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:

17/4/2021

### 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(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:datasets':  
##   
## Orange

library(visdat)

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

library(DataExplorer)

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

data("flights")  
  
dataFlight <- flights  
  
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>

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

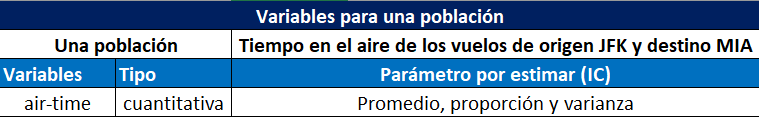
### Variables de la base datos

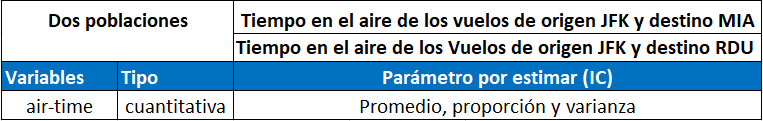
dataFlight

## # 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>

### 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.

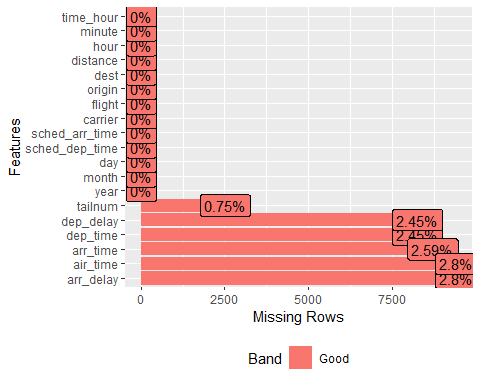
 -



Variables para una y dos poblaciones

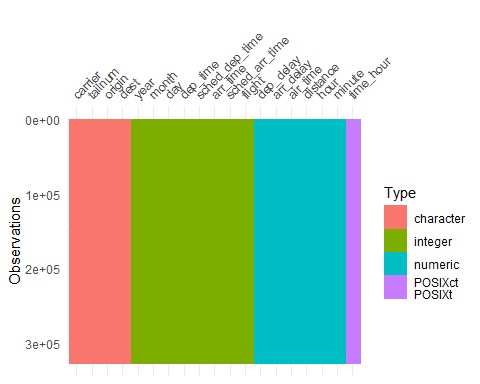
# Gráficos

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, que no presenta datos en blanco, lo cual es 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.

# Pruebas de intervalos de confianza para una población

## **Distribución normal Z**

En la siguiente prueba se procederá a realizar las respectivas 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 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 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: ]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 tomaron los primeros 25 datos de 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 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, conf.level = 0.95)$conf.int

## [1] 148.6905 155.5495  
## attr(,"conf.level")  
## [1] 0.95

### **Conclusión**

Al realizar los respectivos cálculos de la distribución TStudent con un muestra de 25 datos de tiempos de vuelo, con origen JFK a destino MIA se obtuvo un intervalo de confianza de: ]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 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)  
  
  
#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 100 minutos, se obtuvo un intervalo de confianza de: ]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 igresar 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 los respectivos cálculos de la variancia con un intervalo de confianza del 95% para los tiempos de vuelos de origen JFK a destino MIA se obtuvo un intervalo de confianza de: ]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 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  
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 distribucion 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 significancia del 95%.

En conclusión, el valor de la primera población *prueba1* es mayor, ya que los valores de los intervalos de confianza son mayores, 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 muestra 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] 71.42276 85.05724  
## 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 significancia del 95%.

En conclusión, el valor 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.

## **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 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))  
  
#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)), conf.level = 0.95)$conf.int

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

#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)), 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 significancia del 95%.

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

En conclusión, se destaca que proporción para p2 estimador 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 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), 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 significancia del 95%.

En conclusión con los resultados obtenidos, la varianza del numerador es mayor a la varianza del denominador. No se puede asumir igualdad de varianzas, ya que el 1 no se encuentra en los intervalos.