

- 
1. Se compararon tres dietas respecto al control de azúcar en la sangre en pacientes diabéticos. En el archivo `estad descriptiva.txt` se encuentran los valores de glucosa para las tres dietas consideradas (A, B, C), que contienen las lecturas de glucosa en la sangre de los pacientes. Es deseable que el paciente tenga valores entre 80 — 110 mg/dl.
    - (a) Cargue los datos al R.
    - (b) Para cada una de las tres dietas calcule medidas de centralidad: la media, la mediana, la media  $\alpha$ -podada para  $\alpha = 0.1, 0.2$ . Para cada dieta compare los valores obtenidos de las cuatro medidas de posición, si observa una notable diferencia ¿a que podría deberse?
    - (c) Calcule medidas de dispersión: el desvío estándar, la distancia intercuartil (o intercuartos) en cada una de las dietas. Compare los valores de dispersión obtenidos, si observa una notable diferencia ¿a que podría deberse? ¿Cuál de las dietas parece ser la más estable?
    - (d) Obtenga los percentiles 10, 25, 50, 75 y 90. Compare los valores de los percentiles obtenidos entre las distintas dietas.
    - (e) Construya histogramas que permitan visualizar los valores de glucosa para cada dieta. Compare la distribución de glucosa. ¿Alguna de ellas parece bimodal? ¿En alguna de ellas parece haber valores alejados? ¿Las dietas mantienen a los pacientes en los valores deseados? ¿La distribución de glucosa es asimétrica en alguno de los grupos? ¿En algún caso el ajuste normal parece razonable?
    - (f) Grafique los box-plots correspondientes. ¿Cómo se compara la información que dan estos gráficos con la obtenida con los histogramas? En base a los gráficos obtenidos, discuta simetría, presencia de outliers y compare dispersiones nuevamente.
    - (g) Grafique los qqplots correspondientes. ¿En algún caso el ajuste normal parece razonable?
    - (h) ¿En base al análisis anterior, cuál le parece la dieta más aconsejable?

## Instrucciones para resolver el ejercicio de estadística descriptiva de la clase práctica 16 (Prueba (C))

```
# para leer datos
gluco<-read.table("C:\\Documents and Settings\\PC\\Escritorio\\datos\\estad
                descriptiva.txt", header = TRUE)

#o bien
gluco2<-read.table("C:/Documents and Settings/PC/Escritorio/datos/estad
                descriptiva.txt", header = TRUE)

#de este modo el objeto gluco es un dataframe.
#si el archivo que tiene los datos fuera una lista de numeros sin titulos ni
#nada se puede usar la funcion scan() de dos maneras:
  #primera posibilidad
  graduados<-scan()
#luego, tipear los datos (o usar copy-paste)
#segunda posibilidad
graduados<-scan("C:\\Documents and
                Settings\\PC\\Escritorio\\datos\\graduados.txt")
attach(gluco) #esta instruccion le permite a R trabajar con dietaa, dietab,

#chequear que los esta tomando bien
dietaa
dietab
dietac

#medidas de resumen:
summary(gluco)
#media muestral
gluco<-as.matrix(gluco)
mean(gluco)
mean(dietaa)
#una de las ventajas del R es que trabaja con los objetos: mean(vector) devuelve
#un numero, mean(dataframe) devuelve un vector.
#mediana muestral
median(dietaa)
sort(dietaa) #ordena de menor a mayor al vector
dietaa[1:12]
sort(dietaa)[1:12]
#media alfa-podada
mean(dietaa,trim=0.1) #poda el 10% de la muestra (ordenada) a cada lado, y
#luego calcula la media de lo que resta. Deberia coincidir con
a<-sort(dietaa)
mean(a[11:90])
#varianza muestral y desvio estandar muestral (divide por n-1)
var(dietaa)
cov(dietaa,dietab)
var(gluco) #en la diagonal aparecen las varianzas, en el lugar ij la
covarianza entre la columna i y la columna j de gluco
sqrt(var(dietaa))
sqrt(var(gluco)) #notar la facilidad de R para aplicar funciones numricas
#coordenada a coordenada
#el coeficiente de correlacion muestral
cor(dietaa,dietab)
#que deberia coincidir con
cov(dietaa,dietab)/sqrt(var(dietaa)*var(dietab))
#distancia intercuartil o intercuartiles (ver mas adelante)
#cuantiles o percentiles muestrales
quantile(dietaa) #da por default los cuantiles 0% 25% 50% 75% 100%
```

```

quantile(dietaa,0.6)
quantile(dietaa,0.9) #da igual a
0.9*sort(dietaa)[90]+ 0.1*sort(dietaa)[91]
#pues 0.9*101 = 90.9
quantile(dietaa, probs=c(1,2,5,10,50)/100)
#rango muestral
range(dietaa)
range(dietab)
range(dietac)
#distancia intercuartil o intercuartos, ver ms adelante
#histograma
#los alineamos para compararlos mejor
par(mfrow=c(3,1))
hist(dietaa)
hist(dietab)
hist(dietac)
#para obtener la tabla de frecuencias relativas
hist(dietaa,plot = FALSE)
#o bien, asignarla a alguna lista:
ha<-hist(dietaa)
ha$counts #guarda la cantidad total de observaciones en cada intervalo
#variantes: imponiendo los puntos de cortes
hist(dietaa,breaks=c(65,75,85,95,105,125))
#variantes: imponiendo la cantidad de puntos de cortes
hist(dietaa,breaks=12)
hist(dietaa,breaks=20)
#variantes: no da las frecuencias relativas sino rea total 1 para que distintos
#histogramas (basados en muestras de distinto nmero de observaciones) sean
comparables
hist(dietaa,freq=FALSE)
#boxplot
boxplot(dietaa)
ba<-boxplot(dietaa)
#ba guarda una lista con ba$stats los valores necesarios para hacer el dibujo:
#mediana, cuartos, puntos hasta donde se extienden los "bigotes"
ba$stats
median(dietaa)
#cuarto inferior
median(sort(dietaa)[1:50])
#cuarto superior
median(sort(dietaa)[51:100])
#distancia intercuartos (que es una medida robusta de dispersin)
di<-quantile(dietaa,.75)-quantile(dietaa,.25)
#qqplot
qqnorm(dietaa)
qqnorm(dietab)
qqnorm(dietaa)

```

## 2. Ejercicio 2 de los adicionales de la práctica 5

```
#a) primero generamos dos v.a. uniformes o un vector uniforme
#bivariado, con ambas coordenadas independientes
aa<-runif(2)
xbarra<-mean(aa)
#ahora repetimos 1000 veces
#con un for:
xbarra<-rep(0,1000)
for (i in 1:1000)
{
  aa<-runif(2)
  xbarra[i]<-mean(aa)
}
# generamos lo mismo pero vectorialmente
a1<-runif(2000)
#convertimos al vector en matriz
matriz2<-matrix(a1,nrow=2, ncol=1000)
#aplicamos el promedio por columnas
xbarra2<-apply(matriz2,2,mean)
hist(xbarra2)

#tem b)

#b) idem a) pero con 5
matriz5<-matrix(runif(5000),nrow=5, ncol=1000)
#chequeamos
dim(matriz5)
xbarra5<-apply(matriz5,2,mean)
# Para comparar ambos histogramas en un mismo gráfico,
#dividimos la pantalla en 6:
par(mfrow=c(2,3))
hist(xbarra2,probability=T)
hist(xbarra5,probability=T)

#c) idem 30
matriz30<-matrix(runif(30000),nrow=30, ncol=1000)
xbarra30<-apply(matriz30,2,mean)
hist(xbarra30,probability=T)

#d) idem 200
matriz200<-matrix(runif(200000),nrow=200, ncol=1000)
xbarra200<-apply(matriz200,2,mean)
hist(xbarra200,probability=T)

# idem 500
matriz500<-matrix(runif(500000),nrow=500, ncol=1000)
xbarra500<-apply(matriz500,2,mean)
hist(xbarra500,probability=T)
# idem 1000
matriz1000<-matrix(runif(1000000),nrow=1000, ncol=1000)
xbarra1000<-apply(matriz1000,2,mean)
hist(xbarra1000,probability=T)

#Mejor an, pongamos la misma escala en x para comparar
hist(xbarra2,probability=T,xlim=c(0,1))
```

```
hist(xbarra5,probability=T,xlim=c(0,1))
hist(xbarra30,probability=T,xlim=c(0,1))
hist(xbarra200,probability=T,xlim=c(0,1))
hist(xbarra500,probability=T,xlim=c(0,1))
hist(xbarra1000,probability=T,xlim=c(0,1))

#notemos que queda distorsionado si no obligamos a que la
#escala en el eje y sea la misma
hist(xbarra2,probability=T,xlim=c(0,1),ylim=c(0,45))
hist(xbarra5,probability=T,xlim=c(0,1) ,ylim=c(0,45))
hist(xbarra30,probability=T,xlim=c(0,1) ,ylim=c(0,45))
hist(xbarra200,probability=T,xlim=c(0,1) ,ylim=c(0,45))
hist(xbarra500,probability=T,xlim=c(0,1) ,ylim=c(0,45))
hist(xbarra1000,probability=T,xlim=c(0,1) ,ylim=c(0,45))

#Para generar exponenciales:
# rexp(n, rate = lambda)
#Para generar binomiales
#rbinom(n, size, prob)
#Observar que debemos tomar size = 1, y n =2, 5, 30, 200, 500,
#1000.
```