Exploring applications of Machine Learning and Artificial Intelligence in Formula One

LM118 – Bachelor of Engineering in Electronic and Computer Engineering

Project Final Report

Laurence Hearne: Laurenceh19@gmail.com

Associate Prof Patrick Denny and Prof Pepijn Van de Ven

Abstract

Formula One is the highest level of motorsport in the world, with 20 drivers and 10 teams all competing for the World Drivers and Constructors Championships. The desire to win is high, driven by the desire to be the best teams and drivers in the world, as well as reap the financial rewards that come with doing so.

For this project, the applications of artificial intelligence and machine learning in Formula One were explored. Formula One race strategy was found to be an area where machine learning and artificial intelligence could be applied to drive better performance in the sport. As per the regulations of the sport, every driver in a Formula One race must use at least two different compounds of tyre during each race, thus at least one pitstop is required by each driver. Pitting for new tyres will mean the drivers will be much faster on track and so, drivers will often undertake more than the one mandatory pitstop during a race. The lap on which a driver decides to pit is a crucial decision, as poor decisions in this area can lead to valuable race wins and points being lost.

As it stands, simulations are done by the teams prior to the race in order to plan what the optimal lap to pit is. Simulations are limited, however; in that they can fail to capture the dynamic nature of the race. For this reason, machine learning and artificial intelligence were selected as tools that could be applied. Having contacted several Formula One engineers, it was confirmed that ML and AI are not used in the decision-making process for race strategy at present.

In this project, a machine learning model was built using TensorFlow, which had sufficient accuracy for the model to be considered as a tool for Formula One strategists to use as part of their toolbox while making decisions. The model was trained on five years of race data, with one year of data reserved for evaluation and testing. The model successfully handled several common situations encountered during a Formula One race such as undercuts and

safety cars. The model also tends to work best for top drivers which is a desirable characteristic.

This model produced impressive results and is certainly an area where Formula One teams could apply Machine Learning and Artificial Intelligence. With the vast amounts of extra data available to a Formula One team, as well as the extra resources, a Formula One team could further improve the model, further boosting its usefulness.

Declaration

This report is presented in part fulfilment of the requirements for the Bachelor of Engineering in Electronic and Computer Engineering Final Year Project.

It is entirely my own work and has not been submitted to any other University or Higher Education Institution or for any other academic award within the University of Limerick.

Where there has been made use of work of other people it has been fully acknowledged and referenced.

Name	Laurence Hearne
Signature	*Laurence Hearne*
Date	02/04/24

Table of Contents

List of Figures

List of Tables

Table of Terminology

Term	Explanation
Pitlane	The pitlane is a designated area into which drivers can pull into to get off the track. Team garages are located along the pitlane. While in the pitlane, drivers must adhere to a speed limit of either 80km/h or 60km/h, depending on the track.
Pitstop	A pitstop occurs when a driver pulls into the pitlane during a track session in which they change tyres, repair car damage, or make adjustments to the car.
Stint	Refers to the period between pitstops during a race.
Undercut	A strategic move where one driver pits before another to attempt to overtake. This will be discussed further in background theory.
Overcut	A strategic move where one driver pits after another to attempt to overtake. This will be discussed further in background theory.
Box	Another word used to describe a pitstop.
Pit Window	A period of a race in which it would be expected for a driver to pit.
Safety Car	Physical car brought out to limit the speed of drivers on track by leading them around. Used when track Marshalls must enter the

	track to clean up debris after an accident generally. Cars will bunch up behind during a safety car.
Virtual Safety Car	Similar to a Safety car except a physical car is not used to limit drivers' speed, instead, drivers are mandated to follow a set minimum lap time. Cars will not bunch up during virtual safety car.
Yellow Flags	This signifies that there is a hazard on track and driver should slow down and drive with caution. If the hazard cannot be resolved under yellow flags, a Safety car or Virtual Safety car will be deployed.
Green Flags	Signifies that the track is clear and there are no issues on track. Shown under normal racing conditions.
Points	Points are awarded to the top ten finishers of a Formula One Grand Prix. The winner of the race receives 25 points, with 18, 15, 12, 10, 8, 6, 4, 2 and 1 points awarded between second and tenth place respectively. Points are awarded to the driver and their team.
Drivers' Championship	Awarded to the driver with the most points at the end of the F1 season.
Constructors Championship	Awarded to the team with the most points at the end of the season.
Backend driver	Used to describe a driver generally close to last place in races.

FIA	The Federation Internationale de l'Automobile (FIA) is the governing body for many international racing series, including Formula 1. They set the regulations for car design and the races.
Compound	Material used to make Formula One tyres. Softer compounds will offer a higher peak grip at the cost of reduced longevity. Harder compounds will have a lower peak grip but reduced wear.

Introduction

In this project the area of Machine Learning was applied to Formula One race strategy, to assess just how applicable ML and AI are in Formula One. With the vast amounts of data gathered by Formula One teams during each race, and the availability of high-performance computing, Formula One is ripe for the application of AI and ML. There are a great number of areas in Formula One that could benefit from the application of ML and AI, but race strategy was chosen to be the area of focus for this project. Race strategy was chosen due to some recent high-profile mistakes by some Formula One teams and the availability of suitable public data for model training.

For all the complexities of Formula One race strategy, the problem faced by an F1 strategist is a simple question: “When do I call my drivers in for a pitstop?”. The decision-making behind this question involves simulations, data analysis and human intellect. Having contacted several F1 engineers, it was confirmed that ML and AI tools are currently not part of this decision-making toolbox, and with this, the premise of this project was born.

A Machine Learning model was successfully built which produced a probability of a given driver pitting for each lap of the race. The model was trained, evaluated and tested using race data from 2018 to 2023. On separate test data, the model displayed an ability to identify various common race situations and provide a probabilistic output which would be useful for a Formula One strategist to use as part of their decision-making toolbox.

In this report the background theory of the model will be explained, the methodology for developing it, and the model’s performance, will be discussed. As part of analysing the model's results, the model’s performance for several races will be presented, in the same way that a Formula One strategist would use the model. Finally, as part of the conclusion some thoughts will be given on the usefulness of the model developed and future work that could be done to improve the model.

Formula One Race Strategy

Formula One is the highest level of motorsport in the world, with 20 drivers and 10 teams all competing for the World Drivers and Constructors Championships. In each race, it is an FIA regulation that each driver use at least two different tyre compounds during a race[1]. This means that at least one pitstop must be made by each driver during each race. Drivers will often undertake more than just the one mandatory pitstop however. Although time will be lost during a pitstop, usually around 20 seconds, drivers will be much quicker on the new tyres after the pitstop. This is because the tyres used in F1 degrade and offer less grip as the number of laps completed on the tyre increases [2]. There are three possible tyre compounds at each Grand Prix, Soft, Medium and Hard. The Soft will offer the highest peak grip but will degrade the most quickly, with the Hard offering the lowest peak grip but degrading the slowest and the medium falls into the middle of the Soft and Hard. Tyre changes are the most common reason for a pitstop, however pitstops are made to repair damage and to make adjustments to the car occasionally. For this project, only tyre change pitstops are being considered.

As mentioned earlier, the problem for a Formula One race strategist is in answering the question, “When do I call my drivers in for a pitstop?”. Making the correct decision regarding this question can have a major impact on the outcome of the race, with the competitive nature of F1, it can often be the difference between winning and losing races. The Ferrari F1 team during the 2022 F1 season can serve as a good case study on just how important F1 race strategy is[3]. During the 2022 season, Ferrari lost numerous race wins for their drivers through poor strategy decisions. At the 2022 British Grand Prix, Leclerc was leading in the late stages of the race, when a safety car came out. The time lost making a pitstop under a safety car is much less than under normal race conditions. Furthermore, under a safety car, all cars are bunched up on track, so a driver is vulnerable to attack immediately once the race restarts. While a large portion of the teams decided to pit their drivers under the safety car, Ferrari decided to keep Charles Leclerc out on 14-lap old hard tyres. With most drivers on soft tyres, Leclerc was a sitting duck once the race restarted and ended up finishing fourth. At the Hungarian Grand Prix, Ferrari decided to pit Leclerc early from the lead on the medium tyre to switch to the Hard compound tyre. On that weekend, teams had largely decided that the Hard tyre was too slow to use in the race. This had been proven true when Alpine pitted for

the Hard tyre earlier in the race, on the tyre they had poor pace. The same happened to Ferrari when they decided to fit the Hard tyre, this pitstop mistiming and poor tyre choice meant Leclerc went from 1st to finishing 6th. These strategy mistakes cost Charles Leclerc crucial points in his Championship battle with Max Verstappen.

As it stands, race strategy decisions are based on simulations and data analysis[4]. Approximately six weeks before a race, teams will begin to gather data on a particular race. All this gathered data will be fed into a simulation program which will determine the optimal strategy for a race. Hundreds of variables are considered in these simulations including tyre wear, historical race data, and weather conditions. With these simulations complete, several strategic options are formed, these are often broken down into plan A, plan B and so on. Due to the dynamic nature of races, these strategies are often just used as a starting point, with the strategy being constantly reconsidered live during a race. This is where the ML model built in this project may be of use. The model's output of the probability of a driver making a pitstop during a race could be used by F1 strategists to inform their decision-making for their drivers but to also gauge what the competition is likely to do. Having spoken to several F1 engineers it was confirmed that AI and ML are currently not used for F1 strategy.

Two strategic tactics are often employed by F1 strategists in order to overtake and defend against other drivers. Understanding these tactics is paramount to understanding F1's strategy due to their widespread use.

The Undercut

The first of these is called the undercut. This is where one driver is behind another, within approximately two seconds, and the pit window is approaching. It is common in circumstances where the two cars are close on pure pace, meaning on-track overtaking is difficult. The pit window is a period of laps during which the majority of pit stops occur. For this explanation, the driver ahead will be referred to as Driver A and the driver behind as Driver B. So, in an undercut, driver B will pit earlier than Driver A, and swap their worn tyres for new fresh ones. If an initial gap of 1.5 seconds is assumed, and a pitlane loss time of 20 seconds, driver B is now 21.5 seconds behind driver A. However, on a new set of tyres driver

B is now travelling much faster than driver A, potentially in the region of one second a lap. This means that within two laps, driver B will have reduced the gap to 19.5 seconds, and so when Driver A pits they will come out behind Driver B. Driver A will have slightly fresher tyres than Driver B, however, if the cars are close on performance and the track is one at which it is difficult to overtake, Driver B has a good chance of remaining ahead. The driver ahead, Driver A, can protect against this tactic by pitting on the lap after Driver B. This will nullify Driver B's pace advantage before they have the chance to gain enough time on Driver A. This is known as "covering" the undercut. It will be important for the model to be able to successfully cover undercut attempts as they commonly occur in races.

The Overcut

An overcut occurs in the same situation when two cars are within roughly two seconds and the pit window is approaching. However, in this circumstance, driver B would wait for Driver A to pit and then would push on and try to lap faster than Driver A, even though Driver A is on new tyres. This strategy only works if driver B has driven conservatively on their tyres and so has a lot of remaining grip. This strategy may also work if conditions are cold and so driver A will not be able to get the maximum performance out of their new tyres as they wait for them to warm up. The overcut is generally not as popular and effective as the undercut but is still used when the circumstances are right.

Related Work

A review of work related to this project was conducted in order to draw from existing research done on the topic. In the following piece, some of the interesting papers found will be discussed.

As mentioned previously, race simulations are performed in order to determine the most optimal strategies prior to the race, the work of Sulsters[5] was reviewed to better understand current methods. In this paper, mathematical and statistical methods were used to determine the optimum race strategies prior to a race.

A virtual strategy engineer was created using Artificial Neural Networks [6]. This project saw the use of race data between 2014 and 2019 to train two neural networks, one to decide when to pit a driver and one to decide on what tyres to fit. Several interesting filters were applied to their dataset as part of data pre-processing, the application of such filters will be explored as part of this project. A particularly interesting one was only removing laps of drivers who finished outside the top ten, the reasoning here is that the model would only be trained on drivers with good strategy. This does however reduce the dataset size available for training greatly.

An Artificial Neural Network was built to predict the finishing positions of drivers during the 2017 F1 season[7]. This paper by Stoppels focussed on final finishing position prediction, however detail on ANNs was very inciteful.

The work of Tulabandhula [8], had a similar objective to the one of this report, however the predictive model built is for the American racing series NASCAR. An interesting point mentioned in this paper relates to the artificial jumps in lap times caused by pitstops. This effect will also be present in the dataset available for this project. For this reason, the use of LapTime as a feature in the model was avoided in the project also.

Marinaro applied reinforcement learning to plan F1 race strategies [9], although the decision on whether or not to make a pitstop is not considered.

Outside of the motorsport domain, parallels can be drawn to the prediction of machinery failures. In the work of Papathanasiou, survival analysis was applied to the problem [10], and the work on feature selection in this paper was particularly interesting. In this paper a Random Survival Forest model was used, this model may be applied to the problem at hand in this project as predicting a pitstop and machine failure have similarities. This topic will be discussed further later in this report.

There has also been some related work done in other sporting domains, Guttag built a model for in-game decisions for major league baseball [11]. Although centred around baseball, this paper still makes for interesting reading in the context of this project.

To better understand the race strategy decision process used by Formula One teams at present a two-prong approach was taken. Material from the Mercedes AMG Petronas Formula One team was reviewed on what goes into Formula One race strategy decisions [12]. Secondly, I reached out to several engineers across multiple Formula One teams to get a first-hand perspective of the current strategy decision-making process. They confirmed that the race strategy decision process is currently a combination of the use of simulation tools as well as human domain knowledge.

Methodology

With the goal being to develop a model that can be used to aid in Formula One race strategy decisions, the first step was to plan. This began by creating a Gantt chart, which can be found in Appendix A. This is a risk containment tool and will ensure that all activities and deliverables can be completed before deadlines. Weekly meetings were set up with my project supervisors Assoc Prof Dr Patrick Denny and Prof Pepijn Van de Ven, meeting notes for each of these meetings can be found in Appendix D. As an additional organisational tool, an Open Issues List document was set up, as suggested by Patrick. This is an extremely helpful tool in aiding productivity and allowing my supervisors to track progress. A sample of the Open Issues List can be found in Appendix C. Having reviewed related work to this project the next step was to gather data for training a model.

Data Acquisition

Data for the project has been gathered from a publicly available source. The Python library FastF1[13] was used to acquire all race lap data for this project. Fastf1 is a free-to-use open-source library which allows access to data for all practice, qualifying and race sessions between 2018 and the present. All data is provided in the form of pandas' data frames[14]. Fastf1 sources its data from the official Formula One data feed, although the library itself is not in any way associated with Formula One. The library is very well documented and there is a large active GitHub community surrounding the library, I made use of this community extensively when I was learning how to use the functions of the library.

To access data from the library, first, a session object must be loaded. This is seen here in the first line of the code snippet of Figure 1. The fastf1 function '`get_session()`' takes three inputs. First the year the event we are trying to access data for took place. Second the event that year we are trying to access can be expressed as a round number or a geographical location of the race. The third input specifies which session from the event we are trying to access, in the case here 'r' specifies that we are accessing the data from the race. This line will

load a 'Session Object' which will contain general information on the event including the date it took place, a list of drivers that took part in the event and session results. To access lap-by-lap data, as is required in this case, the second line of the below code snippet must be run. This loads data for every lap of every driver for that session. The '`load()`' function takes two inputs. First, it must be specified that we want to load all laps and secondly, we specify that we want to also load weather data for the session.

```
raceData = fastf1.get_session(year, round, 'r')
raceData.load(laps=True, weather=True)
```

Figure 1: Snippet of code used to access data from FastF1.

The lap data and weather data are acquired as separate data frames and then combined. A script was created that loops through every race of a season, saving the pandas' data frame for each race as an Excel sheet. From the 2021 season, Formula One introduced Sprint Races. These are additional shortened races that run on the Saturdays of some race weekends, due to them being shorter, there are generally no pitstops made during these races, for this reason, they were not included in the dataset.

When initially acquiring the data, the intention was to use the lap times of each driver to calculate the intervals between drivers. This would work by calculating the cumulative lap times of each driver at the end of each lap. This would be the race time of each driver. The leader of the race would then have the lowest race time naturally so subtracting each following driver race time from the driver ahead would give the gap between each driver. However, when it came to performing this calculation there were some issues. The first of these was on laps where there were red flags or safety cars. When the cars come back out on track after a red flag, the lap times of the drivers may not be reliable as all drivers do not re-enter the track at the same time. The lap times will be reliable again once the race officially restarts, but for the purposes of calculating the gaps between drivers, this makes the calculation inaccurate. Similar problems can occur when there is a safety car period. Furthermore, there can be issues where drivers crash or stop on track.

Due to these issues, another solution was sought. The fastf1 core python library that was used so far in the project does not have information on the gaps between drivers. A second library called the Fastf1 API was found which does offer data on the gaps between drivers. The gap information from this library was sampled multiple times per lap and so had to be sampled once per lap instead to allow it to be merged with the previous dataset. Furthermore, as a useful additional feature, this library offered data on the number of pitstops completed by a driver during a race. In total then, three new features were acquired from this dataset. These were, Gap to leader in seconds, Gap to car ahead in seconds and the number of pitstops undertaken by a driver.

Data Cleaning

The acquired raw data needed some cleaning to fix some to make it suitable for EDA and enable data to be understood to decide on features and a model to use. Following an investigation of the dataset, and a review of data pre-processing best practices[15], the following actions were taken:

Lap times are in a *pandas datetime format*, which when exported to a CSV file are not easily understood, hence these times should be converted to milliseconds.

The features 'PitInTime' and 'PitOutTime' represent the times the driver entered or left the pits. These columns contain a substantial number of missing values, as by nature drivers will not be entering and leaving the pitlane each lap. Hence, it was decided to create a new feature called 'InPits', this feature would answer the question, 'Is the diver in the pits this lap?'.

Sector1, Sector2 and Sector 3 times also converted to milliseconds for the same reason as Lap time conversion.

Wet races were removed. This decision was taken due to the complex nature of wet races, strategic decisions during wet races are largely based on driver feedback and the amount of rainfall. Heilmeier in his work also took this action[6]. This was further confirmed by

contacting a data scientist working at the Redbull Formula One team. For this reason, it was decided the model would not be relevant for wet races. For this project, a wet race is defined as any race where either the Intermediate or Wet tyres are used.

Missing data was addressed as part of the pre-processing process. Firstly, it was assessed how much missing data there was for each feature. For features with a missing data percentage of less than 2%, Spline interpolation[16] was applied to fill in missing values. It was applied to the following features, SpeedI1, SpeedI2, SpeedFL, SpeedST, Sector1time, Sector2time and sector3time. Spline interpolation was chosen because it produces a smooth fit to the data, this is suitable here because each data point is likely to be similar to neighbouring data.

Several features were then removed from the data. The removed features along with the reason for removal are seen below in Table 1.

Table 1: Showing removed features and reasons for removal.

Feature	Reason for removal
Lap Start Date	Seasonality is reflected in the weather data that is also part of the dataset.
Deleted reason	Missing data exceeded 99%.
Lap Start time	Lap start time is not required when the time to complete a lap and Lap completion time is also featured.
Timestamp for when sectors 1,2 and 3 were completed	Lap start time as well as time to complete each sector are already features, hence feature is not necessary as it can be calculated using others.
Driver number	Serves the same purpose as the driver's name.
FastF1Generated	True when Data has been manually added by Fastf1 as opposed to being sourced from the Formula One official DataStream, False for 99.9% of laps.

In the dataset for each race, the tyres used by each driver are described as soft, medium, and hard. What compound specifically the soft, medium or hard tyre, depends on the track as they are set to be different compounds by Pirelli for each race[17]. The hard tyre for the 2023 Bahrain Grand Prix was the C1 compound, whereas, for the Jeddah Grand Prix, the hard was the C2 compound. Furthermore, the naming and number of the tyre compounds, change across the seasons included in the dataset. This is summarised in Table 2. The solution to this problem was to create a new tyre naming scheme that could be applied across all years of the data. This new scale is seen in the last row. A Python script was created to convert the soft, medium, and hard tyre descriptions to the correct value in the new scale for each track and year.

Table 2: Tyre compounds across years of the dataset.

Year	Softest						Hardest
2018	HYPERSOFT	ULTRASOFT	SUPERSOFT	SOFT	MEDIUM		HARD
2019	C5	C4		C3	C2		C1
2020	C5	C4		C3	C2		C1
2021	C5	C4		C3	C2		C1
2022	C5	C4		C3	C2		C1
2023	C5	C4		C3	C2	C1	C0
New Scale	SSSS	SSS	SS	S	H	HH	HHH

Clustering

In the dataset, there are thirty-one tracks at which races were held. Some tracks were used in only one season. To allow the model to build up an understanding of similar tracks in terms of ability to overtake, tyre stress and track surface, clustering was used to relate similar tracks to each other. This would allow the model to build relationships between similar tracks. Pirelli, the tyre manufacturer for all Formula One teams, publishes data on the characteristics of each track type. [25]. To determine the number of clusters needed an elbow plot was created, see Figure 2 Additionally as is seen in Figure 3 through Figure 7, silhouette plots were created using sci-kit-learn[18] in Python. Observing the elbow plot, it appears that 5 clusters is the optimal number. From the Silhouette plots, we want the silhouettes to appear 'natural' as defined in the work of Rousseeuw [19]. In this, we want to avoid clusters appearing below the average silhouette score as well as avoid weak cluster silhouettes. Considering the insight offered by the silhouette plots and the elbow plot, five clusters appears to be the optimal number.

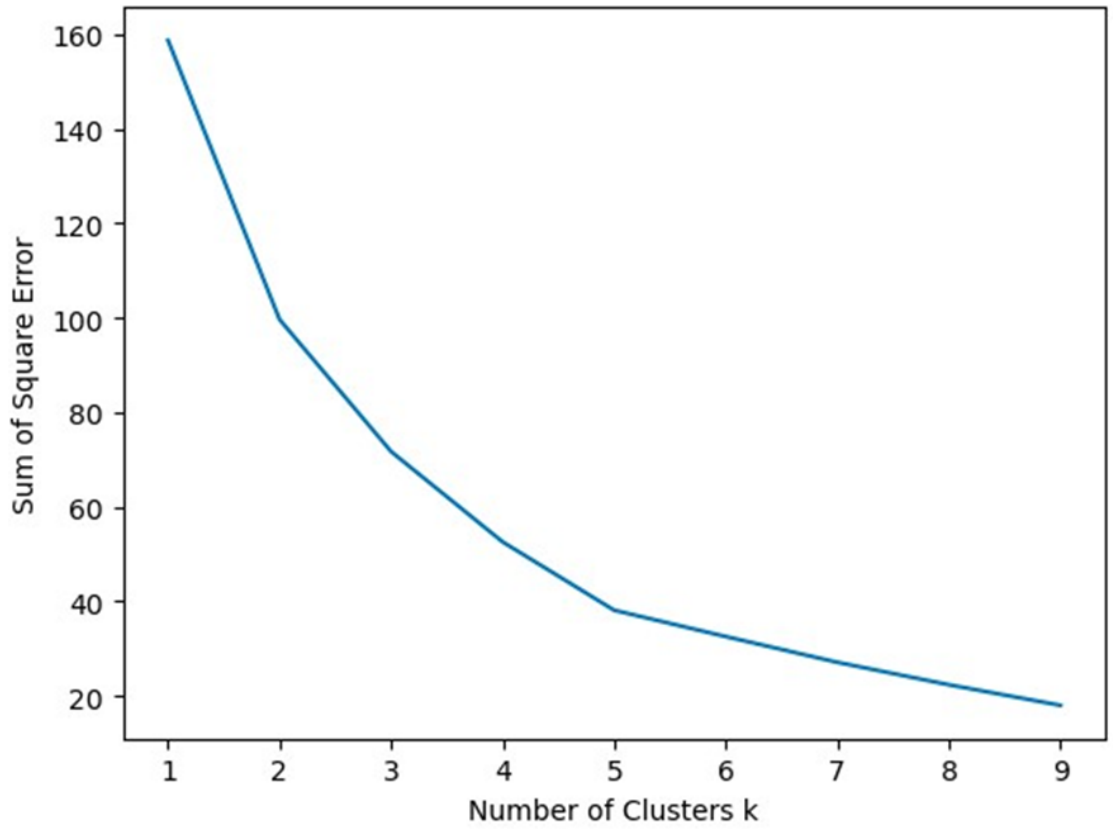


Figure 2: Elbow plot of sum of square error across values of k.

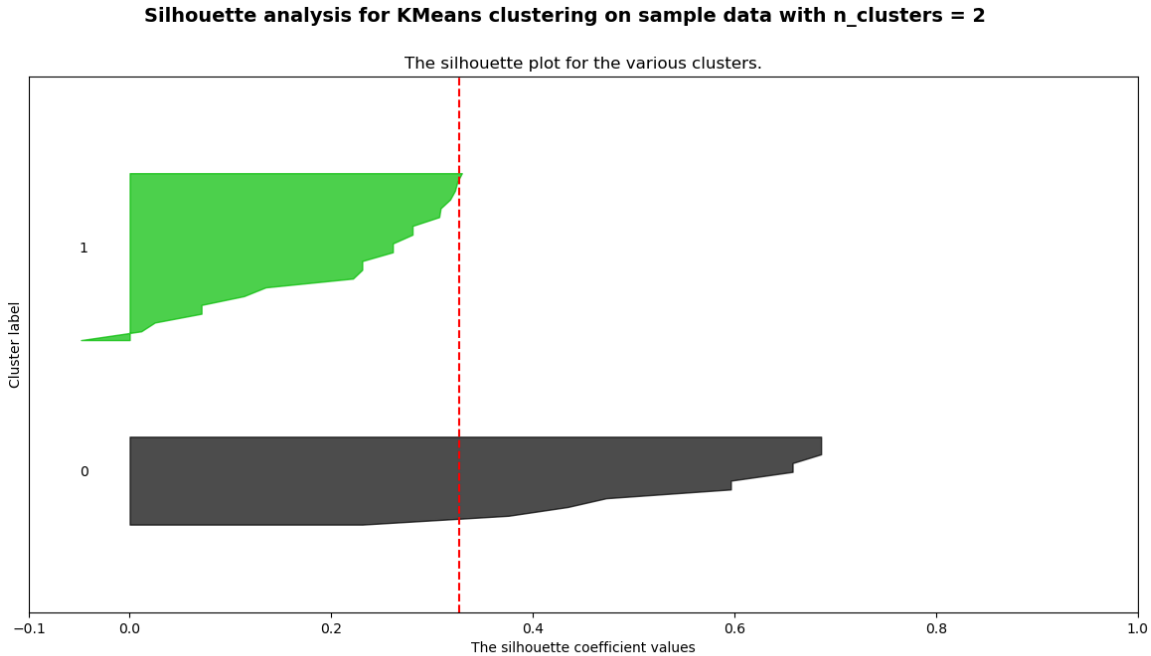


Figure 3: Silhouette plot for two clusters.

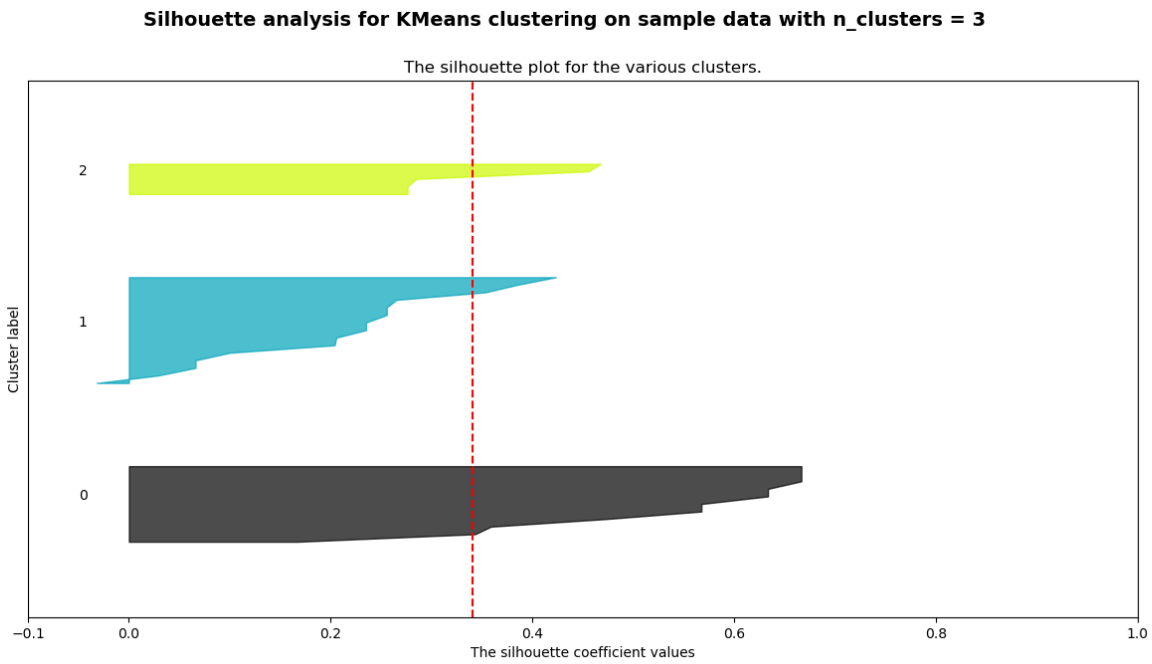


Figure 4:Silhouette plot for three clusters.

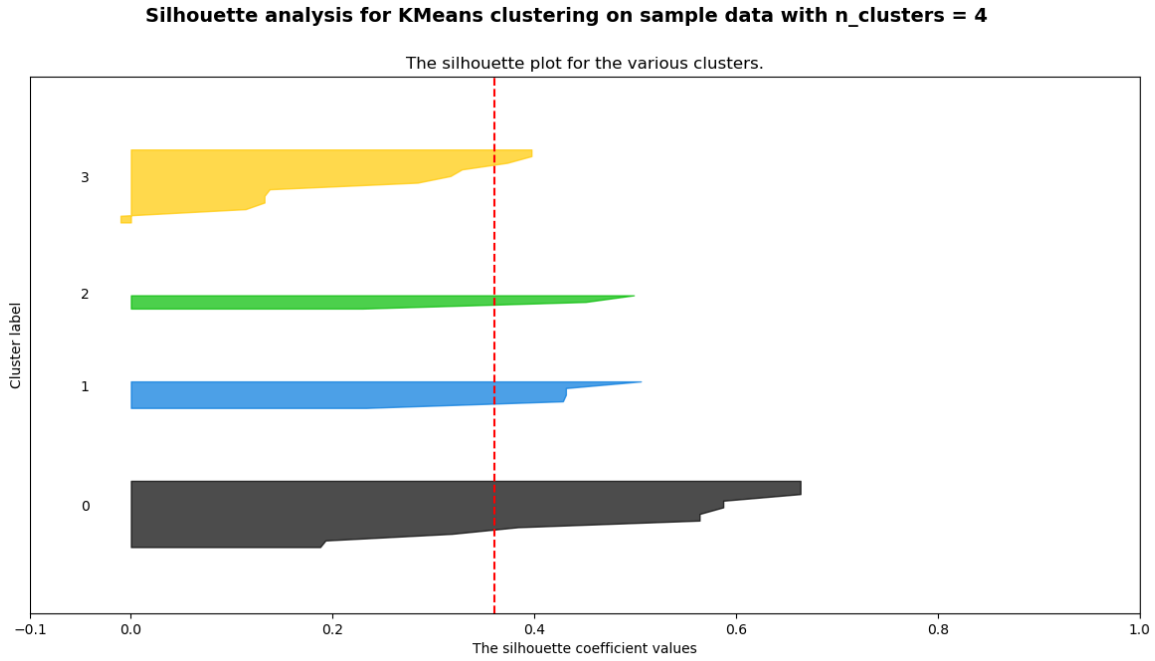


Figure 5:Silhouette plot for four clusters.

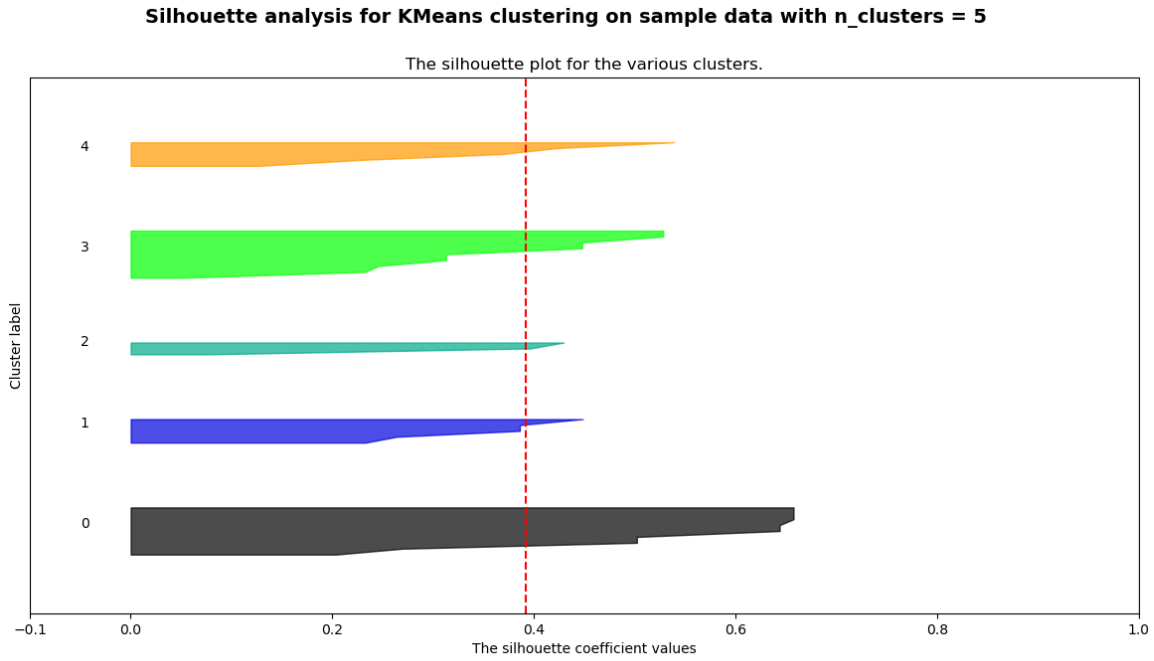


Figure 6:Silhouette plot for five clusters.

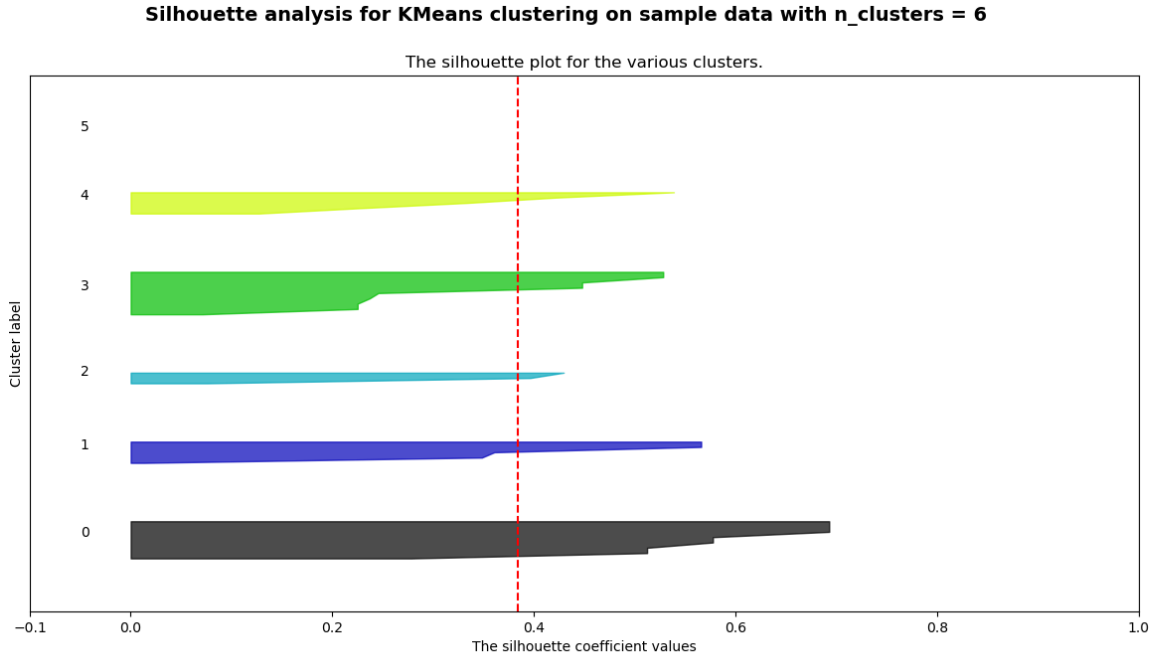


Figure 7:Silhouette plot for six clusters.

The groupings of tracks by k-means are seen in Table 3, on observation they appear to be reasonably assigned. For instance, Monaco, Hungry, Singapore, Mexico and Russia would all be very difficult to overtake and track position would be key when making strategy decisions at these tracks. Track position refers to a driver being ahead of others during a race. Track position can be more valuable than absolute pace on some tracks when overtaking is difficult. On tracks such as Silverstone and Bahrain, track position would not be as important as faster cars will be able to overtake, absolute pace is more important than track position. The tracks that have been grouped together in this process make sense when applying some domain knowledge. To further verify these groupings, I reached out to a data scientist at a Formula One team through LinkedIn. He stated that it is heavily dependent on the context of the groupings but said for this case in terms of strategy the groupings would be considered accurate.

Table 3: Track groupings by k-means.

Type 0	**Type 1**	**Type 2**	**Type 3**	**Type 4**
Spain	Monaco	Azerbaijan	Jeddah	Bahrain
Great Britain	Hungary	Canada	Australia	Austria
Belgium	Singapore	Monza	Imola	Brazil
Netherlands	Mexico		Miami	Sakhir
Japan	Russia		France	Eifel
United States			Abu Dhabi	
Qatar			Shanghai	
Turkey			Portugal	
Tuscany			German	

Exploratory Data Analysis

As part of the exploratory data analysis (EDA) process, a data frame in Python was created which contained all the dry races between 2018 and 2023. The Python EDA libraries, Pandas-profiling and Sweetviz, were used to gain a better understanding of the dataset. These libraries allowed for numerical and categorical association tables to be produced. Plots were also produced to better understand how outliers can be removed from the dataset during data pre-processing. A particular target of the pre-processing is to remove races where drivers are on suboptimal strategies. Examples of this may be if the driver pits to repair damage or has a problem during their pit stop. Exploratory data analysis will be used to directly inform the data pre-processing for the model.

Lap time

The numerical and categorical associations of feature 'LapTime' are seen in Table 4. As would be expected, lap times are highly correlated with Sector 1, 2 and 3 times as well as speed trap values (SpeedFL, SpeedI1, and SpeedST). Note a speed trap is a line placed across the track at which the car's speed over the line is measured in km/h. The correlation with the sector times can be explained because lap times are the sum of the three-sector times. For the correlation with the speed trap values, the faster a car goes in Km/h the lower the lap time will be. It can also be observed that as the LapNumber increases, lap times decrease, this is because of the fuel in each car burning off leading to lighter cars and lower lap times. From the categorical associations, there is a relationship between the track type and lap time. This would be expected as different types of tracks would produce varying lap times.

Table 4: Numerical and Categorical associations of feature 'LapTime'.

NUMERICAL ASSOCIATIONS
(PEARSON, -1 to 1)

Sector1Time	0.71
Sector2Time	0.68
Sector3Time	0.60
SpeedFL	-0.42
SpeedI1	-0.37
SpeedST	-0.32
LapNumber	-0.27
Pressure	0.24
SpeedI2	-0.23
TyreLife	-0.23
TrackTemp	-0.17
WindDirection	0.12
Humidity	0.11
WindSpeed	0.08

CATEGORICAL ASSOCIATIONS
(CORRELATION RATIO, 0 to 1)

Track_Type	0.22
IsAccurate	0.12
Driver	0.12
Year	0.11
Team	0.11
Stint	0.10
Compound	0.08
InPits	0.06
IsPersonalBest	0.04
Deleted	0.01
FreshTyre	0.01

A histogram of the' LapTime' feature is seen in Figure 8 below. Lap Times had been converted to milliseconds as part of the data cleaning process, as spoken about earlier. Observing the plot, the distribution is a Poisson. If a driver were to suffer damage during a lap, they would have lap times much greater than the median lap time for a particular track due to the loss in performance from the damage. The histogram created here can be used to determine a cut-off, above which the lap would be considered to be one where a driver is carrying damage. Drivers who suffer damage are on suboptimal strategies because they will have to enter the pits to repair the car or retire from the race. Allowing the model to train on data from these situations would have a negative effect on the results of the model. From the histogram below in Figure 8, it can be observed that the majority of lap times fall below 150000 milliseconds and so this was the chosen cut-off lap time.

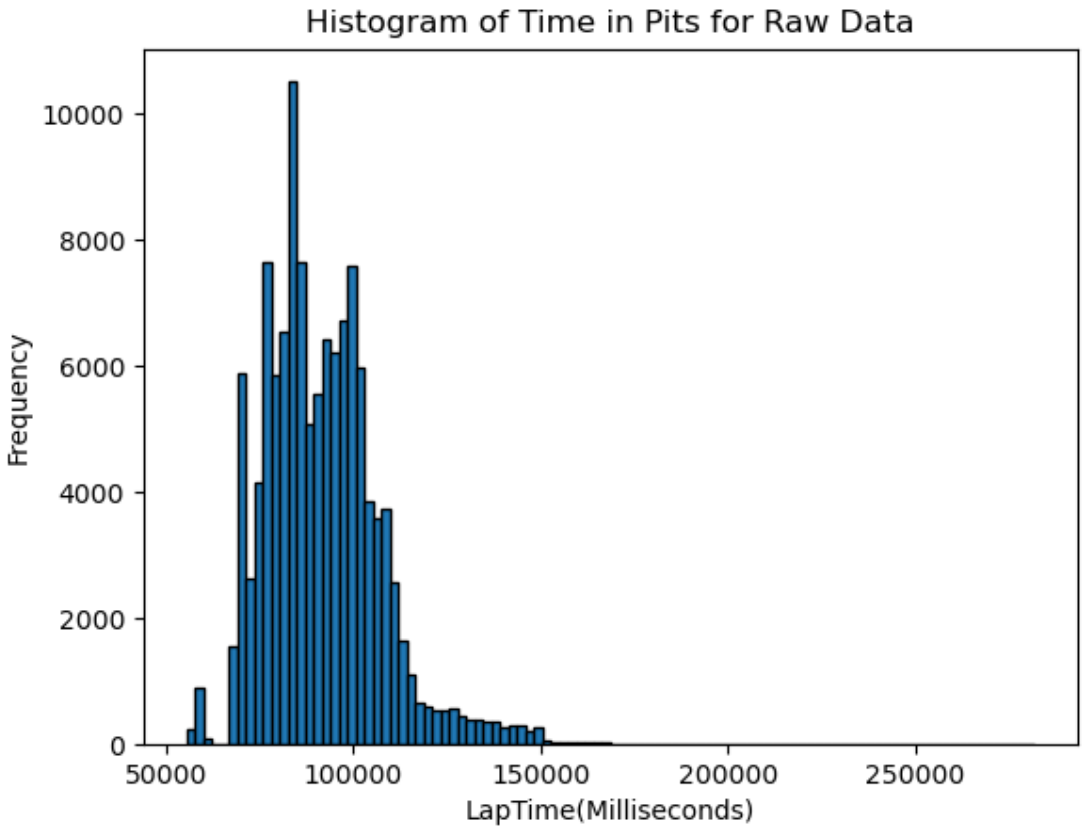


Figure 8: Histogram of Lap Time in Milliseconds for entire dataset.

Track Usage

In Figure 9, the distribution of different track types is seen. T0 tracks are the most common, with tracks like Silverstone (Great Britain), Spa-Francorchamps (Belgium) and Suzuka (Japan) all falling into this type.

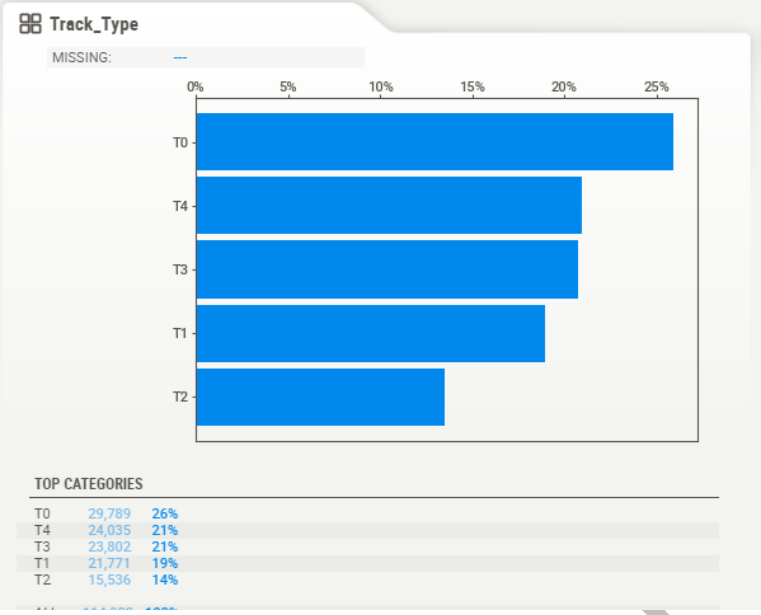


Figure 9: Breakdown of track type usage from dataset.

TimeInPits

The feature 'TimeInPits' expresses how long each driver spent in the pitlane each lap. For most laps, this is zero as drivers remain on the racetrack and so this column is zero. Drivers will come into the pitlane for a change of tyres generally one or two times per race. However, occasionally drivers will have to come into repair damage suffered in races. Damage is unpredictable and is never part of the optimal race strategy, for this reason, it is desirable to remove race data of drivers who pit for damage repairs during a particular race. Pit stops due to damage repairs will take longer and thus give a longer pitlane time. As can be seen from the plot below in Figure 10, the vast majority of pitstops have a TimeInPits of less than 50 seconds. This distribution is once again Poissonian. From this, it was decided to filter out the laps of drivers who had an InPits time of greater than 50 seconds.

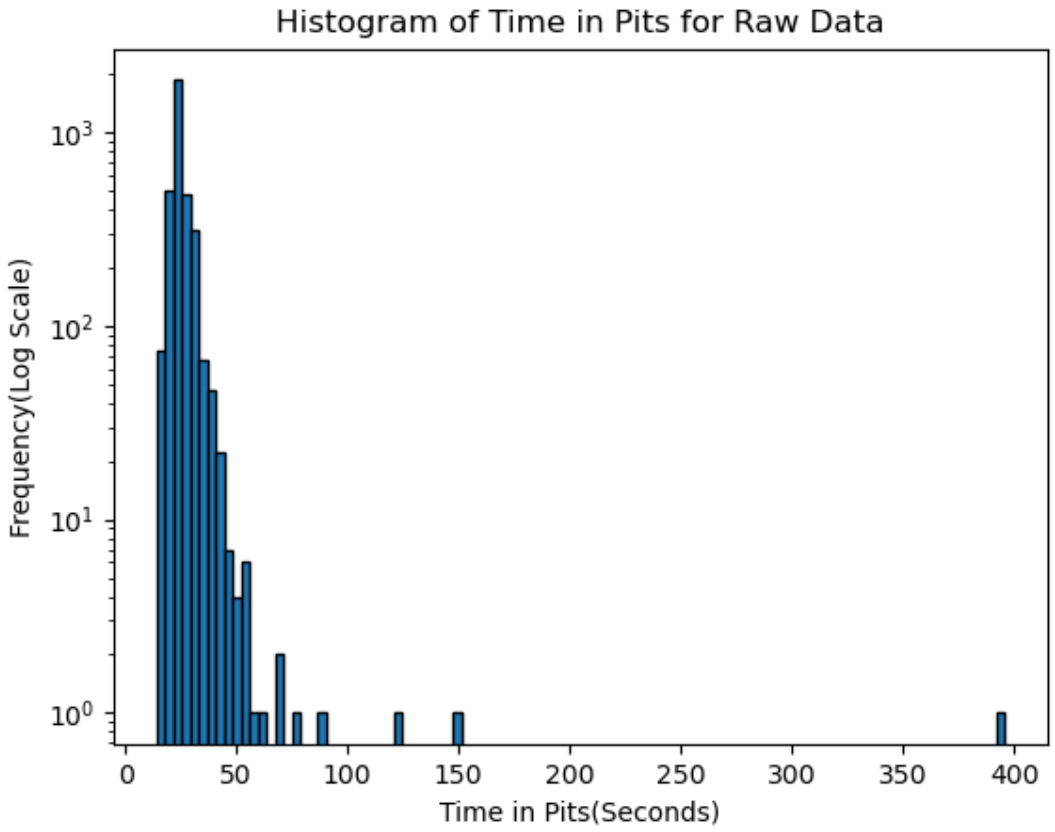


Figure 10: Histogram of Time in Pits for entire dataset. Note Log Scale Used.

Number of pitstops made by a driver

The number of pitstops made by each driver was explored during the EDA process. Drivers who pit multiple times are often on a suboptimal strategy because of damage or poor decisions. To decide a threshold for a maximum number of pitstops, a histogram was created in order to examine the distribution of the number of pitstops undertaken by each driver during races. This histogram can be seen in Figure 11. It can be seen from this that several drivers undertook no pit stop, this indicates that they did not finish the race as at least one pit stop is required during a Formula One race. It can also be observed that in the majority of races, drivers pitted between 1 and 4 times. Conservatively we can save any driver with over 4 pitstops should be excluded. Furthermore, it is beneficial to remove incomplete race data by removing data from drivers who did not pit during a race.

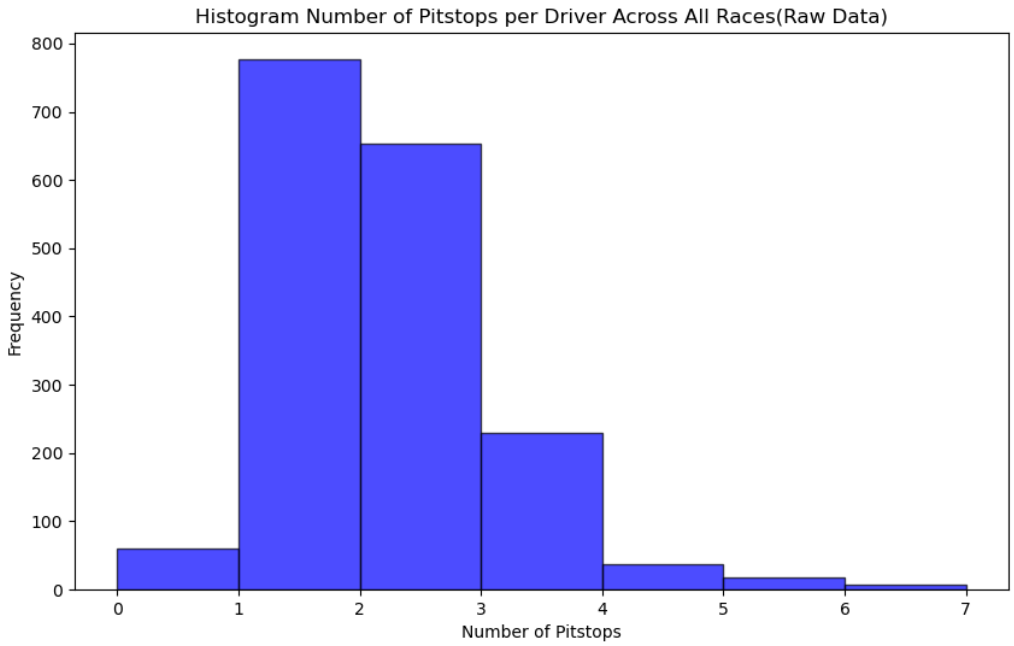


Figure 11: Histogram of the number of pitstops in each race on raw data.

Association Matrix

Below in Figure 12, an association table is seen. From here it can be observed how strongly each feature is associated or correlated to another. Below are some comments concerning the correlations seen:

- A strong positive correlation is seen between Team and Driver, indicating that the drivers often stayed at the same team across the dataset. Secondly, a strong positive correlation is seen between both team and position as well as driver and position. This indicates that the same drivers and teams were winning and losing throughout the years in the dataset. This is confirmed by examining the Constructors' and Drivers' World Championship Results over the seasons included in the dataset. Mercedes won the Constructors' World Championship in four out of the six seasons included in the dataset, with Redbull taking the other two titles. Lewis Hamilton took the Drivers Title three out of the six years, with Max Verstappen taking the other three.[20]
- Stint and Lap Number are also highly positively correlated, which also makes sense as the Stint number would increase as the race goes by and drivers change tyres.

- Sector 1, 2 and 3 times are also positively correlated with Lap time. This correlation has been discussed previously.
- Lap number increases with time, which makes good sense. Other expected correlations are seen with Track Temperature and Air temperature also correlated positively.
- There is a positive correlation observed between track type and Sector 3 time. This is an interesting one as it would not be immediately obvious. This may be down to many tracks having similar corners and straights in their third sectors.
- The feature IsAccurate which is used to determine if the lap time was set during normal race conditions or unrepresentative conditions such as during a safety car, is seen to be associated with the feature InPits. InPits communicates whether or not a driver made a pitstop that lap, this is an interesting observation and indicates that drivers often make pit stops during safety car periods. This does make sense however because the time lost making a Pitstop during a Safety Car or Virtual Safety Car is much less than under normal race conditions.
- Strong negative correlation is observed for SpeedL1(Speed Trap sector One), SpeedL2(Speed Trap sector two), SpeedFL (Speed trap at Finish Line), and SpeedST (Speed trap at start line) against Lap time.

Some correlations are not as obviously explainable such as the correlation seen between Windspeed and Year. A plot of wind across years is seen below. The Wind speed data for the year 2021 appears to be much lower. This is clear from the plot seen in Figure 13.

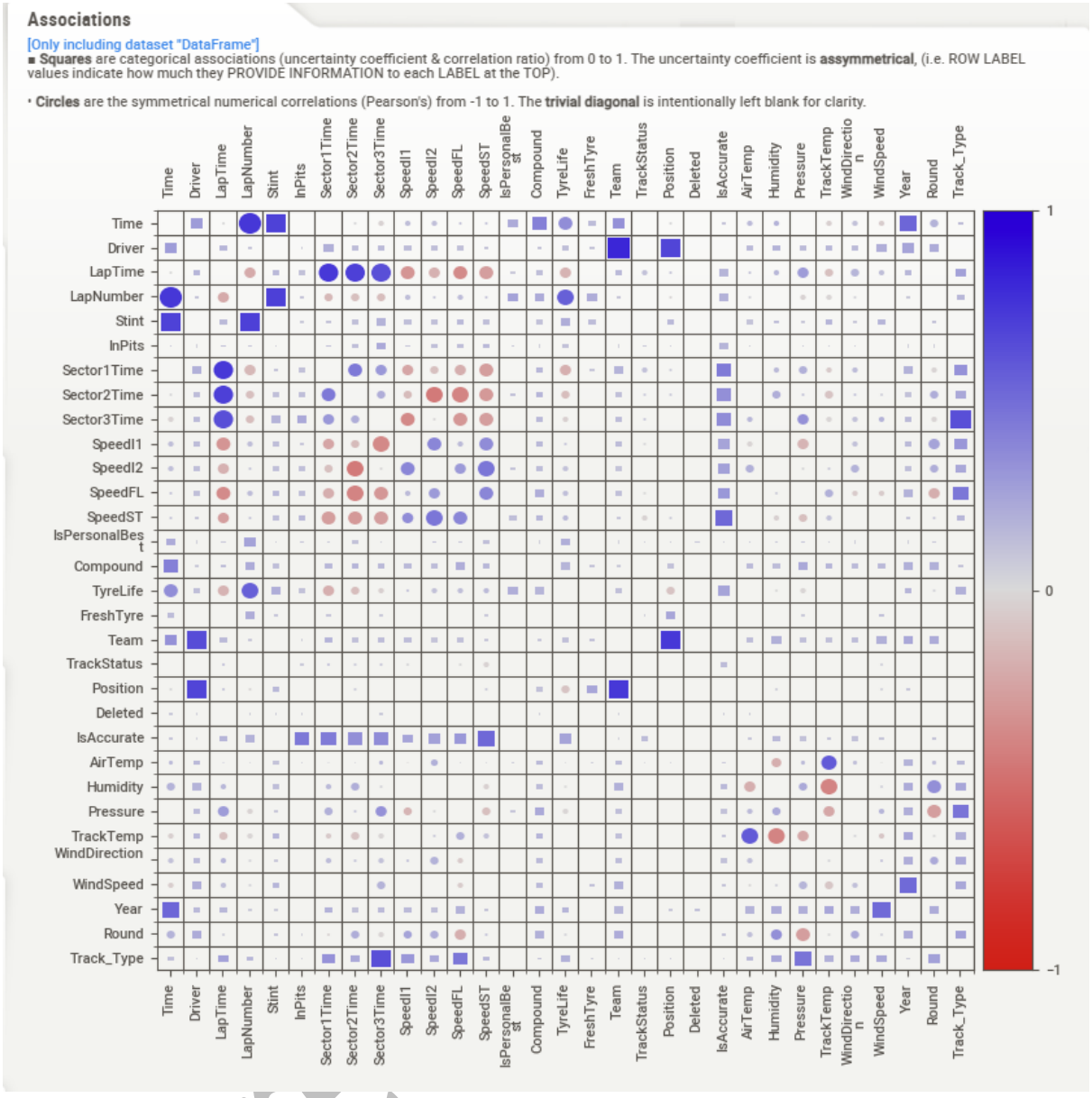


Figure 12: Association Matrix of dataset before pre-processing.

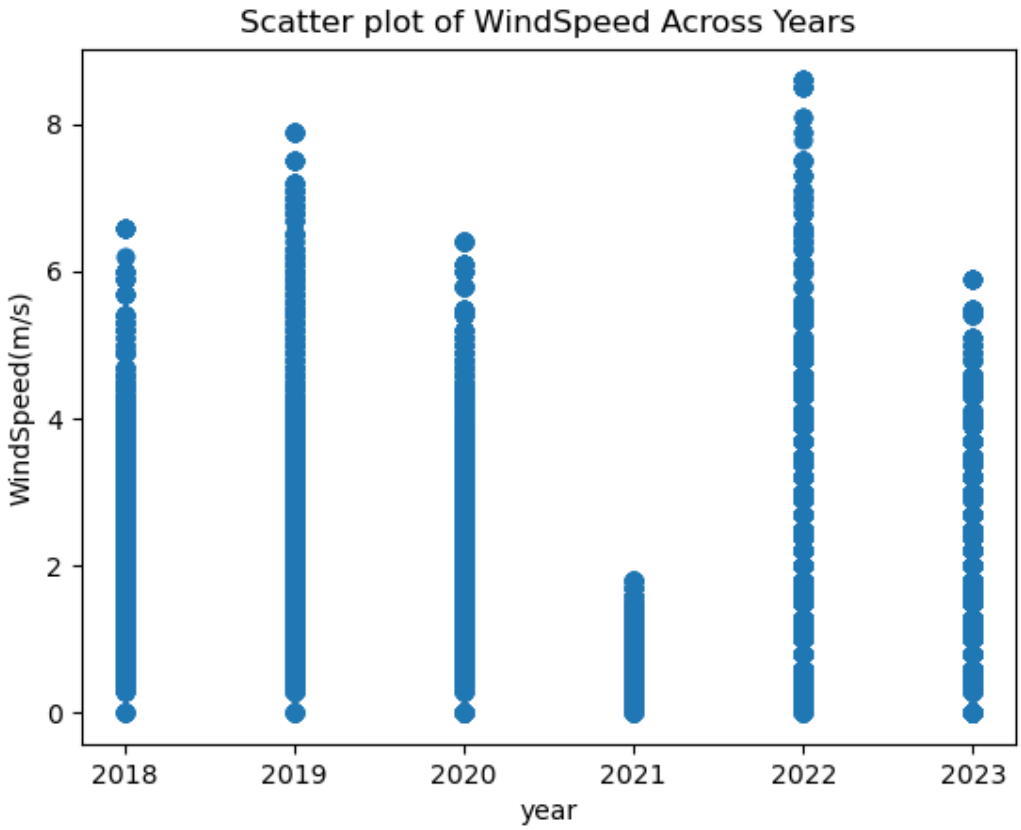


Figure 13: Distribution of Wind speed across years.

Data Pre-processing

Once the data had been cleaned, as spoken about earlier, it was passed through a pre-processing script which included several actions. To prepare the data for training, outliers were removed, and some new features were created. The actions taken during pre-processing were decided based on the evidence of EDA. For the pre-processing of the data, each race was processed individually such that if the laps of a driver are being removed, those laps are only for a particular race.

Outlier Removal

As mentioned earlier, one of the main objectives of data pre-processing here is to remove drivers on suboptimal strategies as including these in the model would likely lead to worse results. Following the threshold discovered during EDA, the following actions were taken to remove rivers on suboptimal strategies. When removing the laps related to a driver, laps were only removed for the specific race at which they were outside the threshold.

- Removal of driver's laps with 'TimeInPits' time of greater than 50 seconds.
- Removal of driver's laps with 'LapTime' greater than 150000 milliseconds.
- Removal of driver's laps with more than 4 pitstops during a race.
- Removal of driver's laps that pit on the first two or last laps of the race.

Normalization

As part of any machine learning pre-processing, normalising that data is a useful and recommended step. As part of this, the original full-scale values remained columns in the data frame, with the normalised values being added to the data frame as a new column. This gives the flexibility of having both values available in the dataset. When normalising 'LapNumber' and 'TyreAge' features, they were done on a race-by-race basis to account for the varying number of laps in a race as per the length of the racetrack. For instance, the Hungarian Grand Prix which takes place at the Hungaroring has 70 laps for the Grand Prix, this is compared to the Saudi Arabian Grand Prix which takes place at the Jeddah Corniche Circuit which has 50

laps. All F1 races are approximately 300 km so this makes the number of laps feature comparable between racetracks.

Dealing with lapped Drivers

When the data on the gaps between drivers, from the Fastf1 API, is received it is expressed in seconds for all drivers who are on the lead lap of the race. This means these drivers have not been lapped by the leading driver in the race. For drivers who get lapped by the leader of the race, they are said to be a lap down. In the data from the API, this is expressed as '1L' when the driver is one lap down and '2L' when two laps down and so on. It is desirable to convert these back into seconds. To do this, the median lap time for the race is computed, converted to seconds, and filled in to replace the '1L', '2L' etc. values.

Championship position

Having reviewed domain knowledge on the topic of Formula One race strategy, it can be observed that the race strategy for a driver in a car which is inherently the fastest car will be different to that of a driver in an inherently slow car. This is an important strategic consideration for any strategist. Thus, it is important for the model, to have awareness of the inerrant pace of each driver's car. This can be expressed through the championship position for each driver and team at the end of each season. As part of the pre-processing, two new features were added to each lap, the Drivers' Championship position of the driver who set the lap and the Constructor's Championship position for the drivers' team. The championship positions were based on the end of each season.

Two tyre compounds used

As part of the FIA regulations, each driver must use at least two tyre compounds during a Formula One race. Failure to do this will mean the driver will be disqualified from the race. For this reason, the model must develop an understanding of the necessity to use two different tyre compounds during the race. To do this, a new feature was created in the

dataset, which simply expresses (True/False), if a driver has used two tyre compounds during a race up to the current lap.

Calculate number of laps until next pitstop

This feature was added to the dataset after issues with the initial model due to the imbalanced data on InPits. This issue will be discussed further later in the report. This feature calculated the number of laps until a driver's next pitstop. It will count down to zero on the lap that the driver pits. This allows for a regression model to be created and would serve as a more useful prediction for a Formula One strategist as it would allow for the confidence of the model to be expressed.

Determine if a Drivers are close on track

Based upon domain knowledge of the Formula One race strategy, two seconds is considered the range within which the undercut and overcut may be attempted. To give the model awareness of these strategic options three new features were added. One expressed whether a driver was within 2 seconds of the car in front, this would indicate they could use an undercut or possibly the overcut. Another feature was created to express whether there was a car within 2 seconds behind, this is used to express if they at risk of being undercut. Finally, a feature was created to express if a car had another one close, regardless of whether it was behind or ahead. This came from the idea that having two cars close together on track during a race increased the likelihood that at least one car pits. These features were created to allow for options during training and may not be used in the final model.

In the final model, another feature was added to give the model further awareness of undercuts. This new feature was named 'PitStopBehind'. This feature was a Boolean and was true when the following occurs. Driver A is ahead of Driver B and Driver B is within two seconds of Driver A. Driver B pits. This will mean that 'PitStopBehind' will be true for the next two laps for Driver A. This feature allows for memory to be added to the model through a feature.

Add race ID and Lap ID

A unique Lap and race ID were assigned to each lap and race respectively. These allowed for easy divisions of data between training, validation and testing. They also made it easier to analyse the results of the model allowing for the easy comparison of model results and the true results.

Model Building

All models were built in TensorFlow and Keras. All models were trained and tested using a MacBook Pro 2020. Before settling on the final model, M10, there were several variants. In the following section, the milestone models during the model development will be discussed.

The division of data between training, validation and testing is illustrated in Figure 14. It can be seen that out of the 110 dry races between 2018 and 2023. All races between 2018 and 2022 have been assigned to be used as training data. The 2023 season was chosen to be used as validation and test data. This was done as the 2023 season is the most relevant to future F1 races as the regulations for car design have not changed a great deal since 2023.

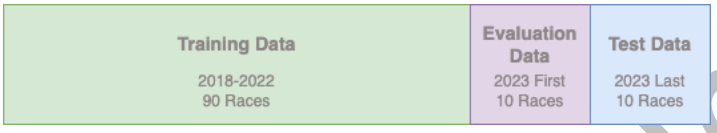


Figure 14: Diagram displaying division of data between training, validation, and test data.

M0

The first model built was a classification model. The target variable here was InPits, a variable which is true on the lap that drivers pit and false otherwise. Hence with this model, the intention was to predict the laps that drivers pit. The model structure and hyperparameters can be seen below in Table 5.

Table 5: Hyperparameters of M0.

Hyperparameter	Value
Number of hidden Layers	3 dense layers
Number of Neurons per layer	64
Activation functions for Hidden Layers	ReLU
Kernel Regularizer	L2(0.0005)
Optimizer	Nadam
Loss Function	Binary Cross entropy
Training class weights	InPits=False: 1, InPits=True:5
Training batch size	256
Early Stopping	Patience of 5

This model was built using Keras and TensorFlow. An overview of the features and predictor variables is seen in

Table 6. Many of these were carried over from the VSE[6] project but there have been some changes. In the VSE project, the tyre compound is a relative compound, soft medium and hard, even if the absolute compound changed from track to track. For this initial model, the absolute compound was used. The 'Close' feature was used, this is a combination of close ahead and close behind as spoken about previously. The features for Drivers' Championship position and twotyrecompounds used were also new. As ReLU activation functions are being used in this model, all numerical features were normalised. All categorical features were one hot encoded.

Table 6: Features Used by M0.

Feature	Value Range
Race Progress	[0.0, 1.0]
Tyre Life progress	[0.0, 1.0]
Position (Normalised)	[0.0, 1.0]
Compound	[SSSS, SSS, SS, S, HH, HHH]
Track Type	[T0, T1, T2, T3, T4]
Track Status	[0, 1, 2, 3, 4]
Close	[True, False]
Drivers' Championship Position (Normalised)	[0.0, 1.0]
Two Tyre Compounds Used	[True, False]

The previously spoken about train/evaluation/test split of data was used in training this model.

The target variable for this model was heavily imbalanced, with InPits being false for 97% of the dataset. This makes sense as drivers will generally only pit one or two times a race, and so they will only be in the pits once or twice a race, which gives rise to the heavily imbalanced dataset. This led to an issue while training as the model could achieve a 97% accuracy by predicting False for InPits, for all laps. Class weights were applied to rectify this issue, as it had been seen they had been applied in the VSE project[6]. Even with these applied, the issue persisted.

M1

To circumvent the issue of the imbalanced dataset, for M1 a change was made in the target variable. A new feature was created in the dataset which counted down the remaining laps until a driver's next pitstop. This feature was added as part of the data pre-processing script. This now meant the model became a regression model. The structure of this model was largely unchanged with the exception of the output layer which now had a single connected output node. The same hyperparameters were used here as M0, and the data splits were the same as M0.

The results here were better than in the previous M0, although still not to an acceptable level of accuracy. The model captured the general trend but failed to ever predict values close to 0 for the laps shortly before a pitstop. It also failed to predict the high number of laps initially after a change of tyres, when the number of laps until next the pitstop would be at a maximum.

Several variants of M1 were created to form M2 and M3, but no noteworthy progress was made with each of these models.

M4

After the issues of the previous two models, several development directions for the model were considered including using an LSTM. However, while writing up the report section on M0, several mistakes were discovered in the original model. One issue meant that the validation dataset size was much smaller than intended. The track status feature was modified to ensure that it only contained 5 possible categories, as previously there may have been more. The extra categories occurred when the track status changed during a lap, this led to new categories being created. This means the tract status on each lap is now one of the following.

- 0: Red Flag
- 1: Green flag conditions
- 2: Yellow Flag conditions in any sector
- 3: Virtual Safety Car
- 4: Safety Car

The final change made to create M4 was to the output layer. Previously there had been a mistake made in having two output neurons on the output layer. For M4, a single neuron with a sigmoid activation function was used, with the output now being interpreted as a probability. The new updated network structure is seen in Table 7.

Table 7: M4 Hyperparameters. Updated classification model.

Hyperparameters	**Value**
Number of Hidden layers	3
Number of Neurons Per Hidden Layer	64
Hidden Layer Activation Functions	ReLU
Class weights	Balancing Function used
Kernal Regularizer	L2(0.0005)
Optimizer	Nadam
Loss	Binary Cross Entropy
Early Stopping	Monitoring validation loss with a patience of 5

This model did not suffer the same issues as M0 which were caused by the imbalanced dataset. In order to demonstrate the performance of the model, let us analyse two races from the test set the first is the 2023 Belgian Grand Prix, with driver George Russel and Sergio Perez in focus. In Figure 15, a plot is seen in which the x-axis represents the Lap Number of the race, and the Y-axis is the model's prediction of the probability of the driver pitting that lap. The vertical dotted lines represent when a pitstop was actually made by the driver. This kind of plot would be especially useful to a Formula One strategist as it would allow for decision confidence to be communicated by the model as opposed to a simple True/False decision.

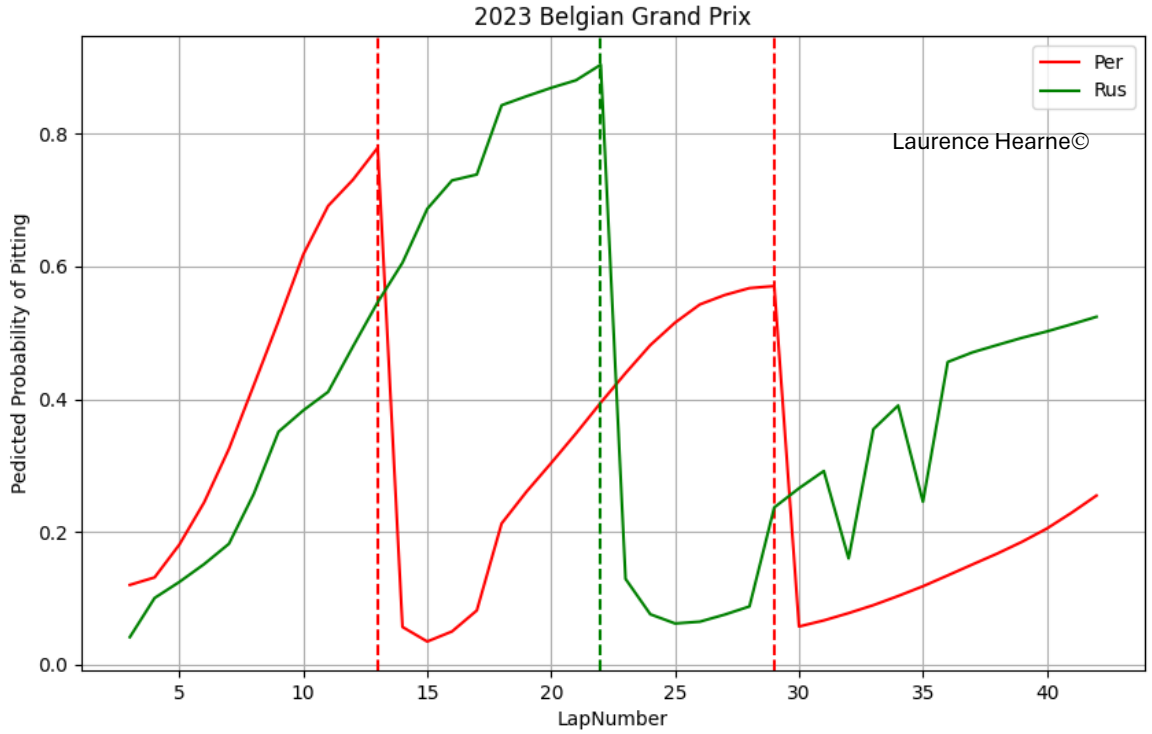


Figure 15: M4 performance on 2023 Belgian Grand Prix. Focussing on Sergio Perez(red) and George Russell(green).

From Figure 15, the following comments can be made:

- Russel started on the medium tyre and Perez on the soft. The model appears to have identified that Russel was less likely to pit earlier in the race due to being on the more durable tyre.
- The model increased the probability of Russel making a second pitstop late in the race, even though he did not make a second pitstop.
- It is difficult to determine a threshold at which a Formula One strategist would trigger a pitstop. For example, if we look at the Perez trace, above 75% could be used as a good threshold at which to trigger the first pitstop, however, this would mean the second would be missed. The use of a threshold of approximately 55% would be suitable for each driver's second pitstop but would also mean that the first stop of each driver would be triggered much earlier than it did actually occur. This does of course assume that Perez stopped on the optimal lap when he did actually stop.

Nonetheless, an improvement that should be sought to improve the model would be a faster ramp-up in the probability of pitting, rather than the gradual increase seen here.

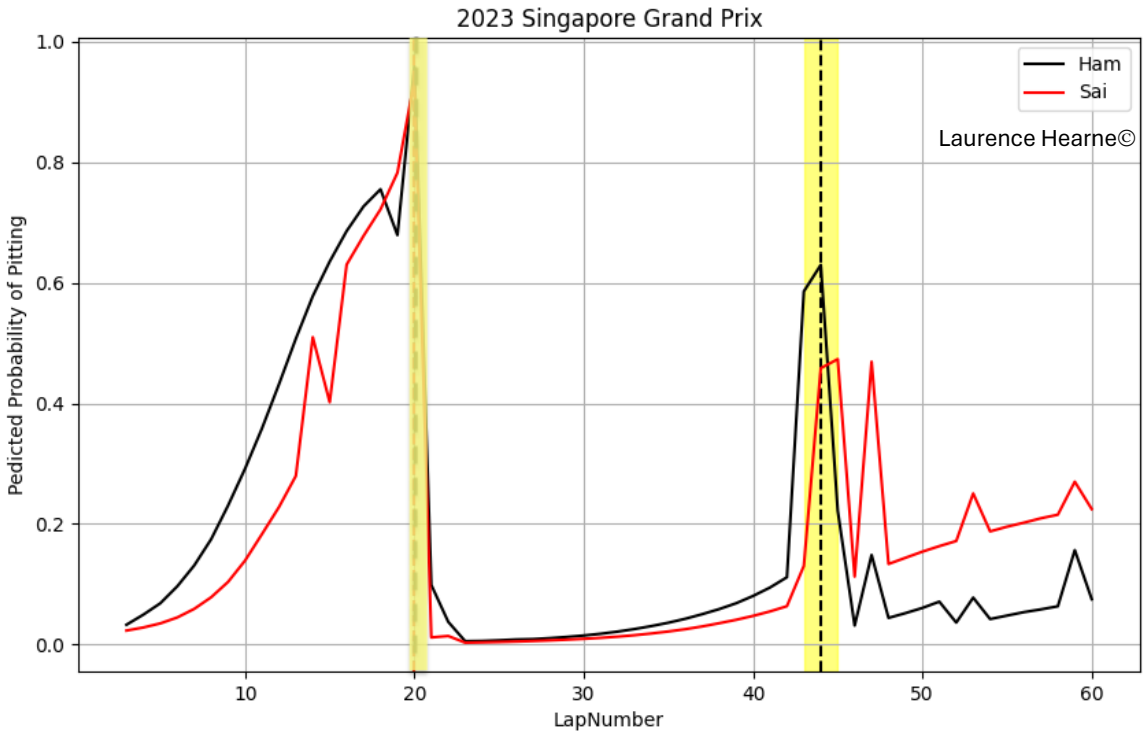


Figure 16:M4 performance on 2023 Singapore Grand Prix. Focussing on Lewis Hamilton(black) and Carlos Sainz(red).

Looking at the plot for Drivers Lewis Hamilton and Carlos Sainz in the 2023 Singapore Grand Prix, seen in Figure 16 above, the following comments can be made:

- The model correctly increased the likelihood of the drivers pitting during the two safety car periods, as highlighted in yellow. In the case of the first safety car period, both drivers did pit. The gradual ramp up in probability is not as big of an issue during this race. In the case of the second safety car period Hamilton did take this opportunity to pit. Sainz did not pit, the model identified that he had a lower probability of pitting as he was the race leader. The probability of Sainz dropped back down once the safety car period was over.

- The second spike shortly after the safety car is down to yellow flags that came out but did not turn into full safety car.
- There is a slight upwards drift in the probability of both drivers pitting towards the end of the race. In reality, at a race like Singapore, it would be very unlikely for a driver to pit in the last number of laps of a race,

Hyperparameter optimisation of M4

Having achieved a baseline model which appears to have identified the underlying trends in the data, it was sought to optimise the hyperparameters of the model to maximise the performance of this model and complete the ML model development pipeline. A run was completed in which all combinations of the below hyperparameters were trialled. The results were then analysed to identify the common hyperparameter values for successful models.

Table 8: Hyperparameter values used in Hyperparameter Tuning Run.

Hyperparameter	Values
L2 Regularization	0.0003,0.0004,0.0005,0.0006
Activation Functions	ReLU, sigmoid
Layer 1 Neurons	16, 32, 64
Layer 2 Neurons	16, 32, 64
Layer 3 Neurons	16, 32, 64
Learning rate	1e-5, 1e-4, 1e-3

When the above run was completed, there were various combinations with similar performance, however many of the best-performing combinations had common

hyperparameters. Having evaluated the predictions of various hyperparameter combinations, the hyperparameters seen in Table 9 were selected as the optimal hyperparameters for M4.

Table 9: Optimal Hyperparameters determined through hyperparameter tuning run.

Hyperparameters	**Value**
Number of Hidden layers	3
Neurons Hidden Layer 1	16
Neurons Hidden Layer 2	64
Neurons Hidden Layer 3	32
Hidden Layer Activation Functions	ReLU
Output Layer Activation Function	Sigmoid
Class weights	1:21, 0:1
Kernel Regularizer	L2(0.0003)
Optimizer	Adam
Loss	Binary Cross Entropy
Early Stopping	Monitor Validation Loss. A patience of 5 was used.

M10

Following M4, there were numerous variants of the model with slight variations made to each model. These led to the final model version, M10, this model produced all the data seen in the results section. This version of the model saw the addition of the 'PitStopBehind' feature. The main purpose of this feature is to give the model awareness of undercut attempts of

Drivers behind. It will be seen as part of the results section how the M10 model can successfully identify and cover undercut attempts. Hyperparameter optimisation was once again run on this model version. The final features and hyperparameters of the model developed are seen in Table 10 and Table 11 respectively.

Table 10: Features used in M10.

Feature	Value Range
RaceProgress	[0.0, 1.0]
TyreLifeProgress	[0.0, 1.0]
Position	[0.0, 1.0]
Compound	[SOFT, MEDIUM, HARD]
TrackType	[T0, T1, T2, T3, T4]
TrackStatus	[0, 1, 2, 3, 4]
CarClose	[True, False]
DriversChampionshipPosition	[0.0, 1.0]
TwoTyreCompoundsUsed	[True, False]
PitStopBehind	[True, False]

Table 11: Hyperparameters used in M10.

Hyperparameter	Value
Number of Hidden layers	3
Neurons Hidden Layer 1	16
Neurons Hidden Layer 2	64
Neurons Hidden Layer 3	32
Hidden Layer Activation Function	ReLU
Class weights	1:21, 0:1
Kernel Regularizer	L2(0.0003)
Optimizer	Adam
Loss	Binary Cross Entropy
Early Stopping	Monitor Validation Loss. A patience of 5 was used.

Results

All results presented as part of this results section are from the model M10 as spoken about previously. The hyperparameters for this model have been optimised by performing hyperparameter tuning as spoken about earlier. All results were from the last ten dry races of the 2023 season, this data was not used during training and evaluation, making it separate.

Undercut Situation

Undercuts are a commonly used race strategy as they allow for one driver to overtake another without physically doing it on track.[5]

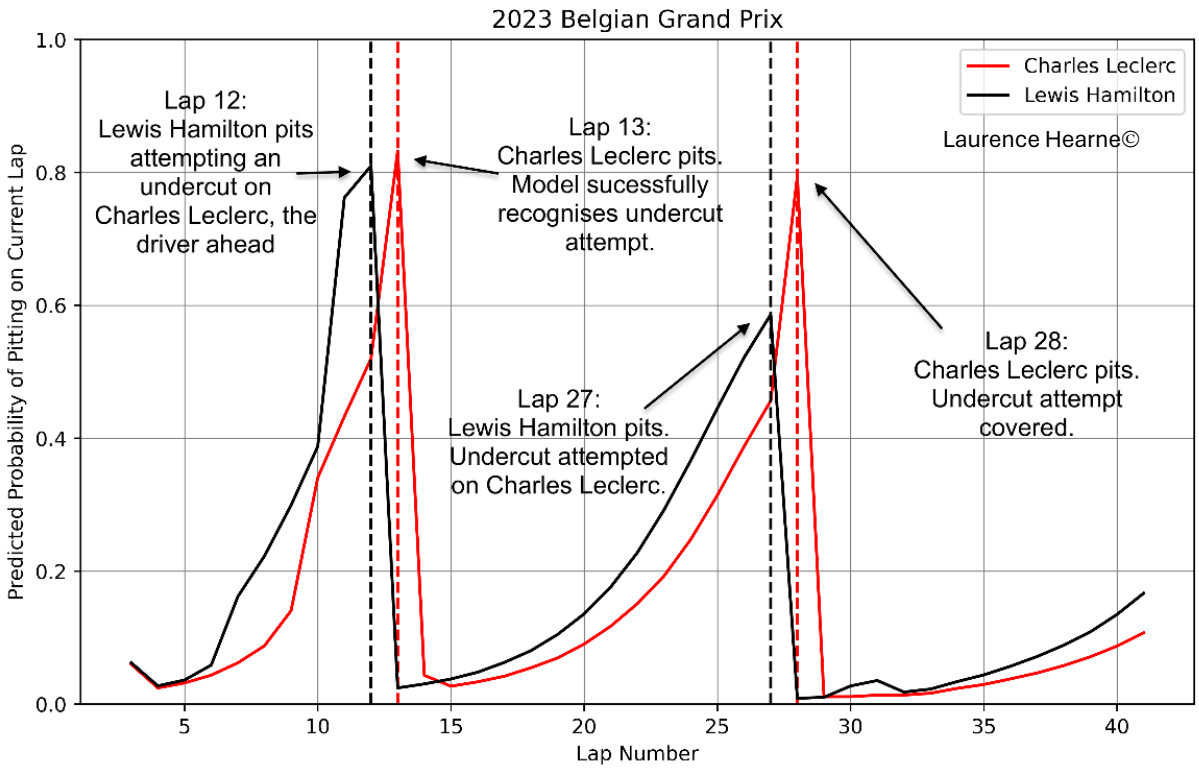


Figure 17: M10 predictions for the 2023 Belgian Grand Prix. This plot demonstrates the model's ability to cover undercut attempts.

In Figure 17 above a plot from the 2023 Belgian Grand Prix is seen. On the y-axis of the plot, the model-predicted probability of a given driver pitting is expressed. The x-axis relates to the

lap number of the race. As part of the previously mentioned pre-processor actions, data relating to laps 1 and 2 of each race are removed as any pitstops made during these laps are generally related to the repair of damage and would not be part of an optimal strategy. The black and red lines of the plot are the predicted probabilities of Lewis Hamilton and Charles Leclerc pitting on a given lap, respectively. This is not a cumulative probability, but instead an independent probability for each lap. The vertical dotted lines signify the laps on which a driver of the matching colour pitted. For the purposes of this analysis, Lewis Hamilton and Carlos Sainz will be referred to as HAM and LEC, respectively. From this plot the following observations and comments can be made:

- As would be expected, the predicted probability of both drivers pitting early in the race is low. As the first pitstop approaches, LEC is in 2nd position and HAM is close behind in 3rd position.
- As the pit window is approaching, HAM remains close to LEC but is unable to overtake him on track. In these situations, the undercut may be attempted. This would mean HAM would pit before LEC, come out on fresh tyres, and put in some quick laps such that by the time LEC pits, LEC would fall behind HAM. This can be seen to be triggered by HAM on lap 12. The model does a good job of spotting this opportunity to utilise such a strategic move. When an undercut is attempted, the driver ahead, LEC in this case, can come in the next lap to "cover" the undercut. This would mean that Hamilton's fresh tyre advantage would be nullified as they would both be on new tyres. The model recognises this and greatly increases the likelihood of LEC pitting after HAM has made his pitstop. The probability of LEC pitting went from approximately 50% on the lap that HAM pitted to over 80% on the lap after HAM pitted. This is extremely positive as attempting and covering undercuts are two important and common strategic situations.
- The model greatly reduces the probability of a pitstop once each driver has made their pitstop, it had developed the understanding that it is not optimal to pit on successive laps.
- A similar situation as occurred with the first round of pitstops, is seen again with the second round of pitstops. The model does not predict the second pitstop for HAM with

a high degree of confidence but does recognise the need for LEC to respond to the pitstop of HAM behind him. It shoots the probability of LEC pitting from 50% before HAM pits to 80% once HAM pits.

- The probability of both drivers pitting remains low for the remainder of the race. This was an issue in the earlier version of the model with the predicted probability of drivers pitting late in the race drifting towards 50%.

Safety Car Situation

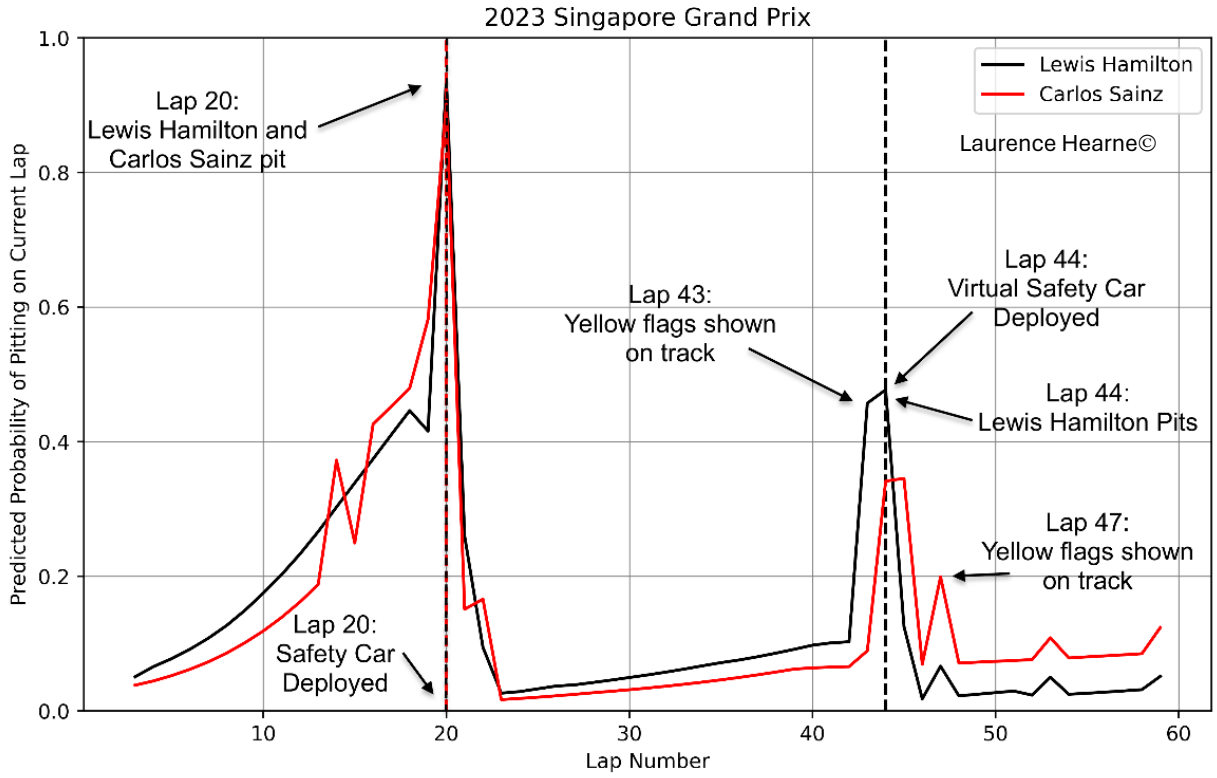


Figure 18: M10 predictions for the 2023 Singapore Grand Prix. This plot demonstrates the model's ability to recognise the strategic benefits of a safety car.

In Figure 18 plot of the predicted probability of pitting for the 2023 Singapore Grand Prix is seen. The axes are as described earlier. The purpose of analysing this race is to examine the model's ability to recognise the benefit strategic benefit of pitting during a safety car or virtual

safety car. The cost of a pitstop during a Safety car and Virtual safety car is greatly reduced from normal race conditions as all other cars on track are travelling much slower so fewer positions will be lost during a pitstop. The ability to identify the opportunity presented by a safety car is important whenever strategic decisions are being made during a race. The black and red lines of the plot are the predicted probabilities of Lewis Hamilton and Carlos Sainz pitting, respectively. For the purposes of this analysis, Lewis Hamilton and Carlos Sainz will be referred to as HAM and SAI, respectively. From this plot the following observations and comments can be made:

- As expected, the probability of both drivers pitting is low at the start of the race. SAI starts the race in 1st place and leads the race as the first pitstop approaches. HAM is in 5th as the first Pitstop approaches.
- On Lap 20 a safety car is deployed after Logan Sargeant hits the wall. On the lap, before the safety car was deployed, the probability of both HAM and SAI pitting was approximately 45%. Once the safety car is deployed, the probability of each pitting increases above 90%. This clearly shows that the model has identified that it is optimal to pit during a pitstop. Both HAM and SAI take advantage of this opportunity and pit.
- Later in the race, on lap 44, a Virtual Safety car is deployed. Singapore is generally a one stop race and so making a pitstop, even under the Virtual Safety Car at this stage of the race would have been considered riskier. SAI was in 1st place and would not have much to gain in making a pitstop under this Virtual Safety Car. HAM was in 4th position and so would have less to lose and more to gain in making a pitstop under this Virtual Safety Car. The model appears to have recognised the situation and increased the probability of HAM pitting more than the probability of SAI pitting under the Virtual Safety Car. HAM did make a pitstop under the Virtaul Safety Car and SAI did not.

Case of Poor Strategy

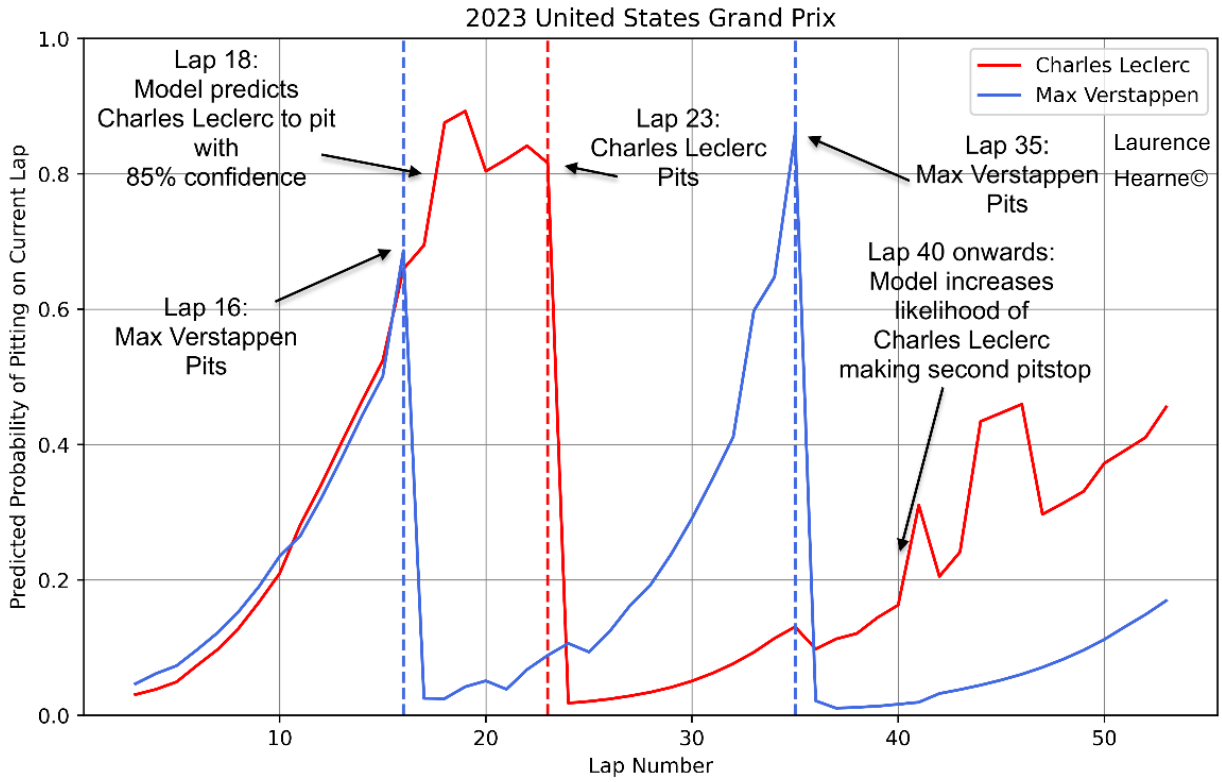


Figure 19: M10 predictions for the 2023 United States Grand Prix. This plot compares a driver on an optimal strategy versus a suboptimal strategy.

In Figure 19, above a plot of the predicted probability of pitting is seen for the 2023 United States Grand Prix, with Charles Leclerc seen in red and Max Verstappen seen in blue. The axes are as described earlier. For this analysis, Charles Leclerc and Max Verstappen will be referred to as LEC and VER, respectively. The purpose of analysing the model's performance for LEC and VER in this race is to show an example of a driver on an optimal strategy and a driver on a suboptimal strategy. In this race, it was widely considered after the race that LEC was on a suboptimal strategy. LEC started the race in 1st and finished in 6th. Ferrari was widely criticized after the race for its poor race strategy. VER started the race in 6th and finished the race in 1st. On Lap 18, it can be seen that the model has predicted with high confidence that LEC should make a pitstop. LEC does not pit until Lap 23, which Ferrari publicly admitted after the race

was the wrong decision. They attempted a one-stop race in leaving LEC out for longer, this was ultimately the wrong decision and LEC lost many positions in the race as a result. The model also began to increase the likelihood of LEC abandoning the one-stop strategy and making a second stop later in the race.

Top driver and backend driver compared

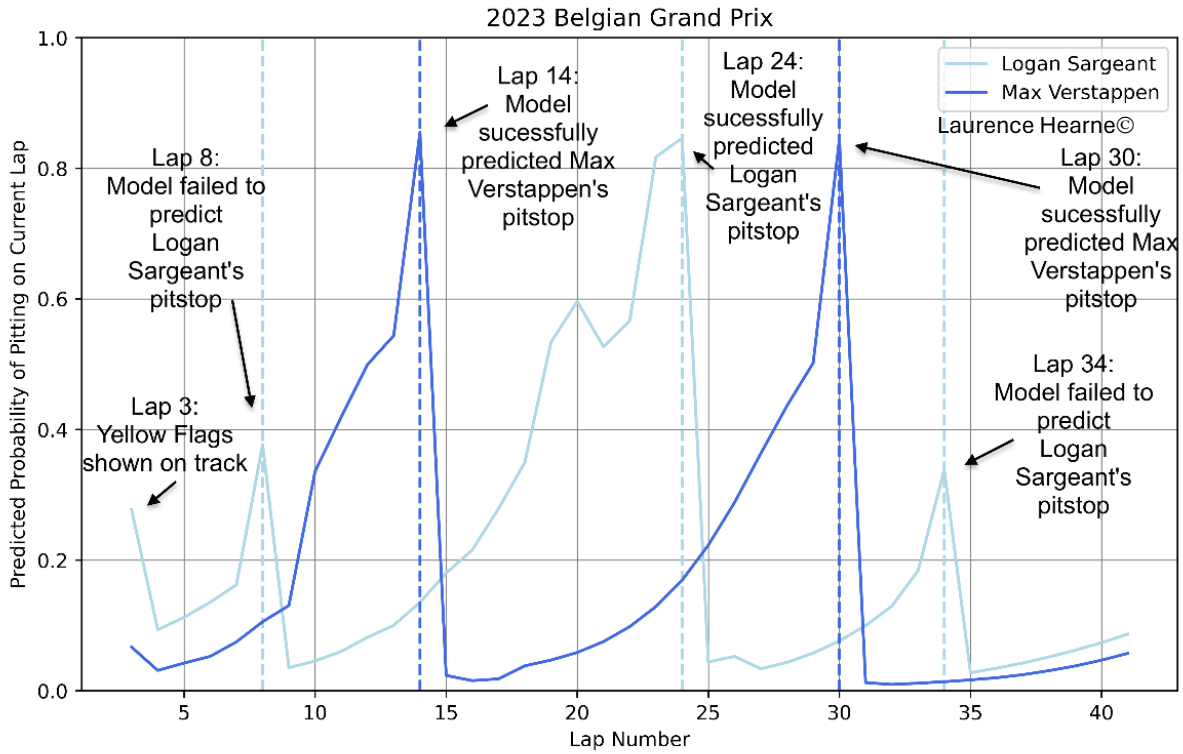


Figure 20: M10 predictions for the 2023 Belgian Grand Prix. This plot serves to compare to the model's performance on a front running team versus a team closer to the back.

In above a plot of the predicted probability of pitting is seen for the 2023 Belgian Grand Prix, with Logan Sargeant seen in light blue and Max Verstappen seen in dark blue. The axes are as described earlier. During this analysis, Logan Sargeant and Max Verstappen will be referred to as SAR and VER, respectively. The purpose of analysing the model's performance on these two drivers is to compare a top-placed driver versus a driver at the backend of the championship. VER finished the test set season, 2023 in 1st place and SAR finished the season

in 21st place. When examining the plot in

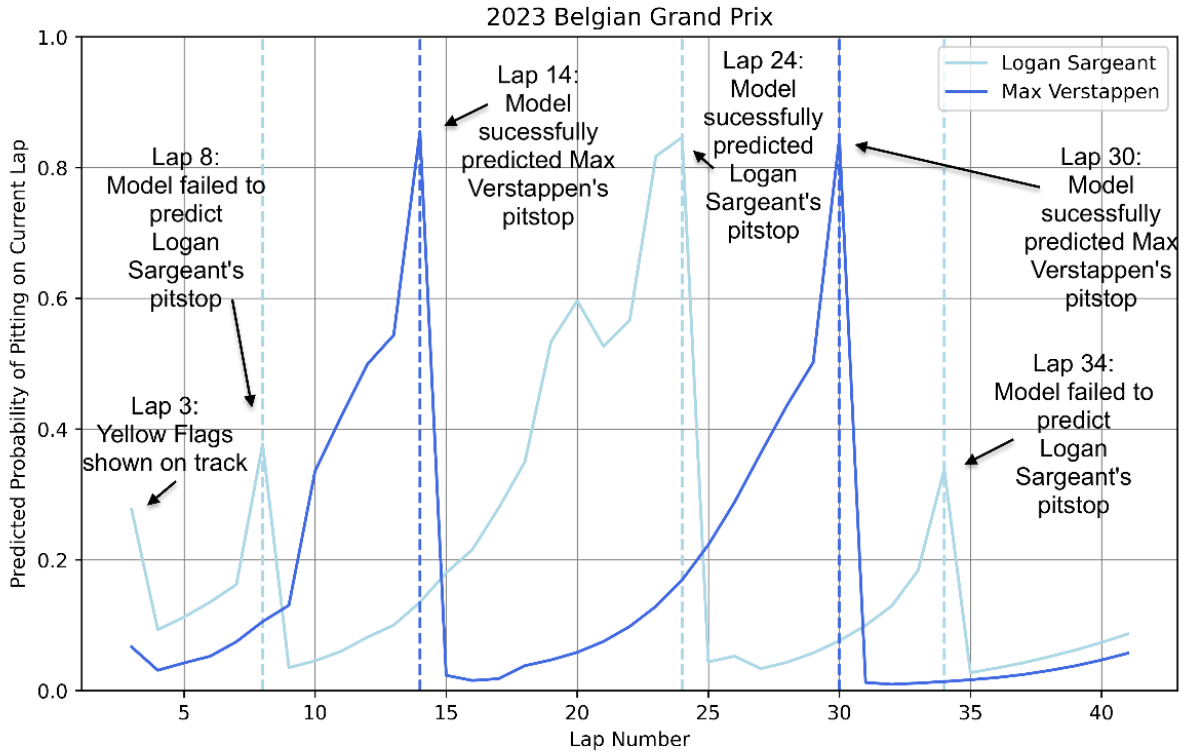


Figure 20, the following comments can be made:

- On the first lap of data passed to the model, lap 3, there were yellow flags shown on the track. This elevated the probability of both drivers pitting but by a much greater amount for SAR. It would not be the optimal strategy for a driver to pits so early in the race and so the true probability of either driver pitting was quite low. This situation is an example of how the model performs better on higher championship-placed drivers.
- The probability of SAR pitting remains higher than VER after the yellow flags, indicating indecision from the model on what SAR will do. The model then fails to predict the first pit stop for SAR on lap 8 with a high degree of confidence. On the lap that SAR pits the probability remains below 40%. For the first pitstop of VER, the model had reached a confidence level of 85% on the lap that VER did pit.
- The model does predict the second pitstop of SAR with a confidence of over 80% but the ramp-up in probability is much less aggressive. This less aggressive ramp-up in probability would be less useful to a Formula One strategist as it makes it more difficult to decide on a threshold for the model.

- For the third pitstop for SAR, the model once again fails to predict the pitstops with any significant confidence. This compares to the second pitstop of VER, which the model predicts with a confidence of over 80%. Furthermore, for both of VER's pitstops, the ramp-up in probability is aggressive. This sort of prediction would be much more useful to any Formula One strategist.
- Across the entire race, the predicted probability by the model is smoother for VER than SAR.

This race serves as a good example of how the model tends to perform better on drivers higher in the championship. This trend is seen across most races. When considering why this is, the following could be considered the reason.

Drivers at the top of the championship table will drive for the best teams. This means they will often have the fastest car which means the standard and optimal strategies can be used more often. For a driver at the backend of the championship, more experimental and risky strategies will often have to be used to give themselves a shot at scoring championship points. This is seen here in the SAR three-stop race, this is likely a suboptimal strategy but is different from the majority of other drivers. In being different, SAR has the opportunity to score points. The model seems to have learnt a better understanding of the actions of top drivers and so predicts their race strategy decisions reasonably well. This then makes it poor at predicting the actions of drivers on suboptimal risky strategies.

Example of Model Poor Performance

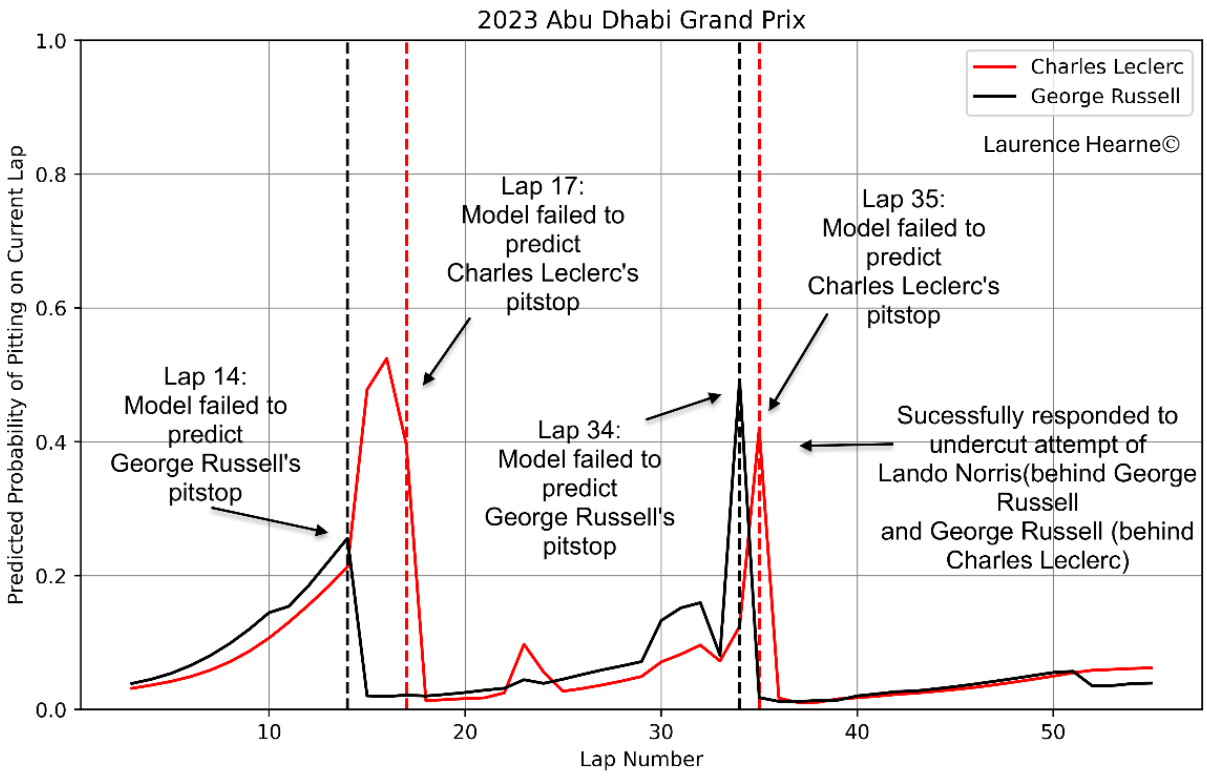


Figure 21: M10 predictions for the 2023 Abu Dhabi Grand Prix. This plot is an example of a situation where the model performed poorly in predicting pitstops with a good level of confidence.

Figure 21 shows an example of a situation where the model failed to predict the pitstops of two drivers with a meaningful level of confidence. This plot is related to the Abu Dhabi Grand Prix in 2023 with drivers Charles Leclerc seen in red and George Russell seen in black. Charles Leclerc will be referred to as LEC and George Russell as RUS. The following comments can be made regarding this plot:

- The model failed to predict the first pitstop for RUS on Lap 14, reaching a confidence of just over 25% on the Lap that RUS did pit. The first pitstop for LEC is predicted with greater confidence but still not to the level seen in previous plots. Furthermore, the probability of LEC pitting on Lap 17, the lap he did pit, was 10% lower than on Lap 16.

- The second round of pitstops for both LEC and RUS was better predicted than the first, with a very aggressive ramp-up in probability seen, however a high confidence is not reached. When RUS made his second pitstop he was attempting an undercut on LEC, the model did recognise this and responded by increasing the probability of LEC pitting.

The model fails to predict any of the pitstops for these drivers in this race with a confidence level comparable to the earlier plots shown. Both LEC and RUS did not have any issues during this race and made their pitstops at times that were deemed to be optimal.

Determining a threshold value

As with all probabilistic models, the difficulty often comes in determining a suitable threshold. To get a sense of the impact of varying the threshold, ROC (Receiver operating characteristic) curves were plotted. These plot the True Positive Rate (Sensitivity) versus the False Positive Rate (Specificity) as the threshold value is varied. The M10 model results were once again used to produce all the following plots.

For the first ROC plot, the test data from all drivers was plotted. This plot can be seen in Figure 22 below. The threshold values are marked along the curve. The plot of a purely random classifier is seen as the black dashed line. The M10 model for all drivers is seen as the blue line. The area under the curve (AUC) of the model is displayed in the legend to be 0.87. The ideal classifier would have an area under the curve of 1.0. To achieve this, the curve would have to reach up to the top left corner of the plot, thus this means that the closer the apex of the ROC curve is to the top left corner, the better the classifier is generally. With an area of 0.87, the classifier here is performing much better than a random classifier and is making useful predictions. ROC curves allow for a threshold value to be decided based on a trade-off between false positives (predicting a pitstop when there was not one) and true positives (successfully predicting actual pitstops). The general rule of thumb with regard to this is to draw a 45° angle down from the top left. The threshold value used at the point on the curve which the line intercepts is generally taken to be a good threshold value. In the ROC plot of

all drivers seen in Figure 22 below, the threshold value is around 0.45. This trade-off between false positives and true positives is generally domain-specific. It would take a domain expert to determine the optimal trade-off between the two and so an optimal threshold will not be decided on as part of this project. Instead, the ROC curves will be presented, with it up to the interpretation of a domain expert as to what the optimal trade-off would be.

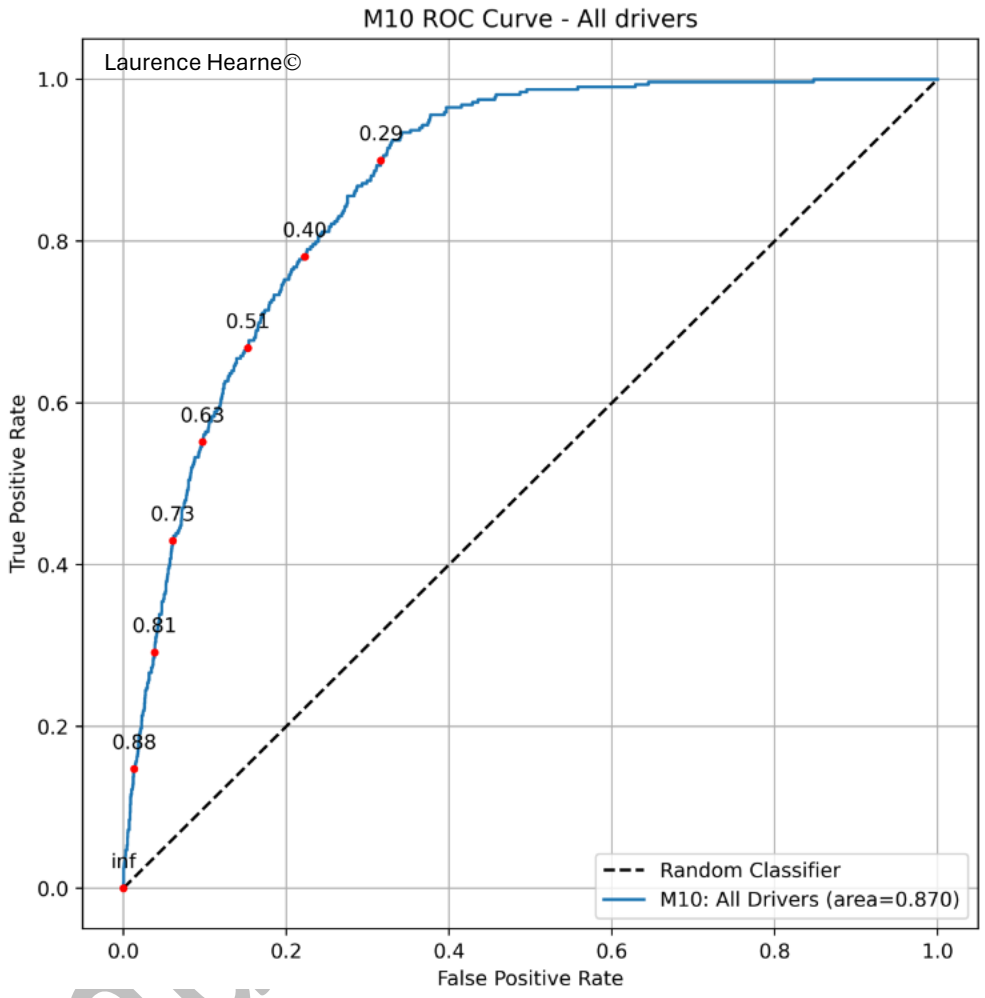


Figure 22: ROC Curve. Test Data from all drivers.

As was seen earlier in the results, the model generally performs better on top drivers. For this reason, an ROC curve was also produced including the data of only top drivers. The test data of Max Verstappen, Lewis Hamilton and Carlos Sainz was used in the ROC curve seen below in Figure 23. This plot further confirms the model's better performance on top drivers, with the AUC now 0.905. The apex of the curve also reaches further towards the top left, indicating

the classifier is closer to ideal for top drivers than it is for all drivers. In Figure 24, the two ROC curves are compared. This comparison demonstrates clearly that the model performs better on top drivers than it does on all drivers as a whole.

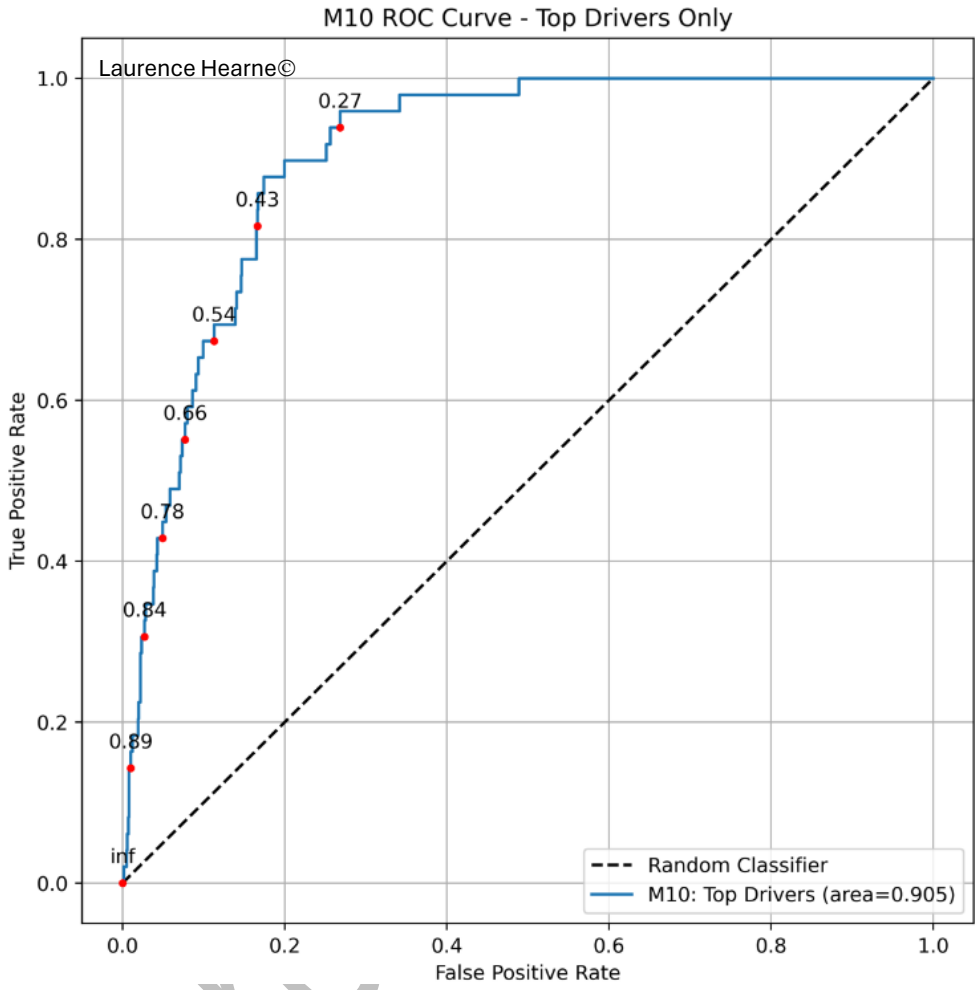


Figure 23: ROC Curve. Test Data from top drivers only.

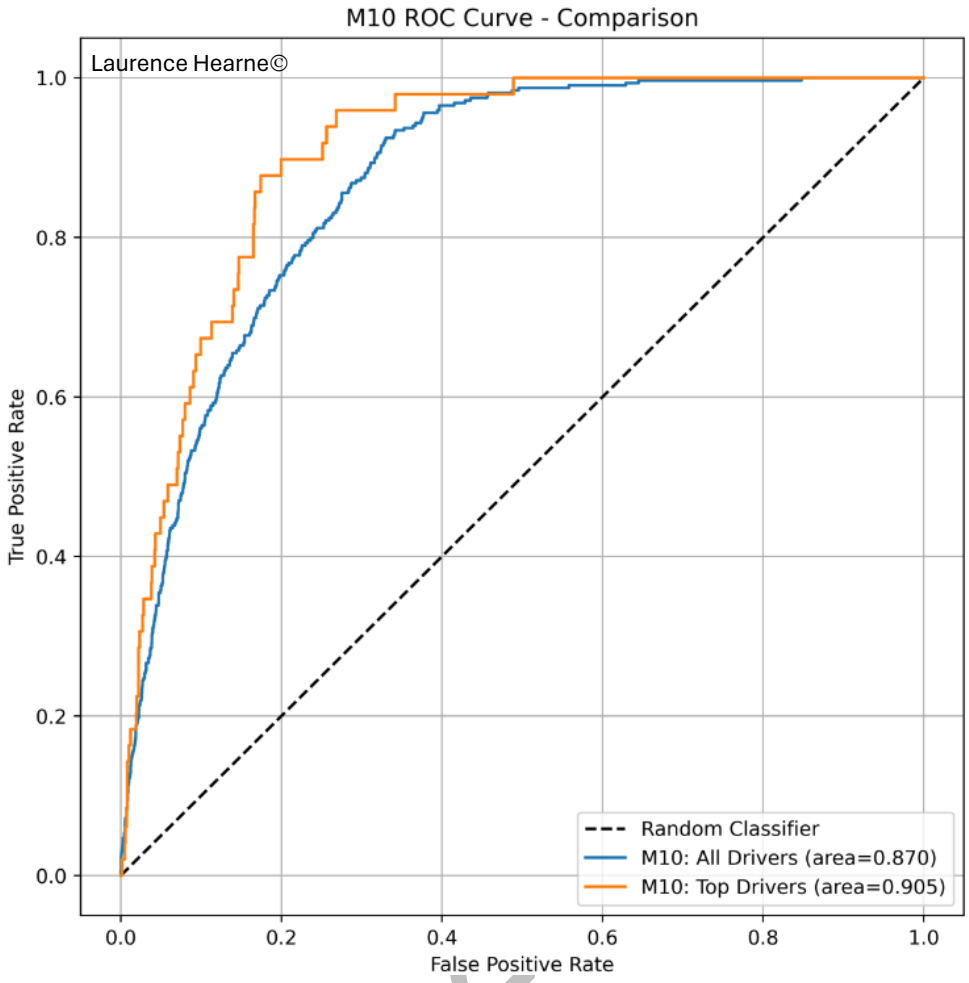


Figure 24: Comparison of ROC curves for All Drivers and Top Drivers.

Confusion Matrices

With a threshold value of 0.45, confusion matrices were produced for both all drivers and for the top drivers. The confusion matrix for all drivers can be seen in *Table 12* below. The data used for these is the test set once again. In *Table 13*, the Confusion matrices are given in percentage terms. A strong diagonal is seen here which is positive and demonstrates the model is providing useful predictions.

Table 12: Confusion Matrix. Threshold set to 0.45 on test set results. All Drivers.

		Prediction	
		Pitstop	**No Pitstop**
Truth	**Pitstop**	234	85
	No Pitstop	1628	6695

Table 13: Confusion Matrix. Threshold set to 0.45 on test set results. All Drivers. Percentage based.

		Prediction	
		Pitstop	**No Pitstop**
Truth	**Pitstop**	73%	27%
	No Pitstop	20%	80%

The model performance is once again seen to be stronger on Top drivers. The confusion matrices relating to just the top drivers from the test set can be seen in Table 14 and Table 15.

Table 14: Confusion Matrix. Threshold set to 0.45 on test set results. Top Drivers.

		Prediction	
		Pitstop	**No Pitstop**
Truth	**Pitstop**	38	11
	No Pitstop	234	1188

Table 15: Confusion Matrix. Threshold set to 0.45 on test set results. Top Drivers. Percentage Based.

		Prediction	
		Pitstop	**No Pitstop**
Truth	**Pitstop**	78%	22%
	No Pitstop	16%	84%

Conclusions and Future Work

The model created during the project displayed an ability to predict the pitstops of drivers with a good level of confidence. The model successfully handled safety car situations, a common strategic situation. Furthermore, the model was able to identify when a driver was at risk of being undercut and acted appropriately. As seen in the results however the model was not perfect, this was exemplified by the 2023 Abu Dhabi Grand Prix, where the model failed to predict the pitstops of drivers with a high degree of confidence. A highlight of the model's traits was its ability to predict the pitstops of top drivers more accurately. This is very positive as top drivers would generally follow more optimal strategies and so the model has learned how to do “good” race strategy.

Overall, the model was a good classifier with it far outperforming a random classifier. It is very possible to see how the model could be used as an additional tool by a Formula One team when they are making race strategy decisions live during a race. As mentioned previously, with the vast amounts of extra data available to a Formula One Team this model could be further improved. Along with this, the extra technical capabilities of a Formula One team would benefit the model’s performance.

The future work of this model would be to retrain the model with a larger dataset. A second dataset was found during the project, with race data back as far as 2014, it was decided not to include this data in the project's model due to issues around merging the dataset, however in the future, this could be done to improve the model. In the future, an LSTM (long short-term memory) layer could be added to the model. This would mean that the model would have information on previous laps when predicting the probability of a pitstop. This would make sense as when Formula One strategists are making decisions on race strategy they consider multiple laps at a time. Finally, I believe merging an ML model with a race strategy simulator would be the optimal solution. In doing this, it would allow for the implications of the model predictions to be evaluated in real-time. This could be used in a reinforcement learning configuration, with the model constantly learning.

I have thoroughly enjoyed this project and have acquired a multitude of knowledge and skills from working on it. These include data analysis, machine learning model development, project management and report writing.

I would like to thank my project supervisors, Associate Prof Patrick Denny and Prof Pepijn Van de Ven, for their constant guidance and support on the project. They have provided great knowledge and advice to me throughout the project and the project would not have been possible without them.

References

[1] W. Racing, "Everything you need to know about F1 tyres in 2023." [Online]. Available: https://www.williamsf1.com/posts/faf938a0-663c-4ccc-b76b-199433864f6b/everything-you-need-to-know-about-f1-tyres-in-2023

[2] F. Chronicle, "Why Do F1 Cars Need Pit Stops?." [Online]. Available: https://f1chronicle.com/why-do-f1-cars-need-pit-stops/

[3] M. Morlidge, "Formula 1 2022: Explaining Ferrari's blunders and Charles Leclerc's dwindling Formula 1 title dreams." [Online]. Available: https://www.skysports.com/f1/news/12433/12663464/hungarian-gp-explaining-ferraris-latest-blunder-and-charles-leclercs-dwindling-formula-1-title-dreams

[4] motorsport, "INSIDER'S GUIDE: HOW F1 RACE STRATEGY WORKS." [Online]. Available: https://www.motorsport.com/f1/news/how-f1-race-strategy-works/6791893/

[5] C. Sulsters and R. Bekker, "Simulating formula one race strategies," *Vrije Universiteit Amsterdam,* 2018.

[6] A. Heilmeier, A. Thomaser, M. Graf, and J. Betz, "Virtual Strategy Engineer: Using Artificial Neural Networks for Making Race Strategy Decisions in Circuit Motorsport," *Applied Sciences,* vol. 10, no. 21, p. 7805, 2020. [Online]. Available: https://www.mdpi.com/2076-3417/10/21/7805.

[7] E. Stoppels, "Predicting race results using artificial neural networks," ed, 2017.

[8] T. Tulabandhula, " Changes, Fresh Air, and Yellow Flags: Challenges in Predictive Analytics for Professional Racing," *Big Data,* vol. 2, no. 2, pp. 97-112, 2014, doi: 10.1089/big.2014.0018.

[9] M. Davide, "S-MARL: An Algorithm for Single-To-Multi-Agent Reinforcement Learning: Case Study: Formula 1 Race Strategies," ed, 2023.

[10] D. Papathanasiou, K. Demertzis, and N. Tziritas, "Machine Failure Prediction Using Survival Analysis," *Future Internet,* vol. 15, no. 5, p. 153, 2023. [Online]. Available: https://www.mdpi.com/1999-5903/15/5/153.

[11] G. Gartheeban and J. Guttag, "A data-driven method for in-game decision making in MLB: when to pull a starting pitcher," presented at the Proceedings of the 19th ACM

SIGKDD international conference on Knowledge discovery and data mining, Chicago, Illinois, USA, 2013. [Online]. Available: https://doi.org/10.1145/2487575.2487660.

[12] M. A. P. F. Team. "What Goes into F1 Strategy?" https://www.mercedesamgf1.com/news/what-goes-into-f1-strategy (accessed October, 2023).

[13] theOerly. "FastF1." https://docs.fastf1.dev/ (accessed.

[14] *pandas-dev/pandas: Pandas*. (2020). Zenodo. [Online]. Available: https://doi.org/10.5281/zenodo.3509134

[15] J. Brownlee, *Data Preperation for Machine Learning. Data Cleaning, Feature Selection, and Data Transforms in Python.* 2020.

[16] S. A. Dyer and J. S. Dyer, "Cubic-spline interpolation. 1," *IEEE Instrumentation & Measurement Magazine,* vol. 4, no. 1, pp. 44-46, 2001.

[17] Pirelli. "Emotions and Numbers." https://www.pirelli.com/global/en-ww/emotions-and-numbers/ (accessed.

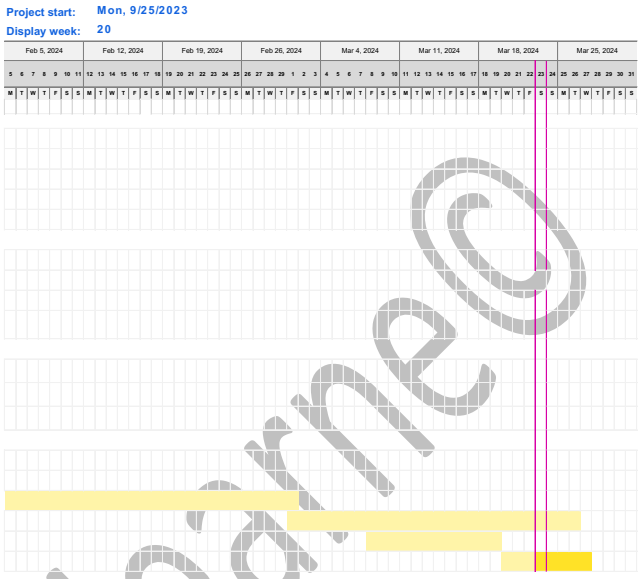
[18] F. a. V. Pedregosa, G. and Gramfort, A. and Michel, V., B. a. G. and Thirion, O. and Blondel, M. and Prettenhofer, P., R. a. D. and Weiss, V. and Vanderplas, J. and Passos, A. and, and D. a. B. Cournapeau, M. and Perrot, M. and Duchesnay, E., "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research,* vol. 12, pp. 2825--2830, 2011.

[19] P. J. Rousseeuw, "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis," *Journal of Computational and Applied Mathematics,* vol. 20, pp. 53-65, 1987/11/01/ 1987, doi: https://doi.org/10.1016/0377-0427(87)90125-7.

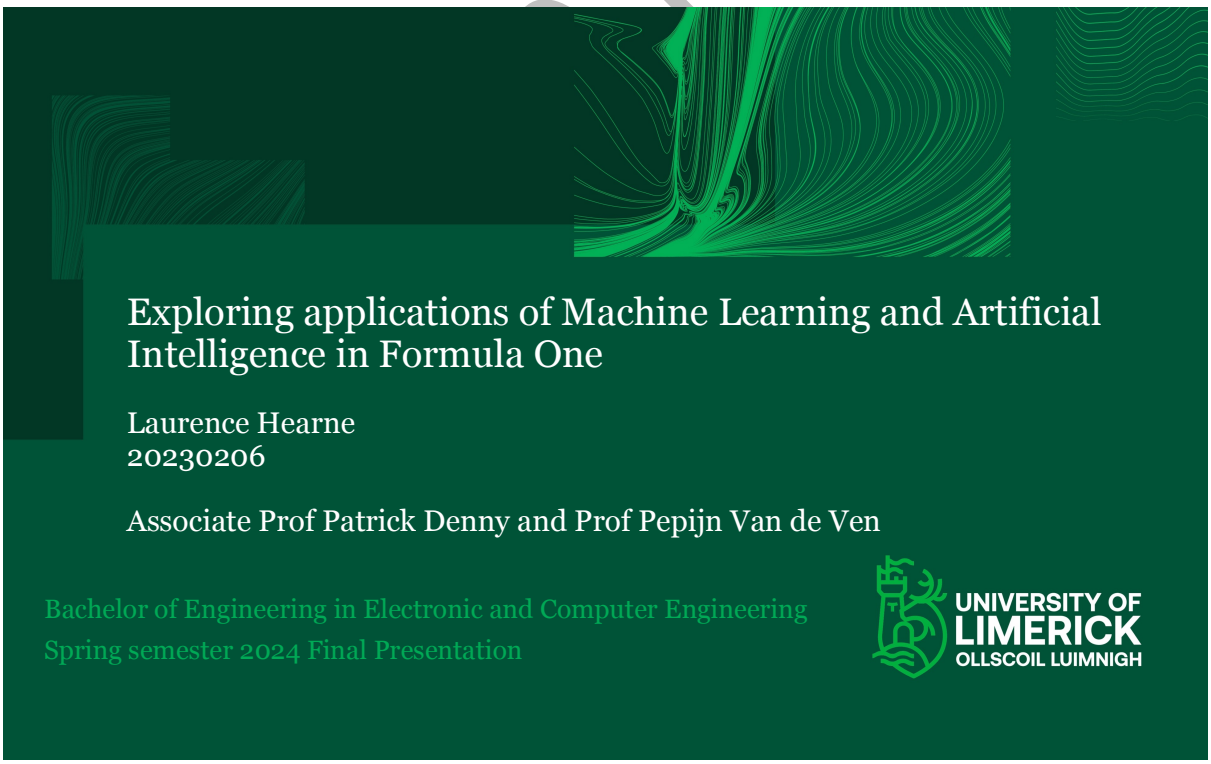
[20] F. One. "2023 Constructor Standings." https://www.formula1.com/en/results.html/2023/team.html (accessed 26th October, 2023).

Appendix A: Updated Project Gantt Chart

TASK	ASSIGNED TO	PROGRESS	START	END
Project Organisation				
Project Definition	Laurence	100%	25-Sep-23	28-Sep-23
Background Research	Laurence	100%	28-Sep-23	30-Sep-23
Identify Steps	Laurence	100%	30-Sep-23	04-Oct-23
Planning	Laurence	100%	04-Oct-23	29-Sep-23
Python Education	Laurence	100%	17-Sep-23	30-Sep-23
Dataset Activities				
Data Acquisition	Laurence	100%	14-Sep-23	21-Sep-23
Data cleaning	Laurence	100%	30-Sep-23	08-Oct-23
Exploratory Data Analysis	Laurence	100%	09-Oct-23	18-Oct-23
Selection of features and ML Algorithm	Laurence	100%	18-Oct-23	27-Oct-23
Model Activities				
Pre-Processing data for training	Laurence	100%	30-Oct-23	01-Dec-23
Model Training/Evaluation Cycles(Part One)	Laurence	100%	20-Nov-23	01-Dec-23
Model Training/Evaluation Cycles(Part Two)	Laurence	100%	08-Jan-24	28-Jan-24
Report Writing				
Interim report Writing	Laurence	100%	16-Oct-23	27-Oct-23
Interim Presentation Writing	Laurence	100%	27-Oct-23	01-Nov-23
Final Report First Draft	Laurence	100%	17-Jan-24	01-Mar-24
Final Report Second Draft	Laurence	100%	01-Mar-24	26-Mar-24
Poster Creation	Laurence	100%	08-Mar-24	19-Mar-24
Final Presentation Writing	Laurence	40%	20-Mar-24	27-Mar-24



Appendix B: Final Presentation Slides



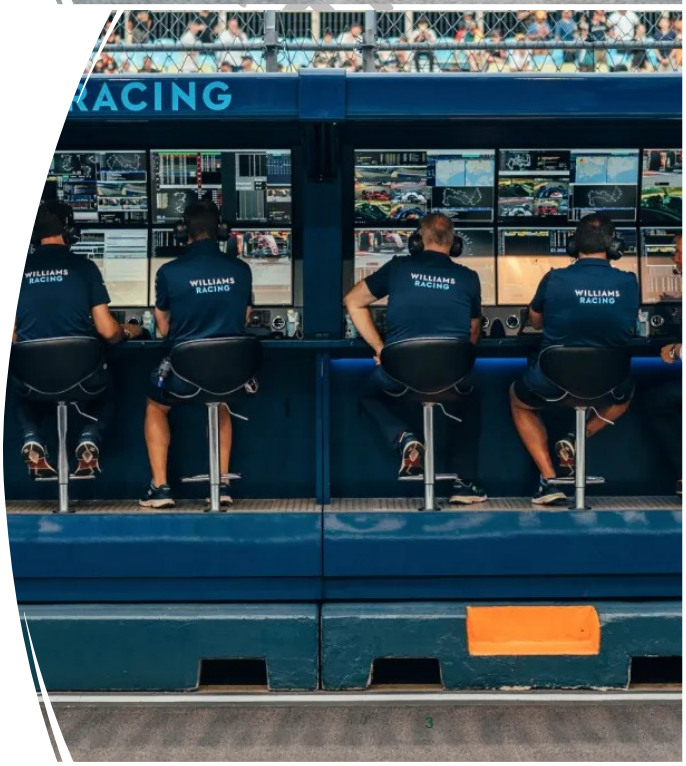
Presentation Overview

- Introduction to the problem faced by a Formula One strategist.
- Methodology:
  - Data acquisition.
  - EDA and Data pre-processing.
  - Model Building.
- Final Implementation.
- Model Results and analysis.
- Conclusions and future work.
- Acknowledgements.

2

Project overview

- The Fundamentals of Formula One strategy.
- Problem statement: “When do I call my drivers in for a pitstop?”
- Current solution to the problem.
- Build a Machine Learning Model that will produce a probability of a driver pitting on each lap of a race.



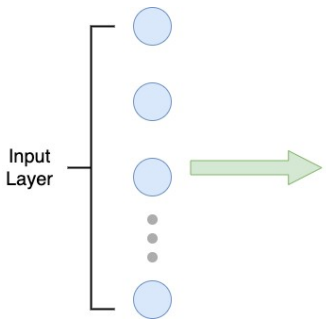
3

Methodology

- Acquired race data between 2018 and 2023.
- Cleaned data. Removed wet races.
- Exploratory Data Analysis performed.
- Data pre-processing.
- Model developed in TensorFlow.
- Model Analysis.

4

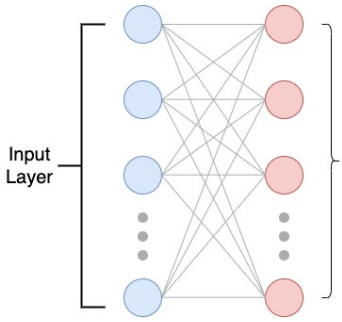
Neural Network



Feature	Value Range
RaceProgress	[0.0, 1.0]
TyreLifeProgress	[0.0, 1.0]
Position	[0.0, 1.0]
Compound	[SOFT, MEDIUM, HARD]
TrackType	[T0, T1, T2, T3, T4]
TrackStatus	[0, 1, 2, 3, 4]
CarClose	[True, False]
DriversChampionshipPosition	[0.0, 1.0]
TwoTyreCompoundsUsed	[True, False]
PitStopBehind	[True, False]

5

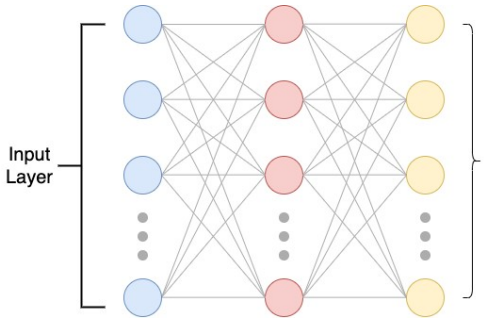
Neural Network



Hidden Layer 1:
- 32 densely connected neurons.
- ReLU activation functions.
- Kernel Regularizer: l2(0.003)

6

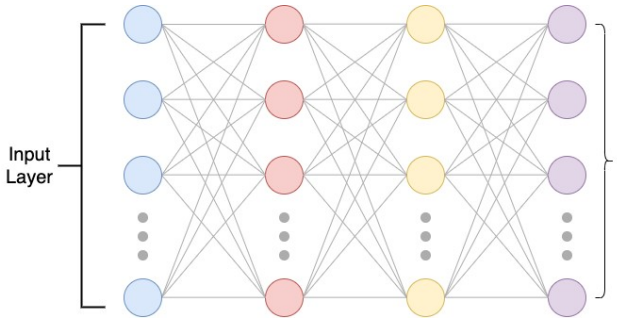
Neural Network



Hidden Layer 2:
- 64 densely connected neurons.
- ReLU activation functions.
- Kernel Regularizer: l2(0.003)

7

Neural Network

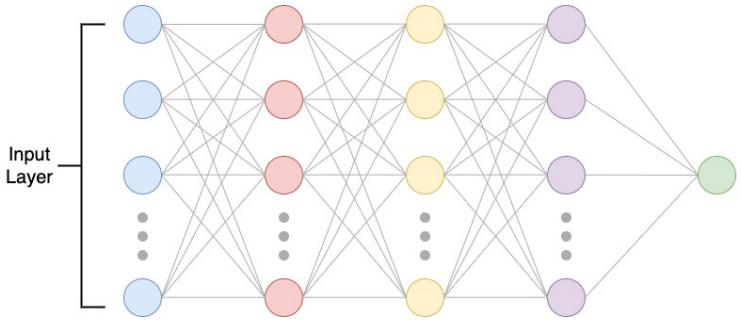


Hidden Layer 3:
- 32 densely connected neurons.
- ReLU activation functions.
- Kernel Regularizer: l2(0.003)

8

UNIVERSITY OF LIMERICK
OLLSCOIL LUIMNIGH

Neural Network

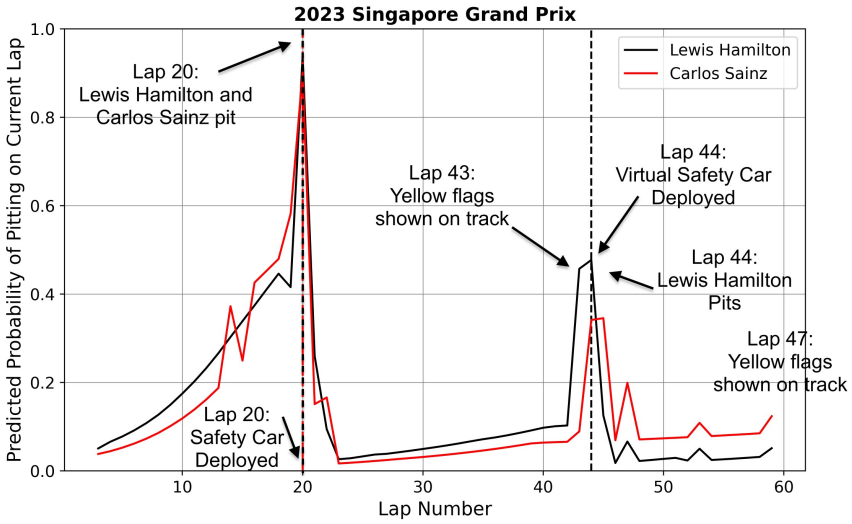


Output Layer:
- 1 densely connected neuron.
- Sigmoid activation function.
- Probabilistic Output.
- Adam Optimiser.
- Class weights.

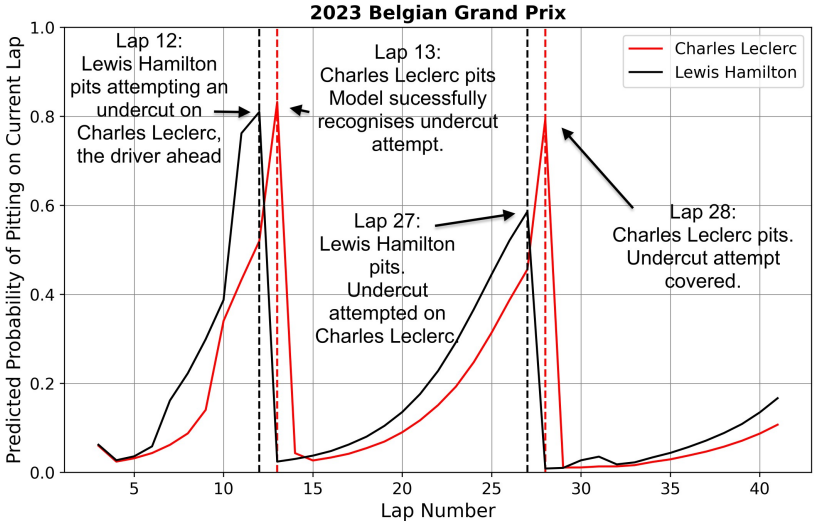
9

UNIVERSITY OF LIMERICK
OLLSCOIL LUIMNIGH

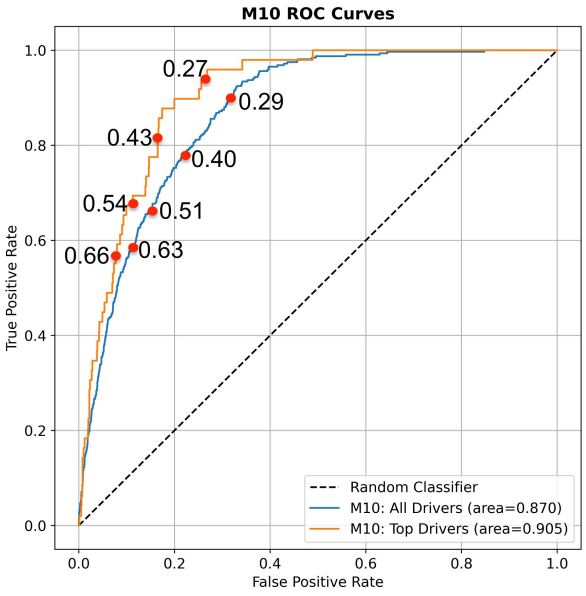
Test Set Results



Test Set Results



ROC Curve



12

Potential uses of the system and future work

- Model could serve as another tool in the toolbox of a F1 Strategist.
- Connect model to simulation program.
- Reinforcement learning configuration.
- Project results have been sent to Redbull Racing.

13

Conclusions

- Thoroughly enjoyed the project.
- Skills learnt:
  - Data processing and Analysis.
  - Machine Learning Model Development.
  - Project Management.
  - Report Writing.
- Thank you to my supervisors, Associate Prof. Patrick Denny and Prof. Pepijn Van de Ven

14

Thank you.
Any questions?

UNIVERSITY OF LIMERICK
OLLSCOIL LUIMNIGH

University of Limerick,
Limerick, V94 T9PX,
Ireland.
Ollscoil Luimnigh,
Luimneach,
V94 T9PX, Éire.
+353 (0) 61 202020

ul.ie