CS910 FOUNDATIONS OF DATA ANALYTICS - PROJECT

Driver expertise & Team Strategies: A comprehensive study of Formula 1 World Championships from 1950 to 2022

Abishekraj Vinayagar Gnanasambandam (u2238887)

Department of Comuputer Science, University of Warwick

January 11, 2023

Abstract

Every era in the Formula 1 championship has seen some form of dominance. From the genesis of F1 racing, Ferrari, a key contributor and protege participator, has had a few of their own domination stints over the years. The recent dominating stints from Redbull Racing from 2010 to 2013 and Mercedes from 2014 to 2020. Such subjugating successes to an average spectator's view could be due to changing engine technologies, evolving regulations, driver skills or team strategies. This project aims to conduct a deep dive into the facts through exploratory data analysis on accumulated Formula 1 World Championship data from 1950 to 2022, to understand success patterns in F1 and if Driver's expertise, Team's performance or any other such factors significantly attribute towards domination.

1 Introduction

After claiming the 2019 drivers and constructors championship, Mercedes-AMG famously tweeted [1] "Races are won at the track. Championships are won at the Factory." This was, at that time, Mercedes' sixth consecutive victory both at the driver's and constructor's championships, and they would, in fact, go on to win consecutively for the next two years until just recently, in 2022, Redbull Racing was able to dethrone them. Even though Mercedes has had tremendous success in the turbo-hybrid era of Formula 1 championships, it is clear that the manufacturer's vehicle design and car performance play a vital role in achieving victory.

But is it just purely mechanical supremacy that aids in success? is it the engine? the aerodynamic designs or the conjunction of many such factors. Modern Formula 1 racing has grown to be more data-reliant, and every team has created a strong data analytics foundation to sustain its competitiveness in F1. F1 cars nowadays have a medley of sensors governed by a central ECU or engine control unit, coupled with high sleep radio transmission, which allows teams to monitor and analyse every bit of development happening on the car in real-time to make better decisions during a race. This huge data is also backed by world-class big-data cloud architectures majorly handled by Amazon's AWS and Microsoft's Azure, coupled with powerful machine learning and Data Analytics algorithms help teams predict various outcomes such as fuel usage, engine life span, tire wear, the temperature of engine and tires etc to better device race strategies.

As an ardent enthusiast of Formula 1, I have decided to analyze a series of 14-attribute Formula-1 data set posted to Kaggle [2] accumulating all information on the Formula 1 races, drivers, constructors, qualifying, circuits, lap times, pit stops, championships from 1950 till the latest 2022 season

2 Background

Formula 1, also known as F1 or Formula One, is the highest class of single-seater auto racing sanctioned by the Fédération Internationale de l'Automobile (FIA). It is considered the pinnacle of motorsport and features open-wheel race cars that are designed and built to specific technical regulations. The races, known as Grands Prix, take place on a variety of circuits around the world, including iconic tracks such as Monaco's Circuit de Monaco and Italy's Monza Circuit. The championship is contested annually by teams and drivers, with the team and driver that accumulate the most points over the course of the season being crowned champions.

The point system used in Formula 1 to determine the drivers' and constructors' championships has undergone several changes over the years, but the current system has been in place since 2010. Under the current system, the top ten finishers in each race are awarded points as follows:

Additionally, the driver who sets the fastest lap in the race will be awarded an additional point. In case of a tie in the points, the higher position in the final standing goes to the driver who has more wins through-

Final Position	Points
First	25 points
Second	18 points
Third	15 points
Fourth	12 points
Fifth	10 points
Sixth	8 points
Seventh	6 points
Eigth	4 points
Ninth	2 points
Tenth	1 point

out the season. The point system is intended to reward drivers who consistently finish in the top positions, with more points being awarded for higher finishing positions. This encourages drivers to compete for wins rather than settling for lower points-paying positions. The constructors' championship points are calculated from the total points scored by the two drivers from each team.

The Formula One Drivers' Championship is awarded to the driver who scores the most points over the course of a Formula One World Championship season. Points are awarded for each race, with the winner typically receiving the most points and the following drivers receiving fewer points in descending order. The number of points awarded for a win and other finishing positions can vary depending on the specific race and the rules in place at the time. In addition to the Drivers' Championship, there is also a Constructors' Championship, which is awarded to the team that scores the most points over the course of a season. Points are awarded to teams based on their drivers' finishing positions in each race.

3 The Dataset

Modern motor sports is an absolute amalgam of evolving technology and mechanical prowess. F1 is at the pinnacle of such modern motor sports and is no exception to having an evolutionary technological growth over the years since its inception in the 1950s. This last decade has however seen a monumental shift in the Data-driven nature of strategies and processes in Formula 1. Formula 1 is a data-driven sport: During each race, 120 sensors on each car generate 3 GB of data, and 1,500 data points are generated each second. [3] A current generation Formula 1 car, on average, generates around 3 Terabytes of data in one race. It uses more than 300 sensors and radio communication to transfer data in real-time to the pit wall, the factory, and the FIA. To observe the patterns in performance between drivers and teams and also to classify categorical attributes into machine learning supervised models, we use a 14-attribute Formula-1 data set posted to Kaggle [2] accumulating all information on the Formula 1 races, drivers, constructors, qualifying, circuits, lap times, pit stops, championships from 1950 till the latest 2022 season.

4 Data Analysis

**All relevant code and outputs are maintained <github URL>

4.1 Chronological Analysis

In this section, we explore the data to understand how F1 racing has evolved over time. Here I'm specifically focusing on the turbo-hybrid era from 2005 to the present 2022 season to find patterns in driver and team performances.

Firstly focusing on the cars. It is well established that an apparent increase in car performance is inevitable with increasing technology and improved data analytics in modern motorsports. But the increase in driver skills, increase in the overall quality of teams and recursively revised race regulations make judging the growth of car performance over the years not so straightforward. Thankfully, we have enough raw data to understand this car's performance progress and answer this conundrum. Drawing inspiration from an older historical f1 analysis [4], I was able to script a small R program to help understand car performance through the pretty well-organized f1 championship data set. The main file for this use case is results.csv, coupled with circuits.csv, constructors.csv, drivers.csv, and races.csv, bound by their respective unique IDs. Weighing our decisions on the fastest lap time metric would be a great way to judge cars' performance properly. Summarizing the fastest laps from all circuits between the years 2004 and 2022 generates the following plot.

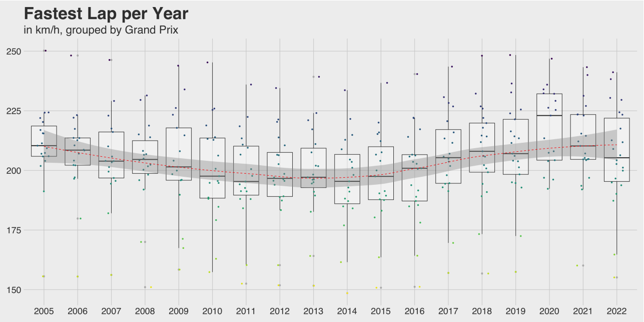


Figure 1: Fastest Lap times per Circuit from 2005 to 2022

We can observe that, in general, there is a decrease in the average fastest lap per year. There is also a sharp overall dip in lap times during the years 2013-2017, which interestingly can be explained by multiple regulation changes [5] during that era and the all-new rise of turbo hybrid v10 engines ditching the old v12 engines of the past. [6]

The second major change over the years is the evolving circuits. I have used a similar strategy to analyse the median lap times on these tracks to help judge the length of a circuit and if, in turn, the races have grown shorter over the years. The following plot depicts the average fastest lap times from 2005 to 2022.

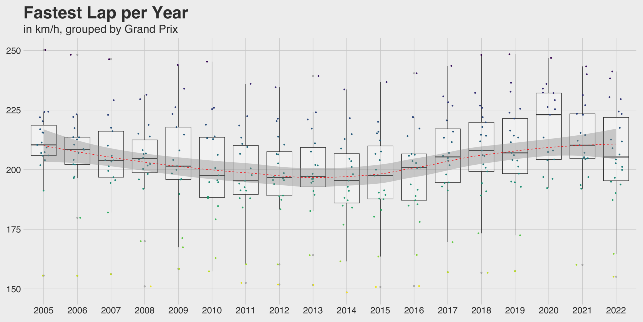


Figure 2: Average Lap times per Circuit from 2005 to 2022

4.2 Driver Analysis

The driver data frame is obtained by joining drivers.csv and driverstandings.csv, calculating the current driver's ages and joining the complete result with the main results.csv and races.csv data in R. Using this complete driver info; we can draw a simple wins bar ranking using ggplot, which is one of the most popular statistical representations in F1. We can clearly see the top 3 drivers being Hamilton, Schumacher and Vettel, with Hamilton recently in the 2022 season crossing over 100 race victories in his career.

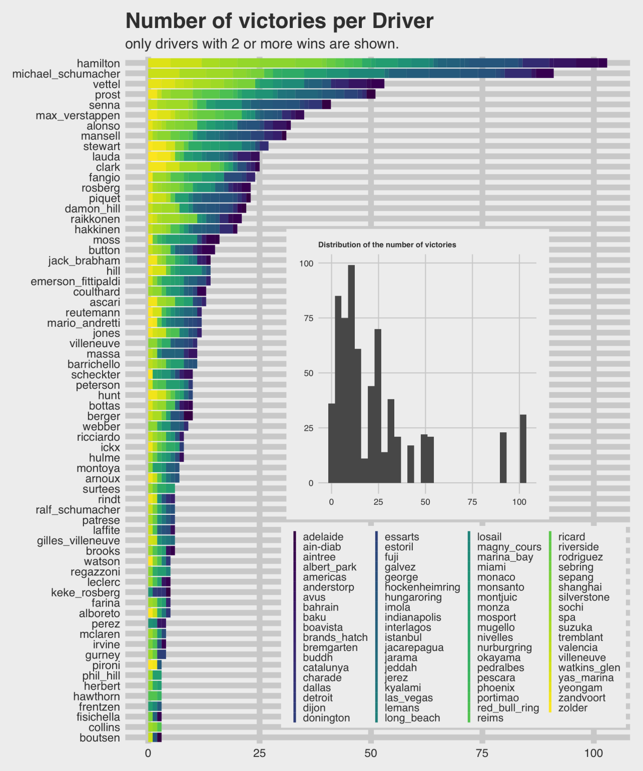


Figure 3: Driver ranking based on total career victories

So far, it is evident that any one single factor does not contribute to winning races in F1. Using regression, we could closely analyse the different degrees of correlations between such factors and try and derive some patterns of success metrics. The driver's data frame joining results, drivers, driver standings and status information can be further merged with the constructor's information or team data to draw a full-fledged data frame that can be used to understand such intricate correlations. We could validate pure driver skill by comparing the qualification result, which would be the grid starting position and the final race result, which is the final position. The ability to maintain the grid position as the final position can be validated as a driver metric. Using the Pandas library of python, I've recreated the same data frame mentioned above and written a small py script to understand better the pattern between the initial grid position and the final finishing position. I've also utilised the sklearn package in python, using which I was able to build a linear regression model to check the correlation between Final Position and Grid Position. Fitting the model with the driver data frame, we're able to generate the below plot and get a correlation coefficient of 0.17179897, which indicates a strong correlation, as expected.

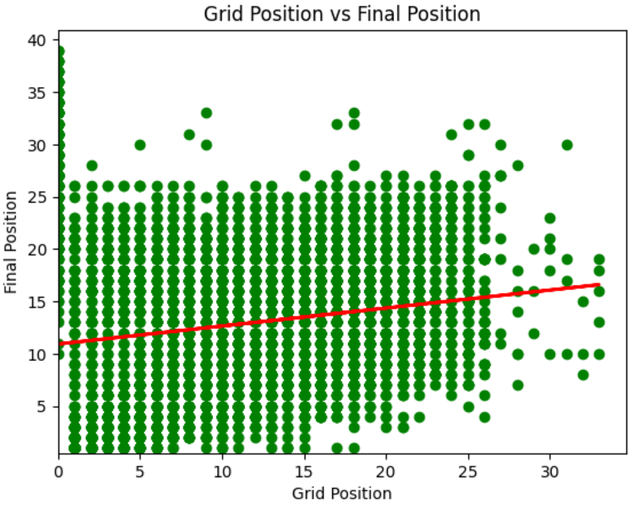


Figure 4: Grid starting position vs Final race position

The lap times over the course of a season are also a very good indicator of the driver's peak performance. Using the same driver's data frame, I was able to plot lap times for the circuit throughout the 2022 season, as depicted in the below figure. It is evident that except for some extreme outliers, such as Valteri Bottas, Esteban Ocon and Carlos Sainz, having lap times over 150 seconds. It can also be observed that such discrepancies happen only during a few tracks such as Villeneuve, Redbull ring and notably the Silverstone British circuit, which is a tough track as it is but the 2022 British grand prix saw a violent crash at the start of the race damaging a lot of cars and drivers' performances[7].

A driver's performance could also be validated through lap times measured lap after lap within a single race as well. Choosing Monaco 2021 Grand Prix as an indicative average case race as the Monaco Grand Prix

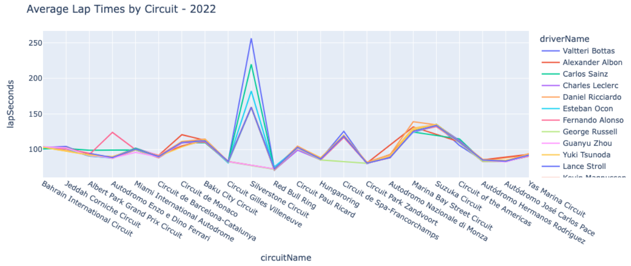


Figure 5: Lap times in each circuit over the 2022 season

is one of the toughest races of the calendar and requires peak driver capability for success. Using the driver's data frame in python, I was able to generate a lap-by-lap analysis of the Monoco GP 2021 and here are the observations. An initial couple of laps have pretty bad lap times for every driver, which stays consistent, maybe due to external physical factors like bringing the engine and tyre up to optimal temperatures. [8] [9]

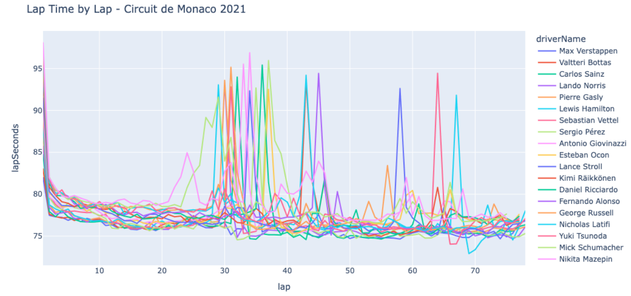


Figure 6: Lap times in each circuit over the 2022 season

4.3 Constructor Analysis

To analyse the constructor data, we need to merge all three constructor datafiles: the constructor.csv, constructorstandings.csv and constructorresults.csv. perform a left join on the unique raceID and constructor ID to produce the constructor data frame. to start with, a great indicator of team performance is lap time. Just like we did in the driver's data frame, we could draw similar plots on the constructor's data frame as well.

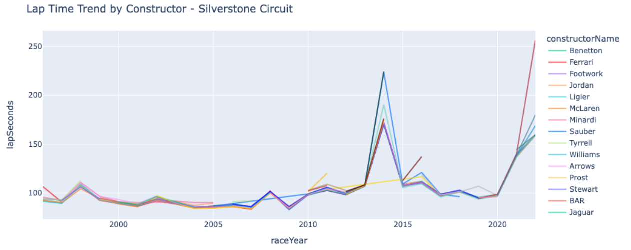


Figure 7: Lap times by Constructor - Silverstone Circuit

The simplest form of constructor performance validation is the number of championships gathered by each team. This can be easily plotted through a ggplot in R script ranking constructors by their victories. From figure 8, it is evident that Ferrari dominates the constructor victories just because of the sheer participation it has had since the inception of F1 racing. Another interesting fact to note is that even though Mercedes has had a very dominating 7-year consecutive sting in this modern era of formula 1 when we consider all the data from the 1950s, Mercedes is only ranked tenth, and Redbull is a little higher at sixth.

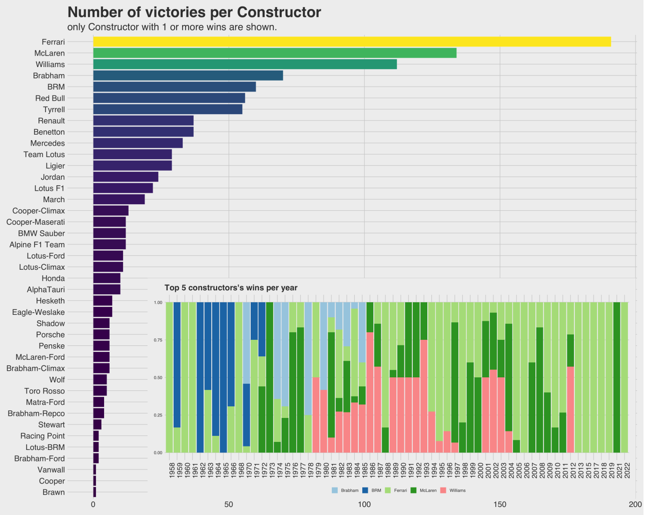


Figure 8: Constructor championships over the years

Furthermore, taking the example of the 2021 Monaco GPs lap-by-lap analysis, it is possible to understand the constructor's contribution towards the performance in the same race by drawing a plot describing lap time distribution throughout the race between different teams. From the below graph, we are able to draw similarities between the previously explored driver statistics for the same race, where the outliers tend to stand out during an initial couple of laps and during the penultimate finishing stages of the race, where the lap times are fairly high. There is definitely also a pattern with team performances when compared to the corresponding drivers as

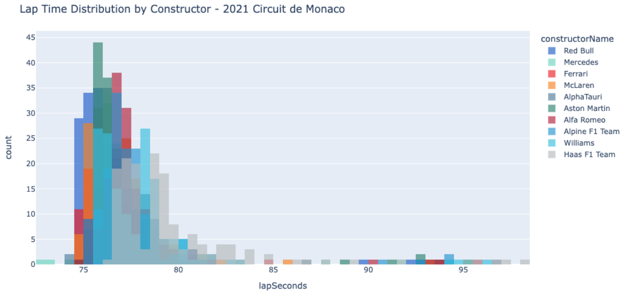


Figure 9: Lap time distribution between Teams in the 2021 Monaco GP

5 Classification

To further investigate the outcome of a race and to validate the performance of teams as well as drivers, we could build classification models and compare their accuracy by fitting the available F1 championship 2015 to 2022 data. Classification in data analytics is a technique used to predict a categorical label (or output variable) based on one or more input variables (or features). The goal of a classification model is to correctly assign new observations to one of the pre-defined classes or categories. This is typically done using supervised learning algorithms, where the model is trained on a labelled dataset that includes both the input features and the corresponding output labels.

5.1 Data Refinement

Using Python and Pandas, NumPy, Sklearn packages, I was able to define an amalgamated data frame consisting of both drivers, races and constructors datasets. We then try and refine data in order to reduce skewness and to check for normalization of all the metadata by estimating their respective probability density functions through Kernel density estimation or KDE

Skewness is the degree of distortion from the symmetrical bell curve or the normal distribution. It measures the lack of symmetry in data distribution. It differentiates extreme values in one versus the other tail. A symmetrical distribution will have a skewness of 0.Skewness is used to check the normality of the data by ranging from -1 to 1.

- -1 –> Left skewed
- 0 –> Normal distribution
- 1 –> Right skewed

So, when is the skewness too much? The rule of thumb seems to be: The data are fairly symmetrical if the skewness is between -0.5 and 0.5. If the skewness is between -1 and -0.5(negatively skewed) or between 0.5 and 1(positively skewed), the data are moderately skewed. If the skewness is less than -1(negatively skewed) or greater than 1(positively skewed), the data are highly skewed.

Outlier Treatment: In the formula 1 data frame, there were some columns are skewed a lot; removal of the outliers will normalize a data bit. Using the below python script, I was able to achieve a pretty acceptable skewness. Still, some columns remained skewed even after outlier removal, and skewness would be normalized while fitting the data into the classification models.

```
# outlier removal
Q1 = df.quantile(0.25)
Q3 = df.quantile(0.75)
IQR = Q3 - Q1
df = df[~((df<(Q1-1.5*IQR)) |
    (df>(Q3+1.5*IQR))).any(axis=1)]
df.head()
```

Using Pandas' dataframe.corr() function, I was able to generate the pairwise correlation of all columns in the refined data frame as follows.

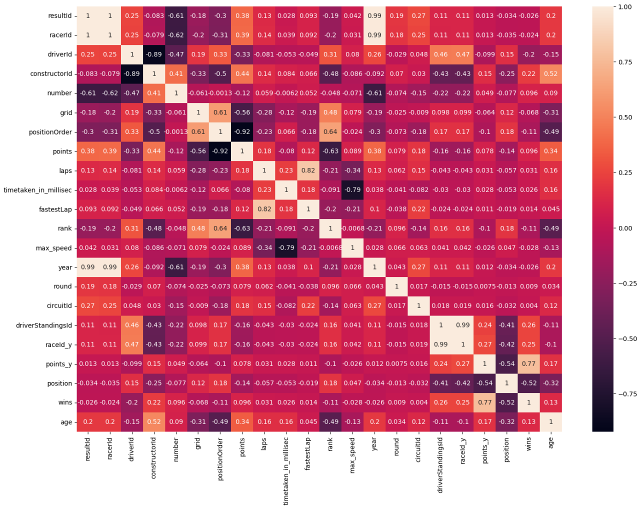


Figure 10: Pairwise correlation matrix

Next, we plot the bivariate distributions using kernel density estimation. A kernel density estimate (KDE) plot is a method for visualizing the distribution of observations in a dataset analogous to a histogram. KDE represents the data using a continuous probability density curve in one or more dimensions. Using seaborn.kdeplot() package function, we generate the following KDE plot for checking the normalization

5.2 Forming models using Supervised ML Algorithms

We proceed by encoding the data frame's metadata from Unicode strings to a string of bytes to help better when fitting to supervised algorithms. Here I'm using LabelEncoding from sklearn.preprocessing package in Python. Using the label encoder, I was able to separate the categorical columns from the numerical columns in the complete data frame and only encode the categorical ones.

Using a train-test split of 70 to 30 per cent, we start off with our first classifying model, the decision tree classifier. Using DecisionTreeClassifier(max_depth=5, random_state=1234) function from the sklearn.tree package in Python, I was able to fit the data frame and generate the following decision tree workflow as depicted in Figure. 12.

We then proceed to model the data using the Logistic Regression, Decision Tree Classifier, Random Forest Classifier, KNeighbors Classifier, Gaussian NB,

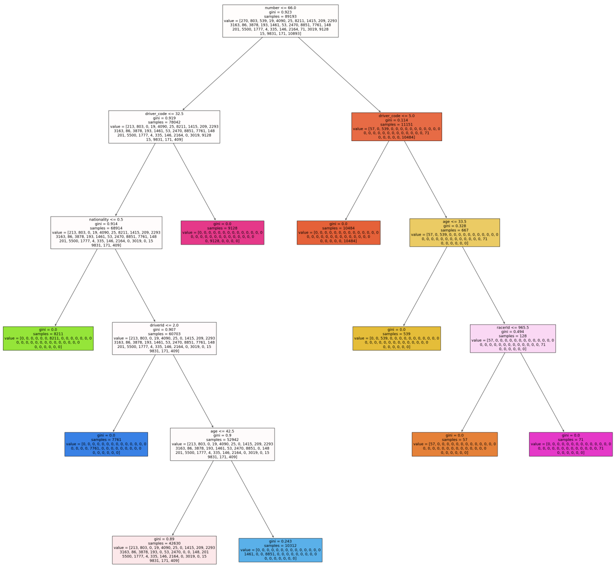
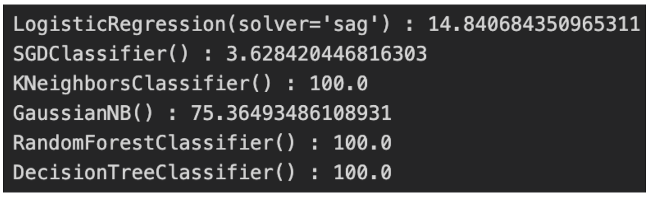


Figure 11: The Decision tree classifier



```
LogisticRegression(solver='sag') : 14.840684350965311
SGDClassifier() : 3.628420446816303
KNeighborsClassifier() : 100.0
GaussianNB() : 75.36493486108931
RandomForestClassifier() : 100.0
DecisionTreeClassifier() : 100.0
```

Figure 12: Accuracy measurement between models

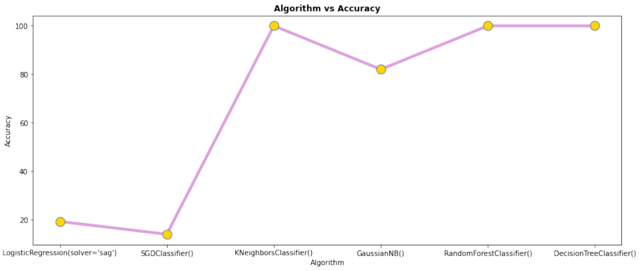
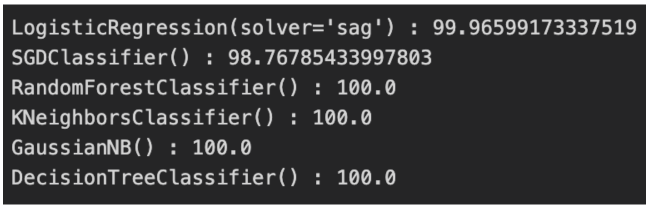


Figure 13: Algorithm vs Accuracy

and SGD Classifier classification algorithms and understand their respective accuracy percentages.

When we try to plot the obtained accuracy values against the algorithms, we can observe that the accuracy of the basic (logistic) algorithm and the SGD are not as good as expected. The primary reason for the drastic inaccuracy is the data being skewed or denormalized. This problem can be fixed using scaling.

As discussed, we would try different scaling techniques in an effort to help with skewed or de-normalized data to maximize accuracy. Firstly we try Min-Max Scaler. For each value in a feature, MinMaxScaler subtracts the minimum value in the feature and then divides it by the range. The range is the difference between the original maximum and the original minimum. MinMaxScaler preserves the shape of the original distribution. As you can see, the accuracy is getting high for Logistic Regression and SGDClassifier; both the algorithms are performing well proving the importance of normalizing the data.



```
LogisticRegression(solver='sag') : 99.96599173337519
SGDClassifier() : 98.76785433997803
RandomForestClassifier() : 100.0
KNeighborsClassifier() : 100.0
GaussianNB() : 100.0
DecisionTreeClassifier() : 100.0
```

Figure 14: MinMaxScaler: Accuracy measurement between models

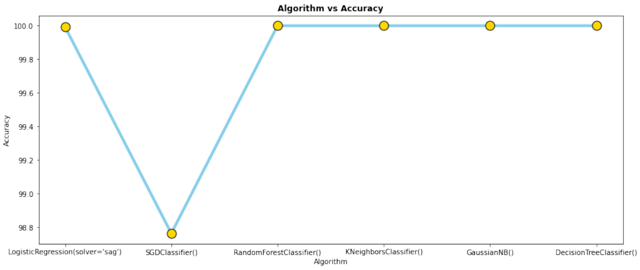
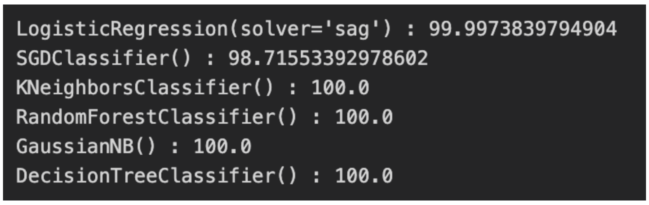


Figure 15: MinMaxScaler: Algorithm vs Accuracy

StandardScaler removes the mean and scales each feature/variable to unit variance. This operation is performed feature-wise in an independent way. StandardScaler can be influenced by outliers (if they exist in the dataset) since it involves the estimation of the empirical mean and standard deviation of each feature.



```
LogisticRegression(solver='sag') : 99.9973839794904
SGDClassifier() : 98.71553392978602
KNeighborsClassifier() : 100.0
RandomForestClassifier() : 100.0
GaussianNB() : 100.0
DecisionTreeClassifier() : 100.0
```

Figure 16: Standard Scaler: Accuracy measurement between models

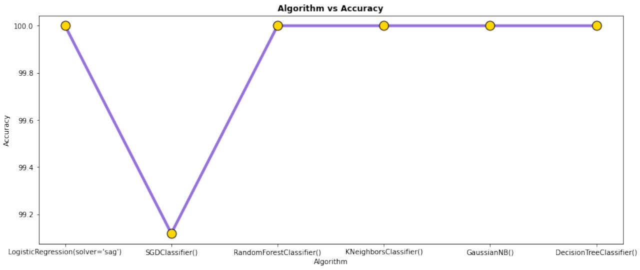
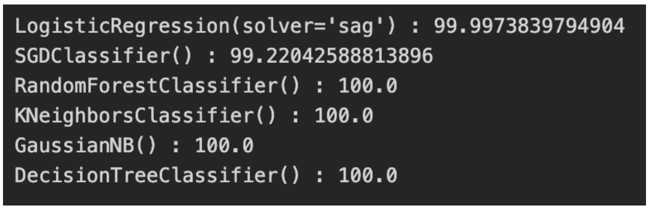


Figure 17: Standard Scaler: Algorithm vs Accuracy

RobustScaler scales features using statistics that are robust to outliers. This Scaler removes the median and scales the data according to the quantile range (defaults to IQR: Interquartile Range). The IQR is the range between the 1st quartile (25th quantile) and the 3rd quartile (75th quantile). Thus after the robust scaler attempt, we have got a good accuracy score of 100% in all algorithms except SGDClassifier with a least of 99.2% for RobustScaler. Even the least (SGDClassifier) is considered as very good accuracy.



```
LogisticRegression(solver='sag') : 99.9973839794904
SGDClassifier() : 99.22042588813896
RandomForestClassifier() : 100.0
KNeighborsClassifier() : 100.0
GaussianNB() : 100.0
DecisionTreeClassifier() : 100.0
```

Figure 18: Robust Scaler: Accuracy measurement between models

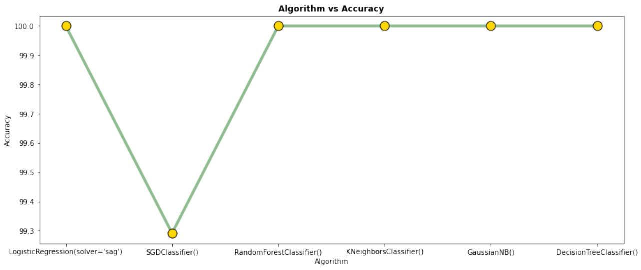


Figure 19: Robust Scaler: Algorithm vs Accuracy

6 Conclusions

The aim of this project was to analytically cirque the historical formula 1 dataset available in order to draw meaningful patterns of performance and success which this paper has been successful in achieving. We were able to establish the historical coherence with formula 1's evolution especially with improvement in car performance over the years and the evolution of circuits from the 1950s to present. Driver performance was also explored and a ranking based on accumulated victories was easy to draw.

Through linear regression, we were able to draw a strong positive correlation between the driver's qualifying position or grid position to the final race results suggesting that the higher the driver qualifies, the more likely he is to win or score on the podium. Lap times dataset was also thoroughly utilized to gather a medeley of information on both driver skills as well as team strategies. Lap times over the years were considered to gather a wholistic pattern where as the 2021 Monaco GP proved as an excellent example to establish a deep understanding between lap times and real world results.

Finally we were able to refine the data frame to help reduce skews and de-normalities in order to fit a 30% train-test data into multiple classification supervised models. The accuracy measures of each of these models were compared through plots and using scaling, a better accuracy of 99.2 percent least was achieved. This effort concludes an exploratory analysis into the forumula 1 dataset and has given us some meaningful insights on various performance metrics.

7 Extentions

Possible extensions to this project would be:

- ELO rating algorithm to rank performance of driver and constructor based on relative skill levels or strong oppositions
- Apply regression models to help predict fuel consumption, tire wear to better predict optimal pit stop strategies
- Utilise clustering models to help group drivers and teams of similar perfomance levels to better rate at their respective skill leagues
- Utilise clustering models to group tracks by difficulties and predict racing conditions based on historical track data

References

[1] Mercedes-AMG. Mercedes-AMG PETRONAS F1 Team[@MercedesAMGF1] via Twitter on November 8, 2019: Races are won at the track. Championships are won at the Factory. [Video attached] [Tweet];. Available from: https://twitter.com/mercedesamgf1/status/1192891850560020486.

[2] Vopani. Formula 1 World Championship (1950 - 2022);. Available from: https://www.kaggle.com/datasets/rohanrao/formula-1-world-championship-1950-2020.

[3] AWS. Formula 1 Case Study by Amazon Web Services;. Available from: https://aws.amazon.com/solutions/case-studies/formula-one/.

[4] BOUCHET J. Formula 1 World Championship 2005-2017;. Available from: https://www.kaggle.com/code/jonathanbouchet/f1-data-analysis/report.

[5] Wikipedia. History of Formula One regulations;. Available from: https://en.wikipedia.org/wiki/History_of_Formula_One_regulations.

[6] Wikipedia. Formula One engines;. Available from: https://en.wikipedia.org/wiki/Formula_One_engines.

[7] 2022 British Grand Prix: Zhou Guanyu conscious but taken away in ambulance following huge crash at race start;. Available from: https://www.formula1.com/en/racing/2022/Great_Britain.html.

[8] Tremlett AJ, Limebeer D. Optimal tyre usage for a Formula One car. Vehicle System Dynamics. 2016 08;54:1-26.

[9] Wright P, Matthews T. Formula 1 technology. SAE International; 2001.