Predicción del Salario para los Jugadores de la NBA

Jose Javier Martíb García

8/10/2019

```
nba<-read.csv("C:/Users/Equipo/Desktop/CUNEF/Prediccion/Datos/nba.csv")
#Lo primero he cargado las librerias que iba a necesitar

library(MASS)
library(leaps)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following object is masked from 'package:MASS':
##
##     select
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------- tidyverse 1.2.1 --
```

```
## v ggplot2 3.2.1     v readr   1.3.1
## v tibble  2.1.3     v purrr   0.3.2
## v tidyr   1.0.0     v stringr 1.4.0
## v ggplot2 3.2.1     v forcats 0.4.0
```

```
## -- Conflicts ------------------------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## x dplyr::select() masks MASS::select()
```

```
library(fBasics)
```

```
## Loading required package: timeDate
```

```
## Loading required package: timeSeries
```

```r
library(ISLR)
library(car)
```

```
## Loading required package: carData
```

```
##
## Attaching package: 'car'
```

```
## The following object is masked from 'package:fBasics':
##
##     densityPlot
```

```
## The following object is masked from 'package:purrr':
##
##     some
```

```
## The following object is masked from 'package:dplyr':
##
##     recode
```

```r
library(gvlma)

#La base de datos contiene un total de 485 registros agrupados en 28 variables.
#Se va a tratar de dar explicacion a la relacion existente entre el salario de
#cada jugador con sus estadisticas individuales y ase obtener un modelo de prediccion
#tomando como variable dependiente el salario y como variables independientes dichas
#estadisticas. Sabemos que existen dos NAs

names(nba)
```

```
##  [1] "Player"          "Salary"          "NBA_Country"
##  [4] "NBA_DraftNumber" "Age"             "Tm"
##  [7] "G"               "MP"              "PER"
## [10] "TS."             "X3PAr"           "FTr"
## [13] "ORB."            "DRB."            "TRB."
## [16] "AST."            "STL."            "BLK."
## [19] "TOV."            "USG."            "OWS"
## [22] "DWS"             "WS"              "WS.48"
## [25] "OBPM"            "DBPM"            "BPM"
## [28] "VORP"
```

```r
nba<-na.omit(nba)

#Regresion con todas las variables.

Model0 <- lm(Salary ~ Age + AST. + BLK. + BPM + DBPM + DRB. + DWS + FTr + G + MP +
               NBA_DraftNumber + OBPM + ORB. + OWS + PER + Tm + STL. + TOV. + TRB. + TS. + USG. +
               VORP + WS + WS.48 + X3PAr, data = nba)


#Esta regresiÃ³n unicamente la he hecho para comprobar como explican las variables independientes
```

```r
#a la variable salario y a la vista de los resultados obtenidos voy a descartar dicho modelo.

#Steep forward, esto lo he hecho descartando variables las cuales he considerado que no eran
#importantes.

regfit.fwd<-regsubsets(Salary~.-Player -NBA_Country -Tm, nba ,method ="forward")

resumen<-summary (regfit.fwd)

#Aqui he hecho un analisis del BIC del Steep forward y de los 8 que me han quedado he elegido
#los tres que menor BIC tienen.

resumen$bic
```

```
## [1] -195.4434 -247.4542 -277.9638 -287.7754 -290.6157 -323.9518 -323.0798
## [8] -319.4324
```

```r
#En este momento lo que hice fue hacer otra regresion con las variables que estaban presentes
#en los tres modelos con menor BIC.

modelo <-lm(formula = Salary ~ Age + NBA_DraftNumber + G + MP + USG.+ DBPM+ STL. + WS, data = nba)

summary(modelo)
```

```
##
## Call:
## lm(formula = Salary ~ Age + NBA_DraftNumber + G + MP + USG. +
##     DBPM + STL. + WS, data = nba)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -14435369  -2995507   -404761   2273171  23099343
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -6674087.0  1983563.9  -3.365 0.000828 ***
## Age               521471.4    54982.0   9.484  < 2e-16 ***
## NBA_DraftNumber   -64760.4    12123.6  -5.342 1.43e-07 ***
## G                -160334.7    21668.7  -7.399 6.28e-13 ***
## MP                  5394.4      799.6   6.746 4.43e-11 ***
## USG.              112966.7    51273.1   2.203 0.028058 *
## DBPM              347163.4   143323.3   2.422 0.015799 *
## STL.             -461165.2   276718.8  -1.667 0.096265 .
## WS                850413.1   153783.9   5.530 5.30e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5069000 on 474 degrees of freedom
## Multiple R-squared:  0.5384, Adjusted R-squared:  0.5306
## F-statistic: 69.11 on 8 and 474 DF,  p-value: < 2.2e-16
```

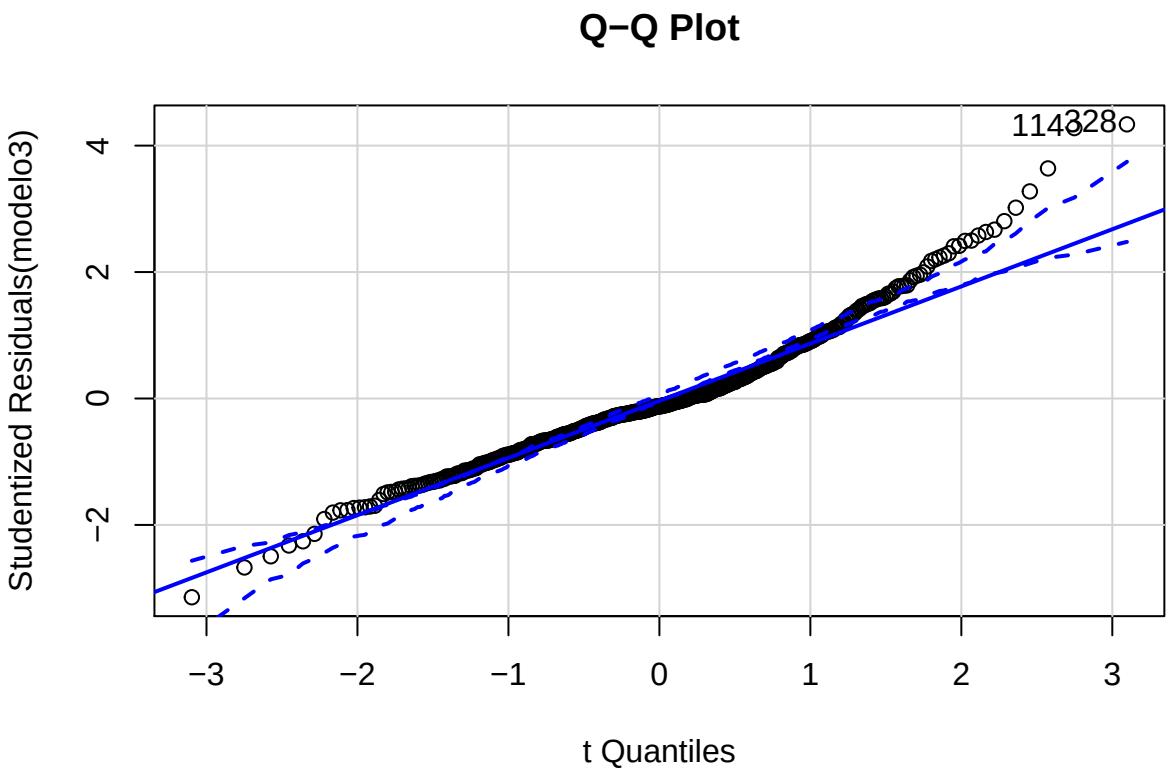
```r
#Este nuevo modelo lineal solo contiene las variables comunes a los 3 modelos que seleccione.
#Con este Steep forward de las variables que son comunes a los tres modelos he decidido quedarme
#solo con la variable edad, numero del draft y G y la variable WS, ya que son las tres que mas se repit

regfit.fwd2=regsubsets(Salary~.-Player -NBA_Country -ORB. -DRB.-Tm -PER -TS. -X3PAr -FTr -AST. -STL. -BL

modelo3<-lm(formula = Salary ~ Age + NBA_DraftNumber + G + WS, data = nba)
summary(modelo3)
```

```
##
## Call:
## lm(formula = Salary ~ Age + NBA_DraftNumber + G + WS, data = nba)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -16286628  -3225291   -653686   2585287  22591402
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -5383469    1640614  -3.281  0.00111 **
## Age               492994      57437   8.583  < 2e-16 ***
## NBA_DraftNumber   -82466      12373  -6.665 7.32e-11 ***
## G                 -48676      12739  -3.821  0.00015 ***
## WS               1609251     119737  13.440  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5350000 on 478 degrees of freedom
## Multiple R-squared:  0.4815, Adjusted R-squared:  0.4771
## F-statistic:   111 on 4 and 478 DF,  p-value: < 2.2e-16
```

```r
qqPlot(modelo3, labels = row.names(nba),id.method="identify", simulate = T, main = "Q-Q Plot")
```




```
## 114 328
## 112 326
```

```
residmodel3<-resid(modelo3)
jbTest(residmodel3)
```

```
## Warning in interpp.old(x, y, z, xo, yo, ncp = 0, extrap = FALSE, duplicate
## = duplicate, : interpp.old() is deprecated, future versions will only
## provide interpp()

## Warning in interpp.old(x, y, z, xo, yo, ncp = 0, extrap = FALSE, duplicate
## = duplicate, : interpp.old() is deprecated, future versions will only
## provide interpp()

##
## Title:
##  Jarque - Bera Normality Test
##
## Test Results:
##   PARAMETER:
##     Sample Size: 483
##   STATISTIC:
##     LM: 101.471
##     ALM: 104.642
##   P VALUE:
```

```
##     Asymptotic: < 2.2e-16
##
## Description:
##  Wed Oct 09 19:02:15 2019 by user: Equipo
```

```
#No puedo asegurar que los residuos se distribuyan de una manera normal por tanto, no puedo
#aceptar la hipotesis nula, ademas este p-valor es menor que el nivel de significacion por
#tanto rechazo la hipotesis de normalidad sobre la distribucion de dichos residuos.

shapiro.test(residmodel3)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residmodel3
## W = 0.96319, p-value = 1.225e-09
```

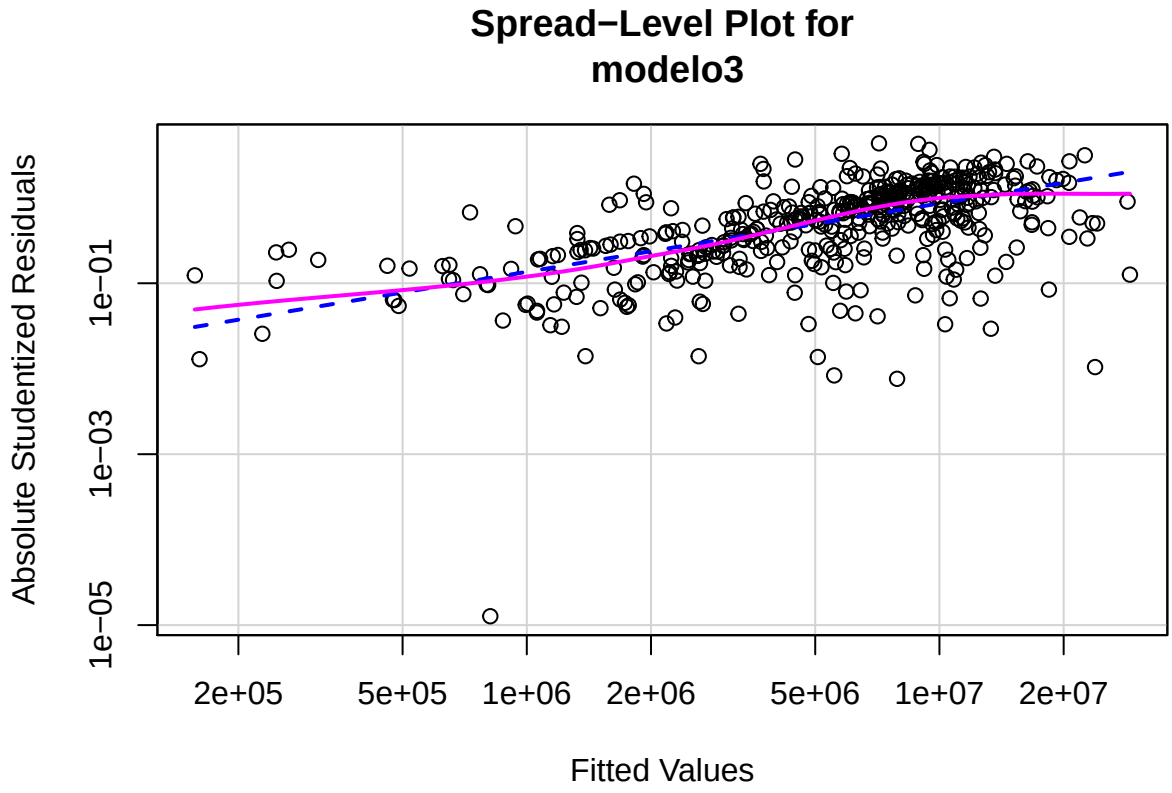
```
#Segun el Test de Shapiro - Wilk puedo asegurar que la muestra no sigue una distribucion
#normal.

ncvTest(modelo3)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 90.60138, Df = 1, p = < 2.22e-16
```

```
spreadLevelPlot(modelo3)
```

```
## Warning in spreadLevelPlot.lm(modelo3):
## 24 negative fitted values removed
```




```
##
## Suggested power transformation:  0.1990168
```

```
#Analizando el test de Breusch Pagan y el grafico puedo concluir diciendo que este modelo
#presenta un problema de heterocedasticidad.

gvmodelo3<-gvlma(modelo3)
summary(gvmodelo3)
```

```
##
## Call:
## lm(formula = Salary ~ Age + NBA_DraftNumber + G + WS, data = nba)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -16286628  -3225291   -653686   2585287  22591402
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -5383469    1640614  -3.281  0.00111 **
## Age               492994      57437   8.583  < 2e-16 ***
## NBA_DraftNumber   -82466      12373  -6.665 7.32e-11 ***
## G                 -48676      12739  -3.821  0.00015 ***
## WS               1609251     119737  13.440  < 2e-16 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5350000 on 478 degrees of freedom
## Multiple R-squared:  0.4815, Adjusted R-squared:  0.4771
## F-statistic:   111 on 4 and 478 DF,  p-value: < 2.2e-16
##
##
## ASSESSMENT OF THE LINEAR MODEL ASSUMPTIONS
## USING THE GLOBAL TEST ON 4 DEGREES-OF-FREEDOM:
## Level of Significance =  0.05
##
## Call:
##  gvlma(x = modelo3)
##
##                      Value   p-value                   Decision
## Global Stat        109.655 0.000e+00 Assumptions NOT satisfied!
## Skewness            46.321 1.004e-11 Assumptions NOT satisfied!
## Kurtosis            55.150 1.117e-13 Assumptions NOT satisfied!
## Link Function        6.652 9.904e-03 Assumptions NOT satisfied!
## Heteroscedasticity   1.532 2.159e-01    Assumptions acceptable.
```

```
#He utilizado diferentes formas de ver la presencia de heterocedasticidad y al ver que existe
#me surge la duda de si debo seguir estimando el modelo o cambiar de variables ya que segun el
#Steep forward dichas variables eran las mas adecuadas.

vif(modelo3)
```

```
##             Age NBA_DraftNumber               G              WS
##        1.015105        1.149761        1.670714        1.727215
```

```
sqrt(vif(modelo3)) > 2
```

```
##             Age NBA_DraftNumber               G              WS
##           FALSE           FALSE           FALSE           FALSE
```

```
#Al menos no existe problema de multicolinealidad, cosa que he podido comprobar a traves del
#Factor de Inflacion de la Varianza

#En este punto he hecho K-Fold Cross Validation manteniendo el modelo original (Modelo3) y
#otro modelo alternativo sin la variable "Age".

library(boot)
```

```
##
## Attaching package: 'boot'

## The following object is masked from 'package:car':
##
##     logit
```

```
glm.fit1=glm(Salary ~ Age + G + NBA_DraftNumber + WS, data = nba,family = gaussian())
cv.err =cv.glm(nba,glm.fit1,K=5)
cv.err$delta
```

```
## [1] 2.994391e+13 2.976310e+13
```

```
glm.fit2=glm(Salary~  G + NBA_DraftNumber + WS, data = nba,family = gaussian())
cv.err2 =cv.glm(nba,glm.fit2,K=5)
cv.err2$delta
```

```
## [1] 3.334631e+13 3.327263e+13
```

A la vista de los resultados puedo decir que las variables que mejor predicen el salario que deben percibir los jugadores son: la edad (Age), la ronda en la que fueron seleccionados en el draft, G la cual creo que es el numero de partidos jugados y la responsabilidad de victorias (WS). A traves de K-Fold Validation hice una regresion de dos modelos alternativos inicialmente uno con las 4 variables que he usado para el "Modelo3" y otro eliminando la variable edad, una vez calculada la delta (la cual he considerado que es un termino de error) he decido quedarme con el modelo de 4 variables en vez de quedarme con el de 3 ya que para el de 4 variables el coeficiente de delta es menor.

Win Shares es una estadistica que se utiliza para subsanar la falta de datos sobre las perdidas de balon y otras metricas que antes de la dÃ©cada de los 70 no se median.

Para ello:

Calcular los puntos producidos (canastas, asistencias, perdidas de balon forzadas, etc).

Calcular la cantidad de posesiones en las que se vio el jugador involucrado.

Calcular la ofensiva marginal del jugador: Aqui entran en juego muchos factores, se toma la ofensiva del jugador y sus posesiones combinÃ¡ndose con el promedio de posesiones de la liga.

Calcular la ofensiva marginal por victoria: Se introducen factores como: el ritmo de juego promedio, ritmo de juego del equipo y anotacion global de la liga.

Para obtener Defensive Win Shares (DWS) estos son los pilares:

Calcular el rating defensivo de jugador.

Calcular el aporte marginal defensivo del jugador: Una vez mas, en defensa entran muchos factores en la ecuacion pero para este cÃ¡lculo nos centramos en puntos como: minutos jugados, ritmo de juego, eficiencia ofensiva de la liga y cantidad de posesiones defensivas del equipo.

Calcular defensiva marginal del jugador: Mismo calculo que en ofensiva, pero para defensiva. Se dividen los ultimos dos valores y asi obtenemos el Defensive WS (DWS).

Al ser tanto OWS y DWS parte de WS solo he tomado la variable Ws como conjunto de ambas y no he tomado por ello las variables OWS y DWS por ser consideradas marginales de WS.

Fuente: Rompimiento Defensivo

Figure 1: NBA