7-regresion-lineal

September 5, 2023

1 7. Regresión lineal

Francisco Mestizo Hernández A01731549

1.1 Análisis de las variables

Iniciamos las librerias que utilizaremos más adelante para el análisis

[ ]:
```
install.packages("nortest")
library(nortest)
```

```
Installing package into '/usr/local/lib/R/site-library'
(as 'lib' is unspecified)
```

Comenzamos cargando los datos del archivo y mostrando los iniciales para confirmar que están correctos.

[ ]:
```
#Leemos los datos del csv (No los imprimo porque ocupan mucho espacio en la␣
↪pantalla)
M = read.csv('/content/sample_data/Estatura-peso_HyM.csv')
head(M)
#Hay que recordar poner el csv en los archivos del colab
```

A data.frame: 6 × 3

	Estatura <dbl>	Peso <dbl>	Sexo <chr>
1	1.61	72.21	H
2	1.61	65.71	H
3	1.70	75.08	H
4	1.65	68.55	H
5	1.72	70.77	H
6	1.63	77.18	H

Como tenemos nuestros datos para hombres y para mujeres, podemos generar una nueva M llamada M1 donde esten divididos los datos para hombres y los datos para mujeres

[ ]:
```
#Medidas
MM = subset(M,M$Sexo=="M")
MH = subset(M,M$Sexo=="H")
M1=data.frame(MH$Estatura,MH$Peso,MM$Estatura,MM$Peso)
```

```
n=4 #número de variables
d=matrix(NA,ncol=7,nrow=n)
for(i in 1:n){
  d[i,]<-c(as.numeric(summary(M1[,i])),sd(M1[ ,i]))
}
m=as.data.frame(d)

row.names(m)=c("H-Estatura","H-Peso","M-Estatura","M-Peso")
names(m)=c("Minimo","Q1","Mediana","Media","Q3","Máximo","Desv Est")
m
```

A data.frame: 4 × 7

	Minimo <dbl>	Q1 <dbl>	Mediana <dbl>	Media <dbl>	Q3 <dbl>	Máximo <dbl>	Desv Est <dbl>
H-Estatura	1.48	1.6100	1.650	1.653727	1.7000	1.80	0.06173088
H-Peso	56.43	68.2575	72.975	72.857682	77.5225	90.49	6.90035408
M-Estatura	1.44	1.5400	1.570	1.572955	1.6100	1.74	0.05036758
M-Peso	37.39	49.3550	54.485	55.083409	59.7950	80.87	7.79278074

Se muestran las medidas principales para el análisis de los datos.

[ ]:
```
summary(M1)
```

```
  MH.Estatura         MH.Peso        MM.Estatura          MM.Peso     
 Min.   :1.480   Min.   :56.43   Min.   :1.440   Min.   :37.39  
 1st Qu.:1.610   1st Qu.:68.26   1st Qu.:1.540   1st Qu.:49.35  
 Median :1.650   Median :72.97   Median :1.570   Median :54.48  
 Mean   :1.654   Mean   :72.86   Mean   :1.573   Mean   :55.08  
 3rd Qu.:1.700   3rd Qu.:77.52   3rd Qu.:1.610   3rd Qu.:59.80  
 Max.   :1.800   Max.   :90.49   Max.   :1.740   Max.   :80.87  
```

Y con esa misma matriz podemos ver la correlación que tienen las variables del modelo

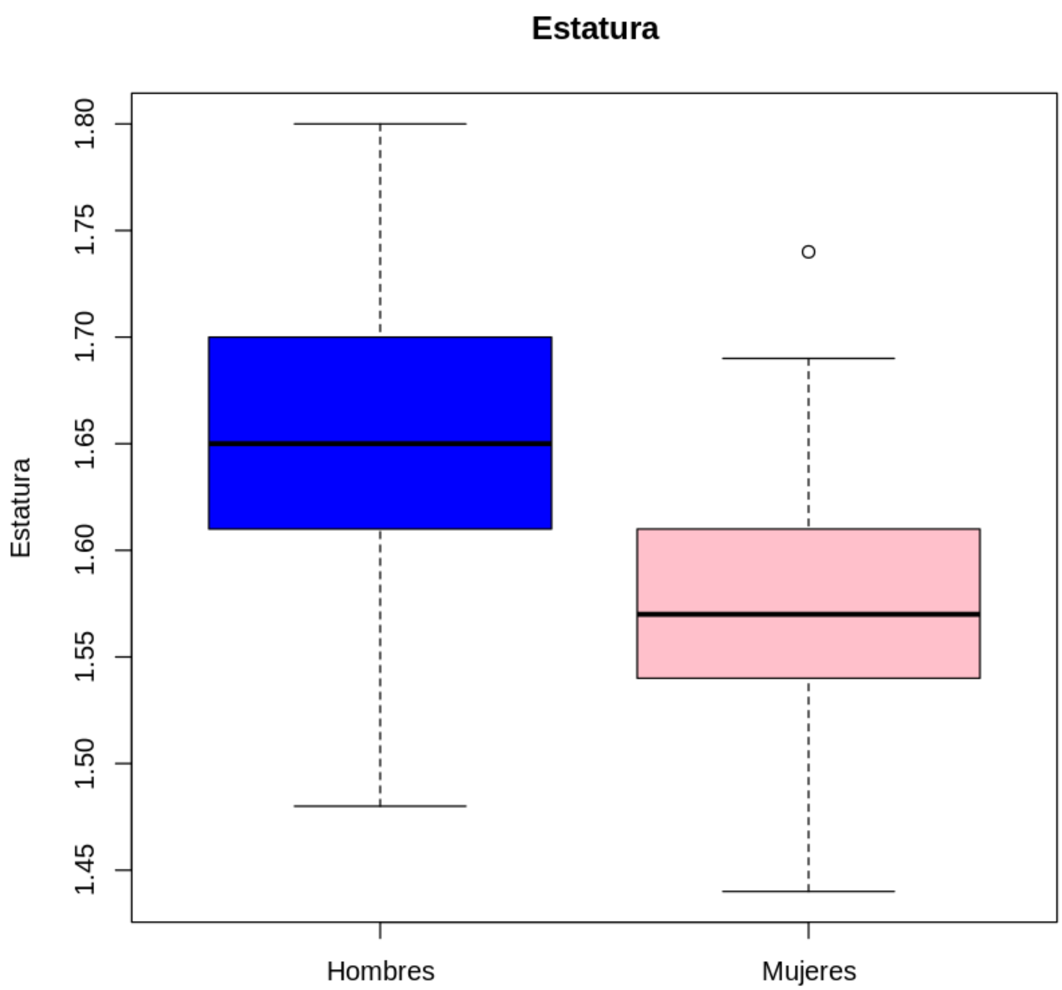
[ ]:
```
cor(M1)
```

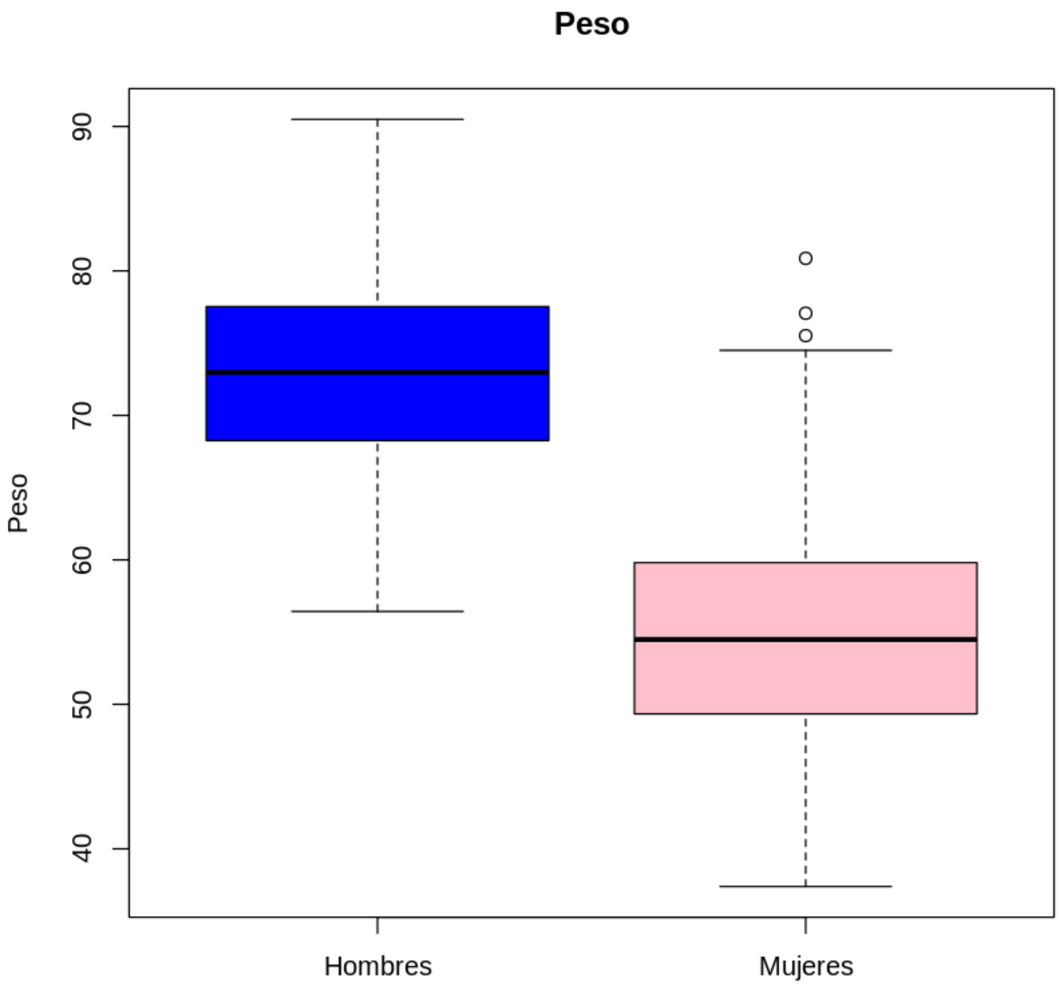
A matrix: 4 × 4 of type dbl

	MH.Estatura	MH.Peso	MM.Estatura	MM.Peso
MH.Estatura	1.0000000000	0.846834792	0.0005540612	0.04724872
MH.Peso	0.8468347920	1.000000000	0.0035132246	0.02154907
MM.Estatura	0.0005540612	0.003513225	1.0000000000	0.52449621
MM.Peso	0.0472487231	0.021549075	0.5244962115	1.00000000

Y tambien, para la descripción de los datos nos podemos fijar en las graficas de caja y pastel.

[ ]:
```
boxplot(M$Estatura~M$Sexo, ylab="Estatura", xlab="", col=c("blue","pink"),␣
 ↪names=c("Hombres", "Mujeres"), main="Estatura")
boxplot(M$Peso~M$Sexo, ylab="Peso",xlab="", names=c("Hombres", "Mujeres"),␣
 ↪col=c("blue","pink"), main="Peso")
```


Estatura
Estatura
1.80
1.75
1.70
1.65
1.60
1.55
1.50
1.45
Hombres
Mujeres



1.2 Creación de los modelos

Por los datos que tenemos podemos probar dos modelos, uno con la interacción de peso y estatura el cual se mostrara primero y lo llamaremos B

[ ]:
```
B = lm(M$Peso~M$Estatura*M$Sexo)
B
summary(B)
```

```
Call:
lm(formula = M$Peso ~ M$Estatura * M$Sexo)
```

```
Coefficients:
       (Intercept)           M$Estatura               M$SexoM  M$Estatura:M$SexoM
            -83.68                94.66                 11.12              -13.51
```

```
Call:
lm(formula = M$Peso ~ M$Estatura * M$Sexo)

Residuals:
     Min       1Q   Median       3Q      Max
-21.3256  -3.1107   0.0204   3.2691  17.9114

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)         -83.685      9.735  -8.597   <2e-16 ***
M$Estatura           94.660      5.882  16.092   <2e-16 ***
M$SexoM              11.124     14.950   0.744    0.457
M$Estatura:M$SexoM  -13.511      9.305  -1.452    0.147
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 5.374 on 436 degrees of freedom
Multiple R-squared:  0.7847,	Adjusted R-squared:  0.7832
F-statistic: 529.7 on 3 and 436 DF,  p-value: < 2.2e-16
```

Podemos ver que no estamos rechazando H_0 para las relacion de estatura y sexo y tampoco la rechazamos con sexo. De todas formas, hay que recordar que no nos podemos deshacer de varias variables a la vez. Podemos eliminar la relacion de estatura y sexo y probamos hacer el modelo solamente con la estatura y el sexo.

Probamos el segundo modelo (A) con la estatura y el peso

[ ]:
```
#Hacemos el modelo lineal

A = lm(M$Peso~M$Estatura+M$Sexo)
A
summary(A)
```

```
Call:
lm(formula = M$Peso ~ M$Estatura + M$Sexo)

Coefficients:
(Intercept)   M$Estatura      M$SexoM
     -74.75        89.26       -10.56
```

```
Call:
lm(formula = M$Peso ~ M$Estatura + M$Sexo)

Residuals:
     Min       1Q   Median       3Q      Max
-21.9505  -3.2491   0.0489   3.2880  17.1243

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -74.7546     7.5555  -9.894   <2e-16 ***
M$Estatura   89.2604     4.5635  19.560   <2e-16 ***
M$SexoM     -10.5645     0.6317 -16.724   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5.381 on 437 degrees of freedom
Multiple R-squared:  0.7837,	Adjusted R-squared:  0.7827
F-statistic: 791.5 on 2 and 437 DF,  p-value: < 2.2e-16
```

Ahora, nos podemso dar cuenta que todas las variables son significativas, por lo que podemos continuar el análisis con este modelo.

[ ]:
```r
b0 = A$coefficients[1]
b1 = A$coefficients[2]
b2 = A$coefficients[3]

cat("Peso = ", b0, "+", b1, " Estatura ", b2, "SexoM")
```

```
Peso =  -74.7546 + 89.26035  Estatura  -10.56447 SexoM
```

No hay mucho que podamos interpretar de β_0 ya que no tiene sentido que exista una persona con estatura 0 y peso negativo.

Por otro lado, para β_1 podemos ver que es la tasa de cambio de que tanto aumenta el peso conforme aumentamos la estatura.

1.3 Verificación del modelo

- Significacncia global
- Significancia individual
- Porcentaje de variacion explicada por el modelo

[ ]:
```r
#verificacion dle modelo
summary(A)
```

```
Call:
lm(formula = M$Peso ~ M$Estatura + M$Sexo)
```

```
Residuals:
     Min       1Q   Median       3Q      Max
-21.9505  -3.2491   0.0489   3.2880  17.1243

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -74.7546     7.5555  -9.894   <2e-16 ***
M$Estatura   89.2604     4.5635  19.560   <2e-16 ***
M$SexoM     -10.5645     0.6317 -16.724   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5.381 on 437 degrees of freedom
Multiple R-squared:  0.7837,        Adjusted R-squared:  0.7827
F-statistic: 791.5 on 2 and 437 DF,  p-value: < 2.2e-16
```

Significancia global Podemos ver el que el modelo es muy significativo porque tiene un valor de 791 en la significancia (f), el cuál se encuentra myuy lejos de 1 (poco significativo).

Significancia individual Además, tenemos un valor grande para todos los valores de t de las variables y también tienen un valor p muy pequeño.

Variación epxlicada por el modelo Finalmente, tenemos un coeficiente de determinación de 0.7828, lo que nos dice que el modelo puede explicar bastante bien la mayoría de los datos.

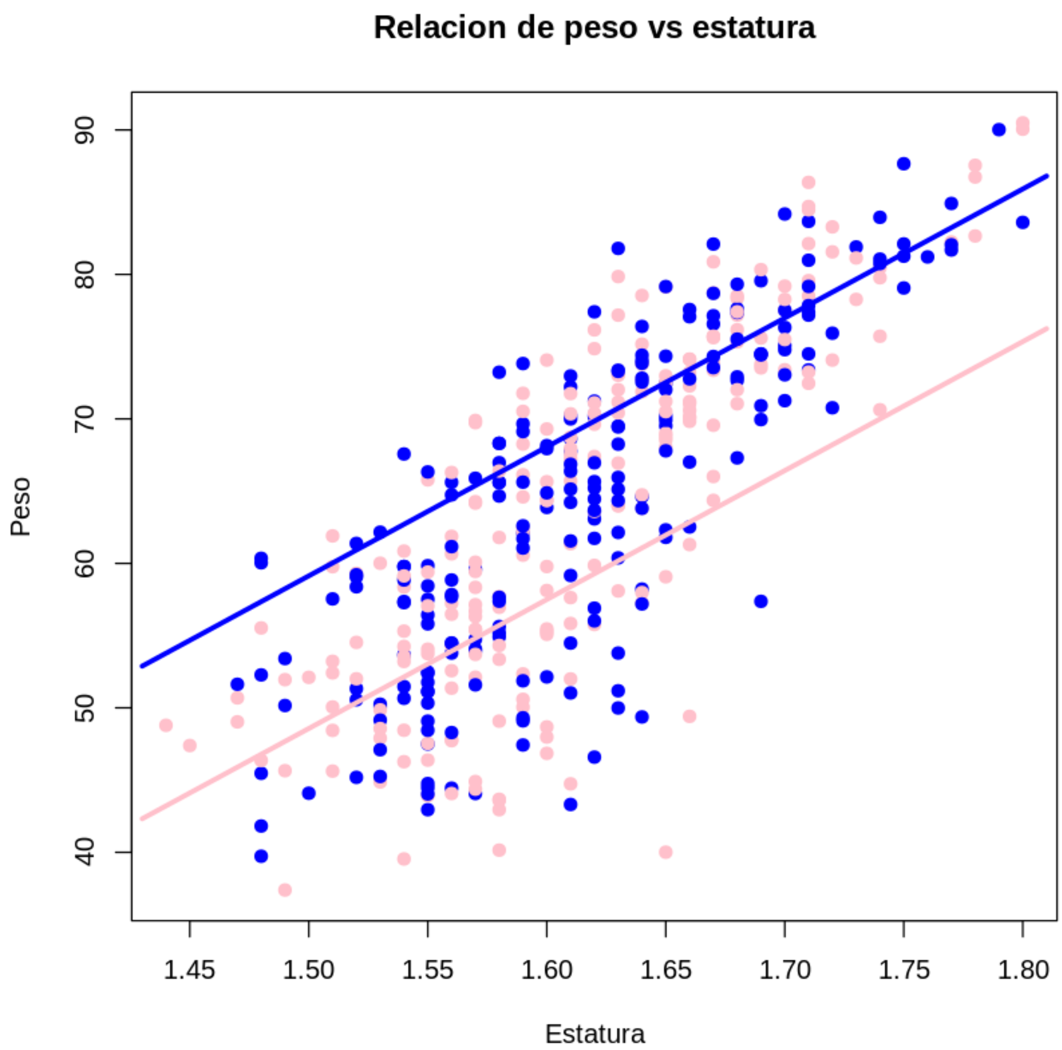
Fianlmente, podemos gráficar los dos modelos, el de mujeres en rosa y el de hombres en azul. Además, en la gráfica se muestran los datos correspondientes a hombres y mujeres del mismo color que las lineas de los modelos.

[ ]:
```
#Plot para mujeres (SexoM = 1)
cat("Función del modelo para mujeres \n")
cat("Peso = ", b0+b2, " + ", b1, "Estatura")

#Plot para mujeres (SexoM = 0)
cat("\nFunción del modelo para hombres\n")
cat("Peso = ", b0, " + ", b1, "Estatura")
```

```
Función del modelo para mujeres
Peso =  -85.31907  +  89.26035 Estatura
Función del modelo para hombres
Peso =  -74.7546  +  89.26035 Estatura
```

[ ]:
```
#Gráfica
Ym = function(x){b0+b2+b1*x}
Yh = function(x){b0+b1*x}

colores = c("blue", "pink")
plot(M$Estatura, M$Peso, col=colores, pch=19, ylab="Peso", xlab="Estatura",␣
 ↪main="Relacion de peso vs estatura")
```

```
x = seq(1.43, 1.81, 0.01)
lines(x, Ym(x), col="pink", lwd=3)
lines(x, Yh(x), col="blue", lwd=3)
```



1.4 Validez del modelo

Para verificar que el modelo es apropiado para el conjunto de datos, podemos comprobar los siguientes puntos:

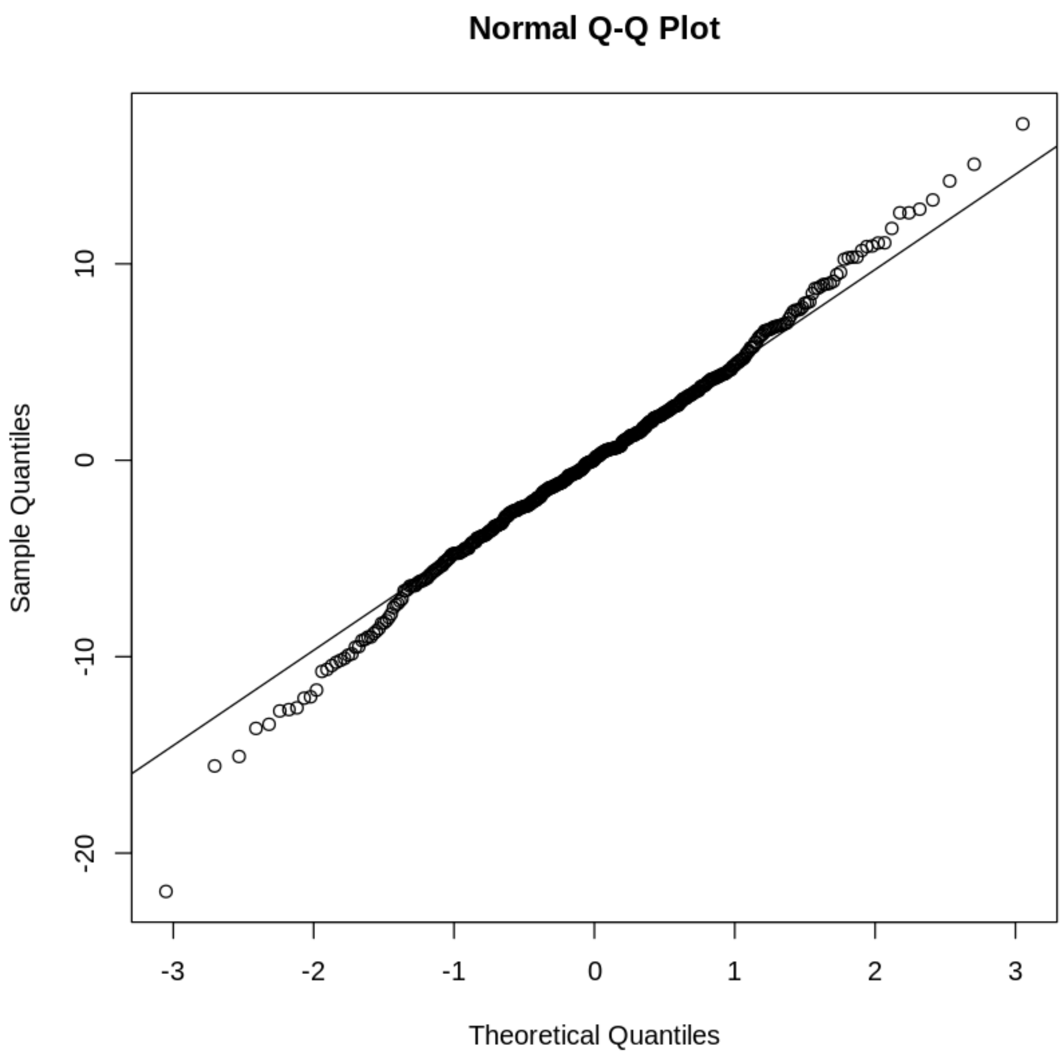
- Normalidad de los residuos
- Verificación de media cero
- Homocedasticidad e independencia

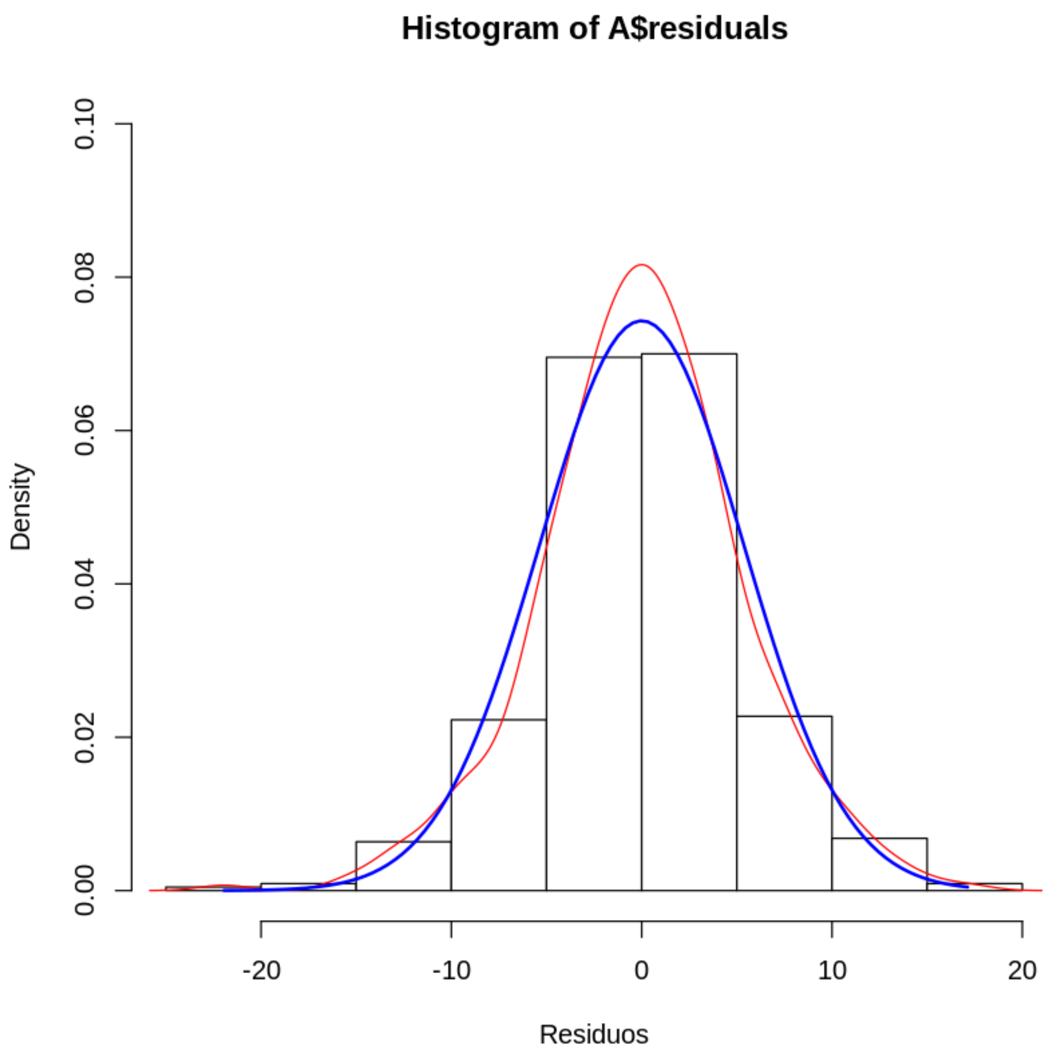
1.4.1 Verificación de normalidad

Primero, podemos verificar si los residuos tienen un comportamiento normal o no.

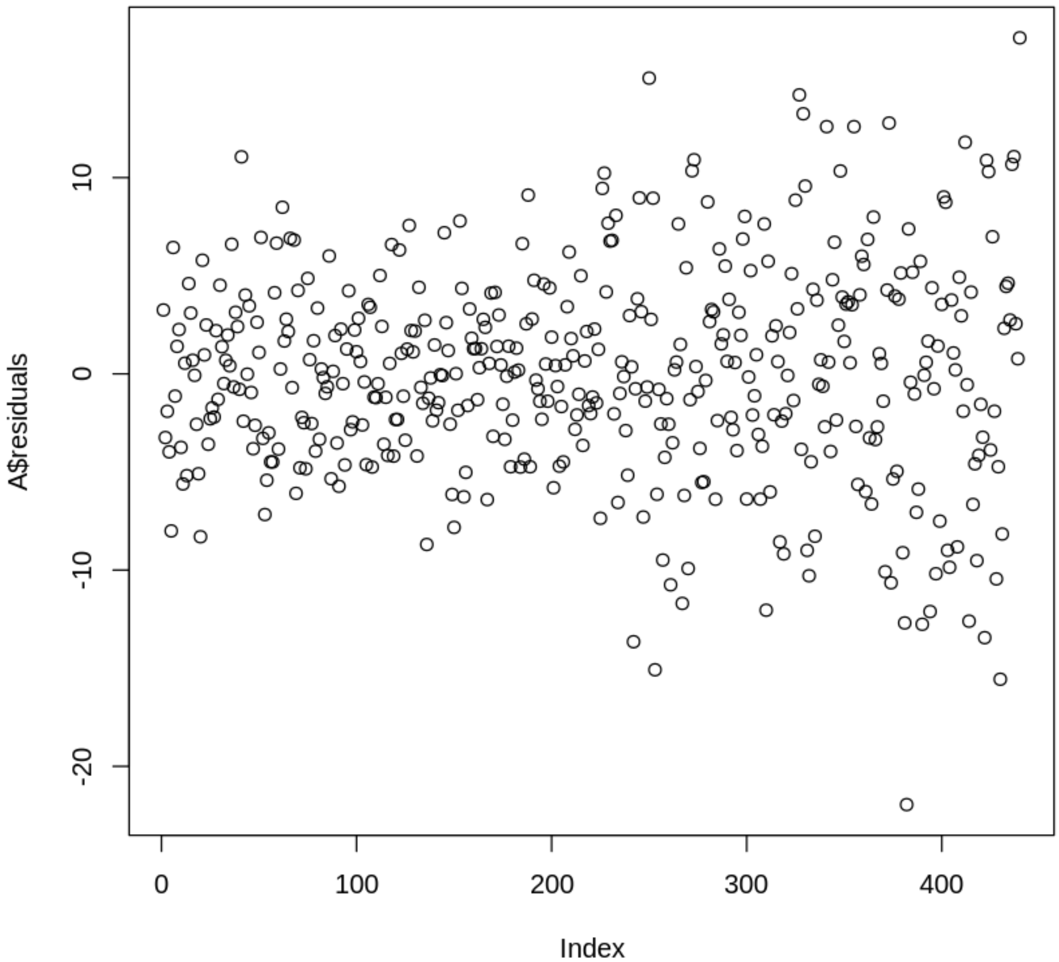
[ ]:
```
shapiro.test(A$residuals)

#Gráficas auxiliares:
qqnorm(A$residuals)
qqline(A$residuals)

hist(A$residuals,freq=FALSE, ylim=c(0,0.1), xlab="Residuos", col=0)
lines(density(A$residuals),col="red")
curve(dnorm(x,mean=mean(A$residuals),sd=sd(A$residuals)),
 ↪from=min(A$residuals),to=max(A$residuals), add=TRUE, col="blue",lwd=2)

plot(A$residuals)
```

```
        Shapiro-Wilk normality test

data:  A$residuals
W = 0.99337, p-value = 0.0501
```


Normal Q-Q Plot
Sample Quantiles
Theoretical Quantiles
-20
-10
0
10
-3
-2
-1
0
1
2
3



Histogram of A$residuals
Density
0.00
0.02
0.04
0.06
0.08
0.10
-20
-10
0
10
20
Residuos



Podemos ver en las gráficas de arriba, que el comportamiento de los residuos es casi normal. No tiene la distribución ideal, pero es lo suficiente parecido como para decir que el modelo es bueno. Primero, en el histograma podemos ver que la linea azul es la linea ideal y la roja es la linea del modelo, por lo que son muy parecidos.

Después, si nos fijamos en el scatter plot podemos ver que los residuos tienen una distribución casi uniforme. Cuando nos acercamos a la derecha podemso ver que aumenta un poco la separación de los puntos.

1.4.2 Verificación de media cero

Para hacer la verificacion de que los residuos tengan media cero podemos correr el siguiente codigo

[ ]:
```
t.test(A$residuals)
```

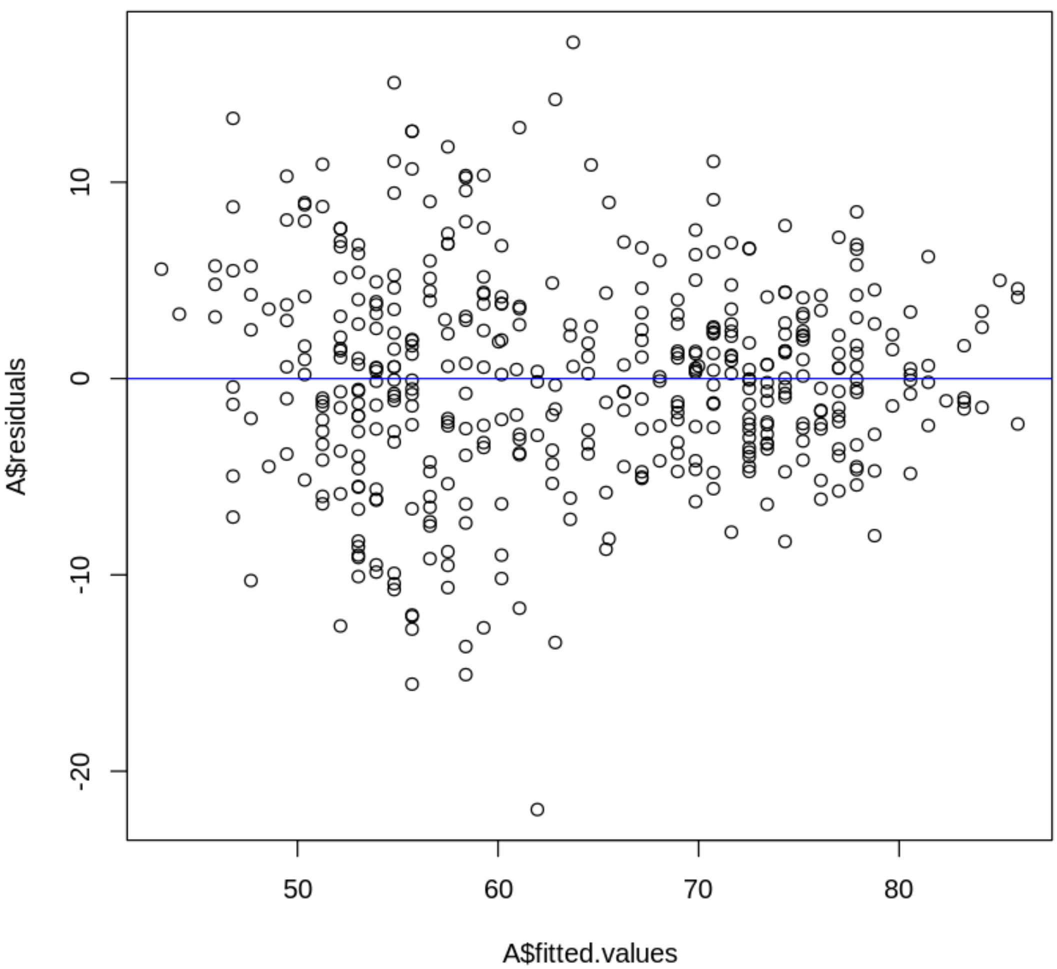
```
        One Sample t-test

data:  A$residuals
t = -3.3793e-16, df = 439, p-value = 1
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 -0.5029859  0.5029859
sample estimates:
    mean of x
-8.648385e-17
```

El resultado del test para la t de student nos dice que la media no es 0, por lo tanto el resultado del modelo puede no se completamente confiable. Esto es lo mismo que veiamos arriba en las gráficas, donde los residuos no parecen ser normales.

1.4.3 Homocedasticidad e independencia

Por ultimo podemos comprobar si el conjunto de residuos presentan homocedasticidad e independencia

[ ]:
```
plot(A$fitted.values,A$residuals)
abline(h=0, col="blue")
```



Viendo la gráfica podemos decir que los datos si presentan un comportamiento de homocedasticidad e independencia. Como lo mencionabamos anteriormente, la distribución no es ideal pero tampoco parece que los datos sigan un comportamiento espedíficio.

1.4.4 Pruebas de hipótesis

Paso 1 Definicion de la hipotesis

Definir las hipótesis

H_0 : Los datos provienen de una poblacion normal

H_1 : Los datos no provienen de una población normal

Paso 2 Regla de decisión

$\alpha = 0.03$

Se rechaza H_0 si valor p $< \alpha$

Paso 3 Análisis del resultado

Sabemos que el valor p es de 0.39 con la prueba de Anderson Darwin y no es menor que α, por lo tanto, no rechazamos H_0.

[ ]:
```
ad.test(A$residuals)
```

```
        Anderson-Darling normality test

data:  A$residuals
A = 0.79651, p-value = 0.03879
```

Paso 4 Conclusiones

- Como valor p (0.03879) es menor que α (0.03), entonces no rechazamos H_0

En el contexto del problema esto significa que no rechazamos la hipótesis, por lo tanto podemos decir que los datos provienen de una población normal.

1.5 Conclusiones

Como pudimos ver con esta actividad, de los dos modelos que hicimos, solamente el que no establecia una relación entre la estatura y el peso nos resulto más útil.

Las variables que fueron importantes para realizar el modelo fueron la estatura, el sexo y el peso.

Podemos ver que los modelos son paralelos, por lo que solamente cambia su intersección, pero la tasa de cambio que tienen es la misma.

Con las pruebas que hicimos nos damos cuenta que el modelo es bueno, aunque puede haber una pequeña variable que no estemos considerando, ya que los residuos no son completamente normales.

1.6 Intervalos de confianza

Los intervalos de confianza nos permite dar un rango sobre el que podremos predecir y asegurar que los resultados tendrán un porcentaje de ser correctos.

Por ejemplo, para el modelo lineal que se había planteado más arriba en el notebook, las betas que obtuvimos no deberían ser precisas. Esto quiere decir que los valores que obtuvimos no son exactos. Para ver el rango en el que podrían estar ejecutamos la siguiente linea.

[ ]:
```
confint(A, level= 0.97)

#Esto es sobre las betas, son intervalos de confianza sobre que tan seguro␣
↪estamos de los resultados de como interactuan las variabels
```

A matrix: 3 × 2 of type dbl

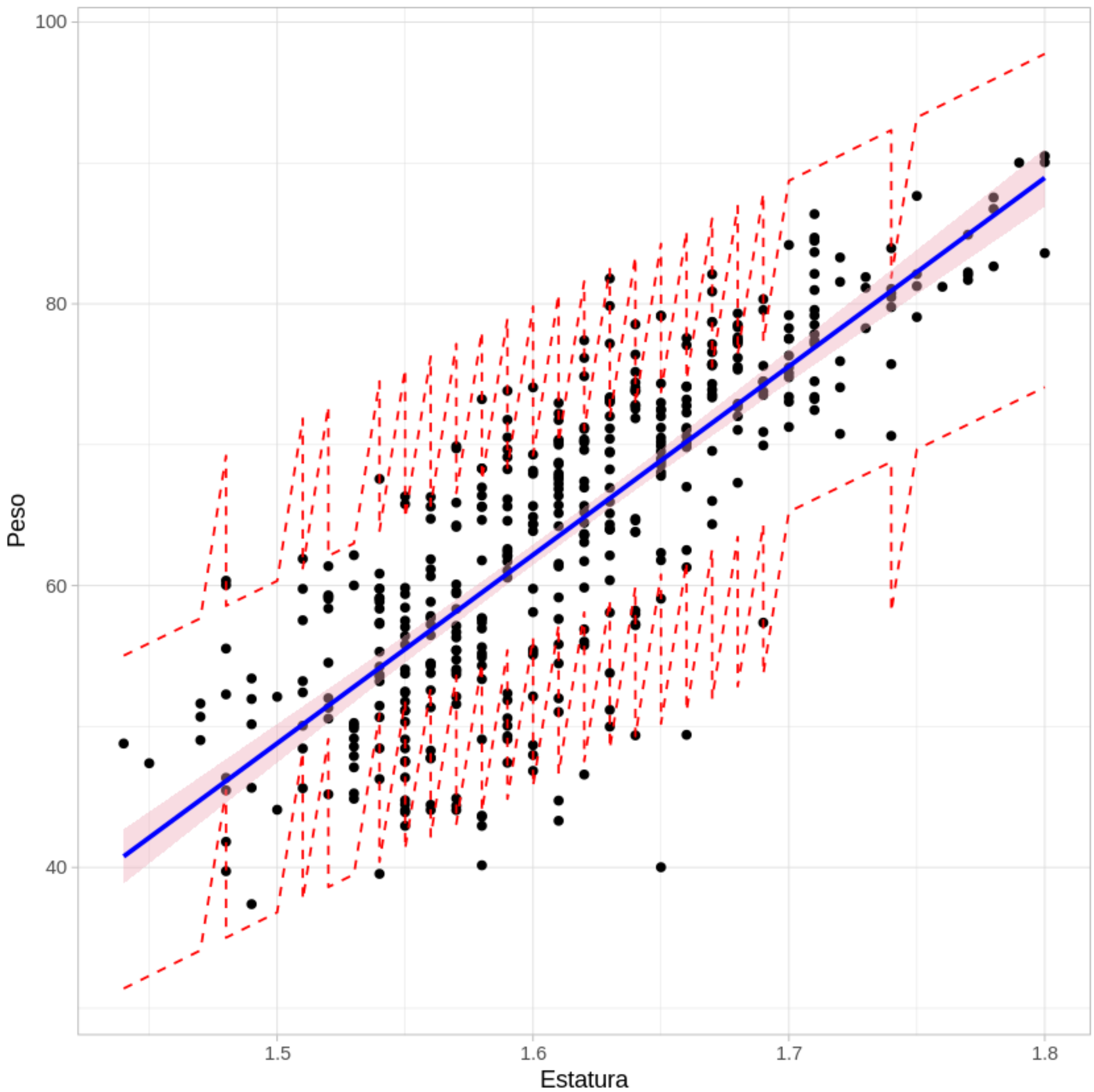
	1.5 %	98.5 %
(Intercept)	-91.20451	-58.304689
M$Estatura	79.32465	99.196052
M$SexoM	-11.93983	-9.189113

Podemos decir que los rangos para las betas, se encuentran con un 97% de confianza en los resultados de la tabla de arriba.

Ahora, podemos calcular y graficar los intervalos de confianza para las predicciones que hará el modelo.

[ ]:
```
#Intervalos de confianza y prediccion para y
Ip=predict(object=A,interval="prediction",level=0.97)
datos1=cbind(M,Ip)
library(ggplot2)
ggplot(datos1,aes(x=Estatura,y=Peso))+
geom_point()+
geom_line(aes(y=lwr), color="red", linetype="dashed")+
geom_line(aes(y=upr), color="red", linetype="dashed")+
geom_smooth(method=lm, formula=y~x, se=TRUE, level=0.97, col="blue",
  ↪fill="pink2")+
theme_light()
```

```
Warning message in predict.lm(object = A, interval = "prediction", level =
0.97):
"predictions on current data refer to _future_ responses
"
```



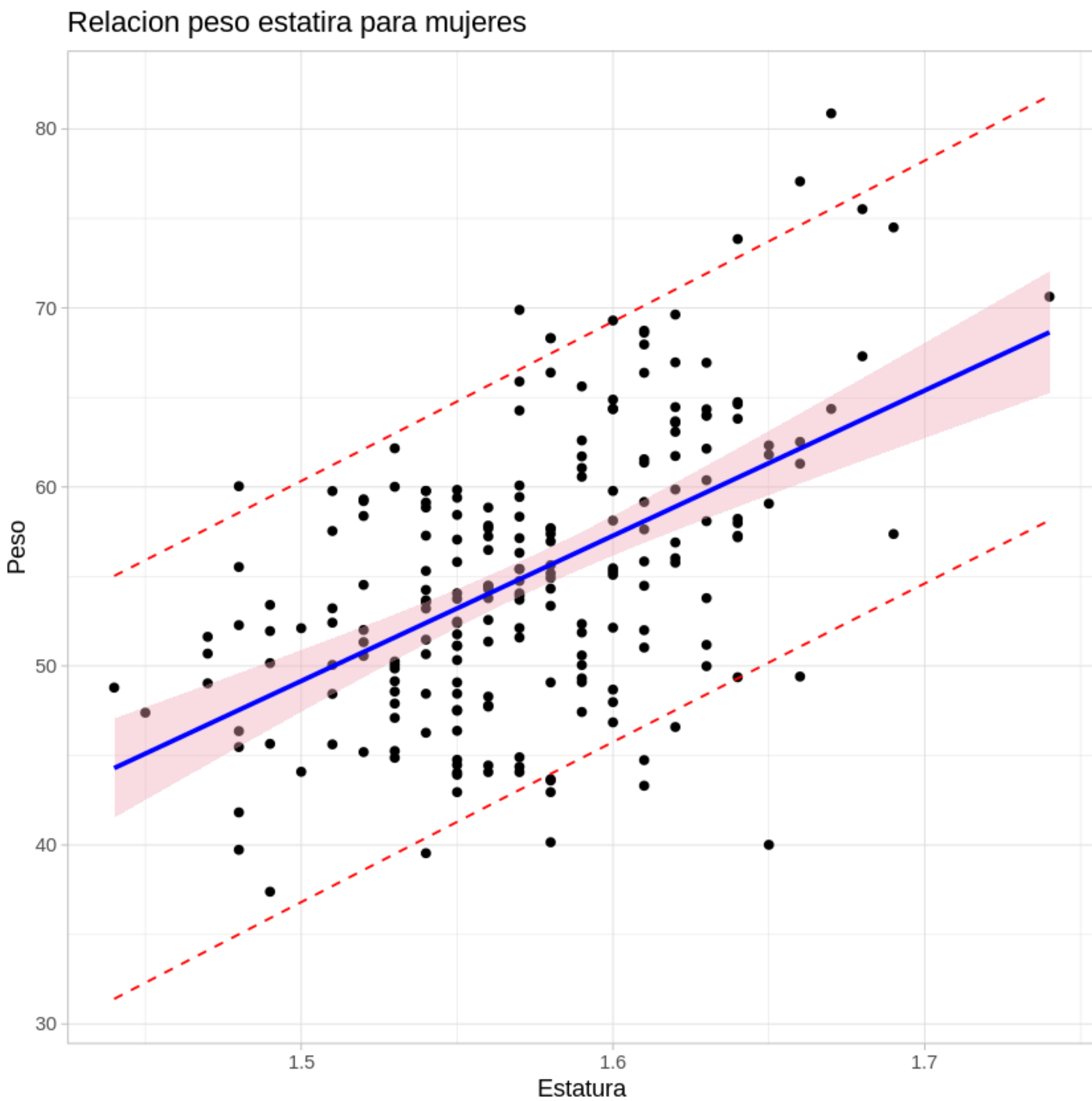
En la gráfica de arriba, la línea azul es el modelo, la sombre roja es el intervalo de confianza y las líneas punteadas rojas son los intervalos de predicción. Pero vemos que tiene un comportamiento extraño pero constante. Lo que está pasando es que en la gráfica no hay una separación de los datos por sexo.

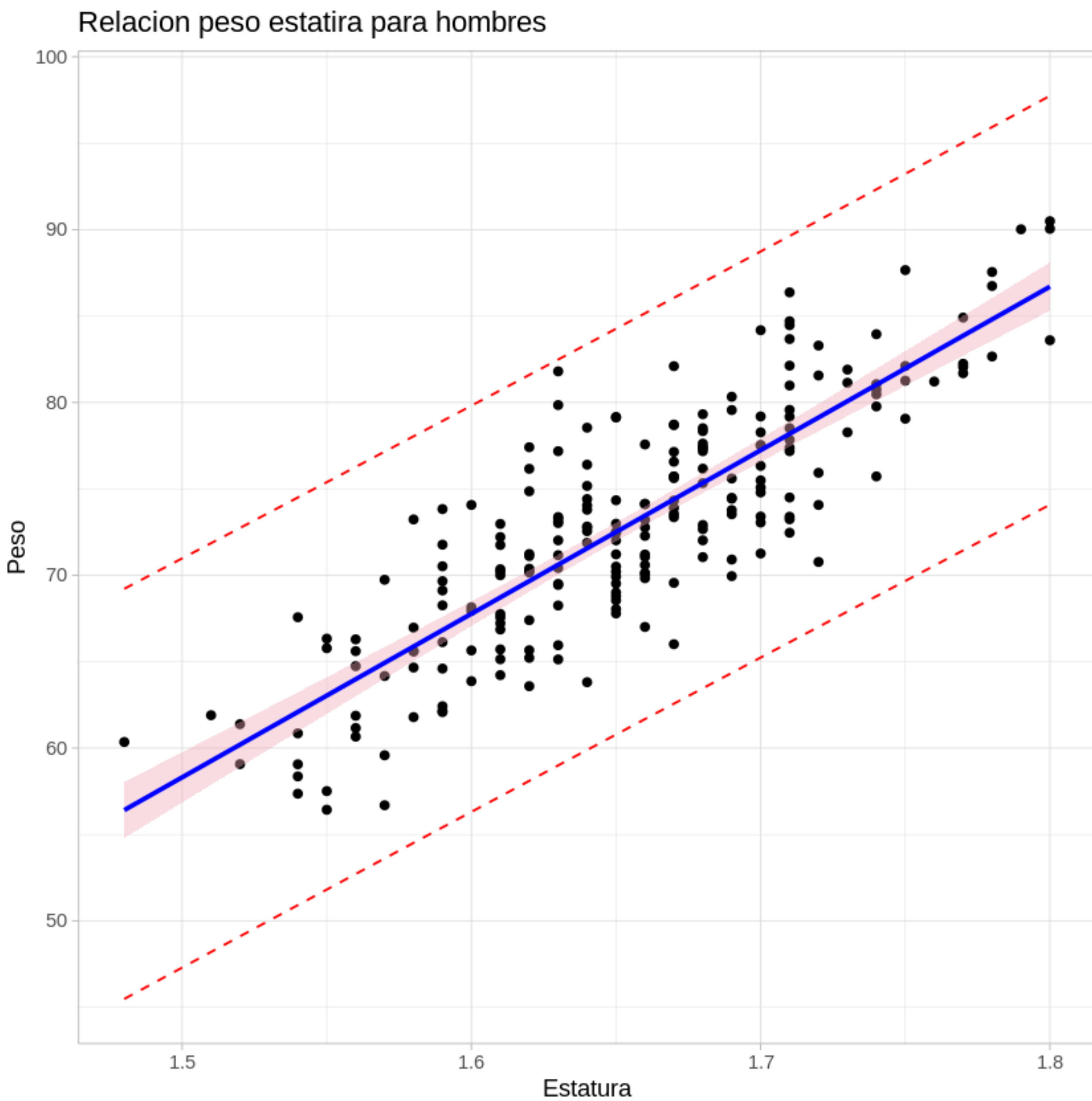
Pero si separamos los datos obtenemos esto.

[ ]:
```
#Intervalos de confianza y prediccion para y
Ip=predict(object=A,interval="prediction",level=0.97)
M2=cbind(M,Ip)
M2m = subset(M2, Sexo == "M")
M2h = subset(M2, Sexo == "H")

library(ggplot2)
```

```
ggplot(M2m,aes(x=Estatura,y=Peso))+
ggtitle("Relacion peso estatira para mujeres")+
geom_point()+
geom_line(aes(y=lwr), color="red", linetype="dashed")+
geom_line(aes(y=upr), color="red", linetype="dashed")+
geom_smooth(method=lm, formula=y~x, se=TRUE, level=0.97, col="blue", 
 ↪fill="pink2")+
theme_light()

ggplot(M2h,aes(x=Estatura,y=Peso))+
ggtitle("Relacion peso estatira para hombres")+
geom_point()+
geom_line(aes(y=lwr), color="red", linetype="dashed")+
geom_line(aes(y=upr), color="red", linetype="dashed")+
geom_smooth(method=lm, formula=y~x, se=TRUE, level=0.97, col="blue", 
 ↪fill="pink2")+
theme_light()
```

```
Warning message in predict.lm(object = A, interval = "prediction", level =
0.97):
"predictions on current data refer to _future_ responses
"
```


Relacion peso estatira para mujeres
80
70
60
50
40
30
Peso
1.5
1.6
1.7
Estatura



Ahora, podemos ver perfectamente los intervalos de predicción y de confianza de nuestro modelo para hombres y mujeres.