

# Taller 2 Programación en Lenguajes Estadísticos

Jose Angel Urquijo Parra, Ramiro Iseda Payares

2022-08-20

## 1 Implementar un programa con las siguientes opciones:

- Exportar el conjunto de datos gapminder en formato “csv”. El 5 % de los valores de las columnas lifeEx, pop, y gdpPercap se debe reemplazar de forma aleatoria por valores no asignados NA.
- Importar el archivo gapminder en formato “csv”.
- Graficar el diagrama de dispersión lifeEx vs pop.
- Graficar el diagrama de dispersión gdpPercap vs pop.
- Graficar los diagramas de cajas de la variable gdpPercap discriminados por continentes desde 1990 a 2007.

## Librerías

```
library(gapminder)
library(dplyr)
library(ggplot2)
library(readr)
library(openxlsx)
library(gganimate)
library(Rlab)
library(gifski)
```

```

main1<- function(){
  i<-0
  while (i!=6) {
    print("1. Exportar archivo gapminder y reemplazar 5% de lifeExp, pop, gdpPercap por NA.")
    print("2. Importar el archivo gapminder.csv.")
    print("3. Grafica de dispersión lifeExp vs pop.")
    print("4. Grafica de dispersión gdpPercap vs pop.")
    print("5. Diagrama de caja por continente de gdpPercap de los años 1990 a 2007.")
    print("6. Salir")

    i<- as.integer(readline("Escoja una opción: "))

    if(i == 1){
      d=dim(gapminder)
      x=(as.integer(0.05*d[1]))
      indice1=sample(1:d[1],x, replace = F)
      indice2=sample(1:d[1],x, replace = F)
      indice3=sample(1:d[1],x, replace = F)
      gapminder$lifeExp[indice1]=NA
      gapminder$gdpPercap[indice3]=NA
      gapminder$pop[indice2]=NA
      write.table(gapminder, "gapminder.csv")
      cat('\n')
      print("Archivo exportado")
      cat('\n')

    }else if(i == 2){
      df<- read.csv("gapminder.csv")
      cat('\n')
      print("Archivo importado")
      cat('\n')
    }
  }
}

```

```

j<- (readline("?Desea ver el archivo?\n"))
cat('\n')
if (j=="si" | j=="SI"){
  print(df)
}else{
  cat('\n')
  print("Continue")
  cat('\n')
}
}else if (i==3){
  p1<-mean(df$lifeExp,na.rm=TRUE)
  df$lifeExp[indice1]=p1
  p2<-mean(df$pop,na.rm=TRUE)
  df$pop[indice2]=p2
  p3<-mean(df$gdpPercap,na.rm=TRUE)
  df$gdpPercap[indice3]=p3
  cat('\n')
  print(ggplot(df, aes(lifeExp, pop, col=continent))+geom_point()+
        labs(x="LifeExpo", y="Población",
             title="Dispersión de LifeExp vs Población"))
  cat('\n')
}else if (i==4){
  cat('\n')
  print(ggplot(df, aes(log(gdpPercap), pop, col=continent))+geom_point()+
        labs(x="GdpPercap", y="Población",
             title="Dispersión de GdpPercap vs Población"))
  cat('\n')
}else if (i==5){
  df1<-df %>% select(continent, year, gdpPercap) %>% filter(year <= 2007) %>%
  filter(year >= 1990)
  boxplot(df1$gdpPercap~df1$continent,

```

```

        xlab="Continentes",ylab="GdpPercap",
        main="Diagrama de caja de GdpPercap,
        por continente entre los años 1990 a 2007")
    }else if (i==6){
        cat('\n')
        print("gracias,")
        cat('\n')
    }
}
}

```

## 2 Implementar un programa con las siguientes opciones:

- Cargar dos archivos de datos en formato “csv” llamados “Experimento\_a.csv” y “Experimento\_b.csv” e indicar si la diferencia en la media de los datos es estadísticamente significativa.
- Cargar dos archivos de datos en formato “csv” llamados “Experimento\_a.csv” y “Experimento\_b.csv” y mostrar en pantalla la correlación de person y Sperman de los datos.
- Cargar dos archivos de datos en formato “csv” llamados “Experimento\_a.csv” y “Experimento\_b.csv” y graficar el diagrama de dispersión y la linea recta que aproxime los datos calculada por una regresión lineal por minimos cuadrados.

```

est<-function(){
    print("Bienvenido")
    cat('\n')
    cat('\n')
    print("Dos experimentos con valores de una distribución normal, ingrese:")
    j<-as.integer(readline("ingrese la longitud de los datos para ambos experimentos: \n"))
    x<-as.integer(readline("ingrese la media para el experimento_a: \n"))
    x1<-as.integer(readline("ingrese la media para el experimento_b: \n"))
    d<-as.integer(readline("ingrese la desviación estandar para el experimento_a: \n"))
}

```

```

d1<-as.integer(readline("ingrese la desviación estandar para el experimento_b: \n"))
vec1=rnorm(j,x,d)
vec2=rnorm(j,x1,d1)
write.csv(vec1, "Experimento_a.csv")
write.csv(vec2, "Experimento_b.csv")
vec1a<-read.csv("Experimento_a.csv")$x
vec2a<-read.csv("Experimento_b.csv")$x
layout(matrix(c(1:2),nrow=1, byrow=FALSE))
layout.show(2)
hist(vec1a, main = "Histograma del experimento A",
      xlab = "Experimento_a")
hist(vec2a, main = "Histograma del experimento B",
      xlab = "Experimento_b")
i=0
while (i!=4){
  print("1. Son las medias estadísticamente significativas")
  print("2. Correlación de Pearson y Spearman para los experimentos.")
  print("3. Diagrama de dispersión con linea de tendencia.")
  print("4. Salir")
  i<-as.integer(readline("Ingrese una opción:"))
  if (i==1){
    h<-t.test(vec1a,vec2a)
    h1<-h$p.value
    cat(sprintf("El p-valor de los experimentos es: \n %s", h1))
    if(h1>0.05){
      cat('\n')
      print("Las medias no presentan diferencias estadísticamente significativas")
      cat('\n')
    }else{
      cat('\n')
      print("Las medias presentan diferencias estadísticamente significativas")
    }
  }
}

```

```

        cat('\n')
    }
}else if(i==2){
    cat('\n')
    print("Correlación de Pearson es:")
    cat('\n')
    print(cor(x=vec1a,y=vec2a))
    cat('\n')
    cat('\n')
    print("Correlación de Spearman es:")
    cat('\n')
    print(cor(x=vec1a,y=vec2a, method = "spearman"))
    cat('\n')
}else if(i==3){
    vecT<- data.frame(vec1a, vec2a)
    print(ggplot(vecT, aes(x=vec1a,y=vec2a))+geom_point()+
          geom_smooth(method="lm", colour="Red")+
          labs(x="Experimento_a", y="Experimento_b",
               title="Diagrama de dispersión con línea de tendencia" ))

}else if (i==4){
    cat('\n')
    print("Muchas gracias, adios.")
    cat('\n')
}
}
}

```

### 3 Implementar un programa con las siguientes opciones:

- Graficar las funciones de densidad y distribución de una distribución uniforme.
- Graficar la función de densidad y distribución de una distribución de Bernoulli.
- Graficar la función de densidad y distribución de una distribución Poisson.
- Graficar la función de densidad y distribución de una distribución Exponencial.

```
main <- function(){
  print("1. Ver la función de densidad de una distribución uniforme.")
  print("2. Ver la función de densidad de una distribución Bernoulli.")
  print("3. Ver la función de densidad de una distribución Poisson.")
  print("4. Ver la función de densidad de una distribución Exponencial.")
  val <- as.integer(readline("Ingresar la opción que desea ver: "))
  x <- rnorm(100)
  if (val==1){
    curve(dunif(x),xlim=c(130,210),col="blue",lwd=2,
          xlab="x",ylab="f(x)",
          main="Función de Densidad de una distribución uniforme")
  } else { if (val ==2){
    x <- seq(0, 10, by = 1)
    y <- dbern(x, prob = 0.7)
    plot(y, type = "o",col="blue",lwd=2,
          xlab="x",ylab="f(x)",
          main="Función de masa de una distribución Bernoulli")

  } else { if (val ==3){
    x <- 0:50
    lambda <- 5
    plot(dpois(x, lambda), type = "h", lwd = 2,
          main = "Función de masa de probabilidad de una distribución Poisson",
          ylab = "P(X = x)", xlab = "Número de eventos")
  }
  }
```

```

} else { if (val ==4){
  curve(dexp(x),xlim=c(130,210),col="blue",lwd=2,
        xlab="x",ylab="f(x)",
        main="Función de Densidad de una distribución Exponencial")
}
else {
  print("Ninguna de las anteriores")
}

}
}
}
}

```