean_squared_error')
# Fit the grid search to the data
grid_search_gbm.fit(X_train, y_train)
# Get the best model from grid search
best_gbm_model = grid_search_gbm.best_estimator_
# Make predictions with the best model
y_pred_gbm = best_gbm_model.predict(X_test)

# Evaluate the model
```

```
mse_gbm = mean_squared_error(y_test, y_pred_gbm)
mae_gbm = mean_absolute_error(y_test, y_pred_gbm)
r2_gbm = r2_score(y_test, y_pred_gbm)
print(f"Best Hyperparameters for GBM: {grid_search_gbm.best_params_}")
print(f"Mean Squared Error (MSE) for GBM: {mse_gbm}")
print(f"Mean Absolute Error (MAE) for GBM: {mae_gbm}")
print(f"R-squared for GBM: {r2_gbm}")
```

Support Vector Regression

```
from sklearn.svm import SVR
from sklearn.model_selection import GridSearchCV
from sklearn.metrics import mean_squared_error, r2_score

# Define the SVR model with RBF kernel
svr_model = SVR(kernel='rbf')

# Define the hyperparameters to tune
param_grid_svr = {
    'C': [0.1, 1, 10, 100],          # Regularization parameter
    'epsilon': [0.01, 0.1, 0.2],    # Margin of tolerance for the regression
    'kernel': ['rbf'],              # Using the Radial Basis Function kernel
}

# GridSearchCV with SVM
grid_search_svr = GridSearchCV(svr_model, param_grid_svr, cv=5, n_jobs=-1, scor-
ing='neg_mean_squared_error')

# Fit the grid search to the data
grid_search_svr.fit(X_train, y_train)
```

```
# Get the best model from grid search
best_svr_model = grid_search_svr.best_estimator_

# Make predictions with the best model
y_pred_svr = best_svr_model.predict(X_test)
# Evaluate the model
mse_svr = mean_squared_error(y_test, y_pred_svr)
r2_svr = r2_score(y_test, y_pred_svr)
mae_svr = mean_absolute_error(y_test, y_pred_svr)
print(f"Best Hyperparameters for SVR: {grid_search_svr.best_params_}")
print(f"Mean Squared Error (MSE) for SVR: {mse_svr}")
print(f"R-squared for SVR: {r2_svr}")
print(f"Mean Absolute Error (MAE) for SVR: {mae_svr}")
```

Anova Testing

```
from sklearn.model_selection import cross_val_score
from scipy import stats
import numpy as np
# Perform 5-fold cross-validation and collect MSE
gbr_mse_scores = cross_val_score(gbm_model, X, y, cv=5, scoring='neg_mean_squared_error')
rfr_mse_scores = cross_val_score(best_model, X, y, cv=5, scoring='neg_mean_squared_error')

#  scoring is 'neg_mean_squared_error', convert it to positive MSE
gbr_mse_scores = -gbr_mse_scores  # Convert negative MSE to positive
rfr_mse_scores = -rfr_mse_scores  # Convert negative MSE to positive

# Display the MSE scores for each fold
print("Random Forest MSE scores for each fold:", rfr_mse_scores)
```

```
print("Gradient Boosting MSE scores for each fold:", gbr_mse_scores)

# Calculate the mean MSE for each model
mean_rfr_mse = np.mean(rfr_mse_scores)
mean_gbr_mse = np.mean(gbr_mse_scores)
print(f"Random Forest Mean MSE: {mean_rfr_mse}")
print(f"Gradient Boosting Mean MSE: {mean_gbr_mse}")

# Perform the ANOVA test on the MSE scores of both models
f_stat_mse, p_value_mse = stats.f_oneway(rfr_mse_scores, gbr_mse_scores)
# Output the results of the ANOVA test
print(f"MSE - F-Statistic: {f_stat_mse}")
print(f"MSE - P-Value: {p_value_mse}")
if p_value_mse < 0.05:
    print("There are significant differences between the models' performances in terms of MSE.")
else:
    print("No significant differences found in terms of MSE.")
```

9.2 Data Samples

In this subsection of the dissertation a sample of the dataset used is presented. First the sample of the data without weather information is presented and then the dataset without the weather information in presented.

9.2.1 Without Weather

	season	round	circuitRef_x	country	driverRef	grid	position	points	number_y	constructorRef	...	code	points_after_race	pc
0	2018	1	albert_park	Australia	alonso	10	5	10.0	14	mclaren	...	ALO	10.0	
1	2018	1	albert_park	Australia	vandoorne	11	9	2.0	2	mclaren	...	VAN	2.0	
2	2018	1	albert_park	Australia	hamilton	1	2	18.0	44	mercedes	...	HAM	18.0	
3	2018	1	albert_park	Australia	bottas	15	8	4.0	77	mercedes	...	BOT	4.0	
4	2018	1	albert_park	Australia	stroll	13	14	0.0	18	williams	...	STR	0.0	
5	2018	1	albert_park	Australia	leclerc	18	13	0.0	16	sauber	...	LEC	0.0	
6	2018	1	albert_park	Australia	hulkenberg	7	7	6.0	27	renault	...	HUL	6.0	
7	2018	1	albert_park	Australia	sainz	9	10	1.0	55	renault	...	SAI	1.0	
8	2018	1	albert_park	Australia	raikkonen	2	3	15.0	7	ferrari	...	RAI	15.0	
9	2018	1	albert_park	Australia	vettel	3	1	25.0	5	ferrari	...	VET	25.0	
10	2018	1	albert_park	Australia	ricciardo	8	4	12.0	3	red_bull	...	RIC	12.0	
11	2018	1	albert_park	Australia	max_verstappen	4	6	8.0	33	red_bull	...	VER	8.0	
12	2018	1	albert_park	Australia	perez	12	11	0.0	11	force_india	...	PER	0.0	
13	2018	1	albert_park	Australia	ocon	14	12	0.0	31	force_india	...	OCO	0.0	
14	2018	1	albert_park	Australia	brendon_hartley	16	15	0.0	28	toro_rosso	...	HAR	0.0	

pos_after_race	wins_after_race	quali_pos	qualies_best_secs	con_name	con_points_after_race	con_pos_after_race	con_wins_after_race
5	0	11	83.597	McLaren	12.0	4	0
9	0	12	83.853	McLaren	12.0	4	0
2	0	1	81.164	Mercedes	22.0	2	0
8	0	10	82.089	Mercedes	22.0	2	0
14	0	14	84.230	Williams	0.0	8	0
13	0	18	84.636	Sauber	0.0	7	0
7	0	8	83.532	Renault	7.0	5	0
10	0	9	83.061	Renault	7.0	5	0
3	0	2	81.828	Ferrari	40.0	1	1
1	1	3	81.838	Ferrari	40.0	1	1
4	0	5	82.152	Red Bull	20.0	3	0
6	0	4	81.879	Red Bull	20.0	3	0
11	0	13	84.005	Force India	0.0	6	0
12	0	15	84.503	Force India	0.0	6	0
15	0	16	84.532	Toro Rosso	0.0	9	0

9.2.2 With weather

	Time	AirTemp	Humidity	Pressure	TrackTemp	WindDirection	WindSpeed	round	season	raceId	...	code	points_after_race	po
0	0 days 01:27:58.257000	24.2	28.3	997.0	33.6	326	4.2	1	2018	989	...	ALO	10.0	
1	0 days 01:27:58.257000	24.2	28.3	997.0	33.6	326	4.2	1	2018	989	...	VAN	2.0	
2	0 days 01:27:58.257000	24.2	28.3	997.0	33.6	326	4.2	1	2018	989	...	HAM	18.0	
3	0 days 01:27:58.257000	24.2	28.3	997.0	33.6	326	4.2	1	2018	989	...	BOT	4.0	
4	0 days 01:27:58.257000	24.2	28.3	997.0	33.6	326	4.2	1	2018	989	...	STR	0.0	
5	0 days 01:27:58.257000	24.2	28.3	997.0	33.6	326	4.2	1	2018	989	...	LEC	0.0	
6	0 days 01:27:58.257000	24.2	28.3	997.0	33.6	326	4.2	1	2018	989	...	HUL	6.0	
7	0 days 01:27:58.257000	24.2	28.3	997.0	33.6	326	4.2	1	2018	989	...	SAI	1.0	
8	0 days 01:27:58.257000	24.2	28.3	997.0	33.6	326	4.2	1	2018	989	...	RAI	15.0	
9	0 days 01:27:58.257000	24.2	28.3	997.0	33.6	326	4.2	1	2018	989	...	VET	25.0	
10	0 days 01:27:58.257000	24.2	28.3	997.0	33.6	326	4.2	1	2018	989	...	RIC	12.0	
11	0 days 01:27:58.257000	24.2	28.3	997.0	33.6	326	4.2	1	2018	989	...	VER	8.0	
12	0 days 01:27:58.257000	24.2	28.3	997.0	33.6	326	4.2	1	2018	989	...	PER	0.0	
13	0 days 01:27:58.257000	24.2	28.3	997.0	33.6	326	4.2	1	2018	989	...	OCO	0.0	
14	0 days 01:27:58.257000	24.2	28.3	997.0	33.6	326	4.2	1	2018	989	...	HAR	0.0	

pos_after_race	wins_after_race	quali_pos	qualies_best_secs	con_name	con_points_after_race	con_pos_after_race	con_wins_after_race
5	0	11	83.597	McLaren	12.0	4	0
9	0	12	83.853	McLaren	12.0	4	0
2	0	1	81.164	Mercedes	22.0	2	0
8	0	10	82.089	Mercedes	22.0	2	0
14	0	14	84.230	Williams	0.0	8	0
13	0	18	84.636	Sauber	0.0	7	0
7	0	8	83.532	Renault	7.0	5	0
10	0	9	83.061	Renault	7.0	5	0
3	0	2	81.828	Ferrari	40.0	1	1
1	1	3	81.838	Ferrari	40.0	1	1
4	0	5	82.152	Red Bull	20.0	3	0
6	0	4	81.879	Red Bull	20.0	3	0
11	0	13	84.005	Force India	0.0	6	0
12	0	15	84.503	Force India	0.0	6	0
15	0	16	84.532	Toro Rosso	0.0	9	0