Regresión lineal simple y múltiple

Ejemplo de solución ejercicio prático N°9

Enunciado

Un estudio recolectó medidas anatómicas de 247 hombres y 260 mujeres (Heinz et al., 2003). El estudio incluyó nueve mediciones del esqueleto (ocho diámetros y una profundidad de hueso a hueso) y doce mediciones de grosor (diámetros de circunferencias) que incluyen el tejido.

La siguiente tabla detalla las variables registradas en este estudio:

Variable	Descripción	Unidad
Biacromial.diameter	**Diámetro biacromial (a la altura de los hombros)**	**cm**
Biiliac.diameter	**Diámetro biiliaco (a la altura de la pelvis)**	**cm**
Bitrochanteric.diameter	**Diámetro bitrocantéreo (a la altura de las caderas)**	**cm**
Chest.depth	**Profundidad del pecho (entre la espina y el esternón a la altura de los pezones)**	**cm**
Chest.diameter	**Diámetro del pecho (a la altura de los pezones)**	**cm**
Elbows.diameter	**Suma de los diámetros de los codos**	**cm**
Wrists.diameter	**Suma de los diámetros de las muñecas**	**cm**
Knees.diameter	**Suma de los diámetros de las rodillas**	**cm**
Ankles.diameter	**Suma de los diámetros de los tobillos**	**cm**
Shoulder.Girth	**Grosor de los hombros sobre los músculos deltoides**	**cm**
Chest.Girth	**Grosor del pecho, sobre tejido mamario en mujeres y a la altura de los pezones en varones**	**cm**
Waist.Girth	**Grosor a la altura de la cintura**	**cm**
Navel.Girth	**Grosor a la altura del ombligo**	**cm**
Hip.Girth	**Grosor a la altura de las caderas**	**cm**
Thigh.Girth	**Grosor promedio de ambos muslos bajo el pliegue del glúteo**	**cm**
Bicep.Girth	**Grosor promedio de ambos bíceps, brazos flectados**	**cm**

Variable	Descripción	Unidad
Forearm.Girth	**Grosor promedio de ambos antebrazos, brazos extendidos palmas hacia arriba**	**cm**
Knee.Girth	**Grosor promedio de ambas rodillas, posición levemente flectada, medición arriba de la rótula**	**cm**
Calf.Maximum.Girth	**Grosor promedio de la parte más ancha de ambas pantorrillas**	**cm**
Ankle.Minimum.Girth	**Grosor promedio de la parte más delgada de ambos tobillos**	**cm**
Wrist.Minimum.Girth	**Grosor promedio de la parte más delgada de ambas muñecas**	**cm**
Age	**Edad**	**Años**
Weight	**Peso**	**Kg**
Height	**Estatura**	**cm**
Gender	**Género**	**1: hombre, 0: mujer**

Con estos datos se pide construir un modelo de regresión lineal múltiple para predecir una *variable respuesta*, de acuerdo con las siguientes instrucciones:

1. **Definir la semilla a utilizar, que corresponde a los últimos cuatro dígitos del RUN (sin considerar el dígito verificador) del integrante de menor edad del equipo.**
2. **Seleccionar una muestra de 100 mujeres (si la semilla es un número par) o 100 hombres (si la semilla es impar), y separar 70 casos para trabajar en la construcción de modelos y 30 para su evaluación en datos no vistos.**
3. **Seleccionar de forma aleatoria ocho posibles variables predictoras.**
4. **Seleccionar, de las otras variables, una que el equipo considere que podría ser útil para predecir la variable respuesta, justificando bien esta selección.**
5. **Usando el entorno R y paquetes estándares, construir un modelo de regresión lineal simple con el predictor seleccionado en el paso anterior.**
6. **Usando herramientas estándares para la exploración de modelos del entorno R, buscar entre dos y cinco predictores de entre las variables seleccionadas al azar en el punto 3, para agregar al modelo de regresión lineal simple obtenido en el paso 5.**
7. **Evaluar la bondad de ajuste (incluyendo el análisis de casos atípicos y casos influyentes) y la generalidad (condiciones para RLM) de los modelos y "arreglarlos" en caso de que presenten algún problema.**
8. **Evaluar el poder predictivo de los modelos en datos no utilizados para construirlos.**

Comencemos incluyendo los paquetes que usaremos en este script.

```
library(car)
library(dplyr)
library(ggpubr)
library(psych)
```

Obtengamos los datos en formato ancho.

```
src_dir <- "~/Downloads"
# src_dir <- "C:/Users/ProfeJLJara/Downloads"
src_basename <- "EP09 Datos.csv"
src_file <- file.path(src_dir, src_basename)

datos <- read.csv2(file = src_file, stringsAsFactors = TRUE)
```

Obtengamos la muestra y separémosla en los conjuntos de entrenamiento y prueba, teniendo el cuidado de fijar una semilla para su reproductibilidad.

```
set.seed(1111)
datos <- datos |> filter(Gender == 1) |> select(-Gender) |> sample_n(100, replace = FALSE)
datos_entren <- datos[1:70, ]
datos_prueba <- datos[71:100, ]
```

Para este script de ejemplo, usaremos como variable respuesta **los diámetros de las rodillas** (`Knees.diameter`).

Corresponde seleccionar al azar 8 posibles variables predictoras de este conjunto, teniendo cuidado de no seleccionar la variable de respuesta.

```
nombre_respuesta <- "Knees.diameter"
variables <- colnames(datos_entren)
i_respuesta <- which(variables == nombre_respuesta)
predictores <- sample(variables[-i_respuesta], 8, replace = FALSE)

cat("Predictores seleccionados al azar:\n")
cat(paste(predictores, collapse = "\n"))
```

```
Predictores seleccionados al azar:
Ankles.diameter
Calf.Maximum.Girth
Waist.Girth
Bitrochanteric.diameter
Ankle.Minimum.Girth
Hip.Girth
Biiliac.diameter
Age
```

Estos son los predictores seleccionados al azar para ser considerados en el modelo de regresión lineal múltiple que vamos a construir.

Para seleccionar una de las variables restantes para construir un modelo de regresión lineal simple (RLS), vamos a evaluar su correlación con la variable respuesta.

```
datos_resto <- datos_entren |> select(!all_of(predictores))
i_respuesta_resto <- which(colnames(datos_resto) == nombre_respuesta)
correlacion <- cor(datos_resto[-i_respuesta_resto], y = datos_resto[[nombre_respuesta]])

cat("Correlación con la variable respuesta:\n")
print(correlacion)
```

```
Correlación con la variable respuesta:
                        [,1]
Biacromial.diameter 0.4990345
Chest.depth         0.1833036
Chest.diameter      0.5305838
Elbows.diameter     0.5616106
Wrists.diameter     0.6259288
Shoulder.Girth      0.4762599
Chest.Girth         0.3594377
Navel.Girth         0.2803497
Thigh.Girth         0.5622237
Bicep.Girth         0.3890932
Forearm.Girth       0.5044949
Knee.Girth          0.6105291
Wrist.Minimum.Girth 0.4510171
Weight              0.6174914
Height              0.4652410
```

Asumiendo que el mejor predictor para un modelo de RLS es aquella variable con mayor correlación (directa o inversa) con la variable de respuesta, podemos determinar fácilmente nuestro predictor.

```
i_mejor <- which(correlacion == max(abs(correlacion)))
predictor <- rownames(correlacion)[i_mejor]

cat("Variable más correlacionada con la variable respuesta:", predictor, "\n")
```

```
Variable más correlacionada con la variable respuesta: Wrists.diameter
```

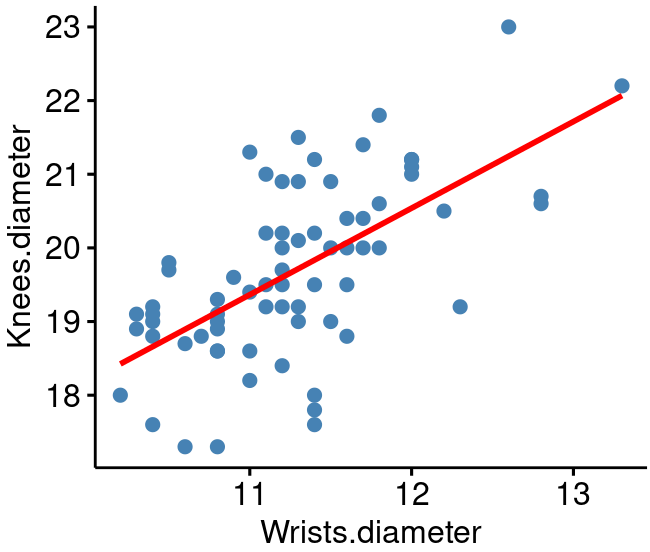
Filtramos para quedarnos con las variables relevantes.

```
datos_entren <- datos_entren |>
  select(all_of(c(predictor, predictores, nombre_respuesta)))
```

Regresión lineal simple

Demos entonces una mirada a los datos.

```
p1 <- ggscatter(datos_entren, x = predictor, y = nombre_respuesta,
                color = "steelblue", fill = "steelblue",
                add = "reg.line", add.params = list(color = "red"))
print(p1)
```



Este gráfico de dispersión parece mostrar una relación lineal positiva entre las variables.

Obtengamos el modelo de regresión lineal simple.

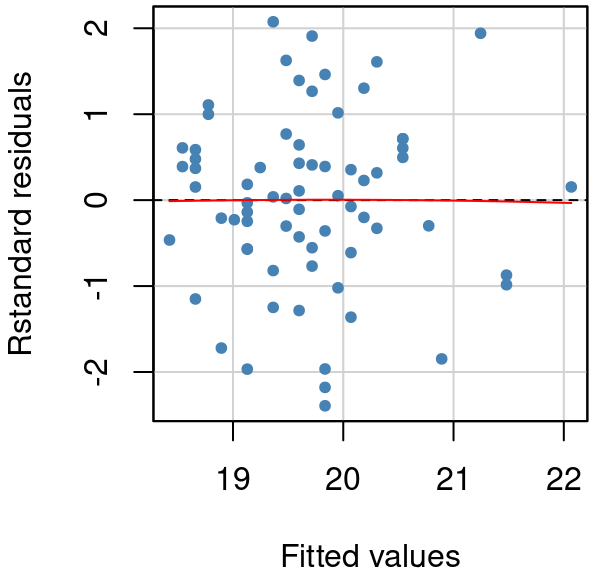
```
fmla <- formula(paste(nombre_respuesta, predictor, sep = " ~ "))
rls <- lm(fmla, data = datos_entren)

cat("Modelo de regresión lineal simple:\n")
print(summary(rls))
```

```
Modelo de regresión lineal simple:

Call:
lm(formula = fmla, data = datos_entren)

Residuals:
     Min       1Q   Median       3Q      Max 
-2.23465 -0.52636  0.04165  0.55385  1.93549 

Coefficients:
                Estimate Std. Error t value Pr(>|t|)    
(Intercept)       6.4357     2.0056   3.209  0.00203 ** 
Wrists.diameter   1.1754     0.1776   6.618 6.86e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.9409 on 68 degrees of freedom
Multiple R-squared:  0.3918,    Adjusted R-squared:  0.3828 
F-statistic:  43.8 on 1 and 68 DF,  p-value: 6.86e-09
```

Podemos ver que el modelo de RLS obtenido explica alrededor del 40% de la varianza en los datos y que es significativamente mejor que simplemente usar la media ($F(1,68) = 43{,}8$; $p < 0{,}001$).

Revisemos los gráficos de los residuos que genera el modelo.

```
cat("Prueba de curvatura:\n")
residualPlots(rls, type = "rstandard", terms = ~ 1, col = "steelblue", pch = 20, col.quad = "re
d")
```

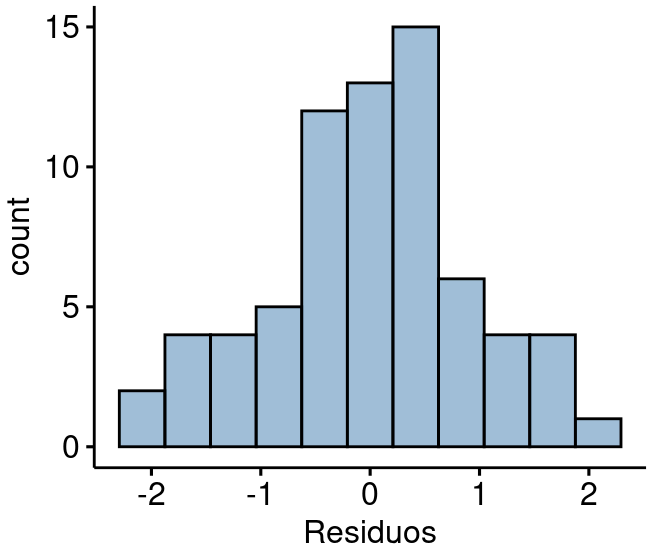



```
Prueba de curvatura:
           Test stat Pr(>|Test stat|)
Tukey test   -0.0499           0.9602
```

Vemos que no hay un patrón identificable y que los residuos parecen repartirse de forma aleatoria arriba y abajo de la línea de regresión. La prueba de curvatura resultan no significativas, por lo que no podemos descartar que el diámetro de las muñecas se relaciona linealmente con el diámetro de las rodillas.

Si tuviéramos dudas, podemos confirmar la normalidad de los residuos con un histograma y usando una prueba de normalidad.

```
h_res <- gghistogram(data.frame(Residuos = resid(rls)), x = "Residuos", bins = 11,
                     fill = "steelblue")
print(h_res)
```




```
sw_res <- shapiro.test(resid(rls))
cat("Test de normalidad de los residuos del modelo de RLS:")
print(sw_res)
```

```
Test de normalidad de los residuos del modelo de RLS:
    Shapiro-Wilk normality test

data:  resid(rls)
W = 0.98444, p-value = 0.5373
```

Si bien se observa cierta asimetría, no hay evidencia suficiente para descartar que los residuos siguen un comportamiento normal.

Confirmemos que la varianza de los residuos se mantienen constante.

```
cat("Prueba de varianza del error no constante:\n")
ncvTest(rls)
```

```
Prueba de varianza del error no constante:
Non-constant Variance Score Test
Variance formula: ~ fitted.values
Chisquare = 0.7204959, Df = 1, p = 0.39598
```

No se puede descartar entonces que los residuos cumplan con la condición de homocedasticidad ($\chi(1) = 0{,}720$; $p = 0{,}396$).

Revisemos que los residuos se comportan de manera independiente como siguiere su gráfico.

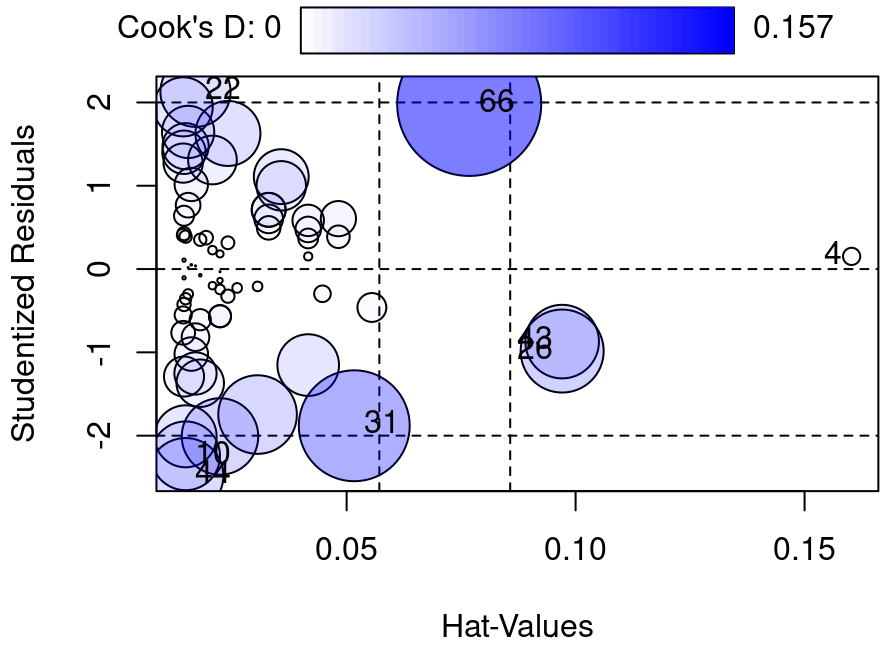
```
cat("Independencia de los residuos\n")
print(durbinWatsonTest(rls))
```

```
Independencia de los residuos
 lag Autocorrelation D-W Statistic p-value
   1    -0.003277224      2.003726   0.972
 Alternative hypothesis: rho != 0
```

Confirmamos que no es posible descartar que la condición de independencia no se esté cumpliendo en este modelo ($\text{D-W} = 2{,}004$; $p = 0{,}972$).

Evaluemos ahora las estadísticas de influencia del modelo de RLS obtenido.

```
cat("Rango para 95% de los residuos studentizados: ")
cat("[", round(qt(0.05/2, nrow(datos_entren) - length(coef(rls)) - 1), 3), ", ", sep = "")
cat(round(qt(1-0.05/2, nrow(datos_entren) - length(coef(rls)) - 1), 3), "]\n", sep = "")
cat("Límite del apalancamiento:", round(2 * mean(hatvalues(rls)), 3), "\n")
cat("Límite de la distancia de Cook:", round(3 * mean(cooks.distance(rls)), 3), "\n")

rls_inf <- influencePlot(rls, id = list(n = 3))
```

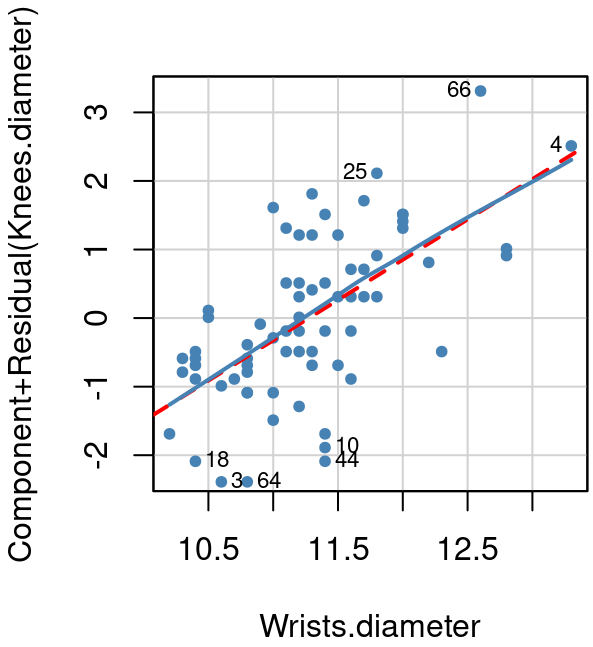



```
cat("\nCasos notorios para el modelo de RLS:\n")
print(rls_inf)
```

```
Rango para 95% de los residuos studentizados: [-1.996, 1.996]
Límite del apalancamiento: 0.057
Límite de la distancia de Cook: 0.041

Casos notorios para el modelo de RLS:
      StudRes        Hat       CookD
4   0.1522091 0.16027504 0.002243178
10 -2.2423594 0.01483604 0.035743448
22  2.1279933 0.01699401 0.037211819
26 -0.9842297 0.09706282 0.052090459
31 -1.8813672 0.05166403 0.092943592
43 -0.8710547 0.09706282 0.040926056
44 -2.4820614 0.01483604 0.043115757
66  1.9826195 0.07676569 0.156667225
```

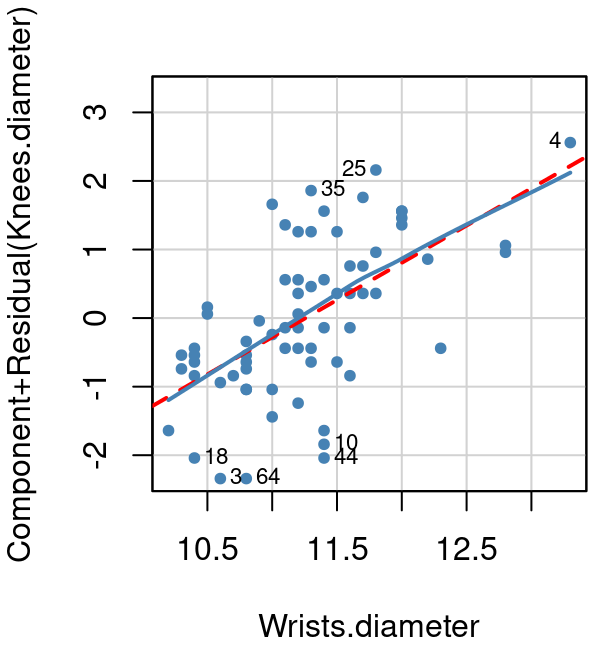
El procedimiento detecta 8 casos que podrían estar influyendo excesivamente en los coeficientes del modelo de RLS obtenido. Revisemos si podemos identificar si estos casos potencialmente problemáticos están distorsionando el modelo.

```
crPlots(rls, ylim = c(-2.3, 3.3),
        col = "steelblue", pch = 20, col.lines = c("red", "steelblue"),
        smooth = list(smoother = loessLine, span = 1),
        id = list(method = "r", n = 8, cex = 0.7, location = "lr"))
```



Vemos que en realidad no parece haber un apalancamiento indebido de alguno de estos casos. Podríamos sospechar del caso 66, pero su potencial influencia parece contrarrestada por valores cercanos, pero por debajo de la línea de regresión. Para comprobar, podemos revisar cómo luce el modelo de RLS sin ese dato.

```
rls2 <- lm(fmla, data = datos_entren[-66, ])
crPlots(rls2, ylim = c(-2.3, 3.3),
        col = "steelblue", pch = 20, col.lines = c("red", "steelblue"),
        smooth = list(smoother = loessLine, span = 1),
        id = list(method = "r", n = 8, cex = 0.7, location = "lr"))
```



Podemos ver que el nuevo modelo es prácticamente igual al original, por lo que no parece necesario quitar casos. Hagamos una conclusión entonces.

El modelo obtenido parece confiable, ya que genera residuos aleatorios y no es posible descartar que sigan una distribución normal, usando un predictor que muestra una relación lineal con la variable respuesta. Tampoco se identifican casos que estén ejerciendo demasiada influencia en el modelo.

Por otro lado, el modelo consigue una bondad de ajuste aceptable, pues explica alrededor del 40% de la variabilidad en la variable predicha, que es una reducción significativa ($F(1;68) = 43{,}8;\ p < 0{,}001$).

Regresión lineal múltiple

Para cumplir con la instrucción 6, vamos a utilizar la estrategia de regresión escalonada implementada en la función `step()`. Para eso usaremos nuestro modelo de RLS como modelo mínimo, y como modelo máximo el que utiliza todos los predictores que seleccionamos anteriormente de forma aleatoria.

```
rlm_max_text <- paste(c(predictor, predictores), collapse = " + ")
rlm_max_fmla <- formula(paste(nombre_respuesta, rlm_max_text, sep = " ~ "))
rlm_max <- lm(rlm_max_fmla, data = datos_entren)

rlm <- step(rls, scope = list(lower = rls, upper = rlm_max), direction = "both")
```

```
Start:  AIC=-6.56
Knees.diameter ~ Wrists.diameter

                          Df Sum of Sq    RSS      AIC
+ Bitrochanteric.diameter  1   12.4092 47.786 -20.7231
+ Hip.Girth                1   10.4109 49.785 -17.8554
+ Calf.Maximum.Girth       1   10.0993 50.096 -17.4186
+ Biiliac.diameter         1    7.0482 53.147 -13.2800
+ Age                      1    6.2971 53.898 -12.2978
+ Ankle.Minimum.Girth      1    5.6329 54.562 -11.4404
+ Ankles.diameter          1    5.6137 54.582 -11.4157
<none>                                60.195  -6.5630
+ Waist.Girth              1    0.0460 60.149  -4.6165

Step:  AIC=-20.72
Knees.diameter ~ Wrists.diameter + Bitrochanteric.diameter

                          Df Sum of Sq    RSS     AIC
+ Age                      1    7.8451 39.941 -31.276
+ Calf.Maximum.Girth       1    3.4070 44.379 -23.901
+ Ankles.diameter          1    2.8811 44.905 -23.076
+ Hip.Girth                1    2.2880 45.498 -22.157
<none>                                47.786 -20.723
+ Waist.Girth              1    1.0187 46.768 -20.231
+ Ankle.Minimum.Girth      1    0.6749 47.111 -19.719
+ Biiliac.diameter         1    0.0020 47.784 -18.726
- Bitrochanteric.diameter  1   12.4092 60.195  -6.563

Step:  AIC=-31.28
Knees.diameter ~ Wrists.diameter + Bitrochanteric.diameter +
    Age

                          Df Sum of Sq    RSS     AIC
+ Ankles.diameter          1    4.9650 34.976 -38.568
+ Hip.Girth                1    2.6861 37.255 -34.150
+ Calf.Maximum.Girth       1    2.0896 37.851 -33.038
<none>                                39.941 -31.276
+ Ankle.Minimum.Girth      1    0.3802 39.561 -29.946
+ Waist.Girth              1    0.1107 39.830 -29.471
+ Biiliac.diameter         1    0.0138 39.927 -29.300
- Age                      1    7.8451 47.786 -20.723
- Bitrochanteric.diameter  1   13.9572 53.898 -12.298

Step:  AIC=-38.57
Knees.diameter ~ Wrists.diameter + Bitrochanteric.diameter +
    Age + Ankles.diameter

                          Df Sum of Sq    RSS     AIC
+ Hip.Girth                1    2.1781 32.798 -41.069
<none>                                34.976 -38.568
+ Calf.Maximum.Girth       1    0.7429 34.233 -38.071
+ Waist.Girth              1    0.1285 34.848 -36.826
```

```
+ Ankle.Minimum.Girth      1    0.0982 34.878 -36.765
+ Biiliac.diameter         1    0.0536 34.922 -36.676
- Ankles.diameter          1    4.9650 39.941 -31.276
- Age                      1    9.9290 44.905 -23.076
- Bitrochanteric.diameter  1   10.5384 45.515 -22.132

Step:  AIC=-41.07
Knees.diameter ~ Wrists.diameter + Bitrochanteric.diameter +
    Age + Ankles.diameter + Hip.Girth

                          Df Sum of Sq    RSS     AIC
+ Waist.Girth              1    1.3737 31.424 -42.064
+ Ankle.Minimum.Girth      1    1.2790 31.519 -41.853
<none>                                 32.798 -41.069
+ Biiliac.diameter         1    0.4357 32.362 -40.005
+ Calf.Maximum.Girth       1    0.0131 32.785 -39.097
- Hip.Girth                1    2.1781 34.976 -38.568
- Bitrochanteric.diameter  1    3.7028 36.501 -35.581
- Ankles.diameter          1    4.4570 37.255 -34.150
- Age                      1   10.2067 43.005 -24.103

Step:  AIC=-42.06
Knees.diameter ~ Wrists.diameter + Bitrochanteric.diameter +
    Age + Ankles.diameter + Hip.Girth + Waist.Girth

                          Df Sum of Sq    RSS     AIC
+ Ankle.Minimum.Girth      1    1.3769 30.047 -43.200
<none>                                 31.424 -42.064
- Waist.Girth              1    1.3737 32.798 -41.069
+ Biiliac.diameter         1    0.2812 31.143 -40.693
+ Calf.Maximum.Girth       1    0.0113 31.413 -40.089
- Bitrochanteric.diameter  1    2.8260 34.250 -38.036
- Age                      1    3.3467 34.771 -36.980
- Hip.Girth                1    3.4233 34.848 -36.826
- Ankles.diameter          1    3.9079 35.332 -35.859

Step:  AIC=-43.2
Knees.diameter ~ Wrists.diameter + Bitrochanteric.diameter +
    Age + Ankles.diameter + Hip.Girth + Waist.Girth + Ankle.Minimum.Girth

                          Df Sum of Sq    RSS     AIC
<none>                                 30.047 -43.200
+ Calf.Maximum.Girth       1    0.4969 29.551 -42.367
+ Biiliac.diameter         1    0.4652 29.582 -42.292
- Ankle.Minimum.Girth      1    1.3769 31.424 -42.064
- Waist.Girth              1    1.4716 31.519 -41.853
- Bitrochanteric.diameter  1    3.4375 33.485 -37.618
- Age                      1    3.8746 33.922 -36.710
- Hip.Girth                1    4.5894 34.637 -35.250
- Ankles.diameter          1    5.2218 35.269 -33.984
```

El modelo obtenido no cumple con lo solicitado en el enunciado, pues tiene un predictor más de lo permitido. Comencemos identificando un predictor para ser eliminado.

```
drop1(rlm, test = "F")
```

```
Single term deletions

Model:
Knees.diameter ~ Wrists.diameter + Bitrochanteric.diameter +
    Age + Ankles.diameter + Hip.Girth + Waist.Girth + Ankle.Minimum.Girth
                      Df Sum of Sq    RSS     AIC F value   Pr(>F)
<none>                             30.047 -43.200
Wrists.diameter          1    4.3025 34.350 -35.833  8.8779 0.004118 **
Bitrochanteric.diameter  1    3.4375 33.485 -37.618  7.0930 0.009847 **
Age                      1    3.8746 33.922 -36.710  7.9948 0.006309 **
Ankles.diameter          1    5.2218 35.269 -33.984 10.7747 0.001693 **
Hip.Girth                1    4.5894 34.637 -35.250  9.4698 0.003108 **
Waist.Girth              1    1.4716 31.519 -41.853  3.0365 0.086372 .
Ankle.Minimum.Girth      1    1.3769 31.424 -42.064  2.8410 0.096913 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Vemos que el menor cambio en AIC ocurre eliminando el predictor `Ankle.Minimum.Girth`, que lleva a un modelo equivalente en cuanto a variabilidad no explicada ($F(1, 62) = 2{,}841; p = 0{,}097$). Quitemos esta variable.

```
rlm <- update(rlm, . ~ . - Ankle.Minimum.Girth)
```

Evaluemos la confiabilidad del modelo de RLM conseguido. Comencemos revisando que no exista niveles inaceptables de **multicolinealidad**.

```
cat("Factores de inflación de la varianza:\n")
print(vif(rlm))
cat("Estadísticos de tolerancia:\n")
print(1 / vif(rlm))
```

```
Factores de inflación de la varianza:
        Wrists.diameter Bitrochanteric.diameter                     Age
               1.643894                1.950384                1.645266
        Ankles.diameter               Hip.Girth             Waist.Girth
               1.456971                4.454118                4.021333
Estadísticos de tolerancia:
        Wrists.diameter Bitrochanteric.diameter                     Age
              0.6083116               0.5127195               0.6078046
        Ankles.diameter               Hip.Girth             Waist.Girth
              0.6863554               0.2245114               0.2486737
```

Vemos que, en general, solo hay indicios de multicolinealidad *moderada*, pues solo dos predictores presentan valores de inflación de la varianza sobre 4. Probablemente estas dos variables están correlacionadas. Eliminemos la que presenta el mayor valor.

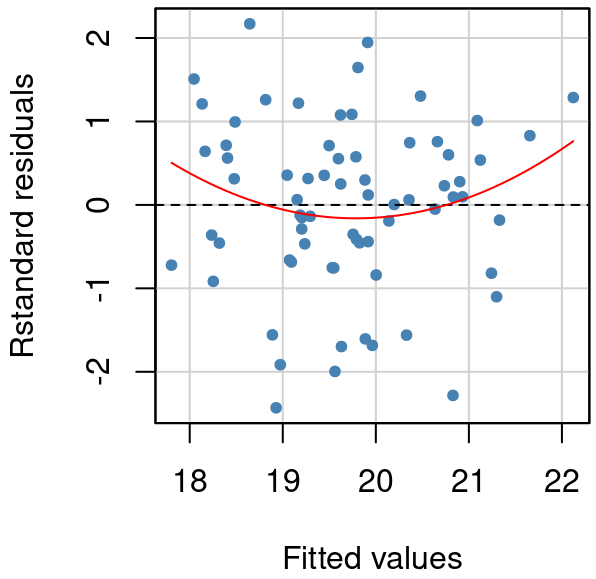
```
rlm <- update(rlm, . ~ . - Hip.Girth)

cat("Factores de inflación de la varianza:\n")
print(vif(rlm))
cat("Estadísticos de tolerancia:\n")
print(1 / vif(rlm))
```

```
Factores de inflación de la varianza:
        Wrists.diameter Bitrochanteric.diameter                     Age
               1.643792                1.492344                1.319838
        Ankles.diameter             Waist.Girth
               1.433072                1.701800
Estadísticos de tolerancia:
        Wrists.diameter Bitrochanteric.diameter                     Age
              0.6083494               0.6700870               0.7576686
        Ankles.diameter             Waist.Girth
              0.6978018               0.5876132
```

Muy bien, hemos eliminado gran parte de la multicolinealidad presente en el modelo anterior manteniendo 4 predictores nuevos agregados al modelo de RLS creado anteriormente.

Revisemos los residuos que genera este modelo.

```
cat("Prueba de curvatura:\n")
residualPlots(rlm, type = "rstandard", terms = ~ 1, col = "steelblue", pch = 20, col.quad = "red")
```

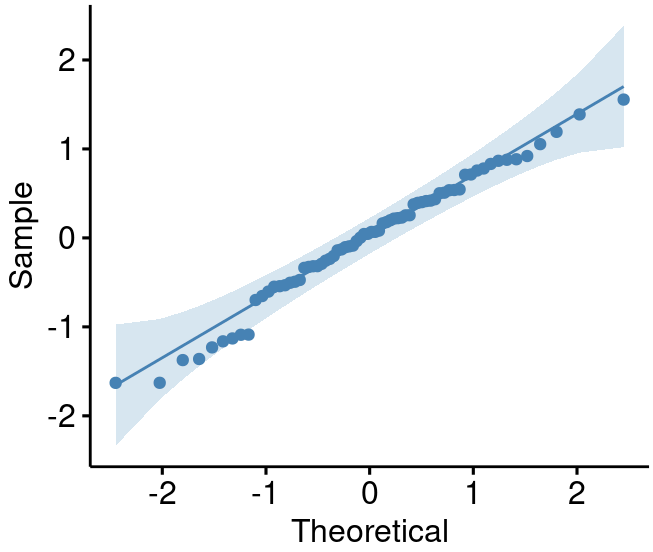


```
Prueba de curvatura:
           Test stat Pr(>|Test stat|)
Tukey test    1.5919           0.1114
```

Se ve cierta curvatura, pero que podría deberse a falta de observaciones en la muestra con diámetros de rodillas bajo los 18 o sobre los 21,5 cm. En el rango entre estos valores, no se ve un patrón preocupante, aunque existe cierta tendencia a patrones por sobre la línea de regresión. La prueba de curvatura también apunta en este sentido.

Revisemos la normalidad de estos residuos.

```
qq_res <- ggqqplot(data.frame(Residuos = resid(rlm)), x = "Residuos", color = "steelblue")
print(qq_res)
```




```
sw_res <- shapiro.test(resid(rlm))
cat("Test de normalidad de los residuos del modelo de RLM:")
print(sw_res)
```

```
Test de normalidad de los residuos del modelo de RLM:
    Shapiro-Wilk normality test

data:  resid(rlm)
W = 0.98203, p-value = 0.413
```

Vemos que los residuos parecen seguir una distribución normal, con algunos casos en el límite, pero que no son suficientes para permitir descartar que se cumple esta condición ($W = 0{,}982$; $p = 0.413$).

Ahora verifiquemos la varianza e independencia de los residuos.

```
cat("Prueba de varianza del error no constante:\n")
ncvTest(rlm)

cat("\nIndependencia de los residuos\n")
print(durbinWatsonTest(rlm))
```

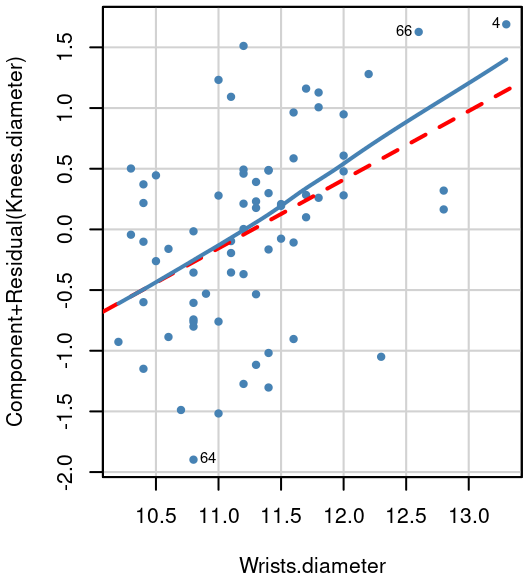
```
Prueba de varianza del error no constante:
Non-constant Variance Score Test
Variance formula: ~ fitted.values
Chisquare = 0.3385501, Df = 1, p = 0.56067

Independencia de los residuos
 lag Autocorrelation D-W Statistic p-value
   1       0.1604509      1.668819   0.156
 Alternative hypothesis: rho != 0
```

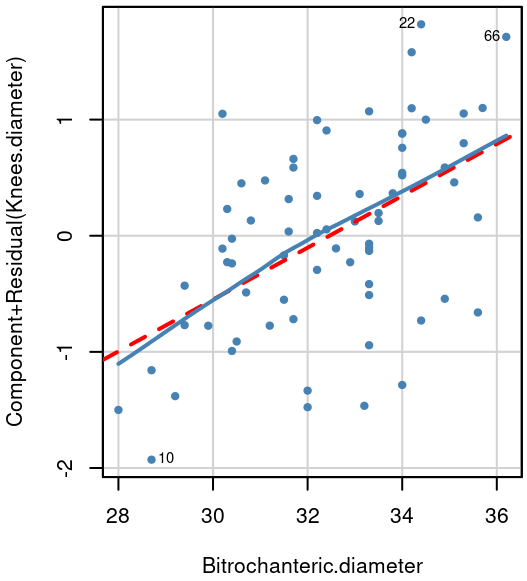
Con esto confirmamos que no es posible descartar que se están cumpliendo las condiciones de homogeneidad de la varianza ($\chi(1) = 0{,}339$; $p = 0{,}561$) e independencia de los residuos ($\text{D-W} = 1{,}669$; $p = 0{,}156$).

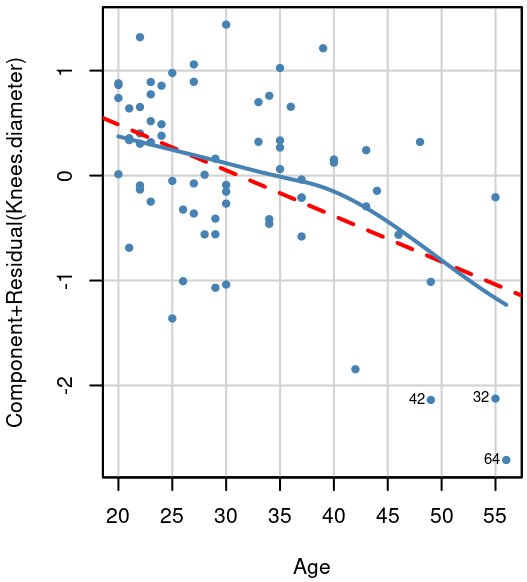
Revisemos si existen relaciones *aproximadamente* lineales entre los predictores y la variable de interés.

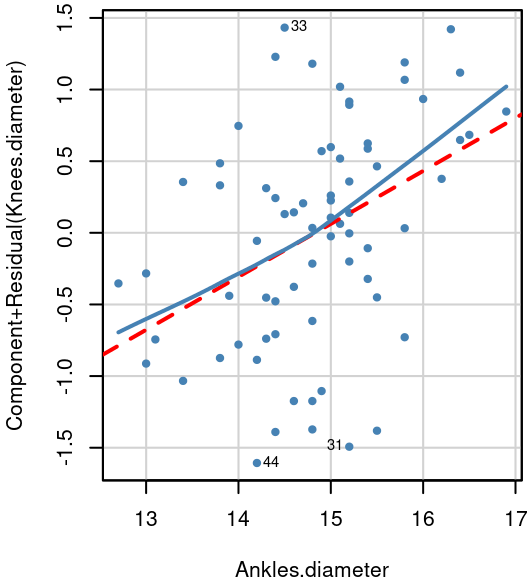
```
crPlots(rlm,
        col = "steelblue", pch = 20, col.lines=c("red", "steelblue"),
        smooth = list(smoother=loessLine, span = 1),
        id = list(method = "r", n = 3, cex = 0.7, location = "lr"))
```

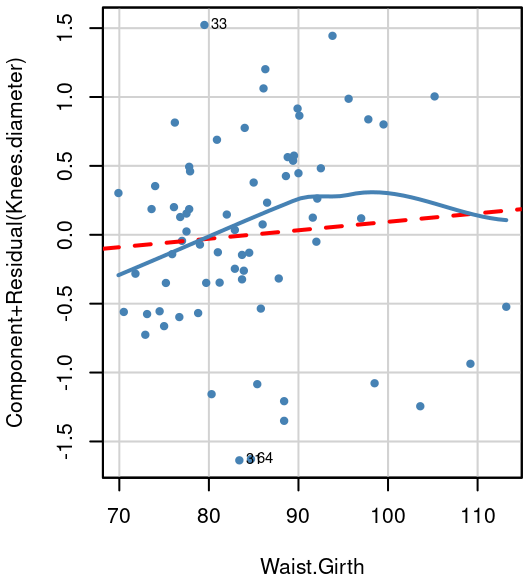
Component + Residual Plots











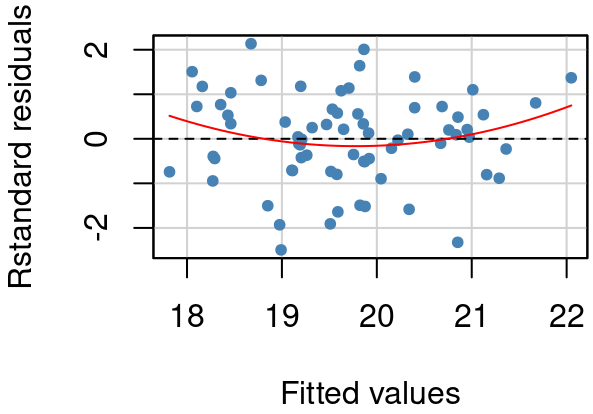
Observamos que las relaciones parecen aproximadamente lineales, aunque alguna duda puede quedar con cómo se distribuyen los residuos al considerar la variable `Waist.Girth` (grosor de la cintura). También podemos notar que la recta de regresión parcial para este predictor tiene una pendiente muy baja, abriendo dudas de su aporte. Revisemos su contribución en relación a los otros predictores.

```
cat("Modelo de RLM obtenido:\n")
print(summary(rlm))
```

```
Modelo de RLM obtenido:

Call:
lm(formula = Knees.diameter ~ Wrists.diameter + Bitrochanteric.diameter +
    Age + Ankles.diameter + Waist.Girth, data = datos_entren)

Residuals:
     Min       1Q   Median       3Q      Max
-1.62930 -0.43887  0.05634  0.48511  1.55466

Coefficients:
                         Estimate Std. Error t value Pr(>|t|)
(Intercept)              1.418884   1.930894   0.735  0.46513
Wrists.diameter          0.565732   0.178570   3.168  0.00235 **
Bitrochanteric.diameter  0.223152   0.055878   3.994  0.00017 ***
Age                     -0.043533   0.010848  -4.013  0.00016 ***
Ankles.diameter          0.369675   0.122204   3.025  0.00358 **
Waist.Girth              0.006145   0.012651   0.486  0.62881
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.7379 on 64 degrees of freedom
Multiple R-squared:  0.6479,    Adjusted R-squared:  0.6204
F-statistic: 23.55 on 5 and 64 DF,  p-value: 2.4e-13
```

Esto confirma que esta variable no aporta al modelo. Siguiendo el principio de parsimonia, es mejor que lo quitemos (y hacer una revisión rápida que nada se altera demasiado al introducir este cambio).

```
rlm <- update(rlm, . ~ . - Waist.Girth)

cat("Modelo de RLM obtenido:\n")
print(summary(rlm))

cat("\nPrueba de curvatura:\n")
residualPlots(rlm, type = "rstandard", terms = ~ 1, col = "steelblue", pch = 20, col.quad = "re
d")
```



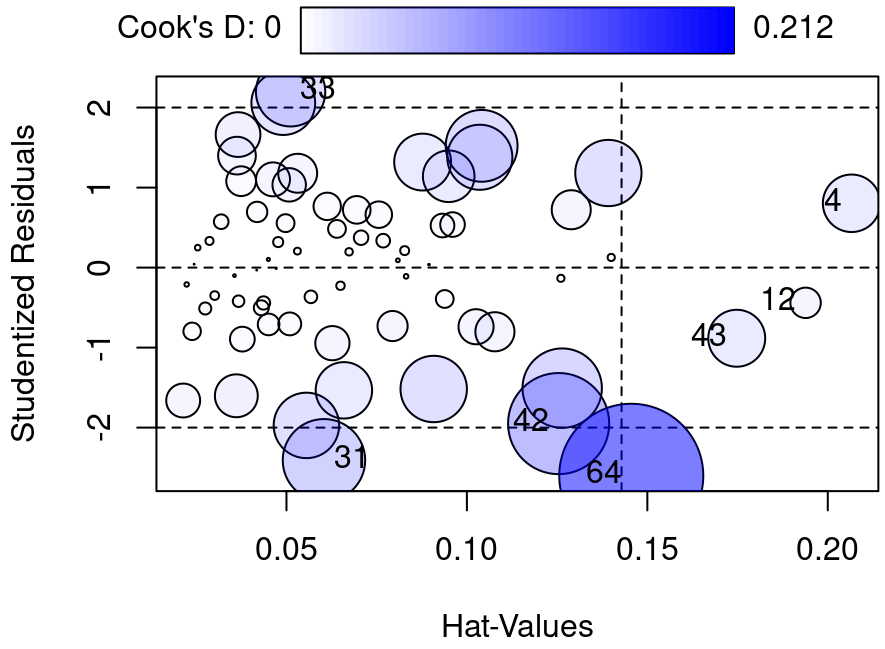

```
cat("\nFactores de inflación de la varianza:\n")
print(vif(rlm))

cat("\nPrueba de varianza del error no constante:\n")
ncvTest(rlm)

cat("\nIndependencia de los residuos\n")
print(durbinWatsonTest(rlm))
```

```
Modelo de RLM obtenido:

Call:
lm(formula = Knees.diameter ~ Wrists.diameter + Bitrochanteric.diameter +
    Age + Ankles.diameter, data = datos_entren)

Residuals:
     Min       1Q   Median       3Q      Max
-1.69107 -0.47170  0.04761  0.48962  1.52555

Coefficients:
                        Estimate Std. Error t value Pr(>|t|)
(Intercept)             1.321201   1.909073   0.692  0.49136
Wrists.diameter         0.588980   0.171023   3.444  0.00101 **
Bitrochanteric.diameter 0.232096   0.052446   4.425 3.76e-05 ***
Age                    -0.041103   0.009569  -4.296 5.94e-05 ***
Ankles.diameter         0.368992   0.121475   3.038  0.00343 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.7335 on 65 degrees of freedom
Multiple R-squared:  0.6466,    Adjusted R-squared:  0.6249
F-statistic: 29.73 on 4 and 65 DF,  p-value: 4.585e-14


Prueba de curvatura:
           Test stat Pr(>|Test stat|)
Tukey test    1.6415           0.1007

Factores de inflación de la varianza:
        Wrists.diameter Bitrochanteric.diameter                    Age
               1.525713                1.330286               1.039155
        Ankles.diameter
               1.432882

Prueba de varianza del error no constante:
Non-constant Variance Score Test
Variance formula: ~ fitted.values
Chisquare = 0.3581494, Df = 1, p = 0.54954

Independencia de los residuos
 lag Autocorrelation D-W Statistic p-value
   1        0.157849      1.674262   0.144
 Alternative hypothesis: rho != 0
```

El nuevo modelo más simple parece mantener el comportamiento del modelo anterior.

Revisemos ahora si existen casos demasiado influyentes utilizando el gráfico de influencia para identificarlos.

```
cat("Rango para 95% de los residuos studentizados: ")
cat("[", round(qt(0.05/2, nrow(datos_entren) - length(coef(rls)) - 1), 3), ", ", sep = "")
cat(round(qt(1-0.05/2, nrow(datos_entren) - length(coef(rls)) - 1), 3), "]\n", sep = "")
cat("Límite del apalancamiento:", round(2 * mean(hatvalues(rlm)), 3), "\n")
cat("Límite de la distancia de Cook:", round(3 * mean(cooks.distance(rlm)), 3), "\n")

rlm_inf <- influencePlot(rlm, id = list(n = 3))
```

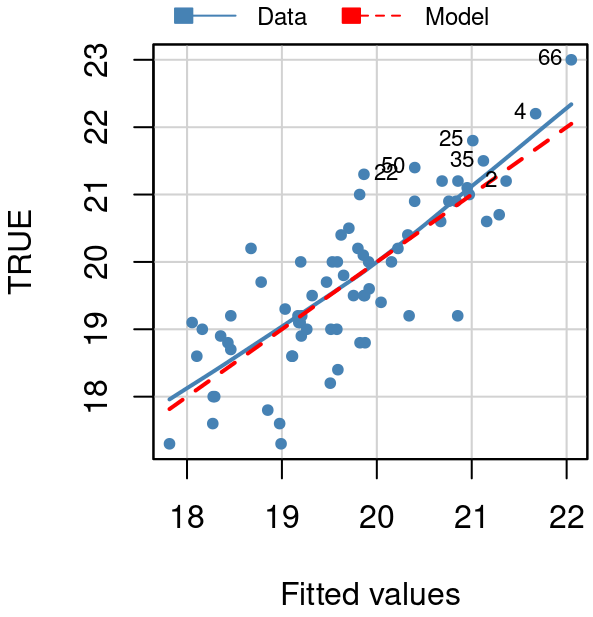



```
cat("\nCasos notorios para el modelo de RLM:\n")
print(rlm_inf)
```

```
Rango para 95% de los residuos studentizados: [-1.996, 1.996]
Límite del apalancamiento: 0.143
Límite de la distancia de Cook: 0.052

Casos notorios para el modelo de RLM:
      StudRes        Hat       CookD
4   0.8036599 0.20660532 0.033822004
12 -0.4404798 0.19385127 0.009448313
31 -2.4080519 0.06039000 0.069413666
33  2.1969458 0.05115061 0.049145017
42 -1.9497935 0.12542112 0.104532595
43 -0.8832504 0.17477019 0.033155965
64 -2.6023356 0.14554713 0.211896164
```

Y el gráfico marginal de los valores predichos (`fitted`) para evaluar su influencia en el modelo.

```
id_inf <- mmp(rlm,
              col = "steelblue", pch = 20, col.line = c("steelblue", "red"),
              smooth = list(smoother=loessLine, span = 1),
              id = list(method = "r", n = 7, cex = 0.7, location = "lr"))
```



Vemos que, a pesar que los casos 4 y 66 curvan la tendencia de los datos hacia arriba, el modelo (línea roja segmentada) no parece estar visiblemente modificada por alguno de los casos notorios identificados.

Notemos que en la función que genera el gráfico marginal, `mmp()` o su equivalente `marginalModelPlot()`, usan el argumento `col.line` para indicar los colores de las curvas ajustadas, primero para la curva suavizada de los datos y el segundo para la curva del modelo. Sin embargo, la función `crPlots()` que genera los gráficos de residuos por componente utiliza el parámetro `col.lines` (plural) para este propósito, debiendo indicar primero el color para la curva del modelo y luego el color para la curva suavizada de los datos. ¡Qué falta de consistencia! Es el precio de construir bibliotecas en comunidad…

Finalmente, cometemos la bondad de ajuste que alcanza el modelo. Vemos que consigue una reducción significativa de la variabilidad aleatoria ($F(4;65) = 29{,}7;\ p < 0{,}001$), pues explica alrededor del 65% de la varianza de la variable de salida.

Con todo este análisis podemos dar la siguiente conclusión.

El modelo de RLM obtenido parece ser confiable, puesto que se ajusta bien a los datos observados, incluye predictores que muestran una relación lineal con la variable de respuesta, genera residuos que parecen seguir una distribución normal y sin problemas evidentes de heterocedasticidad o de dependencia entre ellos. Por otro lado, no hay casos que estén dominando el modelo.

Comparación de los modelos

Vimos que el modelo de RLS construido logra explicar alrededor del 40% de la variabilidad en los datos, mientras que el RLM que tenemos logra explicar cerca del 65% . Confirmemos si esta es una mejora significativa en la bondad de ajuste.

```
cat("Comparación de los modelos de RLS y RLM:\n")
print(anova(rls, rlm))
```

```
Comparación de los modelos de RLS y RLM:
Analysis of Variance Table

Model 1: Knees.diameter ~ Wrists.diameter
Model 2: Knees.diameter ~ Wrists.diameter + Bitrochanteric.diameter +
    Age + Ankles.diameter
  Res.Df    RSS Df Sum of Sq      F    Pr(>F)
1     68 60.195
2     65 34.976  3    25.219 15.623 9.332e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Confirmamos entonces que el modelo de RLM consigue una reducción significativa de la varianza no explicada en los datos con respecto al modelo de RLS ($F(3,65) = 15{,}623; p < 0{,}001$).

Veamos si estos niveles de bondad de ajuste se reflejan en la calidad predictiva de los modelos conseguidos.

Como se indica en el enunciado, es importante hacer esta evaluación con datos **distintos** a los usados en la construcción de los modelos. Por esta razón hemos construido los modelos usando 70% de los datos disponibles, dejando el resto para hacer esta evaluación. Así, podemos comparar las predicciones que hacen con datos vistos (los de entrenamiento) y no vistos (los de prueba).

```
rls_rmse_entre <- sqrt(mean(resid(rls) ** 2))
rls_preds <- predict(rls, datos_prueba)
rls_res_prueba <- datos_prueba[[nombre_respuesta]] - rls_preds
rls_rmse_prueba <- sqrt(mean(rls_res_prueba ** 2))
rls_pct_cambio <- ((rls_rmse_prueba - rls_rmse_entre) / rls_rmse_entre) * 100

rlm_rmse_entre <- sqrt(mean(resid(rlm) ** 2))
rlm_preds <- predict(rlm, datos_prueba)
rlm_res_prueba <- datos_prueba[[nombre_respuesta]] - rlm_preds
rlm_rmse_prueba <- sqrt(mean(rlm_res_prueba ** 2))
rlm_pct_cambio <- ((rlm_rmse_prueba - rlm_rmse_entre) / rlm_rmse_entre) * 100

cat(sprintf("Resumen de la variable de salida (%s):\n", nombre_respuesta))
print(describe(datos |> pull(all_of(nombre_respuesta)), skew = FALSE))
cat("\n")
cat("Rendimiento del modelo de RLS:\n")
cat(sprintf("RMSE para el conjunto de entrenamiento: %.3f\n", rls_rmse_entre))
cat(sprintf("RMSE para el conjunto de prueba: %.3f\n", rls_rmse_prueba))
cat(sprintf("Cambio en el error: %.1f%%\n", rls_pct_cambio))
cat("\n")
cat("Rendimiento del modelo de RLM:\n")
cat(sprintf("RMSE para el conjunto de entrenamiento: %.3f\n", rlm_rmse_entre))
cat(sprintf("RMSE para el conjunto de prueba: %.3f\n", rlm_rmse_prueba))
cat(sprintf("Cambio en el error: %.1f%%\n", rlm_pct_cambio))
```

```
Resumen de la variable de salida (Knees.diameter):
   vars   n  mean   sd median  min max range   se
X1    1 100 19.72 1.18  19.55 17.3  23   5.7 0.12

Rendimiento del modelo de RLS:
RMSE para el conjunto de entrenamiento: 0.927
RMSE para el conjunto de prueba: 1.138
Cambio en el error: 22.7%

Rendimiento del modelo de RLM:
RMSE para el conjunto de entrenamiento: 0.707
RMSE para el conjunto de prueba: 1.000
Cambio en el error: 41.5%
```

Podemos observar que, efectivamente, el modelo de RLM obtiene menores tasas de error que el modelo de RLS. Sin embargo, esta disminución es más acentuada en los datos de entrenamiento y no se exhibe de igual magnitud en los de prueba. Por otro lado, un error de $\pm 1{,}0$ podría ser alto si se considera que el rango de la variable de salida ($17{,}3$–$23{,}0$) es de solo $5{,}7$. Así, podemos concluir lo siguiente.

El modelo de RLM logra mejorar el rendimiento del modelo de RLS pero hay indicios de sobreajuste en él, ya que el error aumenta más de un 40% al pasar de datos vistos a datos no vistos. La calidad predictiva del modelo tampoco parece ser muy buena.

A pesar de que este modelo de RLM resultó confiable, parece tener **problemas de generalización y calidad predictiva**.

Lo que correspondería entonces es analizar la eliminación de uno o dos de los predictores y evaluar nuevamente la confiabilidad y el poder predictivo del nuevo modelo de RLM. *Esto se deja como ejercicio.*

Referencias

Heinz, G., Peterson, L. J., Johnson, R. W., & Kerk, C. J. (2003). Exploring relationships in body dimensions. Journal of Statistics Education, 11(2).