A Visual Analytics Tool to Analyze F1 Drivers' Performances in Races

Federico Perotto

Abstract

The following paper describes a system developed to examine and analyze the unfolding of Formula 1 races as well as the performances of the drivers during those races.
The project relies on the FastF1 python package to access the results, schedules, timing data and telemetry of the races of the 2023 season. While the data provided is not perfect and somewhat limited, It is more than enough to produce comprehensive and meaningful metrics, as well as a summary of race progression, race pace, and strategies.
The aim of the project is to create a tool for journalists, pundits and content creators to form, check and present their opinions or content pieces by providing an insight into drivers' statistics over the race, aiding in identifying factors that contributed to the race result, such as strategy choices or outlier performances.

1 Introduction

Formula 1 is widely considered as the pinnacle of motorsports, where high-performing athletes and incredibly advanced engineering combine in the pursuit of every possible advantage to achieve victory.

As such all Formula 1 teams are always looking at every possible aspect of a race and of a driver to save every possible millisecond. This inevitably means collecting a lot of data. Formula 1 cars are equipped with hundreds of sensors to log every possible interaction between the driver, the car, the track and the environment, resulting in hundreds of gigabytes of data per race weekend.

Unfortunately, the vast majority of this data is kept as a closely guarded secret by all teams, and the general public, as well as journalists, pundits and commentators have access to a limited amount of superficial data. Therefore, in order to form opinions, articles, videos and other content, they have to rely either on what the teams report about themselves, or on the incomplete view offered by a race broadcast, in which it is essentially impossible to keep track of every factor and decision that is going on at all times.

The goal of this project is to provide a tool that can fill this gap in the access to information and that be used mainly by journalists, pundits, and content creators to have a holistic view of a race. The users will be able to interact with this dashboard, revisiting the races and gaining understanding and insight into different aspects and factors of any race, and would then be able to create new opinions, check their existing assumptions against real data or also use the dashboard itself to present and accompany the content pieces.

2 Related Works

This project fits in the large field of sports data visualization and analysis. Research in this field deals with designing new visualization techniques, adapting existing techniques to new domains as well as evaluating solutions. Sports data is particularly well suited for research in the field of visual analytics, since it has to deal with a vast amount of multi-variate and heterogeneous data which contains static and dynamic patterns that can be discovered. Being able to identify trends, correlations, clusters or anomalies and outliers helps understanding these events, but it also quite challenging to capture the complex nature of a sport and its dynamics and rules.

In the paper State of the Art of Sports Visualization[1], the authors conducted an analysis of research contributions to the field of sports data visualization from 98 papers and articles and they identified three categories of sports data:

- *Box Score data:* consists of statistical summaries of a sport event, and refers to in-game events dictated by the sport rules, such as goals, fouls etc.
- *Tracking data:* consists of precise spatio-temporal information about players and equipment captured in real time during the course of play, utilizing the ever-growing capabilities of technology and in particular machine vision systems.
- *Meta-data:* consists of additional data that surrounds a sport event and can add context, such as data about rules or physical characteristics of players.

The article goes on to describe techniques often used to represent these types of data, providing and highlighting useful solutions and techniques for each of them, while also noting that a lot of these solutions and techniques do not include all types of data in the same visualization.

Spatio-Temporal Analysis of Team Sports [2] focuses exclusively on the spatio-temporal data that pertains to team sports and proposes and implements an algorithmic approach to perform non-trivial computations on the data to identify structures within it. This paper produces data models and visualizations that are quite advanced but have a strong focus on the team aspect and on the attack or defend aspect of those sports.

Moving to works focused on Formula 1 and racing, the literature is surprisingly sparse.
In the article A Web-Based Tool for analysing Formula 1 Races and Seasons [3] the authors present a web-based tool for analysing single Formula 1 races and seasons. Unfortunately this tool is no longer available at the URL they provide, but they specifically mention that the tool is developed with non-expert users in mind, and by the images provided, the visualizations appear quite simple, but they also include a way to compare multiple races, as well as the overall results during a season.

The work described in this thesis [4] is arguably the closest to this project in terms of goals. The author has access to a vast amount of data since He is able to collect the data directly from the sensors installed on a Formula Student car, and his focus is on a single lap performance rather than a whole race. Nonetheless, many of the metrics and visualizations proposed could be extrapolated and extended to be representative of a whole race.

Finally, there are some fan-made web-based tools such as F1 Dash[5] and F1 Tempo[6] that utilize the same data source as this project. While they do not seem to have any particular goal, they provide some examples of what can be done with the same dataset, and some visualization techniques.

3 Dataset

As mentioned before, the majority of the data is obtained through the FastF1 python package. The package provides a collection of functions and data objects to access and analyze Formula 1 timing and telemetry data, and the data is provided in the form of Pandas DataFrames.
I set up a python notebook that retrieves multiple data objects for all the races of 2023. The data objects are then converted into CSV files. In particular, for each race We retrieve:

- *Laps:* this object contains detailed information about the lap timings for all the drivers in the race, such as lap time, sector times, lap number, track status, current position, tyre compound and stint number.
- *Telemetry:* this object is a dictionary. Each key of the dictionary correspond to a driver, and each value of the dictionary contains the telemetry of the car, in the form of a DataFrame, which includes speed of the car, gear, RPM of the engine and application of throttle or breaking. The data is sampled multiple times per second for the whole duration of the race. The telemetry of each car is saved in separate CSV files.
- *Results:* this object contains information about the results and the drivers that participated in the selected race, such as starting position, finishing position, team name, driver number and abbreviation.
- *Weather Data:* this object contains information about the weather conditions, air temperature, track temperature and wind during the race.

Additionally, to obtain data about the exact duration of pit stops, I used the information provided by Formula 1 official partner DHL at this site. It features tables of all the pit stops for every race, which have also been converted to CSV files.

4 Preprocessing

The majority of the data, specifically the *Laps*, *Telemetry*, *Results*, and *Pit stops* data is simply converted to CSV files and passed to the application, where the code then handles it. As I have noted before, the data is not perfect, so the code actually handles some edge cases and fixes the data where it can.
A notable exception to this approach however is in how the CSV file for the scatter plot is created. In fact, since this file is meant to contain the result of a dimensionality reduction technique, I decided to aggregate the data and apply the dimensionality reduction technique by using python and the sklearn library. To accomplish this, I set up a python notebook in which I once again utilize the FastF1 package to access the races. For each race, It fetches the laps data and loads it into a DataFrame. Then, for each lap in the race, It gets the telemetry and weather data that correspond to that lap and computes various metrics from this data: average speed, maximum speed, minimum speed, average throttle level, percentage of lap spent at full throttle, percentage of lap spent braking, air temperature, track temperature, wind direction, and wind speed. This data is then appended to the original dataset.
Afterwards the data is cleaned: rows corresponding to laps where the driver pitted are removed, as those laps are obvious outliers. Several columns that are not needed for the analysis are also removed. Laps in which the drivers aren't able to race (i.e., the track status of the lap is not 1) are removed. Finally rows with missing data are removed and time-related columns are converted from timedelta64 to floats.
At this point the dimensionality reduction technique is applied. The data is scaled using StandardScaler and then t-SNE (t-Distributed Stochastic Neighbor Embedding) is applied. The result is a 2-dimensional representation of the data.
The results are then appended back to the original DataFrame which is then saved to a CSV file.

5 Visualizations & Interactions

The system is composed of 5 views. Each of the views offers some kind of interaction, meant to aid the user in refining the analysis of the race, its different moments and relative metrics. On top of the interactions implemented in each view, there are a few global interactions.

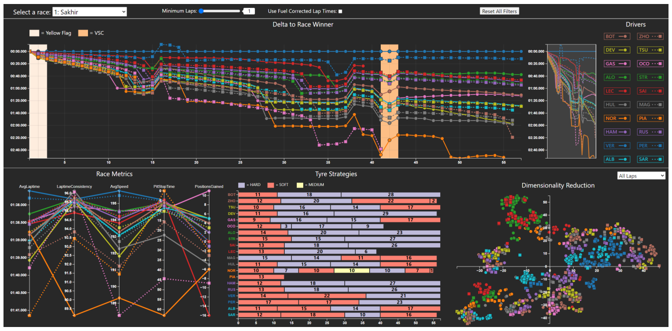

Figure 1: Complete View of the Dashboard

- *Dropdown:* through the drop down menu in the top left, the user can choose which race to analyze. Upon selecting a race, the dataset corresponding to that race will be loaded, and the dataset will be used to compute the necessary values to populate all the views.
- *Minimum Laps Slider:* this range slider can be used to set a required minimum amount of laps completed to be visualized in any of the charts.
- *Fuel Correction Checkbox:* this checkbox allows the user to decide if they want to use fuel corrected lap times, which rely on a few assumptions and are therefore not completely reliable, but can be useful to compare stints that happened at different stages of the race.
- *Reset:* in the top right corner, the button can be used to reset all the filters and interactive elements to their default status.

5.1 Drivers Legend

5.1.1 Visualization

This view serves as a global interactive legend. It displays the abbreviations of all the drivers that participated in the selected race, and next to the abbreviations it shows how each driver is encoded in the other charts: either through a solid line and a circle, or through a dashed line and a square. Furthermore, the color of each element represents the team of the driver.

5.1.2 Interactions

- *Click:* By clicking one of the drivers the user can switch the status of the driver from included to excluded, or vice versa. If the users chooses to exclude a driver, the other views will update to hide the elements corresponding to that driver. Similarly, if a user decides to include a driver, the other views will include the elements that correspond to that driver. By default all the drivers are included.

Figure 2: Drivers Legend

- *Hover:* If a driver is excluded and the user hovers its mouse over it, all the charts will show the elements corresponding to that driver until the mouse is moved off.

5.2 Line Chart + Context

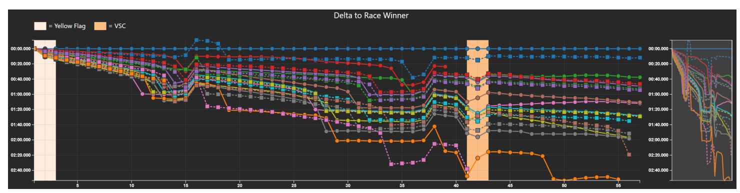

Figure 3: Line chart

5.2.1 Visualization

The line chart is used to visualize the time delta between each driver and the race leader during the progression of a race. Each line corresponds to a driver, and the time difference is computed at the start of every lap. Every lap of every driver is highlighted by either a circle or a square, according to their encoding shown in the drivers legend.
This chart provides insight into the race evolution, the consistency of performances and the impact of strategies. The slope of the lines indicates whether a driver is gaining or losing time relative to the leader, but also relative to the other drivers, which allows the user to compare their performances, and the lines also represent the current position of the drivers, which add context to the strategy chosen by teams and drivers.

In the line chart, colored rectangles are also used to denote the track status at different points of the race, and a corresponding legend is implemented to explain the meaning of these rectangles. These rectangles are relevant to explain the progression of a race, as they can represent strategic options or opportunities for the drivers and also provide context to some of the sudden jumps in the lines (the other sudden jumps are due to pit stops).

Next to the line chart, a context overview is provided. Through an interactable vertical brush, this section provides a simplified view of the line chart and what part of it is currently visible.

5.2.2 Interactions

- *Hover:* By hovering the mouse over a driver's circle or square, a tooltip is created which contains information about that driver's lap, such as lap number, current delta to the winner, lap time and current position.
- *Zoom:* By using the scroll wheel on the line chart, the user can zoom the chart along the y axis. This is useful to avoid having outlier performances reduce the readability of the chart, and also to better highlight trends in the data that may not have been as clear before zooming. The zoom level and the area in view are reflected in the brush area of the context chart.
- *Brushing:* The user can brush over the context chart, and the line chart will update to visualize the selected area.

5.3 Parallel Coordinates

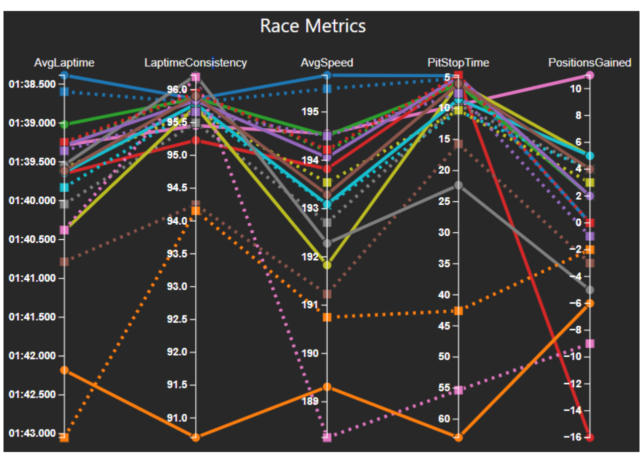

Figure 4: Parallel Coordinates

5.3.1 Visualization

The parallel coordinates plot allows to analyze and compare multivariate data.
In this case, each vertical axis represents a performance metric relevant to the race. Each driver is represented by a line that intersects each axis at a point corresponding to their data value for that metric. The point of intersection is highlighted by either a circle or a square. The different drivers are differentiated by using different colors, line styles and shapes, as shown in the drivers legend.
The axis have been oriented so that 'good' values for each metric are on the top side, which makes grouping performances or identifying outliers easier.
This technique can be useful to identify or highlight patterns, clusters or outliers, which reflect in the evaluation of the drivers' performances.

5.3.2 Interactions

- *Hover:* Upon moving the mouse over any data point, a tooltip is created and displayed. The tooltip provides the precise numerical value of the data point.
- *Brushing:* The user can interact with each axis by brushing. The drivers whose metrics fall outside the selected range of any of the axes are then excluded from the visualization. To determine which drivers are included or excluded, the system considers also the drivers that have been included or excluded through the drivers legend.

5.4 Stacked Bar Chart

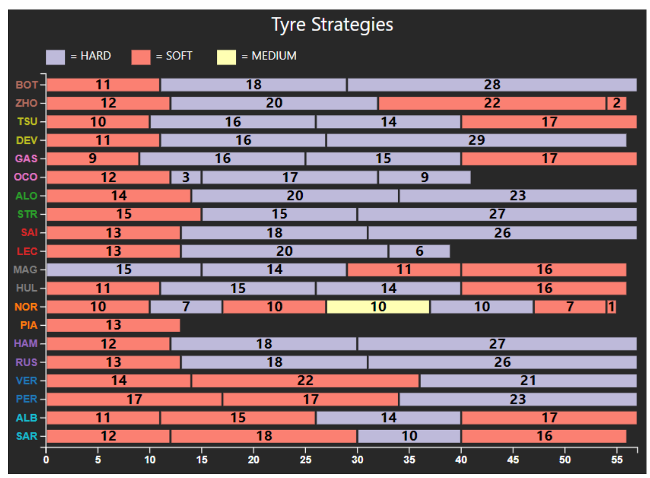

Figure 5: Stacked Bar Chart

5.4.1 Visualization

The stacked bar chart is used to visualize the tyre strategies chosen by the drivers and teams during the course of the race, and can help understand and contextualize how these choices impacted the drivers and their performances.
Each bar corresponds to the strategy of one driver, and its length corresponds to the number of laps the driver completed.
Each bar is divided in multiple segments (*stints*) that

represent the tyre compounds used by a driver over the course of the race. The color of the segments indicates the compound of the tyre, and the length of the segments corresponds to the number of laps that the driver spent on that compound (*stint length*). The stint length is also written inside of the bar segment, for easier comparisons between stints.
This visualization should allow the user to compare the different tyre strategies and to identify common choices or unique approaches, while also granting insight on the ability of drivers to manage or utilize the different compounds.

5.4.2 Interactions

- *Hover:* When hovering over any of the segments of the bars, a tooltip is created that displays information relative to that stint, such as the driver, the laps of that stint, and the compound of the tyre.

- *Click:* The user can click on any of the stints of any of the driver to dynamically update the parallel coordinates plot and the line chart. In particular, in the parallel coordinates plot, the metrics will be recomputed and the plot will be updated to include only the data of the selected stints, and the line chart will be redrawn to only include the laps of the selected stints.

5.5 Scatter Plot

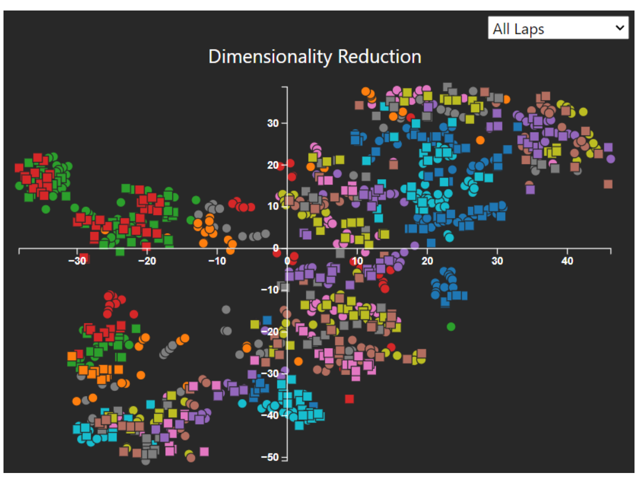

Figure 6: Scatter Plot

5.5.1 Visualization

The scatter plot is used to visualize the result of the t-SNE dimensionlity reduction technique. The goal for this chart is to allow the user to identify patterns or clusters in a two dimensional representation of the data. The method used to obtain the data for this visualization is described in the *preprocessing* section, and each element in the scatter plot represents a single lap from one of the drivers

Although t-SNE can introduce both false positives and false negatives, the relative position of each element should correlate to a measure of similarity. Distant and isolated elements are outliers and close elements are similar laps.

5.5.2 Interactions

- *Dropdown:* Using the dropdown menu, the user can switch between viewing all the laps and viewing the centroid of their distributions. This option gives the user the possibility of viewing an overall evaluation of the similarity of the performances of two drivers over the whole race.

6 Analytics

The analytics part of this project is focused on dynamically computing the drivers' metrics in response to the user's interaction with the system. In particular, the metrics computed are the *delta*, which is used to draw the line chart, and the *AvgLaptime*, *LaptimeConsistency*, *AvgSpeed*, *PitStopTime* and *PositionsGained* which are used in the parallel coordinates plot, as well as the centroids of the distribution of the laps in the scatter plot.
The *AvgLaptime* is computed for every driver as the average of the lap times of all the *racing* laps, i.e. laps in which there weren't any track status issues, such as yellow flags or safety cars.
The *LaptimeConsistency* is computed for every driver as the standard deviation for the laptime of the racing laps (the same laps considered for the average lap time) divided by the mean. This result is called coefficient of variation. This value is multiplied by 100 to be expressed as a percentage and subtracted from 100 to express it as a consistency metric.
The *AvgSpeed* is computed for every driver by parsing through their car's telemetry data and computing the mean of all the speeds recorded during racing laps. Each of the telemetry files is 30k-40k lines, although some lines are not considered because the telemetry polling starts from before the start of the race.
The *PitStopTime* for every driver is simply computed as the sum of their pitstops duration.
PositionsGained for every driver is simply computed as the difference between the starting and finishing position.

These metrics are computed whenever the user chooses a new race through the dropdown menu, but also in response to other events.

- By checking the fuel correction checkbox, the system will re-compute the *AvgLaptime* and update the data in the parallel coordinates. The fuel correction computation assumes that every driver starts the race with the maximum allowed load of fuel (110 Kg) and finishes the race with no fuel, and that each kg of fuel costs 0.03 seconds per lap. From these assumptions, we get the time loss per lap as the fuel loss per lap times the time loss per kg. The laptime is then adjusted considering this timeloss and the current fuel load of the car (i.e. the current lap number).

- When clicking any of the stints in the bar chart, the whole data for the parallel coordinates plot will be re-computed.
  The *AvgLaptime*, *LaptimeConsistency* and *AvgSpeed* are re-computed considering just the racing laps that are part of the selected stints; The *PitStopTime* only considers pitstops that have occurred at the end of the selected stint; The *PositionsGained* is computed as the difference between the position at the start and at the end of the stint.

7 Insights

The system is designed to provide a multifaceted visualization and understanding of drivers' performances and dynamics in a Formula 1 race. By using the filters and interactions provided in the charts, the user can refine his analysis of the race.

By setting a minimum amount of laps, the user can exclude from the visualization the drivers who didn't finish the race and often appear as outliers in the charts.

The line chart provides a two dimensional view of the progression of the race and present various key information about each driver. First of all the relative distance between drivers is proportional to the actual distance on the track, and the order in which the drivers are presented is the same order as the drivers on track. Furthermore, the slope of the lines are a representation of a trend in terms of laptimes and therefore performances. A line that trends upwards is indicative of a better performance than the other drivers, and likewise a downwards trend is indicative of a bad performance. A line that stays at the same distance from his competitors can be indicative of similar performance

The parallel coordinates plot presents some insights implicitly. The patterns created by the lines can highlight positive or negative correlation or variance between the metrics, as well as isolating outliers.Then brushing over any of the axes, the user can decide which metric and which results are relevant to his analysis. For example if the user is interested in the efficiency of the pitstops, or the consistency of the lap times, he can focus the analysis on drivers that are within a certain range, or can decide to compare drivers who are close in some metrics while distant in others.

As I have already highlighted, the stacked barchart provides insight into the strategy chosen by drivers, it highlights common patterns and unique approaches to tyre strategy and pit stop timing. Since the race of each driver can be divided in stints, and each driver has to complete at least two stints, by interacting with this chart, the user can refine the analysis down to a set of stints performed on the same tyre compound, and can decide to compare drivers only based on that subset. The fuel correction checkbox allows the user to normalize the laptimes and compare stints even if they happened at different points in the race.

The scatterplot provides an overview of the similarity of the laps and thorugh clustering can provide insight into drivers who performed similarly both by looking at all laps but also by looking at their average (*drivers' centroids* option). Knowing which drivers performed similarly can bring up more insights into the factors that influenced their performances by referencing the other graphs.

Obviously these insights can be combined to improve the understanding of the results, and these combined insights are different for every race and for every driver that is being analyzed. I will provide a couple of examples of analyses to demonstrate the capabilities of the system.

7.1 Mexico City

This race is an interesting case, because on lap 34 a red flag was deployed, which meant that the race was temporarily stopped. Upon restarting the race all teams had served their mandatory change of tyres, which meant that the only strategy from that point on was choosing which compound to use for the remaining 37 laps. In the aftermath of this race, much of the commentary rightly revolved around Max Verstappen dominating this second part of the race, managing to pull a gap of over 15 seconds in just those 37 laps, and keeping a remarkably high consistency, but there were other interesting performances to note.

By setting the minimum laps slider to a number greater than 66, we can remove from the visualization the drivers who didn't finish the race, which helps cleaning up the visualization a bit. Immediately from the parallel coordinates we notice that one driver, Lando Norris, managed to gain 12 positions in a race.
By looking at the stacked bar chart then, we can see that he went for a different strategy compared to all the other drivers. He started off on a soft compound tyre, and we can assume that the reasoning was to attempt to complete a good amount of overtakes at the start of the race by using a softer faster tyre. By checking the fuel-correction checkbox and clicking on his first stint, we can see that actually his pace wasn't really that strong, even when compared to the stints of some other drivers which were longer and on slower tyres. This can be in part explained by the fact that when a driver is battling another driver for a position, they both tend to lose time. At the

end of this stint, Norris only gained two positions, as we can see from the parallel coordinates axis.

After only 11 laps, Norris pitted for a change of tyres, moving on to the slower but more resilient hard compound. By clicking this stint, we can see in the parallel coordinates plot that his pace improves quite a bit and during this stint he manages to gain 10 positions on track. By examining the linechart we can deduce that many of the positions gained during this stint were due to his direct competitors doing a pitstop. His improved pace then allowed him to take advantage of a safety car deployed on lap 33 and pit again by only losing two positions of the 10 he had gained. A safety car is a safety measure that can be deployed onto the track if there are dangerous conditions. When the Safety Car is on track, all competing cars must line up behind it and follow at a reduced speed, which means that by entering the pits and changing the tyres, a driver loses less time relative to its competitors. Unfortunately for him and a few other drivers, the safety car was transformed in a red flag and the race was suspended, meaning he lost 2 positions for nothing, as under a red flag all drivers can change the tyres to the car and essentially get a free pit stop.

The final stint of his race was by far the strongest, he was overall the fourth fastest car on track, slightly behind the Ferrari of Leclerc, and he managed to gain another 5 positions. In the linechart we can see through the slope of the line that he was the only driver that managed to match, or even stay close to the pace of Verstappen for multiple laps and that he mostly lost time when having to overtake other drivers on the track.

Using the driver' centroids option in the scatter plot, we can see that the system clusters Norris, Sainz and Piastri as having had a relatively similar performances, which is interesting and reasonable as they finished relatively close to each other and many of the metrics fall within a similar range, but then they are positioned quite far from Verstappen. This may be due to the fact that while Norris's race was good, the really impressive part was only in the latter half, which isn't enough to bring his overall performance close to Verstappen or Hamilton, who had a stronger performance in the first half of the race.

7.2 Marina Bay

One of only 3 races not won by Max Verstappen in 2023, the Marina Bay race in Singapore provides an excellent insight on the impact that a race strategy and a fortuitous safety car can have. In this case the topic of discussion will be a strategic choice made a team, and whether or not it was the correct decision.

After selecting the race from the dropdown menu, we can choose which chart to consider first to begin our analysis. From the parallel coordinates we can see that two drivers, Hamilton and Russel, have an average laptime considerably faster, almost 0.3 seconds per lap, than the next closest drivers, and despite that neither of them won the race. Since they drive for the same team, Mercedes, it makes sense that they have a similar performance, and this similarity is also visible in the scatter plot after selecting the drivers' centroids option from the dropdown menu. To understand what prevented them form winning despite being so much faster overall, we can start by looking at the stacked bar chart and notice a couple of things: first, is that almost every driver made a pitstop at lap 20, and second that Hamilton and Russel are two of only seven drivers that opted for a 2-stop strategy. By looking at the line chart we can understand in what context these pitstops have been made.
In fact, we can see that at lap 20 a safety car was deployed, which meant that the drivers could enter the pits for a pit stop and lose less time relative to the other drivers. The only drivers that elected to not pit during this safety car period are the drivers who were using the hard compound, which could not justify pitting so early while using a tyre compound that is supposed to last more than double.
Then we can see that at lap 44 a VSC (virtual safety car) was deployed. A VSC is essentially the same thing as a safety car, but with no physical extra car on track. Drivers are required to drive slower and have a target laptime which they must not be faster than. Therefore the VSC offers similar advantages in terms of pitstop opportunity.
By zooming in the linechart, we can see that when Hamilton and Russel decided to pit under the VSC condition, they were running in second and fourth and were relatively close to the drivers ahead, but they knew that by staying on the same strategy as them they would not have the chance to overtake them, so they decided to take a risk.
Again from the linechart we can see that after the second pitstop they dropped to fourth and fifth on the track but then were substantially faster than every other driver, with the exception of Verstappen who has a similar trend in reducing his delta in the last part of the race.

To make the charts easier to read we can set the minimum laps filter to a number greater than 51 (as that is the lap in which Bottas retired, which is an outlier in the parallel coordinates plot), and using the parallel coordinates plot and the drivers legend we can filter out the drivers we are not interested in.
Now simply by looking at the slope of the lines we can confidently say that, even if the risky choice did not pay off, choosing to pit and give up the track position was the correct call as both cars easily got past Leclerc, and should have overtook Norris and Sainz a couple of laps before the end of the race, and that these two drivers deserve commendation for managing to hold position against cars that were over a second per lap faster.

It is also interesting to note that Verstappen, despite having started from ninth and having had a bad race by his standards, in the last stint of the race he had the same tyre compound and a similar stint length to Hamilton and Russel and was on average only 0.2 seconds per lap slower while also gaining 10 positions, which suggests that with a strategy similar to the Mercedes' cars, he could have also fought for at least a podium.

8 Conclusions

This project provides a comprehensive system to analyze Formula 1 race by allowing its users to produce well-informed and insightful content and can offer insights into driver performances and race dynamics and strategies.

Despite its merits, the system could be improved in numerous ways. Future work could involve adding more relevant filters and integrating additional data, both from FastF1 and other data sources, such as integrating weather data or tyre wear into the analysis. Using more advanced metrics and analytical methods, such as cornering ability or tyre preserving capabilities or correlation analysis might provide even more insights. Furthermore, it might prove useful to explore more visualization techniques and interaction, such as heatmaps of tracks highlighting where events tend to happen or also the drivers that used different racing lines during a race.

References

[1] Charles Perin et al. "State of the Art of Sports Data Visualization". In: *Computer Graphics Forum* 37.(3) (2018), pp. 1–24. DOI: https://onlinelibrary.wiley.com/doi/10.1111/cgf.13447.

[2] Joachim Gudmundsson and Michael Horton. "Spatio-Temporal Analysis of Team Sports". In: *ACM Comput* Surv. 50,2.Article 22 (2017), 34 pages. DOI: https://onlinelibrary.wiley.com/doi/10.1111/cgf.13447.

[3] Sebastian Eckl, Nick Kerschagel, and Uwe Kloos. "A Web-Based Tool for analysing Formula 1 Races and Seasons". In: *The 14th International Symposium on Visual Information Communication and Interaction (VINCI '21)* (2021), 5 pages. DOI: https://dl.acm.org/doi/pdf/10.1145/3481549.3481559.

[4] Pau Matas Albiol. *Racing Driving Data Visualization*. Thesis. https://upcommons.upc.edu/handle/2117/394636. June 2023.

[5] F1Dash. *F1Dash*. URL: https://f1dash.azurewebsites.net/.

[6] F1Tempo. *F1Tempo*. URL: https://www.f1-tempo.com/.