Introducción a la Analítica
Módulo 1 - Tarea 2

Universidad Nacional de Colombia - Escuela de Estadística

Luisa María Acosta—Laura Camila Agudelo—Sebastián Agudelo
Andrea Amaya—Estefanía Echeverry

Septiembre 2020

1. (10 pts. Teórico)

Considere el estadıstico leverae:

$$h_{ii} = \frac{1}{n} + \frac{(x_i - \bar{x})^2}{\sum_{i'=1}^{n} (x_{i'} - \bar{x})^2}$$

Demuestre que

$$\frac{1}{n} \leq h_{ii} \leq 1$$

Solución

PRUEBA DEL ESTADÍSTICO LEVERAGE

$$\frac{1}{n} \leq h_{ii} \leq 1$$

$$\bar{h_{ii}} = \frac{\sum_{i=1}^{n} h_{ii}}{n} = \frac{p+1}{n}$$

Probando el lado izquierdo: $\frac{1}{n} \leq h_{ii}$

$$\frac{1}{n} \leq \frac{1}{n} + \frac{(x_i - \bar{x})^2}{\sum_{i\prime=1}^{n} (x_{i\prime} - \bar{x})^2}$$

Se sabe que $(x_i - \bar{x})^2$ es una diferencia al cuadrado, por lo cual siempre es positiva. Y esto, hace también que $\sum_{i'=1}^{n} (x_{i'} - \bar{x})^2$ sea igualmente positiva (suma de valores positivos). Por tanto, como:

$$\frac{(x_i - \bar{x})^2}{\sum_{i'=1}^{n} (x_{i'} - \bar{x})^2} \geq 0$$

$$\Longrightarrow \frac{1}{n} \leq h_{ii}$$

Probando el lado derecho: $h_{ii} \leq 1$ El apalancamiento de la observación i es el valor del i-ésimo término diagonal principal(h_{ii}) de la matriz de sombrero, **H**nxn, donde $H = X(X^TX)^{-1}X^T$

$$\mathbf{H} = \begin{bmatrix} h_{11} & h_{12} & \cdots & h_{1n} \\ h_{21} & h_{22} & & h_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ h_{n1} & h_{n2} & \cdots & h_{nn} \end{bmatrix}$$

Propiedades de la matriz **H**:
La matriz **H** es simétrica: $h_{ij} = h_{ji}$, es decir $H = H^T$.
La matriz **H** es idempotente:
$H^2 = X(X^TX)^{-1}X^T * X(X^TX)^{-1}X^T = I * X(X^TX)^{-1}X^T = H$
Con lo anterior, se tiene que:

$$\mathbf{H}^2 = \begin{bmatrix} h_{11}^2 + \sum_{i \neq j}^{n} h_{ij}^2 & \cdots & \cdots & \vdots \\ \vdots & h_{22}^2 + \sum_{i \neq j}^{n} h_{ij}^2 & \ddots & \vdots \\ \cdots & \cdots & \ddots & h_{nn}^2 + \sum_{i \neq j}^{n} h_{ij}^2 \end{bmatrix}$$

Si se igualan los elementos de la diagonal principal de la matriz **H** con los elementos de la matriz H^2. Se tiene que:

$$h_{ii} = \underbrace{h_{ii}{}^2 + \sum_{i \neq j}^{n} h_{ij}{}^2}_{\nabla} \geq 0$$

Como h_{ii} es igual a ∇ (un término al cuadrado más una suma de cuadrados), entonces:

$$h_{ii} \geq h_{ii}{}^2 \implies h_{ii} \leq 1 \ ;$$

2. (50 pts. Práctico)

Considere el conjunto de datos anexo (bank.csv) el cual tiene 17 variables. Asuma que el supervisor es la variable loan.

Solución

a) Cree un conjunto de datos de entrenamiento del 75 % y el restante 25 % trátelo como datos de test o de prueba.

Antes de dividir la base, observemos la estructura de los datos.

```
> str(bank)
'data.frame':   11162 obs. of  17 variables:
 $ age      : int  59 56 41 55 54 42 56 60 37 28 ...
 $ job      : Factor w/ 12 levels "admin.","blue-collar",..: 1 1 10 8 1 5 5 6 10 8 ...
 $ marital  : Factor w/ 3 levels "divorced","married",..: 2 2 2 2 2 3 2 1 2 3 ...
 $ education: Factor w/ 4 levels "primary","secondary",..: 2 2 2 2 3 3 3 2 2 2 ...
 $ default  : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...
 $ balance  : int  2343 45 1270 2476 184 0 830 545 1 5090 ...
 $ housing  : Factor w/ 2 levels "no","yes": 2 1 2 2 1 2 2 2 2 2 ...
 $ loan     : Factor w/ 2 levels "no","yes": 1 1 1 1 1 2 2 1 1 1 ...
 $ contact  : Factor w/ 3 levels "cellular","telephone",..: 3 3 3 3 3 3 3 3 3 3 ...
 $ day      : int  5 5 5 5 5 5 6 6 6 6 ...
 $ month    : Factor w/ 12 levels "apr","aug","dec",..: 9 9 9 9 9 9 9 9 9 9 ...
 $ duration : int  1042 1467 1389 579 673 562 1201 1030 608 1297 ...
 $ campaign : int  1 1 1 1 2 2 1 1 1 3 ...
 $ pdays    : int  -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...
 $ previous : int  0 0 0 0 0 0 0 0 0 0 ...
 $ poutcome : Factor w/ 4 levels "failure","other",..: 4 4 4 4 4 4 4 4 4 4 ...
 $ deposit  : Factor w/ 2 levels "no","yes": 2 2 2 2 2 2 2 2 2 2 ...
```

De la anterior figura se aprecia que hay 10 variables de tipo de factor y 7 de tipo numérico.

Luego, se procede a dividir aleatoriamente la base en dos conjuntos de datos, train y test. Como se muestra a continuación:

```
set.seed(2224)
df=data.frame(bank)

smp_size <- floor(0.75* nrow(df))
train_ind <- sample(seq_len(nrow(df)), size = smp_size)
train <- df[train_ind, ]
test <- df[-train_ind, ]
y_train <- df[train_ind,8]
y_test <- df[-train_ind,8]
```

b) Con los datos de entrenamiento, implemente Naive Bayes usando loan como el supervisor y las demás como predictores.

```
modelonb <- naive_bayes(loan~.,data=train,laplace=0.128)
```

Usando la función summary, vemos que el aproximadamente el 87 % de los datos son clasificados como no y el 13 % como si.

```
> summary(defaulted_classifier)

================================== Naive Bayes ==================================


- Call: naive_bayes.formula(formula = loan ~ ., data = train, laplace = 0.128)
- Laplace: 0.128
- Classes: 2
- Samples: 8371
- Features: 16
- Conditional distributions:
    - Bernoulli: 3
    - Categorical: 6
    - Gaussian: 7
- Prior probabilities:
    - no: 0.8699
    - yes: 0.1301
```

c) Con los datos de entrenamiento, implemente Knn usando loan como el supervisor y las demas como predictores. Ensaye con varios valores de K y reporte solo uno de acuerdo a su preferencia. Observe que algunas variables son categóricas y se deben crear variables dummies.

Para crear un modelo usando el método knn, se hizo uso de la función knn de la libreria class del software R. Se ensayo con cuatro valores distintos para k, los cuales fueron 1, 3, 5 y 10. De estos cuatro modelos implementados se escogio el modelo con k=1, esta decisión se baso en la matriz de confusión de los cuatro modelos (los cuales se reportan en los literales f) y g)). Acontinucaión se muestra la implementación del modelo knn con k=1. Para este modelo se crearon variables dummies para cada variable categorica y se normalizo las variables numéricas, así:

```
#Creando variables dummy
df=data.frame(bank)

djob <- dummy(df$job ,sep="_")
dmar <- dummy(df$marital, sep="_")
dedu <- dummy(df$education, sep="_")
dcon <- dummy(df$contact, sep="_")
dmon <- dummy(df$month, sep="_")
dpou <- dummy(df$poutcome, sep="_")
ddef <- dummy(df$default, sep="_")
```

```
dhou <- dummy(df$housing, sep="_")
ddep <- dummy(df$deposit, sep="_")

#Normalizando las variables numericas
nage <- scale(df$age)
nbal <- scale(df$balance)
nday <- scale(df$day)
ndur <- scale(df$duration)
ncam <- scale(df$campaign)
npda <- scale(df$pdays)
npre <- scale(df$previous)

#Nueva base de datos
Newdata <- cbind(nage,djob,dmar,dedu,ddef,nbal,dhou,dcon,
nday,dmon,ndur,ncam,npda,npre,dpou,ddep)
head(Newdata)

k1 <- knn(train1, train1, y_train1, k=1, prob=T)
> summary(k1)
  no  yes
8192 179
```

Note que este modelo clasifica 7157 datos como no y 1214 datos como si.

d) Con los datos de entrenamiento, implemente Regresión logística usando loan como el supervisor y las demás como predictores.

```
mod <- glm(y_train~., data = train, family = binomial())

summary(mod)

Call:
glm(formula = y_train ~ ., family = binomial(), data = train)

Deviance Residuals:
```

```
       Min          1Q      Median          3Q         Max
-2.409e-06  -2.409e-06  -2.409e-06  -2.409e-06   2.409e-06

Coefficients:
                     Estimate Std. Error z value Pr(>|z|)
(Intercept)        -2.657e+01  4.094e+04  -0.001    0.999
age                -9.379e-15  4.751e+02   0.000    1.000
jobblue-collar     -2.279e-13  1.541e+04   0.000    1.000
jobentrepreneur     7.763e-14  2.632e+04   0.000    1.000
jobhousemaid        2.030e-13  2.845e+04   0.000    1.000
jobmanagement      -1.857e-13  1.624e+04   0.000    1.000
jobretired          3.299e-13  2.192e+04   0.000    1.000
jobself-employed    6.084e-14  2.395e+04   0.000    1.000
jobservices        -2.331e-13  1.784e+04   0.000    1.000
jobstudent         -2.333e-13  2.624e+04   0.000    1.000
jobtechnician       4.825e-14  1.510e+04   0.000    1.000
jobunemployed      -1.762e-13  2.485e+04   0.000    1.000
jobunknown         -3.577e-14  5.200e+04   0.000    1.000
maritalmarried     -1.849e-13  1.267e+04   0.000    1.000
maritalsingle      -2.082e-13  1.472e+04   0.000    1.000
educationsecondary -1.834e-13  1.326e+04   0.000    1.000
educationtertiary   7.055e-14  1.590e+04   0.000    1.000
educationunknown    7.196e-14  2.249e+04   0.000    1.000
defaultyes         -5.439e-12  3.200e+04   0.000    1.000
balance             1.893e-18  1.228e+00   0.000    1.000
housingyes          4.187e-13  9.439e+03   0.000    1.000
loanyes             5.313e+01  1.200e+04   0.004    0.996
contacttelephone    2.066e-14  1.619e+04   0.000    1.000
contactunknown     -8.208e-14  1.361e+04   0.000    1.000
day                 2.473e-14  5.205e+02   0.000    1.000
monthaug           -2.231e-13  1.796e+04   0.000    1.000
monthdec           -2.510e-14  4.244e+04   0.000    1.000
monthfeb            3.480e-13  2.097e+04   0.000    1.000
monthjan            3.388e-13  2.631e+04   0.000    1.000
```

```
monthjul           -6.334e-13  1.773e+04   0.000    1.000
monthjun            3.530e-13  2.052e+04   0.000    1.000
monthmar            1.963e-13  2.831e+04   0.000    1.000
monthmay            5.187e-15  1.703e+04   0.000    1.000
monthnov            2.956e-13  1.917e+04   0.000    1.000
monthoct            1.358e-13  2.542e+04   0.000    1.000
monthsep            2.154e-13  2.746e+04   0.000    1.000
duration           -5.946e-17  1.334e+01   0.000    1.000
campaign            2.319e-14  1.501e+03   0.000    1.000
pdays               4.189e-16  6.574e+01   0.000    1.000
previous            3.617e-14  2.151e+03   0.000    1.000
poutcomeother      -9.315e-14  2.145e+04   0.000    1.000
poutcomesuccess    -3.394e-14  1.868e+04   0.000    1.000
poutcomeunknown    -4.031e-14  2.182e+04   0.000    1.000
deposityes         -2.485e-13  1.027e+04   0.000    1.000

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 6.4718e+03  on 8370  degrees of freedom
Residual deviance: 4.8565e-08  on 8327  degrees of freedom
AIC: 88

Number of Fisher Scoring iterations: 25
```

Del summary de este modelo vemos que ninguna variable es significativa pues sus valores p son grandes, además todas poseen un error estándar pequeño y se reporta un AIC de 88.

e) Con los datos de entrenamiento, implemente LDA usando loan como el supervisor y las demás como predictores.

```
model <- lda(loan~., data=train)
Prior probabilities of groups:
      no      yes
0.869908 0.130092
```

De este modelo se ve que aproximadamente el 87 % de los datos son clasificados como no y el 13 % como si.

f) Con los datos de entrenamiento, para cada uno de los métodos anteriores, calcule el training-MSE, la matriz de confusión y grafique la curva ROC.

	Train Naive Bayes	
Predict	no	yes
no	6602	847
yes	680	242
MSE	0.1824	

	Train Knn	
Predict	no	yes
no	2366	264
yes	41	120
MSE	0.1092	

	Train RL	
Predict	0	1
0	7282	0
1	0	1089
MSE	0	

	Train LDA	
Predict	no	yes
no	7247	1061
yes	35	28
MSE	0.1309	

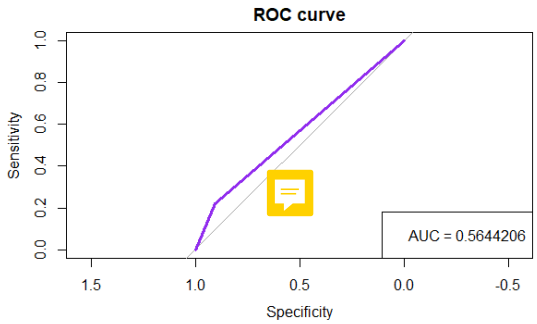


Figura 1: Curva ROC con training set para Naive Bayes.

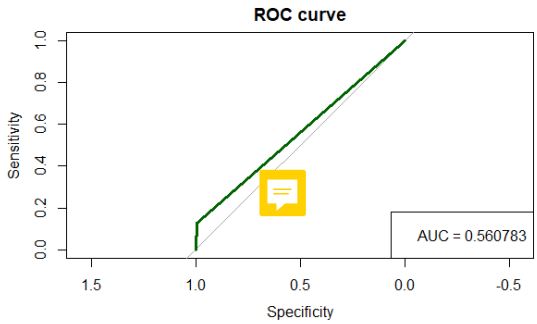


Figura 2: Curva ROC con training set para knn.

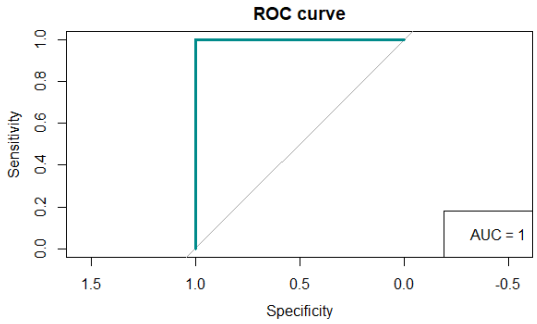


Figura 3: Curva ROC con training set para regresión logística.

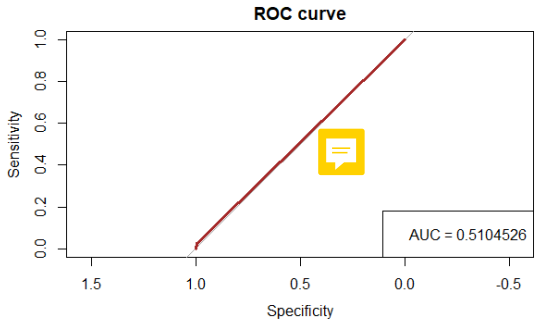


Figura 4: Curva ROC con training set para LDA.

g) Use los respectivos ajustes de cada uno de los modelos anteriores y con el conjunto de prueba, calcule el test-MSE, la matriz de confusión y grafique la curva ROC.

	Test Naive Bayes	
Predict	no	yes
no	2169	297
yes	251	74
MSE	0.1963	

	Test KNN	
Predict	no	yes
no	2366	264
yes	41	120
MSE	0.1092	

	Test RL	
Predict	0	1
0	2420	0
1	0	371
MSE	0	

	Test LDA	
Predict	no	yes
no	2402	352
yes	18	19
MSE	0.1325	

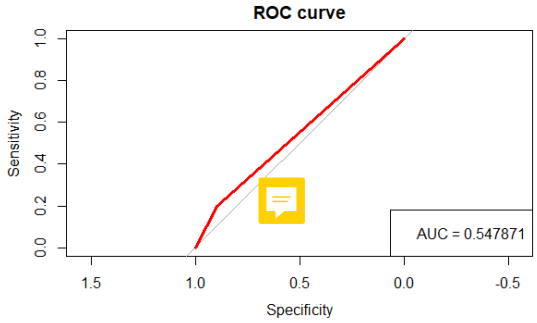


Figura 5: Curva ROC con test set para Naive Bayes.

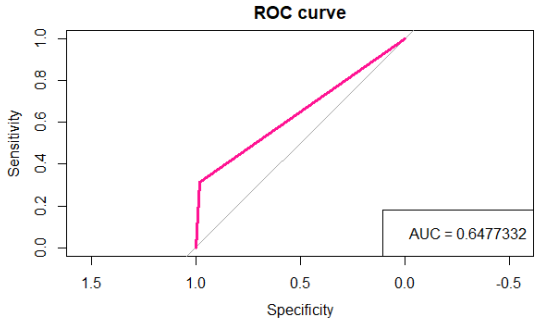


Figura 6: Curva ROC con test set para Knn.

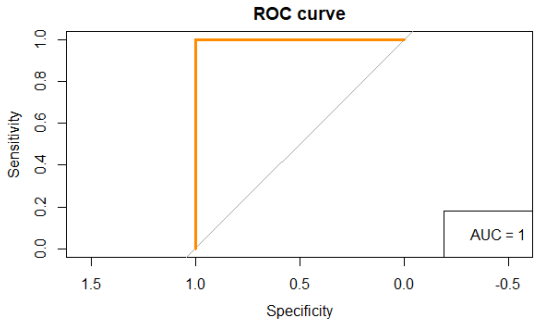


Figura 7: Curva ROC con test set para Regresión logística.

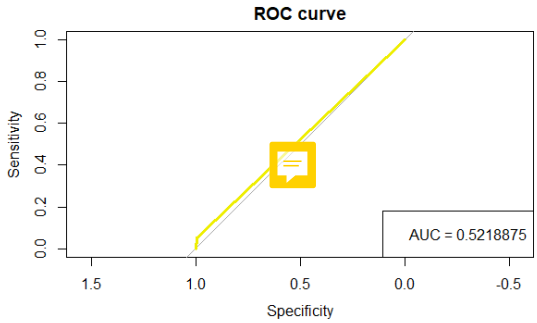


Figura 8: Curva ROC con test set para LDA.

h) ¿Con cuál modelo observó un mejor desempeño y porqué?

El mejor modelo observado, fue el modelo con regresión logística, pues esté obtuvó una predicción perfecta. Además, cuando se realizó la curva ROC, se puede apreciar que es impecable y que el área bajo la curva es de uno, tanto con el conjunto de entrenamiento como con el conjunto de prueba.

3. (40 pts. Práctico)

Considere el conjunto de datos anexo (costumer loan details.csv) el cual tiene 12 variables incluyendo el ID. Asuma que el supervisor es la variable income.

Solución

a) Cree un conjunto de datos de entrenamiento del 75 % y el restante 25 % trátelo como datos de test o de prueba.

Solución

En primer lugar hacemos un análisis exploratorio de los datos. A continuación podemos ver que la base de datos tiene 114 observaciones con 12 variables, de las cuales 5 son de tipo factor, 4 de tipo numérico y 3 de tipo carácter.

```
> str(data)
'data.frame':   114 obs. of  12 variables:
 $ applicantId       : chr  "004NZMX60E" "004NZMX60E" "017STAOLDV" "017WEFEN7S" ...
 $ state             : chr  "CA" "CA" "OH" "OH" ...
 $ gender            : Factor w/ 2 levels "Female","Male": 2 2 1 2 2 2 1 1 1 2 ...
 $ age               : int  36 36 34 48 32 44 60 60 60 48 ...
 $ race              : Factor w/ 7 levels "American Indian or Alaska Native",..: 5 5 7 5 7 6 2 2 2 1 ...
 $ marital_status    : Factor w/ 3 levels "Divorced","Married",..: 2 2 2 2 3 3 3 3 3 2 ...
 $ occupation        : chr  "NYPD" "NYPD" "IT" "Accout" ...
 $ credit_score      : int  710 720 720 670 720 540 840 824 824 490 ...
 $ income            : num  9371 9371 9010 6538 8679 ...
 $ debts             : num  2000 3014 1000 2099 1000 ...
 $ loan_type         : Factor w/ 4 levels "Auto","Credit",..: 4 1 2 3 3 4 4 1 2 4 ...
 $ loan_decision_type: Factor w/ 3 levels "Approved","Denied",..: 1 1 1 1 1 2 1 1 1 2 ...
```

Para realizar la paritición de datos utilizamos el siguiente código en el software estadístico **R**, el cual divide los datos aleatoriamente con el 75 % y 25 % para los conjuntos de entrenamiento y prueba respectivamente :

```
n = nrow(data)
trainIndex = sample(1:n, size = round(0.75*n), replace=FALSE)
train = data[trainIndex ,]
test = data[-trainIndex ,]
```

De esta manera, la base de entrenamiento queda con 86 obsevarciones mientras la base de prueba con 28.

b) Con los datos de entrenamiento, implemente Knn (con al menos tres valores para K) usando income como el supervisor y debts como predictor. Grafique e interprete.

Solución

Para esto hacemos uso del software estadístico R, la función knn.reg de la librería *FNN* para realizar regresión con k-vecinos más cercanos. La variable respuesta es *income* y su respectiva variable predictora es *debts*.

```
par(mfrow=c(2,2))
for (i in 1:4){
  fit.knn <- knn.reg(train = as.data.frame(data2$debts) ,
                     test = as.data.frame(data2$debts) ,
                     as.data.frame(data2$income),k = 4*i,
                     algorithm=c("brute"))

  plot(data2$debts, data2$income, col="black", ylab = "income",
       xlab = "debts")
  points(data2$debts,fit.knn$pred,pch=1,col="red")
}
```

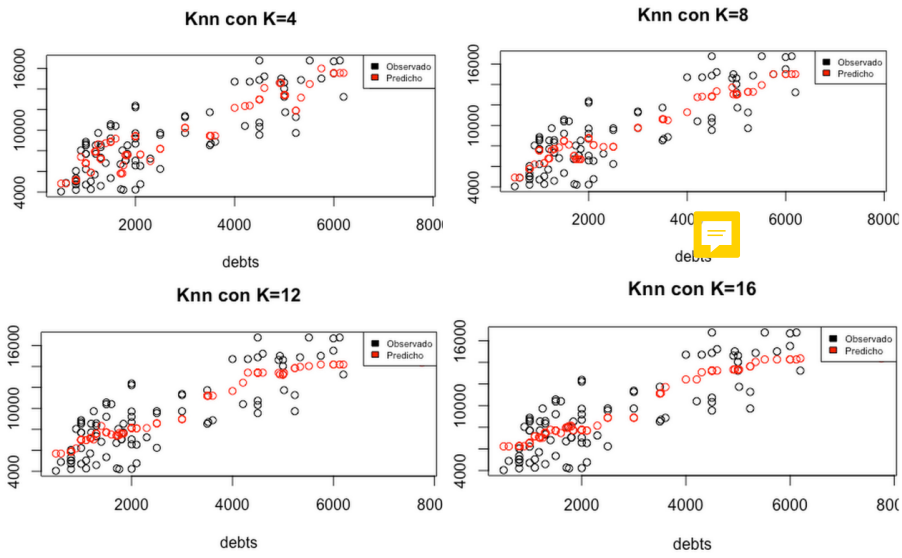


Figura 9: Regresión KNN con K = 4,8,12,16

Como podemos observar el modelo knn parece captar mejor la señal de los datos a medida que se aumenta el número de vecinos k, aunque se deja llevar un poco por el ruido o la variación parece realizar una buena estimación. Es este caso, dado los 4

valores de k, el que mejor comportamiento lo presenta k = 16.

c) Con los datos de entrenamiento, implemente regresión lineal simple usando income como el supervisor y debts como predictor. Grafique e interprete.

Solución

Veamos su implemetación en el software R, haciendo uso de la función lm() de la libreria stats:

```
mod <- lm(income ~ debts, data = data2)
```

Obteniendo los siguientes resultados:

```
    Call:
lm(formula = income ~ debts, data = data2)

Residuals:
    Min      1Q  Median      3Q     Max
-4123.6 -1658.1     4.7  1486.8  4359.0

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 4608.4708   421.4484   10.94   <2e-16 ***
debts          1.7313     0.1364   12.70   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2051 on 84 degrees of freedom
Multiple R-squared:  0.6574,Adjusted R-squared:  0.6533
F-statistic: 161.2 on 1 and 84 DF,  p-value: < 2.2e-16
```

Según los resultados, por cada unidad de pesos que aumenta la deuda, los ingresos incrementan $1.7313, además la variable debts es significativa, aunque, de acuerdo al $R^2_{ajustado}$ el modelo logra explicar en un 65 % la variabilidad de los datos.

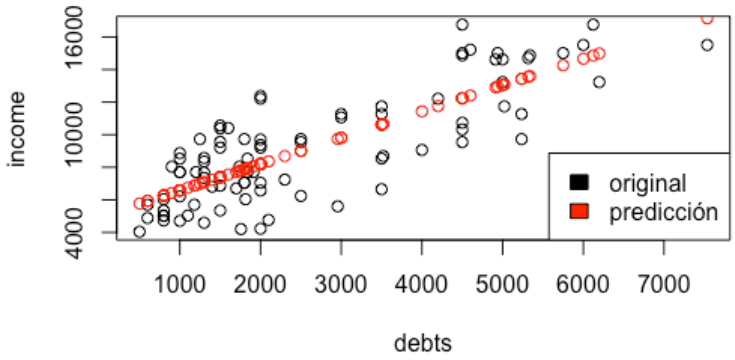


Figura 10: Regresión simple

Como podemos apreciar en la figura 10, el modelo de regresión lineal simple logra captar la señal de los datos, pero de acuerdo la dispersión que se aprecia es sugerente hacer uso de otra técnica o incluir más variables para mejorarlo.

d) Use los respectivos ajustes de cada uno de los modelos anteriores y con el conjunto de prueba, calcule el test-MSE. ¿Que observa?

Solución

El error cuadrático medio (MSE) es una diferencia cuadrática promedio entre los valores estimados y los observados o reales y nos interesa que esta medida del error sea lo más pequeña posible. lo podemos calcular con los datos de prueba como:

$$MSE\_test = \frac{1}{n}\sum_{i-1}^{n}(y_i - \hat{y}_i)^2$$

- Así entonces, obtenemos el MSE test para el modelo KNN para su respectivo valor de k corresponde a:

k	**4**	**8**	**12**	**16**
MSE_test	4'314.833	4'422.548	4'121.219	4'109.444

- Por otro lado, el MSE para el modelo de regresión simple corresponde a $4'547.590

Podemos notar que el error cuadrático medio resulta ser muy similar usando k-vecinos más cercanos y regresión simple. Por lo cual, en este caso Knn resulta ser un buen competidor con la regresión, obteniendo incluso una mejor medida de MSE.

e) Usando todos los datos y regresión lineal múltiple seleccione un modelo usando forward, backward y stepwise.

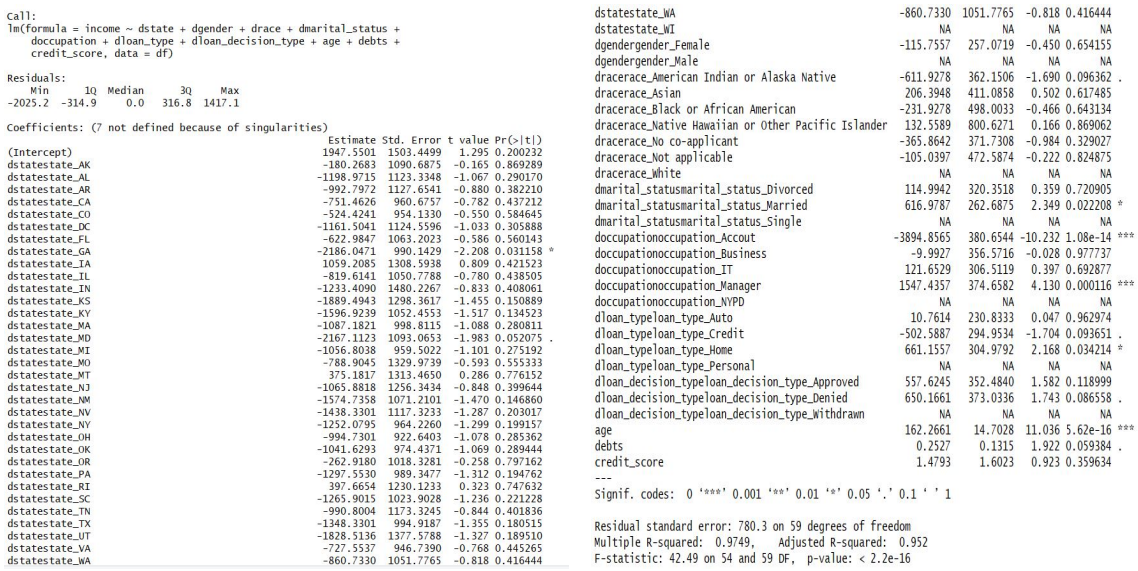
Solución

Para realizar la regresión lineal múltiple se toman todas las variables excepto el applicationId ya que es el ID de cada observación, también se crearon variables dummies para cada variable categórica.

Modelo Completo

```
#Creando Variables dummy
dstate <- dummy(df$state ,sep="_")
dgender <- dummy(df$gender, sep="_")
drace <- dummy(df$race, sep="_")
dmarital_status <- dummy(df$marital_status, sep="_")
doccupation <- dummy(df$occupation, sep="_")
dloan_type <- dummy(df$loan_type, sep="_")
dloan_decision_type <- dummy(df$loan_decision_type, sep="_")

#Modelo Regresión Lineal Múltiple
modelo <- lm(income ~ dstate + dgender + drace + dmarital_status + doccupation + dloa
summary(modelo)
```



```
Call:
lm(formula = income ~ dstate + dgender + drace + dmarital_status +
    doccupation + dloan_type + dloan_decision_type + age + debts +
    credit_score, data = df)

Residuals:
    Min      1Q  Median      3Q     Max
-2025.2  -314.9     0.0   316.8  1417.1

Coefficients: (7 not defined because of singularities)
                                                  Estimate Std. Error t value Pr(>|t|)
(Intercept)                                     1947.5501  1503.4499   1.295 0.200232
dstatestate_AK                                  -180.2683  1090.6875  -0.165 0.869289
dstatestate_AL                                 -1198.9715  1123.3348  -1.067 0.290170
dstatestate_AR                                  -992.7972  1127.6541  -0.880 0.382210
dstatestate_CA                                  -751.4626   960.6757  -0.782 0.437212
dstatestate_CO                                  -524.4241   954.1330  -0.550 0.584645
dstatestate_DC                                 -1161.5041  1124.5596  -1.033 0.305888
dstatestate_FL                                  -622.9847  1063.2023  -0.586 0.560143
dstatestate_GA                                 -2186.0471   990.1429  -2.208 0.031158 *
dstatestate_IA                                  1059.2085  1308.5938   0.809 0.421523
dstatestate_IL                                  -819.6141  1050.7788  -0.780 0.438505
dstatestate_IN                                 -1233.4090  1480.2267  -0.833 0.408061
dstatestate_KS                                 -1889.4943  1298.3617  -1.455 0.150889
dstatestate_KY                                 -1596.9239  1052.4553  -1.517 0.134523
dstatestate_MA                                 -1087.1821   998.8115  -1.088 0.280811
dstatestate_MD                                 -2167.1123  1093.0653  -1.983 0.052075 .
dstatestate_MI                                 -1056.8038   959.5022  -1.101 0.275192
dstatestate_MO                                  -788.9045  1329.9739  -0.593 0.555333
dstatestate_MT                                   375.1817  1313.4650   0.286 0.776152
dstatestate_NJ                                 -1065.8818  1256.3434  -0.848 0.399644
dstatestate_NM                                 -1574.7358  1071.2101  -1.470 0.146860
dstatestate_NV                                 -1438.3301  1117.3233  -1.287 0.203017
dstatestate_NY                                 -1252.0795   964.2260  -1.299 0.199157
dstatestate_OH                                  -994.7301   922.6403  -1.078 0.285362
dstatestate_OK                                 -1041.6293   974.4371  -1.069 0.289444
dstatestate_OR                                  -262.9180  1018.3281  -0.258 0.797162
dstatestate_PA                                 -1297.5530   989.3477  -1.312 0.194762
dstatestate_RI                                   397.6654  1230.1233   0.323 0.747632
dstatestate_SC                                 -1265.9015  1023.9028  -1.236 0.221228
dstatestate_TN                                  -990.8004  1173.3245  -0.844 0.401836
dstatestate_TX                                 -1348.3301   994.9187  -1.355 0.180515
dstatestate_UT                                 -1828.5136  1377.5788  -1.327 0.189510
dstatestate_VA                                  -727.5537   946.7390  -0.768 0.445265
dstatestate_WA                                  -860.7330  1051.7765  -0.818 0.416444
dstatestate_WI                                        NA         NA      NA       NA
dgendergender_Female                            -115.7557   257.0719  -0.450 0.654155
dgendergender_Male                                    NA         NA      NA       NA
dracerace_American Indian or Alaska Native      -611.9278   362.1506  -1.690 0.096362 .
dracerace_Asian                                  206.3948   411.0858   0.502 0.617485
dracerace_Black or African American             -231.9278   498.0033  -0.466 0.643134
dracerace_Native Hawaiian or Other Pacific Islander 132.5589 800.6271  0.166 0.869062
dracerace_No co-applicant                       -365.8642   371.7308  -0.984 0.329027
dracerace_Not applicable                        -105.0397   472.5874  -0.222 0.824875
dracerace_White                                       NA         NA      NA       NA
dmarital_statusmarital_status_Divorced           114.9942   320.3518   0.359 0.720905
dmarital_statusmarital_status_Married            616.9787   262.6875   2.349 0.022208 *
dmarital_statusmarital_status_Single                  NA         NA      NA       NA
doccupationoccupation_Accout                  -3894.8565   380.6544 -10.232 1.08e-14 ***
doccupationoccupation_Business                   -9.9927   356.5716  -0.028 0.977737
doccupationoccupation_IT                        121.6529   306.5119   0.397 0.692877
doccupationoccupation_Manager                  1547.4357   374.6582   4.130 0.000116 ***
doccupationoccupation_NYPD                            NA         NA      NA       NA
dloan_typeloan_type_Auto                         10.7614   230.8333   0.047 0.962974
dloan_typeloan_type_Credit                     -502.5887   294.9534  -1.704 0.093651 .
dloan_typeloan_type_Home                        661.1557   304.9792   2.168 0.034214 *
dloan_typeloan_type_Personal                          NA         NA      NA       NA
dloan_decision_typeloan_decision_type_Approved  557.6245   352.4840   1.582 0.118999
dloan_decision_typeloan_decision_type_Denied    650.1661   373.0336   1.743 0.086558 .
dloan_decision_typeloan_decision_type_Withdrawn       NA         NA      NA       NA
age                                             162.2661    14.7028  11.036 5.62e-16 ***
debts                                             0.2527     0.1315   1.922 0.059384 .
credit_score                                      1.4793     1.6023   0.923 0.359634
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 780.3 on 59 degrees of freedom
Multiple R-squared:  0.9749,    Adjusted R-squared:  0.952
F-statistic: 42.49 on 54 and 59 DF,  p-value: < 2.2e-16
```

(a) (b)

Figura 11: Summary del modelo completo

Mirando el summary se puede ver que tiene un R^2 ajustado $= 0.952$, es decir, que el modelo explica un 95.2 % la variabilidad de income, sin embargo se logra ver que puede haber variables que no son significativas por su valor-p, y esten asociadas debilmente con la respuesta.
Se procede a realizar los 3 modelos con los métodos forward, backward, y stepwise, y se obtuvo los siguientes resultados.

```
#Regresión hacia adelante
reg.adel <-  ols_step_forward_aic(modelo)
reg.adel
```

```
> reg.adel

                        Selection Summary
--------------------------------------------------------------------------------
Variable         AIC            Sum Sq              RSS          R-Sq      Adj. R-Sq
--------------------------------------------------------------------------------
age            2054.821    1006482664.253    426207906.976    0.70251       0.69986
doccupation    1869.919    1355578734.167     77111837.063    0.94618       0.94369
debts          1859.825    1363340054.177     69350517.052    0.95159       0.94888
--------------------------------------------------------------------------------
```

Figura 12: Regresión hacia delante

Según la regresión hacia delante las variables que se deben quedar en el modelo es `age,doccupation,debts`, las cuales son edad, ocupación de la persona y deudas.

```
#Regresión hacia atras
reg.atr <-  ols_step_backward_aic(modelo)
reg.atr
```

```
> reg.atr

                          Backward Elimination Summary
-----------------------------------------------------------------------------------------
Variable                AIC          RSS             Sum Sq         R-Sq      Adj. R-Sq
-----------------------------------------------------------------------------------------
Full Model             1892.828    35920283.953    1396770287.276    0.97493      0.95198
dstate                 1878.024    57279038.724    1375411532.505    0.96002      0.95089
drace                  1869.074    59873471.024    1372817100.205    0.95821      0.95181
dgender                1865.090    59881686.253    1372808884.976    0.95820      0.95229
credit_score           1863.697    60201373.493    1372489197.737    0.95798      0.95252
dloan_type             1863.470    64449661.950    1368240909.279    0.95501      0.95065
dloan_decision_type    1860.570    66225921.144    1366464650.085    0.95378      0.95025
dmarital_status        1859.825    69350517.052    1363340054.177    0.95159      0.94888
-----------------------------------------------------------------------------------------
```

Figura 13: Regresión hacia atrás

Para la regresión hacia atras se muestra las variables que deben eliminar como son `dstate,drace,dgender,credit_score,dloan_type,dloan_decision_type,dmarital_status`, algunas de las cuales son género, estado, estado civil, raza entre otras.

```
#Regresión por Segmentos
reg.seg <- ols_step_both_aic(modelo)
reg.seg
```

```
> reg.seg

                                  Stepwise Summary
-------------------------------------------------------------------------------------------------
Variable        Method       AIC          RSS             Sum Sq         R-Sq      Adj. R-Sq
-------------------------------------------------------------------------------------------------
age           addition    2054.821    426207906.976    1006482664.253    0.70251      0.69986
doccupation   addition    1869.919     77111837.063    1355578734.167    0.94618      0.94369
debts         addition    1859.825     69350517.052    1363340054.177    0.95159      0.94888
-------------------------------------------------------------------------------------------------
```

Figura 14: Regresión por segmentos

Según la regresión por segmentos las variables que se deben quedar en el modelo es `age,doccupation,debts`, las cuales son edad, ocupación de la persona y deudas.

Ahora realizando los modelos con las variables seleccionadas en los métodos forward, backward y stepwise son:

```
modeloa <- lm(income ~ age + doccupation + debts ,data=df)
summary(modeloa)
```

```
> summary(modeloa)

Call:
lm(formula = income ~ age + doccupation + debts, data = df)

Residuals:
     Min       1Q   Median       3Q      Max
-2602.05  -351.83    25.08   419.63  2332.49

Coefficients: (1 not defined because of singularities)
                               Estimate Std. Error t value Pr(>|t|)
(Intercept)                   2.310e+03  2.522e+02   9.163  4.0e-15 ***
age                           1.737e+02  9.039e+00  19.219  < 2e-16 ***
doccupationoccupation_Accout -4.035e+03  2.598e+02 -15.534  < 2e-16 ***
doccupationoccupation_Business -1.481e+02  2.580e+02  -0.574 0.567111
doccupationoccupation_IT      -2.791e+02  2.321e+02  -1.202 0.231949
doccupationoccupation_Manager  1.110e+03  2.648e+02   4.192  5.7e-05 ***
doccupationoccupation_NYPD            NA         NA      NA       NA
debts                          2.708e-01  7.825e-02   3.460 0.000776 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 805.1 on 107 degrees of freedom
Multiple R-squared:  0.9516,    Adjusted R-squared:  0.9489
F-statistic: 350.6 on 6 and 107 DF,  p-value: < 2.2e-16
```

Figura 15: Summary de regresión hacia adelante

```
> summary(modeloatr)

Call:
lm(formula = income ~ doccupation + age + debts, data = df)

Residuals:
     Min       1Q   Median       3Q      Max
-2602.05  -351.83    25.08   419.63  2332.49

Coefficients: (1 not defined because of singularities)
                              Estimate Std. Error t value Pr(>|t|)
(Intercept)                  2.310e+03  2.522e+02   9.163  4.0e-15 ***
doccupationoccupation_Accout -4.035e+03  2.598e+02 -15.534  < 2e-16 ***
doccupationoccupation_Business -1.481e+02  2.580e+02  -0.574 0.567111
doccupationoccupation_IT     -2.791e+02  2.321e+02  -1.202 0.231949
doccupationoccupation_Manager  1.110e+03  2.648e+02   4.192  5.7e-05 ***
doccupationoccupation_NYPD            NA         NA      NA       NA
age                          1.737e+02  9.039e+00  19.219  < 2e-16 ***
debts                        2.708e-01  7.825e-02   3.460 0.000776 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 805.1 on 107 degrees of freedom
Multiple R-squared:  0.9516,    Adjusted R-squared:  0.9489
F-statistic: 350.6 on 6 and 107 DF,  p-value: < 2.2e-16
```

Figura 16: Summary de regresión hacia atrás

```
> summary(modeloseg)

Call:
lm(formula = income ~ age + doccupation + debts, data = df)

Residuals:
     Min       1Q   Median       3Q      Max
-2602.05  -351.83    25.08   419.63  2332.49

Coefficients: (1 not defined because of singularities)
                              Estimate Std. Error t value Pr(>|t|)
(Intercept)                  2.310e+03  2.522e+02   9.163  4.0e-15 ***
age                          1.737e+02  9.039e+00  19.219  < 2e-16 ***
doccupationoccupation_Accout -4.035e+03  2.598e+02 -15.534  < 2e-16 ***
doccupationoccupation_Business -1.481e+02  2.580e+02  -0.574 0.567111
doccupationoccupation_IT     -2.791e+02  2.321e+02  -1.202 0.231949
doccupationoccupation_Manager  1.110e+03  2.648e+02   4.192  5.7e-05 ***
doccupationoccupation_NYPD            NA         NA      NA       NA
debts                        2.708e-01  7.825e-02   3.460 0.000776 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 805.1 on 107 degrees of freedom
Multiple R-squared:  0.9516,    Adjusted R-squared:  0.9489
F-statistic: 350.6 on 6 and 107 DF,  p-value: < 2.2e-16
```

Figura 17: Summary de regresión por segmentos

Se puede observar que los los tres procedimientos concuerdan en la selección de variables importantes, pues los tres modelos dejan las variables `age,doccuaption` y `debts`, concordando también con su R^2 ajustado el cual es igual a 0.9489, es decir, el modelo con las tres variables explica un 94.89 % la variabilidad de la respuesta.

f)Seleccione uno de los modelos del paso anterior y responda con argumentación la pregunta: ¿ajusta bien dicho modelo?

Como los anteriores tres modelos escogieron las mismas variables, entonces se puede decir, que es un solo modelo y para decir si se ajusta bien, se puede observar el R^2 ajustado, el cual denota la porción que el modelo explica la variabilidad en la variable respuesta, también se sabe que R^2 esta entre los números 0 y 1, donde un número cercano a 1, indica, como se mencino anteriormente que el modelo explica una gran porción de la variabilidad en la variable respuesta, y observando el summary el R^2 ajustado = 0.9489, es decir, el modelo explica un 94.89 % de la variabilidad de la respuesta, se puede decir que es un buen R^2, ya que es un número cercano a 1.

Por otro lado si se compara el R^2 ajustado del modelo completo, con el segundo modelo (forward,backward,stepwise), se puede ver que hubo una disminución del R^2 ajustado pero muy pequeña, ya que del modelo completo su R^2 es igual a 95.2 %, y del segundo modelo es de 94.89 %; sin embargo hay un gran diferencia de selección de variables, pues el primer modelo tiene 10 variables, y el segundo tiene solo 3 variables, dando a entender que las variables eliminadas estan asociadas debilmente con la respuesta, pues a pesar que el R^2 siempre se incrementa cuando se incluyen más variables en el modelo,la ecuación de OLS debe permitir ajustar de manera más precisa el conjunto de datos, en otras palabras, adicionar al modelo las variables `dstate, drace, dgender, credit_score, dloan_type, dloan_decision_type, dmarital_status`, conlleva solo a un incremento muy pequeño de R^2, y los métodos (forward,backward y stpewise) proporcionan evidencia adicional de que se deben eliminar estas variables del modelo.