Estadística Actuarial I (CA-303)
Laboratorio #1

Prof. Maikol Solís

Vamos a analizar una tabla de datos compuesta por las siguientes variables:

- MaxO3: Concentración Máxima de ozono observada en el día en μgr/m^3
- T9, T12, T15: Temperaturas observadas a 9, 12 y 15h
- Ne9, Ne12, Ne15: Nebulosidad observada a 9, 12 et 15h
- Wx9, Wx12, Wx15: Componente Este-Oeste del viento a 9, 12 et 15h
- MaxO3y: Valor máximo de ozono el día anterior.
- wind: Orientación del viento a 12h
- rain: Ocurrencia o no de lluvias.

Carga de datos y análisis preliminar

Primero cargaremos los datos a R.

```
ozone <- read.csv("ozone.txt", sep = ",")
```

Luego veamos algunas características generales de la tabla de datos.

```
ncol(ozone)
colnames(ozone)
sapply(ozone, class)
head(ozone)
```

Calculamos algunos estimadores generales.

```
summary(ozone)
sapply(ozone, mean) # Medias empiricas
sapply(ozone, sd) # Desviaciones estandar empíricas
```

Tablas de contigencia

Vemos que las variables `wind` y `rain` son categóricas, así que podemos hacer una tabla de contingencia para ver la distribución de estos datos.

```
# Tabla de contigencias con las variables categoricas
tc <- table(ozone[,13:14])
tc
margin.table(tc, margin = 1)
margin.table(tc, margin = 2)

tcp <- prop.table(tc)
tcp
margin.table(tcp, margin = 1)
margin.table(tcp, margin = 2)
```

Gráficos de dispersión, boxplots e histogramas

Para las variables numéricas podemos dibujar algunos gráficos de dispersión

```
plot(ozone[2:12])
plot(ozone[2:12],col=ozone$wind)
plot(ozone[2:12],col=ozone$rain)
```

También es posible visualizar algunos *boxplots* por grupos de variables.

```
boxplot(ozone[,c(2,12)]) # Max03, Max03y
boxplot(ozone[,3:5]) # T9, T12, T15
boxplot(ozone[,6:8]) # Ne9, Ne12, Ne15
boxplot(ozone[,9:11]) # Wx9, Wx12, Wx15
```

Para complementar esta información podemos dibujar los histogramas de estas variables. Observe el uso del comando `par(mfrow = c (# Líneas, # Columnas))` para dibujar múltiples histogramas en un solo gráfico.

```
par(mfrow = c(1,2)) # Max03, Max03y
hist(ozone$max03)
hist(ozone$max03y)

par(mfrow = c(1,3)) # T9, T12, T15
hist(ozone$T9)
hist(ozone$T12)
hist(ozone$T15)

par(mfrow = c(1,3)) # Ne9, Ne12, Ne15
hist(ozone$Ne9)
hist(ozone$Ne12)
hist(ozone$Ne15)

par(mfrow = c(1,3)) # Wx9, Wx12, Wx15
hist(ozone$Wx9)
hist(ozone$Wx12)
hist(ozone$Wx15)

par(mfrow = c(1,1)) # Obligatorio para restablecer el sistema de gráficos
```

Compare los histogramas de `max03` y `Wx15` con los siguientes diagramas de tallo-hoja.

```
stem(ozone$max03)
stem(ozone$Wx15)
```

Comparemos la media, la mediana y la moda de una distribución

```
hist(ozone$max03, freq = FALSE, breaks = 20)
abline(v = mean(ozone$max03), col = "red", lwd = 3)
abline(v = median(ozone$max03), col = "blue", lwd = 3)
abline(v = which.max(ozone$max03), col = "green4", lwd = 3)
text(mean(ozone$max03),0.025,labels = expression(hat(mu)),pos = 4)
text(median(ozone$max03),0.025,labels = "Q2",pos = 4)
text(which.max(ozone$max03),0.025,labels = "Moda", pos = 2)
```

Veamos que pasa si cambiamos el ancho de banda `h` de un histograma

```
hist(ozone$T9, freq = FALSE, breaks = 2)
hist(ozone$T9, freq = FALSE, breaks = 60)
hist(ozone$T9, freq = FALSE)
```

QQ-plots

Ahora comparemos la variable `maxO3` con la distribución normal.

```
qqnorm(ozone$maxO3) ; qqline(ozone$maxO3,col="red")
```

Observe que esta variable es asimétrica hacia la izquierda (de tipo lognormal). Para "normalizarla" podemos aplicar una transformación logarítmica.

```
par(mfrow = c(2,2))
hist(ozone$maxO3)
hist(log(ozone$maxO3))
qqnorm(ozone$maxO3) ; qqline(ozone$maxO3,col="red")
qqnorm(log(ozone$maxO3)) ; qqline(log(ozone$maxO3),col="red")
par(mfrow = c(1,1))
```

Análisis en componentes principales

Es posible calcular la matriz de varianza-covarianza y la de correlaciones de estos datos, para observar las relaciones entre las variables.

```
cov(ozone[2:12])
cor(ozone[2:12])
```

Haremos un análisis en componentes principales de nuestros datos.

Primero apliquemos la función `prcomp` con las variables sin modificar.

```
acp.raw <- prcomp(ozone[2:12], center = FALSE, scale. = FALSE)
plot(acp.raw$sdev/sum(acp.raw$sdev), type = "l")
plot(cumsum(acp.raw$sdev)/sum(acp.raw$sdev), type = "l")
```

Lo más aconsejable es centrar y escalar las variables con la media y la desviación estándar de cada una de ellas.

```
acp <- prcomp(ozone[2:12], center = TRUE, scale. = TRUE)
acp
```

Es necesario escoger la cantidad de componentes que vamos a preservar.

```
plot(acp$sdev/sum(acp$sdev), type = "l")
plot(cumsum(acp$sdev)/sum(acp$sdev), type = "l")
```

Podemos generar un nuevo conjunto de variables usando la matriz de proyecciones

```
ozone.reduced <- as.matrix(ozone[2:12]) %*% acp$rotation

plot(ozone.reduced[,1:2])
plot(ozone.reduced[,3:4])
```

Clasificación jerárquica

Calculemos la distancia euclidiana entre cada individuo.

```
d.ind <- dist(ozone[2:12])
```

Luego crearemos el dendrograma con la función `hclust` y lo dibujaremos con la función `plot`.

```
dend.ind <- hclust(d.ind)
plot(dend.ind)
```

Este mismo análisis lo podemos hacer para las variables

```
d.var <- dist( t(ozone[2:12]) ) # Note el uso de la función de transposición t()
dend.var <- hclust(d.var)
plot(dend.var)
```

K-means

Recuerden que el ACP y dendrograma nos sugiere que solo hay 4 variables interesantes en nuestra base de datos. Apliquemos un *K-means* con 4 centroides.

```
km.var <- kmeans(t(ozone[2:12]), centers = 4)
```

Podemos confirmar esta información con la función `heatmap`.

```
heatmap(as.matrix(ozone[2:12]))
heatmap(t(km.var$centers))
```