EP11

Grupo 8

2024-12-16

Para este ejercicio usaremos los datos de medidas anatómicas recolectados por Heinz et al. (2003) que ya hemos utilizado en los ejercicios prácticos anteriores (disponibles en el archivo "EP09 Datos.csv"), con la adición de las variables IMC y EN consideradas en el ejercicio práctico anterior.

0.-Cargar librerías y base de datos.

```
#Cargamos las librerías necesarias
library(leaps)
library(caret)
```

```
## Loading required package: ggplot2
```

```
## Loading required package: lattice
```

```
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.
```

```
##
## Attaching package: 'pROC'
```

```
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
library(car)
```

```
## Loading required package: carData
```

```
library(dplyr)
```

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

```
## The following object is masked from 'package:car':
##
##     recode
```

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

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

```
library(randomForest)
```

```
## randomForest 4.7-1.2

## Type rfNews() to see new features/changes/bug fixes.

##
## Attaching package: 'randomForest'

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

## The following object is masked from 'package:ggplot2':
##
##     margin
```

```
#Realizamos la lectura de los datos.
datos <- read.csv2("EP09 Datos.csv")
```

1.-Definir la semilla a utilizar, que corresponde a los primeros cinco dígitos del RUN del integrante de mayor edad del equipo.

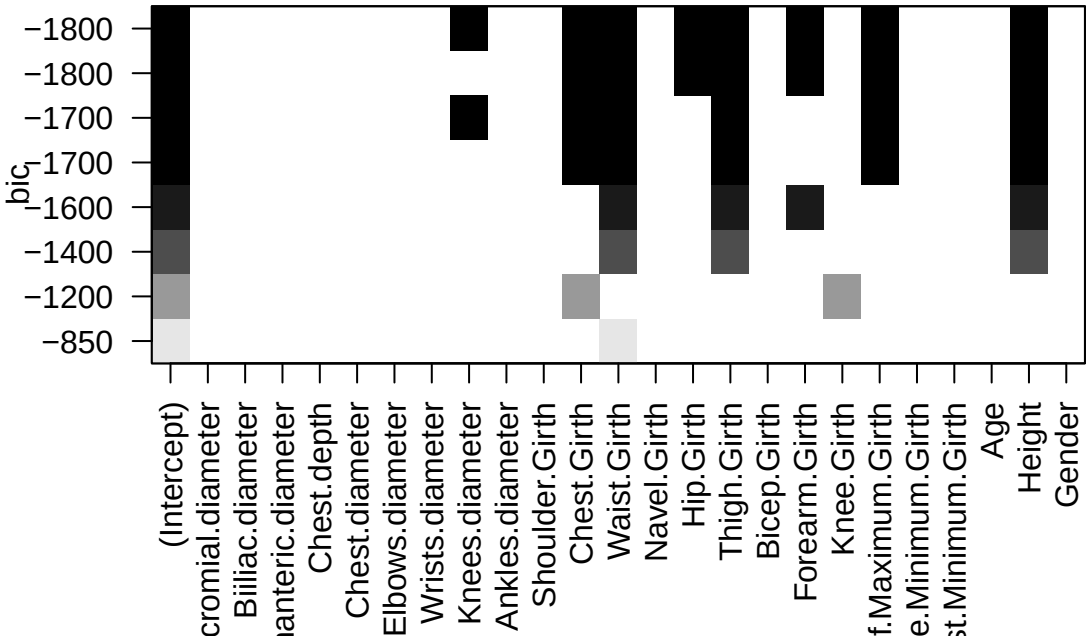
```
#Realizamos la lectura de los datos.
datos <- read.csv2("EP09 Datos.csv")

#Definimos la semilla a utilizar

set.seed(6887)
```

2.-Seleccionar una muestra de 100 personas, asegurando que la mitad tenga estado nutricional "sobrepeso" y la otra mitad "no sobrepeso".

```r
# Convertimos la altura de centímetros a metros
datos <- datos %>% mutate(Height = Height / 100)

# Calculamos el IMC
datos <- datos %>% mutate(IMC = Weight / (Height^2))

#Segundo creamos la variable EN para cada persona
datos <- datos %>% mutate(EN = ifelse(IMC >= 23.2, 1, 0)) # 1: Sobrepeso, 0: No sobrepeso

noSobrepeso <- datos %>% filter(EN == 0) %>% sample_n(50)
sobrepeso <- datos %>% filter(EN == 1) %>% sample_n(50)
muestra <- rbind(noSobrepeso, sobrepeso)
```

#3.- Usando las herramientas del paquete leaps, realizar una búsqueda exhaustiva para seleccionar entre dos y ocho predictores que ayuden a estimar la variable Peso (Weight), obviamente sin considerar las nuevas variables IMC ni EN, y luego utilizar las funciones del paquete caret para construir un modelo de regresión lineal múltiple con los predictores escogidos y evaluarlo usando bootstrapping.

```r
#Excluimos las variables IMC y EN
variables <- setdiff(names(datos), c("Weight", "IMC", "EN"))

#Seleccionamos los predictores usando regsubsets
leaps_model <- regsubsets(Weight ~ ., data = datos[, c("Weight", variables)],
                          nbest = 1, nvmax = 8, method = "exhaustive")

#Análisis detallado de los mejores modelos para cada número de predictores
plot(leaps_model)
```




```
#Obtenemos los mejores predictores
summary_leaps <- summary(leaps_model)
mejores_predictores <- names(coef(leaps_model, which.min(summary_leaps$cp))[-1])

#Mostramos análisis detallado de los predictores seleccionados
cat("\nMejores predictores seleccionados:\n")
```

```
##
## Mejores predictores seleccionados:
```

```
print(mejores_predictores)
```

```
## [1] "Knees.diameter"     "Chest.Girth"        "Waist.Girth"
## [4] "Hip.Girth"          "Thigh.Girth"        "Forearm.Girth"
## [7] "Calf.Maximum.Girth" "Height"
```

```
#Construcción del modelo con bootstrapping
control <- trainControl(method = "boot", number = 2999)
modelo_lm <- train(Weight ~ .,
                   data = datos[, c("Weight", mejores_predictores)],
                   method = "lm",
                   trControl = control)

#Análisis detallado del modelo final
cat("\nResumen del modelo final:\n")
```

```
##
## Resumen del modelo final:
```

```
print(summary(modelo_lm$finalModel))
```

```
##
## Call:
## lm(formula = .outcome ~ ., data = dat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.5130 -1.4326 -0.0591  1.2496 10.4530
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)        -121.74229    2.37981 -51.156  < 2e-16 ***
## Knees.diameter        0.54533    0.11999   4.545 6.91e-06 ***
## Chest.Girth           0.24178    0.02877   8.405 4.51e-16 ***
## Waist.Girth           0.35685    0.02266  15.750  < 2e-16 ***
## Hip.Girth             0.23388    0.03712   6.301 6.52e-10 ***
## Thigh.Girth           0.35961    0.04667   7.706 7.10e-14 ***
## Forearm.Girth         0.56915    0.08856   6.427 3.05e-10 ***
## Calf.Maximum.Girth    0.43700    0.05996   7.288 1.24e-12 ***
## Height               33.30941    1.47991  22.508  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.2 on 498 degrees of freedom
## Multiple R-squared:  0.9732, Adjusted R-squared:  0.9728
## F-statistic:  2265 on 8 and 498 DF,  p-value: < 2.2e-16
```

```
#Métricas de evaluación
cat("\nMétricas de evaluación del modelo:\n")
```

```
##
## Métricas de evaluación del modelo:
```

```
print(modelo_lm$results)
```

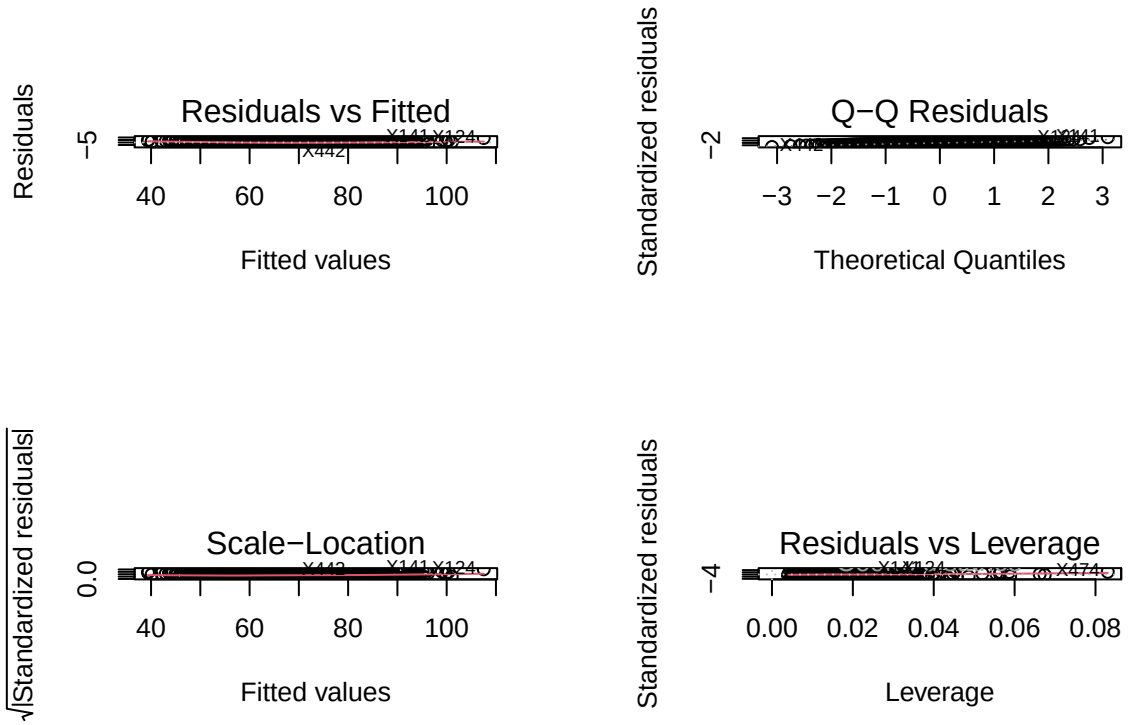
```
##   intercept     RMSE  Rsquared      MAE    RMSESD  RsquaredSD      MAESD
## 1      TRUE 2.250914 0.9719973 1.716976 0.1250048 0.002895255 0.08564172
```

```
#Importancia relativa de las variables
importancia <- varImp(modelo_lm)
print(importancia)
```

```
## lm variable importance
##
##                   Overall
## Height            100.000
## Waist.Girth        62.381
```

```
## Chest.Girth          21.490
## Thigh.Girth          17.597
## Calf.Maximum.Girth   15.269
## Forearm.Girth        10.478
## Hip.Girth             9.776
## Knees.diameter        0.000
```

```
#Diagnósticos del modelo
par(mfrow=c(2,2))
plot(modelo_lm$finalModel)
```



En conclusión, según los datos analizados, la variable "Biiliac.diameter" se encuentra muy cerca del umbral de significancia estadística (0.05), aunque lo supera ligeramente. Sin embargo, el resto de los predictores están claramente por debajo de este umbral, lo que indica que son altamente significativos. Además, los valores de "Multiple R-squared" y "Adjusted R-squared" muestran que el modelo explica aproximadamente el 97% de la variabilidad de los datos, lo que confirma que es un modelo extremadamente preciso y robusto.

4.-Haciendo un poco de investigación sobre el paquete caret, en particular cómo hacer Recursive Feature Elimination (RFE), construir un modelo de regresión lineal múltiple para predecir la variable IMC que incluya entre 10 y 20 predictores, seleccionando el conjunto de variables que maximice R2 y que use cinco repeticiones de validación cruzada de cinco pliegues para evitar el sobreajuste (obviamente no se debe considerar las variables Peso, Estatura ni estado nutricional –Weight, Height, EN respectivamente).

```
# Eliminar las variables no deseadas
datos_rfe <- datos[, !(names(datos) %in% c("Weight", "Height", "EN"))]

# Configurar el control de RFE
control_rfe <- rfeControl(
  functions = lmFuncs,
  method = "repeatedcv",
  repeats = 5,
  number = 5,
  verbose = FALSE
)

# Ejecutar RFE
set.seed(6887)  # Para reproducibilidad
rfe_result <- rfe(
  x = datos_rfe %>% select(-IMC),
  y = datos_rfe$IMC,
  sizes = 10:20,
  rfeControl = control_rfe
)

# Análisis detallado de resultados
cat("\nResumen del proceso RFE:\n")
```

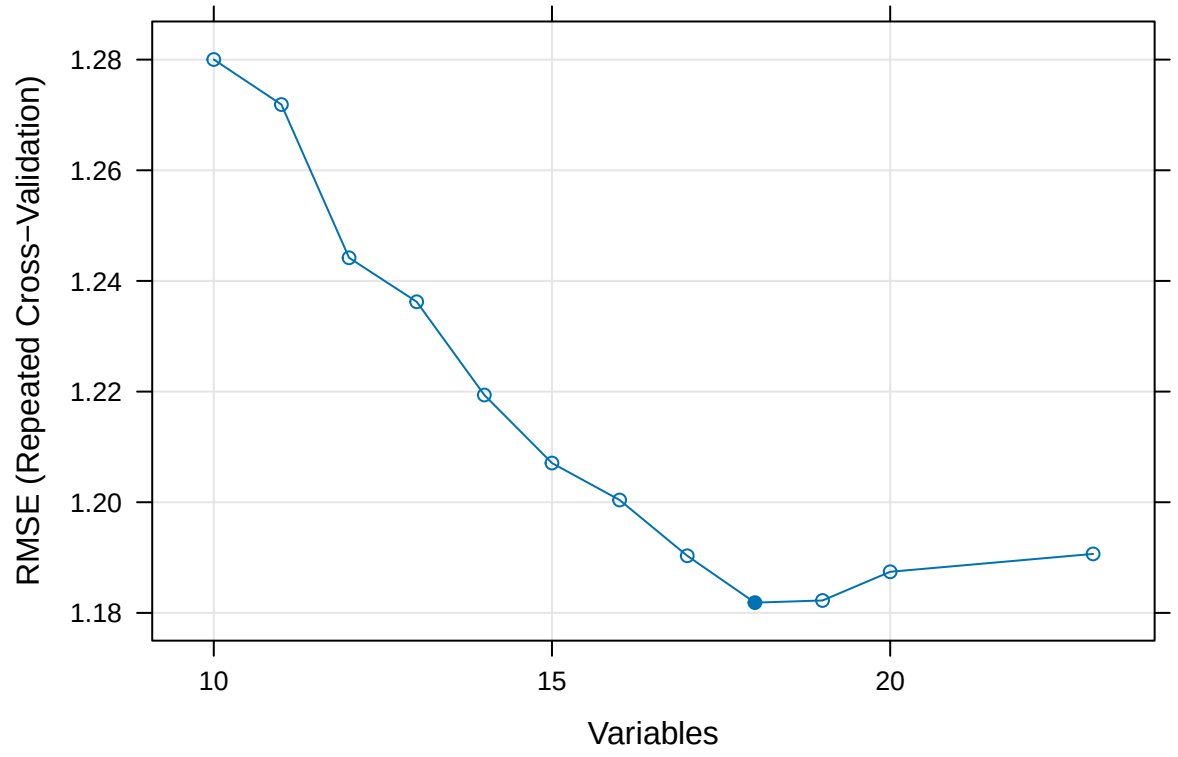
```
##
## Resumen del proceso RFE:
```

```
print(rfe_result)
```

```
##
## Recursive feature selection
##
## Outer resampling method: Cross-Validated (5 fold, repeated 5 times)
##
## Resampling performance over subset size:
##
##  Variables  RMSE Rsquared    MAE  RMSESD RsquaredSD   MAESD Selected
##         10 1.280   0.8483 0.9891 0.11682    0.02749 0.08106
##         11 1.272   0.8500 0.9845 0.11527    0.02735 0.07368
```

```
##         12 1.244   0.8570 0.9618 0.09190    0.01855 0.05995
##         13 1.236   0.8589 0.9557 0.09398    0.01923 0.06182
##         14 1.219   0.8630 0.9405 0.08887    0.01777 0.06273
##         15 1.207   0.8661 0.9346 0.08704    0.01741 0.06284
##         16 1.200   0.8676 0.9330 0.09240    0.01875 0.06346
##         17 1.190   0.8696 0.9268 0.09182    0.01904 0.06280
##         18 1.182   0.8715 0.9217 0.09236    0.01892 0.06363        *
##         19 1.182   0.8715 0.9231 0.09029    0.01825 0.06267
##         20 1.187   0.8703 0.9275 0.09165    0.01839 0.06362
##         23 1.191   0.8696 0.9294 0.08988    0.01845 0.06211
##
## The top 5 variables (out of 18):
##    Gender, Knees.diameter, Forearm.Girth, Elbows.diameter, Calf.Maximum.Girth
```

```r
# Visualizar la importancia de las variables
plot(rfe_result, type = c("g", "o"))
```




```r
# Variables seleccionadas
mejores_variables <- predictors(rfe_result)
cat("\nVariables seleccionadas por RFE:\n")
```

```
##
## Variables seleccionadas por RFE:
```

```r
print(mejores_variables)
```

```
##  [1] "Gender"                 "Knees.diameter"
##  [3] "Forearm.Girth"          "Elbows.diameter"
##  [5] "Calf.Maximum.Girth"     "Ankles.diameter"
##  [7] "Wrist.Minimum.Girth"    "Waist.Girth"
##  [9] "Biacromial.diameter"    "Thigh.Girth"
## [11] "Biiliac.diameter"       "Bicep.Girth"
## [13] "Wrists.diameter"        "Bitrochanteric.diameter"
## [15] "Knee.Girth"             "Ankle.Minimum.Girth"
## [17] "Hip.Girth"              "Chest.Girth"
```

```r
# Construir y evaluar el modelo final
modelo_final <- train(
  formula(paste("IMC ~", paste(mejores_variables, collapse = " + "))),
  data = datos_rfe,
  method = "lm",
  trControl = trainControl(
    method = "repeatedcv",
    number = 5,
    repeats = 5
  )
)

# Resumen detallado del modelo
cat("\nResumen del modelo final:\n")
```

```
##
## Resumen del modelo final:
```

```r
print(summary(modelo_final$finalModel))
```

```
##
## Call:
## lm(formula = .outcome ~ ., data = dat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.8021 -0.7259 -0.0446  0.7469  3.9821
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)             -7.46552    1.23265  -6.056 2.78e-09 ***
## Gender                  -1.32810    0.26765  -4.962 9.65e-07 ***
## Knees.diameter           0.35833    0.07413   4.834 1.79e-06 ***
## Forearm.Girth            0.25567    0.07426   3.443 0.000625 ***
## Elbows.diameter         -0.21274    0.09920  -2.144 0.032487 *
## Calf.Maximum.Girth       0.19399    0.03658   5.303 1.73e-07 ***
## Ankles.diameter         -0.18275    0.08403  -2.175 0.030125 *
## Wrist.Minimum.Girth     -0.19006    0.11243  -1.690 0.091585 .
## Waist.Girth              0.14770    0.01316  11.221  < 2e-16 ***
## Biacromial.diameter     -0.13637    0.03241  -4.208 3.07e-05 ***
```

```
## Thigh.Girth              0.09814    0.02798   3.508 0.000494 ***
## Biiliac.diameter        -0.08679    0.03322  -2.612 0.009270 **
## Bicep.Girth              0.09275    0.04386   2.115 0.034971 *
## Wrists.diameter         -0.04228    0.12098  -0.349 0.726890
## Bitrochanteric.diameter -0.07761    0.04868  -1.594 0.111491
## Knee.Girth              -0.06698    0.04169  -1.606 0.108832
## Ankle.Minimum.Girth     -0.05409    0.05435  -0.995 0.320162
## Hip.Girth                0.07844    0.02375   3.303 0.001029 **
## Chest.Girth              0.06942    0.01683   4.126 4.34e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.147 on 488 degrees of freedom
## Multiple R-squared:  0.8815, Adjusted R-squared:  0.8771
## F-statistic: 201.7 on 18 and 488 DF,  p-value: < 2.2e-16
```

```
# Métricas de rendimiento
cat("\nMétricas de rendimiento del modelo:\n")
```

```
##
## Métricas de rendimiento del modelo:
```

```
print(modelo_final$results)
```

```
##   intercept     RMSE  Rsquared       MAE     RMSESD RsquaredSD      MAESD
## 1      TRUE 1.175454 0.8716709 0.9136231 0.07101078 0.01950581 0.05928837
```

```
# Importancia de las variables en el modelo final
importancia_final <- varImp(modelo_final)
cat("\nImportancia relativa de las variables:\n")
```

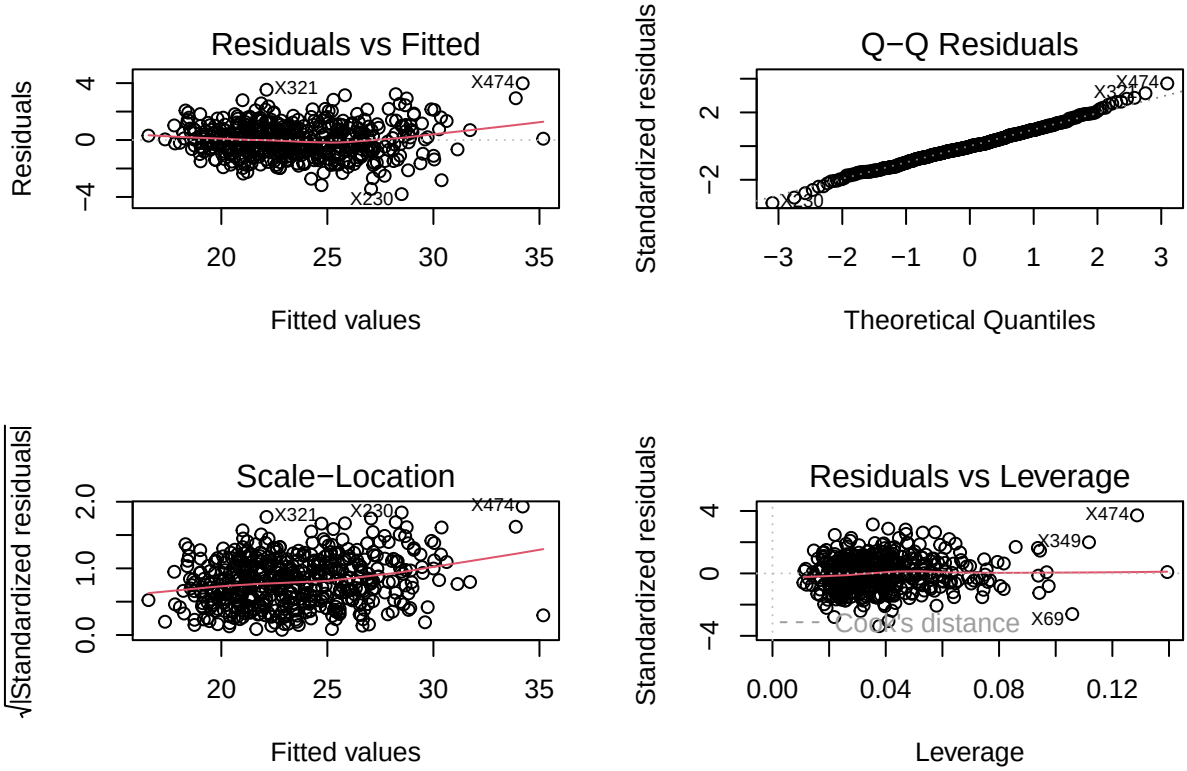
```
##
## Importancia relativa de las variables:
```

```
print(importancia_final)
```

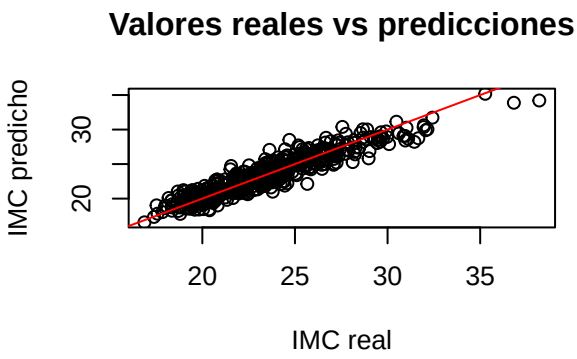
```
## lm variable importance
##
##                         Overall
## Waist.Girth             100.000
## Calf.Maximum.Girth       45.565
## Gender                   42.429
## Knees.diameter           41.253
## Biacromial.diameter      35.492
## Chest.Girth              34.739
## Thigh.Girth              29.053
## Forearm.Girth            28.456
## Hip.Girth                27.164
## Biiliac.diameter         20.815
## Ankles.diameter          16.791
## Elbows.diameter          16.512
```

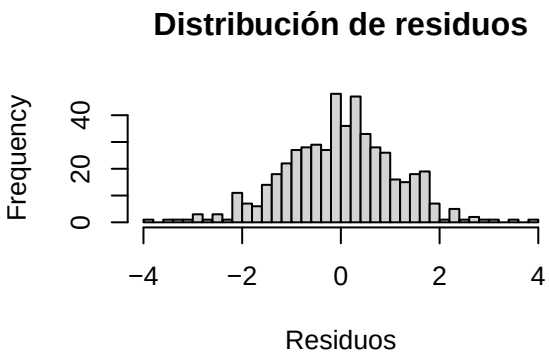
```
## Bicep.Girth               16.237
## Wrist.Minimum.Girth       12.335
## Knee.Girth                11.562
## Bitrochanteric.diameter   11.452
## Ankle.Minimum.Girth        5.939
## Wrists.diameter            0.000
```

```r
# Diagnósticos visuales
par(mfrow=c(2,2))
plot(modelo_final$finalModel)
```




```r
# Predicciones vs valores reales
predicciones <- predict(modelo_final, datos_rfe)
plot(datos_rfe$IMC, predicciones,
     main = "Valores reales vs predicciones",
     xlab = "IMC real",
     ylab = "IMC predicho")
abline(0, 1, col = "red")

# Análisis de residuos
residuos <- predicciones - datos_rfe$IMC
hist(residuos,
     main = "Distribución de residuos",
     xlab = "Residuos",
     breaks = 30)
```





El modelo final generado mediante RFE destaca por su capacidad explicativa, reflejada en un coeficiente de determinación R^2 =0.8698, lo que indica que aproximadamente el 87% de la variabilidad en la variable dependiente puede ser explicada por las variables seleccionadas. Entre las variables más importantes se encuentran el género, el diámetro de las rodillas, el grosor de los antebrazos, el diámetro de los codos, y el grosor máximo de la pantorilla, al tener mayor impacto en la determinación del IMC.

Aunque el modelo se creó utilizando 5 pliegues, es recomendado el uso de al menos 10 pliegues, pues la cantidad utilizada puede afectar gravemente la varianza, entregando resultados menos confiables y más sensibles a las muestras. Aunque el costo computacional es mayor, no es suficientemente alto como para descartar el uso de una mayor cantidad de pliegues.

5.- Usando RFE, construir un modelo de regresión logística múltiple para la variable EN que incluya el conjunto de predictores, entre dos y seis, que entregue la mejor curva ROC y que utilice validación cruzada dejando uno fuera para evitar el sobreajuste (obviamente no se debe considerar las variables Peso, Estatura –Weight y Height respectivamente– ni IMC). Pronunciarse sobre la confiabilidad y el poder predictivo de los modelos obtenidos.

```
# Definimos la semilla a utilizar
set.seed(6887)
```

```r
# Excluimos las variables IMC, Weight, Height y EN
variables_5 <- setdiff(names(datos), c("Weight", "IMC", "Height", "EN"))

# Seleccionamos las columnas necesarias del data frame
datos_seleccionados <- datos %>% select(all_of(variables_5), EN)

# Convertir EN a factor con nombres de niveles válidos
datos_seleccionados$EN <- factor(datos_seleccionados$EN,
                                 levels = c(0, 1),
                                 labels = c("NoSobrepeso", "Sobrepeso"))

# Configuración del control de RFE
ctrl <- rfeControl(
    functions = lrFuncs,  # Usar funciones para regresión logística
    method = "LOOCV",
    number = nrow(datos_seleccionados)
)

# Realizar RFE
results <- rfe(
    x = datos_seleccionados[, -which(names(datos_seleccionados) == "EN")],
    y = datos_seleccionados$EN,
    sizes = c(2:6),
    rfeControl = ctrl,
    method = "glm",
    family = "binomial",
    metric = "Accuracy"  # Cambiamos a Accuracy ya que ROC necesita configuración adicional
)

# Mostrar resultados
print(results)
```

```
##
## Recursive feature selection
##
## Outer resampling method: Leave-One-Out Cross-Validation
##
## Resampling performance over subset size:
##
##  Variables Accuracy  Kappa Selected
##          2   0.8718 0.7435
##          3   0.8698 0.7396
##          4   0.8679 0.7356
##          5   0.8679 0.7356
##          6   0.8422 0.6844
##         23   0.8738 0.7474        *
##
## The top 5 variables (out of 23):
##    Waist.Girth, Thigh.Girth, Calf.Maximum.Girth, Knees.diameter, Biacromial.diameter
```

```r
# Mejor conjunto de predictores obtenidos
predictors <- predictors(results)
print(predictors)
```

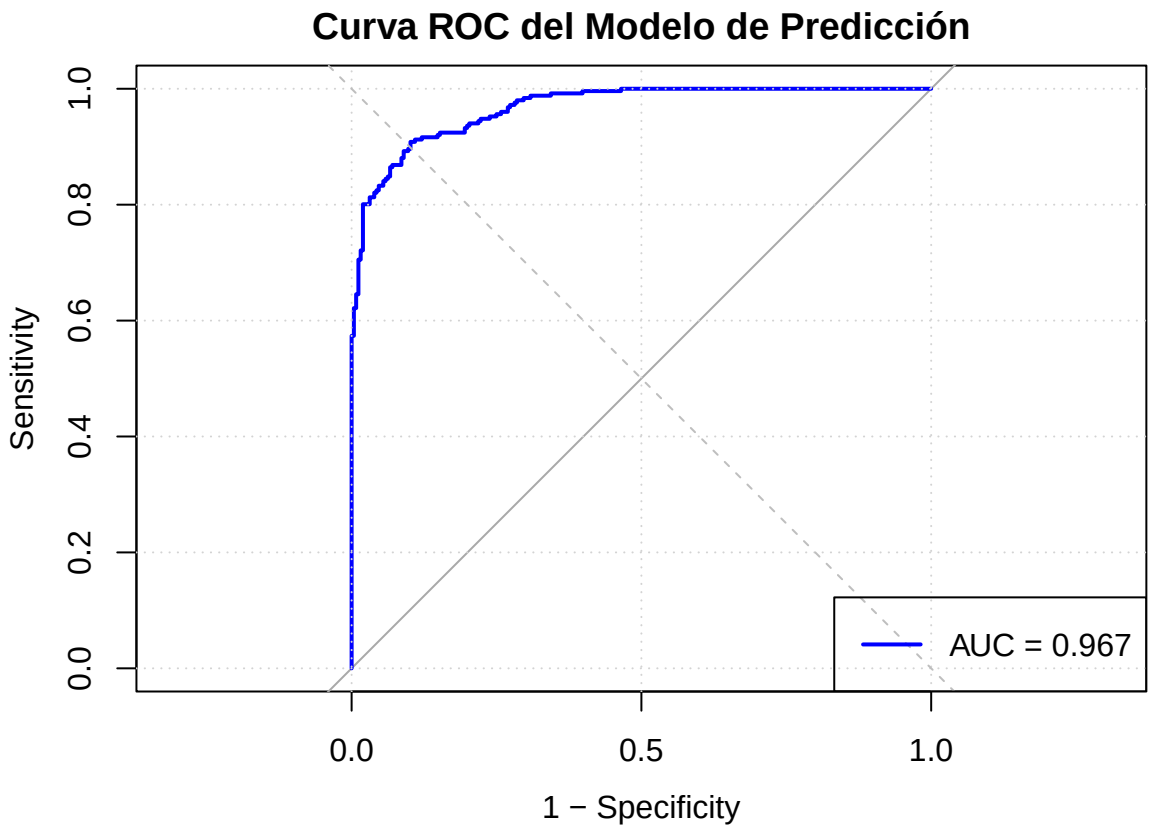
```
##  [1] "Waist.Girth"             "Thigh.Girth"
##  [3] "Calf.Maximum.Girth"      "Knees.diameter"
##  [5] "Biacromial.diameter"     "Knee.Girth"
##  [7] "Navel.Girth"             "Bicep.Girth"
##  [9] "Shoulder.Girth"          "Bitrochanteric.diameter"
## [11] "Chest.diameter"          "Ankle.Minimum.Girth"
## [13] "Chest.Girth"             "Elbows.diameter"
## [15] "Biiliac.diameter"        "Forearm.Girth"
## [17] "Wrist.Minimum.Girth"     "Hip.Girth"
## [19] "Gender"                  "Ankles.diameter"
## [21] "Age"                     "Wrists.diameter"
## [23] "Chest.depth"
```

```r
# Construir el modelo final con los mejores predictores
selected_data <- datos_seleccionados[, c(predictors, "EN")]

# Configurar el control para validación cruzada
train_ctrl <- trainControl(
    method = "LOOCV",
    classProbs = TRUE,
    summaryFunction = twoClassSummary,
    savePredictions = TRUE
)

# Entrenar el modelo final
final_model <- train(
    EN ~ .,
    data = selected_data,
    method = "glm",
    family = binomial,
    trControl = train_ctrl,
    metric = "ROC"
)

# Generar predicciones
pred_probs <- predict(final_model, selected_data, type = "prob")

# Crear y graficar la curva ROC
roc_curve <- roc(response = selected_data$EN,
                 predictor = pred_probs[,"Sobrepeso"],
                 levels = c("NoSobrepeso", "Sobrepeso"))
```

```
## Setting direction: controls < cases
```

```r
# Graficar la curva ROC con detalles
plot(roc_curve,
     main = "Curva ROC del Modelo de Predicción",
     col = "blue",
     lwd = 2,
     legacy.axes = TRUE)
grid()
abline(a = 0, b = 1, lty = 2, col = "gray")
```

```r
# Calcular y añadir el AUC al gráfico
auc_value <- auc(roc_curve)
legend("bottomright",
       legend = sprintf("AUC = %.3f", auc_value),
       col = "blue",
       lwd = 2)
```




```r
# Imprimir resultados detallados
cat("\nResultados del Modelo:\n")
```

```
##
## Resultados del Modelo:
```

```r
cat("------------------------\n")
```

```
## ------------------------
```

```r
cat("AUC:", round(auc_value, 3), "\n")
```

```
## AUC: 0.967
```

```r
cat("Predictores seleccionados:", paste(predictors, collapse = ", "), "\n")
```

```
## Predictores seleccionados: Waist.Girth, Thigh.Girth, Calf.Maximum.Girth, Knees.diameter, Biacromial.
```

```r
# Mostrar matriz de confusión
pred_class <- predict(final_model, selected_data)
conf_matrix <- confusionMatrix(pred_class, selected_data$EN)
print("\nMatriz de Confusión:")
```

```
## [1] "\nMatriz de Confusión:"
```

```r
print(conf_matrix)
```

```
## Confusion Matrix and Statistics
##
##              Reference
## Prediction    NoSobrepeso Sobrepeso
##   NoSobrepeso         232        27
##   Sobrepeso            24       224
##
##                Accuracy : 0.8994
##                  95% CI : (0.8699, 0.9242)
##     No Information Rate : 0.5049
##     P-Value [Acc > NIR] : <2e-16
##
##                   Kappa : 0.7988
##
##  Mcnemar's Test P-Value : 0.7794
##
##             Sensitivity : 0.9062
##             Specificity : 0.8924
##          Pos Pred Value : 0.8958
##          Neg Pred Value : 0.9032
##              Prevalence : 0.5049
##          Detection Rate : 0.4576
##    Detection Prevalence : 0.5108
##       Balanced Accuracy : 0.8993
##
##        'Positive' Class : NoSobrepeso
##
```

Con una exactitud de 0.8994, el modelo clasifíca correctamente aproximadamente el 89.94% de las observaciones. Lo cual indica un alto nivel de precisión.
La Especificidad con un valor de 0.8924 muestra una buena capacidad para identificar a los individuos sin sobrepeso.
Los valores predictivos positivos y negativos ambos con un 90% de predicciones correctas lo que indica que cuando el modelo prediga que es de sobrepeso o no sobrepeso existe una alta probabilidad de que en realidad sea así.