Análisis Multivariado

Producto académico 01

Kevin Heberth Haquehua Apaza

14 de julio del 2025

# Ejercicios Distribución Normal Multivariante e Inferencia

## Ejercicio 1:

Se tiene la matriz de datos

### Solución

library()

En mi caso tengo varios paquetes ya que estuve trabajando tiempo en R

Para mi caso como tengo las ultimas versiones de R, me indica que ambos paquetes ya no los tiene para la versión de R, en el caso de Survival, igualmente hay que ver si se tienen los paquetes

library(MASS)  
library(Survival)

## Error in library(Survival): there is no package called 'Survival'

El caso de la librería Survival no se tiene disponible es en minúscula, así como lo indicaron mis compañeros en el grupo.

library(survival)

Por último veamos la información acerca del paquete Rcmdr

#??Rcmdr

Me envía a la parte de ayuda de R CMD command de los cuales manda como ejemplo lo siguiente

#rcmd(  
# cmd,  
# cmdargs = character(),  
# libpath = .libPaths(),  
# repos = default\_repos(),  
# stdout = NULL,  
# stderr = NULL,  
# poll\_connection = TRUE,  
# echo = FALSE,  
# show = FALSE,  
# callback = NULL,  
# block\_callback = NULL,  
# spinner = show && interactive(),  
# system\_profile = FALSE,  
# user\_profile = "project",  
# env = rcmd\_safe\_env(),  
# timeout = Inf,  
# wd = ".",  
# fail\_on\_status = FALSE  
#)

Código para hacer correr R Comander.

## Ejercicio 3:

A partir del conjunto de datos *anorexia* del paquete MASS, que corresponden a los datos de cambio de peso de pacientes jóvenes con anorexia, mostrad los tipos de datos que contiene y comprobad si existen valores NA y NULL. Para la variable *Treat*, transformad los valores “CBT”, “Cont” y “FT” en “Cogn Beh Tr”, “Contr” y “Fam Tr”, respectivamente

### Solución

Llamamos a la libreria

library(MASS)

Ahora cargamos los datos de anorexia

data("anorexia")

Veamos los primeros registros

head(anorexia)

## Treat Prewt Postwt  
## 1 Cont 80.7 80.2  
## 2 Cont 89.4 80.1  
## 3 Cont 91.8 86.4  
## 4 Cont 74.0 86.3  
## 5 Cont 78.1 76.1  
## 6 Cont 88.3 78.1

Para ver los tipos de datos usamos la función str(anorexia)

str(anorexia)

## 'data.frame': 72 obs. of 3 variables:  
## $ Treat : Factor w/ 3 levels "CBT","Cont","FT": 2 2 2 2 2 2 2 2 2 2 ...  
## $ Prewt : num 80.7 89.4 91.8 74 78.1 88.3 87.3 75.1 80.6 78.4 ...  
## $ Postwt: num 80.2 80.1 86.4 86.3 76.1 78.1 75.1 86.7 73.5 84.6 ...

Se tienen 72 observaciones en los cuales se dividen en tratamiento, wt previo y wt posterior. Ahora comprobemos cuantos valores na se tienen

colSums(is.na(anorexia))

## Treat Prewt Postwt   
## 0 0 0

No se tiene ningún dato vacío, ahora veamos los null

table(is.null(anorexia))

##   
## FALSE   
## 1

Solo se tiene un registro en nulo, por último transformemos la variable *Treat* sus niveles en diferentes nombres

anorexia$Treat <- factor(anorexia$Treat, levels = c("CBT","Cont","FT"), labels = c("Cogn Beh Tr","Contr","Fam Tr"))  
str(anorexia)

## 'data.frame': 72 obs. of 3 variables:  
## $ Treat : Factor w/ 3 levels "Cogn Beh Tr",..: 2 2 2 2 2 2 2 2 2 2 ...  
## $ Prewt : num 80.7 89.4 91.8 74 78.1 88.3 87.3 75.1 80.6 78.4 ...  
## $ Postwt: num 80.2 80.1 86.4 86.3 76.1 78.1 75.1 86.7 73.5 84.6 ...

levels(anorexia$Treat)

## [1] "Cogn Beh Tr" "Contr" "Fam Tr"

## Ejercicio 5:

En el siguiente ejemplo veremos cómo utilizar diferentes operadores sobre el conjunto de datos *birthwt*, así como también algunas funciones que nos permiten obtener más información de las variables:

1. ¿Cuál es la edad máxima de las madres del conjunto de datos?
2. ¿Cuál es la edad mínima de las madres del conjunto de datos?
3. ¿Cuál es el rango de edad de las madres?
4. ¿Fumaba la madre cuyo recién nacido era el de menor peso?
5. ¿Cuánto pesó el recién nacido cuya madre tenía la edad máxima?
6. ¿Listad los pesos de los recién nacidos, cuyas madres visitarán menos de dos veces al médico durante el primer trimestre?

### Solución

1. ¿Cuál es la edad máxima de las madres del conjunto de datos?

data("birthwt") #Leer los datos

max(birthwt$age)

## [1] 45

1. ¿Cuál es la edad mínima de las madres del conjunto de datos?

min(birthwt$age)

## [1] 14

1. ¿Cuál es el rango de edad de las madres?

max(birthwt$age) - min(birthwt$age)

## [1] 31

1. ¿Fumaba la madre cuyo recién nacido era el de menor peso?

library(dplyr) #Usar la librería dplyr

##   
## Attaching package: 'dplyr'

## The following object is masked from 'package:MASS':  
##   
## select

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

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

birthwt %>% filter(bwt == min(bwt)) %>% select(smoke) #Si fumo

## smoke  
## 4 1

1. ¿Cuánto pesó el recién nacido cuya madre tenía la edad máxima?

birthwt %>% filter(age == max(age)) %>% select(bwt)

## bwt  
## 226 4990

1. ¿Listad los pesos de los recién nacidos, cuyas madres visitarán menos de dos veces al médico durante el primer trimestre?

birthwt %>% filter(ftv <= 2) %>% select(bwt)

## bwt  
## 85 2523  
## 87 2557  
## 88 2594  
## 89 2600  
## 91 2622  
## 92 2637  
## 93 2637  
## 94 2663  
## 95 2665  
## 96 2722  
## 97 2733  
## 98 2751  
## 99 2750  
## 100 2769  
## 101 2769  
## 102 2778  
## 104 2807  
## 105 2821  
## 106 2835  
## 108 2836  
## 109 2863  
## 111 2877  
## 112 2877  
## 113 2906  
## 114 2920  
## 115 2920  
## 116 2920  
## 117 2920  
## 118 2948  
## 119 2948  
## 120 2977  
## 121 2977  
## 123 2977  
## 124 2977  
## 125 2922  
## 126 3005  
## 127 3033  
## 128 3042  
## 129 3062  
## 130 3062  
## 131 3062  
## 132 3062  
## 133 3062  
## 135 3090  
## 136 3090  
## 137 3090  
## 138 3100  
## 139 3104  
## 140 3132  
## 141 3147  
## 142 3175  
## 143 3175  
## 144 3203  
## 145 3203  
## 146 3203  
## 147 3225  
## 148 3225  
## 149 3232  
## 150 3232  
## 151 3234  
## 154 3260  
## 155 3274  
## 156 3274  
## 160 3317  
## 161 3317  
## 162 3317  
## 163 3321  
## 164 3331  
## 166 3374  
## 167 3374  
## 168 3402  
## 169 3416  
## 172 3444  
## 173 3459  
## 174 3460  
## 175 3473  
## 176 3544  
## 177 3487  
## 179 3544  
## 180 3572  
## 181 3572  
## 182 3586  
## 183 3600  
## 184 3614  
## 185 3614  
## 186 3629  
## 187 3629  
## 188 3637  
## 189 3643  
## 190 3651  
## 191 3651  
## 192 3651  
## 193 3651  
## 195 3699  
## 196 3728  
## 197 3756  
## 199 3770  
## 200 3770  
## 201 3770  
## 202 3790  
## 203 3799  
## 204 3827  
## 205 3856  
## 207 3860  
## 208 3884  
## 209 3884  
## 210 3912  
## 211 3940  
## 212 3941  
## 213 3941  
## 214 3969  
## 215 3983  
## 216 3997  
## 217 3997  
## 218 4054  
## 219 4054  
## 220 4111  
## 221 4153  
## 222 4167  
## 223 4174  
## 224 4238  
## 225 4593  
## 226 4990  
## 4 709  
## 10 1021  
## 11 1135  
## 13 1330  
## 15 1474  
## 16 1588  
## 17 1588  
## 18 1701  
## 19 1729  
## 20 1790  
## 22 1818  
## 23 1885  
## 24 1893  
## 25 1899  
## 26 1928  
## 27 1928  
## 28 1928  
## 29 1936  
## 30 1970  
## 31 2055  
## 32 2055  
## 33 2082  
## 34 2084  
## 35 2084  
## 36 2100  
## 37 2125  
## 42 2187  
## 43 2187  
## 44 2211  
## 45 2225  
## 46 2240  
## 47 2240  
## 49 2282  
## 50 2296  
## 51 2296  
## 54 2325  
## 56 2353  
## 57 2353  
## 59 2367  
## 60 2381  
## 61 2381  
## 62 2381  
## 63 2410  
## 65 2410  
## 67 2410  
## 69 2424  
## 71 2438  
## 75 2442  
## 77 2466  
## 78 2466  
## 79 2466  
## 81 2495  
## 82 2495  
## 83 2495

## Ejercicio 6:

A partir del conjunto de datos *anorexia* trabajado en apartados anteriores, cread una matriz que tenga como columnas los valores de Prewt y Postwt, y cada fila sean los valores correspondientes para cada posición.

### Solución

matrix(c(anorexia$Prewt, anorexia$Postwt), ncol = 2)

## [,1] [,2]  
## [1,] 80.7 80.2  
## [2,] 89.4 80.1  
## [3,] 91.8 86.4  
## [4,] 74.0 86.3  
## [5,] 78.1 76.1  
## [6,] 88.3 78.1  
## [7,] 87.3 75.1  
## [8,] 75.1 86.7  
## [9,] 80.6 73.5  
## [10,] 78.4 84.6  
## [11,] 77.6 77.4  
## [12,] 88.7 79.5  
## [13,] 81.3 89.6  
## [14,] 78.1 81.4  
## [15,] 70.5 81.8  
## [16,] 77.3 77.3  
## [17,] 85.2 84.2  
## [18,] 86.0 75.4  
## [19,] 84.1 79.5  
## [20,] 79.7 73.0  
## [21,] 85.5 88.3  
## [22,] 84.4 84.7  
## [23,] 79.6 81.4  
## [24,] 77.5 81.2  
## [25,] 72.3 88.2  
## [26,] 89.0 78.8  
## [27,] 80.5 82.2  
## [28,] 84.9 85.6  
## [29,] 81.5 81.4  
## [30,] 82.6 81.9  
## [31,] 79.9 76.4  
## [32,] 88.7 103.6  
## [33,] 94.9 98.4  
## [34,] 76.3 93.4  
## [35,] 81.0 73.4  
## [36,] 80.5 82.1  
## [37,] 85.0 96.7  
## [38,] 89.2 95.3  
## [39,] 81.3 82.4  
## [40,] 76.5 72.5  
## [41,] 70.0 90.9  
## [42,] 80.4 71.3  
## [43,] 83.3 85.4  
## [44,] 83.0 81.6  
## [45,] 87.7 89.1  
## [46,] 84.2 83.9  
## [47,] 86.4 82.7  
## [48,] 76.5 75.7  
## [49,] 80.2 82.6  
## [50,] 87.8 100.4  
## [51,] 83.3 85.2  
## [52,] 79.7 83.6  
## [53,] 84.5 84.6  
## [54,] 80.8 96.2  
## [55,] 87.4 86.7  
## [56,] 83.8 95.2  
## [57,] 83.3 94.3  
## [58,] 86.0 91.5  
## [59,] 82.5 91.9  
## [60,] 86.7 100.3  
## [61,] 79.6 76.7  
## [62,] 76.9 76.8  
## [63,] 94.2 101.6  
## [64,] 73.4 94.9  
## [65,] 80.5 75.2  
## [66,] 81.6 77.8  
## [67,] 82.1 95.5  
## [68,] 77.6 90.7  
## [69,] 83.5 92.5  
## [70,] 89.9 93.8  
## [71,] 86.0 91.7  
## [72,] 87.3 98.0

## Ejercicio 7:

Copiad el código siguiente en tu consola para generar un data frame con veinticinco registros y seis variables, y responde a los siguientes apartados:

Identificador <- c("I1","I2","I3","I4","I5","I6","I7","I8","I9","I10","I11","I12","I13","I14","I15","I16","I17","I18","I19","I20","I21","I22","I23","I24","I25")  
Edad <- c(23,24,21,22,23,25,26,24,21,22,23,25,26,24,22,21,25,26,24,21,25,27,26,22,29)  
Sexo <-c(1,2,1,1,1,2,2,2,1,2,1,2,2,2,1,1,1,2,2,2,1,2,1,1,2) #1 para mujeres y 2 para hombres  
Peso <- c(76.5,81.2,79.3,59.5,67.3,78.6,67.9,100.2,97.8,56.4,65.4,67.5,87.4,99.7,87.6,93.4,65.4,73.7,85.1,61.2,54.8,103.4,65.8,71.7,85.0)  
Alt <- c(165,154,178,165,164,175,182,165,178,165,158,183,184,164,189,167,182,179,165,158,183,184,189,166,175) #altura en cm  
Fuma <- c("SÍ","NO","SÍ","SÍ","NO","NO","NO","SÍ","SÍ","SÍ","NO","NO","SÍ","SÍ","SÍ","SÍ","NO","NO","SÍ","SÍ","SÍ","NO","SÍ","NO","SÍ")  
Trat\_Pulmon <- data.frame(Identificador,Edad,Sexo,Peso,Alt,Fuma)  
head(Trat\_Pulmon)

## Identificador Edad Sexo Peso Alt Fuma  
## 1 I1 23 1 76.5 165 SÍ  
## 2 I2 24 2 81.2 154 NO  
## 3 I3 21 1 79.3 178 SÍ  
## 4 I4 22 1 59.5 165 SÍ  
## 5 I5 23 1 67.3 164 NO  
## 6 I6 25 2 78.6 175 NO

1. Seleccionad los registros con edad > 22.
2. Seleccionad el elemento 3 de la columna 4 del conjunto de datos (contando el identificador)
3. Usad el comando subset() para seleccionar todas las filas que tienen una edad menor que 27 años y sin incluir la columna **Alt**

### Solución

1. Seleccionad los registros con edad > 22.

Trat\_Pulmon %>% filter(Edad > 22)

## Identificador Edad Sexo Peso Alt Fuma  
## 1 I1 23 1 76.5 165 SÍ  
## 2 I2 24 2 81.2 154 NO  
## 3 I5 23 1 67.3 164 NO  
## 4 I6 25 2 78.6 175 NO  
## 5 I7 26 2 67.9 182 NO  
## 6 I8 24 2 100.2 165 SÍ  
## 7 I11 23 1 65.4 158 NO  
## 8 I12 25 2 67.5 183 NO  
## 9 I13 26 2 87.4 184 SÍ  
## 10 I14 24 2 99.7 164 SÍ  
## 11 I17 25 1 65.4 182 NO  
## 12 I18 26 2 73.7 179 NO  
## 13 I19 24 2 85.1 165 SÍ  
## 14 I21 25 1 54.8 183 SÍ  
## 15 I22 27 2 103.4 184 NO  
## 16 I23 26 1 65.8 189 SÍ  
## 17 I25 29 2 85.0 175 SÍ

1. Seleccionad el elemento 3 de la columna 4 del conjunto de datos (contando el identificador)

Trat\_Pulmon[3,4]

## [1] 79.3

1. Usad el comando subset() para seleccionar todas las filas que tienen una edad menor que 27 años y sin incluir la columna **Alt**

subset(Trat\_Pulmon, Edad < 27, select = -c(Alt))

## Identificador Edad Sexo Peso Fuma  
## 1 I1 23 1 76.5 SÍ  
## 2 I2 24 2 81.2 NO  
## 3 I3 21 1 79.3 SÍ  
## 4 I4 22 1 59.5 SÍ  
## 5 I5 23 1 67.3 NO  
## 6 I6 25 2 78.6 NO  
## 7 I7 26 2 67.9 NO  
## 8 I8 24 2 100.2 SÍ  
## 9 I9 21 1 97.8 SÍ  
## 10 I10 22 2 56.4 SÍ  
## 11 I11 23 1 65.4 NO  
## 12 I12 25 2 67.5 NO  
## 13 I13 26 2 87.4 SÍ  
## 14 I14 24 2 99.7 SÍ  
## 15 I15 22 1 87.6 SÍ  
## 16 I16 21 1 93.4 SÍ  
## 17 I17 25 1 65.4 NO  
## 18 I18 26 2 73.7 NO  
## 19 I19 24 2 85.1 SÍ  
## 20 I20 21 2 61.2 SÍ  
## 21 I21 25 1 54.8 SÍ  
## 23 I23 26 1 65.8 SÍ  
## 24 I24 22 1 71.7 NO

## Ejercicio 8:

Incorporad el dataset ***ChickWeight*** que contienen información sobre el peso de 578 pollitos en gramos *(weight)*, el tiempo desde la medición al nacer *(Time)*, una variable identificadora de cada pollito *(Chick)* a partir del rango de peso y una variable factor con el tipo de dieta experimental que cada pollito recibió *(Diet)*.

1. Incorporad el conjunto de datos ***(ChickWeight)*** del paquete **datasets** a vuestro entorno de trabajo.
2. Generad un gráfico de dispersión de la variable ***weight***.
3. Cread un diagrama de caja con la variable ***Time***.

Para más información sobre ***ChickWeight***: <https://rdrr.io/r/datasets/ChickWeight.html>

### Solución

1. Incorporad el conjunto de datos ***(ChickWeight)*** del paquete **datasets** a vuestro entorno de trabajo.

library(datasets)  
data.frame(ChickWeight)

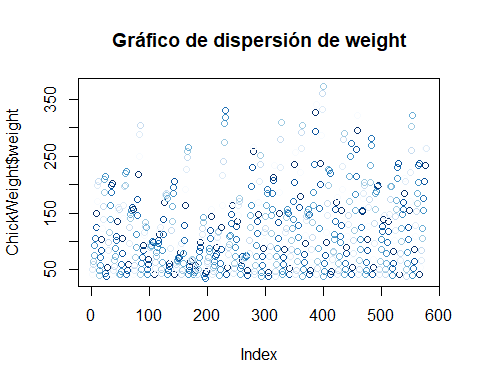
## weight Time Chick Diet  
## 1 42 0 1 1  
## 2 51 2 1 1  
## 3 59 4 1 1  
## 4 64 6 1 1  
## 5 76 8 1 1  
## 6 93 10 1 1  
## 7 106 12 1 1  
## 8 125 14 1 1  
## 9 149 16 1 1  
## 10 171 18 1 1  
## 11 199 20 1 1  
## 12 205 21 1 1  
## 13 40 0 2 1  
## 14 49 2 2 1  
## 15 58 4 2 1  
## 16 72 6 2 1  
## 17 84 8 2 1  
## 18 103 10 2 1  
## 19 122 12 2 1  
## 20 138 14 2 1  
## 21 162 16 2 1  
## 22 187 18 2 1  
## 23 209 20 2 1  
## 24 215 21 2 1  
## 25 43 0 3 1  
## 26 39 2 3 1  
## 27 55 4 3 1  
## 28 67 6 3 1  
## 29 84 8 3 1  
## 30 99 10 3 1  
## 31 115 12 3 1  
## 32 138 14 3 1  
## 33 163 16 3 1  
## 34 187 18 3 1  
## 35 198 20 3 1  
## 36 202 21 3 1  
## 37 42 0 4 1  
## 38 49 2 4 1  
## 39 56 4 4 1  
## 40 67 6 4 1  
## 41 74 8 4 1  
## 42 87 10 4 1  
## 43 102 12 4 1  
## 44 108 14 4 1  
## 45 136 16 4 1  
## 46 154 18 4 1  
## 47 160 20 4 1  
## 48 157 21 4 1  
## 49 41 0 5 1  
## 50 42 2 5 1  
## 51 48 4 5 1  
## 52 60 6 5 1  
## 53 79 8 5 1  
## 54 106 10 5 1  
## 55 141 12 5 1  
## 56 164 14 5 1  
## 57 197 16 5 1  
## 58 199 18 5 1  
## 59 220 20 5 1  
## 60 223 21 5 1  
## 61 41 0 6 1  
## 62 49 2 6 1  
## 63 59 4 6 1  
## 64 74 6 6 1  
## 65 97 8 6 1  
## 66 124 10 6 1  
## 67 141 12 6 1  
## 68 148 14 6 1  
## 69 155 16 6 1  
## 70 160 18 6 1  
## 71 160 20 6 1  
## 72 157 21 6 1  
## 73 41 0 7 1  
## 74 49 2 7 1  
## 75 57 4 7 1  
## 76 71 6 7 1  
## 77 89 8 7 1  
## 78 112 10 7 1  
## 79 146 12 7 1  
## 80 174 14 7 1  
## 81 218 16 7 1  
## 82 250 18 7 1  
## 83 288 20 7 1  
## 84 305 21 7 1  
## 85 42 0 8 1  
## 86 50 2 8 1  
## 87 61 4 8 1  
## 88 71 6 8 1  
## 89 84 8 8 1  
## 90 93 10 8 1  
## 91 110 12 8 1  
## 92 116 14 8 1  
## 93 126 16 8 1  
## 94 134 18 8 1  
## 95 125 20 8 1  
## 96 42 0 9 1  
## 97 51 2 9 1  
## 98 59 4 9 1  
## 99 68 6 9 1  
## 100 85 8 9 1  
## 101 96 10 9 1  
## 102 90 12 9 1  
## 103 92 14 9 1  
## 104 93 16 9 1  
## 105 100 18 9 1  
## 106 100 20 9 1  
## 107 98 21 9 1  
## 108 41 0 10 1  
## 109 44 2 10 1  
## 110 52 4 10 1  
## 111 63 6 10 1  
## 112 74 8 10 1  
## 113 81 10 10 1  
## 114 89 12 10 1  
## 115 96 14 10 1  
## 116 101 16 10 1  
## 117 112 18 10 1  
## 118 120 20 10 1  
## 119 124 21 10 1  
## 120 43 0 11 1  
## 121 51 2 11 1  
## 122 63 4 11 1  
## 123 84 6 11 1  
## 124 112 8 11 1  
## 125 139 10 11 1  
## 126 168 12 11 1  
## 127 177 14 11 1  
## 128 182 16 11 1  
## 129 184 18 11 1  
## 130 181 20 11 1  
## 131 175 21 11 1  
## 132 41 0 12 1  
## 133 49 2 12 1  
## 134 56 4 12 1  
## 135 62 6 12 1  
## 136 72 8 12 1  
## 137 88 10 12 1  
## 138 119 12 12 1  
## 139 135 14 12 1  
## 140 162 16 12 1  
## 141 185 18 12 1  
## 142 195 20 12 1  
## 143 205 21 12 1  
## 144 41 0 13 1  
## 145 48 2 13 1  
## 146 53 4 13 1  
## 147 60 6 13 1  
## 148 65 8 13 1  
## 149 67 10 13 1  
## 150 71 12 13 1  
## 151 70 14 13 1  
## 152 71 16 13 1  
## 153 81 18 13 1  
## 154 91 20 13 1  
## 155 96 21 13 1  
## 156 41 0 14 1  
## 157 49 2 14 1  
## 158 62 4 14 1  
## 159 79 6 14 1  
## 160 101 8 14 1  
## 161 128 10 14 1  
## 162 164 12 14 1  
## 163 192 14 14 1  
## 164 227 16 14 1  
## 165 248 18 14 1  
## 166 259 20 14 1  
## 167 266 21 14 1  
## 168 41 0 15 1  
## 169 49 2 15 1  
## 170 56 4 15 1  
## 171 64 6 15 1  
## 172 68 8 15 1  
## 173 68 10 15 1  
## 174 67 12 15 1  
## 175 68 14 15 1  
## 176 41 0 16 1  
## 177 45 2 16 1  
## 178 49 4 16 1  
## 179 51 6 16 1  
## 180 57 8 16 1  
## 181 51 10 16 1  
## 182 54 12 16 1  
## 183 42 0 17 1  
## 184 51 2 17 1  
## 185 61 4 17 1  
## 186 72 6 17 1  
## 187 83 8 17 1  
## 188 89 10 17 1  
## 189 98 12 17 1  
## 190 103 14 17 1  
## 191 113 16 17 1  
## 192 123 18 17 1  
## 193 133 20 17 1  
## 194 142 21 17 1  
## 195 39 0 18 1  
## 196 35 2 18 1  
## 197 43 0 19 1  
## 198 48 2 19 1  
## 199 55 4 19 1  
## 200 62 6 19 1  
## 201 65 8 19 1  
## 202 71 10 19 1  
## 203 82 12 19 1  
## 204 88 14 19 1  
## 205 106 16 19 1  
## 206 120 18 19 1  
## 207 144 20 19 1  
## 208 157 21 19 1  
## 209 41 0 20 1  
## 210 47 2 20 1  
## 211 54 4 20 1  
## 212 58 6 20 1  
## 213 65 8 20 1  
## 214 73 10 20 1  
## 215 77 12 20 1  
## 216 89 14 20 1  
## 217 98 16 20 1  
## 218 107 18 20 1  
## 219 115 20 20 1  
## 220 117 21 20 1  
## 221 40 0 21 2  
## 222 50 2 21 2  
## 223 62 4 21 2  
## 224 86 6 21 2  
## 225 125 8 21 2  
## 226 163 10 21 2  
## 227 217 12 21 2  
## 228 240 14 21 2  
## 229 275 16 21 2  
## 230 307 18 21 2  
## 231 318 20 21 2  
## 232 331 21 21 2  
## 233 41 0 22 2  
## 234 55 2 22 2  
## 235 64 4 22 2  
## 236 77 6 22 2  
## 237 90 8 22 2  
## 238 95 10 22 2  
## 239 108 12 22 2  
## 240 111 14 22 2  
## 241 131 16 22 2  
## 242 148 18 22 2  
## 243 164 20 22 2  
## 244 167 21 22 2  
## 245 43 0 23 2  
## 246 52 2 23 2  
## 247 61 4 23 2  
## 248 73 6 23 2  
## 249 90 8 23 2  
## 250 103 10 23 2  
## 251 127 12 23 2  
## 252 135 14 23 2  
## 253 145 16 23 2  
## 254 163 18 23 2  
## 255 170 20 23 2  
## 256 175 21 23 2  
## 257 42 0 24 2  
## 258 52 2 24 2  
## 259 58 4 24 2  
## 260 74 6 24 2  
## 261 66 8 24 2  
## 262 68 10 24 2  
## 263 70 12 24 2  
## 264 71 14 24 2  
## 265 72 16 24 2  
## 266 72 18 24 2  
## 267 76 20 24 2  
## 268 74 21 24 2  
## 269 40 0 25 2  
## 270 49 2 25 2  
## 271 62 4 25 2  
## 272 78 6 25 2  
## 273 102 8 25 2  
## 274 124 10 25 2  
## 275 146 12 25 2  
## 276 164 14 25 2  
## 277 197 16 25 2  
## 278 231 18 25 2  
## 279 259 20 25 2  
## 280 265 21 25 2  
## 281 42 0 26 2  
## 282 48 2 26 2  
## 283 57 4 26 2  
## 284 74 6 26 2  
## 285 93 8 26 2  
## 286 114 10 26 2  
## 287 136 12 26 2  
## 288 147 14 26 2  
## 289 169 16 26 2  
## 290 205 18 26 2  
## 291 236 20 26 2  
## 292 251 21 26 2  
## 293 39 0 27 2  
## 294 46 2 27 2  
## 295 58 4 27 2  
## 296 73 6 27 2  
## 297 87 8 27 2  
## 298 100 10 27 2  
## 299 115 12 27 2  
## 300 123 14 27 2  
## 301 144 16 27 2  
## 302 163 18 27 2  
## 303 185 20 27 2  
## 304 192 21 27 2  
## 305 39 0 28 2  
## 306 46 2 28 2  
## 307 58 4 28 2  
## 308 73 6 28 2  
## 309 92 8 28 2  
## 310 114 10 28 2  
## 311 145 12 28 2  
## 312 156 14 28 2  
## 313 184 16 28 2  
## 314 207 18 28 2  
## 315 212 20 28 2  
## 316 233 21 28 2  
## 317 39 0 29 2  
## 318 48 2 29 2  
## 319 59 4 29 2  
## 320 74 6 29 2  
## 321 87 8 29 2  
## 322 106 10 29 2  
## 323 134 12 29 2  
## 324 150 14 29 2  
## 325 187 16 29 2  
## 326 230 18 29 2  
## 327 279 20 29 2  
## 328 309 21 29 2  
## 329 42 0 30 2  
## 330 48 2 30 2  
## 331 59 4 30 2  
## 332 72 6 30 2  
## 333 85 8 30 2  
## 334 98 10 30 2  
## 335 115 12 30 2  
## 336 122 14 30 2  
## 337 143 16 30 2  
## 338 151 18 30 2  
## 339 157 20 30 2  
## 340 150 21 30 2  
## 341 42 0 31 3  
## 342 53 2 31 3  
## 343 62 4 31 3  
## 344 73 6 31 3  
## 345 85 8 31 3  
## 346 102 10 31 3  
## 347 123 12 31 3  
## 348 138 14 31 3  
## 349 170 16 31 3  
## 350 204 18 31 3  
## 351 235 20 31 3  
## 352 256 21 31 3  
## 353 41 0 32 3  
## 354 49 2 32 3  
## 355 65 4 32 3  
## 356 82 6 32 3  
## 357 107 8 32 3  
## 358 129 10 32 3  
## 359 159 12 32 3  
## 360 179 14 32 3  
## 361 221 16 32 3  
## 362 263 18 32 3  
## 363 291 20 32 3  
## 364 305 21 32 3  
## 365 39 0 33 3  
## 366 50 2 33 3  
## 367 63 4 33 3  
## 368 77 6 33 3  
## 369 96 8 33 3  
## 370 111 10 33 3  
## 371 137 12 33 3  
## 372 144 14 33 3  
## 373 151 16 33 3  
## 374 146 18 33 3  
## 375 156 20 33 3  
## 376 147 21 33 3  
## 377 41 0 34 3  
## 378 49 2 34 3  
## 379 63 4 34 3  
## 380 85 6 34 3  
## 381 107 8 34 3  
## 382 134 10 34 3  
## 383 164 12 34 3  
## 384 186 14 34 3  
## 385 235 16 34 3  
## 386 294 18 34 3  
## 387 327 20 34 3  
## 388 341 21 34 3  
## 389 41 0 35 3  
## 390 53 2 35 3  
## 391 64 4 35 3  
## 392 87 6 35 3  
## 393 123 8 35 3  
## 394 158 10 35 3  
## 395 201 12 35 3  
## 396 238 14 35 3  
## 397 287 16 35 3  
## 398 332 18 35 3  
## 399 361 20 35 3  
## 400 373 21 35 3  
## 401 39 0 36 3  
## 402 48 2 36 3  
## 403 61 4 36 3  
## 404 76 6 36 3  
## 405 98 8 36 3  
## 406 116 10 36 3  
## 407 145 12 36 3  
## 408 166 14 36 3  
## 409 198 16 36 3  
## 410 227 18 36 3  
## 411 225 20 36 3  
## 412 220 21 36 3  
## 413 41 0 37 3  
## 414 48 2 37 3  
## 415 56 4 37 3  
## 416 68 6 37 3  
## 417 80 8 37 3  
## 418 83 10 37 3  
## 419 103 12 37 3  
## 420 112 14 37 3  
## 421 135 16 37 3  
## 422 157 18 37 3  
## 423 169 20 37 3  
## 424 178 21 37 3  
## 425 41 0 38 3  
## 426 49 2 38 3  
## 427 61 4 38 3  
## 428 74 6 38 3  
## 429 98 8 38 3  
## 430 109 10 38 3  
## 431 128 12 38 3  
## 432 154 14 38 3  
## 433 192 16 38 3  
## 434 232 18 38 3  
## 435 280 20 38 3  
## 436 290 21 38 3  
## 437 42 0 39 3  
## 438 50 2 39 3  
## 439 61 4 39 3  
## 440 78 6 39 3  
## 441 89 8 39 3  
## 442 109 10 39 3  
## 443 130 12 39 3  
## 444 146 14 39 3  
## 445 170 16 39 3  
## 446 214 18 39 3  
## 447 250 20 39 3  
## 448 272 21 39 3  
## 449 41 0 40 3  
## 450 55 2 40 3  
## 451 66 4 40 3  
## 452 79 6 40 3  
## 453 101 8 40 3  
## 454 120 10 40 3  
## 455 154 12 40 3  
## 456 182 14 40 3  
## 457 215 16 40 3  
## 458 262 18 40 3  
## 459 295 20 40 3  
## 460 321 21 40 3  
## 461 42 0 41 4  
## 462 51 2 41 4  
## 463 66 4 41 4  
## 464 85 6 41 4  
## 465 103 8 41 4  
## 466 124 10 41 4  
## 467 155 12 41 4  
## 468 153 14 41 4  
## 469 175 16 41 4  
## 470 184 18 41 4  
## 471 199 20 41 4  
## 472 204 21 41 4  
## 473 42 0 42 4  
## 474 49 2 42 4  
## 475 63 4 42 4  
## 476 84 6 42 4  
## 477 103 8 42 4  
## 478 126 10 42 4  
## 479 160 12 42 4  
## 480 174 14 42 4  
## 481 204 16 42 4  
## 482 234 18 42 4  
## 483 269 20 42 4  
## 484 281 21 42 4  
## 485 42 0 43 4  
## 486 55 2 43 4  
## 487 69 4 43 4  
## 488 96 6 43 4  
## 489 131 8 43 4  
## 490 157 10 43 4  
## 491 184 12 43 4  
## 492 188 14 43 4  
## 493 197 16 43 4  
## 494 198 18 43 4  
## 495 199 20 43 4  
## 496 200 21 43 4  
## 497 42 0 44 4  
## 498 51 2 44 4  
## 499 65 4 44 4  
## 500 86 6 44 4  
## 501 103 8 44 4  
## 502 118 10 44 4  
## 503 127 12 44 4  
## 504 138 14 44 4  
## 505 145 16 44 4  
## 506 146 18 44 4  
## 507 41 0 45 4  
## 508 50 2 45 4  
## 509 61 4 45 4  
## 510 78 6 45 4  
## 511 98 8 45 4  
## 512 117 10 45 4  
## 513 135 12 45 4  
## 514 141 14 45 4  
## 515 147 16 45 4  
## 516 174 18 45 4  
## 517 197 20 45 4  
## 518 196 21 45 4  
## 519 40 0 46 4  
## 520 52 2 46 4  
## 521 62 4 46 4  
## 522 82 6 46 4  
## 523 101 8 46 4  
## 524 120 10 46 4  
## 525 144 12 46 4  
## 526 156 14 46 4  
## 527 173 16 46 4  
## 528 210 18 46 4  
## 529 231 20 46 4  
## 530 238 21 46 4  
## 531 41 0 47 4  
## 532 53 2 47 4  
## 533 66 4 47 4  
## 534 79 6 47 4  
## 535 100 8 47 4  
## 536 123 10 47 4  
## 537 148 12 47 4  
## 538 157 14 47 4  
## 539 168 16 47 4  
## 540 185 18 47 4  
## 541 210 20 47 4  
## 542 205 21 47 4  
## 543 39 0 48 4  
## 544 50 2 48 4  
## 545 62 4 48 4  
## 546 80 6 48 4  
## 547 104 8 48 4  
## 548 125 10 48 4  
## 549 154 12 48 4  
## 550 170 14 48 4  
## 551 222 16 48 4  
## 552 261 18 48 4  
## 553 303 20 48 4  
## 554 322 21 48 4  
## 555 40 0 49 4  
## 556 53 2 49 4  
## 557 64 4 49 4  
## 558 85 6 49 4  
## 559 108 8 49 4  
## 560 128 10 49 4  
## 561 152 12 49 4  
## 562 166 14 49 4  
## 563 184 16 49 4  
## 564 203 18 49 4  
## 565 233 20 49 4  
## 566 237 21 49 4  
## 567 41 0 50 4  
## 568 54 2 50 4  
## 569 67 4 50 4  
## 570 84 6 50 4  
## 571 105 8 50 4  
## 572 122 10 50 4  
## 573 155 12 50 4  
## 574 175 14 50 4  
## 575 205 16 50 4  
## 576 234 18 50 4  
## 577 264 20 50 4  
## 578 264 21 50 4

head(ChickWeight)

## weight Time Chick Diet  
## 1 42 0 1 1  
## 2 51 2 1 1  
## 3 59 4 1 1  
## 4 64 6 1 1  
## 5 76 8 1 1  
## 6 93 10 1 1

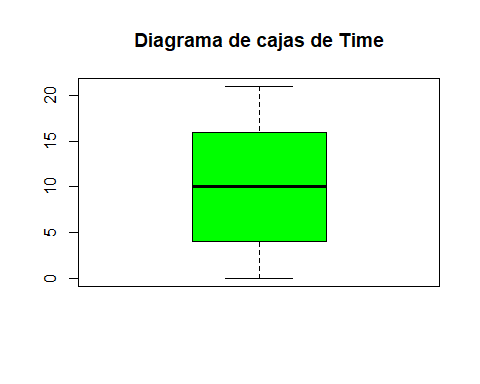
1. Generad un gráfico de dispersión de la variable ***weight***.

plot(ChickWeight$weight, col=blues9, main="Gráfico de dispersión de weight")



1. Cread un diagrama de caja con la variable ***Time***.

boxplot(ChickWeight$Time, col="green", main="Diagrama de cajas de Time")



## Ejercicio 9:

A partir del conjunto de datos *anorexia* del paquete MASS, cread otro data frame que se llame **anorexia\_treat\_df** formado por Treat y por un vector nuevo calculado a partir de la diferencia *Prewt-Postwt*. De esta manera, nos quedará un dataframe que contenga el tipo de tratamiento y el valor del peso ganado o perdido después de haber realizado el tratamiento.

Seleccionad aquellos individuos que han ganado peso después del tratamiento y cread un nuevo conjunto llamado ***anorexia\_treat\_C\_df*** que contenga solo los datos de aquellos que han seguido el tratamiento “Cont” y que han ganado peso después del tratamiento.

### Solución

Cread la variable de la diferencia

anorexia$dif <- anorexia$Prewt - anorexia$Postwt

Ahora seleccionar la variable Treat y la diferencia

anorexia\_treat\_df <- anorexia %>% select(Treat, dif)  
head(anorexia\_treat\_df)

## Treat dif  
## 1 Contr 0.5  
## 2 Contr 9.3  
## 3 Contr 5.4  
## 4 Contr -12.3  
## 5 Contr 2.0  
## 6 Contr 10.2

Seleccionad a los individuos que ganaron peso después del tratamiento y cread el dataframe

anorexia\_treat\_C\_df <- anorexia %>% filter(Treat == "Contr")  
head(anorexia\_treat\_C\_df)

## Treat Prewt Postwt dif  
## 1 Contr 80.7 80.2 0.5  
## 2 Contr 89.4 80.1 9.3  
## 3 Contr 91.8 86.4 5.4  
## 4 Contr 74.0 86.3 -12.3  
## 5 Contr 78.1 76.1 2.0  
## 6 Contr 88.3 78.1 10.2

## Ejercicio 10:

Entrad en <https://rpubs.com/> y registraros. Crearos un perfil y subid un documento **R Markdown**. Los prerequisitos son tener instalados <https://www.r-project.org/> y <https://posit.co/>, y el paquete knitr.

Pasos que tenéis que seguir para publicar vuestro documento:

1. En **RStudio**, cread un documento **R Markdown**.
2. Generad el documento con ***Knit***.
3. En la ventana de previsualización, clicad el botón de publicar.

Como solución de vuestro ejercicio, copiad el enlace de vuestra página de prueba de **RPubs**

### Solución

Enlace de la publicación de este trabajo <https://rpubs.com/UOC72658015/1282283>

## Caso práctico

Resolved los siguientes apartados

1. Cread un conjunto de datos inventado con R. Debe contener treinta observaciones (quince para hombres y quince para mujeres) para seis variables con estas características

| Variable | Nombre | Características |
| --- | --- | --- |
| Identificador | Id | carácter |
| Edad | Edad | numérica |
| Genero | Gene | 2 valores 1 = mujer, 2 = hombre |
| Tratamiento | Trat | Factor. Tres tipos de tratamiento (A, B y C) |
| Peso | Peso | numérica (en kg) |
| Estatura | Alt | numérica (en cm) |

1. Buscad información de vuestro conjunto de datos y de vuestras variables.
2. Cread una nueva variable a partir de alguna de las que tengamos. Por ejemplo, podéis calcular el IMC **(IMC) = peso (kg)/ [estatura (m)]** e incluid la nueva variable en el conjunto de datos.
3. Cread dos data frames diferenciados para hombres y mujeres con dos nombres diferentes: **Df\_Hombres** y **Df\_Mujeres**.
4. Combinad de nuevo los dos ficheros anteriores y cread el primero de nuevo con el comando rbind().

### Solución

1. Cread un conjunto de datos inventado con R. Debe contener treinta observaciones (quince para hombres y quince para mujeres) para seis variables con estas características

set.seed(2025) #Semilla para que los datos sean iguales

Creamos las variables

Id <- c("ID1","ID2","ID3","ID4","ID5","ID6","ID7","ID8","ID9","ID10","ID11","ID12","ID13","ID14","ID15","ID16","ID17","ID18","ID19","ID20","ID21","ID22","ID23","ID24","ID25","ID26","ID27","ID28","ID29","ID30")  
Edad <- round(rnorm(n = 30, mean = 22, sd = 3),0)  
Genero <- rbinom(n = 30, size = 1, prob = 0.5)  
Tratamiento <- rbinom(n = 30, size = 2, prob = 0.5)  
Peso <- round(rnorm(n = 30, mean = 70, sd = 6),2)  
Estatura <- round(rnorm(n = 30, mean = 172, sd = 7),2)

data <- data.frame(Id, Edad, Genero, Tratamiento,  
 Peso, Estatura)  
data$Genero <- ifelse(data$Genero == 0, 1,2)  
data$Tratamiento <- ifelse(data$Tratamiento == 0, "A",  
 ifelse(data$Tratamiento == 1, "B","C"))

head(data)

## Id Edad Genero Tratamiento Peso Estatura  
## 1 ID1 24 1 C 73.57 176.38  
## 2 ID2 22 2 C 73.83 165.67  
## 3 ID3 24 2 B 72.88 184.01  
## 4 ID4 26 1 C 78.24 172.63  
## 5 ID5 23 1 B 63.11 169.02  
## 6 ID6 22 2 B 55.30 164.63

1. Buscad información de vuestro conjunto de datos y de vuestras variables.

data$Genero <- as.factor(data$Genero)  
data$Tratamiento <- as.factor(data$Tratamiento)

str(data)

## 'data.frame': 30 obs. of 6 variables:  
## $ Id : chr "ID1" "ID2" "ID3" "ID4" ...  
## $ Edad : num 24 22 24 26 23 22 23 22 21 24 ...  
## $ Genero : Factor w/ 2 levels "1","2": 1 2 2 1 1 2 2 1 1 1 ...  
## $ Tratamiento: Factor w/ 3 levels "A","B","C": 3 3 2 3 2 2 3 3 2 2 ...  
## $ Peso : num 73.6 73.8 72.9 78.2 63.1 ...  
## $ Estatura : num 176 166 184 173 169 ...

summary(data)

## Id Edad Genero Tratamiento Peso   
## Length:30 Min. :17.0 1:17 A: 6 Min. :55.30   
## Class :character 1st Qu.:21.0 2:13 B:12 1st Qu.:64.70   
## Mode :character Median :22.0 C:12 Median :67.58   
## Mean :22.6 Mean :69.05   
## 3rd Qu.:24.0 3rd Qu.:73.77   
## Max. :31.0 Max. :80.30   
## Estatura   
## Min. :158.7   
## 1st Qu.:164.9   
## Median :170.3   
## Mean :170.4   
## 3rd Qu.:174.2   
## Max. :184.0

1. Cread una nueva variable a partir de alguna de las que tengamos. Por ejemplo, podéis calcular el IMC **(IMC = peso (kg)/ [estatura (m)]** e incluid la nueva variable en el conjunto de datos.

data$Estatura <- data$Estatura/100  
  
data$IMC <- (data$Peso) / (data$Estatura^2)

Veamos los registros

data$IMC

## [1] 23.64842 26.89956 21.52413 26.25405 22.09134 20.40362 25.49995 25.91605  
## [9] 29.05308 22.43169 22.21060 22.61272 20.24045 18.60216 25.22801 25.84181  
## [17] 25.06343 23.73729 21.56068 20.24197 23.63623 27.60687 24.69934 25.27390  
## [25] 23.92997 21.52754 29.82913 21.73353 21.92616 26.43915

1. Cread dos data frames diferenciados para hombres y mujeres con dos nombres diferentes: **Df\_Hombres** y **Df\_Mujeres**.

Df\_Hombres <- data %>% filter(Genero == 2)  
Df\_Hombres

## Id Edad Genero Tratamiento Peso Estatura IMC  
## 1 ID2 22 2 C 73.83 1.6567 26.89956  
## 2 ID3 24 2 B 72.88 1.8401 21.52413  
## 3 ID6 22 2 B 55.30 1.6463 20.40362  
## 4 ID7 23 2 C 64.58 1.5914 25.49995  
## 5 ID12 17 2 C 65.52 1.7022 22.61272  
## 6 ID13 21 2 A 65.07 1.7930 20.24045  
## 7 ID17 22 2 C 66.73 1.6317 25.06343  
## 8 ID18 23 2 B 79.06 1.8250 23.73729  
## 9 ID19 18 2 B 65.54 1.7435 21.56068  
## 10 ID20 29 2 C 66.00 1.8057 20.24197  
## 11 ID21 21 2 C 63.27 1.6361 23.63623  
## 12 ID23 31 2 B 71.23 1.6982 24.69934  
## 13 ID30 24 2 A 76.05 1.6960 26.43915

Df\_Mujeres <- data %>% filter(Genero == 1)  
Df\_Mujeres

## Id Edad Genero Tratamiento Peso Estatura IMC  
## 1 ID1 24 1 C 73.57 1.7638 23.64842  
## 2 ID4 26 1 C 78.24 1.7263 26.25405  
## 3 ID5 23 1 B 63.11 1.6902 22.09134  
## 4 ID8 22 1 C 67.62 1.6153 25.91605  
## 5 ID9 21 1 B 80.30 1.6625 29.05308  
## 6 ID10 24 1 B 67.54 1.7352 22.43169  
## 7 ID11 21 1 B 69.45 1.7683 22.21060  
## 8 ID14 24 1 C 56.10 1.7366 18.60216  
## 9 ID15 25 1 B 66.79 1.6271 25.22801  
## 10 ID16 21 1 C 75.98 1.7147 25.84181  
## 11 ID22 24 1 A 78.82 1.6897 27.60687  
## 12 ID24 24 1 C 63.63 1.5867 25.27390  
## 13 ID25 22 1 A 69.99 1.7102 23.92997  
## 14 ID26 18 1 A 69.85 1.8013 21.52754  
## 15 ID27 19 1 A 77.83 1.6153 29.82913  
## 16 ID28 22 1 B 64.05 1.7167 21.73353  
## 17 ID29 21 1 B 63.71 1.7046 21.92616

1. Combinad de nuevo los dos ficheros anteriores y cread el primero de nuevo con el comando rbind().

data\_bind <- rbind(Df\_Hombres, Df\_Mujeres)  
data\_bind

## Id Edad Genero Tratamiento Peso Estatura IMC  
## 1 ID2 22 2 C 73.83 1.6567 26.89956  
## 2 ID3 24 2 B 72.88 1.8401 21.52413  
## 3 ID6 22 2 B 55.30 1.6463 20.40362  
## 4 ID7 23 2 C 64.58 1.5914 25.49995  
## 5 ID12 17 2 C 65.52 1.7022 22.61272  
## 6 ID13 21 2 A 65.07 1.7930 20.24045  
## 7 ID17 22 2 C 66.73 1.6317 25.06343  
## 8 ID18 23 2 B 79.06 1.8250 23.73729  
## 9 ID19 18 2 B 65.54 1.7435 21.56068  
## 10 ID20 29 2 C 66.00 1.8057 20.24197  
## 11 ID21 21 2 C 63.27 1.6361 23.63623  
## 12 ID23 31 2 B 71.23 1.6982 24.69934  
## 13 ID30 24 2 A 76.05 1.6960 26.43915  
## 14 ID1 24 1 C 73.57 1.7638 23.64842  
## 15 ID4 26 1 C 78.24 1.7263 26.25405  
## 16 ID5 23 1 B 63.11 1.6902 22.09134  
## 17 ID8 22 1 C 67.62 1.6153 25.91605  
## 18 ID9 21 1 B 80.30 1.6625 29.05308  
## 19 ID10 24 1 B 67.54 1.7352 22.43169  
## 20 ID11 21 1 B 69.45 1.7683 22.21060  
## 21 ID14 24 1 C 56.10 1.7366 18.60216  
## 22 ID15 25 1 B 66.79 1.6271 25.22801  
## 23 ID16 21 1 C 75.98 1.7147 25.84181  
## 24 ID22 24 1 A 78.82 1.6897 27.60687  
## 25 ID24 24 1 C 63.63 1.5867 25.27390  
## 26 ID25 22 1 A 69.99 1.7102 23.92997  
## 27 ID26 18 1 A 69.85 1.8013 21.52754  
## 28 ID27 19 1 A 77.83 1.6153 29.82913  
## 29 ID28 22 1 B 64.05 1.7167 21.73353  
## 30 ID29 21 1 B 63.71 1.7046 21.92616