Modelos Actuariales II - Examen 1

Oscar Andrei Zempoalteca Ramírez 164889

Para realizar el siguiente análisis deberás usar la base de datos: Base2.P1.csv. La base de datos contiene la información de 202 deportistas de alto rendimimiento de diferentes disciplinas. La base de datos está constituida por 7 variables:

- Sex : Sexo.
- Sport: Deporte que practica.
- Ht: Altura, medida en centímetros.
- Wt: Peso, medido en kilogramos.
- LBM: Índice de masa corporal magra.
- BMI: Índice de masa corporal.
- PBF: Porcentaje de grasa corporal.

Importemos la base de datos que nos piden:

```
library(readr)
```

```
## Warning: package 'readr' was built under R version 4.0.3
```

```
Base2_P1 <- read_delim("C:/Users/HP/Downloads/Base2.P1.csv",
    ";", escape_double = FALSE, locale = locale(decimal_mark = ","),
    trim_ws = TRUE)
```

```
## Warning: Missing column names filled in: 'X1' [1]
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   X1 = col_double(),
##   Sex = col_character(),
##   Sport = col_character(),
##   Ht = col_double(),
##   Wt = col_double(),
##   LBM = col_double(),
##   BMI = col_double(),
##   PBF = col_double()
## )
```

```
atleta<-Base2_P1
summary(atleta)
```

```
##        X1             Sex                Sport                  Ht
##  Min.   :  1.00   Length:202         Length:202         Min.   :148.9
##  1st Qu.: 51.25   Class :character   Class :character   1st Qu.:174.0
##  Median :101.50   Mode  :character   Mode  :character   Median :179.7
##  Mean   :101.50                                         Mean   :180.1
##  3rd Qu.:151.75                                         3rd Qu.:186.2
##  Max.   :202.00                                         Max.   :209.4
##        Wt              LBM              BMI              PBF
##  Min.   : 37.80   Min.   : 34.36   Min.   :16.75   Min.   : 5.630
##  1st Qu.: 66.53   1st Qu.: 54.67   1st Qu.:21.08   1st Qu.: 8.545
##  Median : 74.40   Median : 63.03   Median :22.72   Median :11.650
##  Mean   : 75.01   Mean   : 64.87   Mean   :22.96   Mean   :13.507
##  3rd Qu.: 84.12   3rd Qu.: 74.75   3rd Qu.:24.46   3rd Qu.:18.080
##  Max.   :123.20   Max.   :106.00   Max.   :34.42   Max.   :35.520
```

```
head(atleta)
```

```
## # A tibble: 6 x 8
##      X1 Sex   Sport    Ht    Wt   LBM   BMI   PBF
##   <dbl> <chr> <chr> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1     1 F     BBall  196.  78.9  63.3  20.6  19.8
## 2     2 F     BBall  190.  74.4  58.6  20.7  21.3
## 3     3 F     BBall  178.  69.1  55.4  21.9  19.9
## 4     4 F     BBall  185   74.9  57.2  21.9  23.7
## 5     5 F     BBall  185.  64.6  53.2  19.0  17.6
## 6     6 F     BBall  174   63.7  53.8  21.0  15.6
```

1. Analiza cada una de las variables de la base de datos para identificar el tipo de variable que es y el soporte de la variable (los valores que puede tomar).

Tipo de variable Sex:

```
class(atleta$Sex)
```

```
## [1] "character"
```

con soporte:

```
levels(factor(atleta$Sex))
```

```
## [1] "F" "M"
```

Tipo de variable Sport:

```
class(atleta$Sport)
```

```
## [1] "character"
```

Con soporte:

```
levels(factor(atleta$Sport))
```

```
##  [1] "BBall"   "Field"   "Gym"     "Netball" "Rowing"  "Swim"    "T400m"
##  [8] "Tennis"  "TSprnt"  "WPolo"
```

Tipo de varible Ht:

```
class(atleta$Ht)
```

```
## [1] "numeric"
```

Soporte continuo en el intervalo:

```
rango<-c(min(atleta$Ht),max(atleta$Ht))
rango
```

```
## [1] 148.9 209.4
```

Tipo de varible Wt:

```
class(atleta$Wt)
```

```
## [1] "numeric"
```

Soporte continuo en el intervalo:

```
rango1<-c(min(atleta$Wt),max(atleta$Wt))
rango1
```

```
## [1]  37.8 123.2
```

Tipo de varible LBM:

```
class(atleta$LBM)
```

```
## [1] "numeric"
```

Soporte continuo en el intervalo:

```
rango2<-c(min(atleta$LBM),max(atleta$LBM))
rango2
```

```
## [1]  34.36 106.00
```

Tipo de varible BMI:

```
class(atleta$BMI)
```

```
## [1] "numeric"
```

Soporte continuo en el intervalo:

```
rango3<-c(min(atleta$BMI),max(atleta$BMI))
rango3
```

```
## [1] 16.75 34.42
```

Tipo de varible PBF:

```
class(atleta$PBF)
```

```
## [1] "numeric"
```

Soporte continuo en el intervalo:

```
rango4<-c(min(atleta$PBF),max(atleta$PBF))
rango4
```

```
## [1]  5.63 35.52
```

2. Determina si existen valores atípicos en cada una de las variables.

En las variables categóricas no encontramos valores atípicos al observar los niveles que puede tomar.
Importemos los paquetes necesarios:

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.0.4
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(gridExtra)
```

```
## Warning: package 'gridExtra' was built under R version 4.0.4
```

```
##
## Attaching package: 'gridExtra'
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.0.3
```

```
library(moments)
```

```
## Warning: package 'moments' was built under R version 4.0.3
```

```
library(GGally)
```

```
## Warning: package 'GGally' was built under R version 4.0.4
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

```
library(infotheo)
```

```
## Warning: package 'infotheo' was built under R version 4.0.3
```

```
library(akima)
```

```
## Warning: package 'akima' was built under R version 4.0.4
```

```
library(ggpubr)
```

```
## Warning: package 'ggpubr' was built under R version 4.0.4
```

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.0.3
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## v tibble  3.0.5     v stringr 1.4.0
## v tidyr   1.1.2     v forcats 0.5.1
## v purrr   0.3.4
```

```
## Warning: package 'tibble' was built under R version 4.0.3
```

```
## Warning: package 'tidyr' was built under R version 4.0.3
```

```
## Warning: package 'purrr' was built under R version 4.0.3
```

```
## Warning: package 'stringr' was built under R version 4.0.3
```

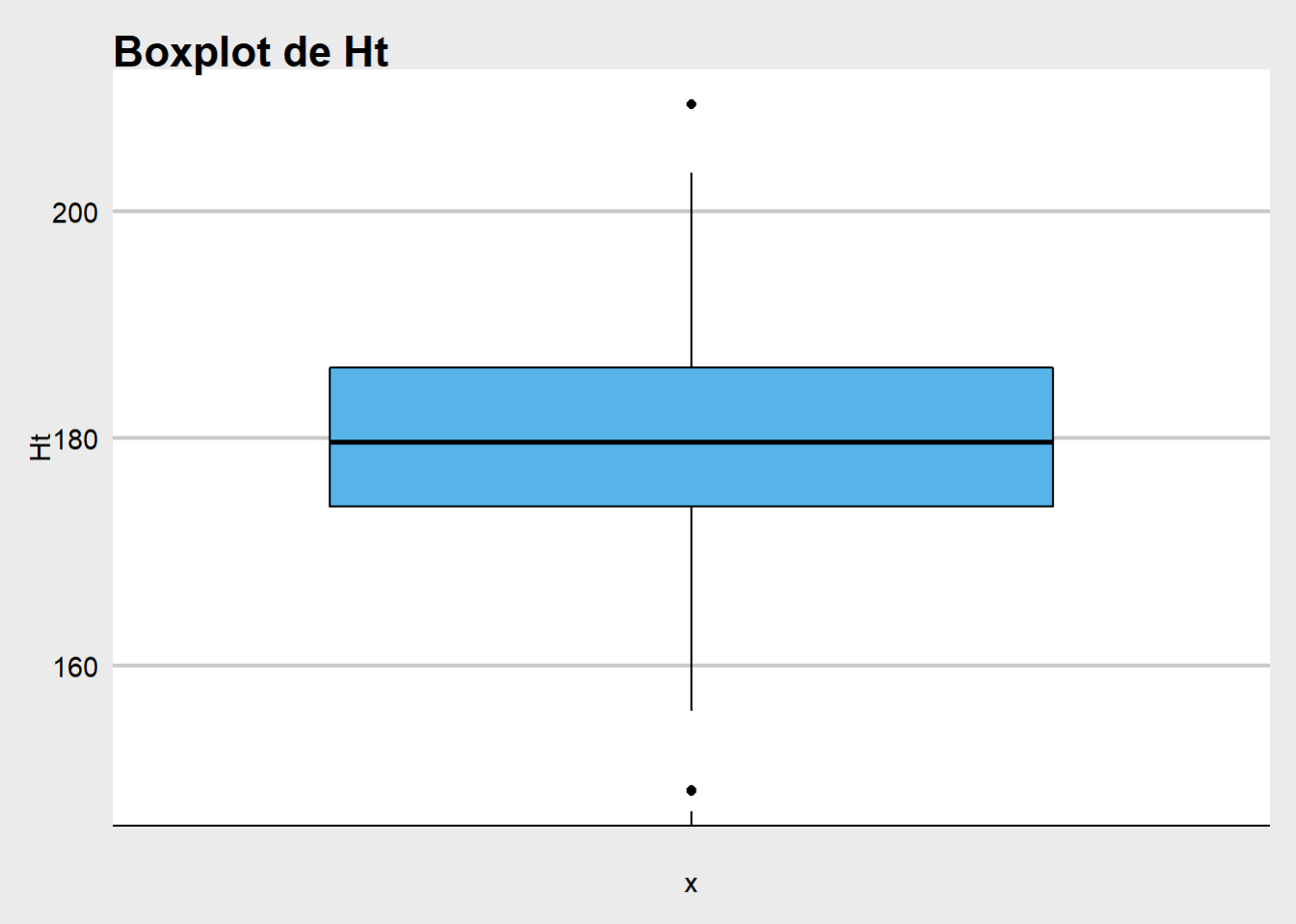
```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x gridExtra::combine() masks dplyr::combine()
## x dplyr::filter()      masks stats::filter()
## x dplyr::lag()         masks stats::lag()
```

```
library(ggthemes)
```

```
## Warning: package 'ggthemes' was built under R version 4.0.4
```

Para las variables numéricas usaremos gráficos de caja para así observar y considerar a los valores que están más alejados de Q3+1.5ICR Y Q1-1.5IQR como valores atípicos.:

```
ggplot(atleta,aes(x="",y=Ht))+geom_boxplot(fill="#56B4E9",color="black")+ggtitle("Boxplot de Ht"
)+theme_economist_white()
```

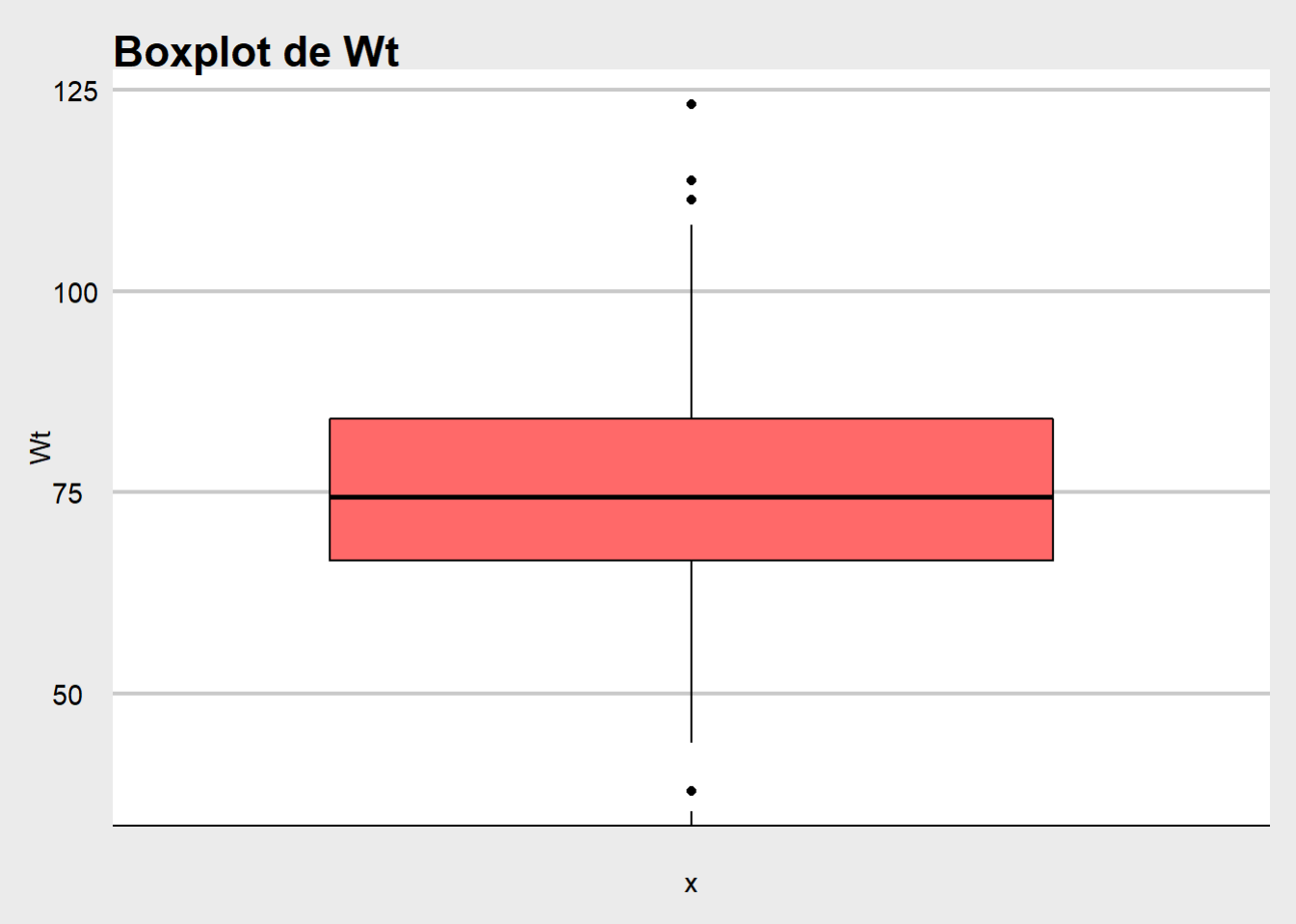


Los valores atípicos que nos muestra el boxplot para Ht son:

```
boxplot.stats(atleta$Ht)$out
```

```
## [1] 148.9 149.0 209.4
```

```
ggplot(atleta,aes(x="",y=Wt))+geom_boxplot(fill="#ff6969",color="black")+ggtitle("Boxplot de Wt"
)+theme_economist_white()
```

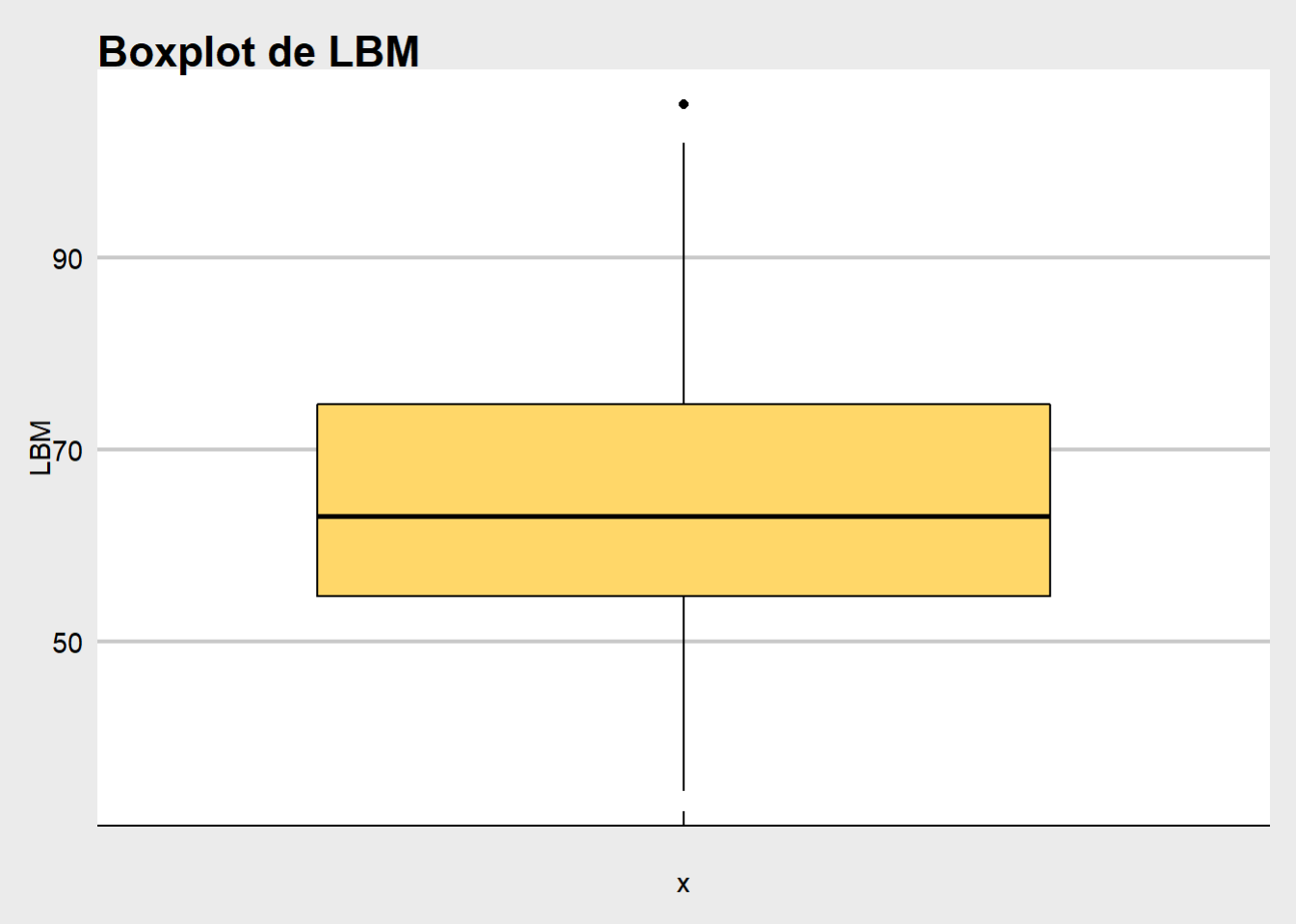


Los valores atípicos que nos muestra el boxplot para Wt son:

```
boxplot.stats(atleta$Wt)$out
```

```
## [1]  37.8 113.7 111.3 123.2
```

```
ggplot(atleta,aes(x="",y=LBM))+geom_boxplot(fill="#ffd769",color="black")+ggtitle("Boxplot de LB
M")+theme_economist_white()
```

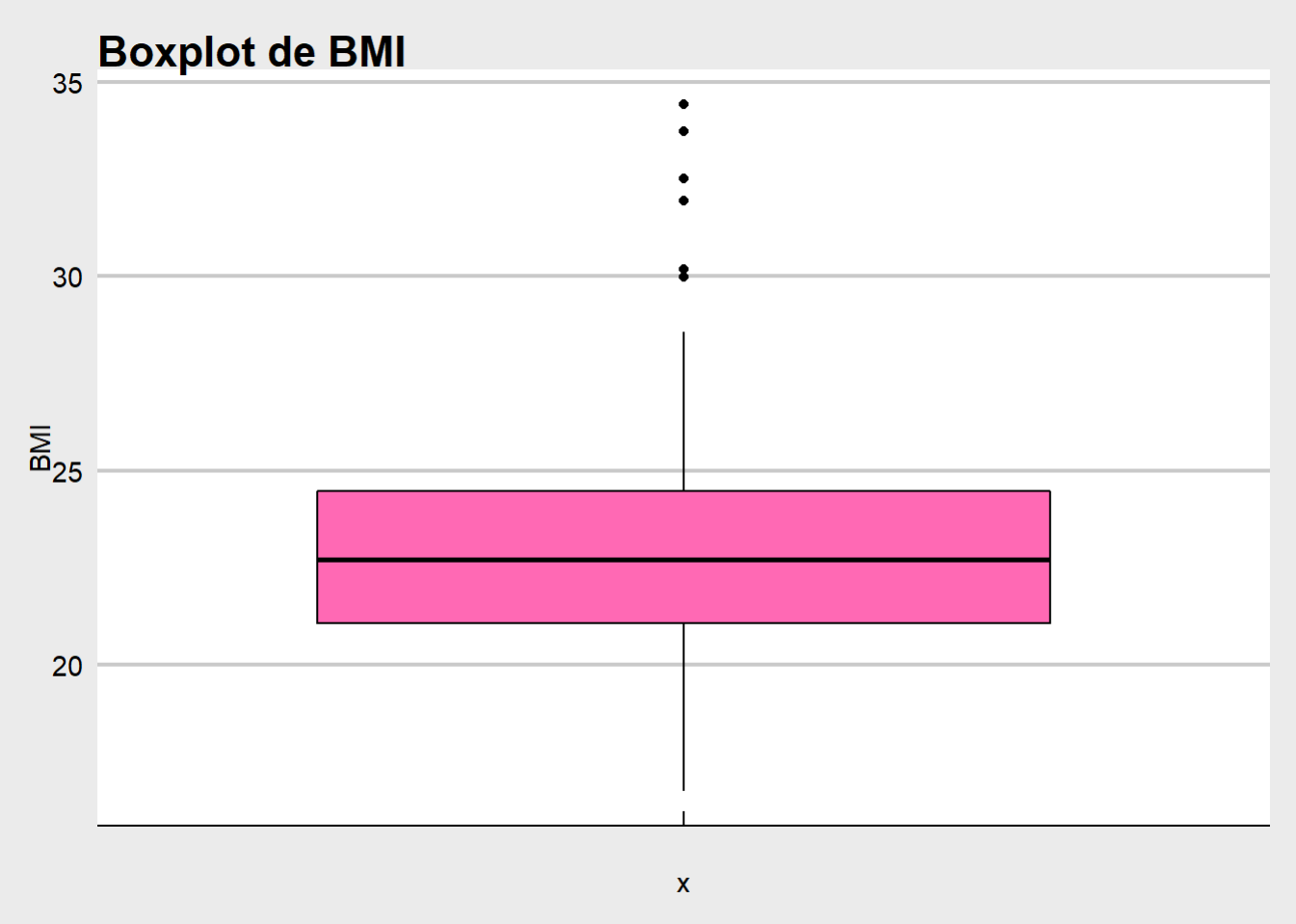


Los valores atípicos que nos muestra el boxplot para LBM son:

```
boxplot.stats(atleta$LBM)$out
```

```
## [1] 106
```

```
ggplot(atleta,aes(x="",y=BMI))+geom_boxplot(fill="#ff69b4",color="black")+ggtitle("Boxplot de BM
I")+theme_economist_white()
```

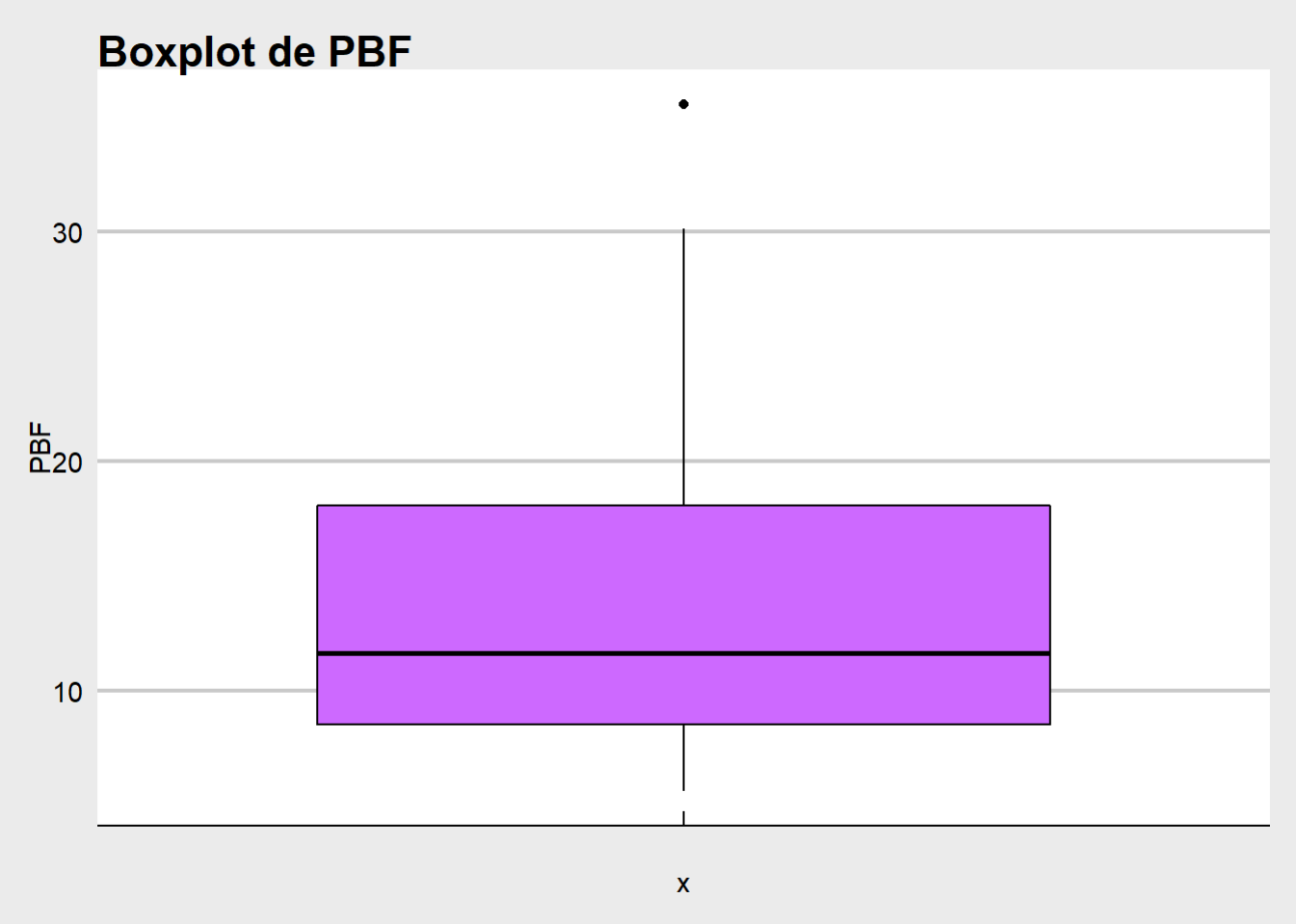


Los valores atípicos que nos muestra el boxplot para BMI son:

```
boxplot.stats(atleta$BMI)$out
```

```
## [1] 31.93 29.97 32.52 30.18 34.42 33.73 30.18
```

```
ggplot(atleta,aes(x="",y=PBF))+geom_boxplot(fill="#cd69ff",color="black")+ggtitle("Boxplot de PB
F")+theme_economist_white()
```



Los valores atípicos que nos muestra el boxplot para PBF son:

```
boxplot.stats(atleta$PBF)$out
```

```
## [1] 35.52
```

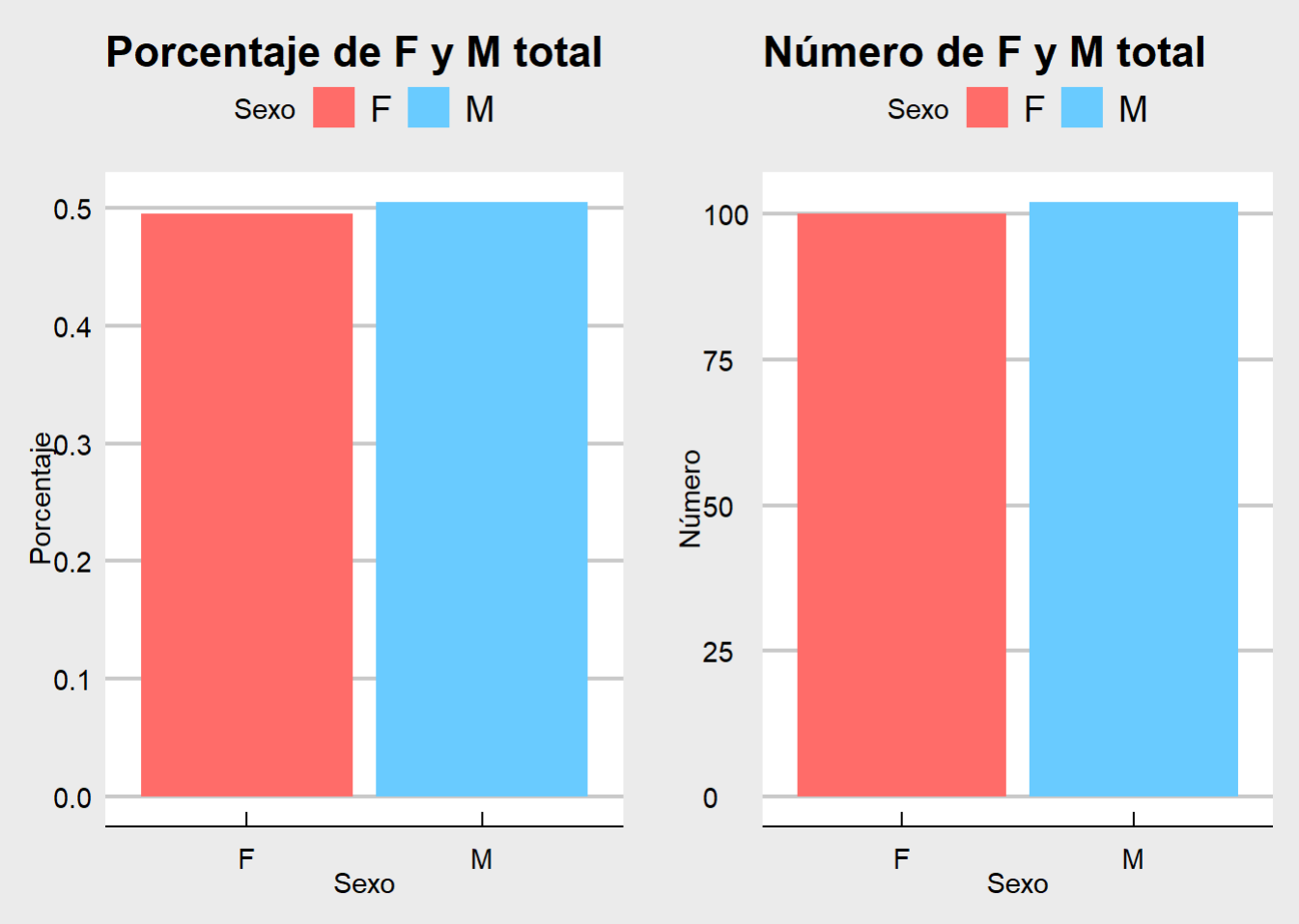
3. Analiza la tabla de contingencia de las variables Sex y Sport, ¿está balanceada la base de datos?

```
addmargins(table(atleta$Sex,atleta$Sport))
```

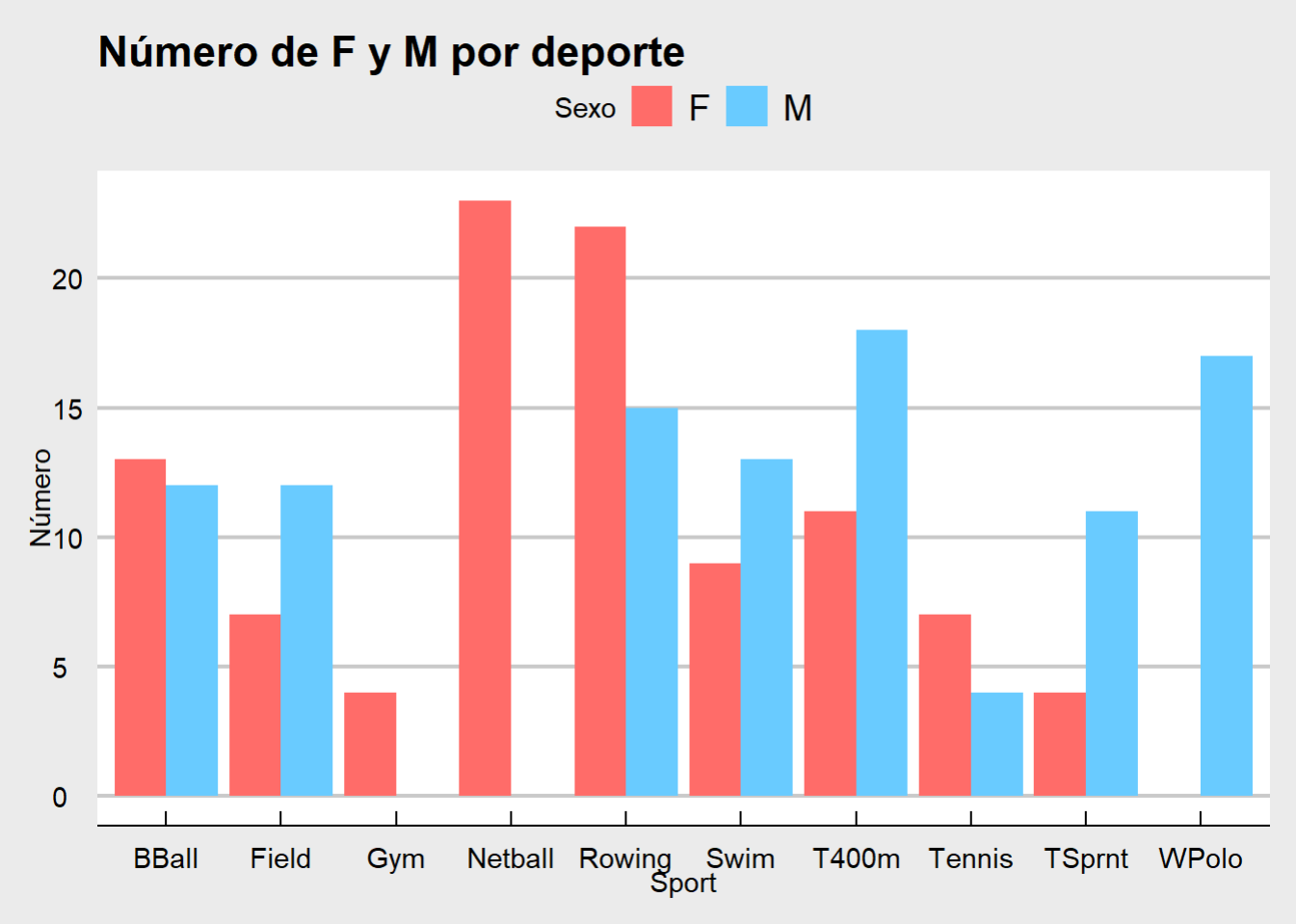
```
##
##       BBall Field Gym Netball Rowing Swim T400m Tennis TSprnt WPolo Sum
##   F      13     7   4      23     22    9    11      7      4     0 100
##   M      12    12   0       0     15   13    18      4     11    17 102
##   Sum    25    19   4      23     37   22    29     11     15    17 202
```

En el número de hombres y mujeres participantes sí está balanceada la base de datos más no en ciertos deportes donde no hubieron participantes masculinos M (Netball y Gym), donde no hubieron participantes femeninos F (WPolo) y donde el número de masculinos fue inferior al de femeninos o al revés. Veamos esto:

```
dframe<-data.frame(addmargins(table(atleta$Sex,atleta$Sport)))
dframe1<-data.frame(table(atleta$Sex,atleta$Sport))
a<-ggplot(dframe[31:32,],aes(x=Var1,y=Freq/202,fill=Var1))+geom_bar(stat="identity")+
            ggtitle("Porcentaje de F y M total")+labs(x="Sexo",y="Porcentaje",fill="Sexo")+
  scale_fill_manual(values=c("#ff6c69","#69cbff"))+theme_economist_white()
b<-ggplot(dframe[31:32,],aes(x=Var1,y=Freq,fill=Var1))+geom_bar(stat="identity")+
  ggtitle("Número de F y M total")+labs(x="Sexo",y="Número",fill="Sexo")+
  scale_fill_manual(values=c("#ff6c69","#69cbff"))+theme_economist_white()

ggarrange(a,b)
```




```
ggplot(dframe1,aes(x=Var2,y=Freq,fill=as.factor(Var1)))+
  geom_bar(stat = "identity",position= "dodge")+
  ggtitle("Número de F y M por deporte")+labs(x="Sport",y="Número",fill="Sexo")+
  scale_fill_manual(values=c("#ff6c69","#69cbff"))+theme_economist_white()
```

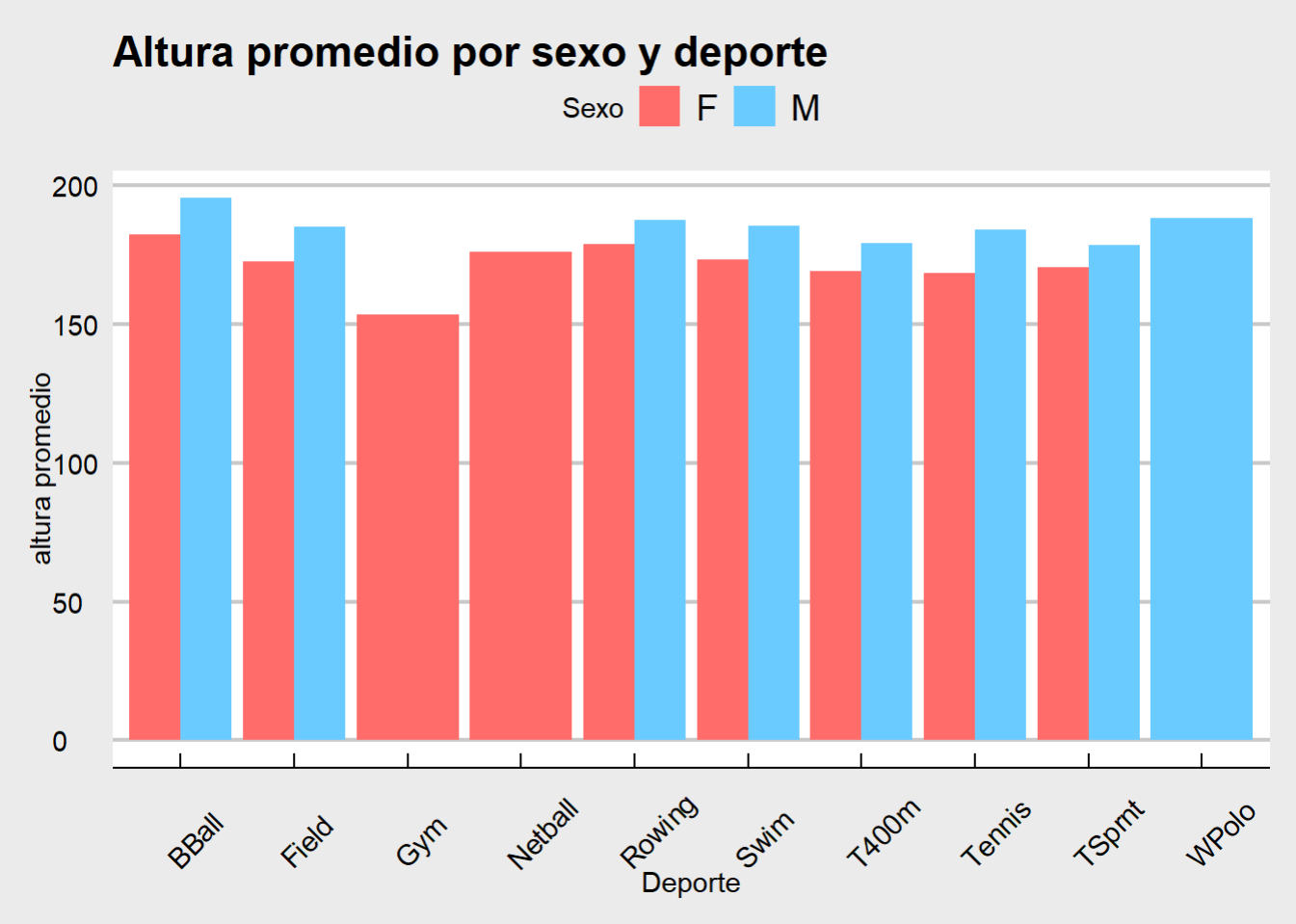


4. Estudia, gráficamente, las diferencias en las variables Ht, Wt, LBM, BMI,PBF por deporte practicado y por sexo del deportista.

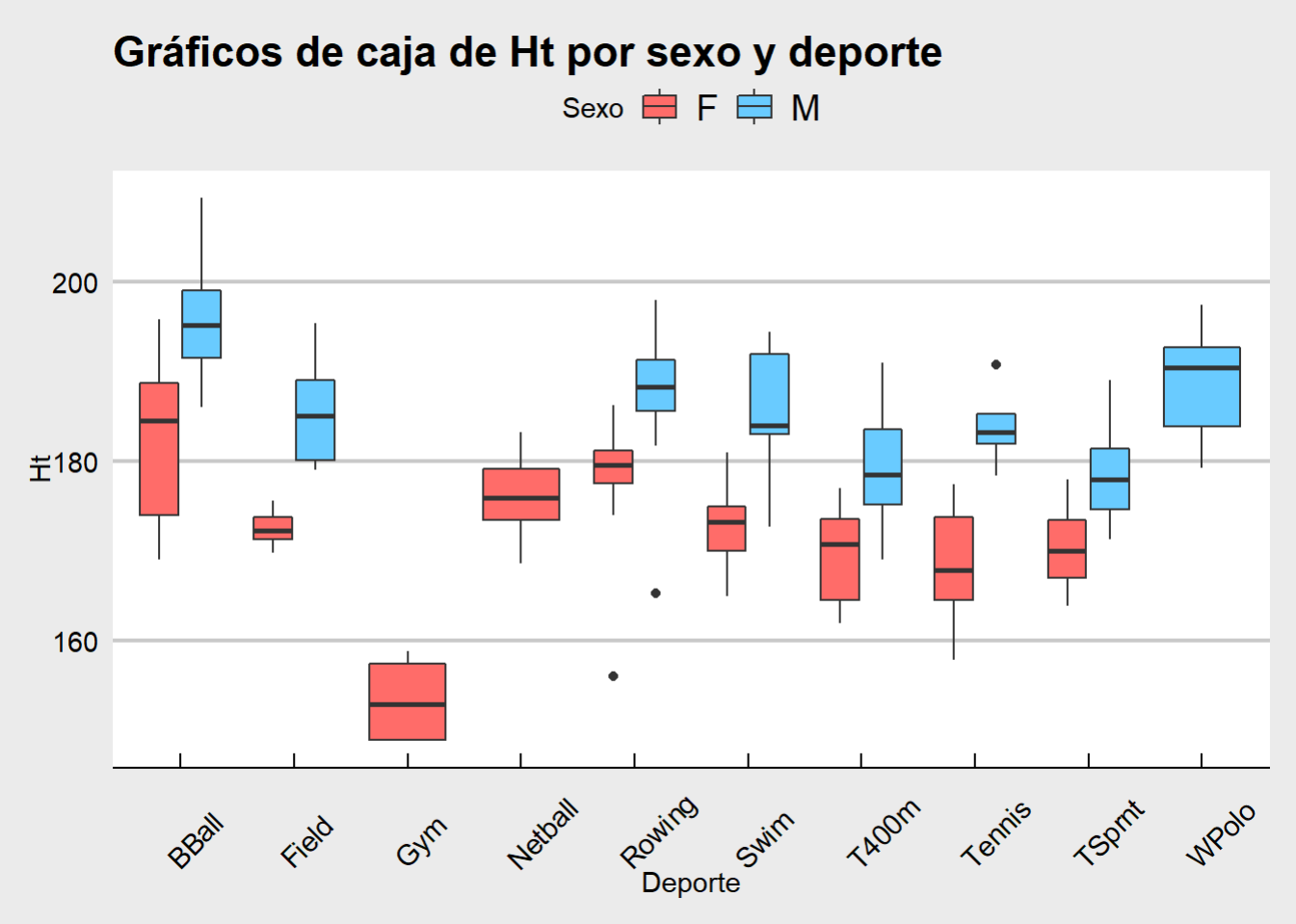
```
v<-atleta %>% group_by(Sex,Sport) %>% summarise(alturapromedio=mean(Ht))
```

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

```
ggplot(v,aes(x=Sport,y=alturapromedio,fill=as.factor(Sex)))+
  geom_bar(stat="identity",position= "dodge")+
  scale_fill_manual(values=c("#ff6c69","#69cbff"))+
  labs(x="Deporte",y="altura promedio",fill="Sexo")+
  ggtitle("Altura promedio por sexo y deporte")+theme_economist_white()+
  theme(axis.text.x = element_text(angle = 45, hjust = .05))
```



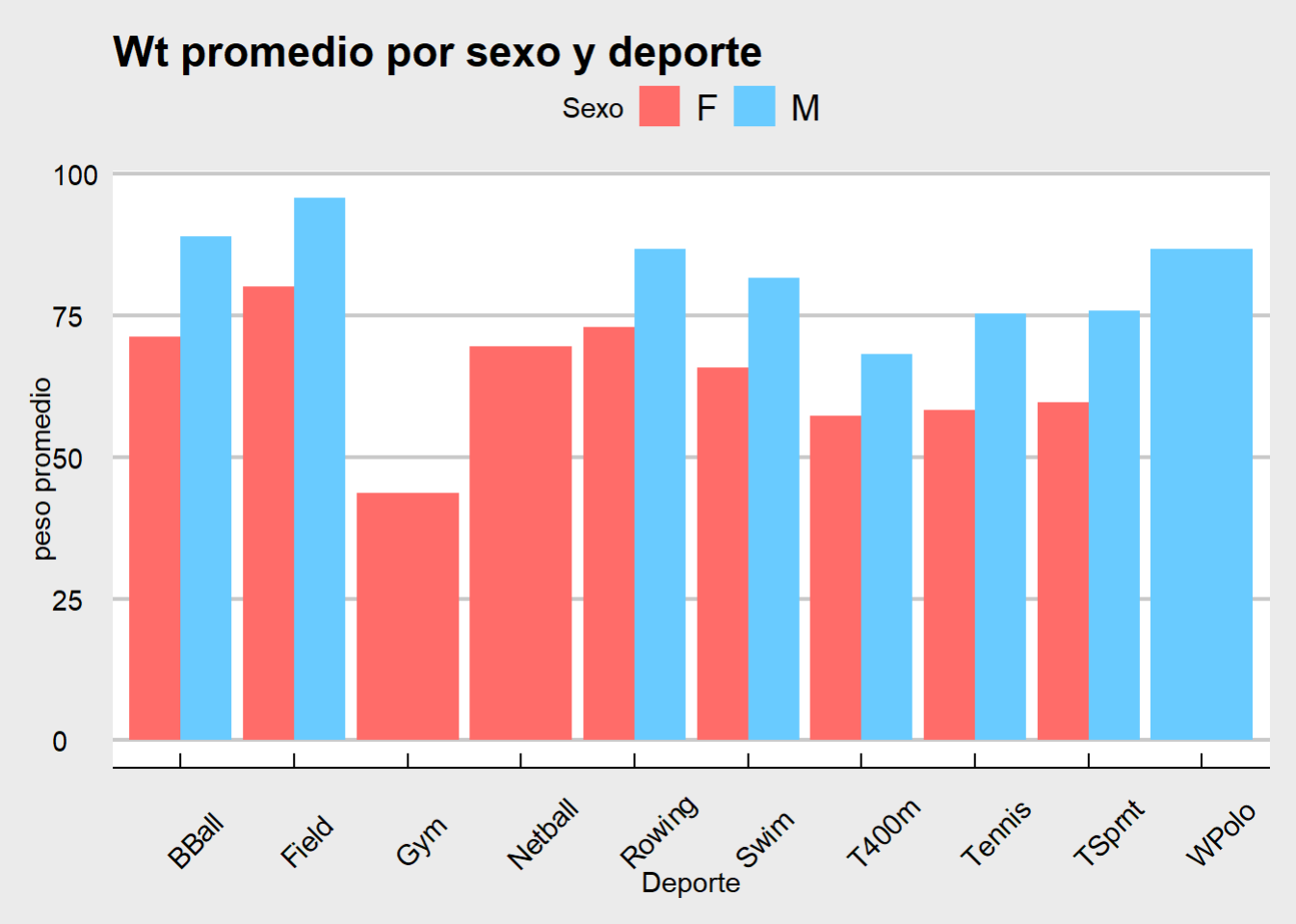

```
ggplot(atleta,aes(x=Sport,y=Ht,fill=as.factor(Sex)))+geom_boxplot()+
  scale_fill_manual(values=c("#ff6c69","#69cbff"))+labs(x="Deporte",fill="Sexo")+
  ggtitle("Gráficos de caja de Ht por sexo y deporte")+theme_economist_white()+
  theme(axis.text.x = element_text(angle = 45, hjust = .05))
```



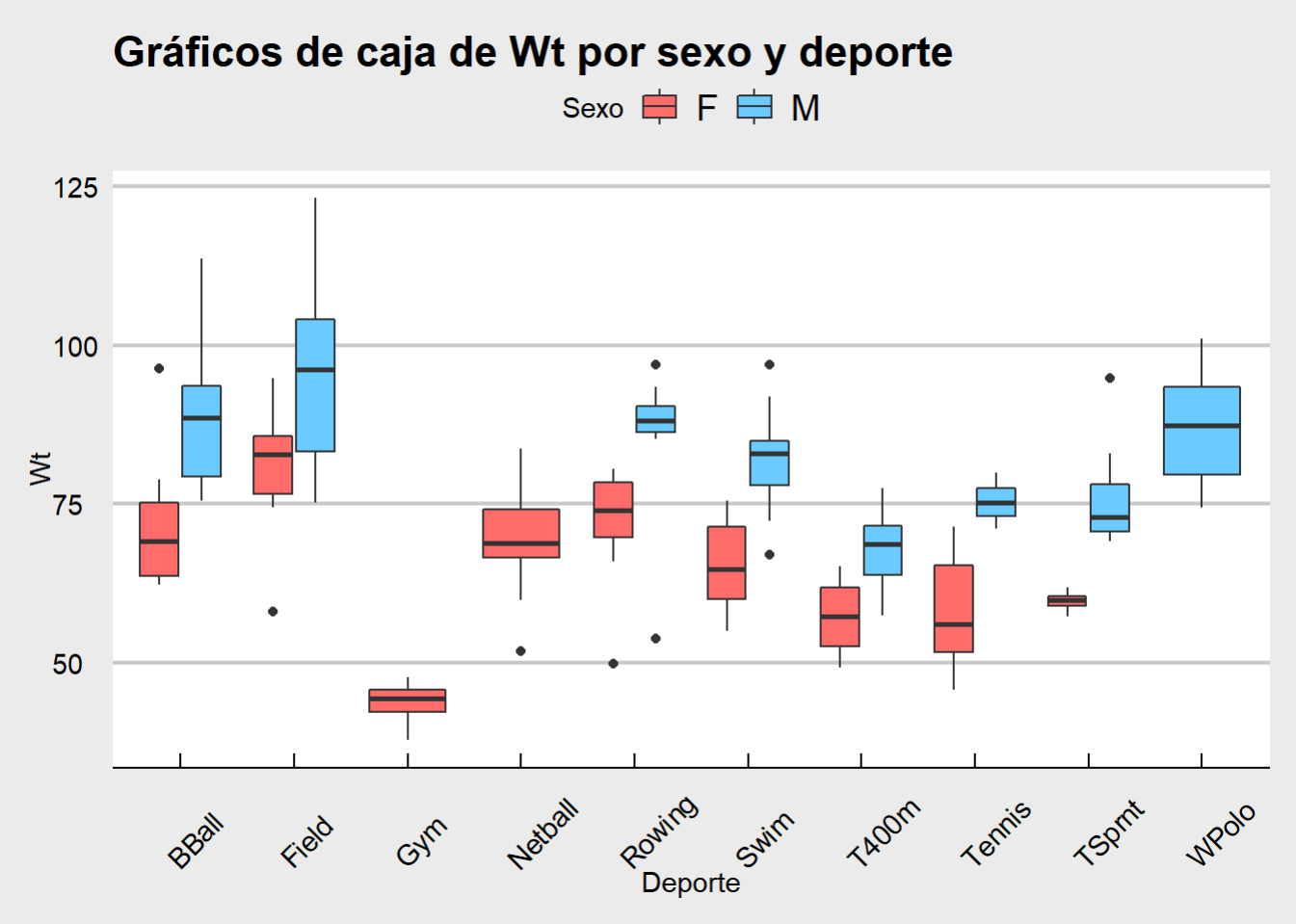

```
v1<-atleta %>% group_by(Sex,Sport) %>% summarise(pesopromedio=mean(Wt),alturapromedio=mean(Ht),
                                                 LBMpromedio=mean(LBM),BMIpromedio=mean(BMI),
                                                 PBFpromedio=mean(PBF))
```

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

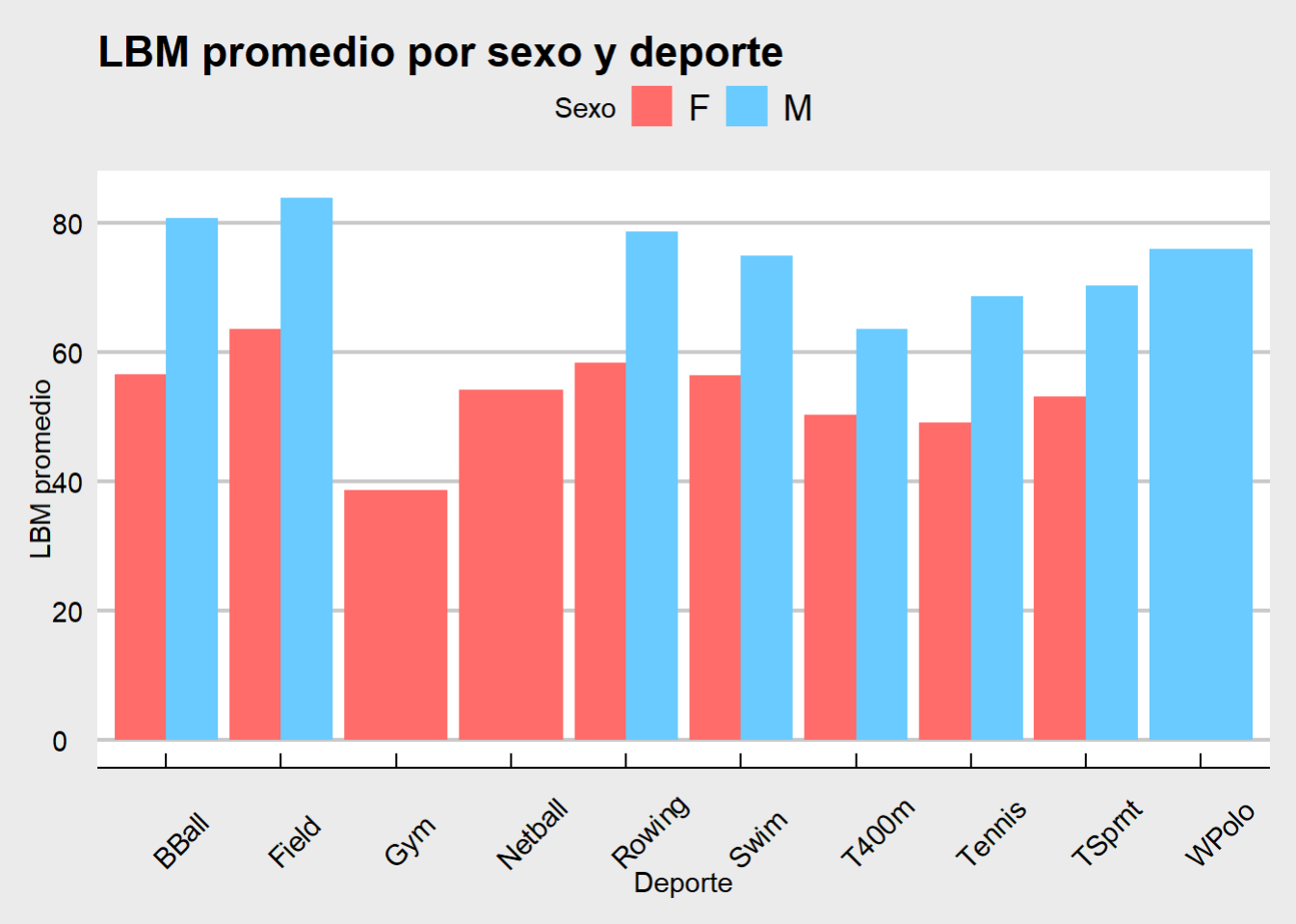
```
ggplot(v1,aes(x=Sport,y=pesopromedio,fill=as.factor(Sex)))+
  geom_bar(stat="identity",position= "dodge")+
  scale_fill_manual(values=c("#ff6c69","#69cbff"))+
  labs(x="Deporte",y="peso promedio",fill="Sexo")+
  ggtitle("Wt promedio por sexo y deporte")+theme_economist_white()+
  theme(axis.text.x = element_text(angle = 45, hjust = .05))
```



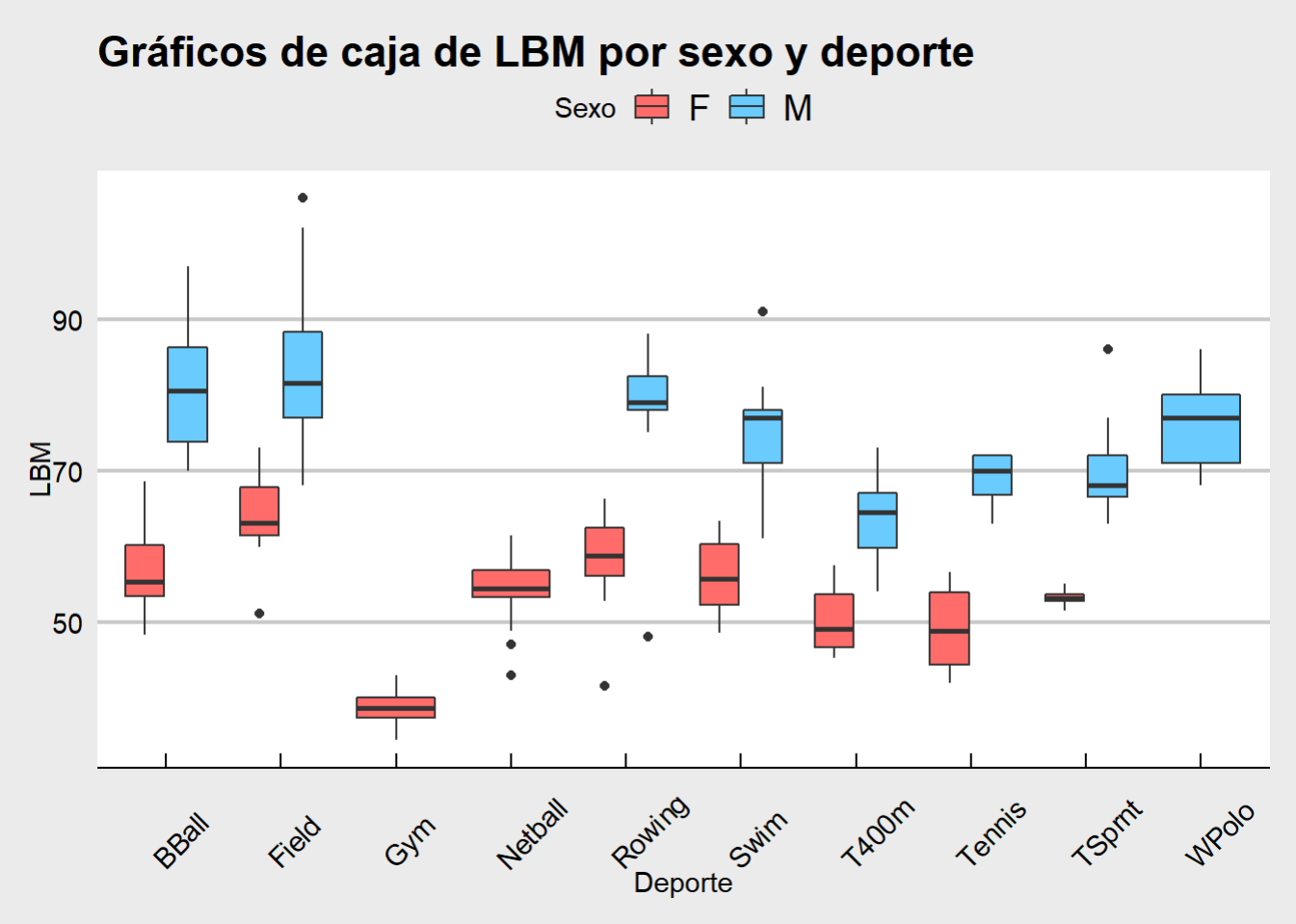

```
ggplot(atleta,aes(x=Sport,y=Wt,fill=as.factor(Sex)))+geom_boxplot()+
  scale_fill_manual(values=c("#ff6c69","#69cbff"))+labs(x="Deporte",fill="Sexo")+
  ggtitle("Gráficos de caja de Wt por sexo y deporte")+theme_economist_white()+
  theme(axis.text.x = element_text(angle = 45, hjust = .05))
```



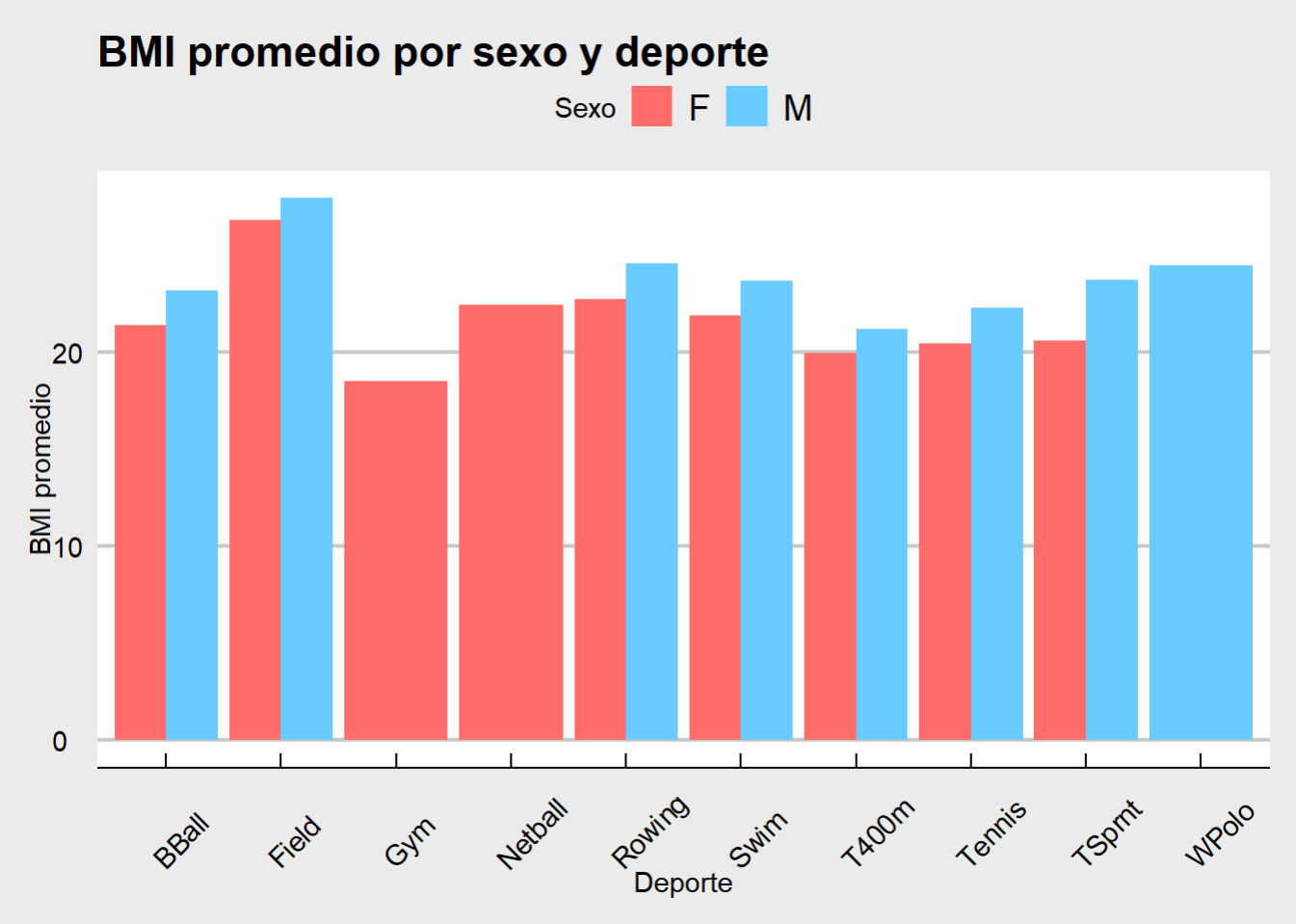

```
ggplot(v1,aes(x=Sport,y=LBMpromedio,fill=as.factor(Sex)))+
  geom_bar(stat="identity",position= "dodge")+
  scale_fill_manual(values=c("#ff6c69","#69cbff"))+
  labs(x="Deporte",y="LBM promedio",fill="Sexo")+
  ggtitle("LBM promedio por sexo y deporte")+theme_economist_white()+
  theme(axis.text.x = element_text(angle = 45, hjust = .05))
```



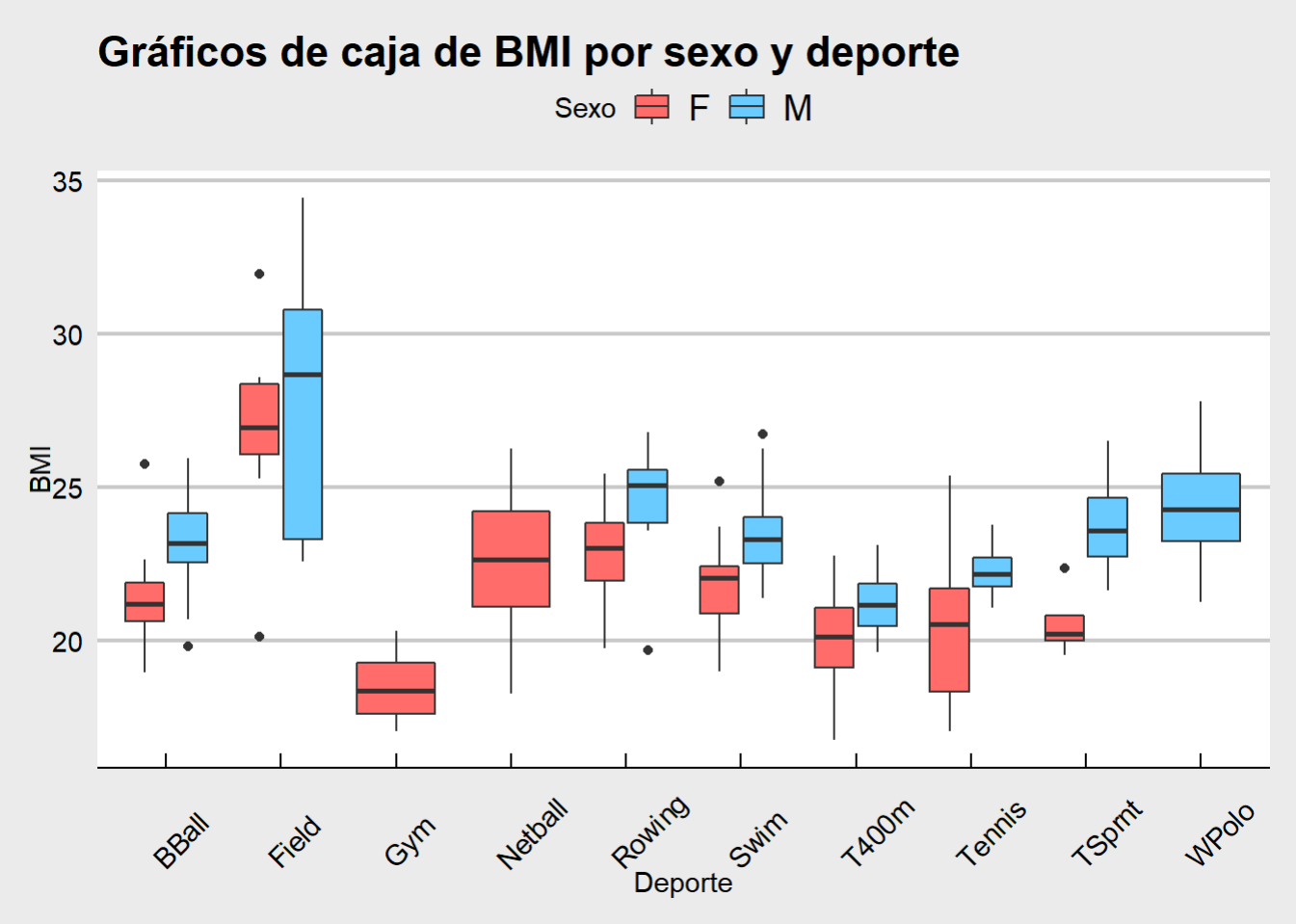

```
ggplot(atleta,aes(x=Sport,y=LBM,fill=as.factor(Sex)))+geom_boxplot()+
  scale_fill_manual(values=c("#ff6c69","#69cbff"))+labs(x="Deporte",fill="Sexo")+
  ggtitle("Gráficos de caja de LBM por sexo y deporte")+theme_economist_white()+
  theme(axis.text.x = element_text(angle = 45, hjust = .05))
```



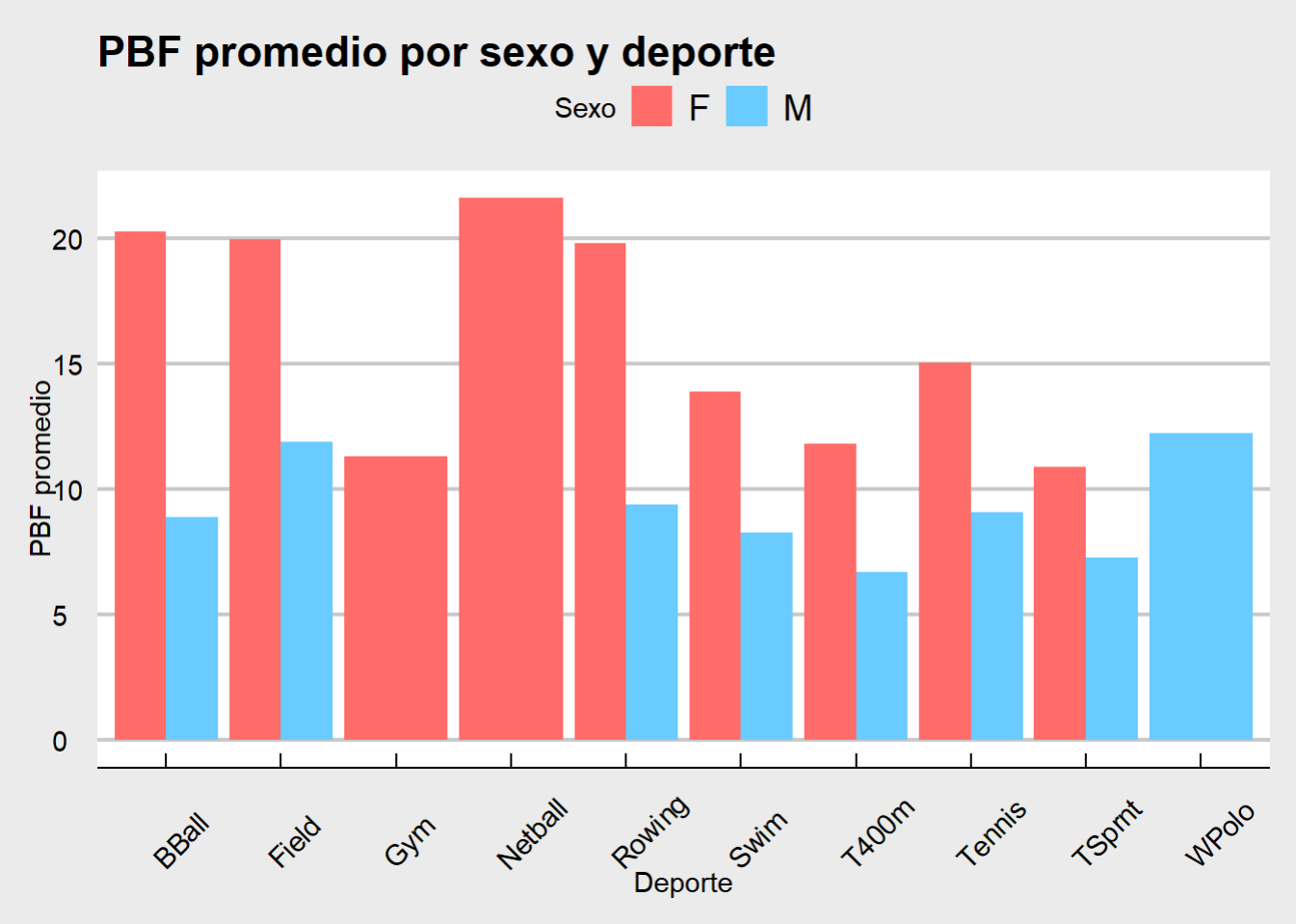

```
ggplot(v1,aes(x=Sport,y=BMIpromedio,fill=as.factor(Sex)))+
  geom_bar(stat="identity",position= "dodge")+
  scale_fill_manual(values=c("#ff6c69","#69cbff"))+
  labs(x="Deporte",y="BMI promedio",fill="Sexo")+
  ggtitle("BMI promedio por sexo y deporte")+theme_economist_white()+
  theme(axis.text.x = element_text(angle = 45, hjust = .05))
```



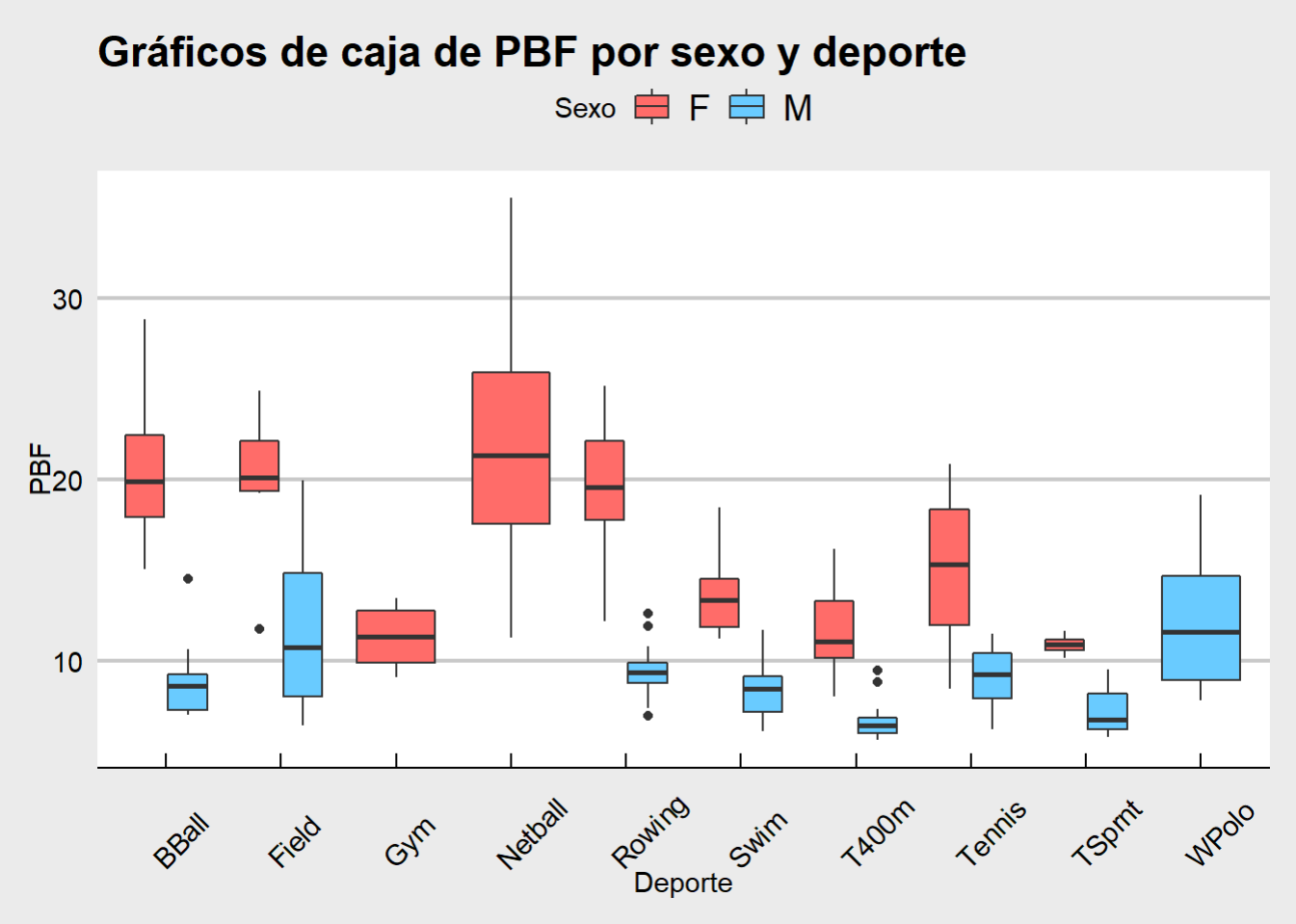

```
ggplot(atleta,aes(x=Sport,y=BMI,fill=as.factor(Sex)))+geom_boxplot()+
  scale_fill_manual(values=c("#ff6c69","#69cbff"))+labs(x="Deporte",fill="Sexo")+
  ggtitle("Gráficos de caja de BMI por sexo y deporte")+theme_economist_white()+
  theme(axis.text.x = element_text(angle = 45, hjust = .05))
```




```
ggplot(v1,aes(x=Sport,y=PBFpromedio,fill=as.factor(Sex)))+
  geom_bar(stat="identity",position= "dodge")+
  scale_fill_manual(values=c("#ff6c69","#69cbff"))+
  labs(x="Deporte",y="PBF promedio",fill="Sexo")+
  ggtitle("PBF promedio por sexo y deporte")+theme_economist_white()+
  theme(axis.text.x = element_text(angle = 45, hjust = .05))
```



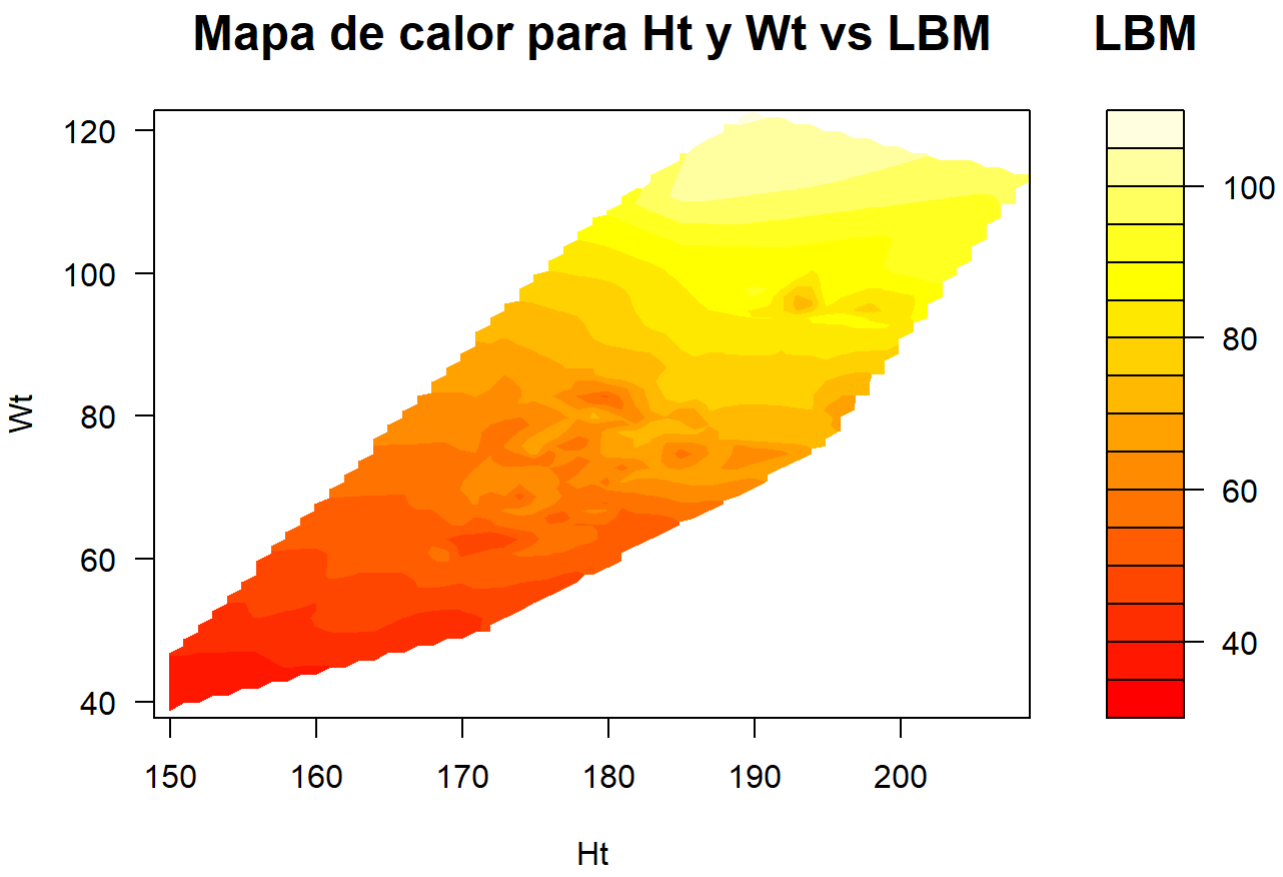

```
ggplot(atleta,aes(x=Sport,y=PBF,fill=as.factor(Sex)))+geom_boxplot()+
  scale_fill_manual(values=c("#ff6c69","#69cbff"))+labs(x="Deporte",fill="Sexo")+
  ggtitle("Gráficos de caja de PBF por sexo y deporte")+theme_economist_white()+
  theme(axis.text.x = element_text(angle = 45, hjust = .05))
```



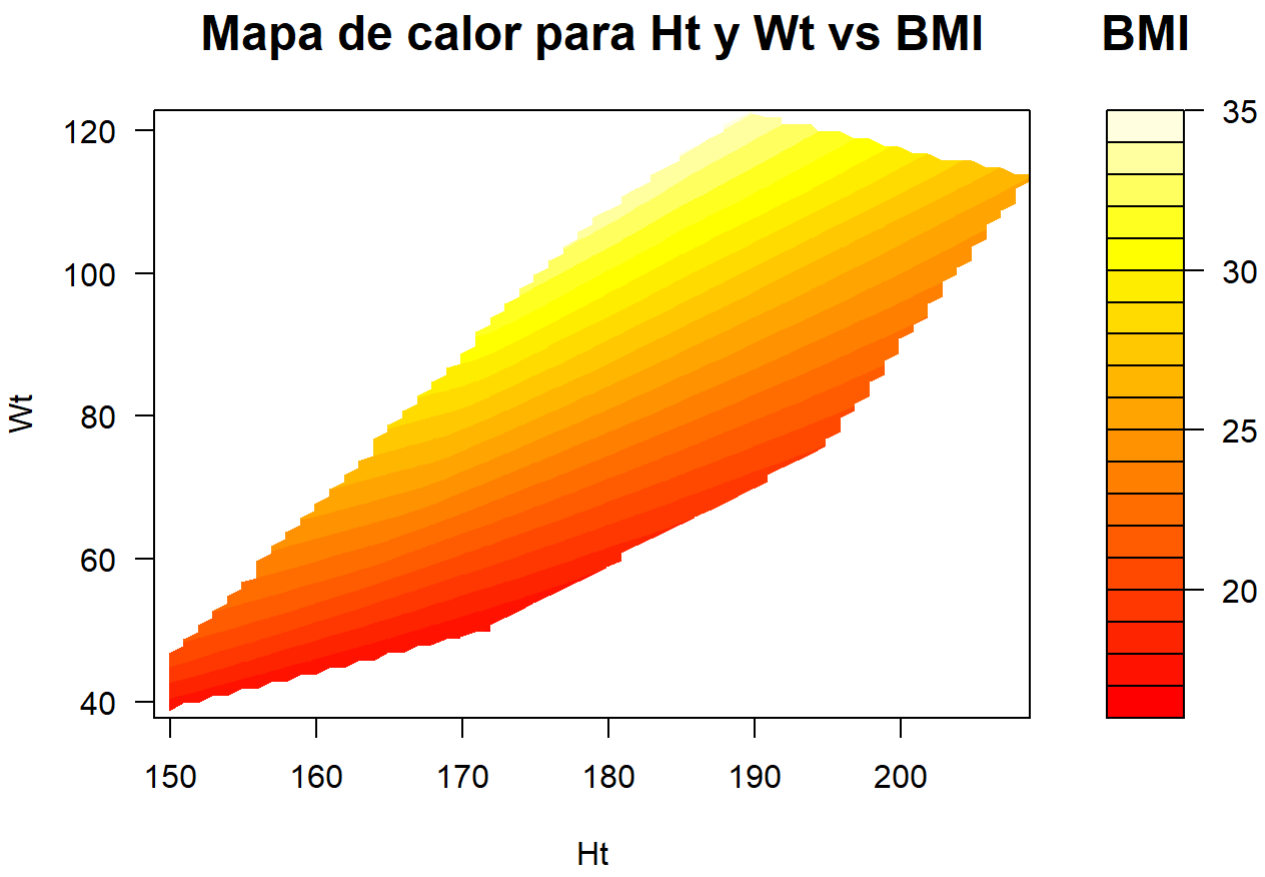
Como era de esperarse, sin importar el deporte que se practique los datos para todas las variables, con excepión de “PBF”, presentan valores más altos para el caso de atletas masculinos y esto tiene que ver con la complexión natural del cuerpo humano. La única variable que claramente presenta valores más altos en todos los deportes para las mujeres es “PBF”. Otro punto a considerar es que los deportistas que practican “Field” presentaron valores más altos de entre todos los deportes para las variables estudiadas con excepción de PBF.

5. Elabora el mapa de calor de las variables Ht, Wt contra cada una de las variables LBM, BMI, PBF, ¿qué conclusiones se pueden obtener?

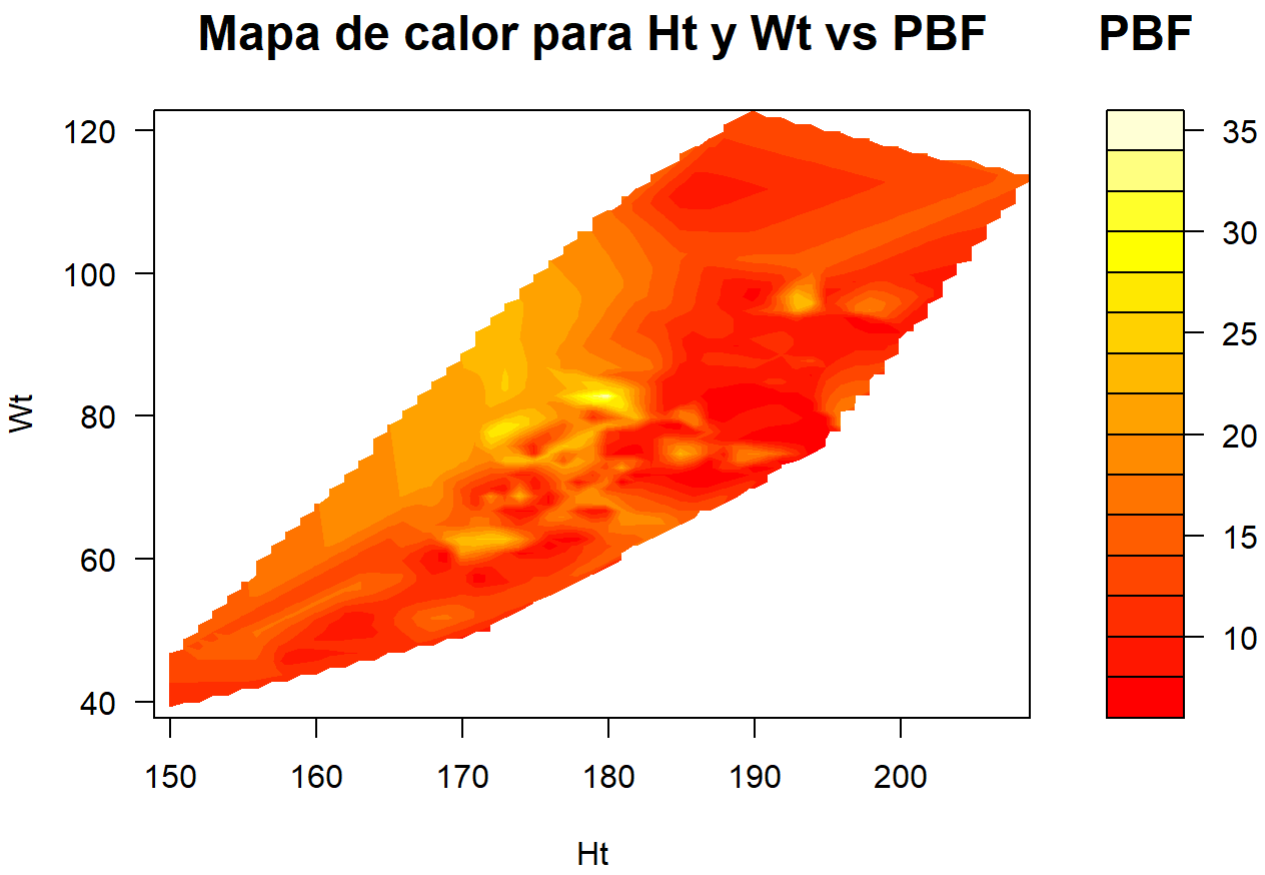
```
resolution <- 1
a <- interp(x=atleta$Ht, y=atleta$Wt, z=atleta$LBM,
            xo=seq(min(atleta$Ht),max(atleta$Ht),by=resolution),
            yo=seq(min(atleta$Wt),max(atleta$Wt),by=resolution), duplicate="mean")
filled.contour(a, color.palette=heat.colors,plot.title = title(main = "Mapa de calor para Ht y W
t vs LBM",
                                                               xlab = "Ht",ylab = "Wt"),key.titl
e = {par(cex.main=1.5);title(main="LBM")})
```




```
b <- interp(x=atleta$Ht, y=atleta$Wt, z=atleta$BMI,
            xo=seq(min(atleta$Ht),max(atleta$Ht),by=resolution),
            yo=seq(min(atleta$Wt),max(atleta$Wt),by=resolution), duplicate="mean")
filled.contour(b, color.palette=heat.colors,plot.title = title(main = "Mapa de calor para Ht y W
t vs BMI",
                                                               xlab = "Ht",ylab = "Wt"),key.titl
e = {par(cex.main=1.5);title(main="BMI")})
```




```
cc <- interp(x=atleta$Ht, y=atleta$Wt, z=atleta$PBF,
             xo=seq(min(atleta$Ht),max(atleta$Ht),by=resolution),
             yo=seq(min(atleta$Wt),max(atleta$Wt),by=resolution), duplicate="mean")
filled.contour(cc, color.palette=heat.colors,plot.title = title(main = "Mapa de calor para Ht y
 Wt vs PBF",
                                                                xlab = "Ht",ylab = "Wt"),key.titl
e = {par(cex.main=1.5);title(main="PBF")})
```



A partir de los mapas de calor observados anteriormente podemos concluir: La relación entre las variables Ht y Wt con LBM es positiva, cuando tanto Wt y Ht aumentan, también aumenta LBM y esto tiene que ver en que las variables estudiadas son el peso y altura de los atletas contra su índice de masa corporal magra. Si ahora nos enfocamos en el mapa de calor de Wt y Ht vs BMI la relación sigue siendo positiva entre las tres variables aunque ahora vemos que sí hay efectos en BMI si una variable se mantiene constante y la otra aumenta de entre las variables Wt o Ht. En este caso podemos observar que si Ht aumenta manteniéndose constante Wt, BMI disminuye mientras que si Wt aumenta y Ht se mantiene constante, BMI aumenta. Estos cambios mencionados le dan esa forma tan particular al mapa de calor. Por último tenemos el mapa de calor entre Wt y Ht contra PBF, en este caso podemos notar la buena condición de los atletas pues en general el mapa de color se mantiene en rojo indicándonos valores bajos para PBF (porcentaje de grasa corporal). La única zona que es importante destacar es cuando los atletas presentan valores para Ht al rededor de 165 y 180, y al mismo tiempo tienen valores altos para Wt. En este caso el mapa de calor nos indica que su PBF aumenta considerablemente.

6. Calcula el coeficiente de correlación, la rho de Spearman y la tau de Kendall para cada par de variables que se pueden formar con las variables LBM, BMI, PBF.

```
correlacion<-atleta[,4:8]
coefdecorrel<-cor(correlacion)
```

Correlación de Pearson

```
coefdecorrel[,3:5]
```

```
##            LBM       BMI           PBF
## Ht   0.8021192 0.3370972 -0.1880216785
## Wt   0.9309040 0.8459551 -0.0001618851
## LBM  1.0000000 0.7138581 -0.3618504448
## BMI  0.7138581 1.0000000  0.1875577578
## PBF -0.3618504 0.1875578  1.0000000000
```

Correlación de Spearman

```
rho<-cor(correlacion,method = "spearman")
rho[,3:5]
```

```
##            LBM       BMI         PBF
## Ht   0.8007044 0.3540467 -0.24988825
## Wt   0.9141622 0.8402575 -0.04764267
## LBM  1.0000000 0.7026060 -0.40605651
## BMI  0.7026060 1.0000000  0.14943455
## PBF -0.4060565 0.1494345  1.00000000
```

Correlación de Kendall

```
tau<-cor(correlacion,method = "kendall")
tau[,3:5]
```

```
##            LBM       BMI          PBF
## Ht   0.6073244 0.2387399 -0.162040862
## Wt   0.7721239 0.6520292 -0.008387814
## LBM  1.0000000 0.5175922 -0.239573787
## BMI  0.5175922 1.0000000  0.109532968
## PBF -0.2395738 0.1095330  1.000000000
```

7. Construye dos bases de datos una para hombres y otra para mujeres, que tengan lo siguiente: Para cada deporte calcula en valor promedio de las variables Ht, Wt, LBM, BMI, PBF en base a los deportistas que practican ese deporte. Nombra los renglones de esa base de datos precisamente con el nombre del deporte que le corresponda.

Para las mujeres:

```
mujer<-atleta[atleta$Sex=="F",]
bdmujer<-mujer %>% group_by(Sport) %>% summarise(Htpromedio=mean(Ht),Wtpromedio=mean(Wt),
                                        LBMpromedio=mean(LBM),BMIpromedio=mean(BMI),
                                        PBFpromedio=mean(PBF))
names(bdmujer)=c("Deporte","Ht promedio","Wt promedio","LBM promedio",
                  "BMI promedio","PBF promedio")
bdmujeres<-data.frame(bdmujer[,2:6])
rownames(bdmujeres)<-bdmujer$Deporte
bdmujeres
```

```
##         Ht.promedio Wt.promedio LBM.promedio BMI.promedio PBF.promedio
## BBall      182.2692    71.33077     56.65923     21.41077     20.26615
## Field      172.5857    80.04286     63.68857     26.83143     19.97571
## Gym        153.4250    43.62500     38.66000     18.52000     11.31750
## Netball    176.0870    69.59348     54.26304     22.43957     21.60913
## Rowing     178.8591    72.90000     58.41955     22.75136     19.79136
## Swim       173.1778    65.73333     56.49778     21.89444     13.88778
## T400m      169.3364    57.23636     50.37727     19.96818     11.82182
## Tennis     168.5714    58.22857     49.10000     20.42571     15.04571
## TSprnt     170.4750    59.72500     53.21250     20.59000     10.89500
```

Para los hombres:

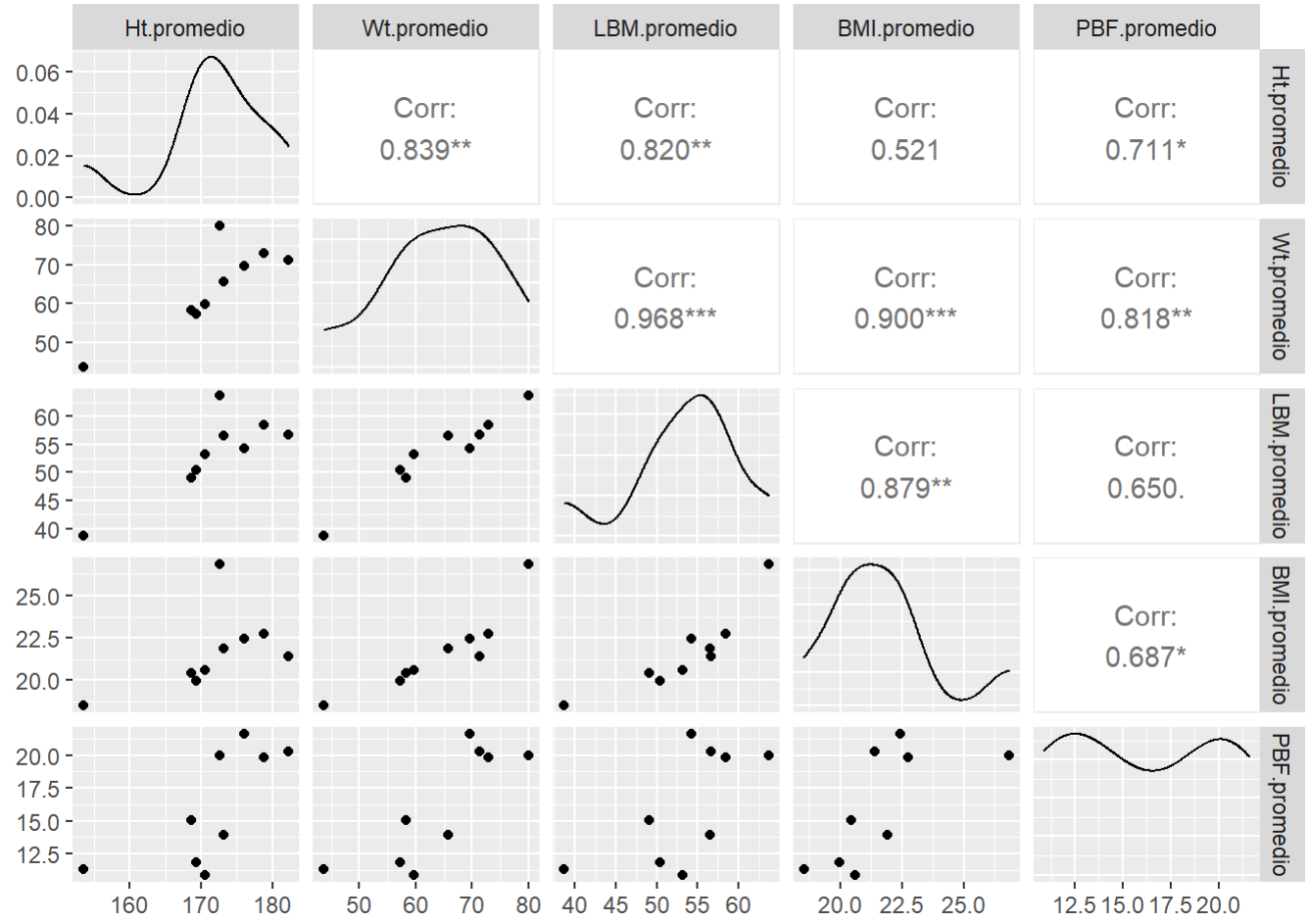
```
hombres<-atleta[atleta$Sex=="M",]
bdhombre<-hombres %>% group_by(Sport) %>% summarise(Htpromedio=mean(Ht),Wtpromedio=mean(Wt),
                                                    LBMpromedio=mean(LBM),BMIpromedio=mean(BM
I),
                                                    PBFpromedio=mean(PBF))
names(bdhombre)=c("Deporte","Ht promedio","Wt promedio","LBM promedio",
                  "BMI promedio","PBF promedio")
bdhombres<-data.frame(bdhombre[,2:6])
rownames(bdhombres)<-bdhombre$Deporte
bdhombres
```

```
##        Ht.promedio Wt.promedio LBM.promedio BMI.promedio PBF.promedio
## BBall     195.5833    88.92500     80.83333     23.17667     8.893333
## Field     185.2750    95.76250     83.91667     27.95250    11.907500
## Rowing    187.5333    86.80667     78.66667     24.59333     9.409333
## Swim      185.6462    81.66154     74.92308     23.66154     8.296154
## T400m     179.1889    68.20833     63.61111     21.21667     6.685556
## Tennis    183.9500    75.40000     68.75000     22.29500     9.080000
## TSprnt    178.5364    75.79091     70.27273     23.73727     7.287273
## WPolo     188.2235    86.72941     75.94118     24.46647    12.245294
```

8. Ajusta un modelo de componentes principales a las bases de datos encontradas en el inciso anterior y compara los gráficos biplot de cada una de las bases.

Para las mujeres:

```
ggpairs(bdmujeres)
```




```
summary(bdmujeres)
```

```
##   Ht.promedio     Wt.promedio     LBM.promedio    BMI.promedio
##  Min.   :153.4   Min.   :43.62   Min.   :38.66   Min.   :18.52
##  1st Qu.:169.3   1st Qu.:58.23   1st Qu.:50.38   1st Qu.:20.43
##  Median :172.6   Median :65.73   Median :54.26   Median :21.41
##  Mean   :171.6   Mean   :64.27   Mean   :53.43   Mean   :21.65
##  3rd Qu.:176.1   3rd Qu.:71.33   3rd Qu.:56.66   3rd Qu.:22.44
##  Max.   :182.3   Max.   :80.04   Max.   :63.69   Max.   :26.83
##   PBF.promedio
##  Min.   :10.89
##  1st Qu.:11.82
##  Median :15.05
##  Mean   :16.07
##  3rd Qu.:19.98
##  Max.   :21.61
```

Notemos que debemos estandarizar

```
mujercomp<-prcomp(bdmujeres,center = TRUE,scale = TRUE)
mujercomp$sdev
```

```
## [1] 2.03312017 0.70311222 0.60902544 0.03165409 0.01190034
```

```
mujercomp$rotation
```

```
##                      PC1         PC2         PC3        PC4         PC5
## Ht.promedio  -0.4207982 -0.67050810 -0.35045877  0.4981825  0.04837965
## Wt.promedio  -0.4909598  0.06693283 -0.05570671 -0.4365256  0.74887961
## LBM.promedio -0.4696238  0.16652451 -0.44789529 -0.4287797 -0.60602053
## BMI.promedio -0.4338316  0.66309594  0.10751531  0.6002767  0.01421949
## PBF.promedio -0.4160901 -0.28020037  0.81357586 -0.1306729 -0.26339260
```

```
mujercomp$x
```

```
##                PC1         PC2         PC3          PC4          PC5
## BBall   -1.4407529 -1.08858633  0.07972040 -0.021828691  0.017610420
## Field   -2.7812735  1.47530128  0.19641343  0.004852564  0.009119916
## Gym      3.8919529  0.43826754  0.79151420 -0.033880312  0.003642444
## Netball -1.2033307 -0.44555362  0.80467556  0.040596688 -0.008315053
## Rowing  -1.6563381 -0.34842013  0.07811747 -0.042601358 -0.008429483
## Swim    -0.1863763  0.16627800 -0.66573103 -0.023669309 -0.019558513
## T400m    1.3589963 -0.12723864 -0.54396133  0.027375156  0.009907340
## Tennis   1.0447075 -0.16753031  0.19034680  0.038491909 -0.008973580
## TSprnt   0.9724149  0.09748221 -0.93109550  0.010663353  0.004996508
```

```
mujercomp$center
```

```
##  Ht.promedio  Wt.promedio LBM.promedio BMI.promedio PBF.promedio
##    171.64295     64.26837     53.43088     21.64794     16.06780
```

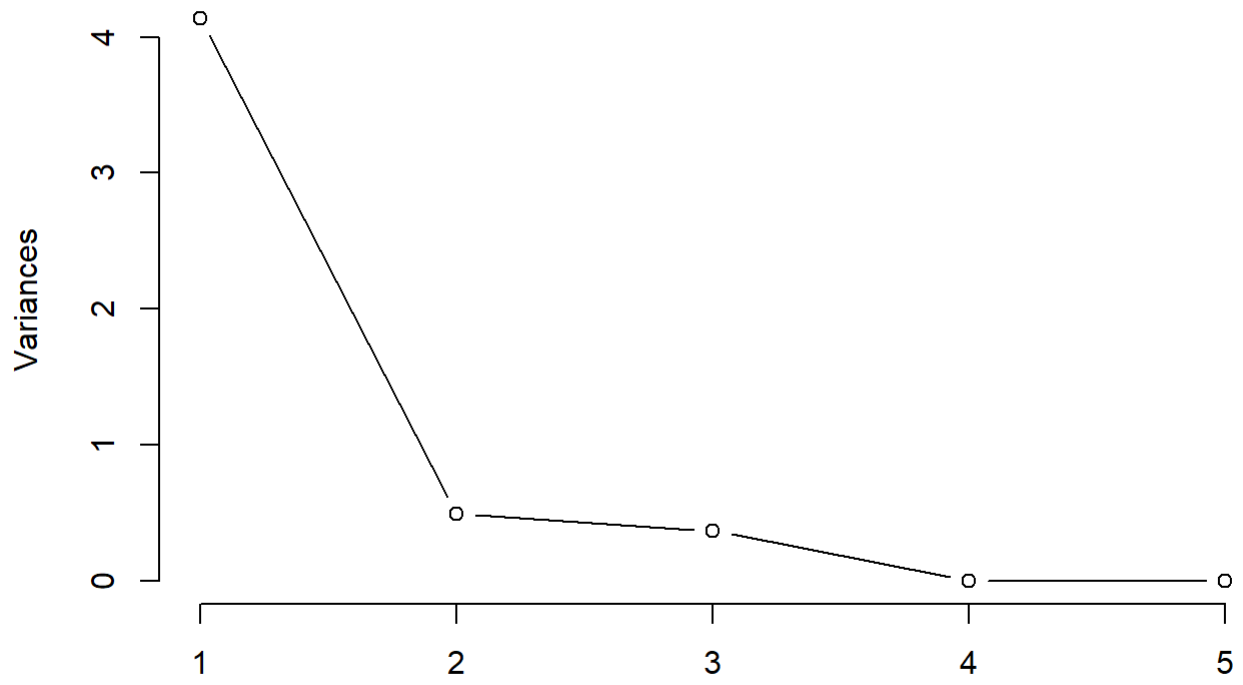
```
mujercomp$scale
```

```
##  Ht.promedio  Wt.promedio LBM.promedio BMI.promedio PBF.promedio
##     8.185327    10.819191     7.043927     2.346499     4.339037
```

```
summary(mujercomp)
```

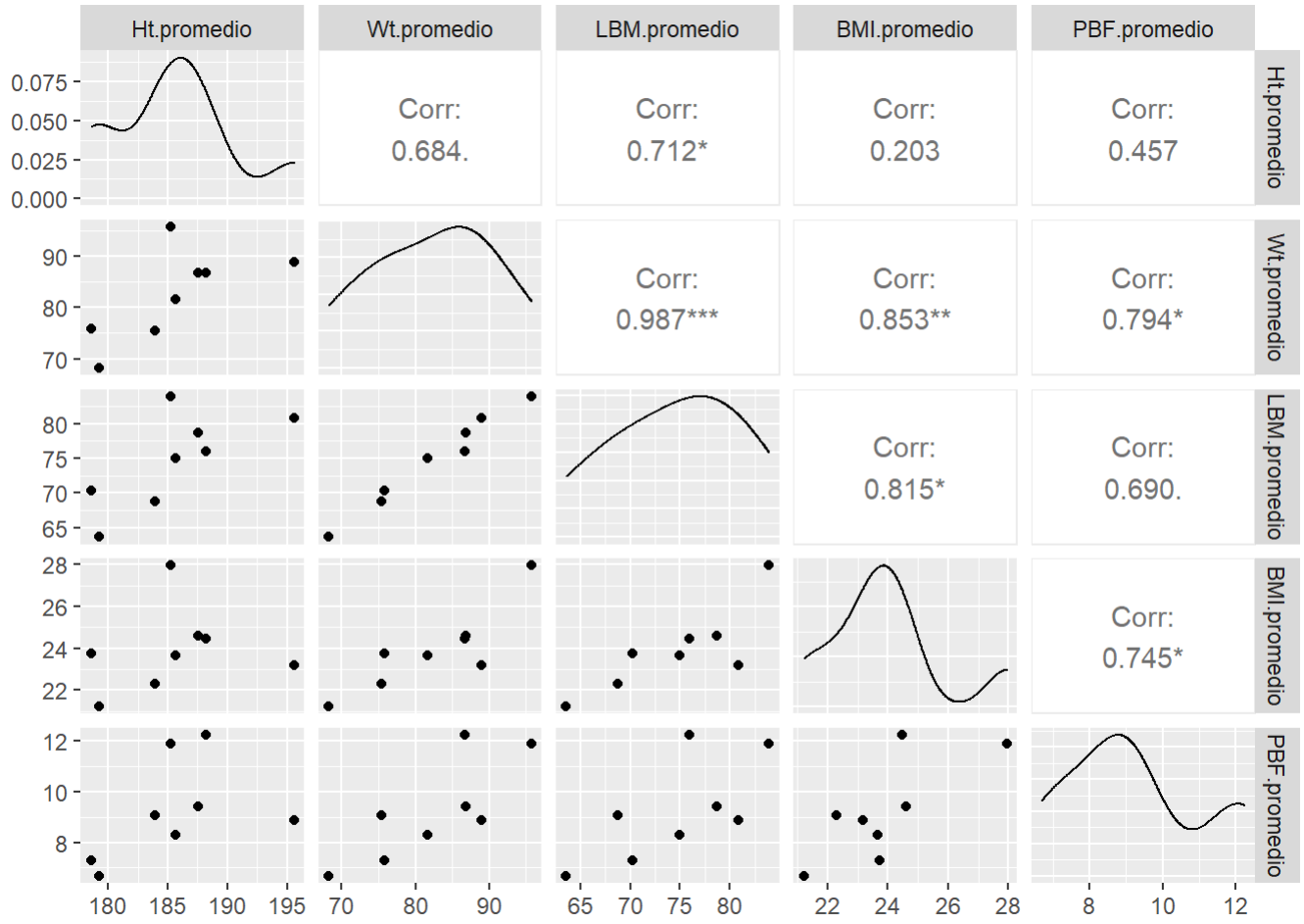
```
## Importance of components:
##                           PC1     PC2     PC3     PC4     PC5
## Standard deviation     2.0331 0.70311 0.60903 0.03165 0.01190
## Proportion of Variance 0.8267 0.09887 0.07418 0.00020 0.00003
## Cumulative Proportion  0.8267 0.92559 0.99977 0.99997 1.00000
```

```
screeplot(mujercomp,type = "l", main="Screeplot base de datos mujeres")
```



Para los hombres:

```
ggpairs(bdhombres)
```




```
summary(bdhombres)
```

```
##   Ht.promedio     Wt.promedio     LBM.promedio    BMI.promedio
##  Min.   :178.5   Min.   :68.21   Min.   :63.61   Min.   :21.22
##  1st Qu.:182.8   1st Qu.:75.69   1st Qu.:69.89   1st Qu.:22.96
##  Median :185.5   Median :84.20   Median :75.43   Median :23.70
##  Mean   :185.5   Mean   :82.41   Mean   :74.61   Mean   :23.89
##  3rd Qu.:187.7   3rd Qu.:87.34   3rd Qu.:79.21   3rd Qu.:24.50
##  Max.   :195.6   Max.   :95.76   Max.   :83.92   Max.   :27.95
##   PBF.promedio
##  Min.   : 6.686
##  1st Qu.: 8.044
##  Median : 8.987
##  Mean   : 9.226
##  3rd Qu.:10.034
##  Max.   :12.245
```

También debemos estandarizar

```
hombrecomp<-prcomp(bdhombres,center = TRUE,scale = TRUE)
hombrecomp$sdev
```

```
## [1] 1.95804703 0.91347870 0.57514734 0.02427820 0.01498755
```

```
hombrecomp$rotation
```

```
##                     PC1         PC2         PC3        PC4         PC5
## Ht.promedio  -0.3479001  0.79894845  0.09910264 -0.2263012  0.42380812
## Wt.promedio  -0.5082845  0.01683439 -0.16492974 -0.5030629 -0.67903561
## LBM.promedio -0.4946854  0.09807195 -0.40183361  0.7587923 -0.09182731
## BMI.promedio -0.4340272 -0.54971575 -0.27752013 -0.2872931  0.59150506
## PBF.promedio -0.4330348 -0.22269364  0.85116930  0.1934229 -0.03141373
```

```
hombrecomp$x
```

```
##               PC1         PC2         PC3          PC4          PC5
## BBall  -1.2512491  1.83028331 -0.34963794 -0.020621049  0.003709222
## Field  -2.9057523 -1.29778076 -0.22369378 -0.024994580  0.005313720
## Rowing -0.8746396  0.15335299 -0.30550393  0.037975728 -0.023210070
## Swim    0.2624431  0.19273701 -0.36889015  0.012730443  0.012495926
## T400m   3.1636687 -0.09540701  0.08725745 -0.033027834 -0.016028017
## Tennis  1.3103056  0.13031021  0.61149403  0.016868117  0.021997760
## TSprnt  1.6011363 -0.84594333 -0.55666801  0.009730158  0.004402314
## WPolo  -1.3059127 -0.06755241  1.10564232  0.001339018 -0.008680855
```

```
hombrecomp$center
```

```
##  Ht.promedio  Wt.promedio LBM.promedio BMI.promedio PBF.promedio
##   185.492075    82.410545    74.614345    23.887431     9.225555
```

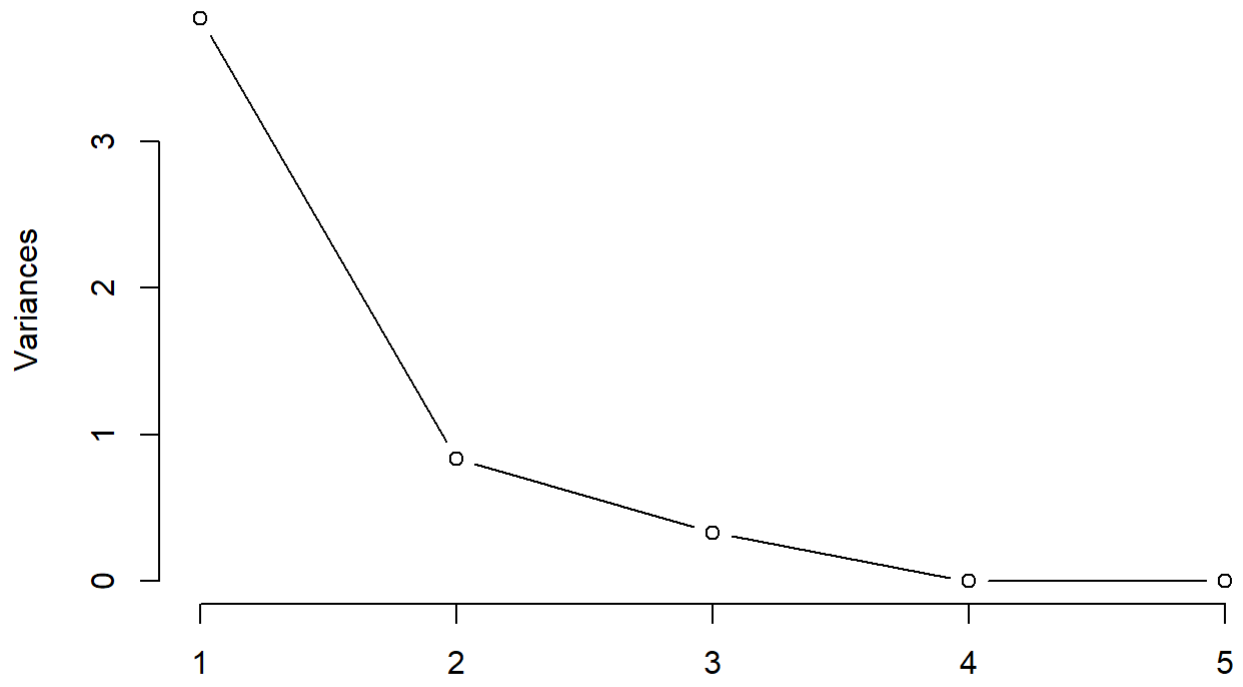
```
hombrecomp$scale
```

```
##  Ht.promedio  Wt.promedio LBM.promedio BMI.promedio PBF.promedio
##     5.398704     8.895605     6.737925     1.985877     1.983839
```

```
summary(hombrecomp)
```

```
## Importance of components:
##                           PC1    PC2     PC3     PC4     PC5
## Standard deviation     1.9580 0.9135 0.57515 0.02428 0.01499
## Proportion of Variance 0.7668 0.1669 0.06616 0.00012 0.00004
## Cumulative Proportion  0.7668 0.9337 0.99984 0.99996 1.00000
```

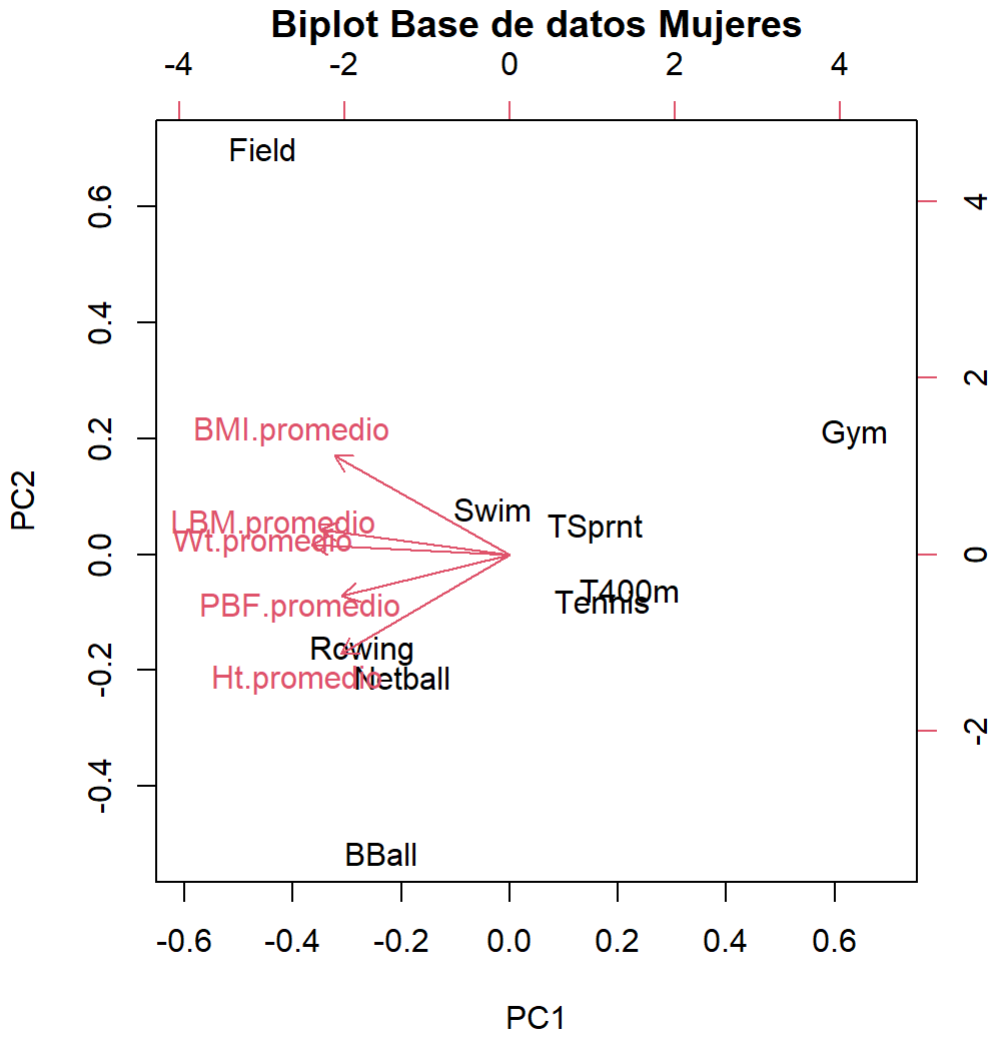
```
screeplot(hombrecomp,type = "l", main="Screeplot base de datos hombres")
```



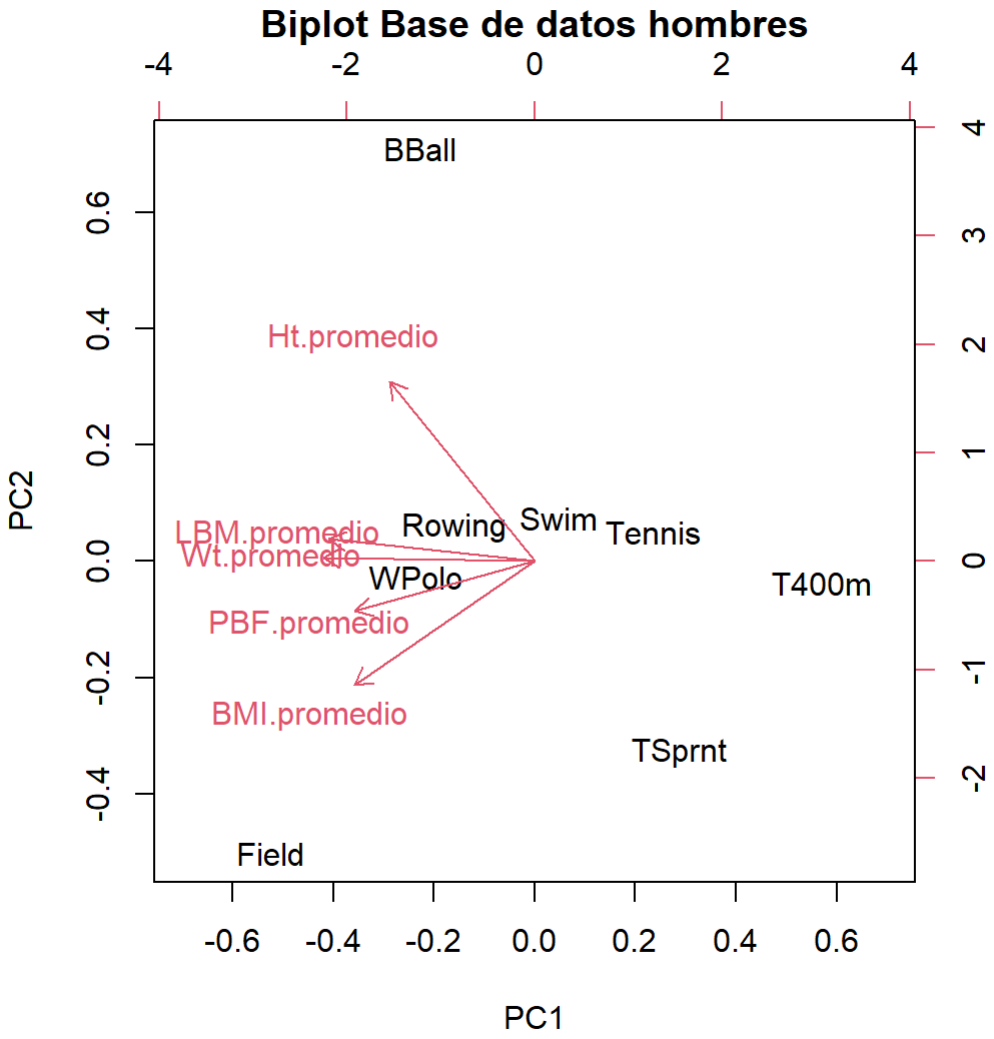
En ambos casos nos quedaremos con dos componentes principales al observar sus screeplot y la proporción acumulada superior al 90%

Comparando ambas biplot:

```
biplot(mujercomp,scale = 1, main="Biplot Base de datos Mujeres",xlim=c(-.6,.7))
```




```
biplot(hombrecomp,scale=1,main="Biplot Base de datos hombres",xlim=c(-.7,.7))
```

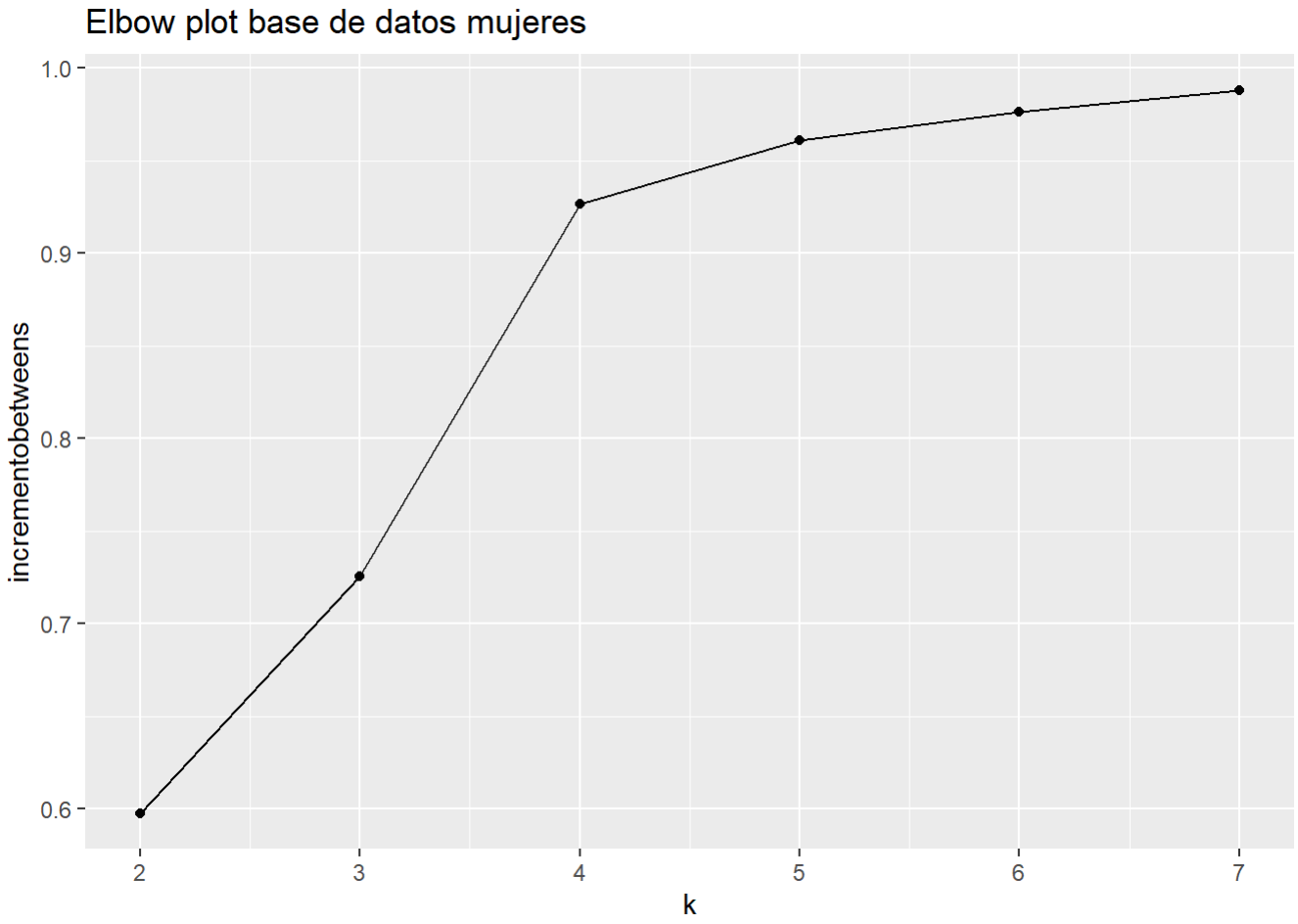


Comparando ambas biplot: Los hombres que practican el deporte T400 se encuentran con valores mucho más bajos en sus respectivos promedios para las variables Ht, Wt, BMI, LBM y PBF a comparación de las mujeres que tienen promedios más altos en las variables mencionadas anteriormente. Por el lado del deporte TSprnt, los hombres que lo practican presentan un BMI promedio más alto que las mujeres además de una Ht promedio inferior. Las mujeres que practican Swim tienen BMI promedio mayores al de los hombres y Rowing es un deporte que en el caso de las mujeres, presenta Ht promedio más altas que en el caso de los hombres donde ninguna variable tiene demasiada influencia en este deporte.

9. Si realizas un análisis por comglomerados (clustering), cuáles son los deportes que más se parecen en cuanto a las características de sus deportistas?

Para las mujeres por el método de k-medias:

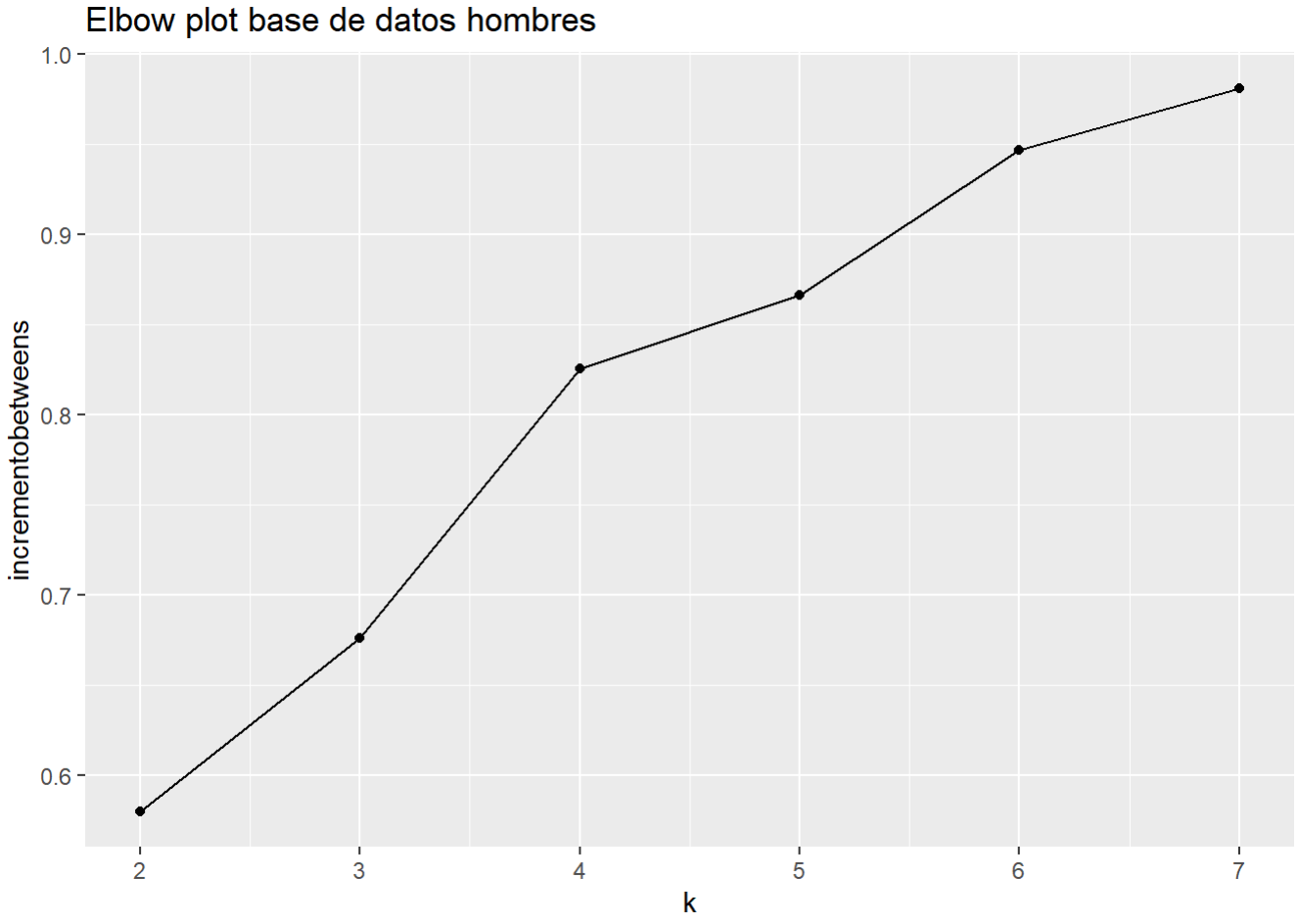
```
clumujeres<-scale(bdmujeres)
kme1<-kmeans(clumujeres,2)
kme2<-kmeans(clumujeres,3)
kme3<-kmeans(clumujeres,4)
kme4<-kmeans(clumujeres,5)
kme5<-kmeans(clumujeres,6)
kme6<-kmeans(clumujeres,7)

elbow<-data.frame(k=c(2:7),incrementobetweens=c(kme1$betweenss/kme1$totss,
                                                kme2$betweenss/kme2$totss,
                                                kme3$betweenss/kme3$totss,
                                                kme4$betweenss/kme4$totss,
                                                kme5$betweenss/kme5$totss,
                                                kme6$betweenss/kme6$totss))

ggplot(elbow,aes(x=k,y=incrementobetweens))+geom_point()+geom_line()+
  ggtitle("Elbow plot base de datos mujeres")
```



Nos quedaremos con 4 clusters

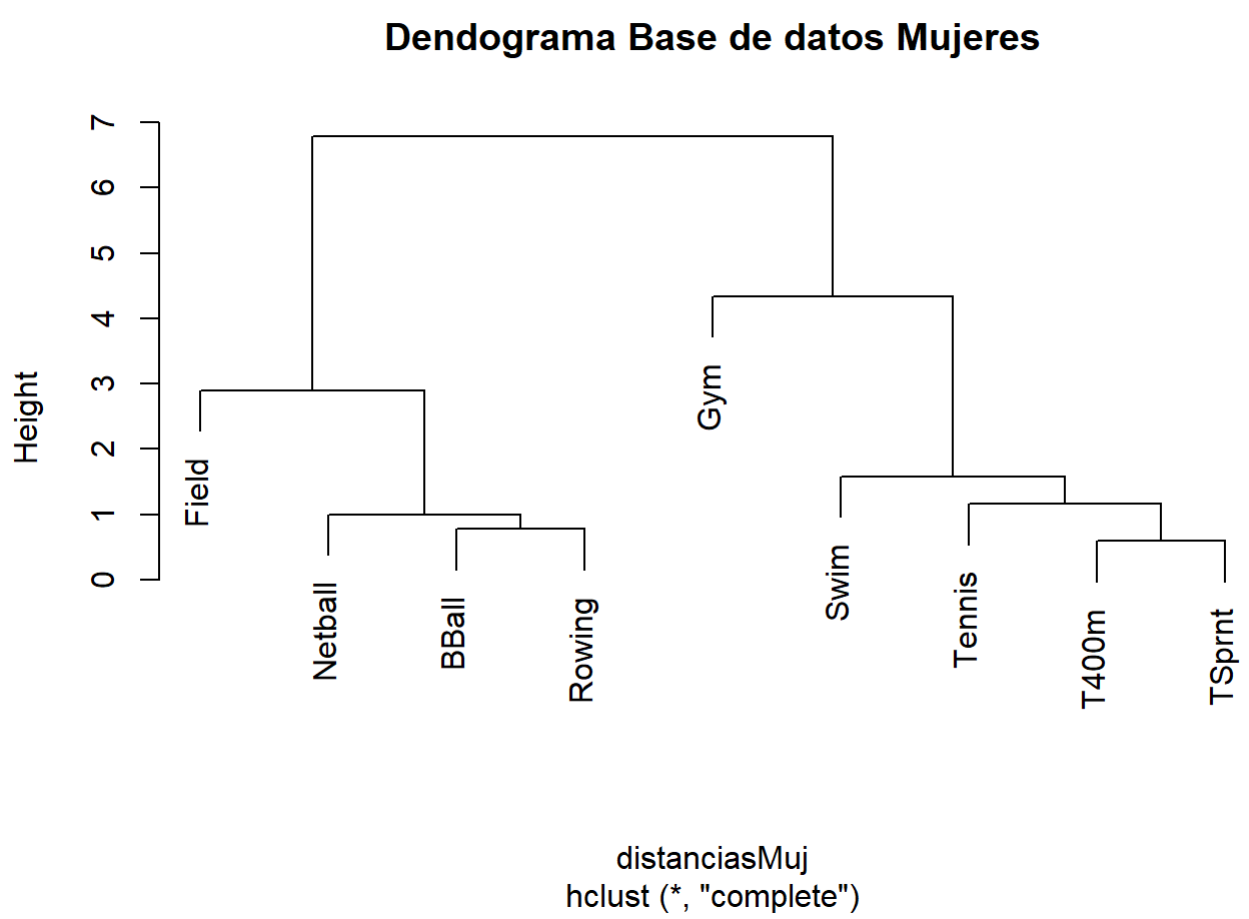
Para los hombres por el método de k-medias:

```
cluhombres<-scale(bdhombres)
kme11<-kmeans(cluhombres,2)
kme21<-kmeans(cluhombres,3)
kme31<-kmeans(cluhombres,4)
kme41<-kmeans(cluhombres,5)
kme51<-kmeans(cluhombres,6)
kme61<-kmeans(cluhombres,7)

elbow<-data.frame(k=c(2:7),incrementobetweens=c(kme11$betweenss/kme11$totss,
                                                kme21$betweenss/kme21$totss,
                                                kme31$betweenss/kme31$totss,
                                                kme41$betweenss/kme41$totss,
                                                kme51$betweenss/kme51$totss,
                                                kme61$betweenss/kme61$totss
                                                ))

ggplot(elbow,aes(x=k,y=incrementobetweens))+geom_point()+geom_line()+
  ggtitle("Elbow plot base de datos hombres")
```



Para las mujeres por el método jerárquico:

```
distanciasMuj<-dist(clumujeres)
dendoMuj<-hclust(distanciasMuj)
plot(dendoMuj,main = "Dendograma Base de datos Mujeres")
```



Confirma que nos quedemos con 4 clusters

```
gruposmu<-cutree(dendoMuj,4)
grafomuj<-data.frame(mujercomp$x[,1:2],as.factor(gruposmu))
names(grafomuj)<-c("PC1","PC2","clustercomplete")
```

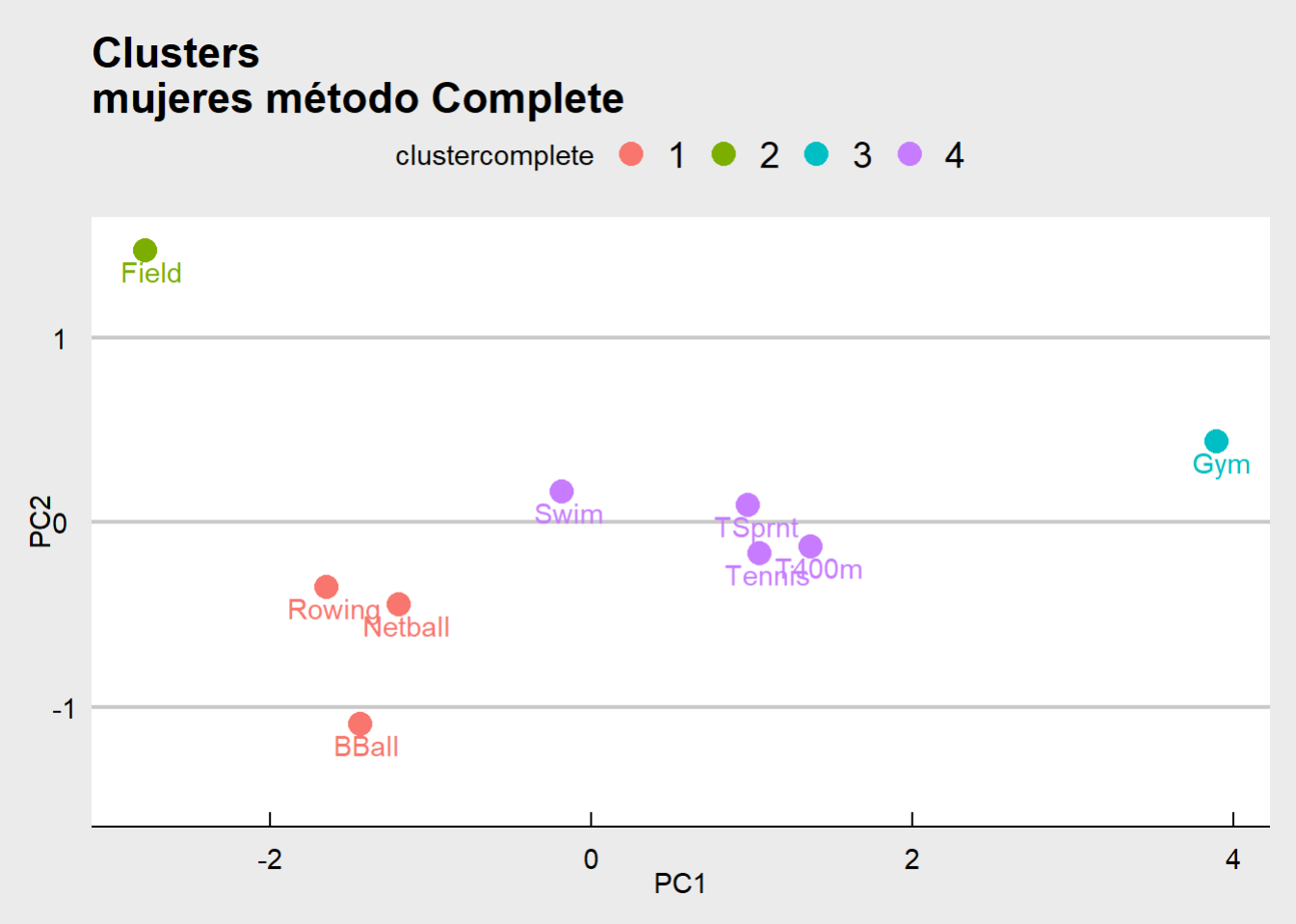
Los deportes en el caso de las mujeres se pueden agrupar como se ve en la siguiente tabla:

```
grafomuj
```

```
##                PC1         PC2 clustercomplete
## BBall   -1.4407529 -1.08858633               1
## Field   -2.7812735  1.47530128               2
## Gym      3.8919529  0.43826754               3
## Netball -1.2033307 -0.44555362               1
## Rowing  -1.6563381 -0.34842013               1
## Swim    -0.1863763  0.16627800               4
## T400m    1.3589963 -0.12723864               4
## Tennis   1.0447075 -0.16753031               4
## TSprnt   0.9724149  0.09748221               4
```

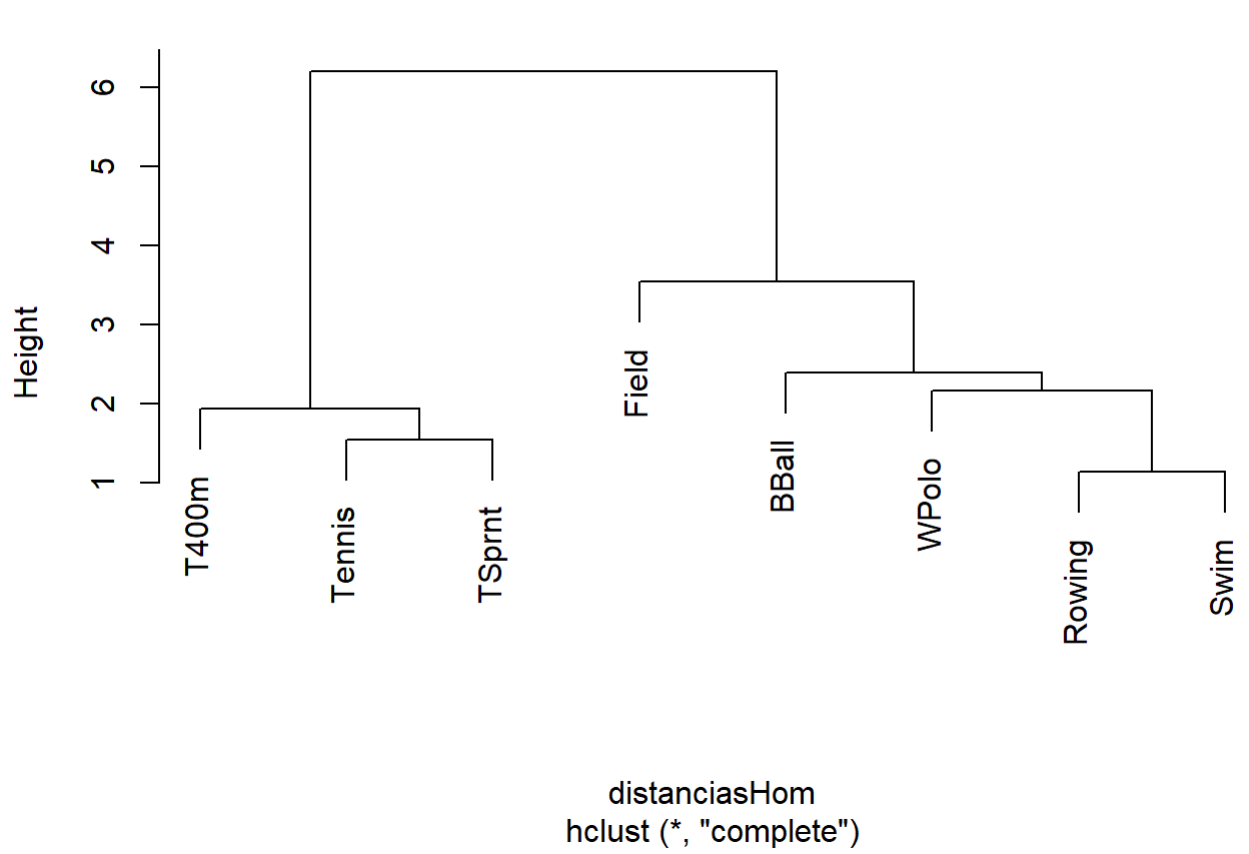
Graficamente se agrupan como sigue:

```
ggplot(grafomuj,aes(x=PC1,y=PC2,col=clustercomplete))+geom_point(size=4)+ylim(-1.5,1.5)+ggtitle(
"Clusters
mujeres método Complete")+theme_economist_white()+geom_text(aes(label=rownames(grafomuj)),hjust=
.4, vjust=1.5)
```



Para los hombres por el método jerárquico:

```
distanciasHom<-dist(cluhombres)
dendoHom<-hclust(distanciasHom)
plot(dendoHom,main = "Dendograma Base de datos Hombres")
```



Nos podemos quedar con 4 clusters

```
gruposhom<-cutree(dendoHom,4)
grafohom<-data.frame(hombrecomp$x[,1:2],as.factor(gruposhom))
names(grafohom)<-c("PC1","PC2","clustercomplete")
```

Los deportes en el caso de los hombres se pueden agrupar como se muestra en la siguiente tabla:

```
grafohom
```

```
##               PC1         PC2 clustercomplete
## BBall  -1.2512491  1.83028331               1
## Field  -2.9057523 -1.29778076               2
## Rowing -0.8746396  0.15335299               3
## Swim    0.2624431  0.19273701               3
## T400m   3.1636687 -0.09540701               4
## Tennis  1.3103056  0.13031021               4
## TSprnt  1.6011363 -0.84594333               4
## WPolo  -1.3059127 -0.06755241               3
```

Graficamente se agrupan como sigue:

```
ggplot(grafohom,aes(x=PC1,y=PC2,col=clustercomplete))+geom_point(size=4)+ylim(-1.5,2)+ggtitle("C
lusters
hombres método Complete")+theme_economist_white()+geom_text(aes(label=rownames(grafohom)),hjust=
.4, vjust=1.5)
```