

GENERAL

```
rm(list = ls())
dat = read.table('coc_hes.txt', header = T)
dat = read.csv('ga_sto_s.csv', header = T)
dat = na.omit(d) # Cojo solo observ. completas
names(d)
dat$sexo = factor (da_t$sexo, labels =c( 'H' , 'M')) # Paso a factor
dat$titu = releve_l(d $titu, ref = 'GITI') # cambiar ref si >2 vars
dat2 = dat[ ,3:8] # c(3,4,..,8) - ordenar datos
dat = dat[ , -1] # Elimino columna 1
row.na.mes (dat2) = dat$VX
round(X, digits=2)
sum(da_t$SEXO == 'mujer')
dim(dat) # Dimensiones base de datos
```

ESTADÍSTICA DESCRIPTIVA

```
hist(dat$V1)
hist(d $V1 [d$V2 == 4])
hist(d [1:50 , 1] )
boxplot(d at$V1) # también en dos variables
summary(d at$V1)
plot( V1 ~ V2, data = dat)
par(mfrow = c(1,2)) # Para dibujar dos gráficos a la vez
```

TÉCNICAS DE REGRESIÓN

```
mod = lm(V1 ~ V2, data = dat) # modelo regresion
mod = lm(V1 ~ V2 + V3, data = dat)
summary(mod) # beta0 y beta1
abline(mod) # Recta de Regresion
pred1 = dat.a.frame(V1=, V2=...)
predict(mod, pred1) # prediccion puntual
predict(mod, pred1, interval = 'prediction', level=0.9)
# prediccion nueva obs
predict(mod, pred1, interval = 'confidence') # promedio
confint(mod) # intervalo de conf - si E0 no sign
mod9 = stepAIC(mod) # Selección de variables singif
res = residuals(mod) # residuos
var(dat) # varianzas covarianzas
cor(dat) # coefs. correlacion (matriz)
pairs(dat) # dispersion multiple
```

LDA - ANÁLISIS DISCRIMINANTE

library(MASS)

```
(m1 = lda(SEXO ~ ., data = dat))
m1$scaling # Coefs de la f.discriminante (no stand)
(a0 = -sum(m1$scaling * colMeans(dat[, -1])))
plot(m1)
(predict = predict(m1))
table(predict = predict$class, real = dat$SEXO)
predictio$class # Decisión del modelo
predictio$posterior # Seguridad decisión
predictio$score #score o puntuación
dat$SEXO[5] # La quinta obs. es mujer
predictio$class[5] # modelo dice
```

library(multis)

```
# .f.discriminante coef. estandarizados
m2 = ldaPlus(x = dat[, -1], grouping = dat$SEXO)
m2$standcoef Within #Coeficientes estandarizados
# Con estos coef. variables son más impor.
# Si hay multicolinealidad: no se pueden interpretar
m2$class.orgTab # M CONFUSIÓN :absoluta
```



LDA - ANÁLISIS DISCRIMINANTE (cont)

```
> m2$class$perTab # M CONFUSIÓN:relativa
m2$classCV$orgTab # MC VC absoluta
m2$classCV$perTab # MC VC relativa
m2$centroids # Centroides
pred1 = predict(m1, flor)
points(pred1$x, cex = 3)
plot(m2, dimen = 1)
```

ANÁLISIS COMPONENTES PRINCIPALES

```
plot(fit, type = 'lines') # grafico codo - num CP
biplot (fit) # añadir rownames A parecido B?
dat$T100 = dat$T100^-1 # si invrs propor cional
fit = princomp( datos, cor = T)
fit$sdev^2 # import ancia
cumsum (m$sdev^2/7) # Import ancias acumuladas
round( fit $sdev^2/6 *100, digits= 2) # imp rel
fit$loadings # pesos (weights)
fit$scores # puntuaciones - valores CP
cor(datos, fit$scores) # cargas
rowSums(cor( datos, fit$scores[, 1:2])^2) # comunalidad
1-rowSums (co r( datos, fit$scores[, 1:2])^2)# especificidad
source ('p rin fac t.R')
sol = prinfact( datos, 6) #num CP
sol$loadings # Cargas
sol$scores # puntuaciones
sol$variances # importancias
sol$eigenvectors[,1] # Pesos [CP1]
CORR PLOT
install.packages('corrplot') # descargar paquete
library(corrplot) # activar paquete
r = cor(dat)
corrplot(r, method = 'ellipse')
```



By **Darktotoro**

cheatography.com/darktotoro/

Not published yet.

Last updated 26th October, 2024.

Page 3 of 4.

Sponsored by **Readable.com**

Measure your website readability!

<https://readable.com>

ANÁLISIS COMPONENTES PRINCIPALES

```

plot(fit, type = 'lines') # grafico codo - num CP
biplot (fit) # añadir rownames A parecido B?
dat$T100 = dat$T100^-1 # si invrs proporcional
fit = princomp( datos, cor = T)
fit$sdev^2 # importancias
cumsum (m$sdev^2/7) # Importancias acumuladas
round( fit $sdev^2/6 *100, digits= 2) # imp rel
fit$loadings # pesos (weights)
fit$scores # puntuaciones - valores CP
cor(datos, fit$scores) # cargas
rowSums(cor( datos, fit$scores[, 1:2])^2) # comunalidad
1-rowSums (cor(datos, fit$scores[, 1:2])^2) # especificidad
source ('princomp.R')
sol = prinfact( datos, 6) #num CP
sol$loadings # Cargas
sol$scores # puntuaciones
sol$variances # importancias
sol$eigenvectors[,1] # Pesos [CP1]
CORRLOT
install.packages('corrplot') # descargar paquete
library(corrplot) # activar paquete
r = cor(dat)
corrplot(r, method = 'ellipse')

```



By **Darktotoro**
cheatography.com/darktotoro/

Not published yet.
 Last updated 26th October, 2024.
 Page 4 of 4.

Sponsored by **Readable.com**
 Measure your website readability!
<https://readable.com>