Instituto Tecnológico de Costa Rica

Proyecto de Estadística

Integrantes:

Jose Ignacio Alfaro Solano

Marco Antonio Espinoza Salas

Warner Fidel Hurtado Laguna

Lester Alexander Trejos Bermúdez

Profesor:

Esteban Ballestero Alfaro

Fecha:

12/05/2021

### Nombre de la base de datos

nycflights13

### Descripción del dataset seleccionado

La base utilizada para el proyecto contiene información sobre todos los vuelos que salían de NUEVA YORK (por ejemplo EWR, JFK Y LGA)a destinos en los Estados Unidos, Puerto Rico y las Islas Vírgenes Americanas) en 2013: 336,776 vuelos en total. Para ayudar a comprender qué provoca retrasos, también incluye otros conjuntos de datos útiles. La base que está siendo utilizada contiene muchas tablas de datos como vuelos, climas, aviones, aeropuertos etc, para el desarrollo del presente proyecto se eligió la columna flights o vuelos.

El paquete flights o vuelos se divide en varias columnas como año, hora de salida, minutos, avión, origen destino y muchos otros. Para el manejo de este proyecto se trabajó con tres columnas respectivas, la de origen, destino y duración en el aire, las cuales fueron utilizadas en pruebas de una y dos poblaciones. Para la prueba se tomaron un origen y un destino en el cual se toma la duración entre ese respectivo origen y destino.

### Librerías necesarias

if (!require(‘devtools’)) install.packages(‘devtools’) devtools::install\_github(‘fhernanb/stests’, force=TRUE)

library(nycflights13)

## Warning: package 'nycflights13' was built under R version 4.0.5

library(mosaicData)

## Warning: package 'mosaicData' was built under R version 4.0.5

library(PASWR2)

## Warning: package 'PASWR2' was built under R version 4.0.5

## Loading required package: lattice

## Loading required package: ggplot2

## Warning: package 'ggplot2' was built under R version 4.0.4

library(stests)

##   
## Attaching package: 'stests'

## The following object is masked from 'package:PASWR2':  
##   
## z.test

## The following object is masked from 'package:stats':  
##   
## var.test

library(BSDA)

## Warning: package 'BSDA' was built under R version 4.0.5

##   
## Attaching package: 'BSDA'

## The following object is masked from 'package:stests':  
##   
## z.test

## The following objects are masked from 'package:PASWR2':  
##   
## normarea, nsize, ntester, SIGN.test, tsum.test, z.test, zsum.test

## The following object is masked from 'package:mosaicData':  
##   
## Alcohol

## The following object is masked from 'package:datasets':  
##   
## Orange

library(visdat)

## Warning: package 'visdat' was built under R version 4.0.4

library(remotes)

## Warning: package 'remotes' was built under R version 4.0.5

library(DataExplorer)

## Warning: package 'DataExplorer' was built under R version 4.0.4

library(MASS)  
library(TeachingDemos)

## Warning: package 'TeachingDemos' was built under R version 4.0.5

##   
## Attaching package: 'TeachingDemos'

## The following object is masked from 'package:BSDA':  
##   
## z.test

## The following object is masked from 'package:stests':  
##   
## z.test

## The following object is masked from 'package:PASWR2':  
##   
## z.test

library(ggplot2)  
library(datos)

## Warning: package 'datos' was built under R version 4.0.4

library(fBasics)

## Warning: package 'fBasics' was built under R version 4.0.5

## Loading required package: timeDate

## Warning: package 'timeDate' was built under R version 4.0.5

## Loading required package: timeSeries

## Warning: package 'timeSeries' was built under R version 4.0.5

library(nortest)  
library(EnvStats)

## Warning: package 'EnvStats' was built under R version 4.0.5

##   
## Attaching package: 'EnvStats'

## The following object is masked from 'package:timeSeries':  
##   
## print

## The following objects are masked from 'package:timeDate':  
##   
## kurtosis, skewness

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

## The following objects are masked from 'package:stats':  
##   
## predict, predict.lm

## The following object is masked from 'package:base':  
##   
## print.default

data("flights")  
  
mosaicData <- mosaicData::KidsFeet  
  
dataFlight <- flights  
dataDiamonds <- ggplot2::diamonds  
  
dataFlight

## # A tibble: 336,776 x 19  
## year month day dep\_time sched\_dep\_time dep\_delay arr\_time sched\_arr\_time  
## <int> <int> <int> <int> <int> <dbl> <int> <int>  
## 1 2013 1 1 517 515 2 830 819  
## 2 2013 1 1 533 529 4 850 830  
## 3 2013 1 1 542 540 2 923 850  
## 4 2013 1 1 544 545 -1 1004 1022  
## 5 2013 1 1 554 600 -6 812 837  
## 6 2013 1 1 554 558 -4 740 728  
## 7 2013 1 1 555 600 -5 913 854  
## 8 2013 1 1 557 600 -3 709 723  
## 9 2013 1 1 557 600 -3 838 846  
## 10 2013 1 1 558 600 -2 753 745  
## # ... with 336,766 more rows, and 11 more variables: arr\_delay <dbl>,  
## # carrier <chr>, flight <int>, tailnum <chr>, origin <chr>, dest <chr>,  
## # air\_time <dbl>, distance <dbl>, hour <dbl>, minute <dbl>, time\_hour <dttm>

mosaicData

## name birthmonth birthyear length width sex biggerfoot domhand  
## 1 David 5 88 24.4 8.4 B L R  
## 2 Lars 10 87 25.4 8.8 B L L  
## 3 Zach 12 87 24.5 9.7 B R R  
## 4 Josh 1 88 25.2 9.8 B L R  
## 5 Lang 2 88 25.1 8.9 B L R  
## 6 Scotty 3 88 25.7 9.7 B R R  
## 7 Edward 2 88 26.1 9.6 B L R  
## 8 Caitlin 6 88 23.0 8.8 G L R  
## 9 Eleanor 5 88 23.6 9.3 G R R  
## 10 Damon 9 88 22.9 8.8 B R L  
## 11 Mark 9 87 27.5 9.8 B R R  
## 12 Ray 3 88 24.8 8.9 B L R  
## 13 Cal 8 87 26.1 9.1 B L R  
## 14 Cam 3 88 27.0 9.8 B L R  
## 15 Julie 11 87 26.0 9.3 G L R  
## 16 Kate 4 88 23.7 7.9 G R R  
## 17 Caroline 12 87 24.0 8.7 G R L  
## 18 Maggie 3 88 24.7 8.8 G R R  
## 19 Lee 6 88 26.7 9.0 G L L  
## 20 Heather 3 88 25.5 9.5 G R R  
## 21 Andy 6 88 24.0 9.2 B R R  
## 22 Josh 7 88 24.4 8.6 B L R  
## 23 Laura 9 88 24.0 8.3 G R L  
## 24 Erica 9 88 24.5 9.0 G L R  
## 25 Peggy 10 88 24.2 8.1 G L R  
## 26 Glen 7 88 27.1 9.4 B L R  
## 27 Abby 2 88 26.1 9.5 G L R  
## 28 David 12 87 25.5 9.5 B R R  
## 29 Mike 11 88 24.2 8.9 B L R  
## 30 Dwayne 8 88 23.9 9.3 B R L  
## 31 Danielle 6 88 24.0 9.3 G L R  
## 32 Caitlin 7 88 22.5 8.6 G R R  
## 33 Leigh 3 88 24.5 8.6 G L R  
## 34 Dylan 4 88 23.6 9.0 B R L  
## 35 Peter 4 88 24.7 8.6 B R L  
## 36 Hannah 3 88 22.9 8.5 G L R  
## 37 Teshanna 3 88 26.0 9.0 G L R  
## 38 Hayley 1 88 21.6 7.9 G R R  
## 39 Alisha 9 88 24.6 8.8 G L R

#if (!require('devtools')) install.packages('devtools')  
#devtools::install\_github('fhernanb/stests', force=TRUE)

### Variables de la base datos

dataFlight

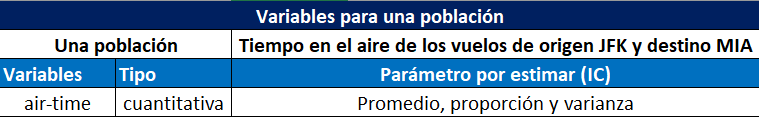
## # A tibble: 336,776 x 19  
## year month day dep\_time sched\_dep\_time dep\_delay arr\_time sched\_arr\_time  
## <int> <int> <int> <int> <int> <dbl> <int> <int>  
## 1 2013 1 1 517 515 2 830 819  
## 2 2013 1 1 533 529 4 850 830  
## 3 2013 1 1 542 540 2 923 850  
## 4 2013 1 1 544 545 -1 1004 1022  
## 5 2013 1 1 554 600 -6 812 837  
## 6 2013 1 1 554 558 -4 740 728  
## 7 2013 1 1 555 600 -5 913 854  
## 8 2013 1 1 557 600 -3 709 723  
## 9 2013 1 1 557 600 -3 838 846  
## 10 2013 1 1 558 600 -2 753 745  
## # ... with 336,766 more rows, and 11 more variables: arr\_delay <dbl>,  
## # carrier <chr>, flight <int>, tailnum <chr>, origin <chr>, dest <chr>,  
## # air\_time <dbl>, distance <dbl>, hour <dbl>, minute <dbl>, time\_hour <dttm>

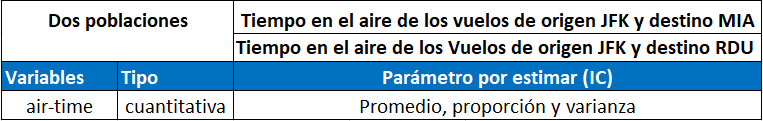
dataDiamonds

## # A tibble: 53,940 x 10  
## carat cut color clarity depth table price x y z  
## <dbl> <ord> <ord> <ord> <dbl> <dbl> <int> <dbl> <dbl> <dbl>  
## 1 0.23 Ideal E SI2 61.5 55 326 3.95 3.98 2.43  
## 2 0.21 Premium E SI1 59.8 61 326 3.89 3.84 2.31  
## 3 0.23 Good E VS1 56.9 65 327 4.05 4.07 2.31  
## 4 0.290 Premium I VS2 62.4 58 334 4.2 4.23 2.63  
## 5 0.31 Good J SI2 63.3 58 335 4.34 4.35 2.75  
## 6 0.24 Very Good J VVS2 62.8 57 336 3.94 3.96 2.48  
## 7 0.24 Very Good I VVS1 62.3 57 336 3.95 3.98 2.47  
## 8 0.26 Very Good H SI1 61.9 55 337 4.07 4.11 2.53  
## 9 0.22 Fair E VS2 65.1 61 337 3.87 3.78 2.49  
## 10 0.23 Very Good H VS1 59.4 61 338 4 4.05 2.39  
## # ... with 53,930 more rows

### Resumen de variables seleccionadas

#### La siguiente variable fue utilizada para una y dos poblaciones, en donde se utilizaba la duración en aire de un destino al otro.

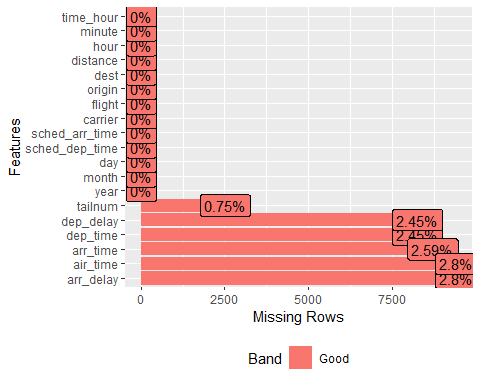
 -



Variables para una y dos poblaciones

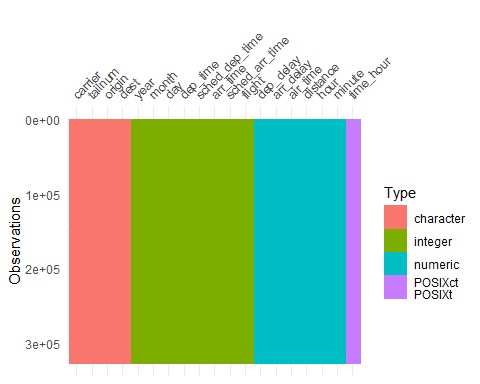
### Gráficos

dataFlight <- flights  
plot\_missing(dataFlight)



El gráfico anterior nos demuestra que la escogencia de los datos está basada en una base datos confiable, ya que como se puede observar, los datos presentan un buen comportamiento. También, se presentan datos en blanco, pero el porcentaje es muy pequeño lo cual hace que el dataset sea óptimo para hacer cálculos estadísticos.

dataFlight <- flights  
vuelos <- na.omit(dataFlight)  
vis\_dat(vuelos,warn\_large\_data = FALSE)



En el gráfico anterior, se puede observar los tipos de variables, el nombre que cada una tiene, así como la cantidad de datos que contienen. Cabe resaltar que para este gráfico se eliminaron los valores en blanco.

# Pruebas de intervalos de confianza para una población

## **Distribución normal Z**

En la siguiente prueba se procederá a realizar los respectivos procedimientos con la distribución normal z, en la cual se tomará un nivel de confianza del 95%. Se tomaron datos de una población con un comportamiento normal. Se tomaron los tiempos en vuelo de todos los vuelos de origen JFK y destino MIA, este es el valor de ‘x’ en la función. Para el valor del sigma.x se tomó la desviación estándar de esos tiempos en aire de los vuelos y el conf.level significa el nivel de confianza para el intervalo, que es nuestro caso es de 95%. Para esta prueba se usó la librería ‘BSDA’.

### **Resultados**

#En la siguiente asignación se toman la cantidad del tiempo de aire de los vuelos que salen de JFK y tienen como destino MIA, y guarda en var1 sólo los vuelos con dichas características.  
var1 <- na.omit(dataFlight[dataFlight$origin == "JFK" & dataFlight$dest == "MIA", ]$air\_time)  
  
#En esta variable se trabaja con el tiempo en el aire de el origen y destino seleccionado anteriormente.  
  
#Se procede a aplicar la función de PASWR2, en la cual se le pasan como parametros la lista de tiempos, la desviación estandar y por último el nivel de significancia.  
BSDA:: z.test(x = var1, sigma.x = sd(var1), conf.level = 0.95)$conf.int

## [1] 151.7496 152.5103  
## attr(,"conf.level")  
## [1] 0.95

### **Conclusión**

Al realizar los respectivos cálculos de la distribución normal z con ’3285’que es cantidad de tiempos de vuelos con origen JFK y destino a MIA se obtuvo un intervalo de confianza de 95% para el promedio del tiempo de vuelo desde JFK a MIA dado por: ]151.7496, 152.5103[.

## **Distribución T**

En la siguiente prueba se procederá a realizar los respectivos cálculos con la distribución TStudent, en la cual se tomará un nivel de confianza del 95%. Esta distribución debe tener un n < 30, por lo que se tomó una muestra aleatoria de tamaño 25, correspondientes a la columna de *air\_time* para dicha prueba. Cabe destacar que la población presenta un compartamiento normal. Para esta prueba se usó la librería ‘stats’.

### **Resultados**

#En la siguiente asignación se toman las n cantidad de vuelos que salen de JFK y tienen como destino MIA, y guarda en var1 sólo los vuelos con dichas características.  
  
#En esta variable se trabaja con el tiempo (air-time) en el aire de el origen y destino seleccionados el cual se van a utilizar 25 datos de manera aleatoria  
var1 <- sample(x = na.omit(dataFlight[dataFlight$origin == "JFK" & dataFlight$dest == "MIA", ]$air\_time), size = 25, replace = TRUE)  
  
#Se procede a aplicar la función T, en la cual se le pasan como parametros la lista de tiempos y por último el nivel de significancia.  
stats::t.test(var1, conf.level = 0.95)$conf.int

## [1] 145.4168 155.6232  
## attr(,"conf.level")  
## [1] 0.95

### **Conclusión**

Al realizar los respectivos cálculos de la distribución TStudent con una muestra aleatoria de tamaño 25,que representa la cantidad de tiempos de vuelos con origen JFK y destino a MIA se obtuvo un intervalo de confianza de 95% dado por: ]146.9743, 157.1857[.

## **Una proporción**

Para realizar el respectivo cálculo de la proporción de tiempos de vuelo se tomaron todos los vuelos menores o iguales que 150, en la cual se tomará un nivel de confianza del 95%. La población sería los tiempos de vuelos, y como “éxitos” los vuelos con un tiempo de vuelo menor a 150. La población se comporta de manera normal. Cabe destacar que para esta prueba se usó la librería ‘stats’.

### **Resultados**

#En la siguiente asignación se toman las n cantidad de vuelos que salen de JFK y tienen como destino MIA, y guarda en var1 sólo los vuelos con dichas características.  
var1 <- dataFlight[dataFlight$origin == "JFK" & dataFlight$dest == "MIA", ]  
  
#En la siguiente variable se asignan los éxitos obtenidos de la condición n <= 100  
exitos <- var1[which(na.omit(var1$air\_time) <= 150), ]  
  
#En esta variable se trabaja con el tiempo en el aire de el origen y destino seleccionado anteriormente  
var1 <- na.omit(var1$air\_time)  
  
  
#Se obtiene la cantidad total de éxitos de la variable anterior éxitos  
totalExitos <- length(exitos$air\_time)  
  
  
#Para obtener la proporción se procede a ingresar el total de éxitos, el total de todas las variables y por último el nivel de confianza  
stats::prop.test(totalExitos, length(var1), conf.level = 0.95)$conf.int

## [1] 0.4677168 0.5021818  
## attr(,"conf.level")  
## [1] 0.95

### **Conclusión**

Al realizar los respectivos cálculos de la proporción con la cantidad de tiempos de vuelos de origen JFK a destino MIA menores a 150 minutos, se obtuvo un intervalo de confianza de 95%, dado por: ]0.4677168, 0.5021818[.

## **Varianza**

En la última prueba para una población, se procede a realizar los cálculos con un intervalo de confianza del 95%, en la cual se utiliza la cantidad total de las variables *air\_time* para una mejor estimación. Para realizar la siguiente estimación del intervalo de confianza se tomó una población que sigue una distribución normal. Cabe resaltar que para esta prueba se usó la librería ‘stests’.

### **Resultados**

#En la siguiente asignación se toman los n cantidad de vuelos que salen de JFK y tienen como destino MIA, y guarda en var1 sólo los vuelos con dichas características.  
var1 <- na.omit(dataFlight[dataFlight$origin == "JFK" & dataFlight$dest == "MIA", ]$air\_time)  
  
#En la siguiente función de stests se procede a ingresar la lista de los tiempos y luego el nivel de confianza  
stests:: var.test(x = var1, conf.level = 0.95)$conf.int

## [1] 117.9459 129.9276  
## attr(,"conf.level")  
## [1] 0.95

### **Conclusión**

Al realizar el análisis de la variancia mediante la estimación por intervalo de confianza de 95% para los tiempos de vuelos de origen JFK a destino MIA se obtuvo el siguiente IC: ]117.9459 129.9276[.

# Pruebas de intervalos de confianza para dos poblaciones

## **Una diferencia de promedios usando distribución Z para dos poblaciones**

En la siguiente prueba se procederá a realizar los respectivos análisis con la distribución normal z, en la cual se tomará un nivel de confianza del 95%. Para dicha prueba se toman dos diferentes poblaciones, cada población tiene un mismo origen y destinos distintos. Ambas poblaciones siguen un comportamiento normal. Para esta prueba se usó la librería ‘BSDA’.

### **Resultados**

#En la siguiente asignación se toman la cantidad de tiempo en el aire de los vuelos que salen de JFK y tienen como destino MIA, y guarda en var1 sólo los vuelos con dichas características.  
prueba1 <- na.omit(dataFlight[dataFlight$origin == "JFK" & dataFlight$dest == "MIA", ]$air\_time)  
  
#Estos son los n cantidad de vuelos que salen de JFK y tienen como destino RDU  
prueba2 <- na.omit(dataFlight[dataFlight$origin == "JFK" & dataFlight$dest == "RDU", ]$air\_time)  
  
#Para la respectiva prueba se ingresan como para metros la primera población, luego la segunda, seguidamente la desviación estándar de cada una, en el mismo orden y por último el nivel de confianza  
BSDA:: z.test(prueba1, prueba2, sigma.x = sd(prueba1), sigma.y = sd(prueba2), conf.level = 0.95 )$conf.int

## [1] 79.82943 80.71615  
## attr(,"conf.level")  
## [1] 0.95

### **Conclusión**

Al realizar los respectivos cálculos de las dos poblaciones con la distribución normal z, con un n1 = cantidad de tiempos de vuelos de JFK a MIA y un n2 = cantidad de tiempos de vuelos de JFK a RDU. Para dicha prueba se obtuvo un intervalo de confianza de: ]79.82943, 80.71615[; con un nivel de confianza del 95%.

En conclusión, el valor del promedio de la primera población *prueba1* es mayor, ya que los valores de los intervalos de confianza son ambos positivos, por ende, no se puede asumir una igualdad de medias, porque el 0 no está presente.

## **Diferencia de promedios utilizando distribución T para dos poblaciones**

En la siguiente prueba se procederá a realizar los respectivos cálculos con la distribución T, en la cual se tomará un nivel de confianza del 95%. Esta distribución debe tener un n < 30, por lo que se tomaron 25 datos aleatorios en ambas poblaciones de la columna de *air\_time* para dicha prueba. Las poblaciones que contienen las dos muestras siguen una distribución normal. Para esta prueba se usó la librería ‘stats’.

### **Resultados**

#En la siguiente asignación se toman los n cantidad de vuelos que salen de JFK y tienen como destino MIA, y guarda en var1 sólo los vuelos con dichas características.  
var1 <- na.omit(dataFlight[dataFlight$origin == "JFK" & dataFlight$dest == "MIA", ]$air\_time)  
  
#Estos son los n cantidad de vuelos que salen de JFK y tienen como destino RDU  
var2 <- na.omit(dataFlight[dataFlight$origin == "JFK" & dataFlight$dest == "RDU", ]$air\_time)  
  
#Se extraen aleatoriamente 25 datos como muestra de la primera poblacion  
prueba1 <- sample(x = var1, size = 25, replace = TRUE)  
  
#Se extraen aleatoriamente 25 datos como muestra de la segunda poblacion  
prueba2 <- sample(x = var2, size = 25, replace = TRUE)  
  
#Para esta función solamente se ingresan las pruebas y el nivel de confianza.  
stats::t.test(prueba1, prueba2, conf.level = 0.95)$conf.int

## [1] 79.51675 91.04325  
## attr(,"conf.level")  
## [1] 0.95

### **Conclusión**

Al realizar los respectivos cálculos de las dos muestras de las poblaciones con la distribución tStudent, con el n1 = 25 y n2 = 25 tiempos de vuelo de los vuelos de origen JFK a destino MIA y de origen JFK a destino RDU, se obtuvo un intervalo de confianza de: ]75.78145, 84.13855[; con un nivel de confianza del 95%.

En conclusión, el valor de la media de la primera muestra es mayor, ya que los resultados de los intervalos de confianza son positivos, por ende, no se asume igualdad, ya que, el cero no está presente.Por ende, se asumen diferencia en los grupos.

## **Diferencia de proporciones para dos poblaciones**

En esta prueba se trabaja con el tiempo en el aire del origen y del destino seleccionados en el siguiente código para ambas poblaciones, donde se seleccionan los viajes menores o iguales a 150 minutos como los ‘éxitos’. Para la respectiva prueba se utiliza un nivel de confianza de 95%. Para la estimación del intervalo de confianza se tomaron dos poblaciones en las que ambas siguen un comportamiento normal. Para esta prueba se usó la librería ‘stats’.

### **Resultados**

#En la siguiente asignación se toman los n cantidad de vuelos que salen de JFK y tienen como destino MIA, y guarda en var1 sólo los vuelos con dichas características.  
prueba1 <- na.omit(dataFlight[dataFlight$origin == "JFK" & dataFlight$dest == "MIA", ]$air\_time)  
  
#Estos son los n cantidad de vuelos que salen de JFK y tienen como destino RDU  
prueba2 <- na.omit(dataFlight[dataFlight$origin == "JFK" & dataFlight$dest == "RDU", ]$air\_time)  
  
  
#El tamaño de p1 y p2 son los éxitos de los vuelos que tienen un tiempo menor o igual a 150  
exitoP1 <- length(which(prueba1 <= 150))  
exitoP2 <- length(which(prueba2 <= 150))  
  
#Usando la función de prop.test se sacan los intervalos de confianza respectivos, primero con los  
#éxitos de la población 1 y su respectivo tamaño, y segundo con la población 2  
stats::prop.test(x = c(exitoP1, exitoP2), n = c(length(prueba1), length(prueba2)), conf.level = 0.95)$conf.int

## [1] -0.5324788 -0.4976582  
## attr(,"conf.level")  
## [1] 0.95

#Usando la función de prop.test se sacan los intervalos de confianza respectivos, primero con los  
#éxitos de la población 2 y su respectivo tamaño, y segundo con la población 1  
stats::prop.test(x = c(exitoP2, exitoP1), n = c(length(prueba2), length(prueba1)), conf.level = 0.95)$conf.int

## [1] 0.4976582 0.5324788  
## attr(,"conf.level")  
## [1] 0.95

### **Conclusión**

Al realizar los respectivos cálculos de las dos poblaciones con la distribución normal, con el p1 = cantidad de tiempos de vuelo, de 1000 vuelos de JFK a MIA y p2 = cantidad de tiempo de vuelo de 1000 vuelos de JFK a RDU, se obtuvo un intervalo de confianza de: ]-0.5324788, -0.4976582[; con un nivel de confianza del 95%.

Intercambiando los éxitos y los p estimadores, se obtiene un intervalo de confianza ]0.4976582, 0.5324788[; con un nivel de confianza del 95%.

En conclusión, se destaca que proporción para p2 es mayor, ya que al aplicar las diferencias se obtienen resultados negativos y al intercambiar p1 estimador por p2 estimador se obtienen resultados positivos, por ende, el valor de p2 estimador es mayor. Y no se puede asumir una igualdad de proporciones, ya que, el 0 no se encuentra en los intervalos.

## **Cociente de variancias de dos poblaciones**

Para la última prueba con dos poblaciones se utiliza un nivel de confianza del 95%.Para realizar los siguientes cálculos se tomaron dos poblaciones en la cuales ambas siguen una distribución normal. Para la primera población se dispuso del tiempo de vuelo de los vuelos con origen JFK y con destino MIA, y para la segunda, lo mismo nada más que el destino es RDU. Para esta prueba se usó la librería ‘stests’.

### **Resultados**

#En la siguiente asignación se toman la cantidad de tiempo de aire de los vuelos que salen de JFK y tienen como destino MIA, y guarda en var1 sólo los vuelos con dichas características.  
var1 <- dataFlight[dataFlight$origin == "JFK" & dataFlight$dest == "MIA", ]  
  
#Esta es la cantidad de tiempo de aire de los vuelos que salen de JFK y tienen como destino RDU  
var2 <- dataFlight[dataFlight$origin == "JFK" & dataFlight$dest == "RDU", ]  
  
#Para la siguiente función se le pasa la columna de air\_time de ambas variables y por último el nivel de confianza.  
stests:: var.test( na.omit(var1$air\_time), na.omit(var2$air\_time), conf.level = 0.95)$conf.int

## [1] 2.862129 3.293065  
## attr(,"conf.level")  
## [1] 0.95

### **Conclusión**

Al realizar los respectivos cálculos de las dos poblaciones con la distribución f, para cociente de varianzas, se obtuvo un intervalo de confianza de: ]2.862129, 3.293065[, con un nivel de confianza del 95%.

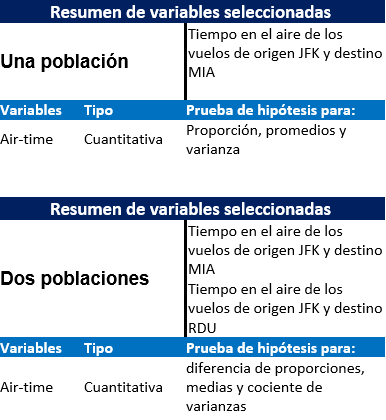
En conclusión, con los resultados obtenidos, la varianza del numerador, que representa a la cantidad de tiempo de aire de los vuelos que tienen como origen JFK y destino MIA es mayor a la varianza del denominador, que es la cantidad de tiempo de aire de los vuelos con el mismo origen, pero con destino en RDU. Por ende, no se puede asumir igualdad de varianzas, ya que el 1 no se encuentra en los intervalos.

### Análisis Inferencial (pruebas de hipótesis de una y dos poblaciones)

* Para la realización de este análisis con las pruebas de hipótesis se utilizará la misma base de datos usada en la primera entrega del proyecto, nycflights13.

### Resumen de variables seleccionadas

#### La siguiente variable fue utilizada para una y dos poblaciones, en donde se utilizaba la duración en aire de un destino al otro.



Variables para una población y dos poblaciones

# Pruebas de hipótesis para una población

## **Distribución normal**

En la siguiente prueba se procederá a realizar los respectivos procedimientos con la distribución normal estándar, en la cuál se tomará un nivel de significancia del 95%. Se tomaron datos correspondientes al tiempo de vuelo de los vuelos con origen en JFK y con destino en MIA, sobre el cuál se quiere estimar que el promedio de vuelo es igual a 120 minutos. Para esto se plantean las siguientes hipótesis: $H\_0: \mu = 120 \\$ - Seguidamente, para validar lo planteado anteriormente, se realizará una prueba de hipótesis para promedio de una población, utilizando un nivel de confianza de 95% que es para elegir si queremos cola izquierda, cola derecha o dos colas; para este caso se usará una prueba de dos colas dado que el tamaño de muestra es mayor que 30.

### **Resultados**

#En la siguiente asignación se toman la n cantidad de vuelos que salen de JFK y tienen como destino MIA, y guarda en var1 sólo los vuelos con dichas características.  
var1 <- na.omit(dataFlight[dataFlight$origin == "JFK" & dataFlight$dest == "MIA", ]$air\_time)  
  
#En esta variable se trabaja con el tiempo en el aire de el origen y destino seleccionado anterionrmente  
  
#Se procede a aplicar la función de TeachingDemos, en la cuál se le pasan como parametros la lista de tiempos, la desviación estandar, el párametro y el tipo de prueba.  
TeachingDemos::z.test(var1, sd(var1),mu = 120,alternative = "two.sided")

##   
## One Sample z-test  
##   
## data: var1  
## z = 165.56, n = 3285.00000, Std. Dev. = 11.12288, Std. Dev. of the  
## sample mean = 0.19407, p-value < 2.2e-16  
## alternative hypothesis: true mean is not equal to 120  
## 95 percent confidence interval:  
## 151.7496 152.5103  
## sample estimates:  
## mean of var1   
## 152.13

### **Conclusión**

## **Distribución T**

En la siguiente prueba se procederá a realizar los respectivos procedimientos con la distribución TStudent, en la cuál se tomará un nivel de significancia del 95%. Se tomó una muestra con un tamaño de 25 correspondientes al tiempo de vuelo de los vuelos con origen en JFK y con destino en MIA, sobre el cuál se quiere estimar que el promedio de vuelo es menor a 120 minutos. Para esto se plantean las siguientes hipótesis: $H\_0: \mu = 120(\ge) \\$ Seguidamente, para validar lo planteado anteriormente, se realizará una prueba de hipótesis para promedio de una población, utilizando un nivel de confianza de 95%, eligiremos si la prueba va a ser de cola izquierda, cola derecha o dos colas; para este caso se usará una prueba de cola izquierda.

### **Resultados**

set.seed(2021)  
#En la siguiente asignación se toman las n cantidad de vuelos que salen de JFK y tienen como destino MIA, y guarda en var1 sólo los vuelos con dichas características.  
  
#En esta variable se trabaja con el tiempo (air-time) en el aire de el origen y destino seleccionados el cual se van a utlizar 25 datos de manera random  
var1 <- sample(x = na.omit(dataFlight[dataFlight$origin == "JFK" & dataFlight$dest == "MIA", ]$air\_time), size = 25, replace = TRUE)  
  
#Se procede a aplicar la función T, en la cual se le pasan como parametros la lista de tiempos y por último el nivel de significancia.  
stats::t.test(var1,mu = 120, alternative = "less" )

##   
## One Sample t-test  
##   
## data: var1  
## t = 15.129, df = 24, p-value = 1  
## alternative hypothesis: true mean is less than 120  
## 95 percent confidence interval:  
## -Inf 154.6839  
## sample estimates:  
## mean of x   
## 151.16

### **Conclusión**

## **Una proporción**

En la siguiente prueba se procederá a realizar los respectivos procedimientos con la distribución normal estándar, en la cuál se tomará un nivel de significancia del 95%. Se tomaron datos correspondientes al tiempo de vuelo de los vuelos con origen en JFK y con destino en MIA, sobre el cuál se quiere estimar que, como máximo el 50% de los vuelos tienen un tiempo de vuelo menor a 150 minutos . Para esto se plantean las siguientes hipótesis: $H\_0: p = 0.5(\le) \\$ Seguidamente, para validar lo planteado anteriormente, se realizará una prueba de hipótesis para una proporción de una población que sigue una distribución normal, utilizando un nivel de confianza de 95%, eligiremos si la prueba va a ser de cola izquierda, cola derecha o dos colas; para este caso se usará una prueba de cola derecha.

### **Resultados**

#En la siguiente asignación se toman las n cantidad de vuelos que salen de JFK y tienen como destino MIA, y guarda en var1 sólo los vuelos con dichas características.  
var1 <- dataFlight[dataFlight$origin == "JFK" & dataFlight$dest == "MIA", ]  
  
#En la siguiente variable se asignan los éxitos obtenidos de la condioción n <= 100  
exitos <- var1[which(na.omit(var1$air\_time) <= 150), ]  
  
#En esta variable se trabaja con el tiempo en el aire de el origen y destino seleccionado anterionrmente  
var1 <- na.omit(var1$air\_time)  
  
  
#Se obtiene la cantidad total de éxitos de la variable anterior éxitos  
totalExitos <- length(exitos$air\_time)  
p = totalExitos/length(var1)  
q= 1-p  
  
n1 = p \* length(var1)  
n2 = q \* length(var1)  
n1

## [1] 1593

n2

## [1] 1692

#El n multiplicado por el p mayor que 5, muestra grande. También con la q  
  
#Para obtener la proporción se procede a ingresar el total de éxitos, el total de todas las variables y por último el nivel de confianza  
stats::prop.test(totalExitos, length(var1),p = 0.5,alternative = "greater", conf.level = 0.95)

##   
## 1-sample proportions test with continuity correction  
##   
## data: totalExitos out of length(var1), null probability 0.5  
## X-squared = 2.9236, df = 1, p-value = 0.9564  
## alternative hypothesis: true p is greater than 0.5  
## 95 percent confidence interval:  
## 0.4704551 1.0000000  
## sample estimates:  
## p   
## 0.4849315

### **Conclusión**

## **Varianza**

En la siguiente prueba se procederá a realizar los respectivos procedimientos con la distribución Chi-Square, en la cuál se tomará un nivel de significancia del 95%. Se tomaron datos correspondientes al tiempo de vuelo de los vuelos con origen en JFK y con destino en MIA, sobre el cuál se quiere contrastar que la varianza del tiempo de aire los vuelos es menor que 0.6. Para esto se plantean las siguientes hipótesis: $H\_0: \sigma^2 = 100(\ge) \\$ Seguidamente, para validar lo planteado anteriormente, se realizará una prueba de hipótesis para una varianza de una población que sigue una distribución normal, utilizando un nivel de confianza de 95%, eligiremos si la prueba va a ser de cola izquierda, cola derecha o dos colas; para este caso se usará una prueba de cola izquierda.

### **Resultados**

#En la siguiente asignación se toman los n cantidad de vuelos que salen de JFK y tienen como destino MIA, y guarda en var1 sólo los vuelos con dichas características.  
var1 <- na.omit(dataFlight[dataFlight$origin == "JFK" & dataFlight$dest == "MIA", ]$air\_time)  
  
#En la siguiente función de stests se procede a igresar la lista de los tiempos y luego el nivel de confianza  
TeachingDemos:: sigma.test(var1, sigmasq = 100,alternative = "less", conf.level = 0.95)

##   
## One sample Chi-squared test for variance  
##   
## data: var1  
## X-squared = 4062.9, df = 3284, p-value = 1  
## alternative hypothesis: true variance is less than 100  
## 95 percent confidence interval:  
## 0.000 128.906  
## sample estimates:  
## var of var1   
## 123.7185

### **Conclusión**

# Pruebas de hipótesis para dos poblaciones

## **Una diferencia de promedios usando distribución normal estándar para dos poblaciones**

En la siguiente prueba se procederá a realizar los respectivos procedimientos con la distribución normal estándar, en la cuál se tomará un nivel de significancia del 95%. Se tomaron datos correspondientes al tiempo de vuelo de los vuelos con origen en JFK y con destino en MIA como el primer promedio, luego del mismo modo, se tomaron datos correspondientes al tiempo de vuelo de los vuelos con origen en JFK y con destino en MIA, como segundo promedio.Se quiere estimar que la diferencia de los promedios de los vuelos es igual a 0 minutos. Para esto se plantean las siguientes hipótesis: $H\_0: \mu\_1 -\mu\_2 = 0 \\$ Seguidamente, para validar lo planteado anteriormente, se realizará una prueba de hipótesis para una diferencia de promedios de dos poblaciones, utilizando un nivel de confianza de 95% que es para elegir si queremos cola izquierda, cola derecha o dos colas; para este caso se usará una prueba de dos colas dado que el tamaño de las muestras es mayor que 30.

### **Resultados**

#En la siguiente asignación se toman los n cantidad de vuelos que salen de JFK y tienen como destino MIA, y guarda en var1 sólo los vuelos con dichas características.  
prueba1 <- na.omit(dataFlight[dataFlight$origin == "JFK" & dataFlight$dest == "MIA", ]$air\_time)  
  
#Estos son los n cantidad de vuelos que salen de JFK y tienen como destino RDU  
prueba2 <- na.omit(dataFlight[dataFlight$origin == "JFK" & dataFlight$dest == "RDU", ]$air\_time)  
  
#Para la respectiva prueba se ingresan como para metros la primera población, luego la segunda, seguidamente la desviación estandar de cada una, en el mismo orden y por último el nivel de confianza  
  
PASWR2::z.test(prueba1, sigma.x = sd(prueba1),prueba2, sigma.y = sd(prueba2),alternative = "two.sided")

##   
## Two Sample z-test  
##   
## data: prueba1 and prueba2  
## z = 354.86, p-value < 2.2e-16  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## 79.82943 80.71615  
## sample estimates:  
## mean of x mean of y   
## 152.12998 71.85719

### **Conclusión**

## **Diferencia de promedios utilizando distribución T para dos poblaciones**

En la siguiente prueba se procederá a realizar los respectivos procedimientos con la distribución TStudent, en la cuál se tomará un nivel de significancia del 95%. Se tomó una muestra con un tamaño de 25 correspondientes al tiempo de vuelo de los vuelos con origen en JFK y con destino en MIA para la primera población, del mismo modo se tomó una muestra con un tamaño de 25 correspondientes al tiempo de vuelo de los vuelos con origen en JFK y con destino en RDU para la segunda población, sobre las cuáles se pretende estimar que la diferencia de los promedios de los vuelo es igual a 0 minutos. Para esto se plantean las siguientes hipótesis: $H\_0: \mu\_1 -\mu\_2 = 0 \\$ Seguidamente, para validar lo planteado anteriormente, se realizará una prueba de hipótesis para diferencia de promedios de dos poblaciones, utilizando un nivel de confianza de 95%, eligiremos si la prueba va a ser de cola izquierda, cola derecha o dos colas; para este caso se usará una prueba de dos colas.

### **Resultados**

#En la siguiente asignación se toman los n cantidad de vuelos que salen de JFK y tienen como destino MIA, y guarda en var1 sólo los vuelos con dichas características.  
set.seed(2021)  
var1 <- na.omit(dataFlight[dataFlight$origin == "JFK" & dataFlight$dest == "MIA", ]$air\_time)  
  
#Estos son los n cantidad de vuelos que salen de JFK y tienen como destino RDU  
var2 <- na.omit(dataFlight[dataFlight$origin == "JFK" & dataFlight$dest == "RDU", ]$air\_time)  
  
#Se extraen aleatoriamente 25 datos como muestra de la primera poblacion  
prueba1 <- sample(x = var1, size = 40, replace = TRUE)  
  
#Se extraen aleatoriamente 25 datos como muestra de la segunda poblacion  
prueba2 <- sample(x = var2, size = 40, replace = TRUE)  
var.test(prueba1, prueba2)

##   
## F test to compare two variances  
##   
## data: prueba1 and prueba2  
## F = 2.9425, num df = 39, denom df = 39, p-value = 0.001066  
## alternative hypothesis: true ratio of variances is not equal to 1  
## 95 percent confidence interval:  
## 1.556291 5.563457  
## sample estimates:  
## ratio of variances   
## 2.942508

#Varianzas diferentes, se trabaja con varianzas diferente. Caso 2 o Caso 3  
  
  
#Para esta función solamente se ingresan las pruebas y el nivel de confianza.  
stats::t.test(prueba1, prueba2, alternative = "two.sided", var.equal = FALSE)

##   
## Welch Two Sample t-test  
##   
## data: prueba1 and prueba2  
## t = 40.938, df = 62.763, p-value < 2.2e-16  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## 77.59276 85.55724  
## sample estimates:  
## mean of x mean of y   
## 152.225 70.650

### **Conclusión**

## **Diferencia de proporciones para dos poblaciones**

En la siguiente prueba se procederá a realizar los respectivos procedimientos con la distribución normal estándar, en la cuál se tomará un nivel de significancia del 95%. Se tomaron datos correspondientes al tiempo de vuelo de los vuelos con origen en JFK y con destino en MIA como primera población, del mismo modo se tomaron datos correspondientes al tiempo de vuelo de los vuelos con origen en JFK y con destino en RDU como segunda población , sobre el cuál se quiere estimar que, la diferencia de la proporción de tiempo de aire de los vuelos de JFK a ORD con la proporción de tiempo de aire de los vuelos de JFK a RDU es igual a 0. Para esto se plantean las siguientes hipótesis: $H\_0: p\_1 -p\_2 = 0 \\$ Seguidamente, para validar lo planteado anteriormente, se realizará una prueba de hipótesis para una diferencia de proporciones de dos poblaciones que se asume que siguen un comportamiento normal, utilizando un nivel de confianza de 95%, eligiremos si la prueba va a ser de cola izquierda, cola derecha o dos colas; para este caso se usará una prueba de dos colas.

### **Resultados**

#En la siguiente asignación se toman los n cantidad de vuelos que salen de JFK y tienen como destino MIA, y guarda en var1 sólo los vuelos con dichas características.  
prueba1 <- na.omit(dataFlight[dataFlight$origin == "JFK" & dataFlight$dest == "MIA", ]$air\_time)  
  
#Estos son los n cantidad de vuelos que salen de JFK y tienen como destino RDU  
prueba2 <- na.omit(dataFlight[dataFlight$origin == "JFK" & dataFlight$dest == "ORD", ]$air\_time)  
  
  
#El length de p1 y p2 son los exitos de los vuelos que tienen un tiempo menor o igual a 150  
exitoP1 <- length(which(prueba1 <= 150))  
exitoP2 <- length(which(prueba2 <= 150))  
n1 <- exitoP1/length(prueba1)  
n2 <- exitoP2/length(prueba2)  
nxp <- n1 \* length(prueba1)  
n2xp <- n2 \* length(prueba2)  
nxp

## [1] 1593

n2xp

## [1] 2216

#Usando la funcion de prop.test se sacan los intervalos de confianza respectivos, primero con los  
#exitos de la población 1 y su respectivo tamaño, y segundo con la poblacion 2  
stats::prop.test(x = c(exitoP1, exitoP2), n = c(length(prueba1), length(prueba2)))

##   
## 2-sample test for equality of proportions with continuity correction  
##   
## data: c(exitoP1, exitoP2) out of c(length(prueba1), length(prueba2))  
## X-squared = 1604.3, df = 1, p-value < 2.2e-16  
## alternative hypothesis: two.sided  
## 95 percent confidence interval:  
## -0.5261450 -0.4905451  
## sample estimates:  
## prop 1 prop 2   
## 0.4849315 0.9932766

#Usando la funcion de prop.test se sacan los intervalos de confianza respectivos, primero con los  
#exitos de la poblacion 2 y su respectivo tamaño, y segundo con la poblacion 1  
#stats::prop.test(x = c(exitoP2, exitoP1), n = c(length(prueba2), length(prueba1)))

### **Conclusión**

## **Cociente de varianzas de dos poblaciones**

En la siguiente prueba se procederá a realizar los respectivos procedimientos con la distribución F, en la cuál se tomará un nivel de significancia del 95%. Se tomaron datos correspondientes al tiempo de vuelo de los vuelos con origen en JFK y con destino en MIA como primer población, del mismo modo se tomaron datos correspondientes al tiempo de vuelo de los vuelos con origen en JFK y con destino en RDU como segunda población, sobre el cuál se quiere contrastar igualdad en las varianzas. Para esto se plantean las siguientes hipótesis: $H\_0: \frac{\sigma^2\_1}{\sigma^2\_2} = 1 \\$ Seguidamente, para validar lo planteado anteriormente, se realizará una prueba de hipótesis para un cociente de varianzas de dos poblaciones que sigue una distribución normal, utilizando un nivel de confianza de 95%, eligiremos si la prueba va a ser de cola izquierda, cola derecha o dos colas; para este caso se usará una prueba de dos colas.

### **Resultados**

#En la siguiente asignación se toman los n cantidad de vuelos que salen de JFK y tienen como destino MIA, y guarda en var1 sólo los vuelos con dichas características.  
var1 <- dataFlight[dataFlight$origin == "JFK" & dataFlight$dest == "MIA", ]  
  
#Estos son los n cantidad de vuelos que salen de JFK y tienen como destino RDU  
var2 <- dataFlight[dataFlight$origin == "JFK" & dataFlight$dest == "RDU", ]  
  
#Para la siguiente función se le pasa la columna de air\_time de ambas variables y por último el nivel de confianza.  
stests:: var.test( na.omit(var1$air\_time), na.omit(var2$air\_time))

##   
## F test to compare two variances  
##   
## data: na.omit(var1$air\_time) and na.omit(var2$air\_time)  
## F = 3.0702, num df = 3284, denom df = 2982, p-value < 2.2e-16  
## alternative hypothesis: true ratio of variances is not equal to 1  
## 95 percent confidence interval:  
## 2.862129 3.293065  
## sample estimates:  
## ratio of variances   
## 3.07023

### **Conclusión**

# **Fase III**

## I PARTE: OTRAS PRUEBAS DE HIPÓTESIS EN R

## **CASO 1:**

feetLength <- split(mosaicData$length, mosaicData$sex)  
feetLength

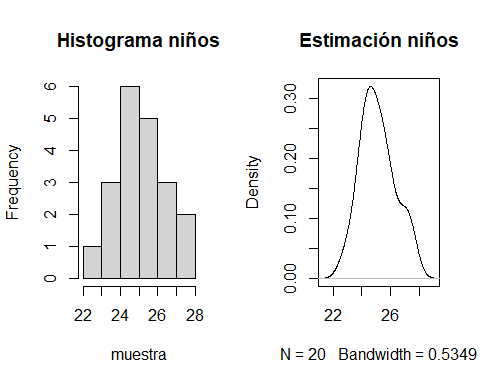
## $B  
## [1] 24.4 25.4 24.5 25.2 25.1 25.7 26.1 22.9 27.5 24.8 26.1 27.0 24.0 24.4 27.1  
## [16] 25.5 24.2 23.9 23.6 24.7  
##   
## $G  
## [1] 23.0 23.6 26.0 23.7 24.0 24.7 26.7 25.5 24.0 24.5 24.2 26.1 24.0 22.5 24.5  
## [16] 22.9 26.0 21.6 24.6

#Esta funcion lo que hace es separar los datos en dos grupos, estos grupos son separados por sexo: niño y niña. Lo que agrega a lo que extrae es es lo que le pasamo en el primer parámetro, en este caso el length.

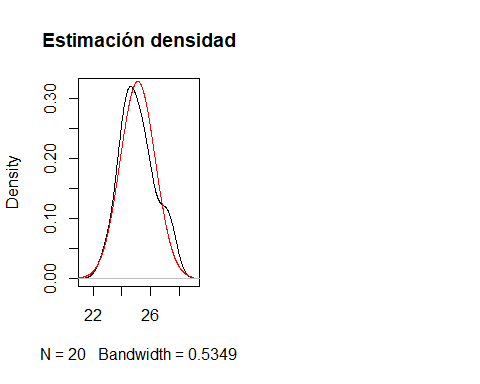
Para cada dataset (niños y niñas), se procederá a realizar un gráfico de la normalidad para cada uno, con el fin de analizar el comportamiento de los datos.

## **Muestra de niños**

# Se toma la muestra de solamente los niños  
muestra <- feetLength$B  
  
par(mfrow = c(1, 2)) # crea una matriz de tamaño 1x2 para imprimir dos gráficos de manera simultánea  
  
hist(muestra, main="Histograma niños")  
plot(density(muestra), main="Estimación niños")



plot(density(muestra), main = "Estimación densidad")  
  
# Se generan valores para estimar la distribución normal de media y desviación estáandar de la lista de feetLength de los niños  
  
#En el from y en el to se agregan un número menos al menor de los datos y un número mayor al de los datos.  
x <- seq(from = 21,to = 29, by = 0.01)  
u <- mean(muestra)  
sigma <- sd(muestra)  
lines(x, dnorm(x, mean = u, sd = sigma), col = "red")

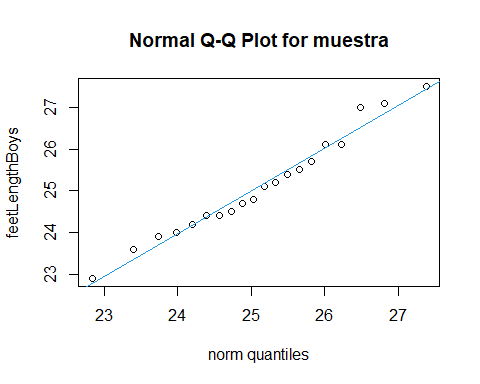


## **¿Se puede intuir una posible normalidad para los datos?**

* En el caso de los niños si se puede intuir una posible normalidad ya que como se puede apreciar en el gráfico anterior de la estimación de densidad es muy parecido el comportamiento de la línea color negro que representa los datos de los niños a la línea roja.

### Gráfico QQ-plot

u <- mean(muestra)  
sigma <- sd(muestra)  
  
EnvStats::qqPlot(muestra, y = NULL, distribution = "norm", param.list = list(mean = u, sd = sigma),  
 add.line = TRUE,   
 points.col = 1, line.col = 100, line.lwd = par("cex"), line.lty = 1,   
 digits = .Options$digits, main = NULL, xlab = "norm quantiles", ylab = "feetLengthBoys")



FALTA EL QQ-PLOT PORQUE MARCO ES UN IRRESPONSABLE Y NO LE HA PREGUNTADO AL PROFE NADA AÚN.

### **Pruebas formales de normalidad S-W test, A-D test,K-S-L test.**

### ***Test de Kolmogorov-Smirnov-Lilliefors o K-S-L test***

lillie.test(muestra)

##   
## Lilliefors (Kolmogorov-Smirnov) normality test  
##   
## data: muestra  
## D = 0.098964, p-value = 0.8746

* En este caso la normalidad de los datos sí se acepta, esto porque el valor de p es mayor a 0.5. Por lo tanto, no hay suficiente evidencia en contra para no aceptar la normalidad de los datos.

### **Test de normalidad de Anderson-Darling o A-D test**

ad.test(muestra)

##   
## Anderson-Darling normality test  
##   
## data: muestra  
## A = 0.24361, p-value = 0.7311

* En este caso la normalidad de los datos sí se acepta, esto porque el valor de p es mayor a 0.5. Por lo tanto, no hay suficiente evidencia en contra para no aceptar la normalidad de los datos.

### **Test de Shapito-Wilks o S-W test**

shapiro.test(muestra)

##   
## Shapiro-Wilk normality test  
##   
## data: muestra  
## W = 0.97195, p-value = 0.7954

* En este caso no se encontró evidencia en contra para no asumir normalidad, esto porque al igual que las pruebas anteriores el valor de p dió mayor a 0.5.

### **Test omnibus de D’Agostino-Pearson**

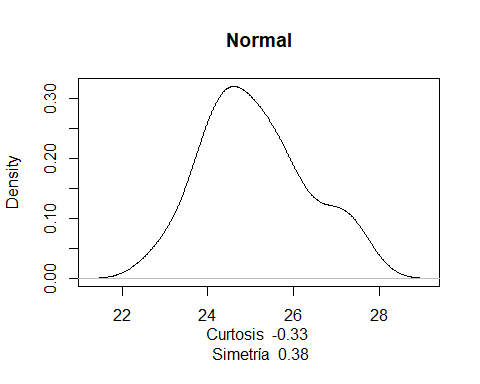
Si es posible realizar esta prueba porque el tamaño de la lista para los niños es de 20, por ende, si se puede realizar la prueba.

dagoTest(muestra)

##   
## Title:  
## D'Agostino Normality Test  
##   
## Test Results:  
## STATISTIC:  
## Chi2 | Omnibus: 0.6384  
## Z3 | Skewness: 0.7831  
## Z4 | Kurtosis: -0.1584  
## P VALUE:  
## Omnibus Test: 0.7267   
## Skewness Test: 0.4335   
## Kurtosis Test: 0.8742   
##   
## Description:  
## Thu Jun 10 22:14:43 2021 by user: warne

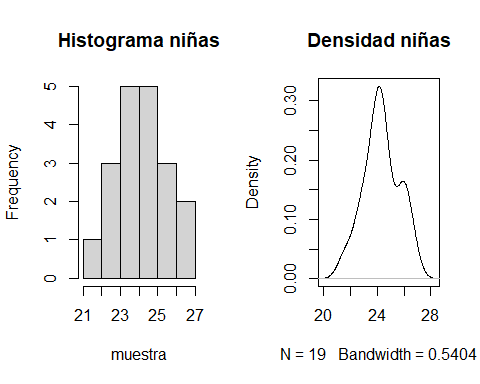
### **Analizando los datos de la curtoris**

curtosis <- kurtosis(muestra)  
 simetria <- skewness(muestra)  
 plot (density(muestra), main = "Normal",  
 xlab = paste("Curtosis ", round(curtosis, 2), "\n",  
 "Simetría ", round(simetria,2)))

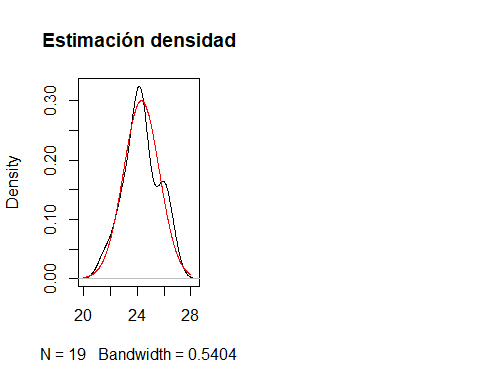
 - Como se puede observar la simetría es de 0.38 por lo cual se acerca al cero, por lo tanto se puede decir que es cimétrica, ya que ambos valores que están a los extremos se encuentran aproximadamente a la misma distancia de la media. En este caso la curtosis muestra una campana muy alta alargada y con las colas bajas por ende se puede decir que es una Leptocúrtica además de que se inclina hacia la izquierda.

## **Muestra de niñas**

# Se toma la muestra de solamente las niñas  
muestra <- feetLength$G  
  
par(mfrow = c(1, 2)) # crea una matriz de tamaño 1x2 para imprimir dos gráficos de manera simultánea  
  
hist(muestra, main="Histograma niñas")  
plot(density(muestra), main="Densidad niñas")



plot(density(muestra), main = "Estimación densidad")  
  
# Se generan valores para estimar la distribución normal de media y desviación estáandar igual a los datos de la lista de feetLength de los niños  
  
#En el from y en el to se agregan un número menos al menor de los datos y un número mayor al de los datos.  
x <- seq(from = 20,to = 28, by = 0.01)  
u <- mean(feetLength$G)  
sigma <- sd(feetLength$G)  
lines(x, dnorm(x, mean = u, sd = sigma), col = "red")



## **¿Se puede intuir una posible normalidad para los datos?**

* En el caso de las niñas si se puede intuir una posible normalidad ya que como se puede apreciar en el gráfico anterior de la estimación de densidad es muy parecido el comportamiento de la línea color negro que representa los datos de las niñas a la línea roja.

# AQUÍ FALTA LA PRUEBA DE QQ-PLOT, PORQUE MARCO ES UN IRRESPONSABLE Y NO LE HA PREGUNTADO AL PROFE

### **Pruebas formales de normalidad S-W test, A-D test,K-S-L test.**

### ***Test de Kolmogorov-Smirnov-Lilliefors o K-S-L test***

lillie.test(muestra)

##   
## Lilliefors (Kolmogorov-Smirnov) normality test  
##   
## data: muestra  
## D = 0.12471, p-value = 0.6094

* En este caso la normalidad de los datos sí se acepta, esto porque el valor de p es mayor a 0.5. Por lo tanto, no hay suficiente evidencia en contra para no aceptar la normalidad de los datos.

### **Test de normalidad de Anderson-Darling o A-D test**

ad.test(muestra)

##   
## Anderson-Darling normality test  
##   
## data: muestra  
## A = 0.26721, p-value = 0.6464

* En este caso la normalidad de los datos sí se acepta, esto porque el valor de p es mayor a 0.5. Por lo tanto, no hay suficiente evidencia en contra para no aceptar la normalidad de los datos.

### **Test de Shapito-Wilks o S-W test**

shapiro.test(muestra)

##   
## Shapiro-Wilk normality test  
##   
## data: muestra  
## W = 0.97303, p-value = 0.835

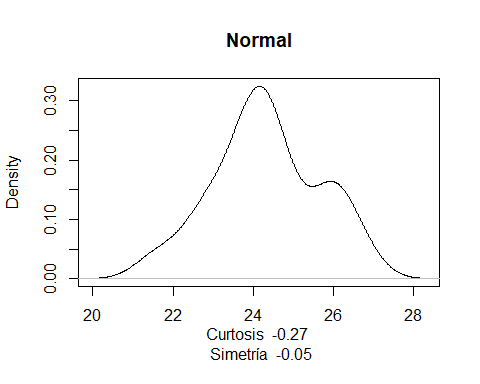
* En este caso no se encontró evidencia en contra para no asumir normalidad, esto porque al igual que las pruebas anteriores el valor de p dió mayor a 0.5.

### **Test omnibus de D’Agostino-Pearson**

En este caso no posible realizar la prueba ya que el tamaño de lista para las niñas es de 19, por ende, no puede aplicarse la prueba.

### **Analizando datos de la curtoris**

curtosis <- kurtosis(muestra)  
 simetria <- skewness(muestra)  
 plot (density(muestra), main = "Normal",  
 xlab = paste("Curtosis ", round(curtosis, 2), "\n",  
 "Simetría ", round(simetria,2)))



Como se puede observar la simetría es de -0.05 por lo cual es menor a cero, por lo tanto, se puede decir que no es simétrica, indicando que el valor mayor está más cerca de la media que el valor menor. En este caso la curtosis muestra una campana muy alta alargada y con las colas bajas por ende se puede decir que es una Leptocúrtica además de que se inclina hacia la izquierda.

## **CASO 2:**

* Se procederá a realizar un gráfico de normalidad con el fin de analizar el comportamiento de los datos de biceps.

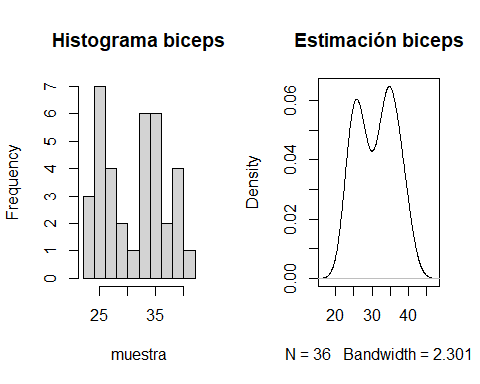
## **Muestra de biceps**

A continuación se utilizarán los datos de biceps de la tabla medidas\_cuerpo.

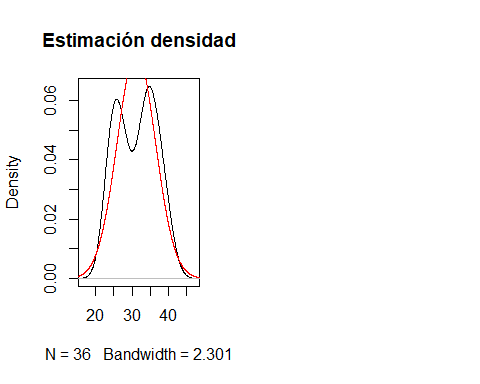
url <- 'https://raw.githubusercontent.com/fhernanb/datos/master/medidas\_cuerpo'   
muestra <- read.table(file=url, header=T)   
muestra <- muestra$biceps  
muestra

## [1] 35.8 35.0 38.5 32.2 32.9 38.5 38.3 35.0 32.1 40.4 36.5 34.2 35.2 34.0 34.8  
## [16] 38.6 36.4 34.0 24.3 27.8 24.0 28.0 26.9 26.5 24.1 29.2 29.0 33.6 24.8 25.4  
## [31] 25.9 24.0 25.6 26.0 23.5 31.0

par(mfrow = c(1, 2)) # crea una matriz de tamaño 1x2 para imprimir dos gráficos de manera simultánea  
  
hist(muestra, main="Histograma biceps")  
plot(density(muestra), main="Estimación biceps")



plot(density(muestra), main = "Estimación densidad")  
  
# Se generan valores para estimar la distribución normal de media y desviación estándar de los biceps  
  
#En el from y en el to se agregan un número menos al menor de los datos y un número mayor al de los datos.  
x <- seq(from = 10,to = 50, by = 0.01)  
u <- mean(muestra)  
sigma <- sd(muestra)  
lines(x, dnorm(x, mean = u, sd = sigma), col = "red")



## **¿Se puede intuir una posible normalidad para los datos?**

* En este caso no se puede intuir ya que a como se puede observar en el gráfico anterior el comportamiento que toma la línea color negro es muy distinta.
* AQUÍ TAMBIÉN FALTA LO DE QQ-PLOT

# AQUÍ FALTA LA PRUEBA DE QQ-PLOT, PORQUE MARCO ES UN IRRESPONSABLE Y NO LE HA PREGUNTADO AL PROFE

### **Pruebas formales de normalidad S-W test, A-D test,K-S-L test.**

### **Test de Kolmogorov-Smirnov-Lilliefors o K-S-L test**

lillie.test(muestra)

##   
## Lilliefors (Kolmogorov-Smirnov) normality test  
##   
## data: muestra  
## D = 0.12583, p-value = 0.1588

* En este caso la normalidad de los datos no se acepta, esto porque el valor de p es menor a 0.5. Por lo tanto, no hay suficiente evidencia para aceptar la normalidad de los datos.

### **Test de normalidad de Anderson-Darling o A-D test**

ad.test(muestra)

##   
## Anderson-Darling normality test  
##   
## data: muestra  
## A = 0.94278, p-value = 0.0151

* En este caso la normalidad de los datos no se acepta, esto porque el valor de p es menor a 0.5. Por lo tanto, no hay suficiente evidencia para aceptar la normalidad de los datos.

### **Test de Shapito-Wilks o S-W test**

shapiro.test(muestra)

##   
## Shapiro-Wilk normality test  
##   
## data: muestra  
## W = 0.92377, p-value = 0.01627

* En este caso se encontró evidencia en contra para no asumir normalidad, esto porque al igual que las pruebas anteriores el valor de p dió meno a 0.5.

### **Test omnibus de D’Agostino-Pearson**

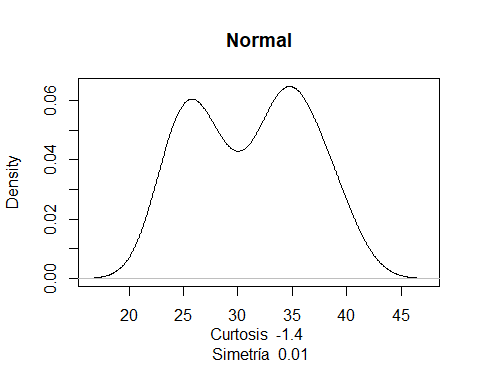
Si es posible realizar esta prueba porque el tamaño de la lista para los biceps es de 36, por ende, si se puede realizar la prueba.

dagoTest(muestra)

##   
## Title:  
## D'Agostino Normality Test  
##   
## Test Results:  
## STATISTIC:  
## Chi2 | Omnibus: 13.6459  
## Z3 | Skewness: 0.0162  
## Z4 | Kurtosis: -3.694  
## P VALUE:  
## Omnibus Test: 0.001088   
## Skewness Test: 0.9871   
## Kurtosis Test: 0.0002208   
##   
## Description:  
## Thu Jun 10 22:14:44 2021 by user: warne

### **Analizando datos de la curtoris**

curtosis <- kurtosis(muestra)  
 simetria <- skewness(muestra)  
 plot (density(muestra), main = "Normal",  
 xlab = paste("Curtosis ", round(curtosis, 2), "\n",  
 "Simetría ", round(simetria,2)))



observar la simetría es de 0.01 por lo cual se acerca mucho al cero, por lo tanto se puede decir que es simétrica, ya que ambos valores que están a los extremos se encuentran aproximadamente a la misma distancia de la media. La curtosis se puede denotar un poco inclinada hacia la derecha y por lo que no es muy alta y su punta semi redonda se puede decir que es Mesocúrtica.