London Marathon Proposal

Anant, Nikunj, Badal, Maitri

Introduction

The dataset from the London Marathon, made available by (2022) as part of the London-Marathon R package, provides a glimpse into various elements of this renowned race. It consists of two datasets obtained from Wikipedia as of November 1,2022:

Winners.csv : This dataset outlines the champions of multiple race categories, featuring their names, nationalities, finishing time and the year of the event.

london_marathon.csv : This dataset encompasses overall statistics such as the count of applicants, accepted entrants, starters and finishers, along with the total amount raised for charity and the official charity associated with each year.

The dataset facilitates the examination of trends in winning performances, participation patterns and the influence of the marathon on charitable donations. This analysis seeks to reveal significant insights and address particular research queries through statistical techniques and modeling.

The data was retrieved from the (2024), with additional information on its origin available in (2023) entry, "Scraping London Marathon data with {rvest}".

Data Preprocessing

The Category, Athlete, and Nationality feature in Winners dataset are then factorized for further analysis.

The data is preprocessed to handle NA values in Raised column in London_marathon dataset by replacing NA values with 0.

Numerical Summaries

Table 1: Table representing the columns that have abnormality

Column Name	Min	1st Quartile	Median	3rd Quartile	Max	Mean
Time.Seconds	5187	6550	7675	8414	16143	7608
Year	1981	1992	2002	2012	2022	2002

- The Years of data span from 1981 to 2022 in winners whereas, there is data from 1981 to 2020 on london marathons.
- The highest time to finish a marathon is 16143 seconds which is a outlier.

Table 2: Table shows the statistics of crucial columns in London Marathon Dataset.

Column Name	Min	1st Quartile	Median	3rd Quartile	Max	Mean
Accepted	77	33057	43057	49903	56398	39269
Starters	77	24488	31369	35671	42906	28886
Finishers	61	23252	30584	35326	42549	28145

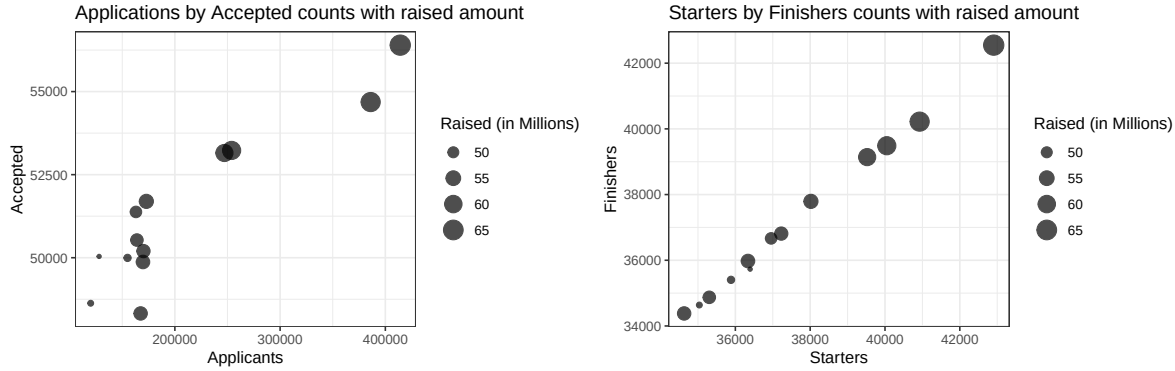
- In one of the marathons only 77 applicants were accepted and started the marathon.

Table 3: Table displays the row from dataset that have only 77 accepted participants

Date	Year	Applicants	Accepted	Starters	Finishers	Raised
2020-10-04	2020	457861	77	77	61	0

By subsetting the dataframe, the marathon was conducted in 2020 and there were 4.5 lakhs of applicants but only 77 were accepted.

Visualization



(a) Applicants and Accepted individuals

(b) Starters and Finishers of marathon

Figure 1: Plots showing the relationship of Applicants - Accepted and Finishers - Starters where there is amount of charity is Raised

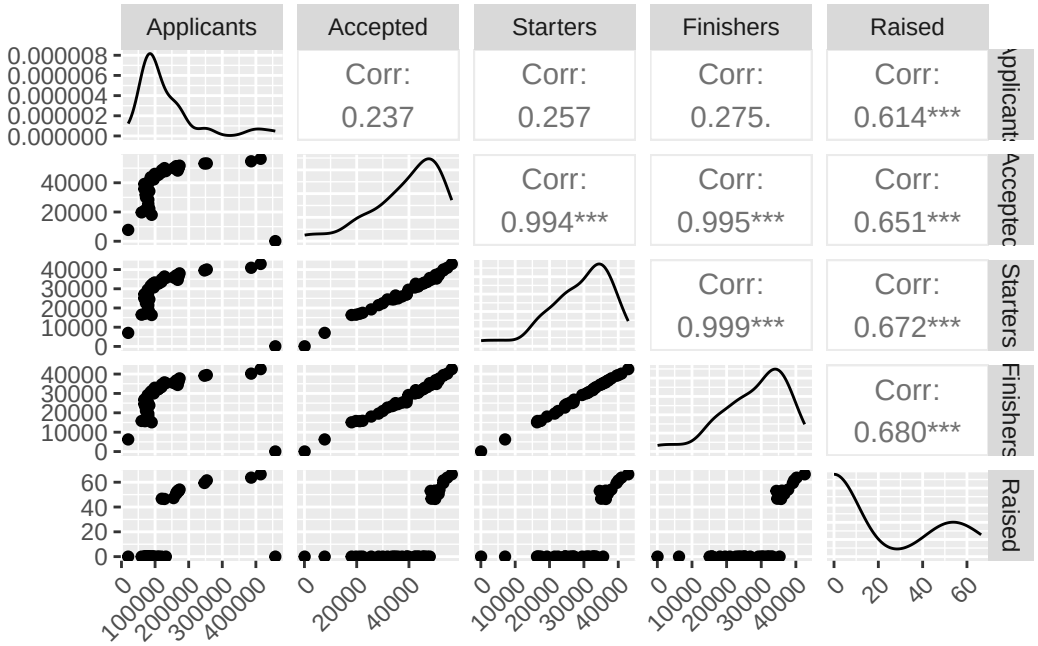


Figure 2: Pair plot between applicants, accepted participants, starters, finishers, and charity amount raised from the marathon

From the scattered bubble plot it is evident that the raised amount increases with the increase in applicants and accepted. Same goes with starters and finishers, there is linearity and the

amount raised also increases.

From the above pair plot it is clear that there is high correlation between Applicants, Accepted, Starters, Finishers, and Raised.

Question: Can we predict the possible charity that can be raised in upcoming london marathons based on the count of Applicants, Accepted participants, starters, and finishers?

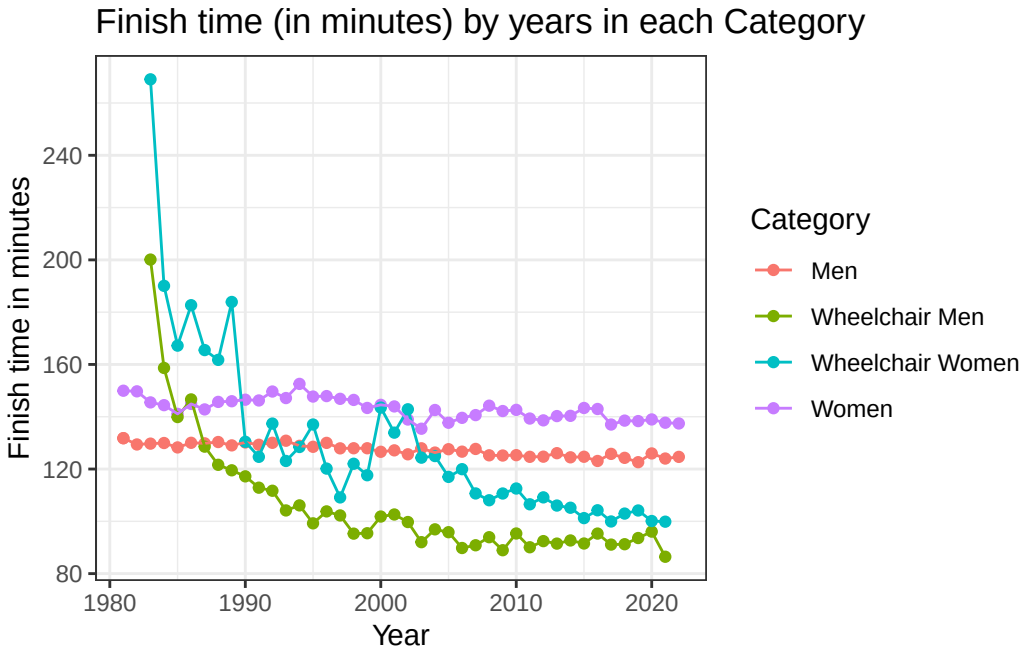


Figure 3: Plot shows the finishing time of each category in each year

Question: Wheelchair individuals have some correlation with time to finish the race? OR Does Category have relation with the finish time over the years?

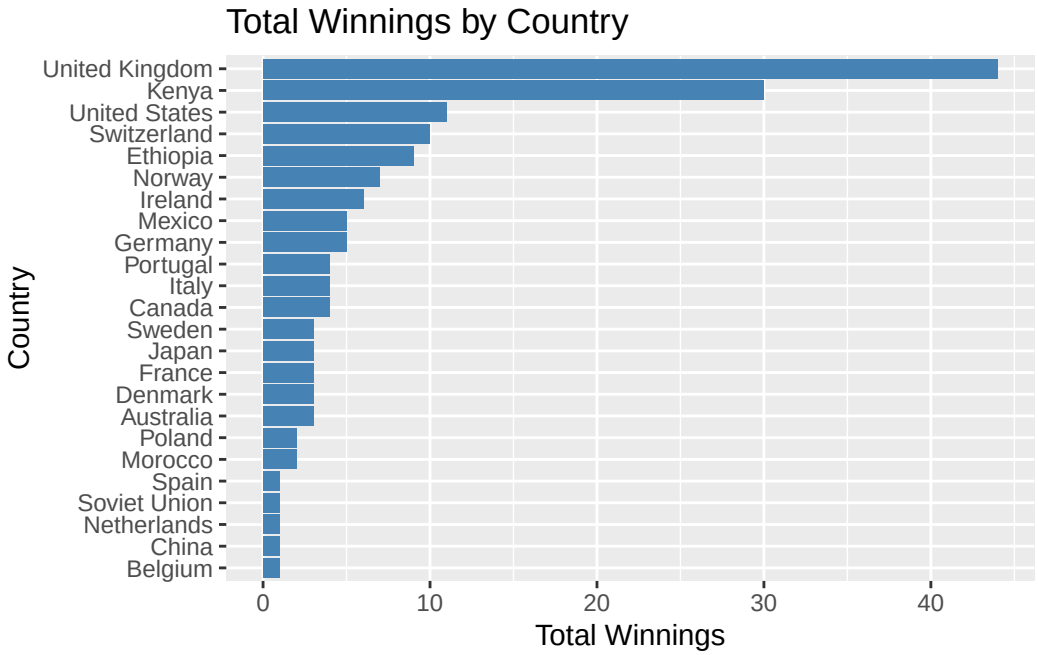


Figure 4: The horizontal barplot show the total winnings of the country in descending order

Analysis

Q1 : Can we predict the possible charity that can be raised in upcoming london marathons based on the count of Applicants, Accepted participants, starters, and finishers?

Assumptions checking

Linearity: From the pair plot it is clear that the relation is linear between Applicants, Accepted participants, starters, finishers, and Raised

Independence: The observation in this data set is independent and thus have independence.

Normality of residuals

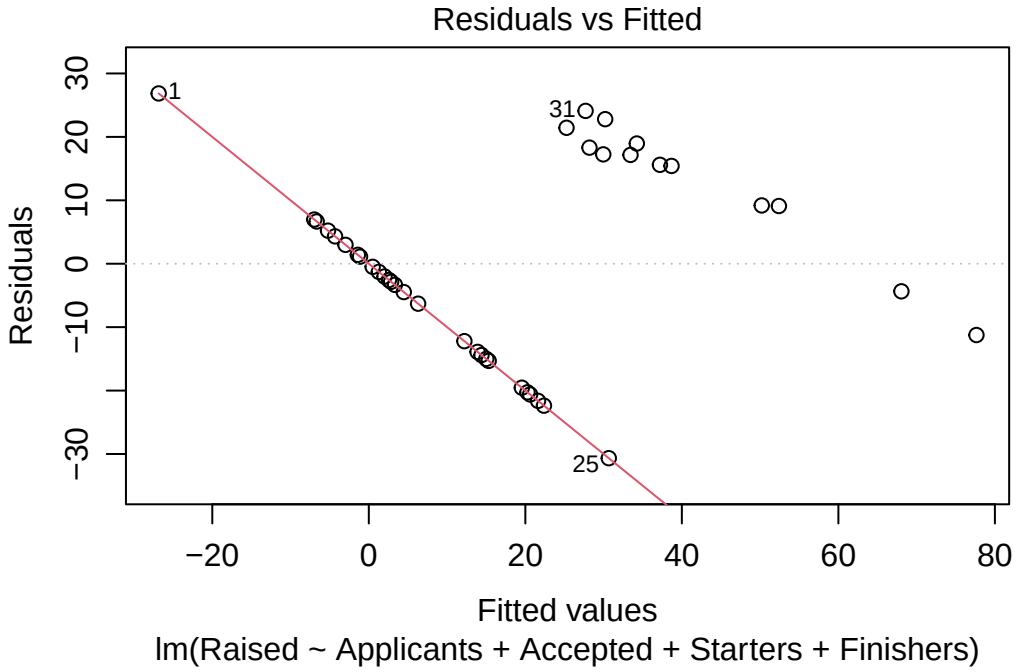
Table 4: Table represents the intercept and slopes of variable in linear model.

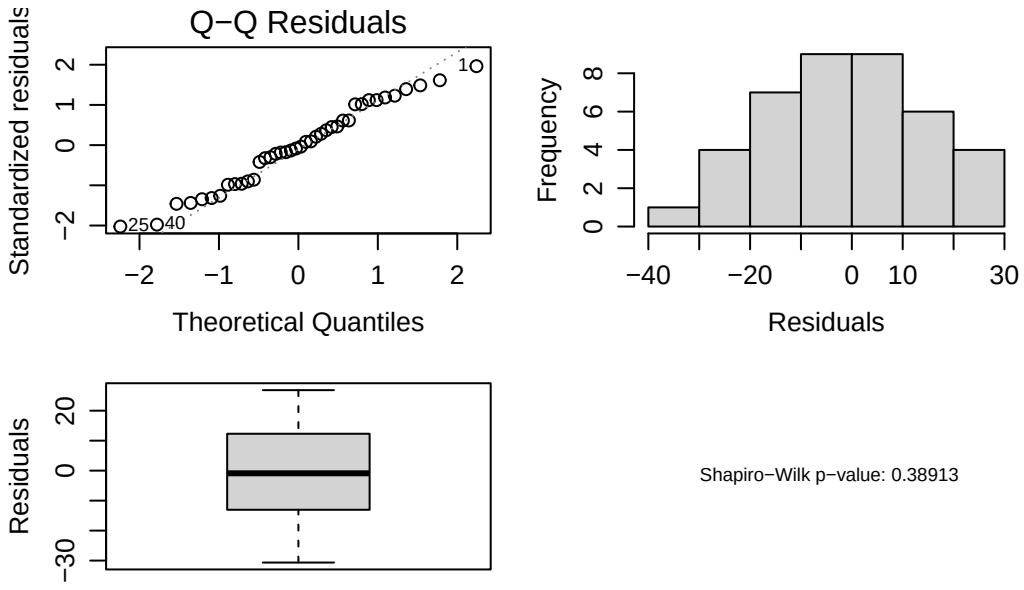
Coefficient	Estimate	Std. Error	t value	Pr(>
(Intercept)	-40.94549533	12.37057088	-3.310	0.002171 **
Applicants	0.00011584	0.00003147	3.681	0.000777 ***

Coefficient	Estimate	Std. Error	t value	Pr(>
Accepted	-0.00192098	0.00196776	-0.976	0.335653
Starters	0.00046931	0.00631594	0.074	0.941190
Finishers	0.00373246	0.00673803	0.554	0.583143

Table 5: Table displays the results of linear model

Metric	Value
Residual standard error	15.8 on 35 degrees of freedom
Multiple R-squared	0.6696
Adjusted R-squared	0.6319
F-statistic	17.73 on 4 and 35 DF
p-value	0.00000004866





The residuals are normally distributed as per the above plots

Q2: Wheelchair individuals have some correlation with time to finish the race? OR Does Category have relation with the finish time over the years?

```
             Df    Sum Sq  Mean Sq F value            Pr(>F)
Category      3 103742537 34580846   23.97 0.000000000000749 ***
Residuals   159 229379141  1442636
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = Time.Seconds ~ Category, data = winners)

$Category
                                     diff        lwr       upr     p adj
Wheelchair Men-Men              -1300.7674 -1990.3587 -611.1762 0.0000140
Wheelchair Women-Men              137.7967  -551.7946  827.3879 0.9544733
Women-Men                         941.3992   264.8460 1617.9524 0.0022746
Wheelchair Women-Wheelchair Men  1438.5641   732.3538 2144.7744 0.0000024
Women-Wheelchair Men             2242.1667  1548.6819 2935.6514 0.0000000
Women-Wheelchair Women            803.6026   110.1178 1497.0873 0.0159801
```

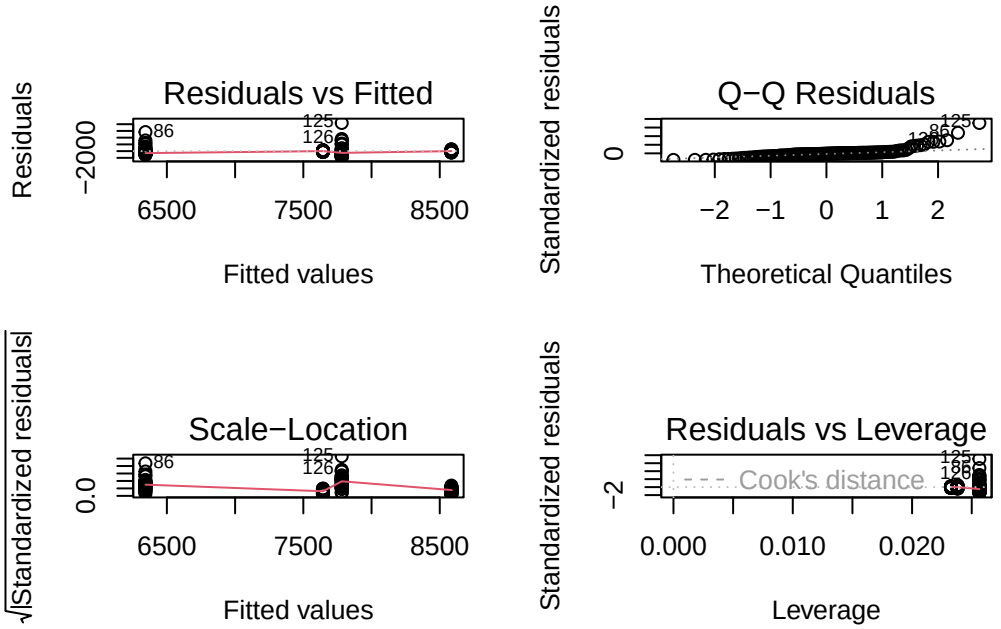


Table 6: Table shows the results of ANOVA test

Source	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Category	3	103742537	34580846	23.97	(7.49 ×10^{-13})***
Residuals	159	229379141	1442636		

ANOVA has proved that there are differences among the group means as p-value < 0.05

Table 7: Table shows the TukeyHSD Post Hoc test

Comparison	Difference	Lower Bound	Upper Bound	Adjusted p-value
Wheelchair Men - Men	-1300.77	-1990.36	-611.18	0.0000140
Wheelchair Women - Men	137.80	-551.79	827.39	0.9544733
Women - Men	941.40	264.85	1617.95	0.0022746
Wheelchair Women - Wheelchair Men	1438.56	732.35	2144.77	0.0000024
Women - Wheelchair Men	2242.17	1548.68	2935.65	0.0000000
Women - Wheelchair Women	803.60	110.12	1497.09	0.0159801

A TukeyHSD Post-hoc test is coducted shows following results:

- Wheelchair Men are significantly faster than Men, with a mean difference of -1300.77 seconds (p = 0.0000140).
- There is no significant difference between Wheelchair Women and Men, with a mean difference of 137.80 seconds (p = 0.9544733).
- Women are significantly slower than Men, with a mean difference of 941.40 seconds (p = 0.0022746).
- Wheelchair Women are significantly slower than Wheelchair Men, with a mean difference of 1438.56 seconds (p = 0.0000024).
- Women are significantly slower than Wheelchair Men, with a mean difference of 2242.17 seconds (p = 0.0000000).
- Women are significantly slower than Wheelchair Women, with a mean difference of 803.60 seconds (p = 0.0159801).

Q3: Is the proportions of winners from each country same?

Table 8: Table shows the results of chi square goodness of fit

Statistic	Value
Chi-squared	334.52
Degrees of Freedom (df)	23
p-value	< 0.00000000000000022

The Chi-square test for goodness of fit has p-value < 0.05 that means we reject the null hypothesis and conclude that there are differences in the proportion of total winnings based on country.

Bibliography

Community, Data Science Learning. 2024. "Tidy Tuesday: A Weekly Social Data Project." https://tidytues.day.

Rennie, Nicola. 2022. "London Marathon r Package: Dataset on London Marathon Winners." https://github.com/nrennie/LondonMarathon.

———. 2023. "Scraping London Marathon Data with rvest." March 16, 2023. https://nrennie.rbind.io/blog/web-scraping-rvest-london-marathon.

Appendix

Code

```
suppressWarnings(library(dplyr))
suppressWarnings(library(ggplot2))
suppressWarnings(library(lubridate))
suppressWarnings(library(GGally))
#libraries
library(lubridate)
library(dplyr)
library(ggplot2)
library(GGally)
# Loading data
tuesdata <- tidytuesdayR::tt_load('2023-04-25')
tuesdata <- tidytuesdayR::tt_load(2023, week = 17)

winners <- tuesdata$winners
london_marathon <- tuesdata$london_marathon

# Convert the time from hour:min:seconds to seconds
winners$Time.Seconds <- period_to_seconds(hms(winners$Time))

# Factoring the variables
winners$Category <- factor(winners$Category)
winners$Athlete <- factor(winners$Athlete)
winners$Nationality <- factor(winners$Nationality)

# Handling the NA in Raised
london_marathon$Raised[is.na(london_marathon$Raised)] = 0
london_marathon <- london_marathon[rowSums(is.na(london_marathon)) <= 2,]
summary(winners)
summary(london_marathon)
# Subset the dataframe for starters = 77
london_marathon[london_marathon$Starters == 77,]
# Option to print values without scientific notation
options(scipen = 999)
# Applicants vs Accepted participants by amount raised
london_marathon %>%
  filter(Raised > 0 ) %>%
  ggplot(aes(x=Applicants, y = Accepted, size = Raised)) +
```

```
  geom_point(alpha = 0.7) +
  scale_size_continuous(name = "Raised (in Millions)")+
  labs(title="Applications by Accepted counts with raised amount") +
  theme_bw()

# Starters and finishers by amount raised
london_marathon %>%
  filter(Raised > 0 ) %>%
  ggplot(aes(x=Starters, y = Finishers, size = Raised)) +
  geom_point(alpha = 0.7) +
  scale_size_continuous(name = "Raised (in Millions)")+
  labs(title="Starters by Finishers counts with raised amount") +
  theme_bw()

# ggpairs plot
london_marathon[,c("Applicants","Accepted","Starters","Finishers","Raised")] %>%
  # filter(Raised > 0) %>%
  ggpairs() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) # chat gpt helped me
# Year vs Time by Category
winners %>%
  ggplot(aes(x = Year, y = Time.Seconds / 60, color = Category)) +
  geom_point() +
  geom_line() +
  labs(
    title = "Finish time (in minutes) by years in each Category",
    x = "Year",
    y = "Finish time in minutes"
  ) +
  theme_bw()

winners_count <- data.frame(table(winners$Athlete))
names(winners_count) <- c("Athlete","Frequency")
#
winners_nationality <- unique(left_join(winners_count,
                                        winners[,c("Athlete","Nationality")],
                                        by="Athlete"))
grouped_nationality <- winners_nationality %>%
  group_by(Nationality) %>%
  summarise(Total_winnings = sum(Frequency))

grouped_nationality %>%
```

```r
  ggplot(aes(x=reorder(Nationality,Total_winnings), y=Total_winnings)) +
  geom_bar(stat="identity",fill="steelblue") +
  labs(title="Total Winnings by Country", x = "Country", y = "Total Winnings")+
  coord_flip()

names(london_marathon)
# raised_mod <- lm(Raised ~ Applicants + Accepted + Starters + Finishers, data = london_marat
raised_mod <- lm(Raised ~ Applicants + Accepted + Starters + Finishers, data = london_maratho
summary(raised_mod)
par(mar = c(4.4, 3.5, 1.2,1), mgp = c(2.25, 0.8, 0))
plot(raised_mod, which = 1)

par(mar = c(3.5, 3.5, 1.2,1), mgp = c(2.25, 0.8, 0), mfrow = c(2,2))
plot(raised_mod, which = 2)
hist(raised_mod$residuals, xlab = "Residuals", main = "")
boxplot(raised_mod$residuals, ylab = "Residuals")
plot(0, 0, col = "white", bty = 'n', xaxt = 'n', yaxt = 'n',
xlab = "", ylab = "")
text(-0.8, 0, paste("Shapiro-Wilk p-value:",
shapiro.test(raised_mod$residuals)$p.value %>% round(5)), cex = 0.75, pos = 4)
winners$Category <- as.factor(winners$Category)

# Fit the two-way ANOVA model with interaction
# anova_model <- aov(Time.Seconds ~ Category * as.factor(Year), data = winners)
anova_model <- aov(Time.Seconds ~ Category, data = winners )

# names(winners)
# Summary of the ANOVA
summary(anova_model)

# Post-hoc analysis using Tukey's HSD to compare categories
tukey_results <- TukeyHSD(anova_model, "Category")
print(tukey_results)

# Diagnostic plots
par(mfrow = c(2, 2))
plot(anova_model)
# Empty vector
nationality_vec <- c()

# Iterating the rows of dataframe and adding values into nationality_vec
for (i in 1:nrow(grouped_nationality)) {
```

```
  row_data <- grouped_nationality[i, ]
  # Access individual values using column names or indexing
  country <- row_data$Nationality
  total_winnings <- row_data$Total_winnings
  # print(country)
  nationality_vec <- c(nationality_vec,rep(paste(country),total_winnings))
  # Do something with the row data
  # print(paste("Country:", country, ", Total Winnings:", total_winnings))
}

# Contingency table for the vector
contingency_tab <- table(nationality_vec)

# Perform the chi-square test
chi_sq_test <- chisq.test(contingency_tab)

# Print the results
print(chi_sq_test)
```