Marathon Performance Analysis: Impact of Different Weather Conditions on Runners

William Qian

October 2024

Introduction

Marathon running is a popular sport that attracts millions of participants worldwide. The completion time of a marathon is influenced by various factors, including weather conditions. In this report, we analyze the impact of different weather conditions on marathon performance across the lifespan in both men and women using data from five major marathons: Boston, NYC, Chicago, Twin Cities, and Grandmas.

Data Description

Two datasets are used in this analysis: Marathon Data and Course Record Data. Those two datasets were combined to form a comprehensive dataset for analysis.

Marathon Data

The Marathon Data contains information about the average completion record percentage grouped by age and sex of runners in five major marathons: Boston, NYC, Chicago, Twin Cities, and Grandmas. The dataset includes the following columns:

Parameter	Coding details
Race	0 = Boston Marathon , 1 = Chicago Marathon , 2 = New York City Marathon , 3 = Twin Cities Marathon (Minneapolis, MN) , 4 = Grandma's Marathon (Duluth, MN)
Year	Year of the marathon
Sex/Gender	0 = Identified as Female , 1 = Identified as Male
Flag	White = WBGT <10°C , Green = WBGT 10-18°C , Yellow = WBGT 18-23°C , Red = WBGT 23-28°C , Black = WBGT >28°C
% CR	Percent off current course record for gender
Td, °C	Dry bulb temperature in Celsius
Tw, °C	Wet bulb temperature in Celsius
%rh	Percent relative humidity
Tg, °C	Black globe temperature in Celsius
SR W/m²	Solar radiation in Watts per meter squared
DP	Dew Point in Celsius
Wind	Wind speed in Km/hr
WBGT	Wet Bulb Globe Temperature

Note that the variable `WBGT` is actually a composite index that combines the effects of `Td`, `Tw`, and `Tg`, which means we can ignore the three variables when analyzing the data.

$$WBGT = 0.7 \times Tw + 0.2 \times Tg + 0.1 \times Td$$

This data contains of 11073 observations and 12 variables. The data documented the 5 major races form year 1993 to 2016, and each observation represents the average performance of a specific age in a particular race during a specific year. It should be noted that in this dataset, only one set of weather data is recorded for each race.

Course Record Data

This dataset contains the course record for each race grouped by sex documented in the Marathon Data. The dataset includes the following columns:

Parameter	Coding details
Race	B = Boston Marathon , C = Chicago Marathon , NY = New York City Marathon , TC = Twin Cities Marathon (Minneapolis, MN) , D = Grandma's Marathon (Duluth, MN)
Sex/Gender	0 = Identified as Female , 1 = Identified as Male
Year	Year of the marathon
CR	Current course record for gender

Different from the Marathon Data, the `CR` in this dataset is in hours, which means we can calculate the completion time of each runner by multiplying the CR percentage in the Marathon Data with the CR in this dataset.

Data Preprocessing

Data Manipulation

For clarity, we firstly rename the `Race` column in the Marathon Data as well as the Course Record Data to the full name of the marathon. And since we are using two datasets, we need to merge them based on the `Race`, `Year`, and `Sex` columns. After merging, we now have a united dataset that contains 15 variables and 11073 observations in total.

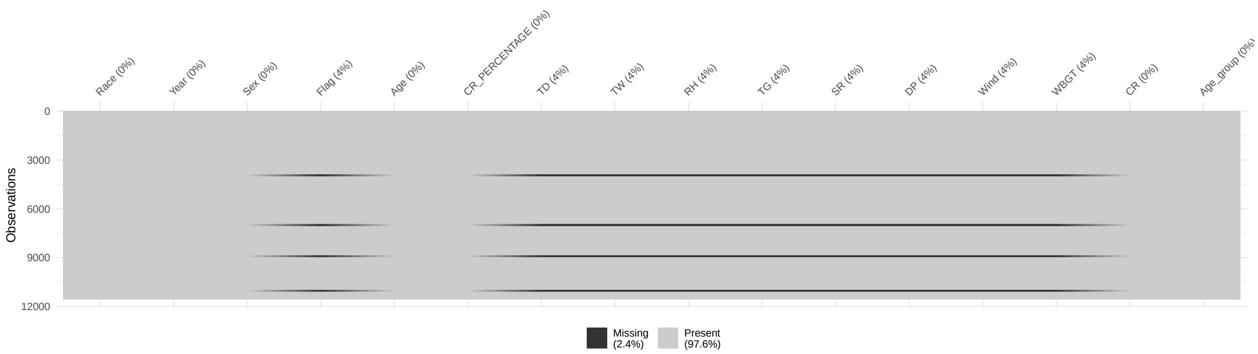
And then, condiering of the nature of the variables and the analysis we are going to perform, we convert the `Year`, `Race`, `Sex`, and `Flag` columns to factors. We also created a new column `Age_group` by grouping the `Age` column into 10-year intervals. This step is essential trying to factorize the `Age` column.

Most importantly, we would like to convert the `CR` in the Course Record Data to seconds. The process is simple, we first convert the `CR` to a period object and then convert it to seconds. After that, we can calculate the completion time of each runner by multiplying the CR percentage in the Marathon Data with the CR in this dataset. The calculation formula is as follows:

$$CR_{adjusted} = (1 + CR_{PERCENTAGE} \times 0.01) \times CR$$

Data Quality Check

Before we start the analysis, we need to check the data quality. We first check for missing values and patterns in the dataset. The missing values are visualized using the `vis_miss` function from the `naniar` package. The plot shows that there are missing values in the weather related columns, and this missingness seems to have a very clear pattern. Based on this pattern, we can assume boldly that the missing values are not missing at random, but are related to specific races.

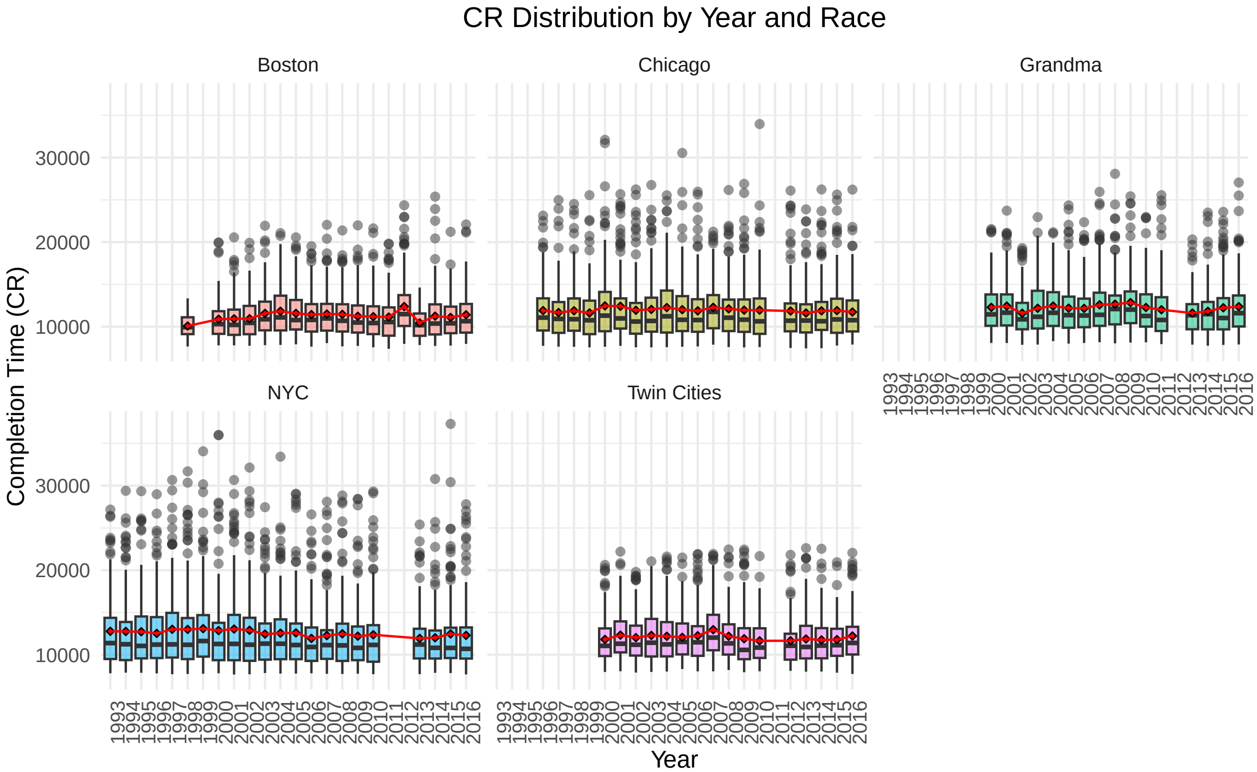


In order to veryify our assumption, we calculate the missing percentage of weather data in each marathon by year. The table below shows that the missing percentage of most of the races are 0, however, for races held in 2011 (Chicago, NYC, and Twin Cities) as well as the one held in 2012 (Grandmas), the missing percentage is 100%. This confirms our assumption that the missing values are related to specific races. It would be a wise choice to remove all of those races from our dataset. To be noted that the NA value in the table means there are no races held in that year.

Table 3: Missing Percentage of Weather Data in Each Marathon by Year

Year	Boston	Chicago	Grandma	NYC	Twin Cities
1993	0	0	NA	0	0
1994	0	0	NA	0	0
1995	0	0	NA	0	0
1996	0	0	NA	0	0
1997	0	0	NA	0	0
1998	0	0	NA	0	0
1999	0	0	NA	0	0
2000	0	0	0	0	0
2001	0	0	0	0	0
2002	0	0	0	0	0
2003	0	0	0	0	0
2004	0	0	0	0	0
2005	0	0	0	0	0
2006	0	0	0	0	0
2007	0	0	0	0	0
2008	0	0	0	0	0
2009	0	0	0	0	0
2010	0	0	0	0	0
2011	0	1	0	1	1
2012	0	0	1	0	0
2013	0	0	0	0	0
2014	0	0	0	0	0
2015	0	0	0	0	0
2016	0	0	0	0	0

And we also want to check the performance distribution of each race by year, since although the weather condition may vary from year to year, the performance of runners may not change significantly within the same track. As we can see in the plot below, despite some disturbances, the performance of runners tend to stay stable over the years. The difference in performance might be due to the different weather conditions in different years. Overall, the stability of the performance indicates that the dataset is reliable for analysis.



Another factor might affect the result of the analysis is `age`. In a balanced dataset, we would expect the number of participants in each age group to be roughly the same. To check this, we count the number of participants in each age group for each races. The table shows a pattern that for each race, participants between age 20 to 79 are the most common and also balanced, while the number of participants in the 10-19, 80-89 and 90-99 age groups are relatively small. This suggests that the inference results obtained from the 10-19, 80-89 and 90-99 age groups may not be as reliable as those obtained from the 20-79 age groups. Moreover, the number of participants in the 90-99 age group is extremely small, so we decide to merge it with the 80-89 age group to become the 80-99 age group.

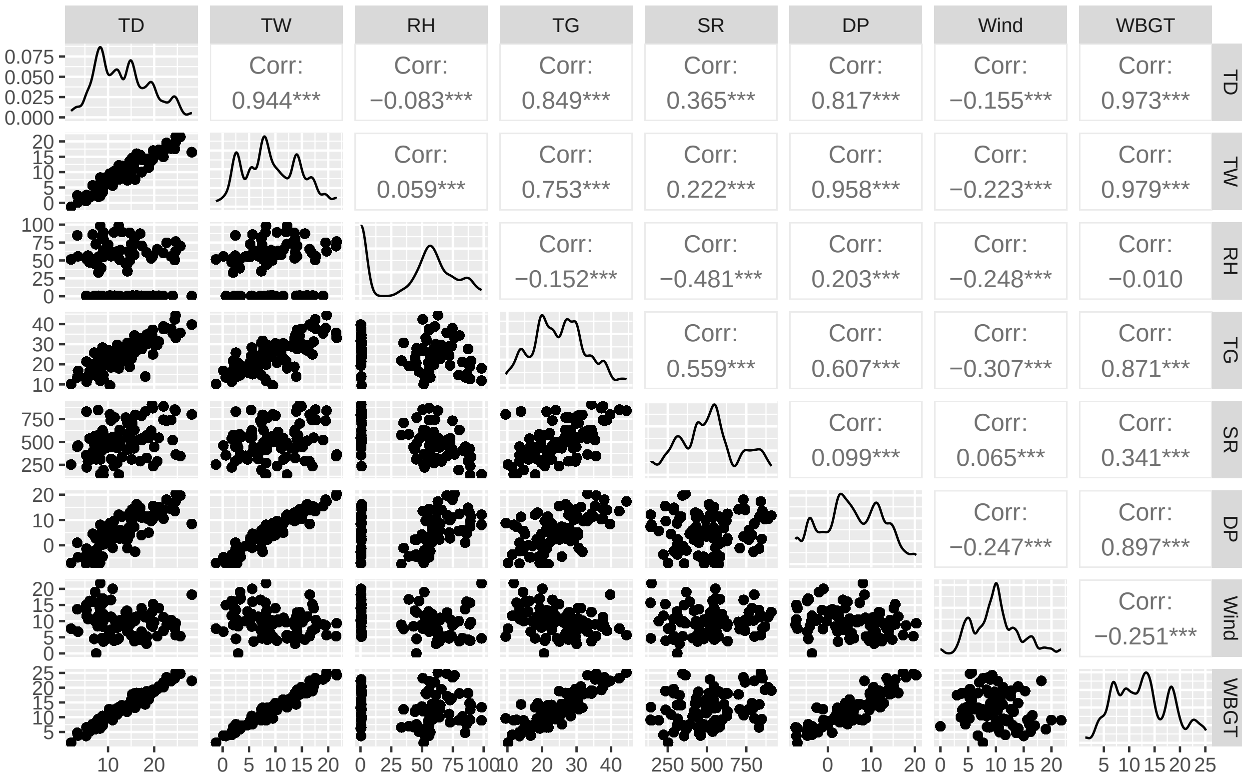
Similarly, we would hope that participants' gender is balanced in the dataset. From the table below, we can tell the Female:Male portion is roughly equal to 1 in all of the races, which is a good feature for our analysis.

Table 4: **Number of Participants by Age Group, Sex and Race**

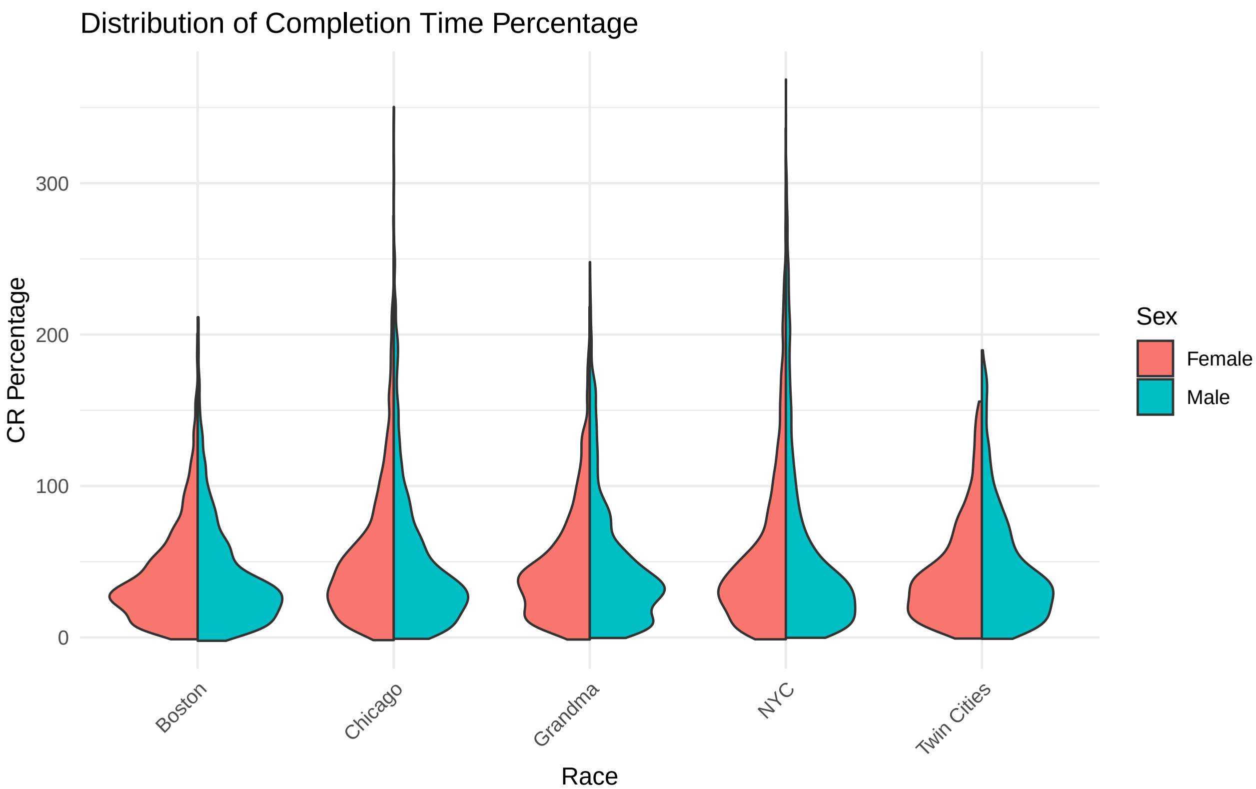
Age Group	**Boston**, N = 2,088	**Chicago**, N = 2,427	**Grandma**, N = 1,884	**NYC**, N = 2,799	**Twin Cities**, N = 1,875
Age_group					
0-9	0 (0%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)
10-19	67 (3.2%)	171 (7.0%)	167 (8.9%)	88 (3.1%)	140 (7.5%)
20-29	360 (17%)	400 (16%)	320 (17%)	440 (16%)	319 (17%)
30-39	360 (17%)	400 (16%)	320 (17%)	440 (16%)	320 (17%)
40-49	360 (17%)	400 (16%)	320 (17%)	440 (16%)	320 (17%)
50-59	359 (17%)	400 (16%)	320 (17%)	440 (16%)	319 (17%)
60-69	337 (16%)	391 (16%)	286 (15%)	438 (16%)	294 (16%)
70-79	215 (10%)	237 (9.8%)	134 (7.1%)	378 (14%)	146 (7.8%)
80-89	30 (1.4%)	28 (1.2%)	17 (0.9%)	130 (4.6%)	17 (0.9%)
90-99	0 (0%)	0 (0%)	0 (0%)	5 (0.2%)	0 (0%)
Sex					
Female	984 (47%)	1,150 (47%)	880 (47%)	1,337 (48%)	867 (46%)

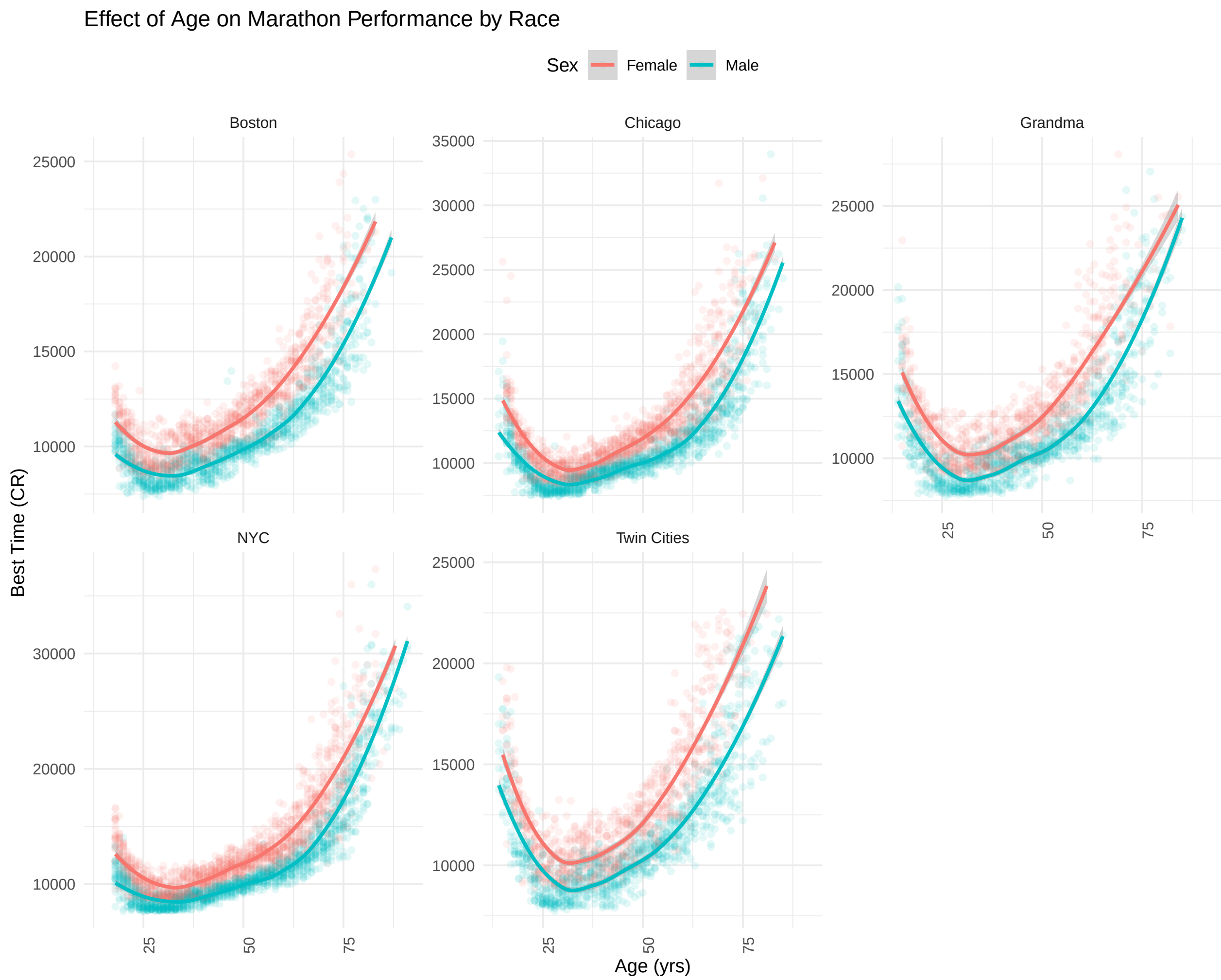
Age Group	**Boston**, N = 2,088	**Chicago**, N = 2,427	**Grandma**, N = 1,884	**NYC**, N = 2,799	**Twin Cities**, N = 1,875
Male	1,104 (53%)	1,277 (53%)	1,004 (53%)	1,462 (52%)	1,008 (54%)

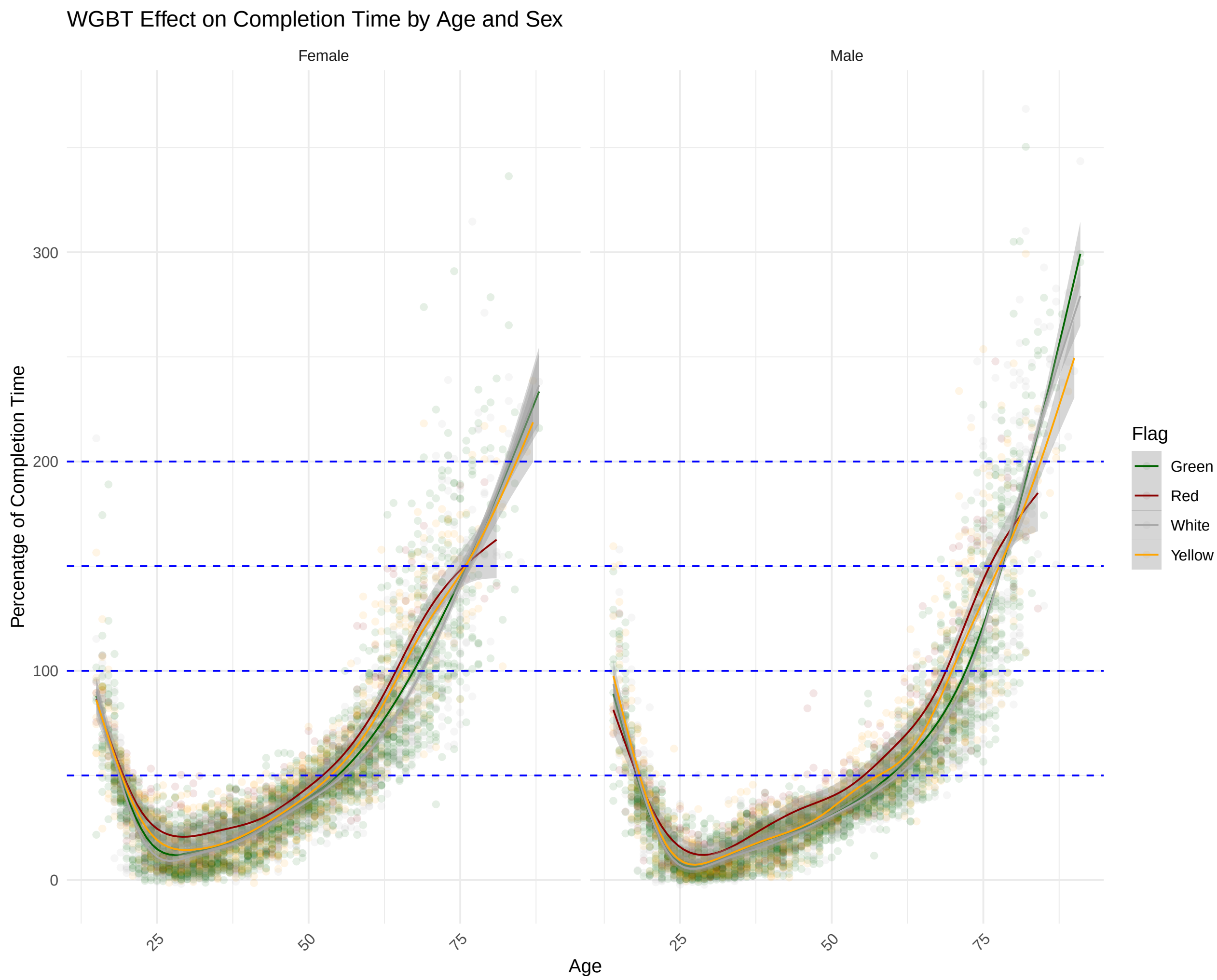
We also want to check the correlation between the weather variables. The correlation plot below shows that `DP` is highly correlated with `WBGT`, which means we can ignore the `DP` variable when analyzing the data. And since the `WBGT` is a composite index that combines the effects of `Td`, `Tw`, and `Tg`, all of those variables are highly correlated with `WBGT`, meaning that they can be ignored as well.



Data Analysis




Effect of Age on Marathon Performance by Race
Sex
Female
Male
Boston
Chicago
Grandma
NYC
Twin Cities
Best Time (CR)
Age (yrs)
25
50
75
10000
15000
20000
25000
30000
35000



WGBT Effect on Completion Time by Age and Sex
Female
Male
Percenatge of Completion Time
300
200
100
0
25
50
75
Age
Flag
Green
Red
White
Yellow

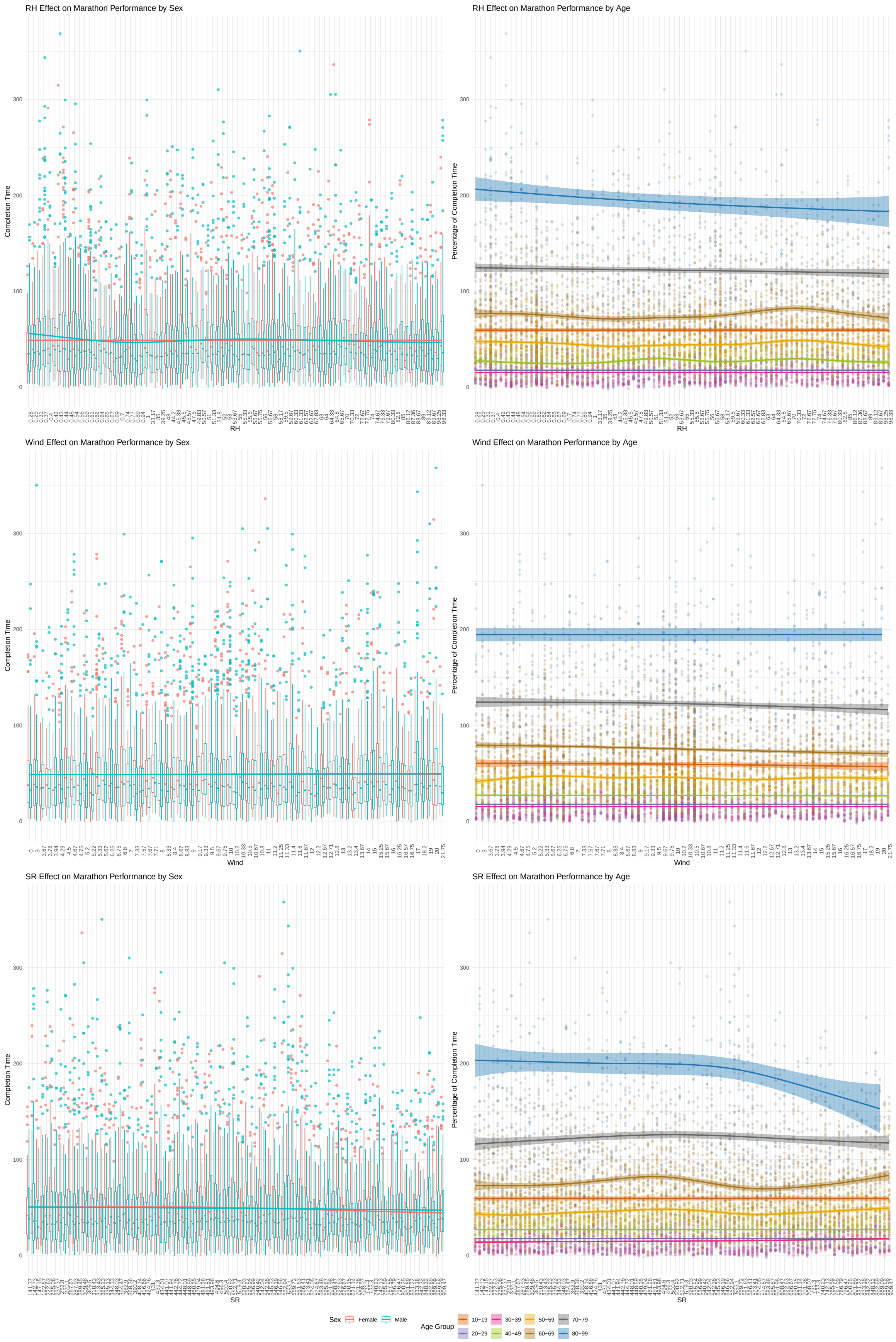

RH Effect on Marathon Performance by Sex
RH Effect on Marathon Performance by Age
Wind Effect on Marathon Performance by Sex
Wind Effect on Marathon Performance by Age
SR Effect on Marathon Performance by Sex
SR Effect on Marathon Performance by Age
Completion Time
Percentage of Completion Time
RH
Wind
SR
Sex
Female
Male
Age Group
10–19
20–29
30–39
40–49
50–59
60–69
70–79
80–99

Table 5: RH Model Coefficients

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-30.2005520	0.9708892	-31.1060755	0.0000000
RH	-0.0003344	0.0094546	-0.0353719	0.9717838
SexMale	-4.7812957	0.6080433	-7.8634129	0.0000000
Age	1.7545477	0.0169002	103.8183237	0.0000000

Table 6: SR Model Coefficients

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-29.4795524	1.2291307	-23.9840664	0.000000
SR	-0.0013813	0.0016146	-0.8555289	0.392277
SexMale	-4.7762996	0.6080477	-7.8551397	0.000000
Age	1.7539441	0.0169126	103.7061018	0.000000

Table 7: DP Model Coefficients

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-32.017656	0.9181339	-34.872534	0
DP	0.287388	0.0435305	6.601993	0
SexMale	-4.801950	0.6068550	-7.912845	0
Age	1.759805	0.0168836	104.231898	0

Table 8: Wind Model Coefficients

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-28.5448183	1.1331262	-25.191208	0.0000000
Wind	-0.1734134	0.0743508	-2.332367	0.0196993
SexMale	-4.7828490	0.6078915	-7.867932	0.0000000
Age	1.7557419	0.0169015	103.880900	0.0000000

Table 9: Flag Model Coefficients

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-30.666331	0.9422152	-32.547057	0.0000000
FlagRed	7.648126	1.3839258	5.526399	0.0000000
FlagWhite	-2.412088	0.6945525	-3.472866	0.0005169
FlagYellow	3.342042	0.8438010	3.960699	0.0000752
SexMale	-4.803419	0.6059713	-7.926810	0.0000000
Age	1.760180	0.0168528	104.444531	0.0000000

Table 10: WBGT Model Coefficients

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-36.2759369	1.1404774	-31.807677	0
WBGT	0.4479189	0.0540060	8.293866	0
SexMale	-4.8069510	0.6061679	-7.930066	0
Age	1.7607921	0.0168625	104.420531	0

Table 11: Model with WBGT Coefficients

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-35.8052408	2.1503858	-16.6506129	0.0000000
RH	-0.0100714	0.0117823	-0.8547937	0.3926839
SR	-0.0117774	0.0021134	-5.5728350	0.0000000
DP	-0.4666199	0.1192422	-3.9132102	0.0000916
Wind	0.0271575	0.0796785	0.3408382	0.7332318
WBGT	1.1062063	0.1550168	7.1360393	0.0000000
SexMale	-4.7708114	0.6053342	-7.8812856	0.0000000
Age	1.7557319	0.0168654	104.1026354	0.0000000

Table 12: Model with Flag Coefficients

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-24.6502924	1.8792540	-13.1170628	0.0000000
RH	-0.0188363	0.0123629	-1.5236100	0.1276347
SR	-0.0089529	0.0019871	-4.5053890	0.0000067
DP	-0.1388574	0.0876138	-1.5848799	0.1130222
Wind	0.0582882	0.0842701	0.6916829	0.4891510
FlagRed	10.7582209	1.7404614	6.1812466	0.0000000
FlagWhite	-4.2883521	1.0310038	-4.1593950	0.0000322
FlagYellow	4.5640988	1.0933738	4.1743263	0.0000301
SexMale	-4.7755005	0.6055284	-7.8865016	0.0000000
Age	1.7556996	0.0168721	104.0591925	0.0000000

Code Appendix

```
knitr::opts_chunk$set(echo = FALSE)
knitr::opts_chunk$set(message = FALSE)
knitr::opts_chunk$set(warning = FALSE)
library(mice, warn.conflicts = FALSE)
library(naniar)
library(ggplot2)
library(dplyr)
library(readr)
library(tidyr)
library(readxl)
library(ggpubr)
library(gtsummary)
library(GGally)
```

```r
library(ggcorrplot)
library(knitr)
library(kableExtra)
library(lubridate)
library(patchwork)
library(introdataviz)

# Load data
marathon_data <- read.csv("../Data/project1.csv")
course_record <- read.csv("../Data/course_record.csv")

# rename the column names that are too long to follow.
colnames(marathon_data)[1] <- "Race"
colnames(marathon_data)[3] <- "Sex"
colnames(marathon_data)[5] <- "Age"
colnames(marathon_data)[6] <- "CR_PERCENTAGE"
colnames(marathon_data)[7] <- "TD"
colnames(marathon_data)[8] <- "TW"
colnames(marathon_data)[9] <- "RH"
colnames(marathon_data)[10] <- "TG"
colnames(marathon_data)[11] <- "SR"

# data type conversion
marathon_data$Year <- as.factor(marathon_data$Year)
marathon_data$Race <- as.factor(marathon_data$Race)
marathon_data$Sex <- as.factor(marathon_data$Sex)
marathon_data$Flag <- as.factor(marathon_data$Flag)

marathon_data$Flag[marathon_data$Flag == ""] <- NA

# replace marathon name with code name in course_record
course_record$Race[course_record$Race == "B"] <- 0
course_record$Race[course_record$Race == "C"] <- 1
course_record$Race[course_record$Race == "NY"] <- 2
course_record$Race[course_record$Race == "TC"] <- 3
course_record$Race[course_record$Race == "D"] <- 4
course_record$Race <- as.factor(course_record$Race)

# replace gender in course_record
course_record$Gender[course_record$Gender == "M"] <- 1
course_record$Gender[course_record$Gender == "F"] <- 0
course_record$Gender <- as.factor(course_record$Gender)
colnames(course_record)[4] <- "Sex"

# Transform records in course_record into seconds
course_record$CR <- period_to_seconds(hms(course_record$CR))

# Join course_record and marathon_data
marathon_data <- merge(marathon_data, course_record, by = c("Race", "Year", "Sex"))

# calculate the record of each runner
marathon_data$CR <- (1 + marathon_data$CR_PERCENTAGE * 0.01) * marathon_data$CR
```

```r
marathon_data <- marathon_data %>%
  mutate(Race = case_when(
    Race == 0 ~ "Boston",
    Race == 1 ~ "Chicago",
    Race == 2 ~ "NYC",
    Race == 3 ~ "Twin Cities",
    Race == 4 ~ "Grandma"
  ),
  Sex = case_when(
    Sex == 1 ~ "Male",
    Sex == 0 ~ "Female"
  )) %>%
  mutate(Age_group = cut(Age, breaks = seq(0, 100, by = 10), right = FALSE,
                         labels = c("0-9", "10-19", "20-29", "30-39", "40-49",
                                    "50-59", "60-69", "70-79", "80-89", "90-99")))

# Check for missing values and patterns
miss_plot <- vis_miss(marathon_data)
ggsave("../Plots/missing_values_plot.png", plot = miss_plot, width = 15, dpi = 300)
miss_plot

# Check the missing percentage of weather data in each marathon by year
marathon_data %>%
  group_by(Race, Year) %>%
  summarise(missing_percentage = sum(is.na(Flag)) / n()) %>%
  pivot_wider(names_from="Race", values_from = missing_percentage) %>%
  arrange(Year) %>%
  replace_na(list(Boston = 0, Chicago = 0, NYC = 0, `Twin Cities` = 0, Grandmas = 0)) %>%
  kable(caption = "Missing Percentage of Weather Data in Each Marathon by Year")

# remove missing data
marathon_data <- marathon_data %>% filter(!is.na(Flag))

completion_time_race <- ggplot(marathon_data, aes(x = Year, y = CR)) +
  geom_boxplot(aes(fill = Race), alpha = 0.5) +
  stat_summary(fun = "mean", geom = "point", shape = 23, size = 1, fill = "red") +
  stat_summary(fun = "mean", geom = "line", aes(group = 1), color = "red") +
  facet_wrap(~ Race) +
  labs(title = "CR Distribution by Year and Race",
       x = "Year",
       y = "Completion Time (CR)") +
  theme_minimal() +
  theme(
    plot.title = element_text(hjust = 0.5),
    axis.text.x = element_text(angle = 90, hjust = 1),
    legend.position = "none"
  )
ggsave("../Plots/completion_time_race.png", plot = completion_time_race, width = 10, height = 8, dpi = 3
completion_time_race

tbl_summary(
  marathon_data %>% select(Race, Age_group, Sex),
  by = Race,
```

```r
    statistic = list(
      Age_group ~ "{n} ({p}%)",
      Sex ~ "{n} ({p}%)"
    )
  ) %>%
  modify_header(label = "**Age Group**") %>%
  modify_caption("**Number of Participants by Age Group, Sex and Race**")
participants_age_plot <- ggplot(marathon_data, aes(x = Age_group, fill = Age_group)) +
  geom_bar() +
  facet_wrap(~ Race) +
  scale_fill_viridis_d() +
  labs(title = "Number of Participants by Age Group for Each Race",
       x = "Age Group",
       y = "Number of Participants",
       fill = "Age Group") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))

ggsave("../Plots/participants_age_plot.png", plot = participants_age_plot, width = 10, height = 8, dpi =
participants_age_plot

marathon_data <- marathon_data %>%
  mutate(Age_group = if_else(Age_group == "90-99", "80-99", Age_group)) %>%
  mutate(Age_group = if_else(Age_group == "80-89", "80-99", Age_group))
sex_distribution_race <- ggplot(marathon_data, aes(x = Sex, fill = Sex)) +
  geom_bar(position = "dodge", alpha = 0.7) +
  facet_wrap(~ Race, scales = "free_y") +
  labs(title = "Sex Distribution by Race",
       x = "Sex",
       y = "Count",
       fill = "Sex") +
  theme_minimal() +
  theme(
    axis.text.x = element_text(face = "italic"),
    legend.position = "none"
  )
ggsave("../Plots/sex_distribution_race.png", plot = sex_distribution_race, width = 10, height = 8, dpi =
sex_distribution_race

cor_plot <- ggpairs(marathon_data %>% select(TD, TW, RH, TG, SR, DP, Wind, WBGT))
ggsave("../Plots/cor_plot.png", plot = cor_plot, width = 10, height = 8, dpi = 300)
cor_plot

cr_distribution_plot <- ggplot(marathon_data, aes(x=Race, y = CR_PERCENTAGE, fill=Sex)) +
  geom_split_violin() +
  labs(title = "Distribution of Completion Time Percentage",
       y = "CR Percentage") +
  theme_minimal() +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1)
  )
ggsave("../Plots/cr_distribution_plot.png", plot = cr_distribution_plot, width = 8, height = 5, dpi = 30
cr_distribution_plot
```

```r
age_effect_plot <- ggplot(marathon_data, aes(x = Age, y = CR, color = Sex)) +
  geom_point(alpha = 0.1) +
  geom_smooth(method = "loess", se = TRUE) +
  facet_wrap(~ Race, scales = "free_y") +
  labs(title = "Effect of Age on Marathon Performance by Race",
       x = "Age (yrs)",
       y = "Best Time (CR)",
       color = "Sex") +
  theme_minimal() +
  theme(
    axis.text.x = element_text(angle = 90, hjust = 1),
    legend.position = "top"
  )
ggsave("../Plots/age_effect_plot.png", plot = age_effect_plot, width = 10, height = 8, dpi = 300)
age_effect_plot

# WGBT effects
flag_colors <- c("Green" = "darkgreen",
                 "Yellow" = "orange",
                 "Red" = "darkred",
                 "White" = "darkgrey")


wgbt_effect <- marathon_data %>%
  ggplot(aes(x = Age, y = CR_PERCENTAGE, color = Flag)) +
  facet_wrap(~ Sex, scales = "fixed") +
  geom_point(alpha = 0.1) +
  geom_smooth(aes(group = Flag, color = Flag), se = T, size=0.5) +
  geom_hline(yintercept = 50, linetype = "dashed", color = "blue") +
  geom_hline(yintercept = 100, linetype = "dashed", color = "blue") +
  geom_hline(yintercept = 150, linetype = "dashed", color = "blue") +
  geom_hline(yintercept = 200, linetype = "dashed", color = "blue") +
  scale_color_manual(values = flag_colors) +
  labs(title = "WGBT Effect on Completion Time by Age and Sex",
       x = "Age",
       y = "Percenatge of Completion Time",
       color = "Flag") +
  theme_minimal() +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1)
  )
ggsave("../Plots/wgbt_effect.png", plot = wgbt_effect, width = 10, height = 8, dpi = 300)
wgbt_effect

age_group_colors <- c("0-9" = "#1b9e77",
                      "10-19" = "#d95f02",
                      "20-29" = "#7570b3",
                      "30-39" = "#e7298a",
                      "40-49" = "#98c61e",
                      "50-59" = "#e6ab02",
                      "60-69" = "#a6761d",
                      "70-79" = "#666666",
                      "80-99" = "#1f78b4")
```

```r
# RH Effects
RH_sex <- marathon_data %>%
      ggplot(aes(x = as.factor(round(RH, 2)), y = CR_PERCENTAGE, color = Sex)) +
      geom_boxplot(alpha = 0.7) +
      geom_smooth(aes(group = Sex), se = FALSE) +
      labs(title = "RH Effect on Marathon Performance by Sex",
           x = "RH",
           y = "Completion Time") +
      theme_minimal() +
      theme(
        axis.text.x = element_text(angle = 90)
      )

RH_age <- marathon_data %>%
      ggplot(aes(x = as.factor(round(RH, 2)), y = CR_PERCENTAGE, fill = Age_group, color = Age_group))
      geom_point(alpha = 0.2) +
      geom_smooth(aes(group = Age_group, color = Age_group), se = T) +
      scale_fill_manual(values = age_group_colors) +
      scale_color_manual(values = age_group_colors) +
      labs(title = "RH Effect on Marathon Performance by Age",
           x = "RH",
           y = "Percentage of Completion Time",
           fill = "Age Group",
           color = "Age Group") +
      theme_minimal() +
      theme(
        axis.text.x = element_text(angle = 90)
      )

# Wind Effects
Wind_sex <- marathon_data %>%
      ggplot(aes(x = as.factor(round(Wind, 2)), y = CR_PERCENTAGE, color = Sex)) +
      geom_boxplot(alpha = 0.7) +
      geom_smooth(aes(group = Sex), se = FALSE) +
      labs(title = "Wind Effect on Marathon Performance by Sex",
           x = "Wind",
           y = "Completion Time") +
      theme_minimal() +
      theme(
        axis.text.x = element_text(angle = 90)
      )

Wind_age <- marathon_data %>%
      ggplot(aes(x = as.factor(round(Wind, 2)), y = CR_PERCENTAGE, fill = Age_group, color = Age_group)
      geom_point(alpha = 0.2) +
      geom_smooth(aes(group = Age_group, color = Age_group), se = T) +
      scale_fill_manual(values = age_group_colors) +
      scale_color_manual(values = age_group_colors) +
      labs(title = "Wind Effect on Marathon Performance by Age",
           x = "Wind",
           y = "Percentage of Completion Time",
           fill = "Age Group",
           color = "Age Group") +
```

```
      theme_minimal() +
      theme(
        axis.text.x = element_text(angle = 90)
      )

# SR Effects
SR_sex <- marathon_data %>%
      ggplot(aes(x = as.factor(round(SR, 2)), y = CR_PERCENTAGE, color = Sex)) +
      geom_boxplot(alpha = 0.7) +
      geom_smooth(aes(group = Sex), se = FALSE) +
      labs(title = "SR Effect on Marathon Performance by Sex",
           x = "SR",
           y = "Completion Time") +
      theme_minimal() +
      theme(
        axis.text.x = element_text(angle = 90)
      )

SR_age <- marathon_data %>%
      ggplot(aes(x = as.factor(round(SR, 2)), y = CR_PERCENTAGE, fill = Age_group, color = Age_group)) +
      geom_point(alpha = 0.2) +
      geom_smooth(aes(group = Age_group, color = Age_group), se = T) +
      scale_fill_manual(values = age_group_colors) +
      scale_color_manual(values = age_group_colors) +
      labs(title = "SR Effect on Marathon Performance by Age",
           x = "SR",
           y = "Percentage of Completion Time",
           fill = "Age Group",
           color = "Age Group") +
      theme_minimal() +
      theme(
        axis.text.x = element_text(angle = 90)
      )

weather_effect <- (RH_sex | RH_age) / (Wind_sex | Wind_age) / (SR_sex | SR_age) +
  plot_layout(guides = "collect", axis_titles = "collect") &
  theme(legend.position = 'bottom')
ggsave("../Plots/weather_effect.png", plot = weather_effect, width = 20, height = 30, dpi = 300)
weather_effect

# RH model
RH_model <- glm(CR_PERCENTAGE ~ RH + Sex + Age, data = marathon_data)
kable(summary(RH_model)$coefficients, caption = "RH Model Coefficients")

# SR model
SR_model <- glm(CR_PERCENTAGE ~ SR + Sex + Age, data = marathon_data)
kable(summary(SR_model)$coefficients, caption = "SR Model Coefficients")

# DP model
DP_model <- glm(CR_PERCENTAGE ~ DP + Sex + Age, data = marathon_data)
kable(summary(DP_model)$coefficients, caption = "DP Model Coefficients")

# Wind model
```

```r
Wind_model <- glm(CR_PERCENTAGE ~ Wind + Sex + Age, data = marathon_data)
kable(summary(Wind_model)$coefficients, caption = "Wind Model Coefficients")

# Flag model
Flag_model <- glm(CR_PERCENTAGE ~ Flag + Sex + Age, data = marathon_data)
kable(summary(Flag_model)$coefficients, caption = "Flag Model Coefficients")

# WBGT model
WBGT_model <- glm(CR_PERCENTAGE ~ WBGT + Sex + Age, data = marathon_data)
kable(summary(WBGT_model)$coefficients, caption = "WBGT Model Coefficients")

# linear model
lm_model_1 <- glm(CR_PERCENTAGE ~ RH + SR + DP + Wind + WBGT + Sex + Age, data = marathon_data)
kable(summary(lm_model_1)$coefficients, caption = "Model with WBGT Coefficients")

lm_mode_2 <- glm(CR_PERCENTAGE ~ RH + SR + DP + Wind + Flag + Sex + Age, data = marathon_data)
kable(summary(lm_mode_2)$coefficients, caption = "Model with Flag Coefficients")
```