CV_RS

Daniel Ramos

24/2/2021

Superficies de Respuesta

1. Seleccionar el valor de p para q (0:99; n), mediante validación cruzada. Estableced set.seed(1). ¿Qué error cuadrático medio tiene?

```
#Cargamos la libreria
library(boot)
#Cargamos los datos
df = read.csv2("CV_RS_quantils.csv")
#Establecemos la semilla de aleatoriedad
set.seed(1)
cv.error10=rep(0,10)
#Hacemos la regresión
for(i in 1:10){
    glm.fit <- glm(q99 ~ poly((1/(n^(1/2))),degree=i),data = df)
    cv.error10[i]=cv.glm(df,glm.fit,K=10)$delta[1]
}</pre>
```

```
cv.error10

## [1] 257.93730 72.21903 47.94495 27.99772 17.91736 16.20601 19.22153
## [8] 17.58199 16.11194 15.58872

#Error cuadrático medio:
mean(cv.error10)
```

[1] 50.87265

2. ¿Crees que el orden pmax = 10 es adecuado?

```
plot(1:10,cv.error10,type="b", xlab= 'Orden', ylab = 'Error CV')
```



Probablemente aumentando el pmax el error seguiria disminuyendo, pero tiene un coste computacional muy alto. Por ello si nos fijamos en el gráfico, podemos observar que el error disminuye mucho en el segundo orden y luego cada vez disminuye menos, por tanto el pmax podría ser menor, por ejemplo 5, ya que apartir de ese punto el error casi no disminuye.

3. Dada la ordenación inicial de las observaciones, ¿crees que el método de validación cruzada k-fold puede tener problemas?

No debería tener problemas ya que la validación cruzada deberia hacer k particiones de muestras aleatorias. Aún así vamos a comprobarlo de manera empírica:

```
set.seed(1)
#Reordenamos los datos de manera aleatoria
rows <- sample(nrow(df))
df <- df[rows, ]
#Volvemos a estimar
cv.error10=rep(0,10)
for(i in 1:10){
   glm.fit <- glm(q99 ~ poly((1/(n^(1/2))),degree=i), data = df)
   cv.error10[i]=cv.glm(df,glm.fit,K=10)$delta[1]
}
cv.error10</pre>
```

```
## [1] 260.66036 66.71267 47.59300 28.77894 17.51349 16.17752 16.16838 ## [8] 16.39122 16.57098 16.30076
```



Por tanto el orden inicial de las observaciones no es relevante a la hora de hacer CV