Análisis Estadístico con R

Técnicas de Análisis Mutivariante

Víctor Morales-Oñate

15 de enero de 2020

Contents

Los contenidos de este material se basa principalemente en Schumacker (2015). Las referencias o extensiones necesarias se citarán conforme se desarrolla el material.

Supuestos de RLM

Librerías usadas en esta sección

```
library(AER)
library(sandwich)
library(lmtest)
library(lmSupport)
```

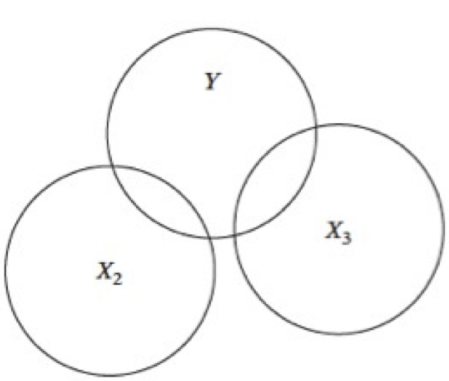
Multicolinealidad

El problema:

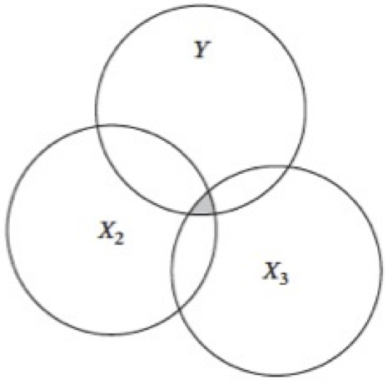
$$\hat{\beta} = (X'X)^{-1}X'Y$$

- Se tiene un problema en cuanto a la transpuesta de la matriz $(X'X)$

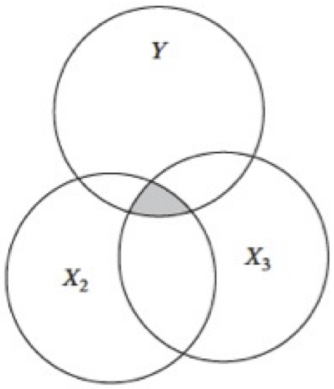
- Perfecta: Si se tiene este tipo, el modelo simplemente no toma en cuenta esta variable
- Imperfecta: El cáclulo de la inversa es computacionalmente exigente



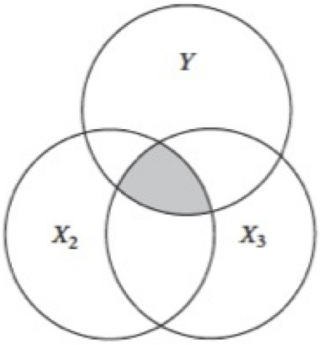
a) No existe colinealidad



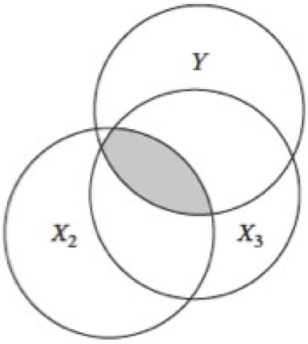
b) Colinealidad baja



c) Colinealidad moderada



d) Colinealidad alta



e) Colinealidad muy alta

Posibles causas

- El método de recolección de información
- Restricciones en el modelo o en la población objeto de muestreo
- Especificación del modelo
- Un modelo sobredeterminado
- Series de tiempo

¿Cuál es la naturaleza de la multicolinealidad?

Causas - ¿Cuáles son sus consecuencias prácticas?

Incidencia en los errores estándar y sensibilidad

- ¿Cómo se detecta?

Pruebas

¿Qué medidas pueden tomarse para aliviar el problema de multicolinealidad?

- No hacer nada
- Eliminar variables
- Transformación de variables
- Añadir datos a la muestra

- Componentes principales, factores, entre otros

¿Cómo se detecta?

- Un R^2 elevado pero con pocas razones t significativas
- Regresiones auxiliares (Pruebas de Klein)
- Factor de inflación de la varianza

$$VIF = \frac{1}{(1 - R^2)}$$

Ejemplo 1

- Haremos uso del paquete AER
- Abrir la tabla 10.8
- Ajusta el modelo

donde

- X_1 índice implícito de deflación de precios para el PIB,
- X_2 es el PIB (en millones de dólares),
- X_3 número de desempleados (en miles),
- X_4 número de personas enlistadas en las fuerzas armadas,
- X_5 población no institucionalizada mayor de 14 años de edad
- X_6 año (igual a 1 para 1947, 2 para 1948 y 16 para 1962).

$$Y_i = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \beta_4 X_4 + \beta_5 X_5 + u_i$$

- Analice los resultados

```
uu <- "https://raw.githubusercontent.com/vmoprojs/DataLectures/master/tabla10_8.csv"
datos<- read.csv(url(uu),sep=";",header=TRUE)
```

Agreguemos el tiempo: - Las correlaciones muy altas también suelen ser síntoma de multicolinealidad

```
ajuste.2 <- lm(Y~X1+X2+X3+X4+X5+TIME,data = datos)
summary(ajuste.2)
```

```
##
## Call:
## lm(formula = Y ~ X1 + X2 + X3 + X4 + X5 + TIME, data = datos)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -381.7 -167.6   13.7  105.5  488.9
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.727e+04  2.324e+04   2.895  0.02005 *
## X1          -2.051e+00  8.710e+00  -0.235  0.81974
## X2          -2.733e-02  3.317e-02  -0.824  0.43385
## X3          -1.952e+00  4.767e-01  -4.095  0.00346 **
## X4          -9.582e-01  2.162e-01  -4.432  0.00219 **
## X5           5.134e-02  2.340e-01   0.219  0.83181
## TIME         1.585e+03  4.827e+02   3.284  0.01112 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 295.6 on 8 degrees of freedom
## Multiple R-squared:  0.9955, Adjusted R-squared:  0.9921
## F-statistic: 295.8 on 6 and 8 DF,  p-value: 6.041e-09
```

```
with(datos,cor(cbind(X1,X2,X3,X4,X5,TIME)))
```

```
##             X1        X2         X3         X4        X5      TIME
## X1   1.0000000 0.9936689  0.5917342  0.4689737 0.9833160 0.9908435
## X2   0.9936689 1.0000000  0.5752804  0.4587780 0.9896976 0.9947890
## X3   0.5917342 0.5752804  1.0000000 -0.2032852 0.6747642 0.6465669
## X4   0.4689737 0.4587780 -0.2032852  1.0000000 0.3712428 0.4222098
## X5   0.9833160 0.9896976  0.6747642  0.3712428 1.0000000 0.9957420
## TIME 0.9908435 0.9947890  0.6465669  0.4222098 0.9957420 1.0000000
```

- Prueba de Klein: Se basa en realizar regresiones auxiliares de *todas contra todas* las variables regresoras.
- Si el R^2 de la regresión aux es mayor que la global, esa variable regresora podría ser la que genera multicolinealidad
- ¿Cuántas regresiones auxiliares se tiene en un modelo en general?

Regresemos una de las variables

```
ajuste.3<- lm(X1~X2+X3+X4+X5+TIME,data = datos)
summary(ajuste.3)
```

```
##
## Call:
## lm(formula = X1 ~ X2 + X3 + X4 + X5 + TIME, data = datos)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -18.8602  -4.3277  -0.3175   4.3726  14.8438
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.529e+03  7.288e+02   2.098   0.0653 .
## X2           2.543e-03  9.453e-04   2.690   0.0248 *
## X3           3.056e-02  1.514e-02   2.019   0.0742 .
## X4           1.011e-02  7.559e-03   1.337   0.2140
## X5          -1.263e-02  7.903e-03  -1.598   0.1445
## TIME        -1.621e+01  1.766e+01  -0.918   0.3826
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.31 on 9 degrees of freedom
## Multiple R-squared:  0.9923, Adjusted R-squared:  0.9881
## F-statistic: 232.5 on 5 and 9 DF,  p-value: 3.127e-09
```

```
tolerancia<-1-0.9923
```

Factor de inflación de la varianza

Si este valor es mucho mayor que 10 y se podría concluir que si hay multicolinealidad

```
vif <- 1/tolerancia
vif
```

```
## [1] 129.8701
```

Ahora vamos a usar el paquete `AER`:

```r
library(AER)
```

```r
vif1 <- vif(ajuste.2)
Raux <- (vif1-1)/vif1
Rglobal <- 0.9955

Rglobal-Raux
```

```
##           X1           X2           X3           X4           X5
##  0.003181137 -0.003829181  0.026533869  0.254649059 -0.001623122
##         TIME
## -0.003160352
```

Se podría no hacer nada ante este problema. O se puede tratar con transformaciones. Deflactamos el PIB: `PIB_REAL <- X2/X1`

```r
# La variable X5 (población)
# esta correlacionada con el tiempo
PIB_REAL <- datos$X2/datos$X1
ajuste.4<-lm(Y~PIB_REAL+X3+X4, data = datos)
summary(ajuste.4)
```

```
##
## Call:
## lm(formula = Y ~ PIB_REAL + X3 + X4, data = datos)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -760.29 -197.71  -53.69  234.77  603.15
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 42716.5646   710.1206  60.154 3.31e-15 ***
## PIB_REAL       72.0074     3.3286  21.633 2.30e-10 ***
## X3             -0.6810     0.1693  -4.023  0.00201 **
## X4             -0.8392     0.2206  -3.805  0.00292 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 389 on 11 degrees of freedom
## Multiple R-squared:  0.9893, Adjusted R-squared:  0.9864
## F-statistic: 339.5 on 3 and 11 DF,  p-value: 4.045e-11
```

```r
vif(ajuste.4)
```

```
## PIB_REAL       X3       X4
## 3.054580 2.346489 2.318500
```

```r
ajuste.5<-lm(Y~PIB_REAL+X3+X4,data = datos)
summary(ajuste.5)
```

```
##
## Call:
## lm(formula = Y ~ PIB_REAL + X3 + X4, data = datos)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -760.29 -197.71  -53.69  234.77  603.15
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 42716.5646   710.1206  60.154 3.31e-15 ***
## PIB_REAL       72.0074     3.3286  21.633 2.30e-10 ***
## X3             -0.6810     0.1693  -4.023  0.00201 **
## X4             -0.8392     0.2206  -3.805  0.00292 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 389 on 11 degrees of freedom
## Multiple R-squared:  0.9893, Adjusted R-squared:  0.9864
## F-statistic: 339.5 on 3 and 11 DF,  p-value: 4.045e-11
```

```
vif(ajuste.5)
```

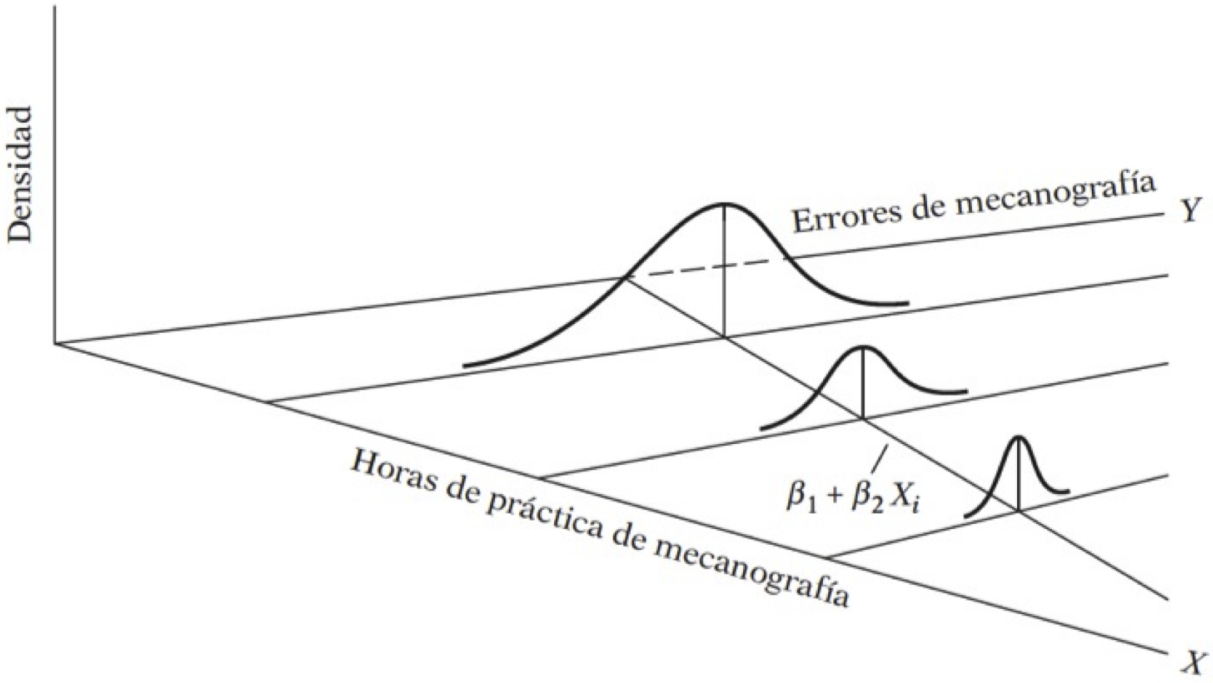
```
## PIB_REAL       X3       X4
## 3.054580 2.346489 2.318500
```

Heterocedasticidad

Ocurre cuando la varianza no es constante.

¿Cuál es la naturaleza de la heterocedasticidad?

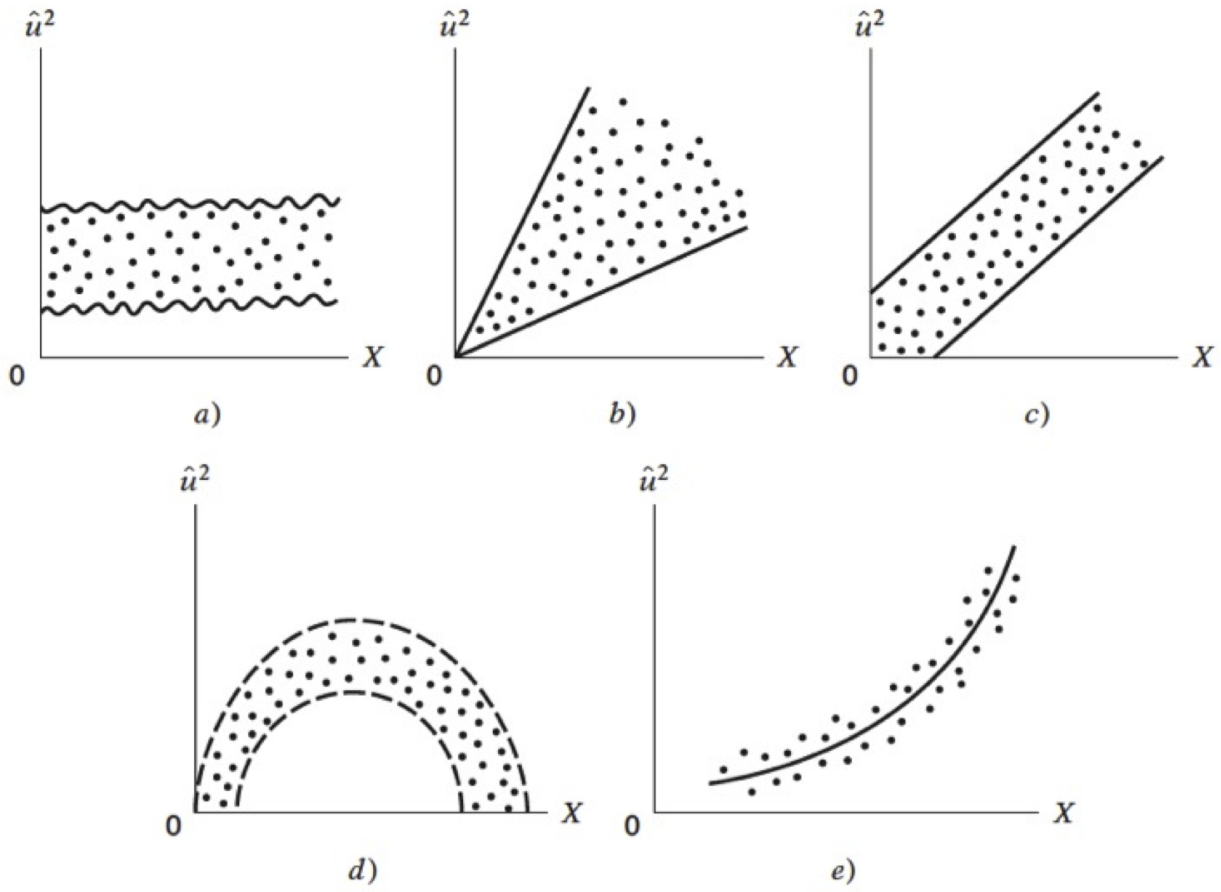
- Modelos de aprendizaje de los errores: con el paso del tiempo, las personas cometen menos errores de comportamiento. Es decir que la varianza disminuye.
- Ingreso direccional: Es probable que la varianza aumente con el ingreso dado que el aumento del ingreso se tiene más opciones del cómo disponer de él.



- Técnicas de recolección de datos: si la técnica mejora, es probable que la varianza se reduzca.
- Datos atípicos o aberrantes: Sensibilidad en las estimaciones
- Especificaciones del modelo: Omisión de variables importantes en el modelo.
- Asimentría: Surge a partir de la distribución de una o más regresoras en el modelo. Ejemplo: Distribución del ingreso *generalmente inequitativo*

¿Cómo detectarla?

Método gráfico



Veamos las pruebas de detección en un ejemplo

- Abrir la base de datos *wage1* de Wooldrigde

```
uu <- "https://raw.githubusercontent.com/vmoprojs/DataLectures/master/wage1.csv"
datos <- read.csv(url(uu),header=FALSE)
names(datos) <- c("wage",  "educ",  "exper",  "tenure",
                  "nonwhite",  "female",   "married",
                  "numdep",    "smsa", "northcen", "south",
                  "west",  "construc", "ndurman",  "trcommpu",
                  "trade", "services", "profserv", "profocc",
                  "clerocc",   "servocc",  "lwage",    "expersq",
                  "tenursq")

casados <- (1-datos$female)*datos$married  # female 1=mujer  married=1 casado
casadas <- (datos$female)*datos$married
solteras <- (datos$female)*(1-datos$married)
solteros <- (1-datos$female)*(1-datos$married)
```

- Correr el modelo

  $lwage = \beta_0+\beta_1 casados+\beta_2 casadas+\beta_3 solteras+\beta_4 educ+\beta_5 exper+\beta_6 expersq+\beta_7 tenure+\beta_8 tenuresq+u_i$

- Hacer un gráfico de los valores estimados y los residuos al cuadrado

Prueba de Breusch Pagan

- Correr un modelo de los residuos al cuadrado regresado en las variables explicativas del modelo global.

  $sqresid = \beta_0 + \beta_1 casados + \beta_2 casadas + \beta_3 solteras + \beta_4 educ + \beta_5 exper + \beta_6 expersq + \beta_7 tenure + \beta_8 tenuresq + u_i$

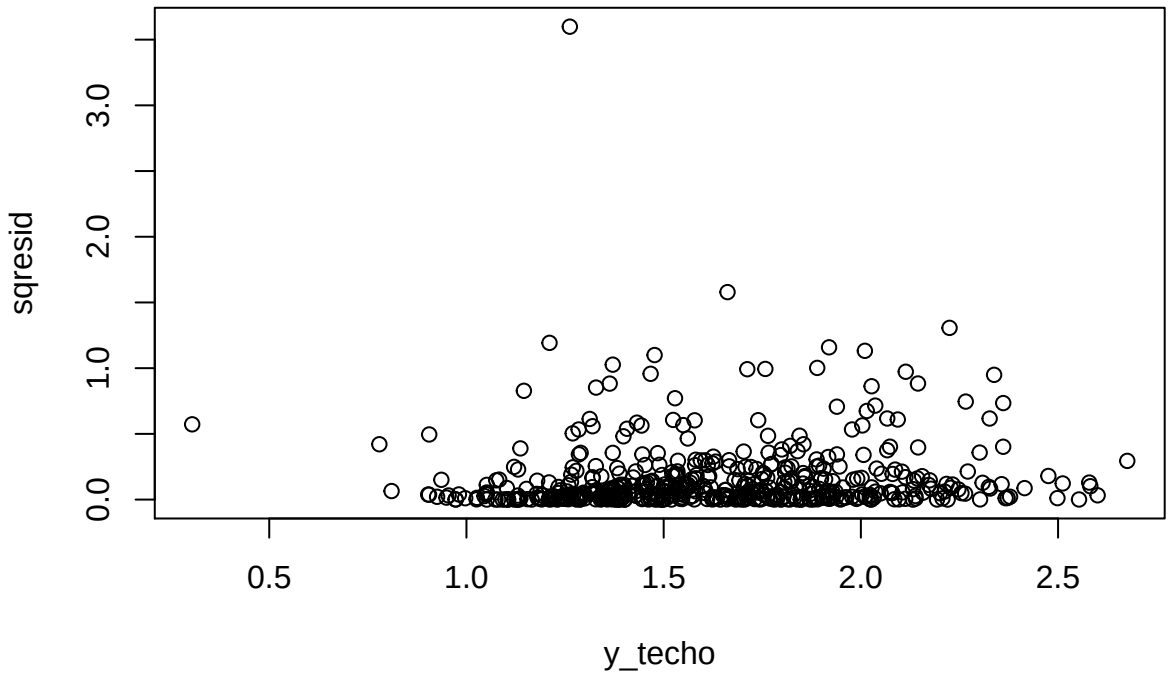
- `bptest(objeto)`: si el pvalor es inferior a 0.05, `Ho: Homocedasticidad`

El código en `R` es:

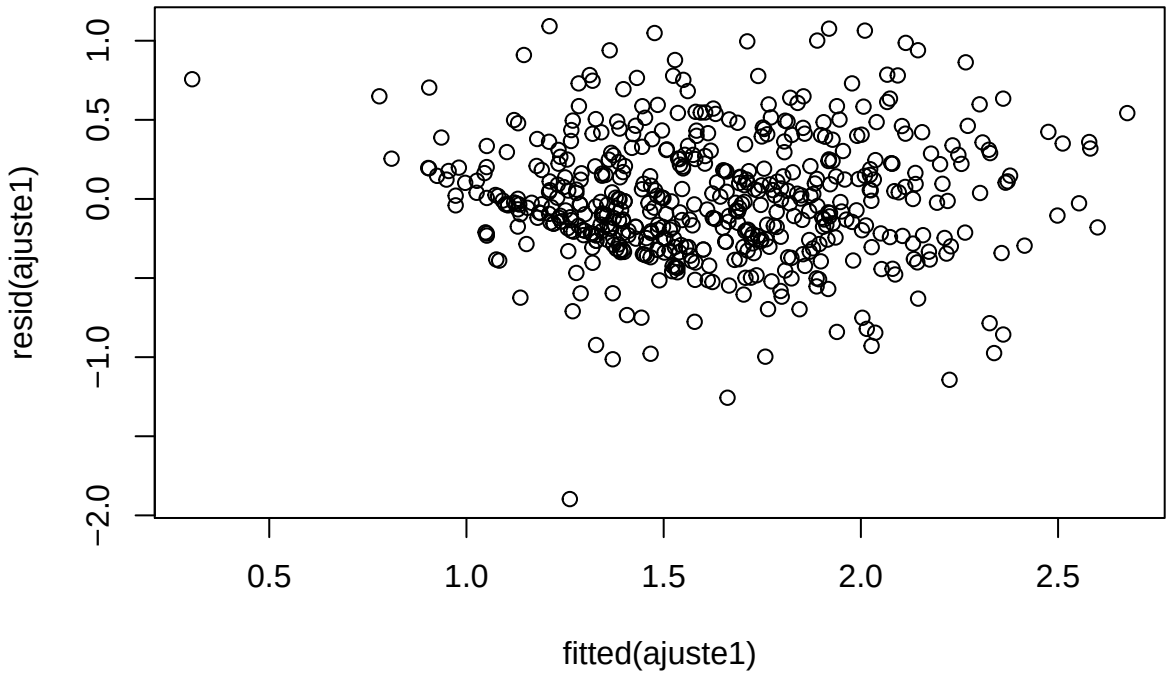
```
ajuste1 <- lm(lwage~casados+casadas+solteras+educ+exper+
                 expersq+tenure+tenursq,data = datos)

summary(ajuste1)
```

```
##
## Call:
## lm(formula = lwage ~ casados + casadas + solteras + educ + exper +
##     expersq + tenure + tenursq, data = datos)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.89697 -0.24060 -0.02689  0.23144  1.09197
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.3213780  0.1000090   3.213 0.001393 **
## casados      0.2126756  0.0553572   3.842 0.000137 ***
## casadas     -0.1982677  0.0578355  -3.428 0.000656 ***
## solteras    -0.1103502  0.0557421  -1.980 0.048272 *
## educ         0.0789103  0.0066945  11.787  < 2e-16 ***
## exper        0.0268006  0.0052428   5.112 4.50e-07 ***
## expersq     -0.0005352  0.0001104  -4.847 1.66e-06 ***
## tenure       0.0290875  0.0067620   4.302 2.03e-05 ***
## tenursq     -0.0005331  0.0002312  -2.306 0.021531 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3933 on 517 degrees of freedom
## Multiple R-squared:  0.4609, Adjusted R-squared:  0.4525
## F-statistic: 55.25 on 8 and 517 DF,  p-value: < 2.2e-16
```

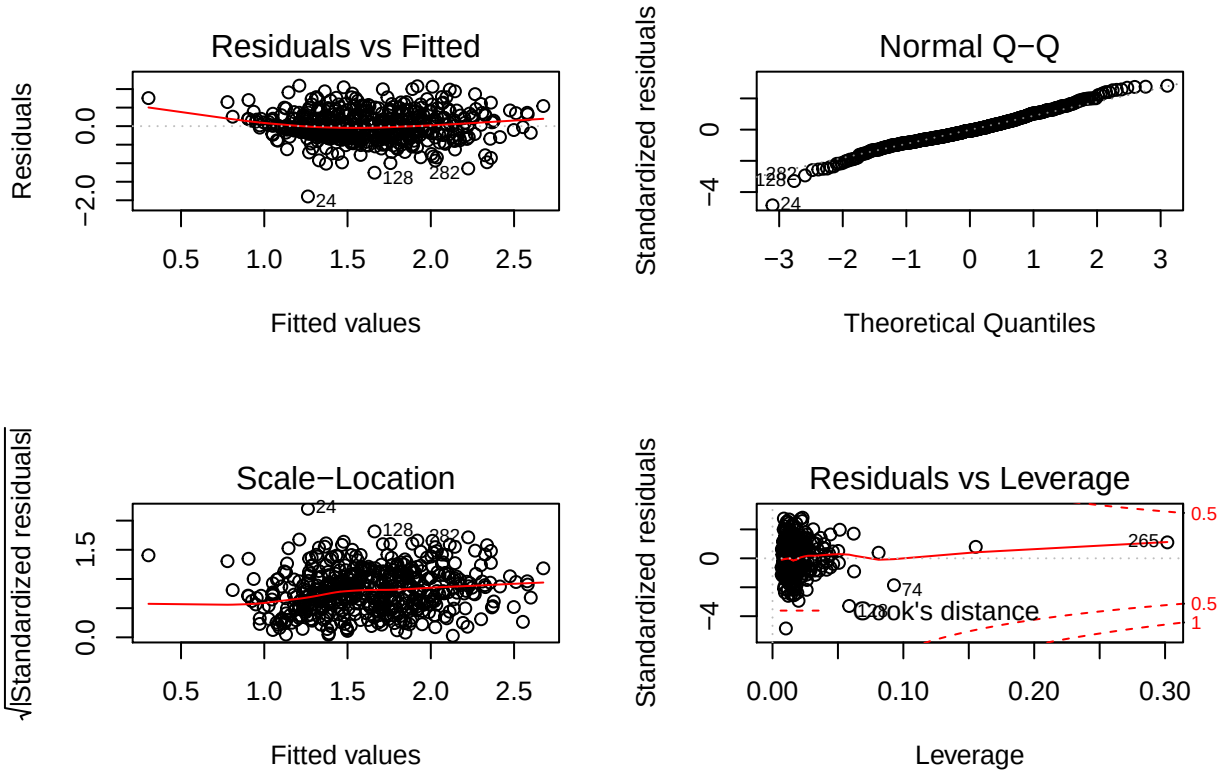
```
residuos <- resid(ajuste1)
sqresid <- residuos^2
y_techo <- fitted(ajuste1)
plot(y_techo,sqresid)
```




```
plot(fitted(ajuste1),resid(ajuste1))
```




```
# Usando el "default" de R:
par(mfrow=c(2,2))
plot(ajuste1)
```




```
par(mfrow=c(1,1))


library(sandwich)
library(lmtest)
#install.packages("lmSupport")
library(lmSupport)

# Test para ver si hay heterocedasticidad
residuos <- resid(ajuste1)
sqresid <- (residuos)^2
ajuste2 <- lm(sqresid~casados+casadas+solteras+educ+exper+expersq+tenure+tenursq,data = datos)
summary(ajuste2)
```

```
##
## Call:
## lm(formula = sqresid ~ casados + casadas + solteras + educ +
##     exper + expersq + tenure + tenursq, data = datos)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.2346 -0.1237 -0.0887  0.0202  3.4689
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.029e-02  6.893e-02   0.729  0.46603
## casados     -4.870e-02  3.816e-02  -1.276  0.20241
## casadas     -5.147e-02  3.986e-02  -1.291  0.19727
## solteras     4.162e-03  3.842e-02   0.108  0.91379
```

```
## educ         3.849e-03  4.614e-03   0.834  0.40462
## exper        1.008e-02  3.614e-03   2.790  0.00546 **
## expersq     -2.071e-04  7.611e-05  -2.720  0.00674 **
## tenure       4.763e-04  4.661e-03   0.102  0.91864
## tenursq      8.670e-05  1.594e-04   0.544  0.58672
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2711 on 517 degrees of freedom
## Multiple R-squared:  0.02507,    Adjusted R-squared:  0.009989
## F-statistic: 1.662 on 8 and 517 DF,  p-value: 0.105
```

```
# F =1.662 y pvalue=0.105 NO EXISTE HETEROCEDASTICIDAD
#Breusch-Pagan test

'bptest es igual a hettest en STATA'
```

```
## [1] "bptest es igual a hettest en STATA"
```

```
bptest(ajuste1)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  ajuste1
## BP = 13.189, df = 8, p-value = 0.1055
```

Para estimar errores robustos (como `robust` en stata):

```
coeftest(ajuste1, vcovHC(ajuste1,"HC0"))
```

```
##
## t test of coefficients:
##
##                Estimate  Std. Error t value  Pr(>|t|)
## (Intercept)  0.32137805  0.10852844  2.9612 0.0032049 **
## casados      0.21267564  0.05665095  3.7541 0.0001937 ***
## casadas     -0.19826765  0.05826506 -3.4029 0.0007186 ***
## solteras    -0.11035021  0.05662552 -1.9488 0.0518632 .
## educ         0.07891029  0.00735096 10.7347 < 2.2e-16 ***
## exper        0.02680057  0.00509497  5.2602 2.111e-07 ***
## expersq     -0.00053525  0.00010543 -5.0770 5.360e-07 ***
## tenure       0.02908752  0.00688128  4.2270 2.800e-05 ***
## tenursq     -0.00053314  0.00024159 -2.2068 0.0277671 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Autocorrelación

- ¿Cuál es la naturaleza de la autocorrelación?
- ¿Cuáles son las consecuencias teóricas y prácticas de la autocorrelación?
- ¿Cómo remediar el problema de la autocorrelación?

  Autocorrelación: correlación entre miembros de series de observaciones ordenadas en el tiempo [como en datos de series de tiempo] o en el espacio [como en datos de corte transversal]:

$$E(u_i, u_j) \neq 0 i \neq j$$

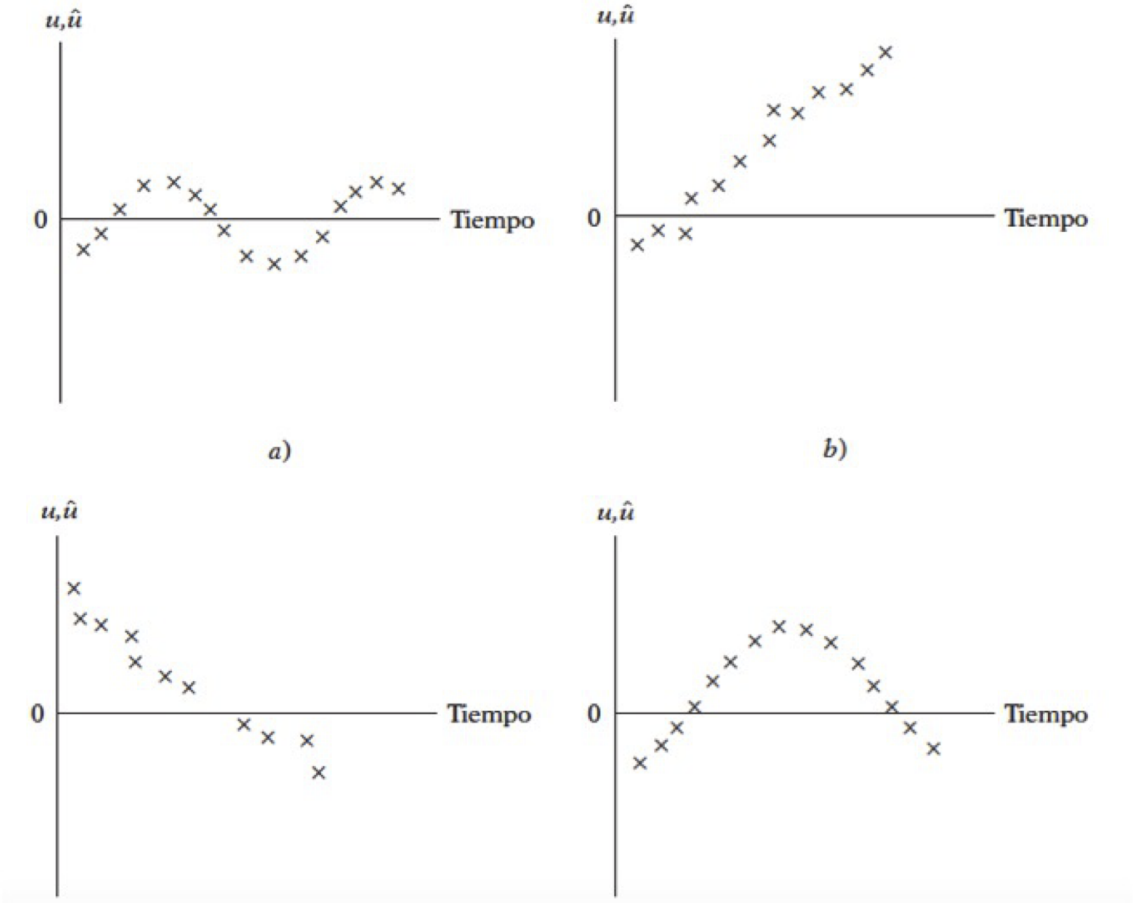
El supuesto es:

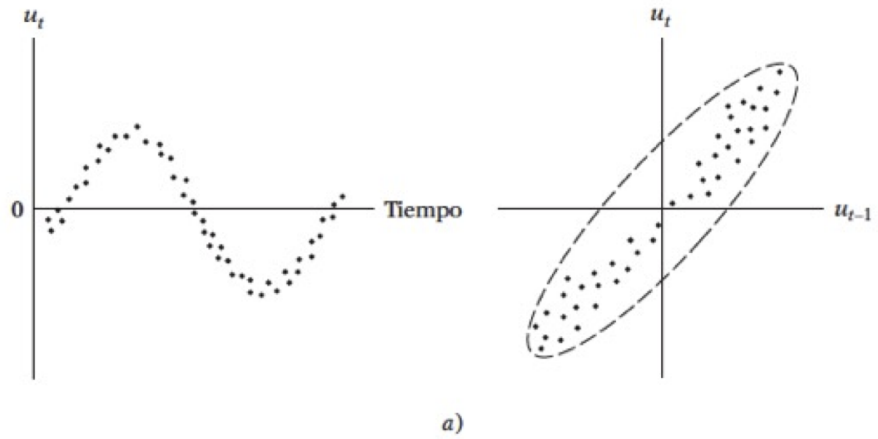
$$cov(u_i, u_j | x_i, x_j) = E(u_i, u_j) = 0 i \neq j$$

- Datos atípicos o aberrantes: Sensibilidad en las estimaciones
- Especificaciones del modelo: Omisión de variables importantes en el modelo.
- Asimentría: Surge a partir de la distribución de una o más regresoras en el modelo. Ejemplo: Distribución del ingreso *generalmente inequitativo*

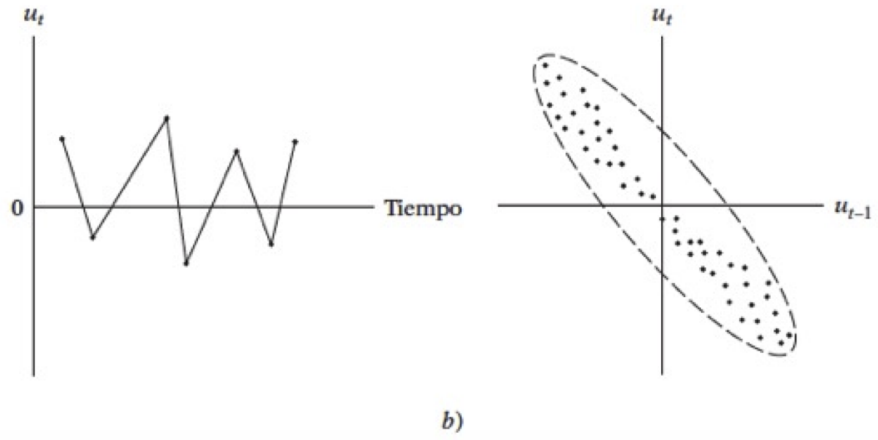
Cómo detectarla sesgos de especificación

Método gráfico





a)



b)

Veamos las pruebas de detección en un ejemplo

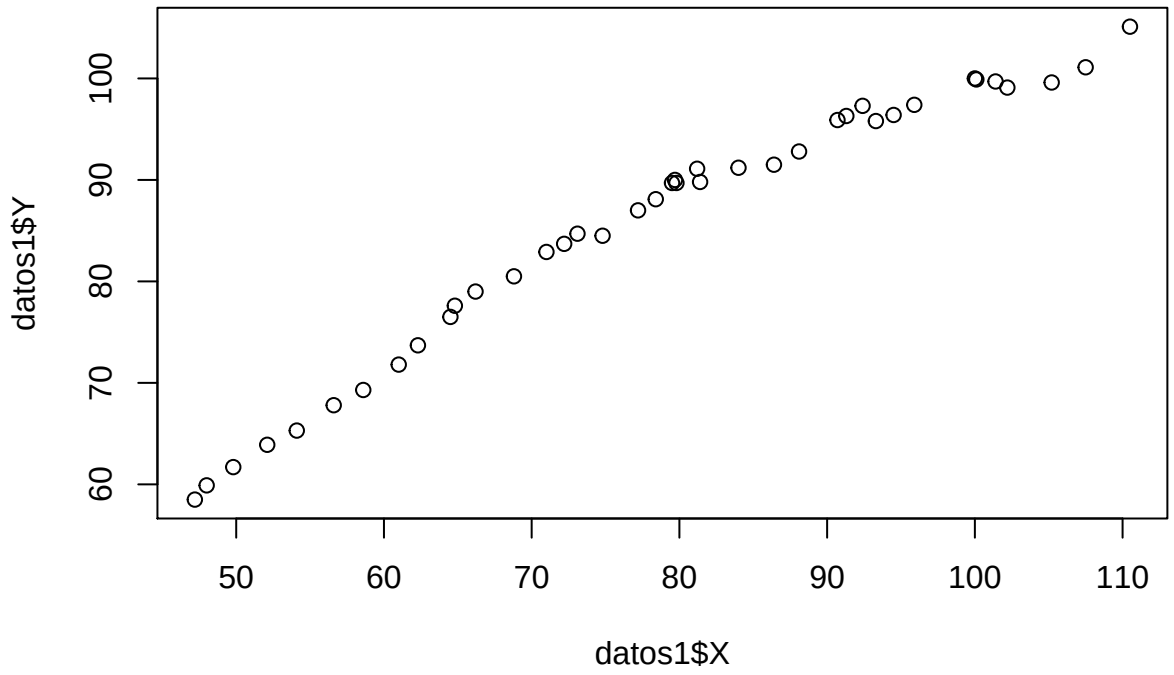
Ejemplo

Abrir la `tabla 12.4`. Veamos los datos en forma gráfica, y corramos el modelo:

- Y, índices de remuneración real por hora
- X, producción por hora X

```
uu <- "https://raw.githubusercontent.com/vmoprojs/DataLectures/master/tabla12_4.csv"
datos1<- read.csv(url(uu), sep=";",dec=".", header=T)
```

```
#Indice de compensacion real (salario real)
plot(datos1$X,datos1$Y)
```

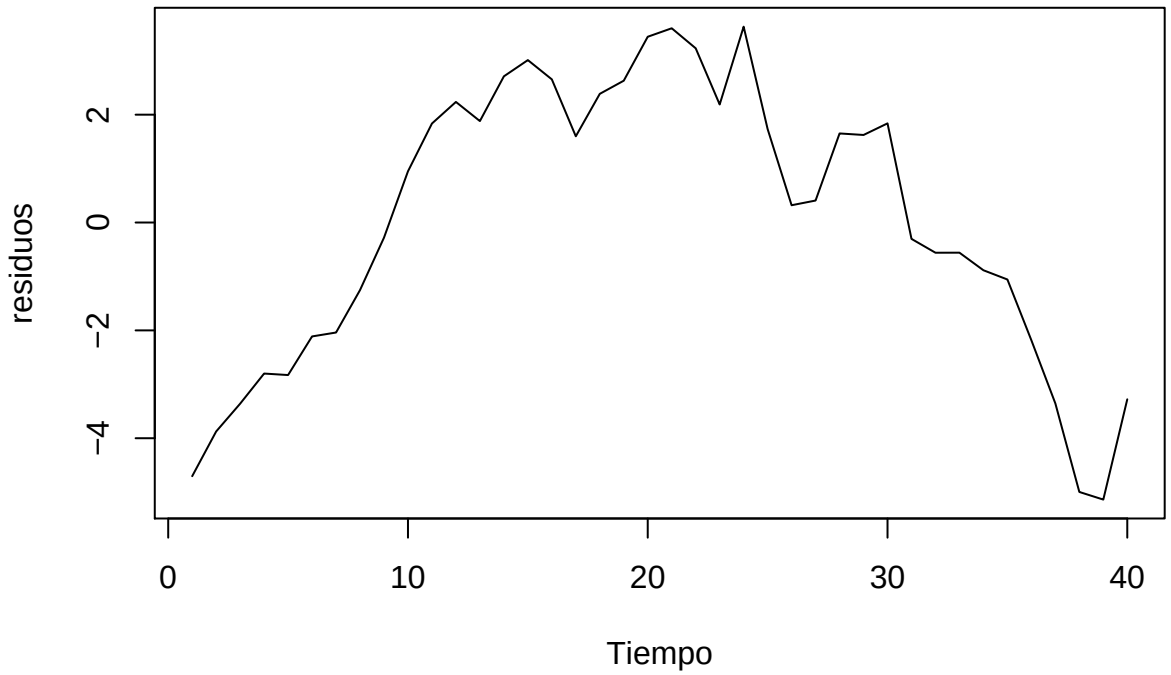



```
ajuste.indice<-lm(Y~X,data = datos1)
summary(ajuste.indice)
```

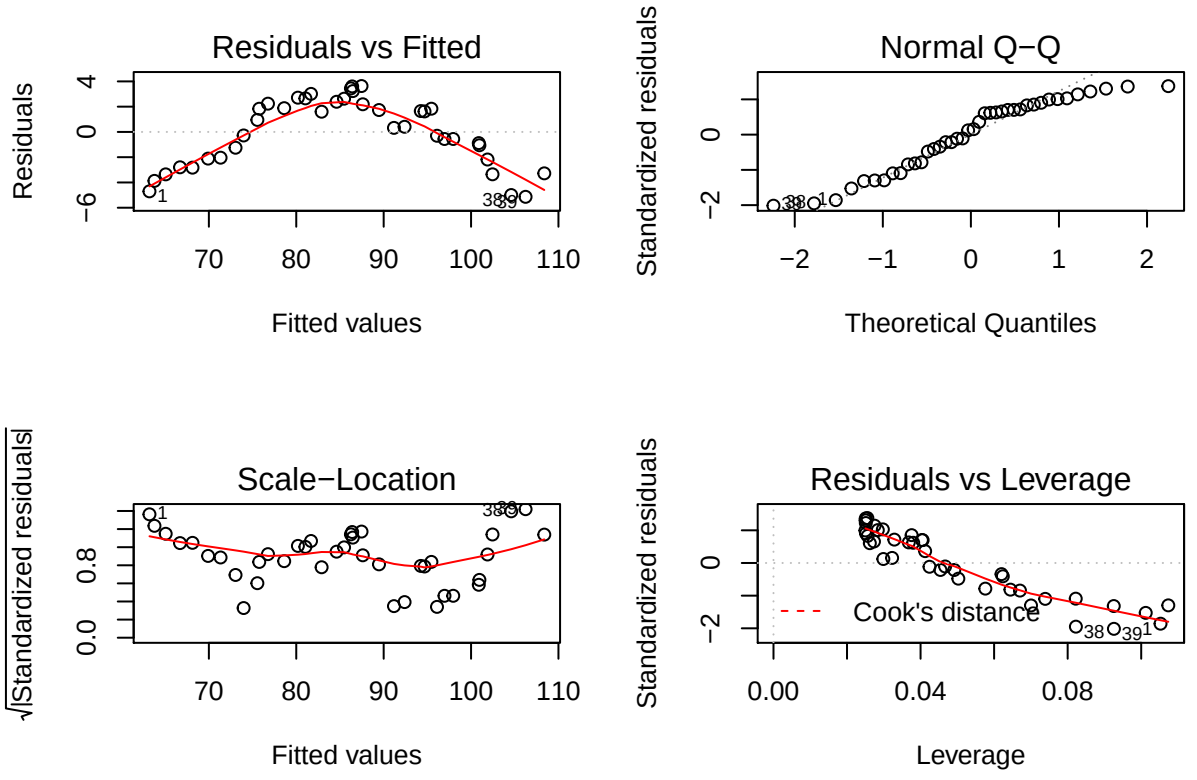
```
##
## Call:
## lm(formula = Y ~ X, data = datos1)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -5.138 -2.130  0.364  2.201  3.632
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  29.5192     1.9424   15.20   <2e-16 ***
## X             0.7137     0.0241   29.61   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.676 on 38 degrees of freedom
## Multiple R-squared:  0.9584, Adjusted R-squared:  0.9574
## F-statistic: 876.5 on 1 and 38 DF,  p-value: < 2.2e-16
```

Revisemos si hay autocorelación:

```
residuos<- resid(ajuste.indice)
plot(residuos,t="l",xlab="Tiempo")
```




```
par(mfrow = c(2,2))
plot(ajuste.indice)
```




```
par(mfrow = c(1,1))
```

- Los datos NO DEBEN TENER UN PATRON (si tienen patron, algo anda mal)
- En este caso se tiene un curva cuadrática, el modelo podría estar mal especificado.
- Podría ser que el modelo no se lineal o estar correlacionado

Veamos si se trata de una función cuadrática y cúbica

```
ajuste2 <- lm(Y~X+I(X^2),data = datos1)
summary(ajuste2)
```

```
##
## Call:
## lm(formula = Y ~ X + I(X^2), data = datos1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.58580 -0.76248  0.09209  0.68442  2.63570
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.622e+01  2.955e+00  -5.489 3.09e-06 ***
## X            1.949e+00  7.799e-02  24.987  < 2e-16 ***
## I(X^2)      -7.917e-03  4.968e-04 -15.936  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9669 on 37 degrees of freedom
## Multiple R-squared:  0.9947, Adjusted R-squared:  0.9944
## F-statistic:  3483 on 2 and 37 DF,  p-value: < 2.2e-16
```

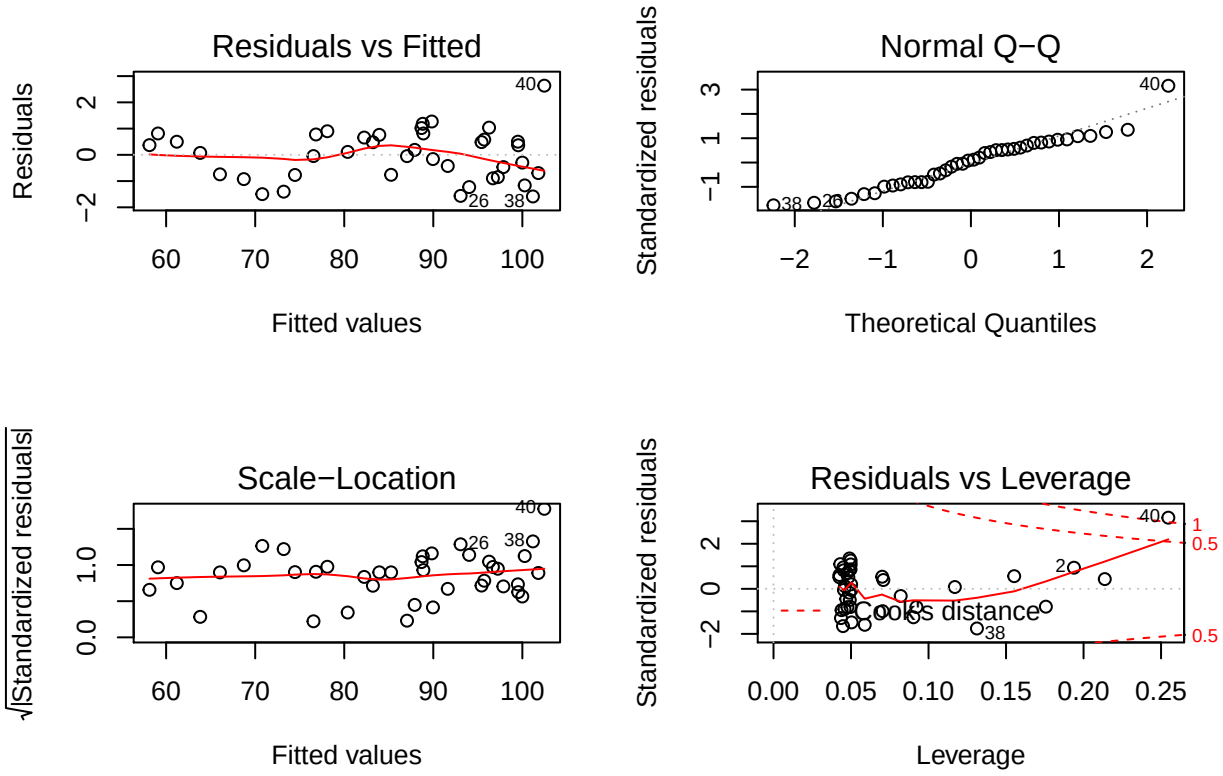
```
ajuste3 <- lm(Y~X+I(X^2)+I(X^3),data = datos1)
summary(ajuste3)
```

```
##
## Call:
## lm(formula = Y ~ X + I(X^2) + I(X^3), data = datos1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.63265 -0.79419  0.06568  0.66627  2.43810
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.222e+01  1.344e+01  -1.653 0.107060
## X            2.196e+00  5.466e-01   4.018 0.000286 ***
## I(X^2)      -1.119e-02  7.178e-03  -1.559 0.127658
## I(X^3)       1.398e-05  3.054e-05   0.458 0.649958
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9774 on 36 degrees of freedom
## Multiple R-squared:  0.9947, Adjusted R-squared:  0.9943
## F-statistic:  2272 on 3 and 36 DF,  p-value: < 2.2e-16
```

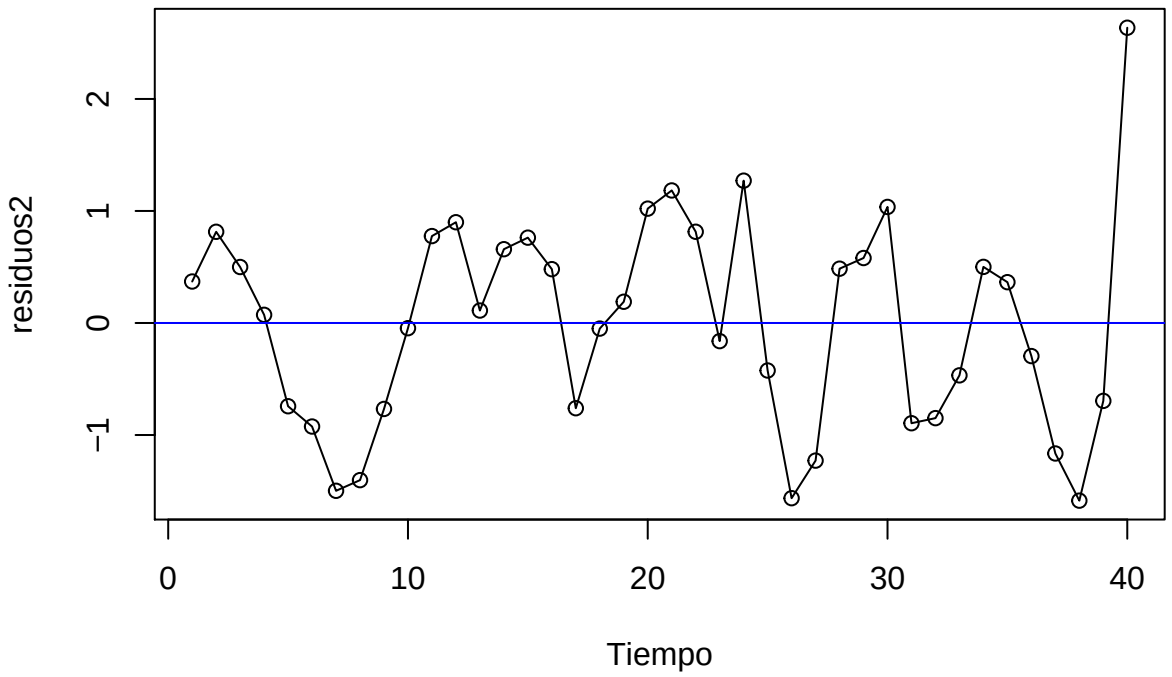
Nos quedamos con el `ajuste2`.

El gráfico de los valores ajustados, muestra que se ha eliminado el patron inicial

```
par(mfrow = c(2,2))
plot(ajuste2)
```

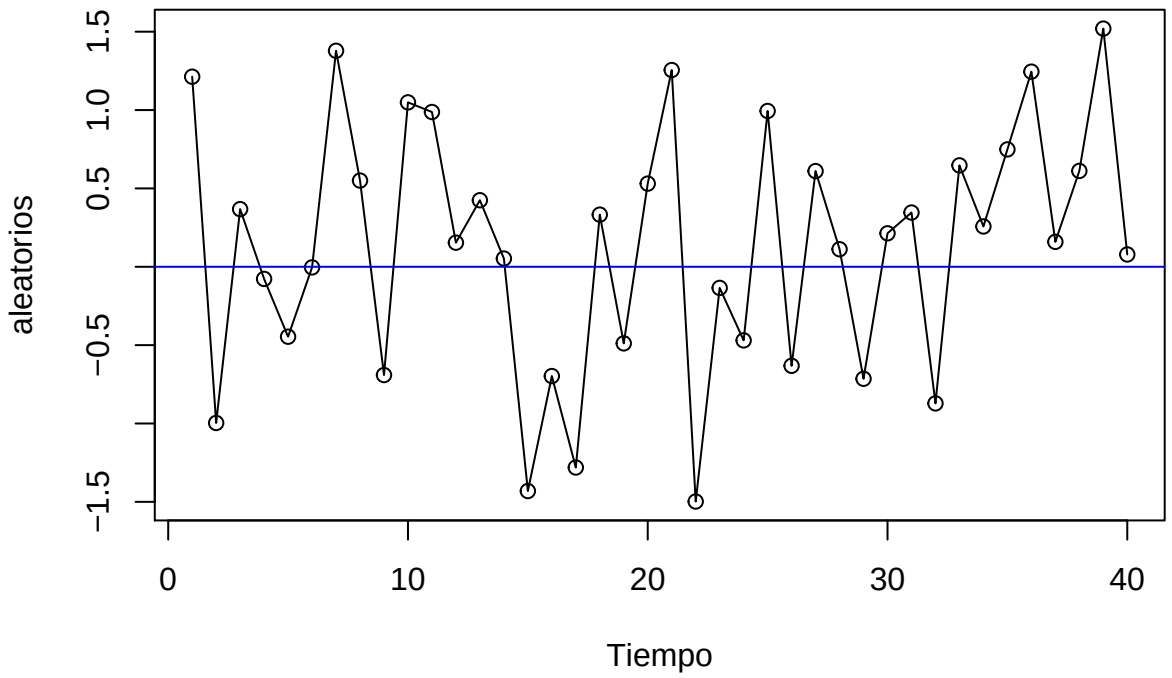



```r
par(mfrow = c(1,1))

residuos2 <- resid(ajuste2)
plot(residuos2,t="l",xlab="Tiempo")
points(residuos2)
abline(h=0,col="blue")
```



¿Cómo debe ser el gráfico?

```r
aleatorios=rnorm(40,0,1)
plot(aleatorios,t="l",xlab="Tiempo")
points(aleatorios)
abline(h=0,col="blue")
```



¿Se parece?

Ejemplo: Pruebas

H_o: No hay autocorrelación

```r
dwtest(ajuste2)
```

```
##
##  Durbin-Watson test
##
## data:  ajuste2
## DW = 1.03, p-value = 0.0001178
## alternative hypothesis: true autocorrelation is greater than 0
```

¿Cuál es la conclusión?

Otra prueba:

```r
# Ajuste Breuch Godfrey (Ho: No hay autocorrelación)
bgtest(ajuste2,order=4)
```

```
##
##  Breusch-Godfrey test for serial correlation of order up to 4
##
## data:  ajuste2
## LM test = 14.945, df = 4, p-value = 0.004817
```

Análisis Discriminante

Librerías usadas en esta técnica

```
library(car)
library(vegan)
library(mvnormtest)
library(MASS)
library(klaR)
```

El **análisis discriminante lineal (LDA)** y el discriminante lineal de Fisher relacionado son métodos utilizados en estadística, reconocimiento de patrones y aprendizaje automático para **encontrar una combinación lineal de características** que separa **dos o más clases de objetos o eventos**. La combinación resultante se puede usar como un clasificador lineal o, más comúnmente, para la **reducción de dimensionalidad antes de la clasificación posterior**.

Considere un conjunto de observaciones x (también llamadas características, atributos, variables o medidas) para cada muestra de un objeto o evento con una clase conocida $y \in \{0, 1\}$. Este conjunto de muestras se llama **conjunto de entrenamiento**. El problema de clasificación es **encontrar un buen predictor** para la clase y de cualquier muestra de la misma distribución (no necesariamente del conjunto de entrenamiento), dado solo una observación x.

Objetivos

- Determinar si existen diferencias significativas entre los perfiles de un conjunto de variables de dos o más grupos definidos a priori.
- Determinar cuál de las variables independientes cuantifica mejor las diferencias entre un grupo u otro.
- Establecer un procedimiento para clasificar a un individuo en base a los valores de un conjunto de variables independientes.

Posibles aplicaciones

- **Predicción de bancarrota**: en la predicción de bancarrota basada en razones contables y otras variables financieras, el análisis discriminante lineal fue el primer método estadístico aplicado para explicar sistemáticamente qué empresas entraron en bancarrota vs. sobrevivieron.
- **Comercialización**: en marketing, el análisis discriminante solía utilizarse para determinar los factores que distinguen diferentes tipos de clientes y/o productos sobre la base de encuestas u otras formas de datos recopilados.
- **Estudios biomédicos**: la principal aplicación del análisis discriminante en medicina es la evaluación del estado de gravedad de un paciente y el pronóstico del desenlace de la enfermedad. Por ejemplo, durante el análisis retrospectivo, los pacientes se dividen en grupos según la gravedad de la enfermedad, forma leve, moderada y grave. Luego, se estudian los resultados de los análisis clínicos y de laboratorio para revelar las variables que son estadísticamente diferentes en los grupos estudiados. Usando estas variables, se construyen funciones discriminantes que ayudan a clasificar objetivamente la enfermedad en un futuro paciente en una forma leve, moderada o severa.

Comparación con otras técnicas

La técnica más común para establecer relaciones, predecir y explicar variables son las técnicas de regresión. **El problema está cuando la variable a explicar no es una variable medible (o métrica)**; en este caso existen dos tipos de análisis con los que resolver el problema, el análisis discriminante y la regresión

logística. En ambos análisis tendremos una variable dependiente categórica y varias variables independientes numéricas.

En muchas ocasiones la variable categórica consta de dos grupos o clasificaciones (por ejemplo, bancarrota-no bancarrota). En otras situaciones la variable categórica tendrá tres o más subgrupos (e.g. bajo, medio y alto nivel de cierta dosis). La regresión logística o logito, en su forma básica está restringida a dos grupos frente al análisis discriminante que vale para más de dos.

Supuestos

- La *variable dependiente* (grupos) debe ser categórica en la que el número de grupos puede ser de dos o más, pero han de ser **mutuamente excluyentes y exhaustivos**. Aunque la variable dependiente puede ser originariamente numérica y que el investigador la cuantifique en términos de categorías.
- Las *variables independientes* numéricas se seleccionan identificando las variables en una investigación previa o mediante información a priori, de tal manera que se sepa que esas variables son importantes para predecir en qué grupo estará la variable dependiente. Se puede utilizar el análisis cluster para formar los grupos, pero se recomienda seguir los siguientes pasos: dividir los datos en 2 grupos, aplicar el análisis cluster en uno de ellos y utilizar los resultados en el DA para el segundo grupo de datos.
- Con respecto al *tamaño de las muestras*, se suele recomendar que los tamaños de cada grupo no sean muy diferentes, ya que con esto la probabilidad de pertenecer a un grupo o a otro puede variar considerablemente. Se necesita que al menos tengamos 4 o 5 veces más observaciones por grupo que el número de variables que utilicemos. Además, el número de observaciones en el grupo más pequeño debe ser mayor que el número de variables.
- También existen dos hipótesis previas que deben ser contrastadas, estas son: la **normalidad multivariante** y la de la **estructura de varianzas-covarianzas desconocidas pero iguales** (*homogeneidad de varianzas* entre grupos). Los datos que no cumplen el supuesto de normalidad pueden causar problemas en la estimación y en ese caso se sugiere utilizar la regresión logística. Si existen grandes desviaciones en las varianzas, se puede solucionar con la ampliación de la muestra o con técnicas de clasificación cuadráticas. **La homogeneidad de varianzas significa que la relación entre variables debe ser similar para los distintos grupos**. Por tanto, una variable no puede tener el mismo valor para todas las observaciones dentro de un grupo.
- Los datos además no deben presentar *multicolinealidad*, es decir, que dos o más variables independientes estén muy relacionadas. Si las variables tienen un valor de correlación de 0.9 o mayor se debe eliminar una de ellas.
- También se supone *linealidad* entre las variables ya que se utiliza la matriz de covarianza.

Si no se cumplen los supuestos de normalidad y homogeneidad, podemos utilizar una transformación logarítmica o de la raíz cuadrada (entre otras).

El modelo

El análisis discriminante implica un valor teórico como combinación lineal de dos o más variables independientes que discrimine entre los grupos definidos a priori. La discriminación se lleva a cabo estableciendo las ponderaciones del valor teórico de cada variable, de tal forma que **maximicen la varianza entre-grupos frente a la intra-grupos**. La combinación lineal o función discriminante, toma la siguiente forma:

$$D_i = a + W_1 X_{1,i} + W_2 X_{2,i} + \ldots + W_n X_{n,i}$$

donde: D_i es la puntuación discriminante (grupo de pertenencia) del individuo i-ésimo; a es una constante; W_j es la ponderación de la variable j-ésima. El resultado de esta función será para un conjunto de variables $X1, \ldots, Xn$ un valor de D que discrimine al individuo en un grupo u otro. Destacamos que el análisis

discriminante **proporcionará una función discriminate menos que los subgrupos que tengamos**, es decir, si la variable categórica tiene dos subgrupos, obtendremos una función discriminante, si tiene tres subgrupos obtendremos dos y así sucesivamente.

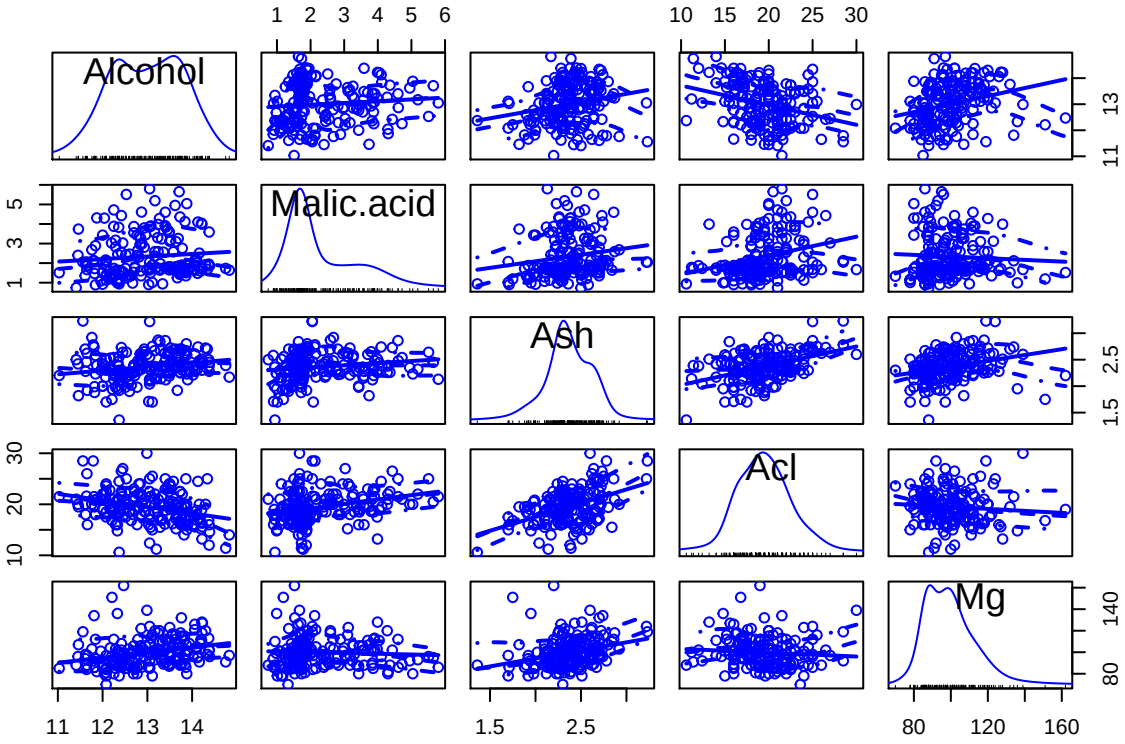
Ejemplo 1: clasificación de vinos

En este primer caso de estudio, el conjunto de datos del vino, tenemos 13 concentraciones químicas que describen muestras de vino de tres cultivos.

```
library(car)
# install.packages('rattle')
uu <- "https://gist.githubusercontent.com/tijptjik/9408623/raw/b237fa5848349a14a14e5d4107dc7897c21951f5/
wine <- read.csv(url(uu))
head(wine)
```

```
##   Wine Alcohol Malic.acid  Ash  Acl  Mg Phenols Flavanoids
## 1    1   14.23       1.71 2.43 15.6 127    2.80       3.06
## 2    1   13.20       1.78 2.14 11.2 100    2.65       2.76
## 3    1   13.16       2.36 2.67 18.6 101    2.80       3.24
## 4    1   14.37       1.95 2.50 16.8 113    3.85       3.49
## 5    1   13.24       2.59 2.87 21.0 118    2.80       2.69
## 6    1   14.20       1.76 2.45 15.2 112    3.27       3.39
##   Nonflavanoid.phenols Proanth Color.int  Hue   OD Proline
## 1                 0.28    2.29      5.64 1.04 3.92    1065
## 2                 0.26    1.28      4.38 1.05 3.40    1050
## 3                 0.30    2.81      5.68 1.03 3.17    1185
## 4                 0.24    2.18      7.80 0.86 3.45    1480
## 5                 0.39    1.82      4.32 1.04 2.93     735
## 6                 0.34    1.97      6.75 1.05 2.85    1450
```

```
scatterplotMatrix(wine[2:6])
```



El propósito del análisis discriminante lineal (LDA) en este ejemplo es encontrar las combinaciones lineales de las variables originales (las 13 concentraciones químicas aquí) que proporcionan la mejor separación posible entre los grupos (variedades de vino aquí) en nuestro conjunto de datos. El análisis discriminante lineal también se conoce como **análisis discriminante canónico**, o simplemente **análisis discriminante**.

Supuestos:

Homogeneidad de varianzas multivariante

```
library(vegan)
# seleccionamos las variables ambientales a analizar
env.pars2 <- as.matrix(wine[, 2:14])
# verificamos la homogeneidad multivariada de las matrices de covarianza intra-grupo
env.pars2.d1 <- dist(env.pars2)
env.MHV <- betadisper(env.pars2.d1, wine$Wine)
anova(env.MHV)
```

```
## Analysis of Variance Table
##
## Response: Distances
##            Df  Sum Sq Mean Sq F value    Pr(>F)
## Groups      2  190082   95041  8.3286 0.0003507 ***
## Residuals 175 1997003   11411
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
permutest(env.MHV)
```

```
##
## Permutation test for homogeneity of multivariate dispersions
## Permutation: free
## Number of permutations: 999
##
## Response: Distances
##            Df  Sum Sq Mean Sq      F N.Perm Pr(>F)
## Groups      2  190082   95041 8.3286    999  0.002 **
## Residuals 175 1997003   11411
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Conclusión: rechazo la hipótesis nula de homogeneidad intra-grupo. Se podría hacer transformaciones logarítmicas para enfrentar este asunto.

Normalidad multivariante

```
library(mvnormtest)
mshapiro.test(t(env.pars2))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Z
## W = 0.83696, p-value = 7.846e-13
```

Rechazamos la H_o de normalidad multivariante

Multicolinealidad

```r
as.dist(cor(env.pars2))
```

```
##                            Alcohol   Malic.acid          Ash          Acl
## Malic.acid            0.094396941
## Ash                   0.211544596  0.164045470
## Acl                  -0.310235137  0.288500403  0.443367187
## Mg                    0.270798226 -0.054575096  0.286586691 -0.083333089
## Phenols               0.289101123 -0.335166997  0.128979538 -0.321113317
## Flavanoids            0.236814928 -0.411006588  0.115077279 -0.351369860
## Nonflavanoid.phenols -0.155929467  0.292977133  0.186230446  0.361921719
## Proanth               0.136697912 -0.220746187  0.009651935 -0.197326836
## Color.int             0.546364195  0.248985344  0.258887259  0.018731981
## Hue                  -0.071747197 -0.561295689 -0.074666889 -0.273955223
## OD                    0.072343187 -0.368710428  0.003911231 -0.276768549
## Proline               0.643720037 -0.192010565  0.223626264 -0.440596931
##                              Mg      Phenols   Flavanoids
## Malic.acid
## Ash
## Acl
## Mg
## Phenols               0.214401235
## Flavanoids            0.195783770  0.864563500
## Nonflavanoid.phenols -0.256294049 -0.449935301 -0.537899612
## Proanth               0.236440610  0.612413084  0.652691769
## Color.int             0.199950006 -0.055136418 -0.172379398
## Hue                   0.055398196  0.433681335  0.543478566
## OD                    0.066003936  0.699949365  0.787193902
## Proline               0.393350849  0.498114880  0.494193127
##                      Nonflavanoid.phenols      Proanth    Color.int
## Malic.acid
## Ash
## Acl
## Mg
## Phenols
## Flavanoids
## Nonflavanoid.phenols
## Proanth                      -0.365845099
## Color.int                     0.139057013 -0.025249931
## Hue                          -0.262639631  0.295544253 -0.521813193
## OD                           -0.503269596  0.519067096 -0.428814942
## Proline                      -0.311385188  0.330416700  0.316100113
##                               Hue           OD
## Malic.acid
## Ash
## Acl
## Mg
## Phenols
## Flavanoids
## Nonflavanoid.phenols
## Proanth
## Color.int
## Hue
## OD                    0.565468293
## Proline               0.236183447  0.312761075
```

```r
library(MASS)
wine.lda <- lda(Wine ~ ., data=wine)
wine.lda
```

```
## Call:
## lda(Wine ~ ., data = wine)
##
## Prior probabilities of groups:
##         1         2         3
## 0.3314607 0.3988764 0.2696629
##
## Group means:
##    Alcohol Malic.acid      Ash      Acl       Mg  Phenols Flavanoids
## 1 13.74475   2.010678 2.455593 17.03729 106.3390 2.840169  2.9823729
## 2 12.27873   1.932676 2.244789 20.23803  94.5493 2.258873  2.0808451
## 3 13.15375   3.333750 2.437083 21.41667  99.3125 1.678750  0.7814583
##   Nonflavanoid.phenols  Proanth Color.int       Hue       OD   Proline
## 1             0.290000 1.899322  5.528305 1.0620339 3.157797 1115.7119
## 2             0.363662 1.630282  3.086620 1.0562817 2.785352  519.5070
## 3             0.447500 1.153542  7.396250 0.6827083 1.683542  629.8958
##
## Coefficients of linear discriminants:
##                                LD1           LD2
## Alcohol              -0.403399781  0.8717930699
## Malic.acid            0.165254596  0.3053797325
## Ash                  -0.369075256  2.3458497486
## Acl                   0.154797889 -0.1463807654
## Mg                   -0.002163496 -0.0004627565
## Phenols               0.618052068 -0.0322128171
## Flavanoids           -1.661191235 -0.4919980543
## Nonflavanoid.phenols -1.495818440 -1.6309537953
## Proanth               0.134092628 -0.3070875776
## Color.int             0.355055710  0.2532306865
## Hue                  -0.818036073 -1.5156344987
## OD                   -1.157559376  0.0511839665
## Proline              -0.002691206  0.0028529846
##
## Proportion of trace:
##    LD1    LD2
## 0.6875 0.3125
```

Esto significa que la primera función discriminante es una combinación lineal de las variables:

$$-0.403 * Alcohol + 0.165 * Malic \cdots - 0.003 * Proline$$

.

Por conveniencia, el valor de cada función discriminante (por ejemplo, la primera función discriminante) se escala de modo que su valor medio sea cero y su varianza sea uno.

La *proporción de traza* que se imprime cuando escribe `wine.lda` (la variable devuelta por la función `lda()`) es la separación porcentual lograda por cada función discriminante. Por ejemplo, para los datos del vino obtenemos los mismos valores que acabamos de calcular (68.75% y 31.25%).

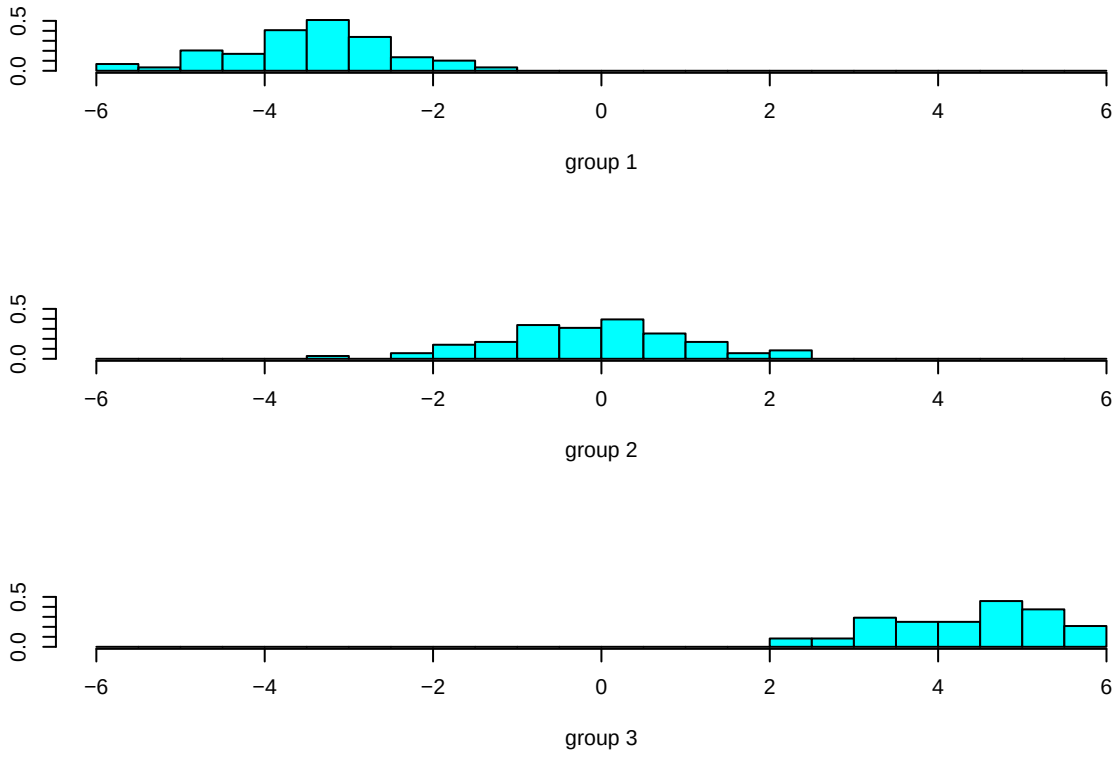
Histrogramas de resultado

Una buena forma de mostrar los resultados de un análisis discriminante lineal (LDA) es hacer un histograma

apilado de los valores de la función discriminante para las muestras de diferentes grupos (diferentes variedades de vino en nuestro ejemplo).

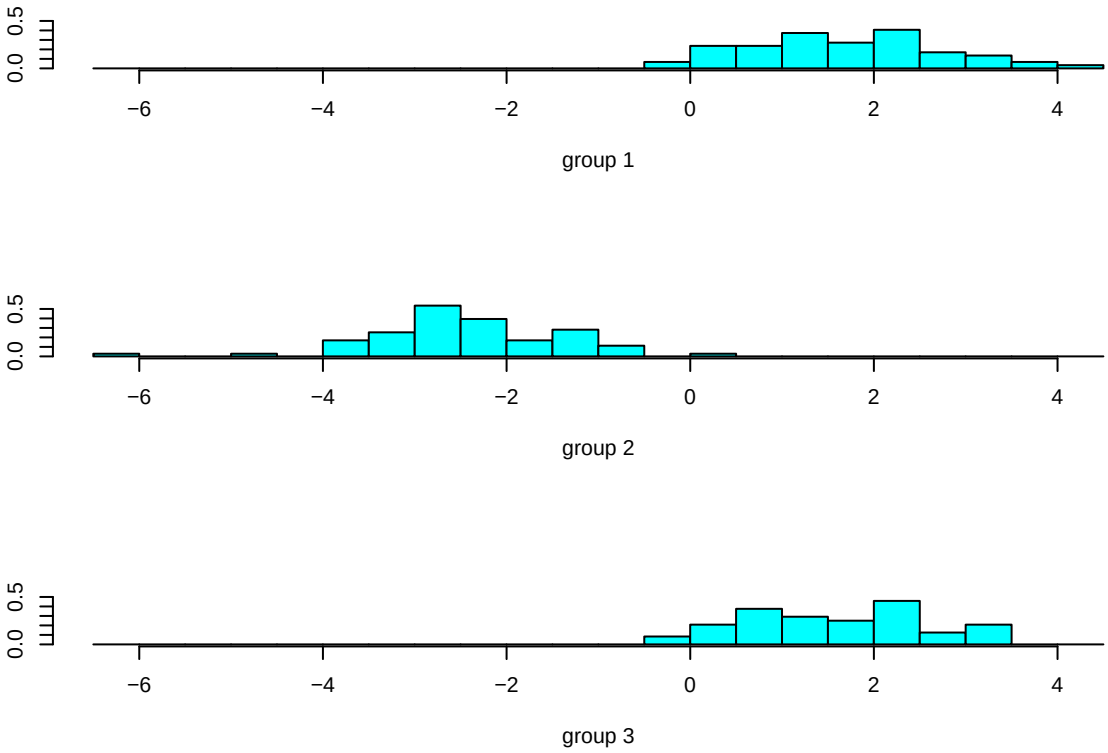
Podemos hacer esto usando la función `ldahist()` en `R`. Por ejemplo, para hacer un histograma apilado de los valores de la primera función discriminante para muestras de vino de los tres diferentes cultivares de vino, escribimos:

```
wine.lda.values <- predict(wine.lda)
ldahist(data = wine.lda.values$x[,1], g=wine$Wine)
```



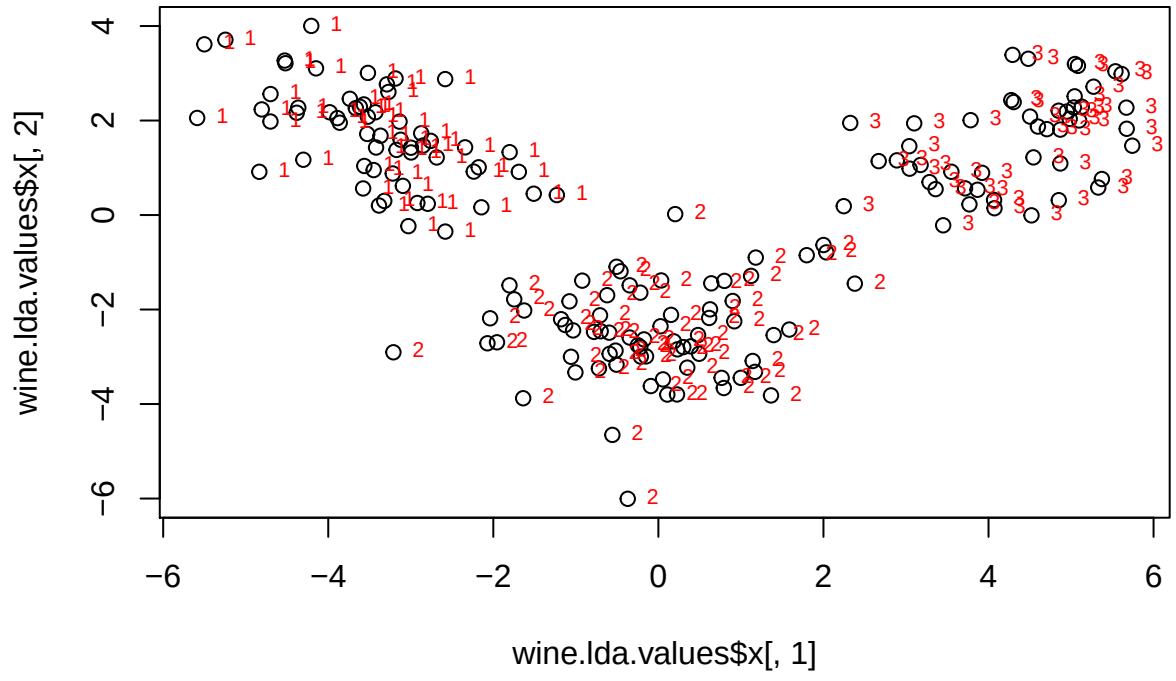
usando la segunda función discriminante:

```
ldahist(data = wine.lda.values$x[,2], g=wine$Wine)
```



Gráficos de las funciones discriminantes

```
plot(wine.lda.values$x[,1],wine.lda.values$x[,2]) # se realiza el grafico
text(wine.lda.values$x[,1],wine.lda.values$x[,2],wine$Wine,cex=0.7,pos=4,col="red") # agregamos etiique
```




```
spe.class <- predict(wine.lda)$class
(spe.table <-table(wine$Wine, spe.class))
```

```
##    spe.class
```

```
##      1  2  3
##   1 59  0  0
##   2  0 71  0
##   3  0  0 48
```

Ejemplo 2: Admisiones

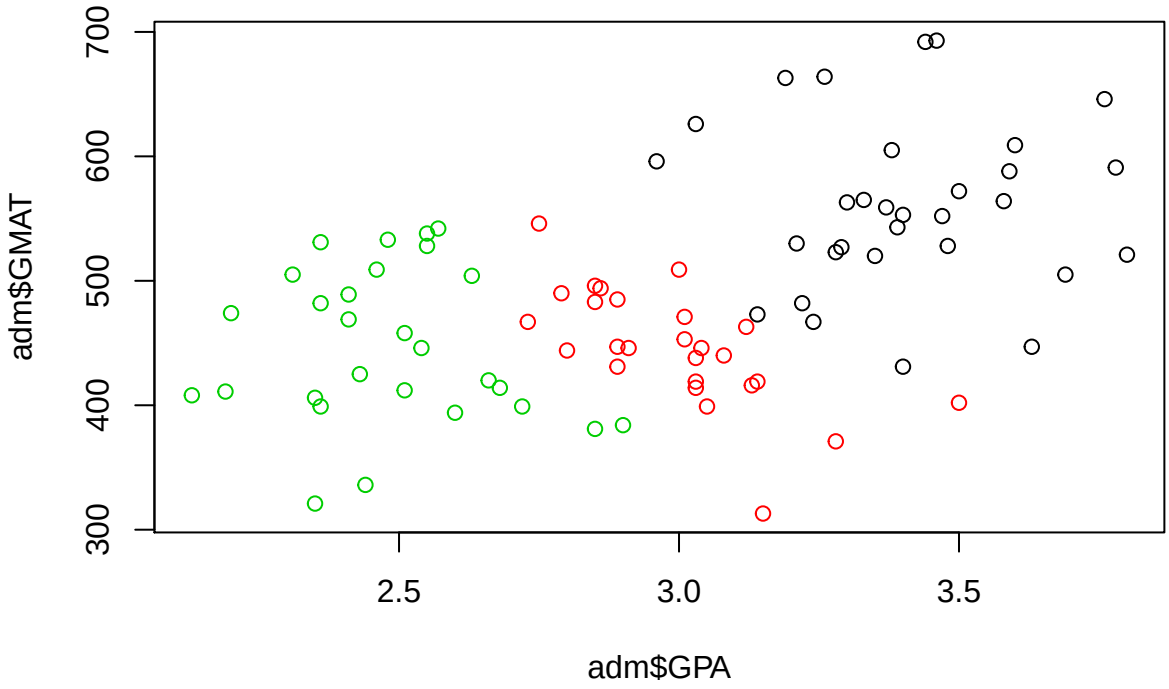
El conjunto de datos proporciona datos de admisión para los solicitantes a las escuelas de posgrado en los negocios. El objetivo es usar los puntajes de GPA y GMAT para predecir la probabilidad de admisión (admitir, no admitir y límite).

```
url <- 'http://www.biz.uiowa.edu/faculty/jledolter/DataMining/admission.csv'
admit <- read.csv(url)

head(admit)
```

```
##    GPA GMAT    De
## 1 2.96  596 admit
## 2 3.14  473 admit
## 3 3.22  482 admit
## 4 3.29  527 admit
## 5 3.69  505 admit
## 6 3.46  693 admit
```

Realizamos un gráfico de los datos:

```
adm <- data.frame(admit)
plot(adm$GPA,adm$GMAT,col=adm$De)
```



Supuestos:

Homogeneidad de varianzas multivariante

```r
library(vegan)
# seleccionamos las variables ambientales a analizar
env.pars2 <- as.matrix(adm[, 1:2])
# verificamos la homogeneidad multivariada de las matrices de covarianza intra-grupo
env.pars2.d1 <- dist(env.pars2)
env.MHV <- betadisper(env.pars2.d1, adm$De)
anova(env.MHV)
```

```
## Analysis of Variance Table
##
## Response: Distances
##           Df Sum Sq Mean Sq F value  Pr(>F)
## Groups     2   6224  3112.0  2.4009 0.09698 .
## Residuals 82 106285  1296.2
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
permutest(env.MHV)
```

```
##
## Permutation test for homogeneity of multivariate dispersions
## Permutation: free
## Number of permutations: 999
##
## Response: Distances
##           Df Sum Sq Mean Sq      F N.Perm Pr(>F)
## Groups     2   6224  3112.0 2.4009    999  0.084 .
## Residuals 82 106285  1296.2
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Conclusión: no rechazo la hipótesis nula de homogeneidad intra-grupo.

Normalidad multivariante

```r
library(mvnormtest)
mshapiro.test(t(env.pars2))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Z
## W = 0.98854, p-value = 0.6623
```

No rechazamos la H_o de normalidad multivariante

Multicolinealidad

```r
as.dist(cor(env.pars2))
```

```
##            GPA
## GMAT 0.4606332
```

```r
library(MASS)
m1 <- lda(De~.,adm)
m1
```

```
## Call:
## lda(De ~ ., data = adm)
```

```
##
## Prior probabilities of groups:
##     admit    border  notadmit
## 0.3647059 0.3058824 0.3294118
##
## Group means:
##               GPA     GMAT
## admit    3.403871 561.2258
## border   2.992692 446.2308
## notadmit 2.482500 447.0714
##
## Coefficients of linear discriminants:
##              LD1         LD2
## GPA  5.008766354  1.87668220
## GMAT 0.008568593 -0.01445106
##
## Proportion of trace:
##    LD1    LD2
## 0.9673 0.0327
```

Comenta los resultados.

Realizamos una predicción:

```
predict(m1,newdata=data.frame(GPA=3.21,GMAT=497))
```

```
## $class
## [1] admit
## Levels: admit border notadmit
##
## $posterior
##       admit    border     notadmit
## 1 0.5180421 0.4816015 0.0003563717
##
## $x
##        LD1      LD2
## 1 1.252409 0.318194
```

Análisis discrimante cuadrático: Se trata de un procedimiento más robusto que el lineal, y es útil **cuando las matrices de covarianza no son iguales**. Se basa en la distancia de Mahalanobis al cuadrado respecto al centro del grupo.

```
m2 <- qda(De~.,adm)
m2
```

```
## Call:
## qda(De ~ ., data = adm)
##
## Prior probabilities of groups:
##     admit    border  notadmit
## 0.3647059 0.3058824 0.3294118
##
## Group means:
##               GPA     GMAT
## admit    3.403871 561.2258
## border   2.992692 446.2308
## notadmit 2.482500 447.0714
```

Realizamos la predicción

```
predict(m2,newdata=data.frame(GPA=3.21,GMAT=497))
```

```
## $class
## [1] admit
## Levels: admit border notadmit
##
## $posterior
##       admit    border     notadmit
## 1 0.9226763 0.0768693 0.0004544468
```

¿Qué modelo es el mejor?

Para responder a esta pregunta, evaluamos el análisis discriminante lineal seleccionando aleatoriamente 60 de 85 estudiantes, estimando los parámetros en los datos de entrenamiento y clasificando a los 25 estudiantes restantes de la muestra retenida. Repetimos esto 100 veces

```
n <- 85
nt <- 60
neval <-n-nt
rep <- 100

### LDA
set.seed(123456789)
errlin <- dim(rep)
for (k in 1:rep) {
train <- sample(1:n,nt)
## linear discriminant analysis
m1 <- lda(De~.,adm[train,])
predict(m1,adm[-train,])$class
tablin <- table(adm$De[-train],predict(m1,adm[-train,])$class)
errlin[k] <- (neval-sum(diag(tablin)))/neval
}
merrlin <- mean(errlin) #media del error lineal
merrlin
```

```
## [1] 0.0916
```

Ahora en el QDA:

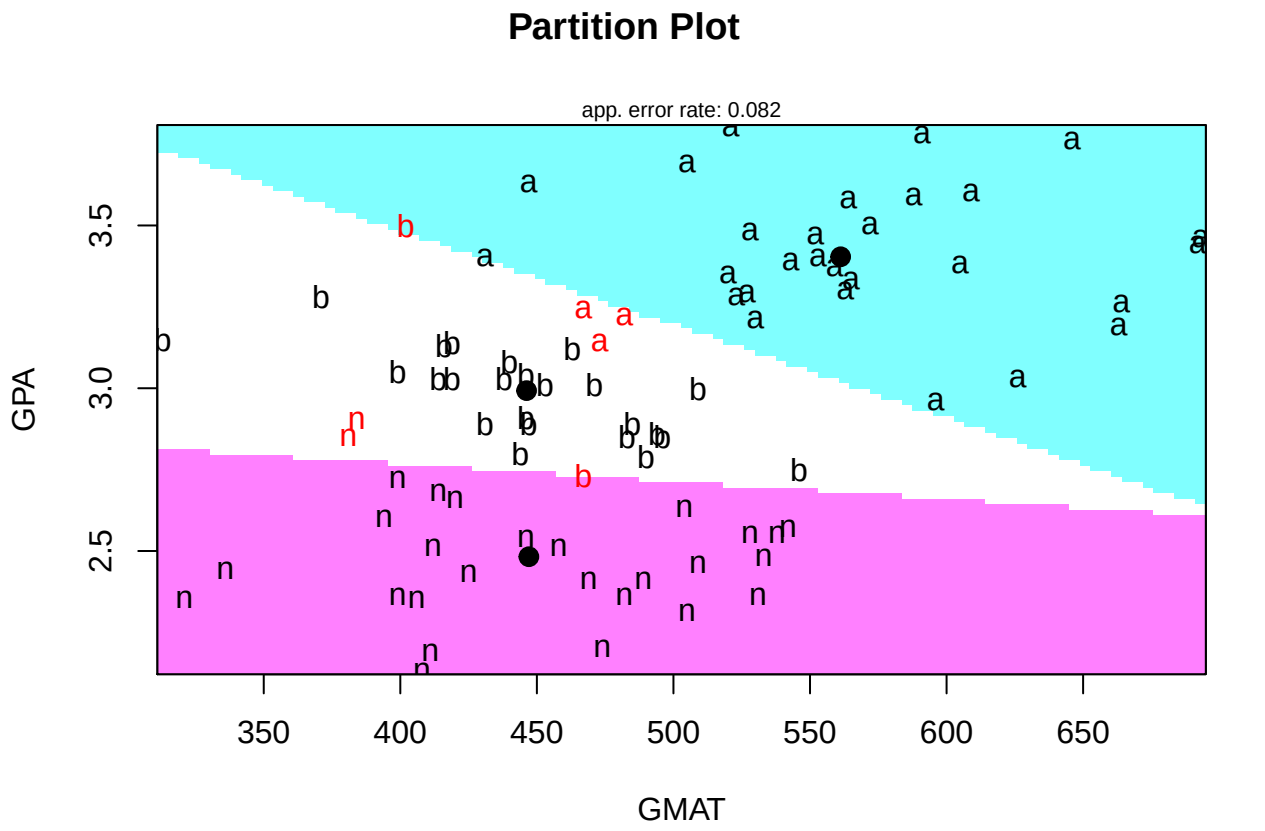
```
### QDA
set.seed(123456789)
errqda <- dim(rep)
for (k in 1:rep) {
train <- sample(1:n,nt)
## quadratic discriminant analysis
m1 <- qda(De~.,adm[train,])
predict(m1,adm[-train,])$class
tablin <- table(adm$De[-train],predict(m1,adm[-train,])$class)
errqda[k] <- (neval-sum(diag(tablin)))/neval
}
merrqda <- mean(errlin)
merrqda
```

```
## [1] 0.0916
```

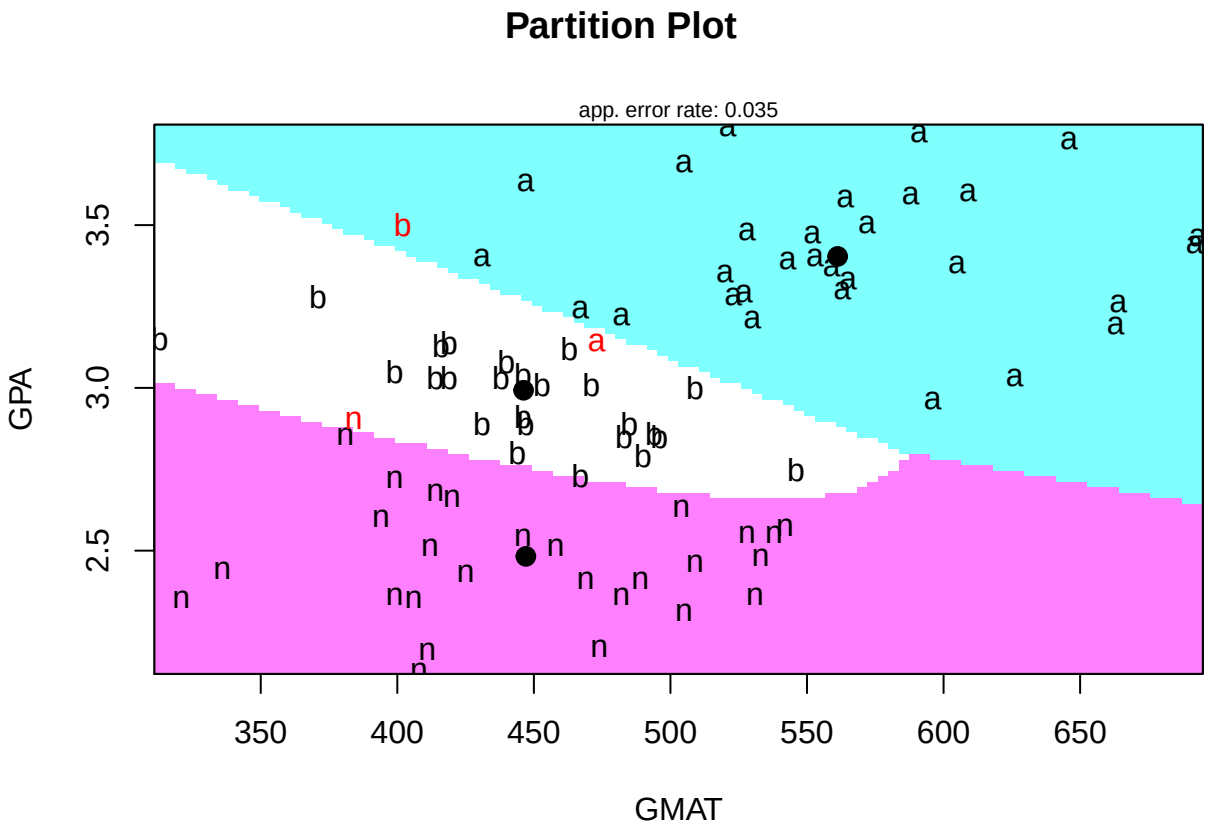
Logramos una tasa de clasificación errónea del 10.2% en ambos casos. R también nos da algunas herramientas

de visualización. Por ejemplo en la librería `klaR`:

```
# Gráficos exploratorios para LDA or QDA
#install.packages('klaR')
library(klaR)
partimat(De~.,data=adm,method="lda")
```




```
partimat(De~.,data=adm,method="qda")
```



Ejemplo 3: Score de crédito de un banco alemán

El conjunto de datos de crédito alemán se obtuvo del Repositorio de aprendizaje automático UCI. El conjunto de datos, que contiene atributos y resultados sobre 1000 solicitudes de préstamo, fue proporcionado en 1994 por el Profesor Dr. Hans Hofmann del Institut fuer Statistik und Oekonometrie de la Universidad de Hamburgo. Ha servido como un importante conjunto de datos de prueba para varios algoritmos de puntuación de crédito. Una descripción de las variables se da en `germancreditDescription.docx` de `DataLectures`. Comenzamos cargando los datos:

```
## read data
credit <- read.csv("http://www.biz.uiowa.edu/faculty/jledolter/DataMining/germancredit.csv")
head(credit,2) # Mira la codificación en el lugar indicado
```

```
##   Default checkingstatus1 duration history purpose amount savings employ
## 1       0            A11        6     A34     A43   1169     A65    A75
## 2       1            A12       48     A32     A43   5951     A61    A73
##   installment status others residence property age otherplans housing
## 1           4    A93   A101         4     A121  67       A143    A152
## 2           2    A92   A101         2     A121  22       A143    A152
##   cards  job liable tele foreign
## 1     2 A173      1 A192    A201
## 2     1 A173      1 A191    A201
```

Como se puede ver, solo las variables: duración, cantidad, plazos y edad son numéricas. Con los restantes (indicadores) los supuestos de una distribución normal serían, en el mejor de los casos, débiles; por lo tanto, estas variables no se consideran aquí.

```r
cred1 <- credit[, c("Default","duration","amount","installment","age")]
head(cred1)
```

```
##   Default duration amount installment age
## 1       0        6   1169           4  67
## 2       1       48   5951           2  22
## 3       0       12   2096           2  49
## 4       0       42   7882           2  45
## 5       1       24   4870           3  53
## 6       0       36   9055           2  35
```

```r
summary(cred1)
```

```
##     Default       duration        amount       installment
##  Min.   :0.0   Min.   : 4.0   Min.   :  250   Min.   :1.000
##  1st Qu.:0.0   1st Qu.:12.0   1st Qu.: 1366   1st Qu.:2.000
##  Median :0.0   Median :18.0   Median : 2320   Median :3.000
##  Mean   :0.3   Mean   :20.9   Mean   : 3271   Mean   :2.973
##  3rd Qu.:1.0   3rd Qu.:24.0   3rd Qu.: 3972   3rd Qu.:4.000
##  Max.   :1.0   Max.   :72.0   Max.   :18424   Max.   :4.000
##       age
##  Min.   :19.00
##  1st Qu.:27.00
##  Median :33.00
##  Mean   :35.55
##  3rd Qu.:42.00
##  Max.   :75.00
```

Transformemos los datos en un data.frame

```r
cred1 <- data.frame(cred1)
```

- Realiza las pruebas de los supuestos y comenta los resultados
- Estima y compara lda con qda
- Estima la matriz de confusión
- ¿Usarías este modelo para una aplicación real?

Referencias

Schumacker, Randall E. 2015. *Using R with Multivariate Statistics*. Sage Publications.