



Project Writeup

Code

Formula 1 - Racing through Data

AUTHOR

Pixels and Patterns

AFFILIATION

College of Information Science, University of Arizona

Introduction

This dataset contains match-level data from the F1 Racing Championship season, covering various fixtures between teams across different dates. For each match, the dataset records essential information, including the home and away teams, the goals scored by each team (both full-time and half-time), and the final result (win, loss, or draw). Additional metrics provide insights into team performances, such as the number of shots on target, fouls committed, and corner kicks. Furthermore, the dataset tracks disciplinary actions with columns for yellow and red cards, offering a comprehensive view of each game's flow and outcome.

The data can be utilized to explore various aspects of soccer performance, team strategies, and match outcomes. By examining trends in team behavior—such as shooting accuracy or foul patterns—analysts can gain insights into factors that influence match results. This makes the dataset valuable for statistical analysis and predictions related to soccer matches.

Question 1: How can we visualize the race winners on the Circuit de Monaco over the past four years (2019-2023) in a clear and insightful way using R?:

Introduction - This project leverages the “f1dataR” library to analyze historical Formula 1 data for the seasons 2021 to 2023. The primary goal is to explore the historical performance of drivers and teams on one of the most iconic circuits in Formula 1, providing a focused analysis of winners and trends at the Circuit de Monaco.

.

```
1 # Uncomment and Run this chunk if you have never installed this package
2 #install.packages("f1dataR")
3 #library(f1dataR)
4 #setup_fastf1()
```

Approach -

The goal of the project was to visually represent the performance of Monaco Grand Prix winners from 2019 to 2023 by mapping their telemetry data and highlighting RPM variations across the circuit.

Data Collection:

- The project utilized the “f1dataR” package to extract telemetry data for each year’s Monaco Grand Prix winner, covering 2019, 2021, 2022, and 2023 (the 2020 Monaco GP was canceled).
- The winners’ details, including driver and constructor names, were manually identified from F1 standings for each year.
- Telemetry data for each driver was extracted, focusing on x, y coordinates (track map) and RPM (engine speed).

1. Data Processing:

- Telemetry data was cleaned by filtering out missing or infinite values to ensure accurate visual representation.
- Team-specific color gradients were assigned to each winner based on their constructor (e.g., Mercedes for Hamilton, Red Bull for Verstappen and Perez).
- A label denoting the team name was added to each plot for easy identification of constructors.

2. Visualization:

- Track maps were created for each driver, with RPM represented as a color gradient along the circuit path.
- Each driver’s telemetry visualization was plotted in a grid using facets, allowing side-by-side comparisons of winners over the years.
- The final visualization included a title and year-specific annotations to provide context and insights into the performance of drivers across the Monaco Grand Prix.

Analysis -

```
1 library(ggplot2)
2 library(dplyr)
3 library(f1dataR)
4 library(gridExtra)
5
6 # Define driver colors and team names
7 driver_colors <- list(
8   hamilton = c("#00A19C", "#C6C6C6"), # Mercedes colors
9   max_verstappen = c("#121F45", "#FFC906"), # Red Bull colors
10  perez = c("#121F45", "#CC1E4A") # Red Bull colors for Perez
11 )
```

```

12
13 # Define the winners and years
14 monaco_winners_df <- data.frame(
15   Year = c(2019, 2021, 2022, 2023),
16   Winner = c("hamilton", "max_verstappen", "perez", "max_verstappen"),
17   Constructor = c("mercedes", "red_bull", "red_bull", "red_bull"),
18   stringsAsFactors = FALSE
19 )
20
21 # Initialize an empty list to store telemetry data for each driver
22 track_data <- list()
23
24 # Load telemetry data for each driver from 2019 to 2023
25 for (i in 1:nrow(monaco_winners_df)) {
26   winner <- monaco_winners_df$Winner[i]
27   year <- monaco_winners_df$Year[i]
28   driver <- ifelse(winner == "hamilton", "HAM",
29                     ifelse(winner == "max_verstappen", "VER", "PER"))
30
31   # Load telemetry for the Monaco Grand Prix
32   telem <- load_driver_telemetry(season = year, driver = driver, round = "Monaco
Grand Prix")
33
34   # Clean telemetry data
35   telem_clean <- telem %>%
36     filter(!is.na(x), !is.na(y), !is.na(rpm)) %>%
37     filter(is.finite(x), is.finite(y), is.finite(rpm))
38
39   # Store the cleaned telemetry data
40   track_data[[paste(driver, year, sep = "_")]] <- telem_clean
41 }
42
43 # Create track map plots for each driver using telemetry data and assigned colors
44 track_plots <- list()
45
46 # Generate track maps for each driver and year
47 for (i in 1:nrow(monaco_winners_df)) {
48   winner <- monaco_winners_df$Winner[i]
49   year <- monaco_winners_df$Year[i]
50   driver <- ifelse(winner == "hamilton", "HAM",
51                     ifelse(winner == "max_verstappen", "VER", "PER"))
52
53   # Get the corresponding color for the driver
54   color_low <- driver_colors[[winner]][1]
55   color_high <- driver_colors[[winner]][2]
56   team_name <- toupper(monaco_winners_df$Constructor[i]) # Get the team name
57
58   # Generate the plot if the telemetry data exists
59   if (paste(driver, year, sep = "_") %in% names(track_data)) {
60     telem_clean <- track_data[[paste(driver, year, sep = "_")]]

```

```

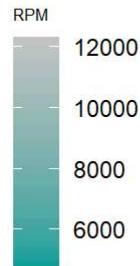
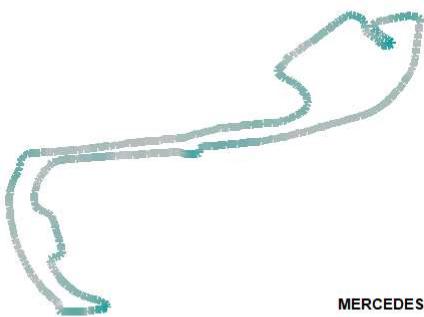
62     track_plots[[paste(driver, year, sep = "_")]] <- ggplot(telem_clean, aes(x = x,
63     y = y)) +
64       geom_path(aes(color = rpm), size = 1.5) + # Color the path based on RPM
65       scale_color_gradient(low = color_low, high = color_high) + # Team colors for
RPM
66       labs(
67         title = paste("Monaco GP Track Map - RPM for", toupper(winner), year),
68         subtitle = paste("Telemetry from", toupper(winner), "in Monaco GP", year),
69         x = NULL,
70         y = NULL
71       ) +
72       theme_minimal() +
73       theme(title = element_text(size = 8),
74             subtitle = element_text(size = 5),
75             axis.text = element_blank(),
76             axis.ticks = element_blank(),
77             panel.grid = element_blank(),
78             legend.title = element_text(size = 6), # Customize legend text
79             legend.position = "right"
80       ) +
81       guides(color = guide_colorbar(title = "RPM")) + # Add RPM color legend
82       annotate("text", x = Inf, y = -Inf, label = team_name,
83                 hjust = 1.2, vjust = -1.5, size = 2, color = "black", fontface =
"bold") # Add team label at bottom right
84   }
85
86 # Create the title grob
87 title <- grid::textGrob("Winners of Monaco GP (2019-2023)", gp =
grid::gpar(fontsize = 20, fontface = "bold"))
88
89 # Display the track maps in a 2x2 grid with a title on top
90 plot_grid <- gridExtra::grid.arrange(
91   title,
92   gridExtra::arrangeGrob(
93     track_plots[["HAM_2019"]], track_plots[["VER_2021"]],
94     track_plots[["PER_2022"]], track_plots[["VER_2023"]],
95     nrow = 2, ncol = 2
96   ),
97   nrow = 2, heights = c(0.1, 1)
98 )

```

Winners of Monaco GP (2019-2023)

Monaco GP Track Map - RPM for HAMILTON 2019

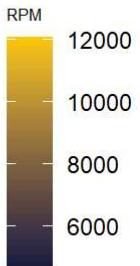
Telemetry from HAMILTON in Monaco GP 2019



MERCEDES

Monaco GP Track Map - RPM for MAX_VERSTAPPEN 2

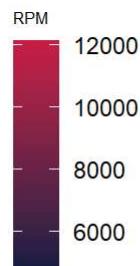
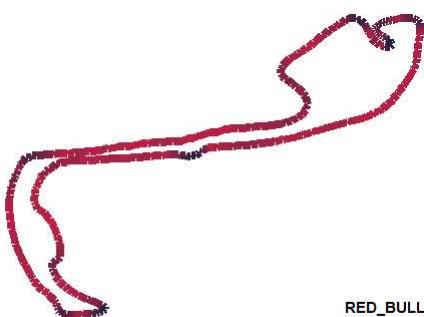
Telemetry from MAX_VERSTAPPEN in Monaco GP 2021



RED_BULL

Monaco GP Track Map - RPM for PEREZ 2022

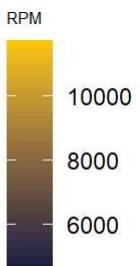
Telemetry from PEREZ in Monaco GP 2022



RED_BULL

Monaco GP Track Map - RPM for MAX_VERSTAPPEN 2

Telemetry from MAX_VERSTAPPEN in Monaco GP 2023



RED_BULL

```
1 #ggsave("monaco_gp_winners_2019_2023.png", plot = plot_grid, width = 12, height = 10, dpi = 300)
```

Question 2. How can a driver's performance on their fastest lap be analyzed using speed and throttle data?

Introduction

This analysis focuses on leveraging telemetry data to evaluate the performance of a Formula 1 driver during their fastest lap. By visualizing speed and throttle data over the course of a lap, we can uncover nuanced insights into the driver's approach to track sections, their use of throttle, and overall efficiency.

Approach

Data Collection

The telemetry data for Max Verstappen (VER) during the 2023 season, Round 1, was retrieved using the f1dataR package. The fastest lap for the driver was identified using the load_laps function, and its corresponding telemetry was extracted with load_driver_telemetry. Data Processing:

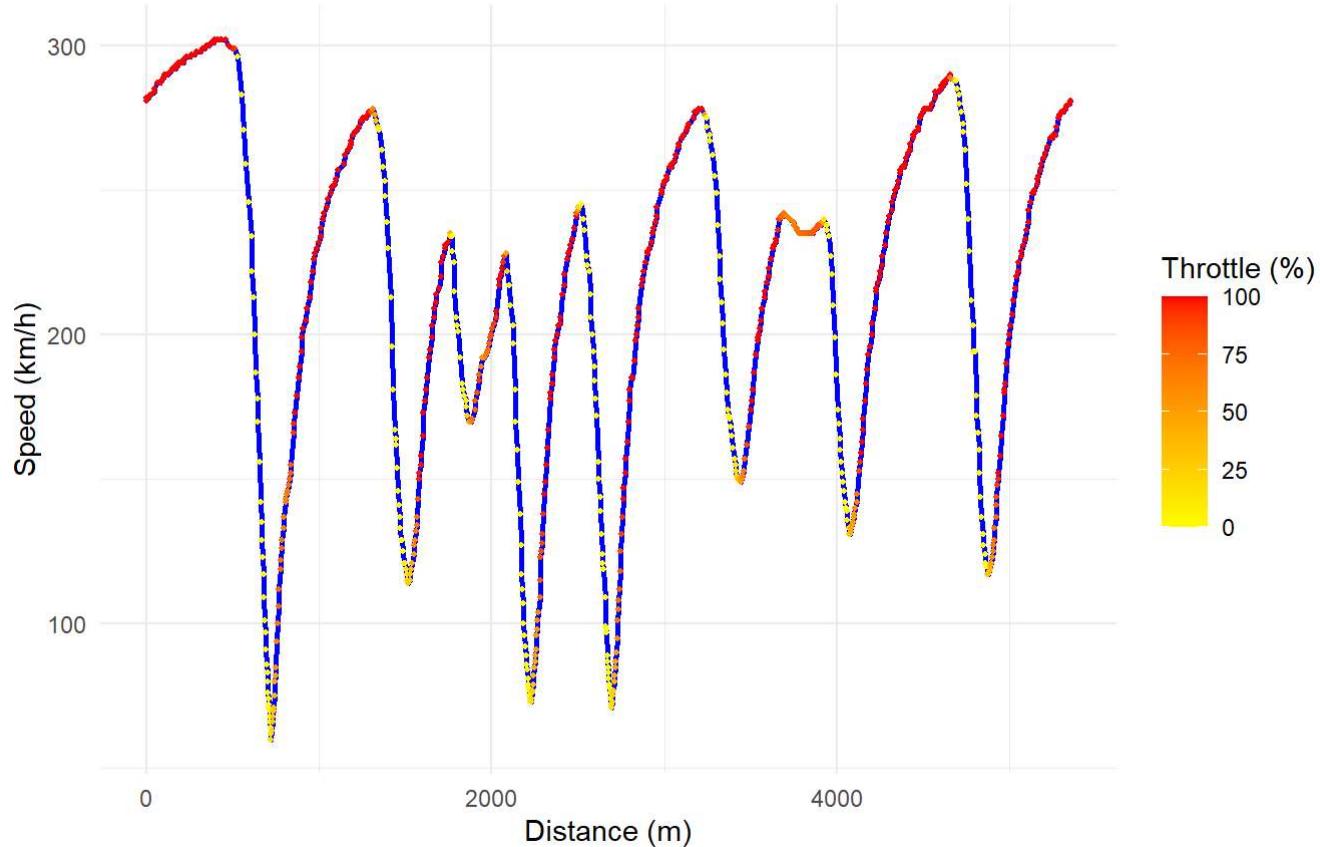
The fastest lap was filtered by finding the lap with the minimum time_sec value. Telemetry data for the specific lap was processed to include speed, throttle, and distance metrics. Visualization:

A line plot was created to visualize speed across the lap distance, with throttle usage represented as a gradient. This allowed us to see the relationship between acceleration, deceleration, and speed consistency across different track sections.

```
1 library(f1dataR)
2 library(ggplot2)
3 library(dplyr)
4
5 year <- 2023
6 round <- 1
7 driver_code <- "VER" # Change as needed
8
9 laps <- load_laps(year, round)
10
11 driver_laps <- laps %>% filter(driver_code == driver_code)
12
13 fastest_lap <- driver_laps %>% filter(time_sec == min(time_sec, na.rm = TRUE))
14
15 fastest_lap_number <- fastest_lap$lap[1]
16
17 telemetry <- load_driver_telemetry(
18   season = year,
19   round = round,
20   session = "R",
21   driver = driver_code,
22   lap = fastest_lap_number
23 )
24
25 # Visualization: Speed vs Distance with Throttle Usage
26 ggplot(telemetry, aes(x = distance, y = speed)) +
27   geom_line(color = "blue", size = 1) +
28   geom_point(aes(color = throttle), size = 0.8) +
29   scale_color_gradient(low = "yellow", high = "red", name = "Throttle (%)") +
30   labs(
31     title = paste("Driver Performance Analysis:", driver_code),
32     subtitle = paste("Fastest Lap - Round", round, year),
33     x = "Distance (m)",
34     y = "Speed (km/h)"
35   ) +
36   theme_minimal() +
37   theme(plot.title = element_text(face = "bold", size = 14))
```

Driver Performance Analysis: VER

Fastest Lap - Round 1 2023



Question 3: How do each F1 team's lap times improve over the course of a race, and how do these improvements compare across different circuit types?

Introduction

This analysis explores how F1 teams' lap times improve during a race and compares their performance across different types of circuits, focusing on the 2022 and 2023 seasons.

Data Collection:

Lap data, race schedules, and circuit information were gathered using the [f1dataR](#) package.

Circuits were categorized as "Technical," "High-Speed," or "Other."

Data Preprocessing:

Lap times were merged with team and circuit data.

Only the top 7 circuits were selected for analysis, and lap times were converted to numeric values.

Data Visualization:

A Scatter plot was created to show average lap times of teams across circuits.

Each circuit is displayed in a separate panel, with teams color-coded for clarity.

Titles, labels, and themes were customized to ensure readability.

Data Analysis:

```
1 # Load necessary libraries
2 library(f1dataR)
3 library(dplyr)
4 library(ggplot2)
5
6 # Define the seasons to include
7 seasons <- c(2022, 2023)
8
9 # Initialize an empty list to store lap data across seasons
10 laps_list <- list()
11
12 # Loop through each season and each race to load lap data
13 for (season in seasons) {
14     # Load the schedule for the current season
15     race_schedule <- load_schedule(season)
16
17     # Loop through each race in the schedule
18     for (i in seq_along(race_schedule$round)) {
19         round_num <- as.integer(race_schedule$round[i])
20
21         # Load lap data for the race
22         laps <- load_laps(season = season, round = round_num)
23
24         # Load results data to get constructor (team) information
25         results <- load_results(season = season, round = round_num)
26
27         # Merge lap data with results data to include the constructor (team)
28         laps_with_team <- laps %>%
29             left_join(results %>% select(driver_id, constructor_id), by = "driver_id")
30             %>%
31             rename(team = constructor_id) %>%
32             mutate(round = race_schedule$round[i], season = as.character(season)) # Add
33             round and season
34
35         # Append to the list
36         laps_list[[length(laps_list) + 1]] <- laps_with_team
37     }
38
39 # Combine all races' lap data into one data frame
40 laps_data <- bind_rows(laps_list)
```

```

40
41 # Load circuits data
42 circuits_data <- load_circuits()
43
44 # Define the top 7 circuits
45 top_7_circuits <- c(
46   "Albert Park Grand Prix Circuit",
47   "Autodromo Nazionale di Monza",
48   "Circuit de Monaco",
49   "Circuit de Spa-Francorchamps",
50   "Red Bull Ring",
51   "Silverstone Circuit",
52   "Suzuka Circuit"
53 )
54
55 # Filter data for the top 7 circuits
56 top_circuits_data <- laps_data %>%
57   inner_join(race_schedule, by = c("season", "round")) %>%
58   filter(circuit_name %in% top_7_circuits) %>%
59   mutate(lapTime = as.numeric(time_sec)) # Convert lap time to numeric
60
61 # Calculate average lap time for each team per circuit
62 heatmap_data <- top_circuits_data %>%
63   group_by(circuit_name, team) %>%
64   summarize(avg_lap_time = mean(lapTime, na.rm = TRUE)) %>%
65   ungroup()
66
67 # Define team colors
68 team_colors <- c(
69   "mercedes" = "#00D2BE",
70   "ferrari" = "#DC0000",
71   "red bull" = "#0600EF",
72   "alpine" = "#0090FF",
73   "haas" = "#F62039",
74   "aston martin" = "#006F62",
75   "alphatauri" = "#2B4562",
76   "mclaren" = "#FF8700",
77   "alfa" = "#900000",
78   "williams" = "#005AFF"
79 )
80
81 # Create the plot
82 team_performance_plot <- ggplot(heatmap_data, aes(x = team, y = avg_lap_time, group
= team, color = team)) +
83   geom_line(size = 1.2, alpha = 0.9) + # Thicker and darker lines
84   geom_point(size = 4, alpha = 1) + # Fully opaque and darker points
85   scale_color_manual(values = team_colors, name = "Team") +
86   facet_wrap(~ circuit_name, scales = "fixed", labeller = label_wrap_gen()) +
87   labs(
88     title = "Team Performance Across Top 7 Famous Circuits",
89     subtitle = "Faceted plots showing lap time trends for each circuit",

```

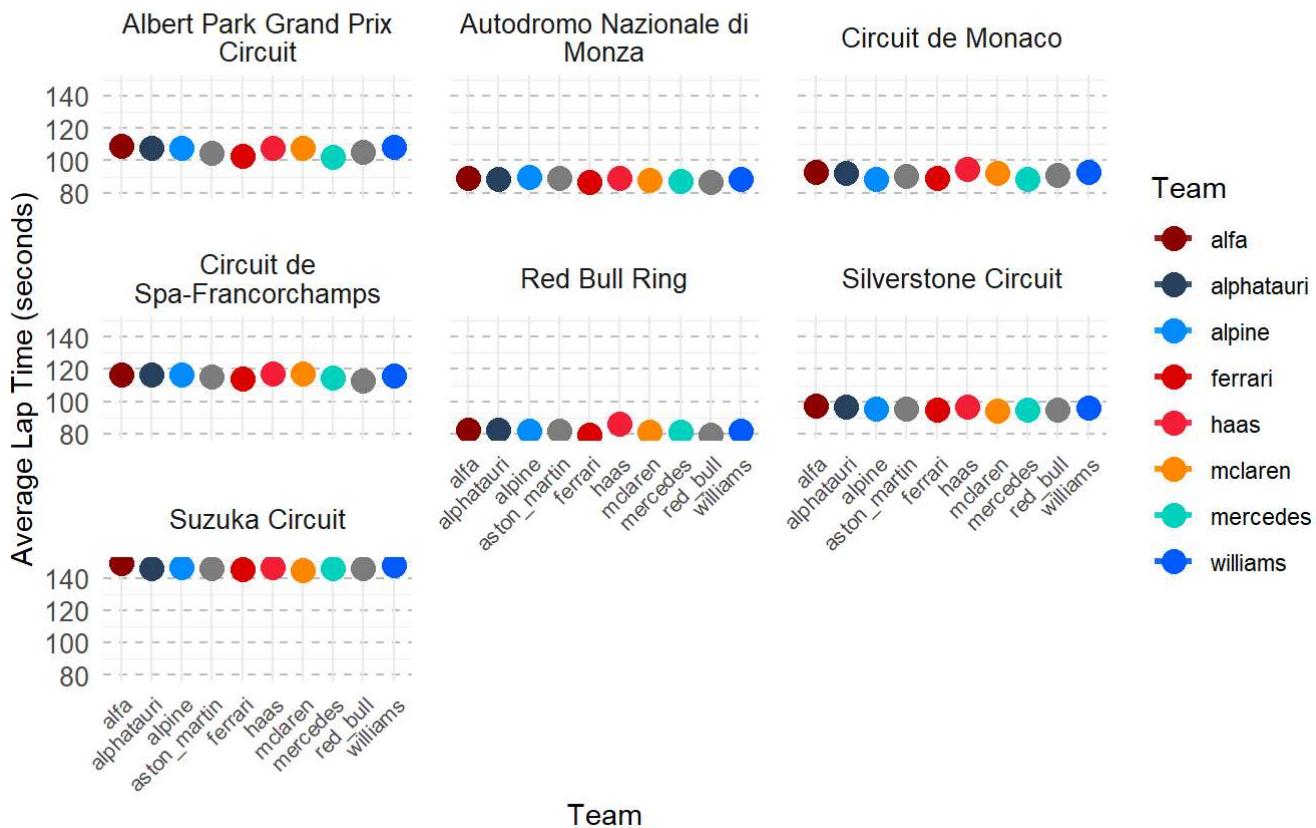
```

90     x = "Team",
91     y = "Average Lap Time (seconds)"
92 ) +
93 theme_minimal() +
94 theme(
95   axis.text.x = element_text(angle = 45, hjust = 1, size = 8),
96   axis.text.y = element_text(size = 10),
97   plot.title = element_text(size = 16, face = "bold"),
98   plot.subtitle = element_text(size = 12, face = "italic"),
99   legend.position = "right", # Place legend on the right
100  panel.spacing = unit(1, "lines"), # Add space between panels
101  strip.text = element_text(size = 10), # Adjust facet label size
102  panel.grid.major.y = element_line(color = "grey", linetype = "dashed") # Enhance y-axis visibility
103 )
104
105 # Display the plot
106 print(team_performance_plot)

```

Team Performance Across Top 7 Famous Circuits

Faceted plots showing lap time trends for each circuit



```

1 # Save the plot as an image file
2 # ggsave("team_performance_plot.png", plot = team_performance_plot, width = 12,
3          height = 8, dpi = 300)

```

Question 4: How does pitstop time affect the race outcome:

Introduction - This project leverages the “f1dataR” library to analyze historical Formula 1 data for the seasons 2021 to 2023. The primary goal is to identify annual champions, examine the circuits they competed on, and evaluate their average pitstop times across races. By visualizing these metrics, the project aims to uncover patterns and insights into the performance of championship-winning drivers and the impact of specific circuits on pitstop durations.

Approach -

Data Collection:

- The project utilized the “f1dataR” library to extract data on driver standings, circuit schedules, and pitstop details for the seasons 2021, 2022, and 2023.
- Standings data were merged with driver details to identify the champion of each season based on final rankings.

1. Data Processing:

- Circuit schedules for each season were extracted, including the names of circuits and corresponding rounds.
- Pitstop data for each champion across all circuits were retrieved. The average pitstop duration was calculated for each circuit where data was available.

2. Visualization:

- A bar chart was created to compare the average pitstop times of champions across different circuits over the three seasons. Distinct colors were assigned to circuits to enhance clarity and visual appeal.
- The visualization incorporated facets for each year, allowing year-wise comparisons while maintaining a focus on circuit-level details.

Analysis -

```
1 library(f1dataR)
2 library(ggplot2)
3
4 years <- 2021:2023
5 champions <- data.frame(Year = integer(), DriverId = character(), Name =
  character())
6
7 for (year in years) {
8   standings <- load_standings(season = year, round = "last", type = "driver")
9   drivers <- load_drivers(season = year)
10  drivers$driverName <- paste(drivers$given_name, drivers$family_name)
11  standings_with_names <- merge(
12    standings,
```

```

13     drivers[, c("driver_id", "driverName")],
14     by = "driver_id",
15     all.x = TRUE
16   )
17   champion <- standings_with_names[standings_with_names$position == 1,
18   c("driver_id", "driverName")]
18   champions <- rbind(champions, data.frame(Year = year, DriverId =
19   champion$driver_id, driverName = champion$driverName))
20 }
21
22
23 for (year in years) {
24   schedule <- load_schedule(season = year)
25   circuits <- rbind(circuits, data.frame(Year = year, Circuit =
26   schedule$circuit_name, Round = schedule$round))
27 }
28
29
30 for (year in years) {
31   champion_id <- champions[champions$Year == year, "DriverId"]
32   year_circuits <- circuits[circuits$Year == year, ]
33
34   for (round in year_circuits$Round) {
35     pitstops <- try(load_pitstops(season = year, round = round), silent = TRUE)
36     if (!inherits(pitstops, "try-error") && !is.null(pitstops)) {
37       champ_pitstops <- pitstops[pitstops$driver_id == champion_id, ]
38       avg_time <- mean(as.numeric(champ_pitstops$duration), na.rm = TRUE)
39       circuit <- year_circuits[year_circuits$Round == round, "Circuit"]
40       champion_pitstops <- rbind(champion_pitstops, data.frame(Year = year, Circuit
41       = circuit, AvgPitstop = avg_time))
42     }
43   }
44
45   circuit_colors <- c(
46     "Bahrain International Circuit" = "#3cb44b",
47     "Autodromo Enzo e Dino Ferrari" = "#ffe119",
48     "Autódromo Internacional do Algarve" = "#4363d8",
49     "Circuit de Barcelona-Catalunya" = "#f58231",
50     "Circuit de Monaco" = "#911eb4",
51     "Baku City Circuit" = "#42d4f4",
52     "Circuit Paul Ricard" = "#f032e6",
53     "Red Bull Ring" = "#e6194B",
54     "Silverstone Circuit" = "#bfef45",
55     "Hungaroring" = "#fabed4",
56     "Circuit de Spa-Francorchamps" = "#469990",
57     "Circuit Park Zandvoort" = "#dcbeff",
58     "Autodromo Nazionale di Monza" = "#9A6324",
59     "Sochi Autodrom" = "#ffffac8",

```

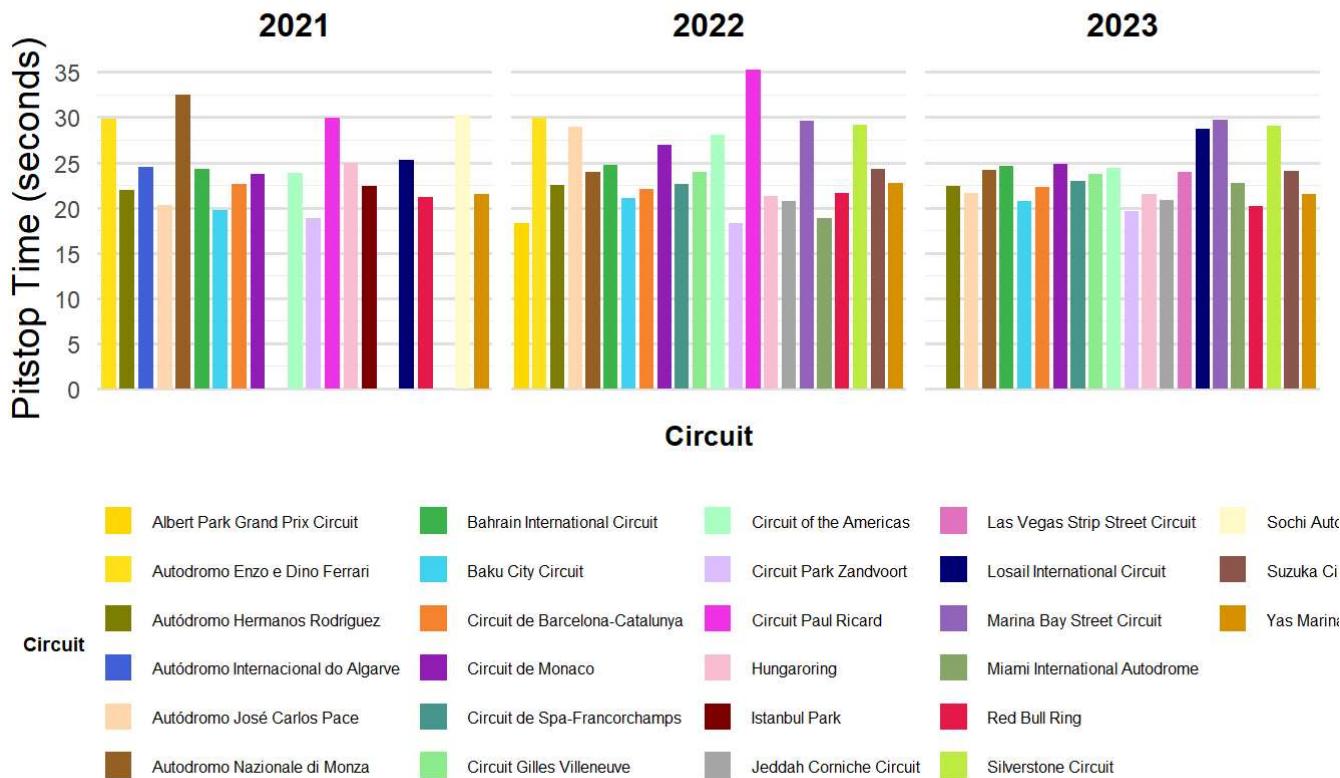
```

60 "Istanbul Park" = "#800000",
61 "Circuit of the Americas" = "#aaffc3",
62 "Autódromo Hermanos Rodríguez" = "#808000",
63 "Autódromo José Carlos Pace" = "#ffd8b1",
64 "Losail International Circuit" = "#000075",
65 "Jeddah Corniche Circuit" = "#a9a9a9",
66 "Yas Marina Circuit" = "#DA9100",
67 "Albert Park Grand Prix Circuit" = "gold",
68 "Miami International Autodrome" = "#87A96B",
69 "Circuit Gilles Villeneuve" = "lightgreen",
70 "Marina Bay Street Circuit" = "#9467bd",
71 "Suzuka Circuit" = "#8c564b",
72 "Las Vegas Strip Street Circuit" = "#e377c2"
73 )
74
75 ggplot(champion_pitstops, aes(x = Circuit, y = AvgPitstop, fill = Circuit)) +
76   geom_bar(stat = "identity", position = "dodge", width = 0.8)
77   +scale_fill_manual(values = circuit_colors) +
78   labs(
79     title = "Pitstop Times of Formula 1 Champions Across Circuits (2021-2023)",
80     subtitle = "Analysis of pitstop times across circuits for the last 3 years",
81     caption = "Data Source: f1dataR",
82     x = "Circuit",
83     y = "Pitstop Time (seconds)"
84   ) +
85   scale_y_continuous(breaks = seq(0, 40, 5))+
86   facet_wrap(~ Year, scales = "free_x") +
87   theme_minimal(base_size = 14) +
88   theme(
89     plot.title = element_text(face = "bold", size = 11, hjust = 0.5),
90     plot.subtitle = element_text(size = 8, hjust = 0.5, color = "gray40"),
91     plot.caption = element_text(size = 6, hjust = 1, face = "italic"),
92     axis.text.y = element_text(hjust = 1, size = 9),
93     axis.text.x = element_blank(),
94     axis.ticks.x = element_blank(),
95     axis.title.x = element_text(face = "bold", size = 10),
96     strip.text = element_text(face = "bold", size = 12, color = "black"),
97     legend.position = "bottom",
98     legend.title = element_text(face = "bold", size = 7),
99     legend.text = element_text(size = 6),
100    panel.grid.major.x = element_blank(),
101    panel.grid.major.y = element_line(color = "gray90"),
102    plot.background = element_rect(fill = "white", color = NA),
103    panel.background = element_rect(fill = "white", color = NA),
104    legend.key.size = unit(4, 'mm')
105  )

```

Pitstop Times of Formula 1 Champions Across Circuits (2021-2023)

Analysis of pitstop times across circuits for the last 3 years



Data Source: f1dataR

Discussion -

Champion Performance:

- The champions for the seasons 2021, 2022, and 2023 were identified, showing consistency in top driver performance across the years.
- The integration of driver standings and detailed circuit data enabled a comprehensive overview of each champion's season.

1. Circuit-Specific Insights:

- Variability in pitstop times across circuits suggests that certain tracks pose unique challenges or opportunities for efficient pitstops.
- Circuits like Monaco and Spa-Francorchamps, known for their technical demands, exhibited distinct average pitstop durations for champions.

2. Implications of Findings:

- The analysis underscores the importance of pitstop strategy as a critical component of championship success. Teams may tailor pitstop strategies to account for circuit-specific characteristics.

- Visualizing data over multiple seasons provides a historical context that can inform future predictions and performance evaluations.

Introduction - The aim is to visualize driver positions over race laps, evaluate the impact of pitstop durations on race outcomes, and identify patterns that influence race strategy. The Abu Dhabi Grand Prix serves as a focal point for this analysis, providing an opportunity to explore how pitstops and race dynamics influence driver rankings.

Approach -

Data Collection:

- Standings and telemetry data for the 2021 season were loaded to contextualize driver performance.
- For the 2023 season, race telemetry, lap data, and pitstop information were extracted, with a specific focus on the Abu Dhabi Grand Prix.
- Driver mappings were utilized to standardize data and enhance interpretability.

1. Data Processing:

- Data for laps and pitstops were merged, allowing for a consolidated view of driver positions and pitstop durations.
- Non-numeric fields, such as pitstop durations, were converted to appropriate numeric types to facilitate analysis.
- Missing data points, such as laps without pitstops, were handled by assigning default values to ensure continuity in visualization.

2. Visualization:

- A line plot was created to visualize driver positions over the course of the race.
- Pitstop durations were represented as point sizes on the plot, highlighting their timing and impact on race positions.
- Custom color mapping was applied to distinguish drivers effectively, while a minimal theme enhanced clarity and focus on the data.

```

1 library(dplyr)
2 chp <- load_standings(season = 2021, round = "last", type = "driver")
3 telem <- load_driver_telemetry(season = 2023, round = "Bahrain", session = "R",
4   driver = "HAM", laps = "all")
5 res <- load_results(season = 2023, round = "last")
6
7 season <- 2023
8 round <- "last"
9 lap_data <- load_session_laps(season = season, round = round, session = "R")

```

```

10
11 pitstop_data <- load_pitstops(season = season, round = round)
12 pitstop_data$duration <- as.numeric(pitstop_data$duration)
13 pitstop_data$lap <- as.numeric(pitstop_data$lap)
14
15 driver_mapping <- load_drivers(season = season)[, c("driver_id", "code")]
16 colnames(driver_mapping)[colnames(driver_mapping) == "code"] <- "driver"
17 pitstop_data <- merge(pitstop_data, driver_mapping, by = "driver_id", all.x = TRUE)
18
19 combined_data <- merge(
20   lap_data,
21   pitstop_data[, c("driver", "lap", "duration")],
22   by.x = c("driver", "lap_number"),
23   by.y = c("driver", "lap"),
24   all.x = TRUE
25 )
26 combined_data$duration[is.na(combined_data$duration)] <- 0
27
28 driver_colors <- c(
29   "ALB" = "#1f77b4", "ALO" = "#ff7f0e", "BOT" = "#2ca02c", "GAS" = "#d62728",
30   "HAM" = "#9467bd", "HUL" = "#8c564b", "LEC" = "#e377c2", "MAG" = "#7f7f7f",
31   "NOR" = "#bcbd22", "OCO" = "#17becf", "PER" = "#ff9896", "PIA" = "#98df8a",
32   "RIC" = "#ffbb78", "RUS" = "#c5b0d5", "SAI" = "#c49c94", "SAR" = "#f7b6d2",
33   "STR" = "#c7c7c7", "TSU" = "#dbdb8d", "VER" = "#9edae5", "ZHO" = "#aec7e8"
34 )
35
36 ggplot(combined_data, aes(x = lap_number, y = position, color = as.factor(driver)))
+ 
37 geom_line(size = 1) +
38 scale_colour_manual(values = driver_colors) +
39 geom_point(data = subset(combined_data, duration > 0), aes(size = duration),
40 shape = 21, fill = "red") +
41 scale_y_reverse(breaks = 1:20) +
42 scale_x_continuous(breaks = seq(0, 60, 5)) +
43 labs(
44   title = "Driver Positions and Pitstop Times in Abu Dhabi Grand Prix",
45   subtitle = "Visualizing race progress and pitstop impacts for all drivers",
46   x = "Lap Number (Multiples of 5)",
47   y = "Race Position (1st = Top)",
48   color = "Driver",
49   size = "Pitstop Duration (seconds)",
50   caption = "Data source: f1dataR"
51 ) +
52 theme_minimal(base_size = 14) +
53 theme(
54   plot.title = element_text(face = "bold", size = 10, hjust = 0.5),
55   plot.subtitle = element_text(size = 8, hjust = 0.5, color = "darkgray"),
56   plot.caption = element_text(size = 6, hjust = 1, face = "italic"),
57   axis.title.x = element_text(face = "bold", size = 8),
58   axis.title.y = element_text(face = "bold", size = 8),
59   axis.text = element_text(size = 7),
60   legend.title = element_text(face = "bold", size = 10),
61   legend.text = element_text(size = 8),
62   legend.key.size = 10
63 )

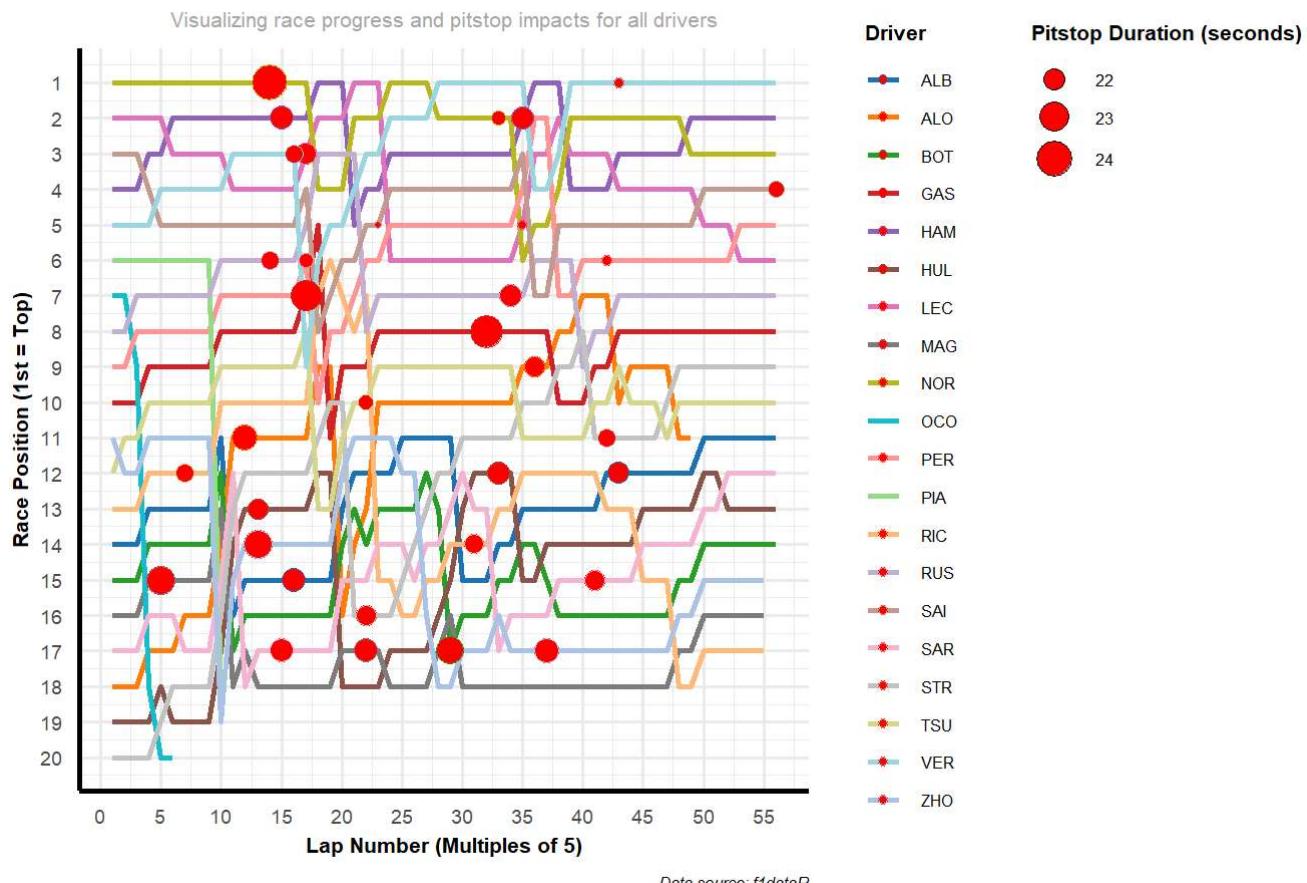
```

```

59     axis.line.x = element_line(color = "black", size = 1),
60     axis.line.y = element_line(color = "black", size = 1),
61     legend.title = element_text(face = "bold", size = 8),
62     legend.text = element_text(size = 6),
63     plot.background = element_rect(fill = "white", color = NA),
64     panel.background = element_rect(fill = "white", color = NA),
65     legend.direction = "vertical",
66     legend.position = "right",
67     legend.box = "horizontal",
68     legend.title.align = 0,
69     legend.key.size = unit(5, 'mm')
70   )

```

Driver Positions and Pitstop Times in Abu Dhabi Grand Prix



Discussion -

Driver Performance Trends:

- The visualization highlights fluctuations in driver positions throughout the race, illustrating how pitstops and race incidents influence rankings.
- Consistent performance by top drivers, such as Lewis Hamilton (HAM), underscores the importance of maintaining competitive pace and minimizing errors.

1. Pitstop Analysis:

- The timing and duration of pitstops emerged as critical factors in shaping race outcomes. Drivers with shorter pitstops generally regained track positions more efficiently.
- Variability in pitstop durations across drivers points to differences in team strategy and execution.

2. Implications for Strategy:

- The combined analysis of lap data and pitstops provides a detailed view of race strategies. Teams can use similar visualizations to refine pitstop timing and assess the impact of their decisions on race outcomes.
- Understanding driver trends over race laps aids in predicting potential overtakes and defending positions effectively.