

Trabajo Final Introducción a la Ciencia de Datos

Juanjo Sierra

27 de diciembre de 2018

Planteamiento

El trabajo final de la asignatura *Introducción a la Ciencia de Datos* se divide en dos secciones. Consiste en realizar un estudio sobre un conjunto de datos de regresión y otro sobre un conjunto de datos de clasificación. Se aplicarán distintas técnicas aprendidas durante la asignatura para conseguir los resultados adecuados.

Librerías y paquetes a cargar

```
library(ggplot2)
library(dplyr)

##
## Attaching package: 'dplyr'
##
## The following objects are masked from 'package:stats':
##
##   filter, lag
##
## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union
library(tidyr)
```

Trabajo Final Regresión

En primer lugar se realizará el estudio de la base de datos de regresión. En este caso el conjunto de datos a analizar es **Friedman**, que se ha descargado desde el repositorio de datasets de la asignatura. Se puede leer utilizando la siguiente orden:

```
friedman = read.csv("Datos/friedman/friedman.dat", header = FALSE, comment.char = "@")
head(friedman)
```

```
##           V1           V2           V3           V4           V5           V6
## 1 0.6964817 0.3584375 0.4258343 0.33031373 0.22249090 11.09496
## 2 0.5903899 0.4306749 0.8690418 0.07091161 0.63430253 13.22921
## 3 0.8276557 0.6178330 0.9494409 0.67013843 0.64080838 25.33973
## 4 0.8107169 0.2621162 0.4541944 0.85470608 0.27976951 15.18159
## 5 0.4068430 0.8161745 0.8611055 0.12890196 0.15747881 14.43310
## 6 0.6299940 0.3821170 0.9819543 0.98471273 0.07506318 20.97857
```

Dado que los nombres asignados a las variables no aportan ninguna información, y en el resumen del dataset en formato KEEL podemos comprobar que sus nombres tampoco son representativos, se procede a asignarles una notación genérica.

```
n = length(names(friedman))-1
names(friedman)[1:n] = paste ("X", 1:n, sep="")
names(friedman)[n+1] = "Y"
head(friedman)
```

```
##           X1           X2           X3           X4           X5           Y
## 1 0.6964817 0.3584375 0.4258343 0.33031373 0.22249090 11.09496
## 2 0.5903899 0.4306749 0.8690418 0.07091161 0.63430253 13.22921
## 3 0.8276557 0.6178330 0.9494409 0.67013843 0.64080838 25.33973
## 4 0.8107169 0.2621162 0.4541944 0.85470608 0.27976951 15.18159
## 5 0.4068430 0.8161745 0.8611055 0.12890196 0.15747881 14.43310
## 6 0.6299940 0.3821170 0.9819543 0.98471273 0.07506318 20.97857
```

Ahora podemos comprobar de forma más directa que existen 5 variables de entrada (X1-5) que determinan una única variable de salida (Y). Es interesante comprobar las dimensiones del dataset para poder asegurar que se está asumiendo lo correcto.

```
dim(friedman)
```

```
## [1] 1200    6
```

Con esto se puede confirmar que existen un total de 1200 ejemplos en el conjunto de datos, cada uno con 6 variables (5 de entrada y 1 de salida).

Utilizando la función `summary` se puede obtener una visión más completa del dataset, arrojando nuevos valores interesantes para su estudio como los rangos de las variables, sus cuartiles o su media y mediana.

```
summary(friedman)
```

```
##           X1           X2           X3
## Min.      :0.001212   Min.      :0.0001603   Min.      :0.0006546
## 1st Qu.:0.249184     1st Qu.:0.2423287   1st Qu.:0.2485096
## Median :0.519293     Median :0.4932687   Median :0.4993111
## Mean    :0.506193     Mean    :0.4999592   Mean    :0.4995141
## 3rd Qu.:0.751131     3rd Qu.:0.7655960   3rd Qu.:0.7441912
## Max.    :0.999719     Max.    :0.9996775   Max.    :0.9990619
##           X4           X5           Y
## Min.      :0.0002123   Min.      :0.0004299   Min.      : 0.664
## 1st Qu.:0.2703118     1st Qu.:0.2578755     1st Qu.:10.859
## Median :0.5328840     Median :0.4753492     Median :14.654
## Mean    :0.5122272     Mean    :0.4928214     Mean    :14.567
## 3rd Qu.:0.7566648     3rd Qu.:0.7385440     3rd Qu.:18.494
## Max.    :0.9994802     Max.    :0.9995394     Max.    :28.590
```

Se puede comprobar también si existen valores perdidos en el dataset. Para esto se puede utilizar la función `anyNA`:

```
anyNA(friedman)
```

```
## [1] FALSE
```

Este resultado indica que no hay valores perdidos y que por lo tanto no es necesario imputar ni tomar ninguna decisión para restablecer dichos valores.

A continuación se puede comprobar si existen ejemplos duplicados, y eliminarlos del dataset. Para ello se utiliza la función `duplicated` acompañado de la función `any`:

```
any(duplicated(friedman))
```

```
## [1] FALSE
```

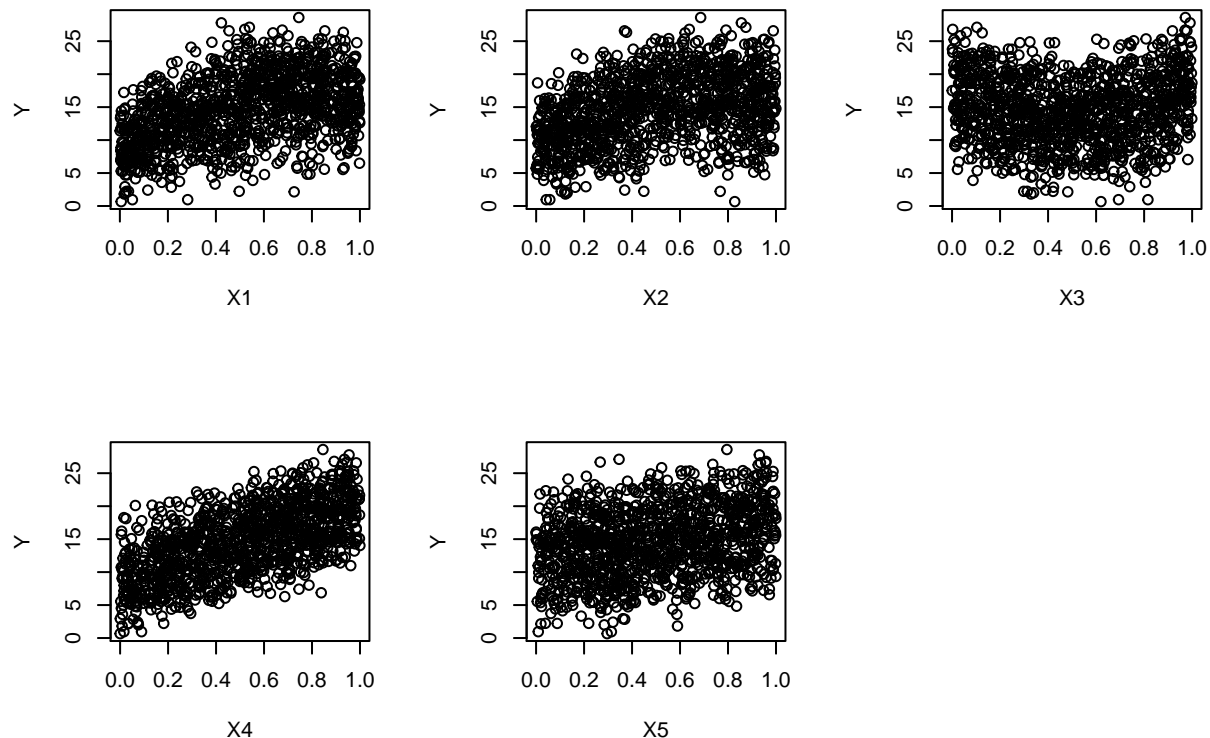
Como no hay duplicados se puede continuar con el estudio sin realizar ninguna alteración en el conjunto de datos.

Además, como el rango de las variables está entre 0 y 1 (como se pudo comprobar anteriormente con el `summary`), no es necesario realizar un escalado ni una transformación en los valores. En este punto se puede afirmar que los datos están listos para poder trabajar con ellos.

Como primer paso para el análisis del dataset se puede mostrar cada una de las variables de entrada con respecto a la variable de salida. Esto permitirá averiguar de un vistazo cuál tiene más potencial de determinar qué valor de salida obtendrá dicho ejemplo.

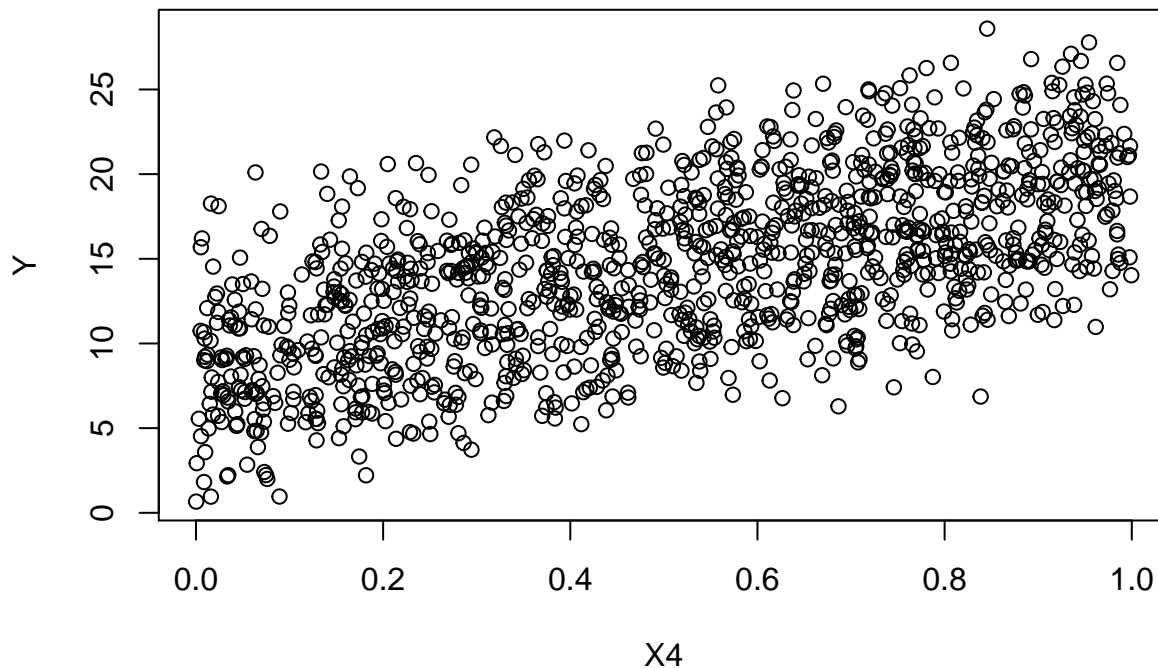
```
plotY = function (x,y) {
  plot(friedman[,y]~friedman[,x], xlab=names(friedman)[x], ylab=names(friedman)[y])
}

par(mfrow=c(2,3))
x = sapply(1:(dim(friedman)[2]-1), plotY, dim(friedman)[2])
par(mfrow=c(1,1))
```



Basándose en un modelo de regresión lineal, se puede especular que la variable X4 parece tener una correlación más alta con la variable de salida Y, y por tanto podría resultar en un mejor modelo. A continuación se muestra la gráfica de X4 frente a Y más grande para poder apreciar mejor la posible correlación.

```
plotY(4,dim(friedman)[2])
```



La correlación de las variables entre sí y con la variable de salida puede obtenerse de forma directa gracias a la función `cor`:

```
cor(friedman)
```

```
##           X1           X2           X3           X4           X5           Y
## X1  1.000000000 -0.03227730  0.009162253  0.09172183  0.01124122  0.4334883
## X2 -0.032277302  1.00000000  0.010233226  0.03639529  0.02452585  0.3713814
## X3  0.009162253  0.01023323  1.000000000  0.03883400  0.02110537  0.0356199
## X4  0.091721829  0.03639529  0.038834002  1.00000000 -0.02445055  0.6157779
## X5  0.011241216  0.02452585  0.021105366 -0.02445055  1.00000000  0.2757470
## Y   0.433488308  0.37138140  0.035619901  0.61577794  0.27574703  1.0000000
```

Como se había supuesto anteriormente en función de las gráficas obtenidas, es la variable X4 la que más correlación tiene con la variable de salida Y (~0.616).

El primer objetivo del trabajo final de regresión es generar un modelo lineal con cada una de las variables de entrada del dataset. De esta forma se puede obtener de una manera sencilla la información sobre qué variable es mejor para un modelo lineal, es decir, qué variable es más representativa de la de salida.

Para realizar los modelos de regresión lineal se va a utilizar la función `lm` que ya viene entre las funciones base de R. Es necesario indicar cuál es la variable de salida y cuál (o cuáles) son las que se van a utilizar para construir el modelo.

Se van a analizar todas las variables X con respecto a la variable de salida Y. Se construye un modelo con cada una de estas variables, y con la función `summary` se obtiene una información más detallada del modelo resultante.

```
lmsimple1 = lm(Y~X1, data=friedman)
lmsimple2 = lm(Y~X2, data=friedman)
lmsimple3 = lm(Y~X3, data=friedman)
lmsimple4 = lm(Y~X4, data=friedman)
lmsimple5 = lm(Y~X5, data=friedman)
```

```
summary(lmsimple1)
```

```
##
## Call:
## lm(formula = Y ~ X1, data = friedman)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.1161  -3.3974   0.0156   3.3261  13.8565
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  10.6586     0.2708   39.37  <2e-16 ***
## X1           7.7211     0.4637   16.65  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.674 on 1198 degrees of freedom
## Multiple R-squared:  0.1879, Adjusted R-squared:  0.1872
## F-statistic: 277.2 on 1 and 1198 DF,  p-value: < 2.2e-16
```

```
summary(lmsimple2)
```

```
##
## Call:
## lm(formula = Y ~ X2, data = friedman)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -16.0532  -3.3110  -0.1019   3.5042  12.8661
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  11.2915     0.2744   41.15  <2e-16 ***
## X2           6.5515     0.4732   13.84  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.816 on 1198 degrees of freedom
## Multiple R-squared:  0.1379, Adjusted R-squared:  0.1372
## F-statistic: 191.7 on 1 and 1198 DF,  p-value: < 2.2e-16
```

```
summary(lmsimple3)
```

```
##
## Call:
## lm(formula = Y ~ X3, data = friedman)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.9798  -3.6750   0.1493   3.8077  13.7227
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  14.2490     0.2981   47.805  <2e-16 ***
## X3           0.6367     0.5161   1.234   0.218
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.183 on 1198 degrees of freedom
## Multiple R-squared:  0.001269,    Adjusted R-squared:  0.0004351
## F-statistic: 1.522 on 1 and 1198 DF,  p-value: 0.2176
```

```
summary(lmsimple4)
```

```
##
## Call:
## lm(formula = Y ~ X4, data = friedman)
##
## Residuals:
```

	Min	1Q	Median	3Q	Max
	-11.3664	-3.1954	-0.0698	3.0166	10.5726

```
##
## Coefficients:
```

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	8.8149	0.2432	36.25	<2e-16 ***
X4	11.2296	0.4151	27.05	<2e-16 ***

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.087 on 1198 degrees of freedom
## Multiple R-squared:  0.3792, Adjusted R-squared:  0.3787
## F-statistic: 731.7 on 1 and 1198 DF,  p-value: < 2.2e-16
```

```
summary(lmsimple5)
```

```
##
## Call:
## lm(formula = Y ~ X5, data = friedman)
##
## Residuals:
```

	Min	1Q	Median	3Q	Max
	-13.2433	-3.8083	0.1361	3.6498	13.2920

```
##
## Coefficients:
```

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	12.0471	0.2918	41.292	<2e-16 ***
X5	5.1132	0.5150	9.929	<2e-16 ***

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.986 on 1198 degrees of freedom
## Multiple R-squared:  0.07604,    Adjusted R-squared:  0.07527
## F-statistic: 98.59 on 1 and 1198 DF,  p-value: < 2.2e-16
```

De los resultados anteriores se pueden extraer varias afirmaciones. En primer lugar, el p-value de los modelos de X1, X2, X4 y X5 es muy pequeño ($< 2.2e-16$) por lo que se puede afirmar con una confianza casi cercana al 100% que existe algún tipo de dependencia lineal entre dichas variables y la variable de salida. En el caso de la variable X3, sin embargo, el p-value es muy alto (> 0.2) por lo que no se puede afirmar lo anterior con suficiente confianza, es decir, es una variable que no se utilizará generalmente para construir un modelo de regresión lineal.

De entre los modelos aceptables, como era de esperar el mejor es el que utiliza X4, la variable que más

correlación mantiene con la salida, a pesar de ser “tan sólo” un valor de R-cuadrado de 0.379. El R-squared o R-cuadrado indica cómo de bueno es el modelo de regresión lineal. Cuanto más próximo a 1 más acertado es, y cuanto más próximo a 0 al contrario. Es por esto que a pesar de que el valor de X_4 no es muy óptimo, es el que más cerca se encuentra del 1, y por tanto el mejor de las variables estudiadas.