Olympic_Athletes

Kanye Smith

2023-06-23

R Markdown

Abstract: Throughout this project, I will analyze the correlation between the physical attributes of Olympic Athletes, as well as conduct a hypothesis test to determine the impact of gender on overall Olympic event performance. To assess the condition of the raw dataset, multiple functions were utilized to detect the presence of NA values and duplicate rows. Moreover, due to the vastness of the dataset, I used five basic research questions to critique the feasibility of testing the data. Regarding the statistical analysis portion of the project, both a simple Linear Regression Model and a Partial Least Squares Regression Model were created in relation to the correlation aspect of the project with the goal of determining which factors were significant in predicting athlete weights. After comparing the results of both models, there were noticeable similarities between the outputs, which led to the Simple Liner Regression Model to be deemed best suited for analysis. Furthermore, a Welch Two-Sample T-Test conducted with male and female gold medalists determined that sex had a slight significance regarding which group was more likely to win.

Introduction:

Introduced in Ancient Greece in the year 776 BCE, the Olympic Games are an international sporting event that gathers athletes from across the globe to compete in various sports/ events. While the more ancient games consisted of events such as javelin throws and combat, the Modern Olympic Games (1896 to present) included more diverse activities such as table tennis, basketball, and swimming. Thus, influenced by the lack of collected data, the athlete_event dataset only includes the Olympic Games from 1896 to 2016. As the trend of Olympic Games continues to occur every 4 years, an influx of athletes from different backgrounds adapts the conditions of the event.

The various analysis techniques introduced within this project will aim to answer two questions: 1)Is there a correlation between the physical attributes of a particular athlete? 2)Does gender have an impact on Olympic event performance?

Data Cleaning and Wrangling The athlete_events dataset is a csv file that aims to describe the performance of Olympic Athletes in their respective events over the course of 120 years, which is roughly 1896 to 2016. Retrieved from a Kaggle post, the dataset was initially consisted of 271116 rows and 15 variable columns: ID, Name, Sex, Age, Height, Weight, Team, NOC, Games, Year, Season, City, Sport, Event, and Medal. Out of these columns, four were integer type, one was numeric type, and ten were character type.

```r
athlete_events <- read.csv("C:\\Users\\House\\Downloads\\athlete_events.csv")
print(head(athlete_events))
```

```
##   ID                     Name Sex Age Height Weight           Team NOC
## 1  1                A Dijiang   M  24    180     80          China CHN
## 2  2                 A Lamusi   M  23    170     60          China CHN
## 3  3      Gunnar Nielsen Aaby   M  24     NA     NA        Denmark DEN
## 4  4     Edgar Lindenau Aabye   M  34     NA     NA Denmark/Sweden DEN
## 5  5 Christine Jacoba Aaftink   F  21    185     82    Netherlands NED
## 6  5 Christine Jacoba Aaftink   F  21    185     82    Netherlands NED
##         Games Year Season      City         Sport
## 1 1992 Summer 1992 Summer Barcelona    Basketball
## 2 2012 Summer 2012 Summer    London          Judo
## 3 1920 Summer 1920 Summer Antwerpen      Football
## 4 1900 Summer 1900 Summer     Paris    Tug-Of-War
## 5 1988 Winter 1988 Winter   Calgary Speed Skating
## 6 1988 Winter 1988 Winter   Calgary Speed Skating
##                                Event Medal
## 1        Basketball Men's Basketball  <NA>
## 2       Judo Men's Extra-Lightweight  <NA>
## 3            Football Men's Football  <NA>
## 4        Tug-Of-War Men's Tug-Of-War  Gold
## 5   Speed Skating Women's 500 metres  <NA>
## 6 Speed Skating Women's 1,000 metres  <NA>
```

```
print(str(athlete_events))
```

```
## 'data.frame':    271116 obs. of  15 variables:
##  $ ID    : int  1 2 3 4 5 5 5 5 5 5 ...
##  $ Name  : chr  "A Dijiang" "A Lamusi" "Gunnar Nielsen Aaby" "Edgar
Lindenau Aabye" ...
##  $ Sex   : chr  "M" "M" "M" "M" ...
##  $ Age   : int  24 23 24 34 21 21 25 25 27 27 ...
##  $ Height: int  180 170 NA NA 185 185 185 185 185 185 ...
##  $ Weight: num  80 60 NA NA 82 82 82 82 82 82 ...
##  $ Team  : chr  "China" "China" "Denmark" "Denmark/Sweden" ...
##  $ NOC   : chr  "CHN" "CHN" "DEN" "DEN" ...
##  $ Games : chr  "1992 Summer" "2012 Summer" "1920 Summer" "1900 Summer"
...
##  $ Year  : int  1992 2012 1920 1900 1988 1988 1992 1992 1994 1994 ...
##  $ Season: chr  "Summer" "Summer" "Summer" "Summer" ...
##  $ City  : chr  "Barcelona" "London" "Antwerpen" "Paris" ...
##  $ Sport : chr  "Basketball" "Judo" "Football" "Tug-Of-War" ...
##  $ Event : chr  "Basketball Men's Basketball" "Judo Men's Extra-
Lightweight" "Football Men's Football" "Tug-Of-War Men's Tug-Of-War" ...
##  $ Medal : chr  NA NA NA "Gold" ...
## NULL
```

When attempting to utilize the raw data for statistical analysis, the most prominent issues were the potential detection of missing data values, as well as duplicated rows. Due to the massive number of rows included in the dataset, roughly 271,000, it seemed impractical to manually review the individual column values. Therefore, the head() function was implemented to view the conditions of the top six columns. Upon review, it was noticeable that there were multiple occurrences of NA values within columns such as Age, Weight, Height, and Medal. To address this issue, 4 detection methods were called:

```
print(mean(athlete_events$Age))

## [1] NA

print(mean(athlete_events$Height))

## [1] NA

print(mean(athlete_events$Weight))

## [1] NA

print(mean(athlete_events$Year))

## [1] 1978.378

print(summary(athlete_events))

##        ID             Name               Sex                 Age
##  Min.   :     1   Length:271116      Length:271116      Min.   :10.00
##  1st Qu.: 34643   Class :character   Class :character   1st Qu.:21.00
##  Median : 68205   Mode  :character   Mode  :character   Median :24.00
##  Mean   : 68249                                         Mean   :25.56
##  3rd Qu.:102097                                         3rd Qu.:28.00
##  Max.   :135571                                         Max.   :97.00
##                                                         NA's   :9474
##      Height         Weight          Team              NOC
##  Min.   :127.0   Min.   : 25.0   Length:271116      Length:271116
##  1st Qu.:168.0   1st Qu.: 60.0   Class :character   Class :character
##  Median :175.0   Median : 70.0   Mode  :character   Mode  :character
##  Mean   :175.3   Mean   : 70.7
##  3rd Qu.:183.0   3rd Qu.: 79.0
##  Max.   :226.0   Max.   :214.0
##  NA's   :60171   NA's   :62875
##     Games              Year         Season             City
##  Length:271116      Min.   :1896   Length:271116      Length:271116
##  Class :character   1st Qu.:1960   Class :character   Class :character
##  Mode  :character   Median :1988   Mode  :character   Mode  :character
##                     Mean   :1978
##                     3rd Qu.:2002
##                     Max.   :2016
##
##     Sport              Event              Medal
##  Length:271116      Length:271116      Length:271116
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
```

```
##
##
```

```
sum_na<-sapply(athlete_events, function(athlete_events)
sum(length(which(is.na(athlete_events)))))
print(sum_na)
```

```
##     ID   Name    Sex    Age Height Weight   Team    NOC  Games   Year
Season
##      0      0      0   9474  60171  62875      0      0      0      0
0
##   City  Sport  Event  Medal
##      0      0      0 231333
```

Within the dataset, there were 4 columns of the “integer” and “Numeric” data type, namely Age, Height, Weight, and Year. Therefore, to confirm the existence of potential NA values, the mean() function returned an error(or NA). As seen in the output, the mean calculation for the Age, Height, and Weight columns returned a value of NA. However, the Year column provided a mean of approx. 1978.38. Considering that the purpose of the Olympics dataset is to analyze the outcomes of the events from the past 120 years, the calculated mean for the Year column wouldn’t provide a meaningful outlook to the trends of the data. Thus, while the lack of errors was acknowledged for cleaning, it was ultimately overlooked for analysis purposes. Moreover, although the is.na() function was initially useful for gathering a breakdown of each column on the grounds of whether a particular value was missing(TRUE) or included(FALSE), the given summary and sapply outputs listed the sum of all missing values associated with a column variable. Strictly comparing the data in the Age, Weight, Height, and Year columns, it was important to note the trend of NAs appearing in columns related to an athlete’s physical attributes. However, considering the dataset in its entirety, the sapply() function detected missing values under the Medal column as well, whereas the summary() function didn’t.

When interpreting the lack of data in the context of being able to determine the relationship between physical attributes and Olympic event performance, there had to be a level of distinction between the types of missing values associated with each column. That is, regarding variables such as Age, Weight, and Height, an NA is simply translated to a value that was missed or lost. On the other hand, NA detection pertaining to accumulated Medals expresses that an athlete wasn’t awarded a medal in their respective event. As such, replacing the “NA” under Medal to “None” prevented the entries from being recognized as missing while still acknowledging their Olympic Event performance. In addition, through the implementation of the drop_na function, the rows containing the remaining missing rows were dropped.

```
library(stringr)
athlete_events$Medal <- str_replace(athlete_events$Medal, "NA", "NONE")
```

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.3.1
## Warning: package 'tidyr' was built under R version 4.3.1

## Warning: package 'readr' was built under R version 4.3.1

## Warning: package 'purrr' was built under R version 4.3.1

## Warning: package 'dplyr' was built under R version 4.3.1

## Warning: package 'forcats' was built under R version 4.3.1

## Warning: package 'lubridate' was built under R version 4.3.1

## ── Attaching core tidyverse packages ──────────────────────── tidyverse
2.0.0 ──
## ✔ dplyr     1.1.2     ✔ purrr     1.0.1
## ✔ forcats   1.0.0     ✔ readr     2.1.4
## ✔ ggplot2   3.4.2     ✔ tibble    3.2.1
## ✔ lubridate 1.9.2     ✔ tidyr     1.3.0
## ── Conflicts ──────────────────────────────────────────
tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all
conflicts to become errors
```

```
athlete_events <- athlete_events %>% drop_na(Age, Height, Weight)
print(summary(athlete_events))
```

```
##        ID             Name               Sex                 Age
##  Min.   :     1   Length:206165      Length:206165      Min.   :11.00
##  1st Qu.: 35194   Class :character   Class :character   1st Qu.:21.00
##  Median : 68629   Mode  :character   Mode  :character   Median :24.00
##  Mean   : 68616                                         Mean   :25.06
##  3rd Qu.:102313                                         3rd Qu.:28.00
##  Max.   :135571                                         Max.   :71.00
##      Height          Weight           Team               NOC
##  Min.   :127.0   Min.   : 25.00   Length:206165      Length:206165
##  1st Qu.:168.0   1st Qu.: 60.00   Class :character   Class :character
##  Median :175.0   Median : 70.00   Mode  :character   Mode  :character
##  Mean   :175.4   Mean   : 70.69
##  3rd Qu.:183.0   3rd Qu.: 79.00
##  Max.   :226.0   Max.   :214.00
##     Games                Year         Season              City
##  Length:206165      Min.   :1896   Length:206165      Length:206165
##  Class :character   1st Qu.:1976   Class :character   Class :character
##  Mode  :character   Median :1992   Mode  :character   Mode  :character
##                     Mean   :1990
##                     3rd Qu.:2006
##                     Max.   :2016
##     Sport             Event              Medal
##  Length:206165      Length:206165      Length:206165
##   Class :character    Class :character    Class :character
##   Mode  :character    Mode  :character    Mode  :character
##
##
##
```

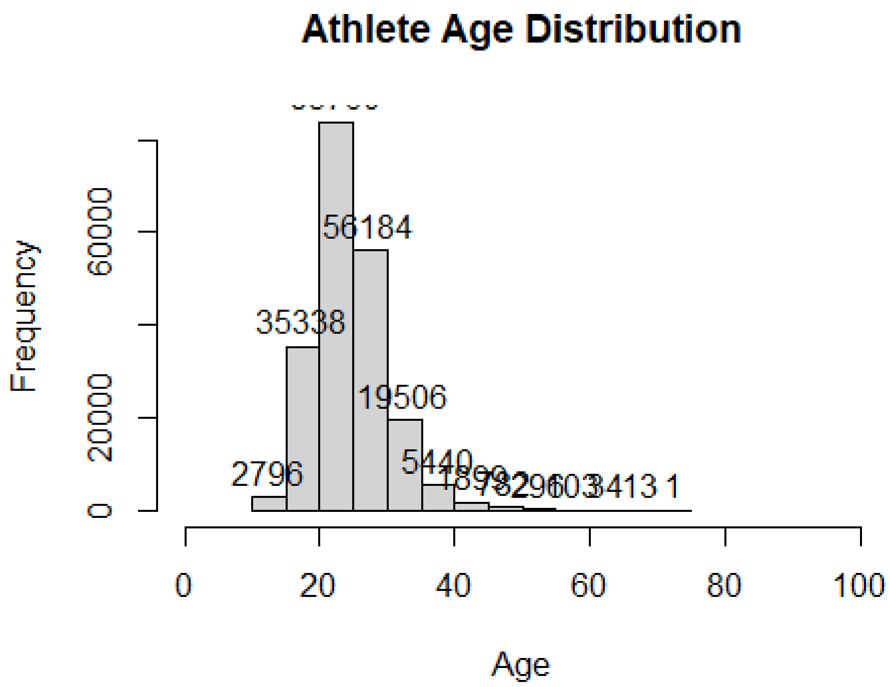
Furthermore, given the large size of the athlete_events dataset, it was prone to include some duplicate rows.

```
print(sum(duplicated(athlete_events)))

## [1] 13

athlete_events <- distinct(athlete_events)

print(sum(duplicated(athlete_events)))

## [1] 0
```

Methodology:

Moving forward with the statistical analysis of the dataset, the initial goal was to graph the frequencies of variable entries in an attempt to recognize trends within the data.

```
print(hist(athlete_events$Age, main="Athlete Age Distribution", xlab="Age",
ylab= "Frequency",xlim=c(0,100),labels = TRUE))
```



```
## $breaks
##  [1] 10 15 20 25 30 35 40 45 50 55 60 65 70 75
```

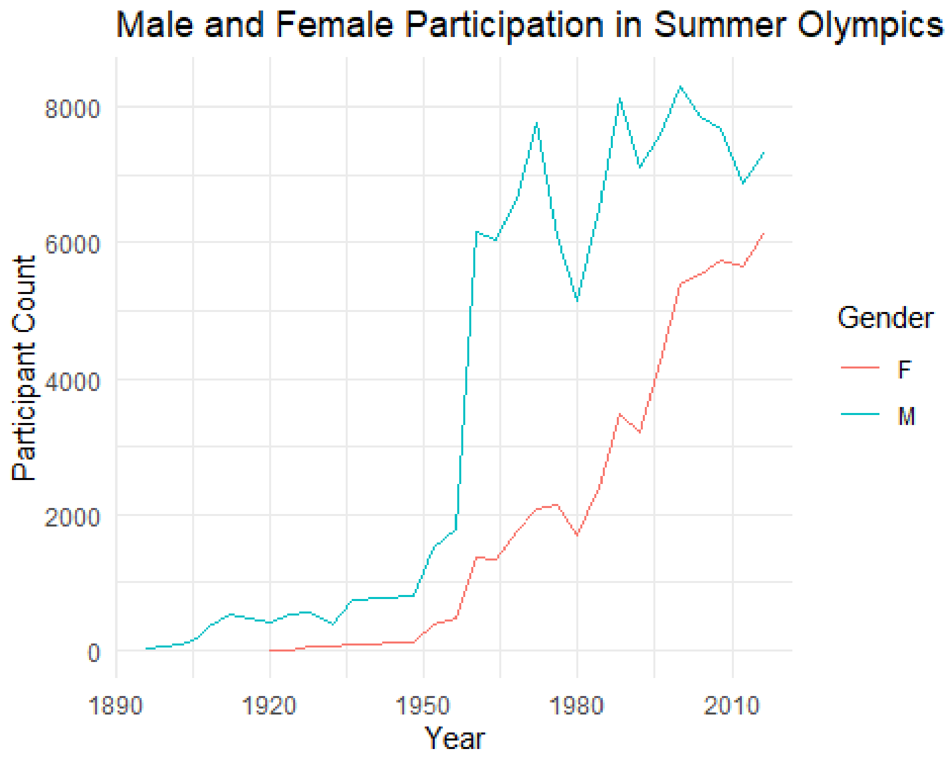
```
##
## $counts
##  [1]  2796 35338 83760 56184 19506  5440  1899   782   296   103    34
13
## [13]     1
##
## $density
##  [1] 2.712562e-03 3.428344e-02 8.126043e-02 5.450735e-02 1.892390e-02
##  [6] 5.277659e-03 1.842330e-03 7.586635e-04 2.871668e-04 9.992627e-05
## [11] 3.298537e-05 1.261205e-05 9.701579e-07
##
## $mids
##  [1] 12.5 17.5 22.5 27.5 32.5 37.5 42.5 47.5 52.5 57.5 62.5 67.5 72.5
##
## $xname
## [1] "athlete_events$Age"
##
## $equidist
## [1] TRUE
##
## attr(,"class")
## [1] "histogram"
```

Looking at the Athlete Age histogram, the decision behind placing the ages in such a plot was that by being placed in "bins" grouped by ten, the frequency of each bin compared to each other would provide evidence as to what was considered the peak or preferred age of competing athletes. As such, the plotted data had a skewed right distribution with most values congregating around ages 20 to 30. Considering more popular domestic sports leagues such as the National Basketball Association (NBA) where there is a tendency of players' peak performance occurring during the ages of 20-30, the results of the histogram were expected. Nevertheless, there were some outliers occurring within the age range between 70-80, which impacted the skewness of the histogram. Moreover, given the discovered trend of younger athletes competing in the Olympics, factors outside of peak performance such as event popularity amongst the age group should be taken into consideration.

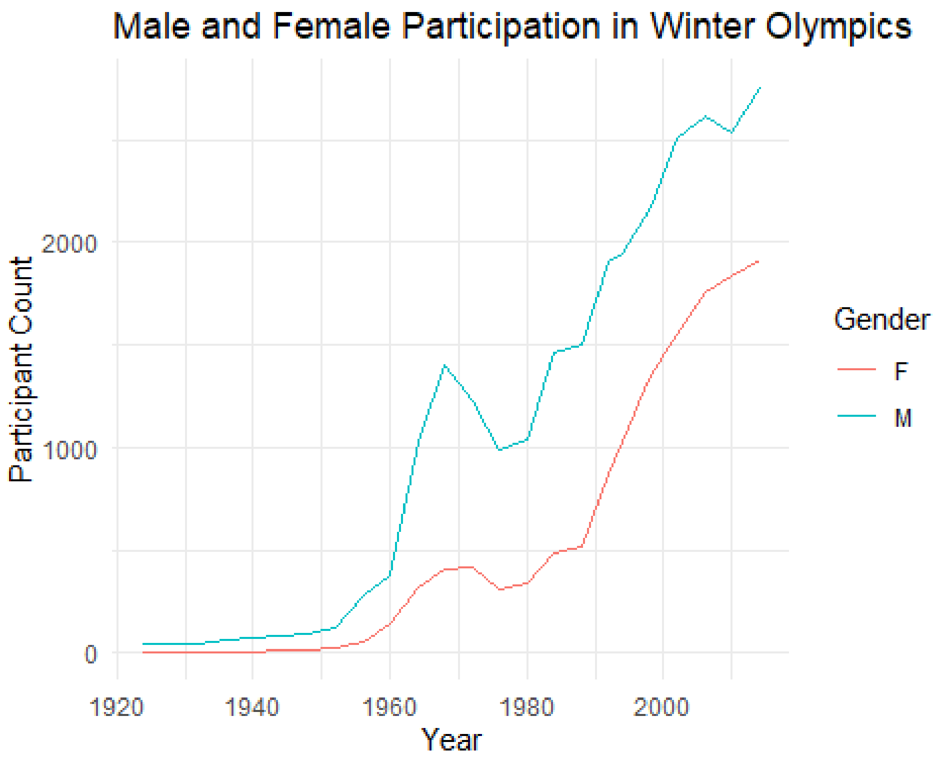
```r
summer_events <- athlete_events %>%
  filter(Season == "Summer") %>%
  select(Year, Sex)
summer_participation <- summer_events %>%
  group_by(Year, Sex) %>%
  summarise(Count = n())
```

```
## `summarise()` has grouped output by 'Year'. You can override using the
## `.groups` argument.
```

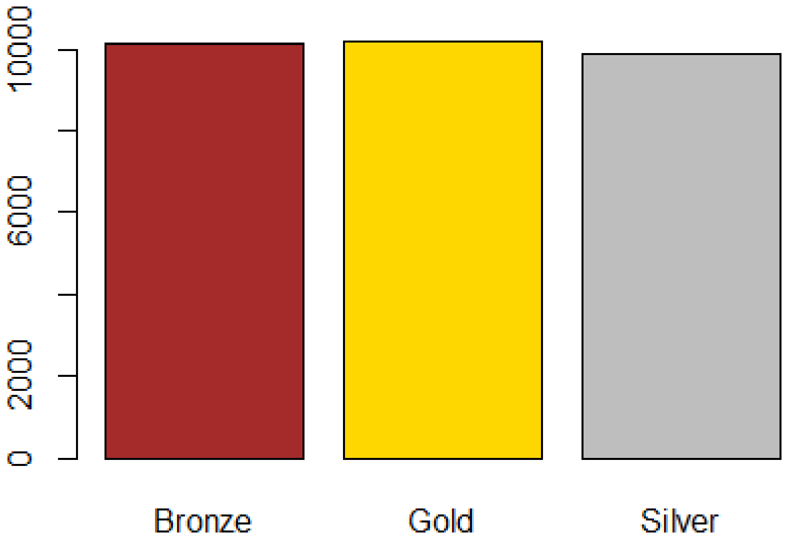
```r
print(ggplot(summer_participation, aes(x = Year, y = Count, color = Sex)) +
  geom_line() +
  labs(title = "Male and Female Participation in Summer Olympics",
       x = "Year",
       y = "Participant Count",
       color = "Gender") +
  theme_minimal())
```



```
winter_events <- athlete_events %>%
  filter(Season == "Winter") %>%
  select(Year, Sex)
winter_participation <- winter_events %>%
  group_by(Year, Sex) %>%
  summarise(Count = n())

## `summarise()` has grouped output by 'Year'. You can override using the
## `.groups` argument.

print(ggplot(winter_participation, aes(x = Year, y = Count, color = Sex)) +
  geom_line() +
  labs(title = "Male and Female Participation in Winter Olympics",
       x = "Year",
       y = "Participant Count",
       color = "Gender") +
  theme_minimal())
```



```
colors = c("brown", "gold", "grey")
print(barplot(table(athlete_events$Medal), col= colors))
```



```
##      [,1]
## [1,]  0.7
## [2,]  1.9
## [3,]  3.1
```

Regarding the “Medal Distribution” barchart, the uniformity of the data was also expected. The final distribution that was observed was the difference in Male vs. Female participation during each of the Winter and Summer Olympics. Referring to the graphs, similarities were prominent in areas such as the number of overall participants increasing over time, as well as the trend of the male participant frequency being relatively higher than the female frequency. In addition, when comparing the lines on both graphs, there is a noticeable gap between 1896 and 1900 where the female competitor data wasn’t collected. This can be explained due to the fact that women weren’t allowed to compete in the games until 1900.

Outside of overall trends within the dataset, several exploratory questions were used to analyze specific features of the data:

```
#Number of games
result <- athlete_events %>%
  distinct(Games) %>%
  count()
print(result)

##    n
## 1 51

#Specific games
result <- athlete_events %>%
  group_by(Games) %>%
  distinct(Games) %>%
  ungroup()
print(result, n = 51)

## # A tibble: 51 × 1
##    Games
##    <chr>
##  1 1992 Summer
##  2 2012 Summer
##  3 1988 Winter
##  4 1992 Winter
##  5 1994 Winter
##  6 2002 Winter
##  7 1980 Winter
##  8 2000 Summer
##  9 1996 Summer
## 10 2014 Winter
## 11 1948 Summer
## 12 1952 Summer
## 13 1998 Winter
## 14 2006 Winter
## 15 2016 Summer
## 16 2004 Summer
## 17 1984 Winter
## 18 1968 Summer
## 19 1972 Summer
## 20 1936 Summer
## 21 1956 Summer
## 22 1960 Summer
## 23 1920 Summer
## 24 1924 Summer
## 25 1928 Summer
## 26 2008 Summer
## 27 1976 Summer
## 28 1988 Summer
## 29 1980 Summer
## 30 1984 Summer
## 31 1964 Summer
## 32 2010 Winter
## 33 1964 Winter
## 34 1968 Winter
## 35 1972 Winter
## 36 1976 Winter
## 37 1924 Winter
## 38 1912 Summer
## 39 1932 Summer
## 40 1932 Winter
## 41 1936 Winter
## 42 1928 Winter
## 43 1908 Summer
## 44 1956 Winter
## 45 1960 Winter
## 46 1952 Winter
## 47 1948 Winter
## 48 1906 Summer
## 49 1904 Summer
## 50 1900 Summer
## 51 1896 Summer
```

```
#Counts amount of games held in the summer
distinct_games_count <- athlete_events %>%
  filter(Season == 'Summer') %>%
  summarize(count = n_distinct(Games))
count_value <- distinct_games_count$count
print(distinct_games_count)
```

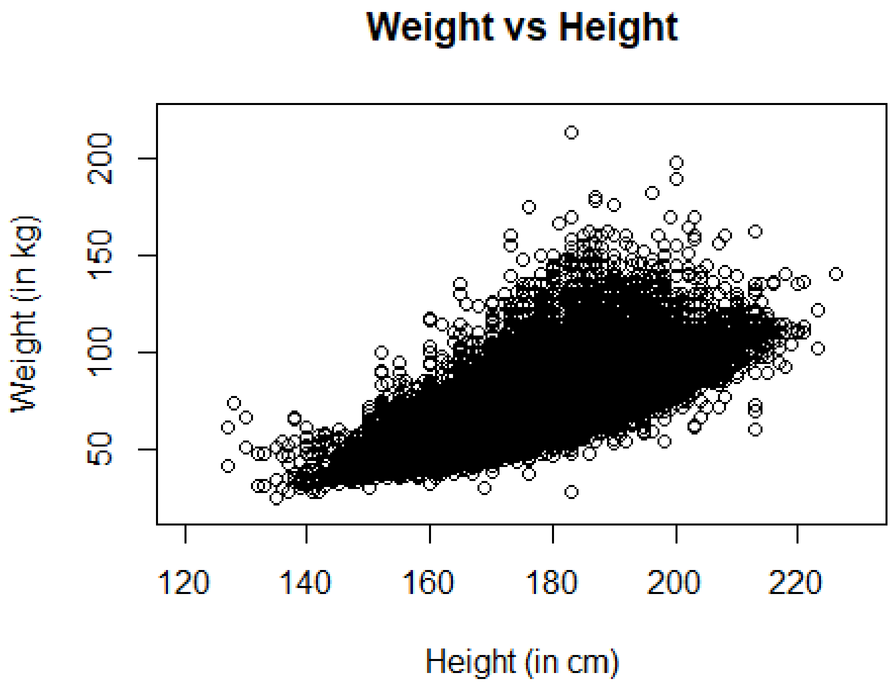
```
##   count
## 1    29
```

```
#Select athletes with the most gold medals
top_gold_medalists <- athlete_events %>%
  filter(Medal == "Gold") %>%
  group_by(Name, Sex) %>%
  summarise(Medal_Count = n()) %>%
  ungroup() %>%
  arrange(desc(Medal_Count)) %>%
  head(5)

## `summarise()` has grouped output by 'Name'. You can override using the
## `.groups` argument.

print(top_gold_medalists)

## # A tibble: 5 × 3
##   Name                                Sex   Medal_Count
##   <chr>                               <chr>       <int>
## 1 "Michael Fred Phelps, II"           M              23
## 2 "Raymond Clarence \"Ray\" Ewry"     M              10
## 3 "Frederick Carlton \"Carl\" Lewis"  M               9
## 4 "Larysa Semenivna Latynina (Diriy-)" F              9
## 5 "Mark Andrew Spitz"                 M               9

#Sport where India had the highest medals
top_sport_india <- athlete_events %>%
  filter(Team == 'India') %>%
  group_by(Sport) %>%
  summarize(count = n()) %>%
  arrange(desc(count)) %>%
  slice(1)
print(top_sport_india)

## # A tibble: 1 × 2
##   Sport  count
##   <chr>  <int>
## 1 Hockey   214
```

Statistical Analysis and Results:

Returning to the research questions previously stated, the ratio of integer/numeric type data compared to character type data set a limitation as to what could be explored. Thus, one of the main objectives was to assess the correlation between the physical attributes of all the Olympic athletes, namely weights and heights.

```
print(plot(x = athlete_events$Height, y = athlete_events$Weight,
     xlab = "Height (in cm)",
     ylab = "Weight (in kg)",
     xlim = c(120, 230),
     ylim = c(20, 220),
     main = "Weight vs Height"
))
```



```
## NULL
```

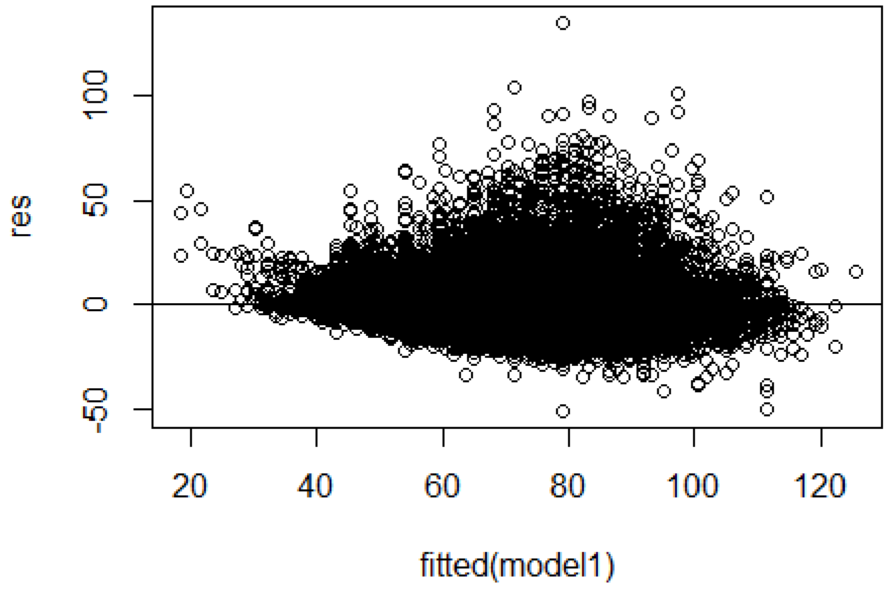
As seen in the scatterplot, while a smaller random sample of the athlete_events dataset would've provided a cleaner outlook, the scatterplot placed the heights of each athlete (measured in cm) against the weights (measured in kg). While the graph indicates a positive liner correlation between the two variables, the accuracy of the dataset might prove to be a limitation. For instance, the scatterplot contains a point in which there is a height of approximately 135 with a weight of around 25kg, which translates to ~50 lbs. Although the anomalies in the dataset weren't necessary deleted, the Partial Least Squares Regression model used to predict weights accounted for these anomalies.

While the scatterplot's correlation briefly explained the relationship between athlete weights and heights, further methods were utilized:

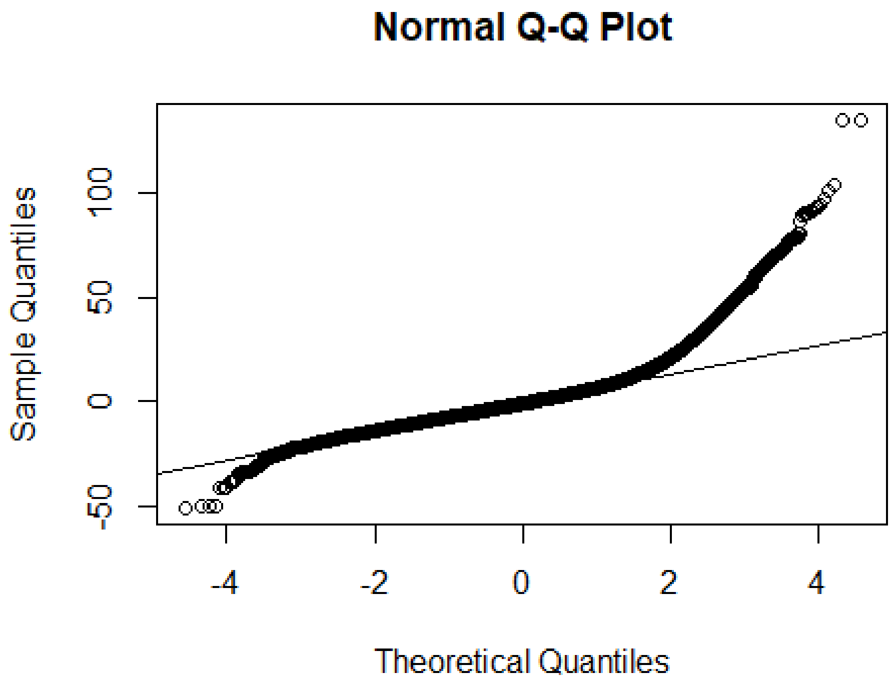
```
model1 = lm(Weight ~ Height, data = athlete_events)
print(summary(model1))

##
## Call:
## lm(formula = Weight ~ Height, data = athlete_events)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -50.951  -5.285  -0.868   3.884 135.049
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.193e+02  3.181e-01  -375.0   <2e-16 ***
## Height       1.083e+00  1.811e-03   598.3   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.669 on 206150 degrees of freedom
## Multiple R-squared:  0.6346, Adjusted R-squared:  0.6346
## F-statistic: 3.58e+05 on 1 and 206150 DF,  p-value: < 2.2e-16
```

Using height as the explanatory variable and weight as the response variable, the simple linear regression model was created to demonstrate how athlete weights were expected to change based on the input height. Observing the output of the call, there were several noticeable issues with the model: 1. The estimates of the Intercept and Height variables provided a model equation of Weight = -119.3 + 1.083*Height. Therefore, considering instances such as the anomaly in the scatterplot, relatively low heights would translate to absurdly low weight calculations. 2. Due to the simplicity of the model, the explanatory variable was instantly deemed as most significant at the .001 level of significance. Therefore, implementing more variables should show a more realistic relationship.

```
res <- resid(model1)
plot(fitted(model1), res)
abline(0,0)
```



```
qqnorm(res)
qqline(res)
```



Continuing to assess the validity of the model, the residuals were plotted against a horizontal line at 0. While an ideal plot would have balance above and below the line, this particular plot showed the majority of the residuals above the line. Also, as seen in the qqplot, the values towards the ends of the curve tended to move away from the qqline, indicating a lack of normality. As previously stated, the methodology to improving these errors was the inclusion of more variables in the model.

```
model2 <- lm(Weight ~ Age+Height+Sex, data = athlete_events)
print(summary(model2))

## 
## Call:
## lm(formula = Weight ~ Age + Height + Sex, data = athlete_events)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -51.307  -4.914  -0.821   3.635 135.378 
## 
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)    
## (Intercept) -1.075e+02  3.423e-01 -313.98   <2e-16 ***
## Age          2.282e-01  3.405e-03   67.02   <2e-16 ***
## Height       9.655e-01  2.007e-03  480.99   <2e-16 ***
## SexM         4.617e+00  4.538e-02  101.73   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 8.342 on 206148 degrees of freedom
## Multiple R-squared:  0.6616, Adjusted R-squared:  0.6616
## F-statistic: 1.343e+05 on 3 and 206148 DF,  p-value: < 2.2e-16
```

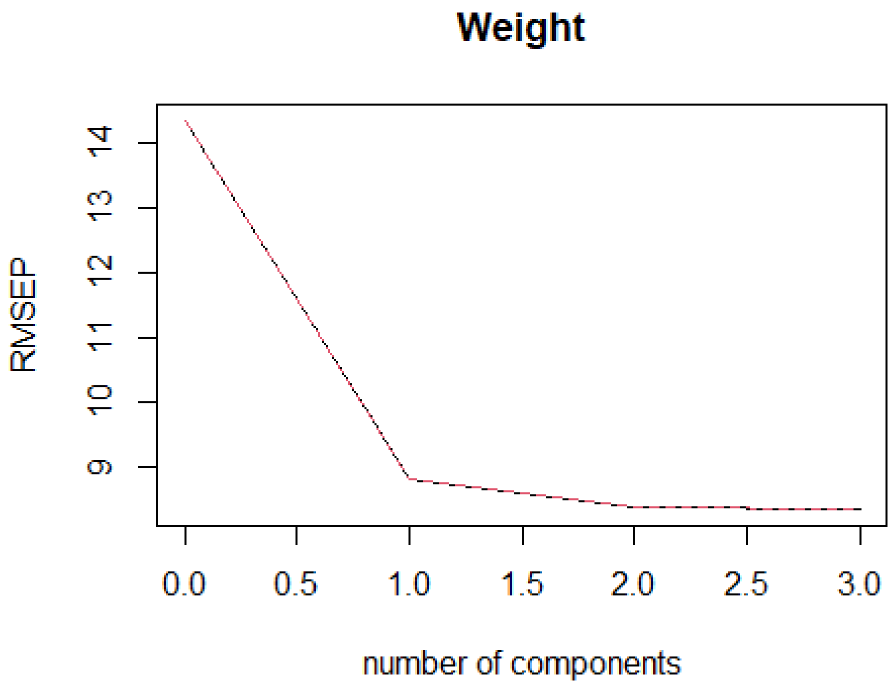
Comparing the two models, the second model showed that despite including variables such as age and sex, there was still equal significance in relation to the weight output. Also, the initial model had a Root Mean Standard Error of 8.669 and an R2 value of .6346. However, the newer linear regression model's respective values were 8.342 and .6616. Therefore, based on the decrease of RMSE and the increase of R2, there is evidence that it is a better model.

As previously stated, there were several anomalies within the data that couldn't necessarily be deleted. To account for these "outliers", a Partial Least Squares Regression model was created:

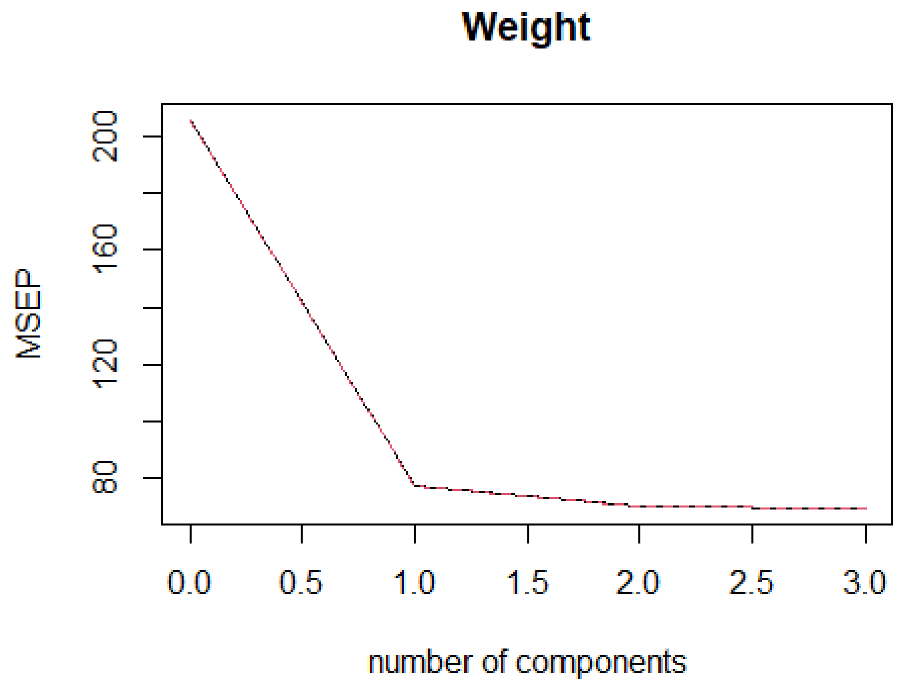
```
library(pls)

## Warning: package 'pls' was built under R version 4.3.1

##
## Attaching package: 'pls'

## The following object is masked from 'package:stats':
##
##     loadings

set.seed(1)
model <- plsr(Weight~Age+Height+Sex, data=athlete_events, scale=TRUE,
validation="CV")
print(summary(model))

## Data:    X dimension: 206152 3
##  Y dimension: 206152 1
## Fit method: kernelpls
## Number of components considered: 3
##
## VALIDATION: RMSEP
## Cross-validated using 10 random segments.
##        (Intercept)  1 comps  2 comps  3 comps
## CV           14.34    8.805    8.363    8.343
## adjCV        14.34    8.805    8.363    8.343
##
## TRAINING: % variance explained
##         1 comps  2 comps  3 comps
## X         51.67    72.45   100.00
## Weight    62.30    65.99    66.16
## NULL
```

Compared to the second linear regression models, there were some acknowledgeable similarities and observations: 1. The RMSEP and R2 values were essentially the same, or off by the .001. This could be explained by the fact that the formula in the Partial Least Squares Regression was the same as the Linear Regression model. 2. The PLS Regression model implemented a built-in K folds cross validation to evaluate the accuracy of the model. As seen in the output, as the number of explanatory variables increased from 1 to 3, the CV value decreased from 14.34 to 8.343. Thus, the validation method proved that the model was effective.

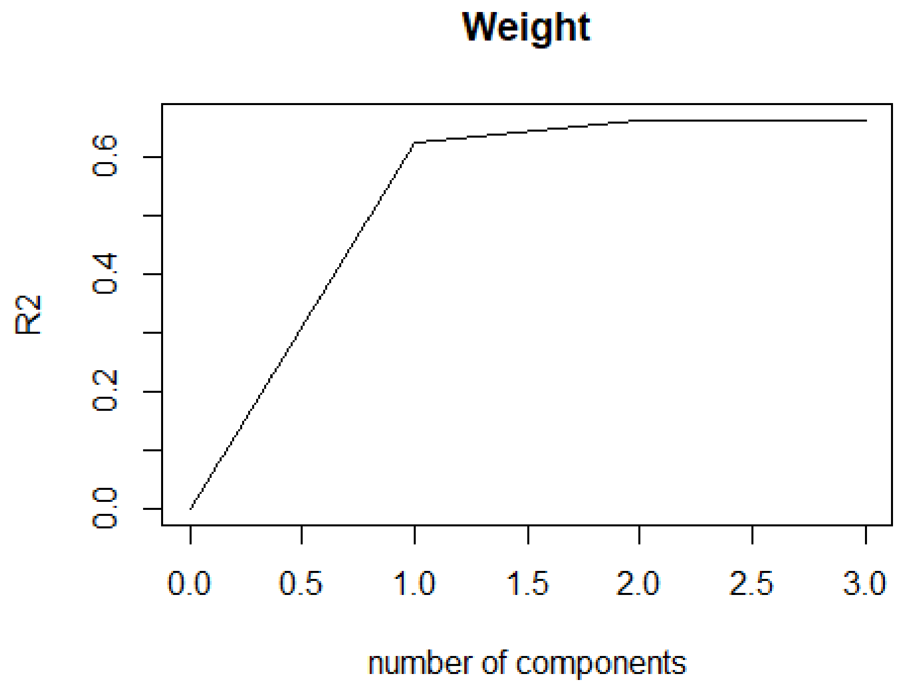
```
validationplot(model)
```



```
validationplot(model, val.type="MSEP")
```



```
validationplot(model, val.type="R2")
```



To further display the change of RMSEP and R2 as the number of explanatory variables increased, the PLS Regression model was visualized based on MSEP, RMSEP, and R2. As previously reported, the decrease in MSEP/RMSEP and increase in R2 proved that the model could be used to predict the weights of Olympic athletes.

Delving into the decision of choosing between the second Linear Regression model and the PLS Regression model, the choice ultimately came down to the RMSEP and R2 for each model. Since the outputs were basically the same, with a slight difference of .001, the models proved to be somewhat equal in terms of efficiency. Therefore, the simpler model was chosen.

With the initial statistical analysis covering the integer-type data for each competitor, a hypothesis test was also coordinated to assess the character-type data. Specifically including the male vs. female participant data from earlier as well as the Medal data to critique event performance, a research question was developed to determine whether gender had an impact on the amount of gold medalists.

As such, the null and alternative hypotheses are as follows: Null Hypothesis: There is equal likelihood between men and women to win gold medals Alternative Hypothesis: Number of male gold medalists ≠ Number of female gold medalists

```
gold_medalists <- athlete_events %>%
  filter(Medal == "Gold") %>%
  group_by(Name, Sex) %>%
  summarise(Medal_Count = n()) %>%
  ungroup() %>%
  arrange(desc(Medal_Count))

## `summarise()` has grouped output by 'Name'. You can override using the
## `.groups` argument.

male_gold <- gold_medalists %>% filter(Sex == "M")
female_gold <- gold_medalists %>% filter(Sex == "F")
print(t.test(male_gold$Medal_Count, female_gold$Medal_Count, alternative =
"two.sided", conf.level = .99))

##
##  Welch Two Sample t-test
##
## data:  male_gold$Medal_Count and female_gold$Medal_Count
## t = -7.0089, df = 4391.8, p-value = 2.768e-12
## alternative hypothesis: true difference in means is not equal to 0
## 99 percent confidence interval:
##  -0.19192553 -0.08873527
## sample estimates:
## mean of x mean of y
##  1.268854  1.409184
```

Visualized in the code and output, the means amount of gold medals for male and female gold medalists were used to conduct a t-test. With a mean of 1.269 for male gold medalists and a mean of 1.409 for female gold medalists, the output stated that we were 99% confident that the difference between the two means was within the range of ~-.192 and -.089. Using the p-value to determine the decision regarding the null hypothesis, the calculated p-value of 2.768e-12 was compared to an alpha level of .01. Since the p- value was less than .01, the null hypothesis was rejected and it can be assumed that the factor of sex is statistically significant in related to number of awarded gold medals.

Regarding the reasoning behind the higher mean number of gold medals for female gold medalists, two things were considered: 1. The earlier participation line graph showed that women weren't allowed in the games until 1900. Therefore, it is possible that they were more so included in event where winning a gold medal was more attainable compared to their male counterparts. 2. Expanding on the previous point, there could be events that favor the physical attributes of typical female competitors.

Conclusion : While heights can be an integral factor for predicting the weights of Olympic athletes, it isn't the sole reason behind a particular output. For instance,a female weightlifting competitor can be the same height as a male gymnast, but different weights. Therefore, by considering other explanatory variables such as Age, Sport, and Sex, a better understanding of weight trends can be collected. When comparing the refined Linear Regression Model to the PLS Regression, both provided a similar fit regarding weight prediction, ultimately leading the Linear Regression Model to be chosen.In addition, when assessing the Olympic event performance based on Sex, there is a slightly unequal distribution of gold medalists between men and women, which can be based on the specific sport.In addition, women didn't begin competing in the Olympics until 1900, which could reduce the amount of potential gold medals collected.