Limpieza y analisis datos College Basketball

Javier Cañón Álvarez

7 de Enero de 2020

Contents

Descripción del Dataset **1**

Integración y selección de los datos **1**

Limpieza de los datos **3**
Valores nulos o vacíos . . . 3
Valores extremos . . . 4

ANÁLISIS DE LOS DATOS **4**
Selección de grupos de datos . . . 4
Comprobación de la normalidad . . . 5
Aplicación de pruebas estadísticas . . . 8
Correlación . . . 9

Representación de resultados **9**

Resolución del problema **14**

Código **14**

Contribuciones **14**

Descripción del Dataset

Este Dataset esta formado por un conjunto de estadísticas de la liga universitaria de Estados Unidos. Por equipo y año se especifican datos como partidos ganados, perdidos, puntos, rebotes, etc.

La pregunta que nos planteamos es si existe una gran diferencia entre las estadísticas de los equipos ganadores y los que no.

Integración y selección de los datos

En este punto integraremos los datos existentes separados por años en un solo conjunto. Luego haremos una selección si procede de las variables que más nos interesen.

Procedemos a cargar los ficheros en formato CSV que tiene una separación de campos mediante coma(,):

```
# utilizando la función read.csv
data15 <- read.csv("cbb15.csv")
data16 <- read.csv("cbb16.csv")
data17 <- read.csv("cbb17.csv")
data18 <- read.csv("cbb18.csv")
data19 <- read.csv("cbb19.csv")

# mostramos la cabecera y estructuctura de uno de los años para comprobar que está bien
head(data17)
```

```
##             TEAM CONF  G  W ADJOE ADJDE BARTHAG EFG_O EFG_D  TOR TORD  ORB
## 1        Gonzaga  WCC 39 37 117.8  86.3  0.9728  56.6  41.1 16.2 17.1 30.0
## 2 North Carolina  ACC 39 33 121.0  91.5  0.9615  51.7  48.1 16.2 18.6 41.3
## 3      Villanova   BE 36 32 122.2  92.5  0.9611  57.5  48.1 17.1 20.1 30.2
## 4         Kansas  B12 36 31 121.5  94.5  0.9472  56.1  48.1 17.6 18.6 34.1
## 5       Kentucky  SEC 38 32 118.3  91.3  0.9517  52.9  47.5 15.7 19.2 33.5
## 6     Louisville  ACC 34 25 117.6  91.5  0.9469  51.4  45.7 16.0 19.6 36.7
##    DRB  FTR FTRD X2P_O X2P_D X3P_O X3P_D ADJ_T  WAB POSTSEASON SEED
## 1 26.2 39.0 26.9  56.3  40.0  38.2  29.0  71.5  7.7        2ND    1
## 2 25.0 34.3 31.6  51.0  46.3  35.5  33.9  72.8  8.4  Champions    1
## 3 27.8 35.0 22.1  59.2  49.1  36.9  31.1  65.6 11.1        R32    1
## 4 29.7 36.0 30.0  53.6  45.3  40.4  35.6  71.4 11.0         E8    1
## 5 27.7 40.9 33.5  52.9  48.3  35.3  30.6  73.7  9.0         E8    2
## 6 28.5 34.0 38.8  50.5  44.8  35.5  31.6  69.4  6.9        R32    2
```

```r
str(data17)
```

```
## 'data.frame':    351 obs. of  23 variables:
##  $ TEAM      : Factor w/ 351 levels "Abilene Christian",..: 102 197 328 132 136 150 72 11 221 20 ...
##  $ CONF      : Factor w/ 32 levels "A10","ACC","AE",..: 32 2 8 7 27 2 2 23 23 7 ...
##  $ G         : int  39 39 36 36 38 34 37 37 38 34 ...
##  $ W         : int  37 33 32 31 32 25 28 32 33 27 ...
##  $ ADJOE     : num  118 121 122 122 118 ...
##  $ ADJDE     : num  86.3 91.5 92.5 94.5 91.3 91.5 95.6 95.6 93.8 93.9 ...
##  $ BARTHAG   : num  0.973 0.962 0.961 0.947 0.952 ...
##  $ EFG_O     : num  56.6 51.7 57.5 56.1 52.9 51.4 54.8 53.7 55.5 52.5 ...
##  $ EFG_D     : num  41.1 48.1 48.1 48.1 47.5 45.7 47.4 47.7 46.4 46.1 ...
##  $ TOR       : num  16.2 16.2 17.1 17.6 15.7 16 16.3 16.6 17.1 20.6 ...
##  $ TORD      : num  17.1 18.6 20.1 18.6 19.2 19.6 17.3 17.4 19.3 17.1 ...
##  $ ORB       : num  30 41.3 30.2 34.1 33.5 36.7 31.7 33.2 32.4 39.8 ...
##  $ DRB       : num  26.2 25 27.8 29.7 27.7 28.5 29.8 26 29.5 29.4 ...
##  $ FTR       : num  39 34.3 35 36 40.9 34 39.3 40.5 34.4 33.9 ...
##  $ FTRD      : num  26.9 31.6 22.1 30 33.5 38.8 31.2 28.6 26.3 30.1 ...
##  $ X2P_O     : num  56.3 51 59.2 53.6 52.9 50.5 53.5 51.5 54.5 52.2 ...
##  $ X2P_D     : num  40 46.3 49.1 45.3 48.3 44.8 48.9 48.4 46.1 45 ...
##  $ X3P_O     : num  38.2 35.5 36.9 40.4 35.3 35.5 37.9 39 38 35.3 ...
##  $ X3P_D     : num  29 33.9 31.1 35.6 30.6 31.6 29.3 31 31.2 32.2 ...
##  $ ADJ_T     : num  71.5 72.8 65.6 71.4 73.7 69.4 69.7 67.1 68.4 65.1 ...
##  $ WAB       : num  7.7 8.4 11.1 11 9 6.9 8.6 7.9 6 7.9 ...
##  $ POSTSEASON: Factor w/ 8 levels "2ND","Champions",..: 1 2 5 3 3 5 5 8 4 8 ...
##  $ SEED      : int  1 1 1 1 2 2 2 2 3 3 ...
```

Para proceder a la integración añadimos una columna a cada dataset con el año correspondiente y luego juntamos todos los años.

```r
# añadimos columna con el año
data15["YEAR"] <- 2015
data16["YEAR"] <- 2016
data17["YEAR"] <- 2017
data18["YEAR"] <- 2018
data19["YEAR"] <- 2019
# juntamos todos los años
data_tot <- rbind(data15, data16, data17, data18, data19)
```

```
attach(data_tot)
```

Ahora efectuamos la selección únicamente de las variables que nos interesan.

```
#seleccionamos las variables que nos interesan
data_tot <- select(data_tot,TEAM,G,W,EFG_O,TOR,TORD,ORB,DRB,FTR,X2P_O,X3P_O,POSTSEASON,YEAR)
tail(data_tot)
```

```
##                         TEAM  G  W EFG_O  TOR TORD  ORB  DRB  FTR X2P_O
## 1752 Mississippi Valley St. 31  6  41.9 18.5 17.5 28.4 31.7 33.7  40.1
## 1753             Alcorn St. 27 10  45.7 24.1 18.2 30.1 31.5 30.5  45.0
## 1754          New Hampshire 27  5  44.0 18.4 16.9 21.5 24.7 21.9  39.4
## 1755            Chicago St. 30  3  44.2 22.5 16.7 22.1 33.9 33.1  43.5
## 1756           Delaware St. 29  6  40.0 19.0 18.9 27.8 31.6 25.5  37.7
## 1757 Maryland Eastern Shore 30  7  43.5 20.7 19.0 22.8 31.7 28.3  44.5
##      X3P_O POSTSEASON YEAR
## 1752  31.0       <NA> 2019
## 1753  31.3       <NA> 2019
## 1754  32.6       <NA> 2019
## 1755  30.7       <NA> 2019
## 1756  29.0       <NA> 2019
## 1757  27.9       <NA> 2019
```

Limpieza de los datos

Valores nulos o vacíos

Se examinan las diferentes columnas para comprobar si existen valores nulos o elementos vacíos.

```
na_count <-sapply(data_tot, function(y) sum(length(which(is.na(y)))))
na_count <- data.frame(na_count)
na_count
```

```
##            na_count
## TEAM              0
## G                 0
## W                 0
## EFG_O             0
## TOR               0
## TORD              0
## ORB               0
## DRB               0
## FTR               0
## X2P_O             0
## X3P_O             0
## POSTSEASON     1417
## YEAR              0
```

Como podemos ver, solo nos aparecen elementos nulos en las columnas de POSTSEASON y SEED. Éstas nos indican la ronda donde un equipo fue eliminado en la elemininatoria final del campeonato. En caso de que sean nulos, sifnifica que no llegaron a las eliminatorias finales.

Sustituimos los NA por NOT PLAYED.

```
data_tot$POSTSEASON <- as.character(data_tot$POSTSEASON)
data_tot[is.na(POSTSEASON),"POSTSEASON"] <- "NOT PLAYED"
```

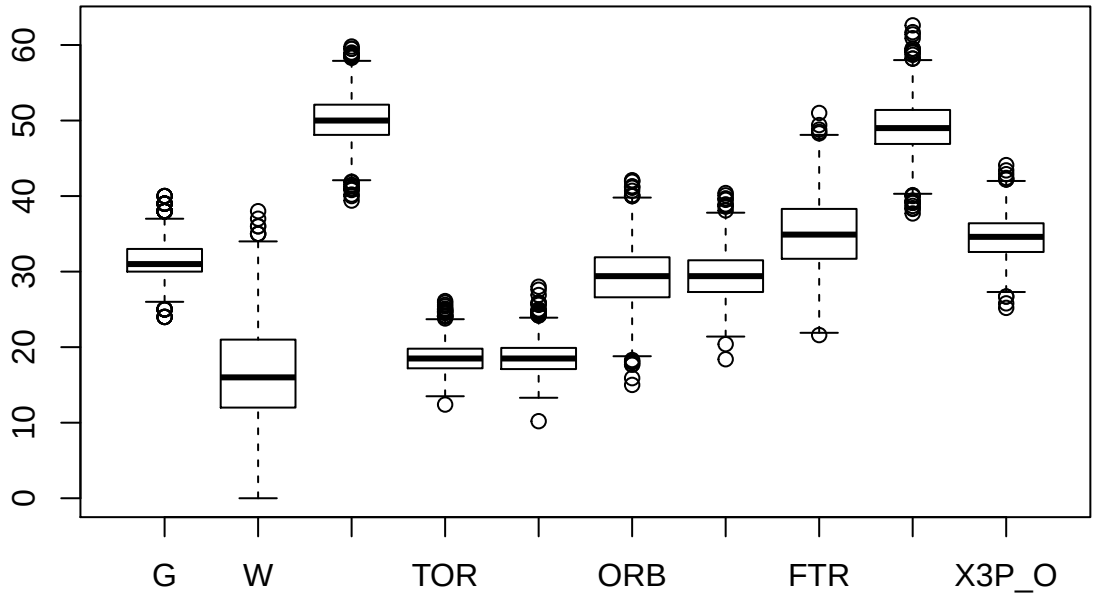
```
data_tot$POSTSEASON <- as.factor(data_tot$POSTSEASON)
levels(data_tot$POSTSEASON)
```

```
## [1] "2ND"         "Champions"   "E8"          "F4"          "NOT PLAYED"
## [6] "R32"         "R64"         "R68"         "S16"
```

Valores extremos

Ahora mediante diagramas de cajas veremos a ver si existen valores extremos que nos puedan llevar a errores posteriormente.

```
boxplot(select(data_tot,-TEAM,-POSTSEASON,-YEAR))
```



A la vista de los valores obtenidos, no veo que los marcados en las gráficas de caja como valores extremos, así lo sean. Por tanto queda descartado que sean errorres del dataset y por tanto son perfectamente utilizables.

ANÁLISIS DE LOS DATOS

Selección de grupos de datos

Para ver si los equipos más ganadores han tenido las mejores estadísticas, vamos a dividir el dataset en dos grupos: los que han llegado a las rondas finales y los que no.

```
#añadimos una columna que diferencie entre finalistas y no finalistas
data_tot["CATEGORIA"] <- "cat"
data_tot[data_tot$POSTSEASON == "NOT PLAYED", "CATEGORIA"] <- "NO FINALISTA"
data_tot[data_tot$POSTSEASON != "NOT PLAYED", "CATEGORIA"] <- "FINALISTA"
```

```r
#eliminamos la variable POSTSEASON
data_tot$CATEGORIA <- as.factor(data_tot$CATEGORIA)
data_tot <- select(data_tot, -POSTSEASON)
head(data_tot)
```

```
##         TEAM  G  W EFG_O  TOR TORD  ORB  DRB  FTR X2P_O X3P_O YEAR
## 1  Wisconsin 40 36  54.8 12.4 15.8 32.1 23.7 36.2  54.8  36.5 2015
## 2       Duke 39 35  56.6 16.3 18.6 35.8 30.2 39.8  55.9  38.7 2015
## 3    Arizona 38 34  53.5 16.5 20.6 34.5 22.4 47.1  53.3  36.0 2015
## 4    Gonzaga 37 34  57.9 16.1 17.1 33.9 28.0 38.7  57.0  40.0 2015
## 5 Louisville 36 27  47.7 17.2 21.3 34.7 30.8 38.7  48.4  30.7 2015
## 6 Notre Dame 38 32  58.3 14.5 17.3 27.9 32.2 36.7  58.2  39.0 2015
##   CATEGORIA
## 1 FINALISTA
## 2 FINALISTA
## 3 FINALISTA
## 4 FINALISTA
## 5 FINALISTA
## 6 FINALISTA
```

De esta forma hemos diferenciado entre los equipos finalistas y no finalistas durante los últimos 5 años.

Comprobación de la normalidad

Para la comprobación de la normalidad utilizaremos el test de Shapiro-Wilk. Asumiremos como hipótesis nula que la población está distribuida normalmente.

```r
shapiro.test(data_tot$W)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  data_tot$W
## W = 0.99058, p-value = 3.108e-09
```

```r
shapiro.test(data_tot$EFG_O)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  data_tot$EFG_O
## W = 0.99889, p-value = 0.3463
```

```r
shapiro.test(data_tot$TOR)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  data_tot$TOR
## W = 0.99373, p-value = 8.759e-07
```

```r
shapiro.test(data_tot$TORD)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  data_tot$TORD
```

```
## W = 0.99203, p-value = 3.534e-08
```

```
shapiro.test(data_tot$ORB)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  data_tot$ORB
## W = 0.99906, p-value = 0.5128
```

```
shapiro.test(data_tot$DRB)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  data_tot$DRB
## W = 0.99842, p-value = 0.09896
```

```
shapiro.test(data_tot$FTR)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  data_tot$FTR
## W = 0.99809, p-value = 0.03811
```

```
shapiro.test(data_tot$X2P_O)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  data_tot$X2P_O
## W = 0.99733, p-value = 0.004458
```

```
shapiro.test(data_tot$X3P_O)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  data_tot$X3P_O
## W = 0.99943, p-value = 0.9012
```

Observando los p-valores se puede decir que solo 4 de las 9 variables siguen una distribución normal. No obstante, por el teorema del límite central, y como las muestras son de gran tamaño (+30 elementos), podemos considerar que toda variable sigue una distribución normal de media 0 y desviación 1.

También podemos comprobar la heterocedasticidad mediante el test de Levene.

```
leveneTest(data_tot$W ~ data_tot$CATEGORIA)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##         Df F value    Pr(>F)
## group    1  30.496 3.846e-08 ***
##       1755
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
leveneTest(data_tot$EFG_O ~ data_tot$CATEGORIA)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
```

```
##         Df F value Pr(>F)
## group    1  1.7508 0.1859
##       1755
```

```
leveneTest(data_tot$TOR ~ data_tot$CATEGORIA)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##         Df F value  Pr(>F)
## group    1  4.9663 0.02597 *
##       1755
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
leveneTest(data_tot$TORD ~ data_tot$CATEGORIA)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##         Df F value Pr(>F)
## group    1  2.2481  0.134
##       1755
```

```
leveneTest(data_tot$ORB ~ data_tot$CATEGORIA)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##         Df F value Pr(>F)
## group    1  1.7855 0.1817
##       1755
```

```
leveneTest(data_tot$DRB ~ data_tot$CATEGORIA)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##         Df F value Pr(>F)
## group    1  0.6793 0.4099
##       1755
```

```
leveneTest(data_tot$FTR ~ data_tot$CATEGORIA)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##         Df F value Pr(>F)
## group    1  0.1991 0.6555
##       1755
```

```
leveneTest(data_tot$X2P_O ~ data_tot$CATEGORIA)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##         Df F value  Pr(>F)
## group    1  2.8471 0.09172 .
##       1755
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
leveneTest(data_tot$X3P_O ~ data_tot$CATEGORIA)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##         Df F value Pr(>F)
## group    1  1.2911  0.256
##       1755
```

Viendo los p-valores, la mayoría son mayores que 0.05 por lo que podemos afirmar igualdad de varianzas entre los grupos comparados: Finalistas y no finalistas.

Aplicación de pruebas estadísticas

Contraste de hipótesis

En este primer estudio vamos a comprobar si existe diferencia entre las estadísticas registradas de los equipos "Finalistas" y los "no finalistas". Para ello vamos a realizar un contraste de hipótesis con algunas variables del dataset.

Planteamos la hipótesis nula y la alternativa:

- H0: stat(equipos ganadores) = stat(equipos perdedores)
- H1: stat(equipos ganadores) > stat(equipos perdedores)

donde "stat" será: EFG_O (Porcentaje de acierto en tiros de campo), FTR (Tiros libres), X3P_O (Porcentaje de acierto en triples)

```
#aplicando la prueba t-Student para las 3 variables de estudio
t.test(EFG_O ~ CATEGORIA, alternative = "greater", conf.level = 0.95, var.equal = TRUE, data = data_tot
```

```
##
##  Two Sample t-test
##
## data:  EFG_O by CATEGORIA
## t = 16.56, df = 1755, p-value < 2.2e-16
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  2.622753      Inf
## sample estimates:
##    mean in group FINALISTA mean in group NO FINALISTA
##                   52.46912                   49.55695
```

```
t.test(FTR ~ CATEGORIA, alternative = "greater", conf.level = 0.95, var.equal = TRUE, data = data_tot)
```

```
##
##  Two Sample t-test
##
## data:  FTR by CATEGORIA
## t = 3.694, df = 1755, p-value = 0.0001138
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  0.6020241       Inf
## sample estimates:
##    mean in group FINALISTA mean in group NO FINALISTA
##                   35.97353                   34.88779
```

```
t.test(X3P_O ~ CATEGORIA, alternative = "greater", conf.level = 0.95, var.equal = TRUE, data = data_tot
```

```
##
##  Two Sample t-test
##
## data:  X3P_O by CATEGORIA
## t = 12.226, df = 1755, p-value < 2.2e-16
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  1.682469      Inf
## sample estimates:
##    mean in group FINALISTA mean in group NO FINALISTA
##                   36.13147                   34.18730
```

Como podemos ver, los p-valores obtenidos son menores al nivel de significación fijado y por tanto se rechaza la hipótesis nula.

Como era de esperar, los equipos finalistas tienen mejores porcentajes de tiro en general, tiros libres y tiros de 2 puntos.

Correlación

Mediante correlación vamos identificar si hay variables claves que influyan en ganar partidos.

Analizaremos la correlación de partidos ganados con: rebotes ofensivos y porcentaje de tiros de 2.

```
#con cor.test podemos ver la relación entre pares de variables
cor.test(data_tot$W, data_tot$ORB)
```

```
##
##  Pearson's product-moment correlation
##
## data:  data_tot$W and data_tot$ORB
## t = 13.001, df = 1755, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.2531394 0.3384682
## sample estimates:
##       cor
## 0.2963952
```

```
cor.test(data_tot$W, data_tot$X2P_O)
```

```
##
##  Pearson's product-moment correlation
##
## data:  data_tot$W and data_tot$X2P_O
## t = 30.281, df = 1755, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.5542243 0.6157033
## sample estimates:
##       cor
## 0.5858059
```

Podemos ver como influye más el porcentaje de acierto en tiros de 2 que los rebotes ofensivos cuando se gana un partido.

Representación de resultados

Mediante un diagrama de barras podemos representar las estadísticas diferenciando entre equipos Finalistas y No Finalistas.

Primeramente creamos los datos medios.

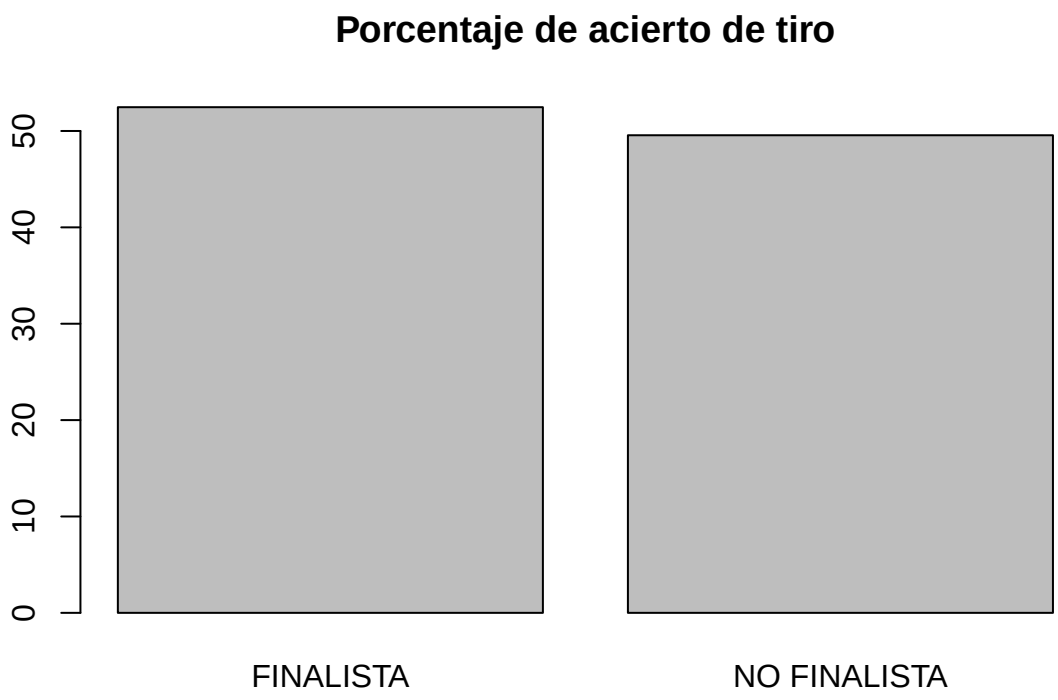
```
mean_stats <- data_tot %>%
  group_by(CATEGORIA) %>%
  summarise(EFG_O_mean = mean(EFG_O),
            DRB_mean = mean(DRB),
            FTR_mean = mean(FTR),
            X2P_O_mean = mean(X2P_O),
            X3P_O_mean = mean(X3P_O))
```

```
#mostramos tabla
mean_stats
```

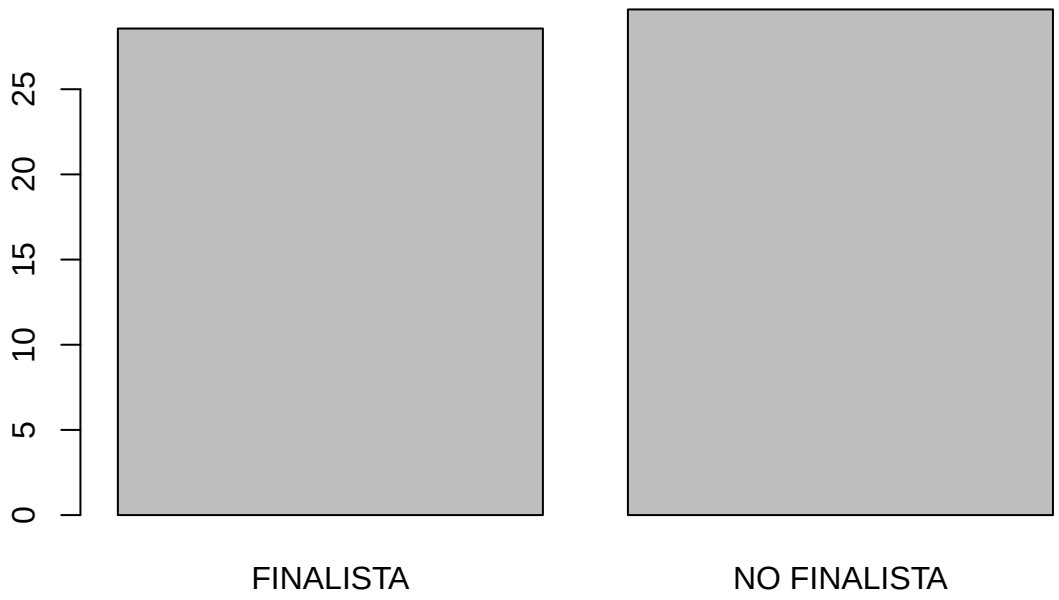
```
## # A tibble: 2 x 6
##   CATEGORIA    EFG_O_mean DRB_mean FTR_mean X2P_O_mean X3P_O_mean
##   <fct>             <dbl>    <dbl>    <dbl>      <dbl>      <dbl>
## 1 FINALISTA          52.5     28.6     36.0       51.5       36.1
## 2 NO FINALISTA       49.6     29.7     34.9       48.6       34.2
```

Y ahora representamos.

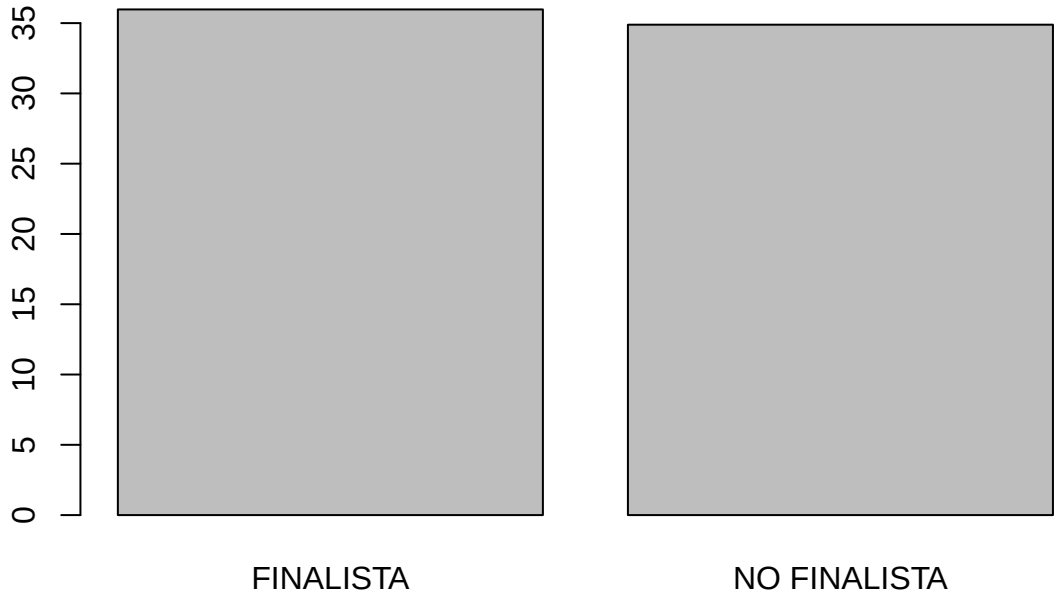
```
barplot(mean_stats$EFG_O_mean, names=mean_stats$CATEGORIA, main = "Porcentaje de acierto de tiro")
```



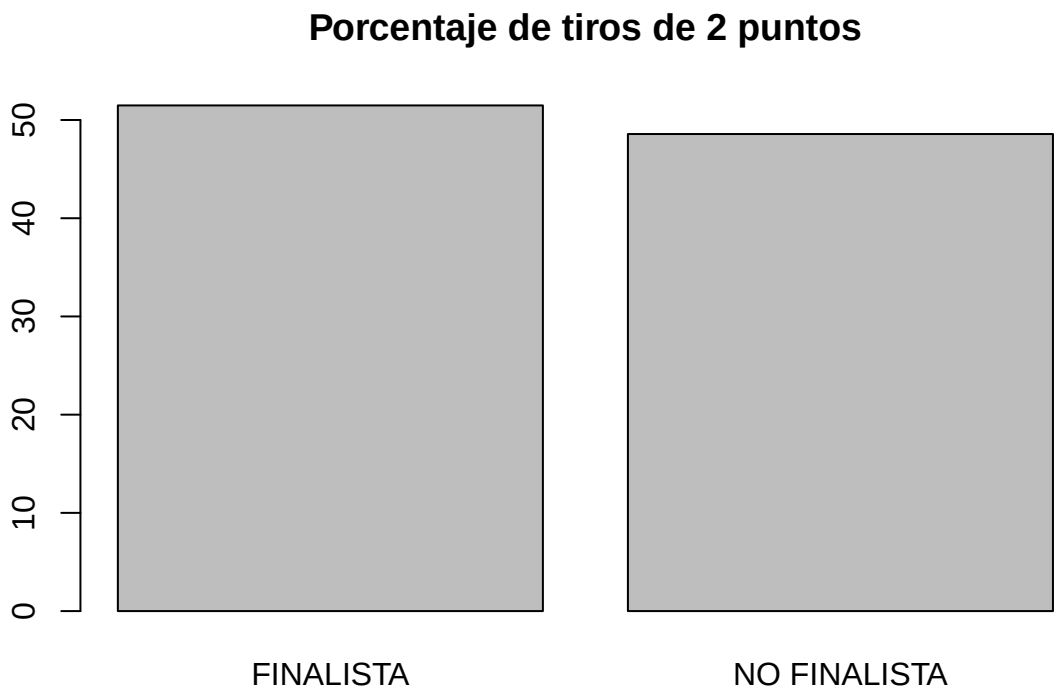

```
barplot(mean_stats$DRB_mean, names=mean_stats$CATEGORIA, main = "Rebotes defensivos")
```



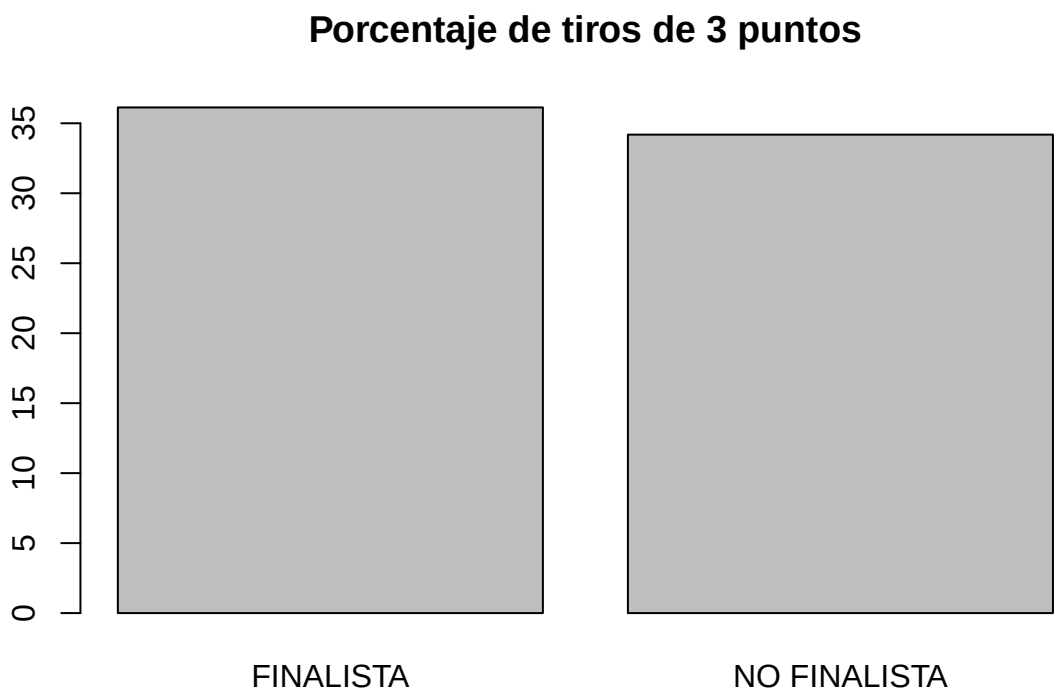

```
barplot(mean_stats$FTR_mean, names=mean_stats$CATEGORIA, main = "Porcentaje de acierto de tiros libres")
```




```r
barplot(mean_stats$X2P_O_mean, names=mean_stats$CATEGORIA, main = "Porcentaje de tiros de 2 puntos")
```




```
barplot(mean_stats$X3P_O_mean, names=mean_stats$CATEGORIA, main = "Porcentaje de tiros de 3 puntos")
```



Vemos como las estadísticas normalmente favorecen a los equipos finalistas que es lo que estábamos buscando.

Resolución del problema

Tras los análisis hechos hemos llegado a una conclusión que apriori era bastante lógica. Los equipos que han llegado a las rondas finales del campeonato de la liga universitaria de EEUU tienen mejores estadísticas de tiro, rebotes, porcentajes de acierto, etc.

En el baloncesto actual las estadísticas y el BIG DATA de las estadísticas han adquirido una importancia vital en la política de fichajes de los diferentes equipos. Análisis como este (más exhaustivos) se llevan a cabo para poder mejorar dentro de los propios equipos.

Código

```
write.csv(data_tot ,"C:\\Users\\JAVIER PC\\Documents\\cbb_final.csv", row.names = FALSE)
```

Contribuciones

- Investigación previa: Javier Cañón Álvarez
- Redacción de las respuestas: Javier Cañón Álvarez
- Desarrollo de código: Javier Cañón Álvarez