EXAMEN TEMA 3 MACHINE LEARNING

Jorge Candia & Alejandro Valbuena

PASO #0:

Instalar el paquete ISLR, llamar a todas las librerías que voy a utilizar y descargar el dataset, además de seleccionar el directorio de trabajo.

PASO #1:

Rápida examinación del conjunto de datos mediante explorar el documento, unos plots y eliminar NAs en caso de haberlos. Aprovechamos para eliminar variables claramente no explicativas, como *region*, ya que sólo se contempla una región.

PASO #2:

Comenzamos a probar modelos.

La R^2 para *wage* con únicamente *age* es la más alta de todas, un 0.2348, seguida de *health_ins*, 0.09505, mientras que *year*, *race* y *health* se mantienen menor o iguales a 0.02, y *maritl* y *jobclass* se llegan a alrededor del 0.05 (y la única que queda es *age* con aprox. 0.03).

Al observar que en un modelo con todas la variables metidas, los vifs son todos muy bajos (todos por debajo de 1.5), se procede a probar modelos haciendo interaccionar las variables, y viendo qué pasa si se quita alguna. Al ser los vifs tan bajos, las variables por sí solas únicamente suman, ya que al no haber correlación entre ellas, no compiten por mayor nivel de significación. Retirar variables no tiene sentido, ya que al calcular los intervalos de confianza y realizar los t-test, se comprueba que en mayor o menor medida, todas son explicativas. Además, los F-test dan buenos resultados, por lo que los modelos son explicativos.

Realizando una comprobación exhaustiva de una gran variedad de modelos, se llega a la conclusión de que el mejor modelo de regresión para este conjunto de datos es el siguiente:

```
model <- lm(wage ~ year+age*maritl+race*education+jobclass+health*health_ins,data=wages)
```

La R^2 obtenida es de tan solo 0.3438. Si multiplicamos *jobclass* a *race* y *education*, se consigue aumentar una centésima este coeficiente, pero hay tanta interacción entre variables que el comando vif() da error, así que se ha decidido dejar el modelo ya expuesto.

#PASO 3:

Por último, se comprobará la eficiencia del modelo con datos aún no explorados. Se hará con el modelo presentado en el paso #2 y con un modelo en el que están todas las variables sin interacción alguna para así compararlos.

Se observa que ambos modelos tienen un resultado muy similar, obteniendo casi la misma desviación típica en los residuos, alrededor de un 36, llegando el modelo que incluye interacciones a media décima menos que el inicial. Para la mejor interpretación de los resultados, se han realizado numerosos plots de los residuos para analizar los valores obtenidos y los que se deberían haber obtenido.

#CONCLUSIÓN (Sobre el modelo con interacciones, extendible al inicial por su similitud)

Se puede observar que a pesar de tener una R^2 baja, las predicciones de los datos de test no son del todo malas, ya que el RSE (Residual Standard Error) es de 29.96%, por lo que en cierta medida nuestro modelo es significativamente explicativo.

Por otra parte, se ha observado que existe una gran cantidad de outliers. Estos sesgan el modelo y lo empeoran, y si se retiraran se ganaría un mayor nivel de significación. En estos momentos no disponemos del conocimiento/habilidad con R para retirar estos datos de forma eficiente, por lo que se dejan en el dataset a sabiendas de que sería beneficioso quitarlos.

Por otra parte, aclarar que *logwage* se ha dejado en el conjunto pero no se ha utilizado, ya que no es un dato como tal, sino la variable dependiente en otra escala.

PARÉNTESIS:
[Al escribir esta última frase hemos pensado que igual la clave está en usar como variable explicada los salarios en escala logarítmica. Tras hacer una copia del script pero tomando como variable explicada *logwage* en vez de *wage* en todo momento, he conseguido elevar la R^2 un 0.04 hasta un 0.3812 (modelo con interacciones), pero sobre todo bajar la RSE a un 5.90% del 29.96% anterior, aunque pensamos que esta medida está sesgada por el cambio de escala.**]**

Estos resultados dejan en evidencia que a pesar de que el coeficiente de determinación es un buen indicador para preveer si el modelo va a funcionar mejor o peor, se puede sesgar, por lo que también hay que ver los resultados del modelo con datos nuevos (es decir, poner el modelo a prueba) en vez de desecharlo de primeras y ver cómo se comporta.

NOTA: Se subirán a moodle ambos scripts, el que toma como variable dependiente *wage* (Examen3.R), y el que toma *logwage* (Examen3Log.R)

CÓDIGO EN R, IGNORAR (ARCHIVO SUBIDO A MOODLE)

```
#PASO 0
setwd('/Users/jorgecandia/UNIVERSIDAD/TERCERO/Machine Learning/Prácticas R/E3')
#install.packages('ISLR')

library(ISLR)
library(tidyverse)      #data manipulation and visualization
library("car")  # For VIF
library("psych")  # For multi.hist
library("corrplot") #For corrplot
library("modelr")  # For add_predictions
library(varhandle)

#data(package = 'ISLR') #Inspecciono datasets de ISLR
wages <- as.data.frame(Wage)


#PASO 1
any(is.na(wages)) #No hay NAs en el dataframe

summary(wages)
str(wages)

wages <- wages[,-c(6)]

pairs(~year+age+maritl+race+education+jobclass+health+health_ins+logwage+wage,
      main='scatterplots', col=c('blue'), data=wages)

pairs(~wage+maritl+race+education+jobclass, #Realmente no se puede apreciar nada
      main='scatterplots', col=c('blue'), data=wages)

wagesPrueba <- wages[,c(1,2,5,10)]

library(ggcorrplot)
model.matrix(~0+., data=wagesPrueba) %>%
  cor(use="pairwise.complete.obs") %>%
  ggcorrplot(show.diag = F, type="lower", lab=TRUE, lab_size=2)



#PASO 2

modelC <- lm(wage ~ year+age+maritl+race+education+jobclass+health+health_ins,
data=wages)
summary(modelC)
vif(modelC)
confint(modelC)
```

```
model <- lm(wage ~ year+age*maritl+race*education+jobclass+health*health_ins,
data=wages)
summary(model)
vif(model)
confint(model)



#PASO 3

set.seed(100)
indexes <- sample(nrow(wages), 0.8*nrow(wages), replace = F)
trainData <- wages[indexes,]
testData <- wages[-indexes,]

#MODELO COMPLETO (modelC)

modelCTrain <- lm(wage ~ year+age+maritl+race+education+jobclass+health+health_ins,
data=trainData)
summary(modelCTrain)

predictionsCTrain = predict(modelCTrain, new=trainData)
predictionsCTest = predict(modelCTrain, new=testData)

resCTrain <- trainData$wage - predictionsCTrain
resCTest  <- testData$wage - predictionsCTest

summary(resCTrain)
summary(resCTest)
sd(resCTrain)
sd(resCTest)


ggplot(data=modelCTrain, mapping= aes(x=modelCTrain$residuals))+
  geom_histogram(binwidth=0.5, col=c('blue'))

hist(resCTest)

sd(wages$wage) #La desviacion de los wages reales

plot(modelCTrain, which=1,  id.n=NULL, col=c('blue'))

plot(modelCTrain, which=2,  id.n=3, col=c('blue'))

plot(modelCTrain, which=4,  id.n=3, col=c('blue'))
```

```
#MODELO REFINADO (model)

modelTrain <- lm(wage ~ year+age*maritl+race*education+jobclass+health*health_ins,
data=trainData)
summary(modelTrain)

predictionsTrain = predict(modelTrain, new=trainData)
predictionsTest = predict(modelTrain, new=testData)

resTrain <- trainData$wage - predictionsTrain
resTest  <- testData$wage - predictionsTest

summary(resTrain)
summary(resTest)
sd(resTrain)
sd(resTest)

#RSE %
RSE <- 33.47/mean(wages$wage) * 100  # 29.96%


summary(wages$wage)
sd(wages$wage) #La desviacion de los wages reales

ggplot(data=modelTrain, mapping= aes(x=modelTrain$residuals))+
  geom_histogram(binwidth=0.5, col=c('blue'))

hist(resTest)

plot(modelTrain, which=1,  id.n=NULL, col=c('blue'))

plot(modelTrain, which=2,  id.n=3, col=c('blue'))

plot(modelTrain, which=4,  id.n=3, col=c('blue'))
```