

Práctica AE

Antonio Galián Gálvez

2024-10-28

```
library(gplots)

##
## Attaching package: 'gplots'
## The following object is masked from 'package:stats':
##
##      lowess
```

0. Cargamos los datos y eliminamos la columna train

```
# Cargamos los datos con separador de tabulador
datos <- read.delim("prostate.data.txt", header = TRUE, sep = "\t")

# Eliminamos la columna train
datos <- datos[, -ncol(datos)]
```

1. Exploración de datos

```
# Vemos las variables que hay
ncol(datos)

## [1] 10

# Eliminamos la columna id
datos <- datos[, -1]

# Comprobamos si hay NA
sum(is.na(datos))

## [1] 0

# Comprobamos si las variables estan estandarizadas
summary(datos)
```

```
##      lccavol      lweight      age      lbph
## Min.   :-1.3471  Min.    :2.375  Min.    :41.00  Min.    :-1.3863
## 1st Qu.: 0.5128  1st Qu.:3.376  1st Qu.:60.00  1st Qu.: -1.3863
## Median : 1.4469  Median :3.623  Median :65.00  Median : 0.3001
## Mean   : 1.3500  Mean   :3.629  Mean   :63.87  Mean   : 0.1004
## 3rd Qu.: 2.1270  3rd Qu.:3.876  3rd Qu.:68.00  3rd Qu.: 1.5581
## Max.    : 3.8210  Max.    :4.780  Max.    :79.00  Max.    : 2.3263
##      svi      lcp      gleason      pgg45
```

```
## Min. :0.0000 Min. :-1.3863 Min. :6.000 Min. : 0.00
## 1st Qu.:0.0000 1st Qu.: -1.3863 1st Qu.:6.000 1st Qu.: 0.00
## Median :0.0000 Median :-0.7985 Median :7.000 Median : 15.00
## Mean :0.2165 Mean :-0.1794 Mean :6.753 Mean : 24.38
## 3rd Qu.:0.0000 3rd Qu.: 1.1787 3rd Qu.:7.000 3rd Qu.: 40.00
## Max. :1.0000 Max. : 2.9042 Max. :9.000 Max. :100.00
##      lpsa
## Min. :-0.4308
## 1st Qu.: 1.7317
## Median : 2.5915
## Mean : 2.4784
## 3rd Qu.: 3.0564
## Max. : 5.5829
```

```
dim(datos)
```

```
## [1] 97 9
```

```
names(datos)
```

```
## [1] "lcavol" "lweight" "age" "lbph" "svi" "lcp" "gleason"
## [8] "pgg45" "lpsa"
```

```
str(datos)
```

```
## 'data.frame': 97 obs. of 9 variables:
## $ lcavol : num -0.58 -0.994 -0.511 -1.204 0.751 ...
## $ lweight: num 2.77 3.32 2.69 3.28 3.43 ...
## $ age : int 50 58 74 58 62 50 64 58 47 63 ...
## $ lbph : num -1.39 -1.39 -1.39 -1.39 -1.39 ...
## $ svi : int 0 0 0 0 0 0 0 0 0 0 ...
## $ lcp : num -1.39 -1.39 -1.39 -1.39 -1.39 ...
## $ gleason: int 6 6 7 6 6 6 6 6 6 6 ...
## $ pgg45 : int 0 0 20 0 0 0 0 0 0 0 ...
## $ lpsa : num -0.431 -0.163 -0.163 -0.163 0.372 ...
```

```
summary(datos)
```

```
##      lcavol      lweight      age      lbph
## Min. :-1.3471 Min. :2.375 Min. :41.00 Min. :-1.3863
## 1st Qu.: 0.5128 1st Qu.:3.376 1st Qu.:60.00 1st Qu.: -1.3863
## Median : 1.4469 Median :3.623 Median :65.00 Median : 0.3001
## Mean : 1.3500 Mean :3.629 Mean :63.87 Mean : 0.1004
## 3rd Qu.: 2.1270 3rd Qu.:3.876 3rd Qu.:68.00 3rd Qu.: 1.5581
## Max. : 3.8210 Max. :4.780 Max. :79.00 Max. : 2.3263
##      svi      lcp      gleason      pgg45
## Min. :0.0000 Min. :-1.3863 Min. :6.000 Min. : 0.00
## 1st Qu.:0.0000 1st Qu.: -1.3863 1st Qu.:6.000 1st Qu.: 0.00
## Median :0.0000 Median :-0.7985 Median :7.000 Median : 15.00
## Mean :0.2165 Mean :-0.1794 Mean :6.753 Mean : 24.38
## 3rd Qu.:0.0000 3rd Qu.: 1.1787 3rd Qu.:7.000 3rd Qu.: 40.00
## Max. :1.0000 Max. : 2.9042 Max. :9.000 Max. :100.00
##      lpsa
## Min. :-0.4308
## 1st Qu.: 1.7317
## Median : 2.5915
## Mean : 2.4784
```

```
## 3rd Qu.: 3.0564
## Max.    : 5.5829
```

- Hay 10 variables, 9 si quitamos el id del paciente
- Las variables son numéricas
- La variable correspondiente al identificador del paciente es la primera columna
- No hay valores nulos
- Las variables no están ni normalizadas ni estandarizadas
- Hay variables que están en escala logarítmica ya que algunas variables tienen valores negativos a pesar de estar definidas estrictamente positivas, como la concentración en ng/m

2. Análisis de variable categóricas

```
# Convertimos las variables en factores
datos$svi <- as.factor(datos$svi)
datos$gleason <- as.factor(datos$gleason)
datos$age <- as.factor(datos$age)

# Comprobamos que las variables son categóricas
str(datos)
```

```
## 'data.frame': 97 obs. of 9 variables:
## $ lcavol : num -0.58 -0.994 -0.511 -1.204 0.751 ...
## $ lweight: num 2.77 3.32 2.69 3.28 3.43 ...
## $ age : Factor w/ 31 levels "41","43","44",...: 6 11 27 11 15 6 17 11 4 16 ...
## $ lbph : num -1.39 -1.39 -1.39 -1.39 -1.39 ...
## $ svi : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
## $ lcp : num -1.39 -1.39 -1.39 -1.39 -1.39 ...
## $ gleason: Factor w/ 4 levels "6","7","8","9": 1 1 2 1 1 1 1 1 1 1 ...
## $ pgg45 : int 0 0 20 0 0 0 0 0 0 0 ...
## $ lpsa : num -0.431 -0.163 -0.163 -0.163 0.372 ...
```

3. Análisis de frecuencias

- ¿Qué porcentaje de pacientes con la puntuación de Gleason igual a 7, presenta índice igual svi igual a 0?

```
# Seleccionamos los pacientes con la puntuación de Gleason igual a 7 y los que tienen svi igual a 0 de
datos.gleason7 <- datos[datos$gleason == "7", ]
datos.gleason7.svi0 <- datos.gleason7[datos.gleason7$svi == "0", ]

# Vemos los pacientes que hay en datos.gleason_7_0 y en datos
patients.gleason7.svi0 <- nrow(datos.gleason7.svi0)
patients <- nrow(datos)

# Dividimos la cantidad de pacientes filtrados entre el total
porcentaje <- patients.gleason7.svi0 / patients * 100
porcentaje
```

```
## [1] 38.14433
```

Vemos que el porcentaje es del 38.14433%.

- ¿Qué porcentaje de pacientes con índice svi igual a 0 tiene la puntuación de Gleason igual a 7?

```
# Seleccionamos los individuos con svi igual a 0 y con gleason igual a 7 dentro de estos
datos.svi0 <- datos[datos$svi == "0", ]
datos.svi0.gleason7 <- datos.svi0[datos.svi0$gleason == "7", ]
```

```
# Hacemos el porcentaje
porcentaje <- nrow(datos.svi0.gleason7) / nrow(datos.svi0) * 100
porcentaje
```

```
## [1] 48.68421
```

Vemos que el porcentaje es del 48.68421%.

- Estas dos variables, ¿son independientes?

```
# Hacemos un attach al dataset
attach(datos)
```

```
# Creamos una tabla con las dos variables
tabla <- table(svi, gleason)
```

```
# Creamos tablas de probabilidad por fila y por columna
addmargins(prop.table(tabla, 1), 2) * 100
```

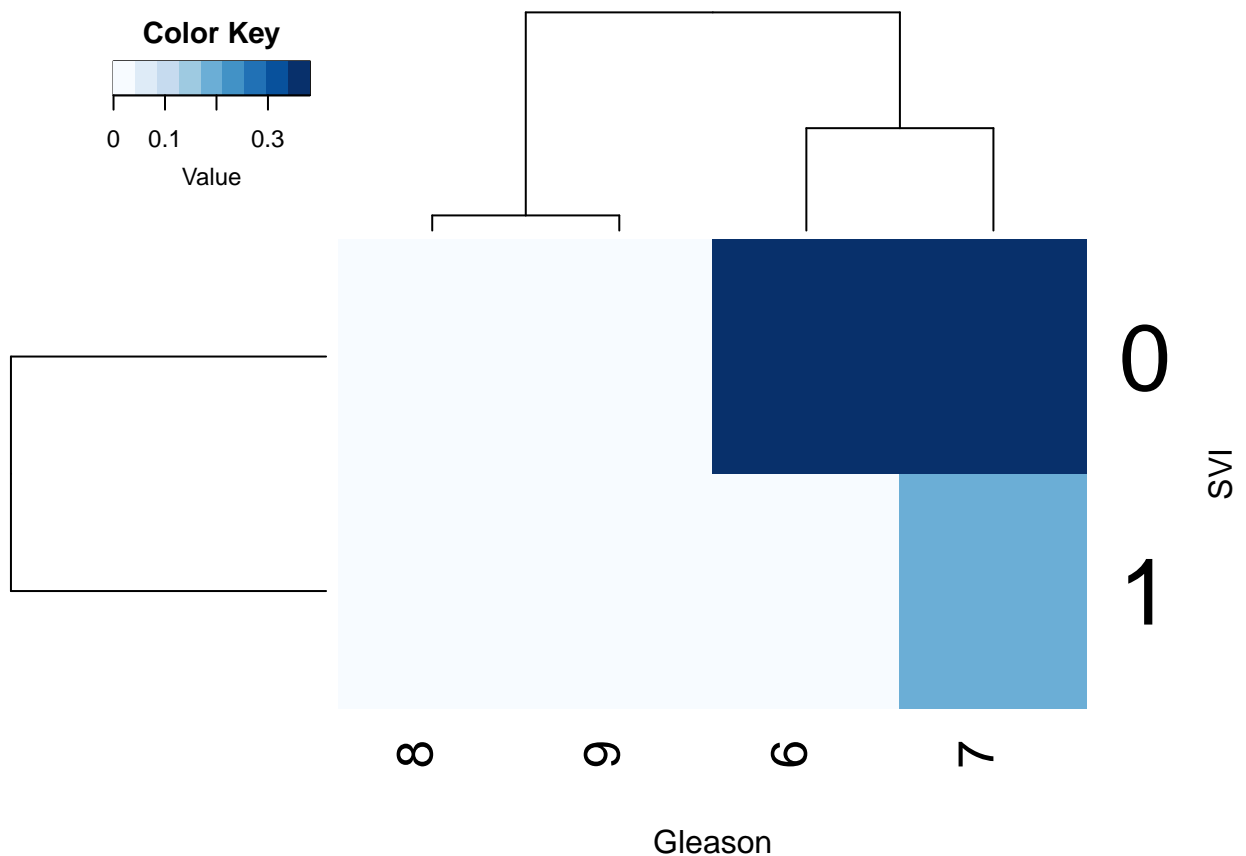
```
##      gleason
## svi      6      7      8      9      Sum
## 0  46.052632 48.684211  1.315789  3.947368 100.000000
## 1   0.000000 90.476190  0.000000  9.523810 100.000000
```

```
addmargins(prop.table(tabla, 2), 1) * 100
```

```
##      gleason
## svi      6      7      8      9
## 0   100.00000 66.07143 100.00000 60.00000
## 1    0.00000 33.92857  0.00000 40.00000
## Sum 100.00000 100.00000 100.00000 100.00000
```

```
# Realizamos un gráfico de la tabla para visualizar mejor la Independencia
```

```
heatmap.2(
  prop.table(tabla),
  xlab = "Gleason", ylab = "SVI",
  density.info = "none",
  col = blues9,
  trace = "none",
)
```



Se puede ver en la gráfica que la mayoría de los casos se acumulan en zonas concretas: - Cuando SVI es 0, se acumulan en Gleason = 6 y 7. - Cuando SVI es 1, se acumulan en Gleason = 7. Por lo tanto, como los datos no se distribuyen por igual en todos los casos, las dos variables son dependientes.

4. Regresión lineal simple

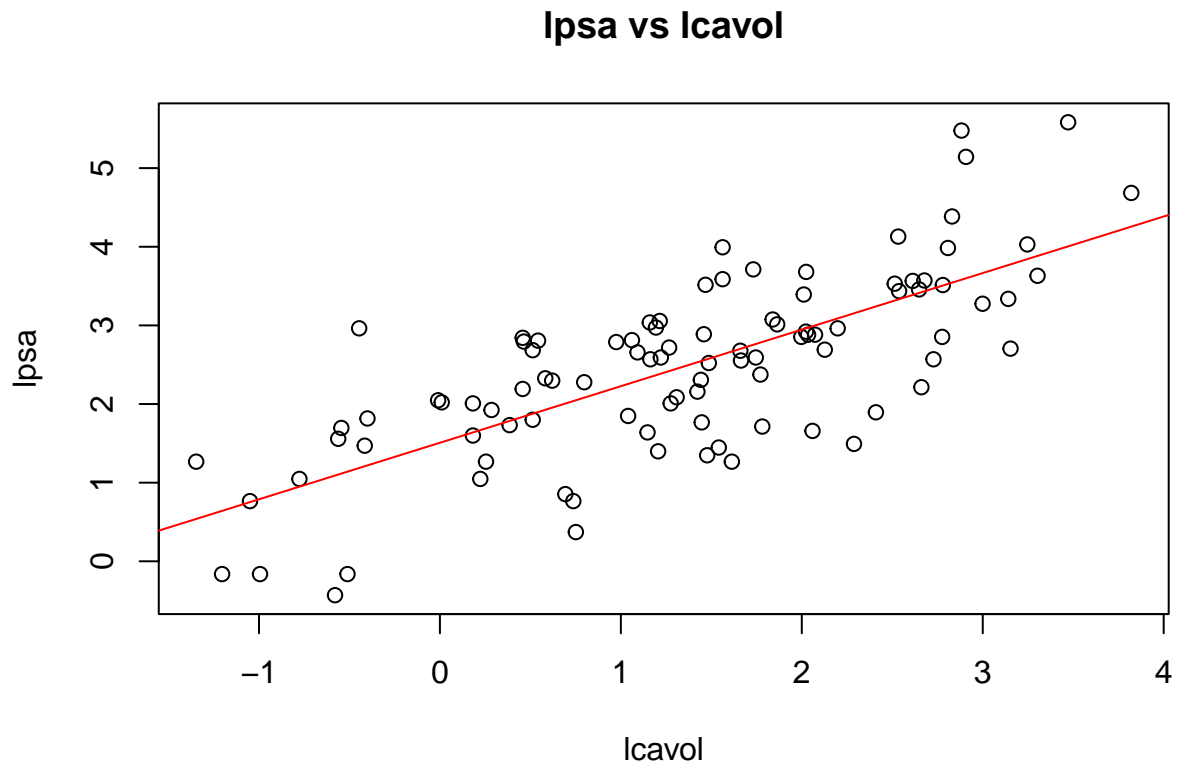
```
# Realizamos el modelo lineal
recta <- lm(lpsa ~ lcavol)

# Vemos el modelo
recta.summary <- summary(recta)
recta.summary
```

```
##
## Call:
## lm(formula = lpsa ~ lcavol)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.67624 -0.41648  0.09859  0.50709  1.89672
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.50730    0.12194   12.36  <2e-16 ***
## lcavol       0.71932    0.06819   10.55  <2e-16 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7875 on 95 degrees of freedom
## Multiple R-squared:  0.5394, Adjusted R-squared:  0.5346
## F-statistic: 111.3 on 1 and 95 DF,  p-value: < 2.2e-16
```

```
# Representamos el modelo sobre los datos
plot(lcavol, lpsa, main = "lpsa vs lcavol")
abline(recta, col = "red")
```



```
# Realizamos el intervalo de confianza al 95%
intervals <- confint(recta, level = 0.95)
intervals
```

```
##              2.5 %      97.5 %
## (Intercept) 1.2652222 1.7493727
## lcavol      0.5839404 0.8547004
```

```
# Calculamos el porcentaje de variación relativo de los intervalos de confianza respecto al valor predi
abs(intervals[1, 1] - intervals[1, 2]) / recta.summary$coefficients[1, 1] * 100
```

```
## [1] 32.12044
```

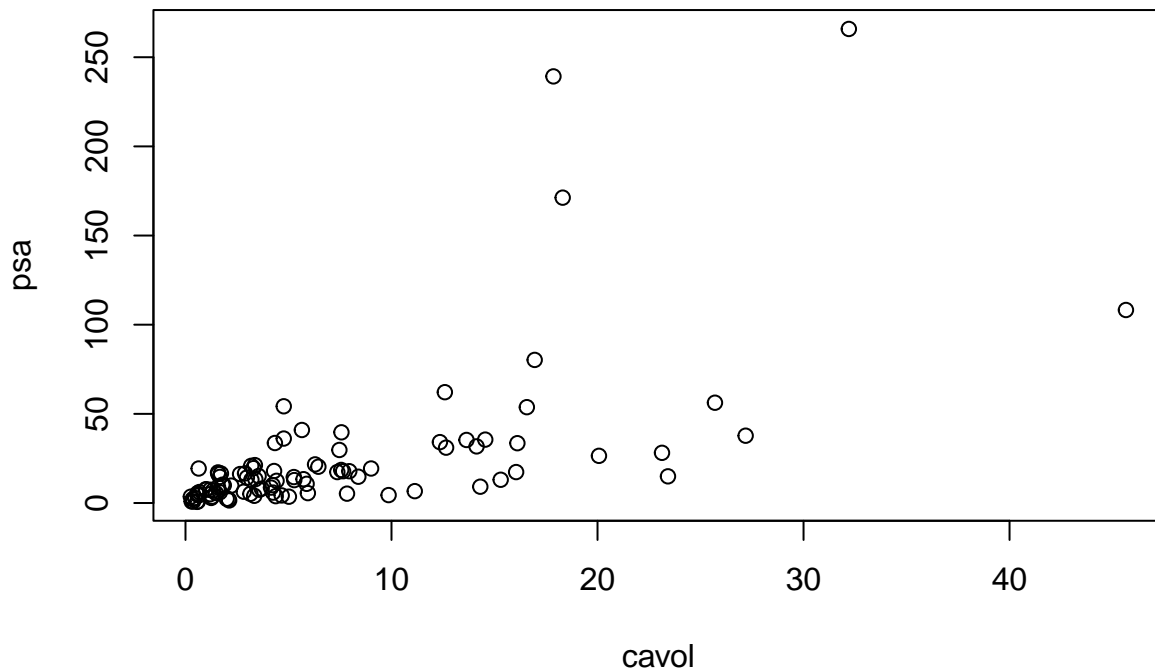
```
abs(intervals[2, 1] - intervals[2, 2]) / recta.summary$coefficients[2, 1] * 100
```

```
## [1] 37.64109
```

```
# Calculamos la suma de cuadrados de los residuos (RSE)
r1 <- residuals(recta)
RSE <- sqrt(sum(r1^2) / (dim(datos)[1] - 2))
RSE / mean(lpsa) * 100
```

```
## [1] 31.77468
```

```
cavol <- exp(lcavol)
psa <- exp(lpsa)
plot(cavol, psa)
```



Vamos a analizar el modelo lineal: - El t-value es bastante alto (12.36 y 10.55), por lo que los coeficientes están bastante alejados de ser nulos. - El p-value es bastante bajo ($<2e-16$), lo que refuerza que los coeficientes no son nulos. - El coeficiente R^2 no es muy alto, por lo que quizás el modelo lineal no sea el mejor modelo al que los datos se ajusten. Como el valor es 0.5394, el modelo explica el 53.94% de la variabilidad de lpsa respecto a lcavol. - El RSE es de 0.7875. Si lo dividimos entre la media de lpsa, vemos que tendríamos un error del 31.77%, lo que indica que el modelo no es muy bueno. - La longitud de los intervalos de confianza de las variables representan un 32.1% y 37.6% respecto a los valores estimados de los coeficientes. Esto es una variabilidad importante.

Está claro que las variables lpsa y lcavol están relacionadas. Sin embargo, aunque el p-value del ajuste lineal sea bajo, otros factores como el R^2 , los residuos y los intervalos de confianza nos indican que los datos están muy dispersos respecto al modelo. En el caso de quedarnos con el modelo, podemos pasar a un modelo exponencial, ya que lcavol y lpsa son logaritmos:

$$\ln(lpsa) = \beta_0 + \beta_1 \ln(lcavol) \Rightarrow psa = e^{\beta_0 + \beta_1 \ln(lcavol)} = \tilde{\beta}_0 cavol^{\beta_1}$$

5

```
library(corrplot)
```

```
## corrplot 0.95 loaded
```

```
datos
```

```
##          lcavol  lweight age          lbph svi          lcp gleason pgg45
## 1 -0.579818495  2.769459  50 -1.38629436  0 -1.38629436      6      0
## 2 -0.994252273  3.319626  58 -1.38629436  0 -1.38629436      6      0
## 3 -0.510825624  2.691243  74 -1.38629436  0 -1.38629436      7     20
## 4 -1.203972804  3.282789  58 -1.38629436  0 -1.38629436      6      0
```

| | | | | | | | | |
|-------|--------------|----------|----|-------------|---|-------------|---|-----|
| ## 5 | 0.751416089 | 3.432373 | 62 | -1.38629436 | 0 | -1.38629436 | 6 | 0 |
| ## 6 | -1.049822124 | 3.228826 | 50 | -1.38629436 | 0 | -1.38629436 | 6 | 0 |
| ## 7 | 0.737164066 | 3.473518 | 64 | 0.61518564 | 0 | -1.38629436 | 6 | 0 |
| ## 8 | 0.693147181 | 3.539509 | 58 | 1.53686722 | 0 | -1.38629436 | 6 | 0 |
| ## 9 | -0.776528789 | 3.539509 | 47 | -1.38629436 | 0 | -1.38629436 | 6 | 0 |
| ## 10 | 0.223143551 | 3.244544 | 63 | -1.38629436 | 0 | -1.38629436 | 6 | 0 |
| ## 11 | 0.254642218 | 3.604138 | 65 | -1.38629436 | 0 | -1.38629436 | 6 | 0 |
| ## 12 | -1.347073648 | 3.598681 | 63 | 1.26694760 | 0 | -1.38629436 | 6 | 0 |
| ## 13 | 1.613429934 | 3.022861 | 63 | -1.38629436 | 0 | -0.59783700 | 7 | 30 |
| ## 14 | 1.477048724 | 2.998229 | 67 | -1.38629436 | 0 | -1.38629436 | 7 | 5 |
| ## 15 | 1.205970807 | 3.442019 | 57 | -1.38629436 | 0 | -0.43078292 | 7 | 5 |
| ## 16 | 1.541159072 | 3.061052 | 66 | -1.38629436 | 0 | -1.38629436 | 6 | 0 |
| ## 17 | -0.415515444 | 3.516013 | 70 | 1.24415459 | 0 | -0.59783700 | 7 | 30 |
| ## 18 | 2.288486169 | 3.649359 | 66 | -1.38629436 | 0 | 0.37156356 | 6 | 0 |
| ## 19 | -0.562118918 | 3.267666 | 41 | -1.38629436 | 0 | -1.38629436 | 6 | 0 |
| ## 20 | 0.182321557 | 3.825375 | 70 | 1.65822808 | 0 | -1.38629436 | 6 | 0 |
| ## 21 | 1.147402453 | 3.419365 | 59 | -1.38629436 | 0 | -1.38629436 | 6 | 0 |
| ## 22 | 2.059238834 | 3.501043 | 60 | 1.47476301 | 0 | 1.34807315 | 7 | 20 |
| ## 23 | -0.544727175 | 3.375880 | 59 | -0.79850770 | 0 | -1.38629436 | 6 | 0 |
| ## 24 | 1.781709133 | 3.451574 | 63 | 0.43825493 | 0 | 1.17865500 | 7 | 60 |
| ## 25 | 0.385262401 | 3.667400 | 69 | 1.59938758 | 0 | -1.38629436 | 6 | 0 |
| ## 26 | 1.446918983 | 3.124565 | 68 | 0.30010459 | 0 | -1.38629436 | 6 | 0 |
| ## 27 | 0.512823626 | 3.719651 | 65 | -1.38629436 | 0 | -0.79850770 | 7 | 70 |
| ## 28 | -0.400477567 | 3.865979 | 67 | 1.81645208 | 0 | -1.38629436 | 7 | 20 |
| ## 29 | 1.040276712 | 3.128951 | 67 | 0.22314355 | 0 | 0.04879016 | 7 | 80 |
| ## 30 | 2.409644165 | 3.375880 | 65 | -1.38629436 | 0 | 1.61938824 | 6 | 0 |
| ## 31 | 0.285178942 | 4.090169 | 65 | 1.96290773 | 0 | -0.79850770 | 6 | 0 |
| ## 32 | 0.182321557 | 3.804438 | 65 | 1.70474809 | 0 | -1.38629436 | 6 | 0 |
| ## 33 | 1.275362800 | 3.037354 | 71 | 1.26694760 | 0 | -1.38629436 | 6 | 0 |
| ## 34 | 0.009950331 | 3.267666 | 54 | -1.38629436 | 0 | -1.38629436 | 6 | 0 |
| ## 35 | -0.010050336 | 3.216874 | 63 | -1.38629436 | 0 | -0.79850770 | 6 | 0 |
| ## 36 | 1.308332820 | 4.119850 | 64 | 2.17133681 | 0 | -1.38629436 | 7 | 5 |
| ## 37 | 1.423108334 | 3.657131 | 73 | -0.57981850 | 0 | 1.65822808 | 8 | 15 |
| ## 38 | 0.457424847 | 2.374906 | 64 | -1.38629436 | 0 | -1.38629436 | 7 | 15 |
| ## 39 | 2.660958594 | 4.085136 | 68 | 1.37371558 | 1 | 1.83258146 | 7 | 35 |
| ## 40 | 0.797507196 | 3.013081 | 56 | 0.93609336 | 0 | -0.16251893 | 7 | 5 |
| ## 41 | 0.620576488 | 3.141995 | 60 | -1.38629436 | 0 | -1.38629436 | 9 | 80 |
| ## 42 | 1.442201993 | 3.682610 | 68 | -1.38629436 | 0 | -1.38629436 | 7 | 10 |
| ## 43 | 0.582215620 | 3.865979 | 62 | 1.71379793 | 0 | -0.43078292 | 6 | 0 |
| ## 44 | 1.771556762 | 3.896909 | 61 | -1.38629436 | 0 | 0.81093022 | 7 | 6 |
| ## 45 | 1.486139696 | 3.409496 | 66 | 1.74919985 | 0 | -0.43078292 | 7 | 20 |
| ## 46 | 1.663926098 | 3.392829 | 61 | 0.61518564 | 0 | -1.38629436 | 7 | 15 |
| ## 47 | 2.727852828 | 3.995445 | 79 | 1.87946505 | 1 | 2.65675691 | 9 | 100 |
| ## 48 | 1.163150810 | 4.035125 | 68 | 1.71379793 | 0 | -0.43078292 | 7 | 40 |
| ## 49 | 1.745715531 | 3.498022 | 43 | -1.38629436 | 0 | -1.38629436 | 6 | 0 |
| ## 50 | 1.220829921 | 3.568123 | 70 | 1.37371558 | 0 | -0.79850770 | 6 | 0 |
| ## 51 | 1.091923301 | 3.993603 | 68 | -1.38629436 | 0 | -1.38629436 | 7 | 50 |
| ## 52 | 1.660131027 | 4.234831 | 64 | 2.07317193 | 0 | -1.38629436 | 6 | 0 |
| ## 53 | 0.512823626 | 3.633631 | 64 | 1.49290410 | 0 | 0.04879016 | 7 | 70 |
| ## 54 | 2.127040520 | 4.121473 | 68 | 1.76644166 | 0 | 1.44691898 | 7 | 40 |
| ## 55 | 3.153590358 | 3.516013 | 59 | -1.38629436 | 0 | -1.38629436 | 7 | 5 |
| ## 56 | 1.266947603 | 4.280132 | 66 | 2.12226154 | 0 | -1.38629436 | 7 | 15 |
| ## 57 | 0.974559640 | 2.865054 | 47 | -1.38629436 | 0 | 0.50077529 | 7 | 4 |
| ## 58 | 0.463734016 | 3.764682 | 49 | 1.42310833 | 0 | -1.38629436 | 6 | 0 |

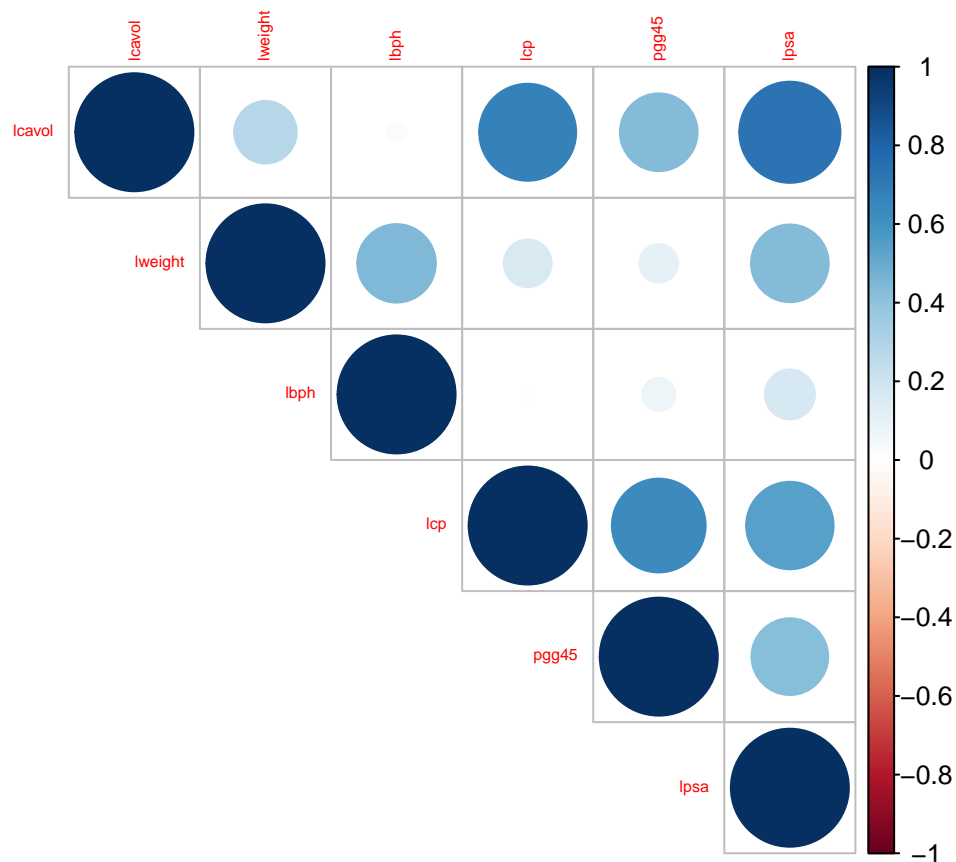
| | | | | | | | | |
|-------|--------------|----------|----|-------------|---|-------------|---|----|
| ## 59 | 0.542324291 | 4.178226 | 70 | 0.43825493 | 0 | -1.38629436 | 7 | 20 |
| ## 60 | 1.061256502 | 3.851211 | 61 | 1.29472717 | 0 | -1.38629436 | 7 | 40 |
| ## 61 | 0.457424847 | 4.524502 | 73 | 2.32630162 | 0 | -1.38629436 | 6 | 0 |
| ## 62 | 1.997417706 | 3.719651 | 63 | 1.61938824 | 1 | 1.90954250 | 7 | 40 |
| ## 63 | 2.775708850 | 3.524889 | 72 | -1.38629436 | 0 | 1.55814462 | 9 | 95 |
| ## 64 | 2.034705648 | 3.917011 | 66 | 2.00821403 | 1 | 2.11021320 | 7 | 60 |
| ## 65 | 2.073171929 | 3.623007 | 64 | -1.38629436 | 0 | -1.38629436 | 6 | 0 |
| ## 66 | 1.458615023 | 3.836221 | 61 | 1.32175584 | 0 | -0.43078292 | 7 | 20 |
| ## 67 | 2.022871190 | 3.878466 | 68 | 1.78339122 | 0 | 1.32175584 | 7 | 70 |
| ## 68 | 2.198335072 | 4.050915 | 72 | 2.30757263 | 0 | -0.43078292 | 7 | 10 |
| ## 69 | -0.446287103 | 4.408547 | 69 | -1.38629436 | 0 | -1.38629436 | 6 | 0 |
| ## 70 | 1.193922468 | 4.780383 | 72 | 2.32630162 | 0 | -0.79850770 | 7 | 5 |
| ## 71 | 1.864080131 | 3.593194 | 60 | -1.38629436 | 1 | 1.32175584 | 7 | 60 |
| ## 72 | 1.160020917 | 3.341093 | 77 | 1.74919985 | 0 | -1.38629436 | 7 | 25 |
| ## 73 | 1.214912744 | 3.825375 | 69 | -1.38629436 | 1 | 0.22314355 | 7 | 20 |
| ## 74 | 1.838961071 | 3.236716 | 60 | 0.43825493 | 1 | 1.17865500 | 9 | 90 |
| ## 75 | 2.999226163 | 3.849083 | 69 | -1.38629436 | 1 | 1.90954250 | 7 | 20 |
| ## 76 | 3.141130476 | 3.263849 | 68 | -0.05129329 | 1 | 2.42036813 | 7 | 50 |
| ## 77 | 2.010894999 | 4.433789 | 72 | 2.12226154 | 0 | 0.50077529 | 7 | 60 |
| ## 78 | 2.537657215 | 4.354784 | 78 | 2.32630162 | 0 | -1.38629436 | 7 | 10 |
| ## 79 | 2.648300197 | 3.582129 | 69 | -1.38629436 | 1 | 2.58399755 | 7 | 70 |
| ## 80 | 2.779440197 | 3.823192 | 63 | -1.38629436 | 0 | 0.37156356 | 7 | 50 |
| ## 81 | 1.467874348 | 3.070376 | 66 | 0.55961579 | 0 | 0.22314355 | 7 | 40 |
| ## 82 | 2.513656063 | 3.473518 | 57 | 0.43825493 | 0 | 2.32727771 | 7 | 60 |
| ## 83 | 2.613006652 | 3.888754 | 77 | -0.52763274 | 1 | 0.55961579 | 7 | 30 |
| ## 84 | 2.677590994 | 3.838376 | 65 | 1.11514159 | 0 | 1.74919985 | 9 | 70 |
| ## 85 | 1.562346305 | 3.709907 | 60 | 1.69561561 | 0 | 0.81093022 | 7 | 30 |
| ## 86 | 3.302849259 | 3.518980 | 64 | -1.38629436 | 1 | 2.32727771 | 7 | 60 |
| ## 87 | 2.024193067 | 3.731699 | 58 | 1.63899671 | 0 | -1.38629436 | 6 | 0 |
| ## 88 | 1.731655545 | 3.369018 | 62 | -1.38629436 | 1 | 0.30010459 | 7 | 30 |
| ## 89 | 2.807593831 | 4.718052 | 65 | -1.38629436 | 1 | 2.46385324 | 7 | 60 |
| ## 90 | 1.562346305 | 3.695110 | 76 | 0.93609336 | 1 | 0.81093022 | 7 | 75 |
| ## 91 | 3.246490992 | 4.101817 | 68 | -1.38629436 | 0 | -1.38629436 | 6 | 0 |
| ## 92 | 2.532902848 | 3.677566 | 61 | 1.34807315 | 1 | -1.38629436 | 7 | 15 |
| ## 93 | 2.830267834 | 3.876396 | 68 | -1.38629436 | 1 | 1.32175584 | 7 | 60 |
| ## 94 | 3.821003607 | 3.896909 | 44 | -1.38629436 | 1 | 2.16905370 | 7 | 40 |
| ## 95 | 2.907447359 | 3.396185 | 52 | -1.38629436 | 1 | 2.46385324 | 7 | 10 |
| ## 96 | 2.882563575 | 3.773910 | 68 | 1.55814462 | 1 | 1.55814462 | 7 | 80 |
| ## 97 | 3.471966453 | 3.974998 | 68 | 0.43825493 | 1 | 2.90416508 | 7 | 20 |
| ## | lpsa | | | | | | | |
| ## 1 | -0.4307829 | | | | | | | |
| ## 2 | -0.1625189 | | | | | | | |
| ## 3 | -0.1625189 | | | | | | | |
| ## 4 | -0.1625189 | | | | | | | |
| ## 5 | 0.3715636 | | | | | | | |
| ## 6 | 0.7654678 | | | | | | | |
| ## 7 | 0.7654678 | | | | | | | |
| ## 8 | 0.8544153 | | | | | | | |
| ## 9 | 1.0473190 | | | | | | | |
| ## 10 | 1.0473190 | | | | | | | |
| ## 11 | 1.2669476 | | | | | | | |
| ## 12 | 1.2669476 | | | | | | | |
| ## 13 | 1.2669476 | | | | | | | |
| ## 14 | 1.3480731 | | | | | | | |

15 1.3987169
16 1.4469190
17 1.4701758
18 1.4929041
19 1.5581446
20 1.5993876
21 1.6389967
22 1.6582281
23 1.6956156
24 1.7137979
25 1.7316555
26 1.7664417
27 1.8000583
28 1.8164521
29 1.8484548
30 1.8946169
31 1.9242487
32 2.0082140
33 2.0082140
34 2.0215476
35 2.0476928
36 2.0856721
37 2.1575593
38 2.1916535
39 2.2137539
40 2.2772673
41 2.2975726
42 2.3075726
43 2.3272777
44 2.3749058
45 2.5217206
46 2.5533438
47 2.5687881
48 2.5687881
49 2.5915164
50 2.5915164
51 2.6567569
52 2.6775910
53 2.6844403
54 2.6912431
55 2.7047113
56 2.7180005
57 2.7880929
58 2.7942279
59 2.8063861
60 2.8124102
61 2.8419982
62 2.8535925
63 2.8535925
64 2.8820035
65 2.8820035
66 2.8875901
67 2.9204698
68 2.9626924

```
## 69 2.9626924
## 70 2.9729753
## 71 3.0130809
## 72 3.0373539
## 73 3.0563569
## 74 3.0750055
## 75 3.2752562
## 76 3.3375474
## 77 3.3928291
## 78 3.4355988
## 79 3.4578927
## 80 3.5130369
## 81 3.5160131
## 82 3.5307626
## 83 3.5652984
## 84 3.5709402
## 85 3.5876769
## 86 3.6309855
## 87 3.6800909
## 88 3.7123518
## 89 3.9843437
## 90 3.9936030
## 91 4.0298060
## 92 4.1295508
## 93 4.3851468
## 94 4.6844434
## 95 5.1431245
## 96 5.4775090
## 97 5.5829322
```

```
num_cols <- which(sapply(datos, is.numeric))

corrplot(cor(datos[,num_cols]),type="upper",tl.cex=0.5)
```



```

datos.num <- datos[,c(-3,-7,-8)]

rectaMul <- lm(lpsa ~ .,data=datos.num)

summary(rectaMul)

##
## Call:
## lm(formula = lpsa ~ ., data = datos.num)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.84878 -0.38372 -0.00413  0.45189  1.55468
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.36496    0.69866  -0.522  0.60269
## lcavol       0.54790    0.08637   6.343 8.56e-09 ***
## lweight      0.53036    0.19769   2.683  0.00867 **
## lbph         0.07999    0.05643   1.418  0.15971
## svi1         0.75975    0.24122   3.150  0.00221 **
## lcp          -0.03638    0.08088  -0.450  0.65391
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7071 on 91 degrees of freedom
## Multiple R-squared:  0.6443, Adjusted R-squared:  0.6248

```

F-statistic: 32.97 on 5 and 91 DF, p-value: < 2.2e-16

#6

```
# library(glmnet)

# x <- model.matrix(lpsa~.,datos.num)[-1] #variables numericas

# y <- datos.num$lpsa #variable de salida

# set.seed(1)

# train <- sample(seq(1, nrow(x)), nrow(x)/2) #filas de entrenamiento

# test <- (-train)

# y.test <- y[test]

# malla <- 10^seq(10,-2,length=100) #mallado de lambdas

# malla.ridge.train <- glmnet(x[train,],y[train],alpha=0,lambda=malla) # regresion ridge sin CV con con

# plot(malla.ridge.train,xvar="lambda" ) # plot de betas vs log(lambda)

# cv.out.train <- cv.glmnet(x[train,],y[train],alpha=0) # regresion ridge con CV con conjunto de entre

# bestlam.train<-cv.out.train$lambda.min #lambda que minmiza el MSE

# bestlam.train

# ridge.train<-glmnet(x[train,],y[train],alpha=0, lambda=bestlam.train) #regresion ridge con CV con mej

# coef(ridge.train)[,1]

# rectaMul.train <- glmnet(x[train,],y[train],alpha=0, lambda=0) # regresion multiple con datos de ent

# coef(rectaMul.train)[,1]

# #el intercept es mayor en ridge
# # beta de lcp mayor en ridge
# # los demas betas son menores en ridge

# plot(cv.out.train) #MSE vs log(lambda)

# ridge.pred <- predict(ridge.train,newx=x[test,]) # predicion de 'ridge.train' en conjunto de testeo

# mean((ridge.pred-y.test)^2) #MSE estimado de Ridge
```

```
# rectaMul.pred <- predict(rectaMul.train,newx=x[test,])  
# mean((rectaMul.pred-y.test)^2) #MSE estimado de rectaMul  
# # El MSE de la regresion multiple de Ridge es menor que la de rectaMul
```