UNIVERSIDAD DEL VALLE DE GUATEMALA
CC3074 - Minería de Datos
Sección 10

Regresión Logística

Hoja de trabajo 6

Cristopher Jose Rodolfo Barrios Solis - 18207

GUATEMALA, 19 de abril de 2023

1. Cree una variable dicotómica por cada una de las categorías de la variable respuesta categórica que creó en hojas anteriores. Debería tener 3 variables dicotómicas (valores 0 y 1) una que diga si la vivienda es cara o no, media o no, económica o no.

````r
#variables dicotomicas
datos$grupo <- ifelse(datos$SalePrice<178000, "3",
                      ifelse(datos$SalePrice<301000, "2",
                             ifelse(datos$SalePrice<756000,"1",NA)))
datos$grupo2 <- ifelse(datos$SalePrice<178000, "3",
                       ifelse(datos$SalePrice<301000, "2",NA))
datos$grupo3 <- ifelse(datos$SalePrice<178000, "3", NA)
datos$grupo <- as.factor(datos$grupo)
datos$grupo2 <- as.factor(datos$grupo2)
datos$grupo3 <- as.factor(datos$grupo3)
datos<-cbind(datos,dummy(datos$grupo,verbose = T),dummy(datos$grupo2,verbose = T), dummy(datos$grupo3,verbose = T))
datos <- datos[,c("LotFrontage","LotArea","GrLivArea","YearBuilt","BsmtUnfSF","TotalBsmtSF","X1stFlrSF","GarageYrBlt"
,"GarageArea","YearRemodAdd", "SalePrice", "datos1","datos2","datos3")]
datos <- na.omit(datos)
head(datos, 30)
porcentaje<-0.7
corte <- sample(nrow(datos),nrow(datos)*porcentaje)
train<-datos[corte,]
test<-datos[-corte,]
head(train)
head(test)
```
````

Se crean tres variables dicotómicas grupo, grupo2, y grupo3, las cuales tienen valores 0 o 1, dependiendo de si la vivienda es cara, intermedia, o económica.

2. Use los mismos conjuntos de entrenamiento y prueba que utilizó en las hojas anteriores.

````r
```{r}
tic("Entrenamiento modelo casas caras")
startm1 <- Sys.time()
modelo<-glm(datos1~., data = train[,c(2:10,12)],family = binomial(), maxit=100)
modelo
finalm1 <- Sys.time()
totalTm1 <- finalm1 - startm1
toc()

```
````

```
Call:  glm(formula = datos1 ~ ., family = binomial(), data = train[,
    c(2:10, 12)], maxit = 100)

Coefficients:
 (Intercept)      LotArea    GrLivArea    YearBuilt    BsmtUnfSF  TotalBsmtSF
  -3.537e+02    3.307e-05    3.274e-03    3.576e-02   -1.360e-03    4.401e-04
   X1stFlrSF  GarageYrBlt   GarageArea YearRemodAdd
   2.386e-03    6.192e-03    2.482e-03    1.284e-01

Degrees of Freedom: 787 Total (i.e. Null);  778 Residual
Null Deviance:      513.2
Residual Deviance: 201.6        AIC: 221.6
Entrenamiento modelo casas caras: 0.14 sec elapsed
```
```

```{r}
tic("Entrenamiento modelo casas intermedias")
startm2 <- Sys.time()
modelo2<-glm(datos2~., data = train[,c(2:10,13)],family = binomial(), maxit=100)
modelo2
finalm2 <- Sys.time()
totalTm2 <- finalm2 - startm2
toc()

```

```
Call:  glm(formula = datos2 ~ ., family = binomial(), data = train[,
    c(2:10, 13)], maxit = 100)

Coefficients:
 (Intercept)       LotArea     GrLivArea     YearBuilt     BsmtUnfSF    TotalBsmtSF     X1stFlrSF    GarageYrBlt
  -1.078e+02     8.968e-06     9.128e-04     2.755e-02     8.386e-04     -5.263e-04     9.131e-05     -1.167e-02
  GarageArea  YearRemodAdd
  -5.879e-04     3.737e-02

Degrees of Freedom: 787 Total (i.e. Null);  778 Residual
Null Deviance:      984.7
Residual Deviance: 758.7        AIC: 778.7
Entrenamiento modelo casas intermedias: 0.16 sec elapsed
```

```{r}
tic("Entrenamiento modelo casas baratas")
startm3 <- Sys.time()
modelo3<-glm(datos3~., data = train[,c(2:10,14)],family = binomial(), maxit=100)
modelo3
finalm3 <- Sys.time()
totalTm3 <- finalm3 - startm3
toc()
```

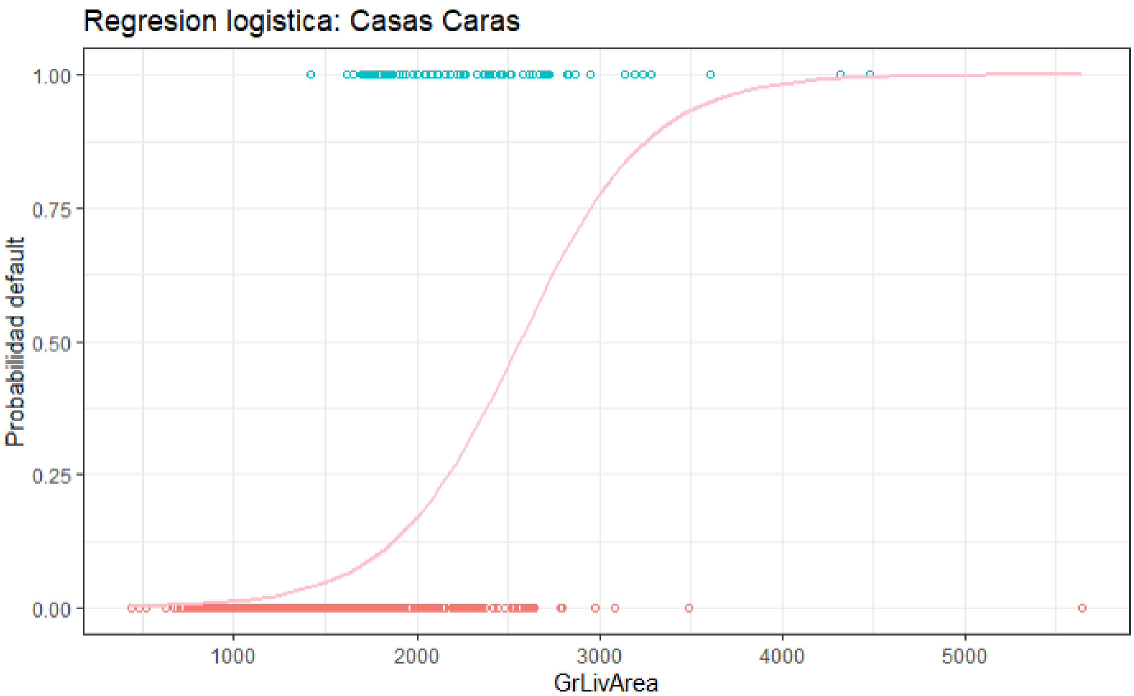
```
Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
Call:  glm(formula = datos3 ~ ., family = binomial(), data = train[,
    c(2:10, 14)], maxit = 100)

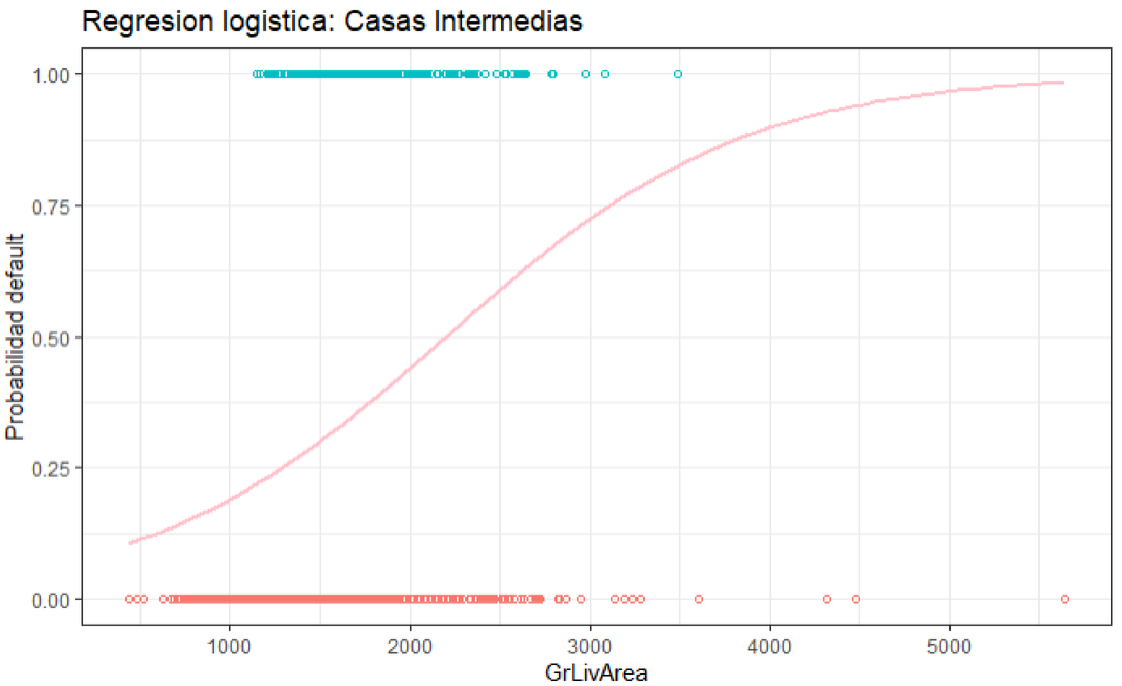
Coefficients:
 (Intercept)       LotArea     GrLivArea     YearBuilt     BsmtUnfSF    TotalBsmtSF     X1stFlrSF    GarageYrBlt
   1.877e+02    -5.634e-05    -4.056e-03    -5.325e-02    -1.615e-05     -1.774e-03     5.601e-04      3.111e-02
  GarageArea  YearRemodAdd
  -2.810e-03    -6.743e-02

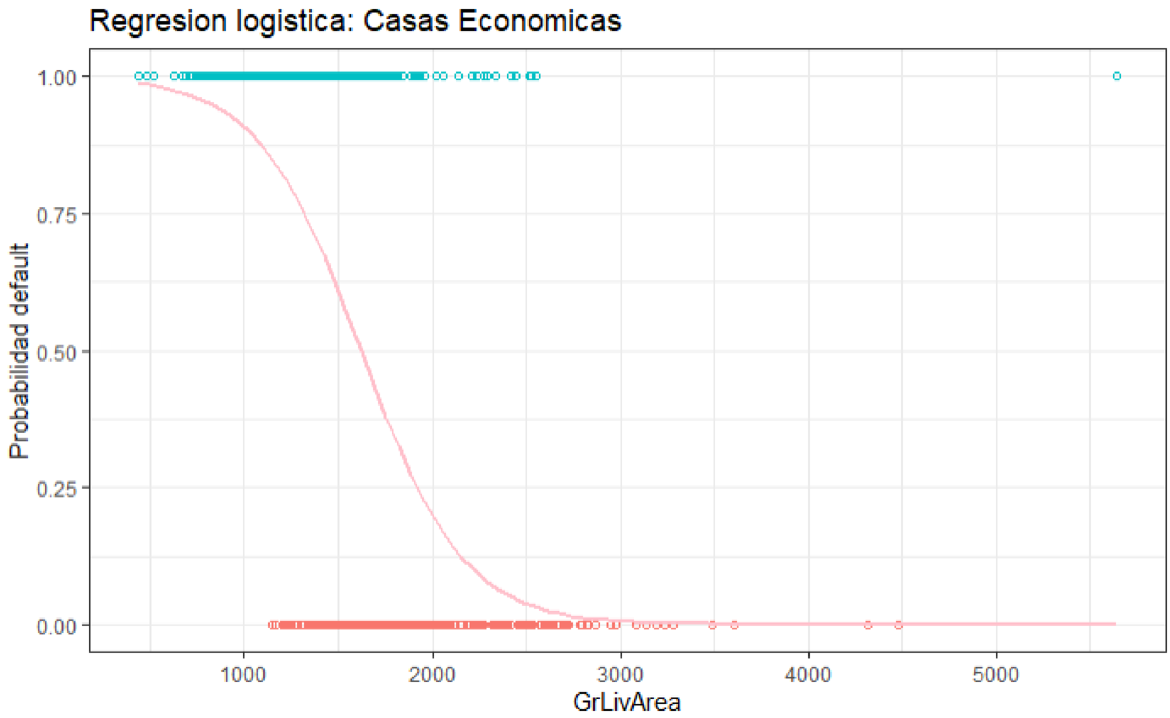
Degrees of Freedom: 787 Total (i.e. Null);  778 Residual
Null Deviance:      1071
Residual Deviance: 416.3        AIC: 436.3
Entrenamiento modelo casas baratas: 0.17 sec elapsed
```

En este caso se pueden ver los conjuntos de entrenamiento de casas caras, intermedias y baratas

3. Elabore un modelo de regresión logística para conocer si una vivienda es cara o no, utilizando el conjunto de entrenamiento y explique los resultados a los que llega. El experimento debe ser reproducible por lo que debe fijar que los conjuntos de entrenamiento y prueba sean los mismos siempre que se ejecute el código. Use validación cruzada.


Regresion logistica: Casas Caras
Probabilidad default
1.00
0.75
0.50
0.25
0.00
1000
2000
3000
4000
5000
GrLivArea

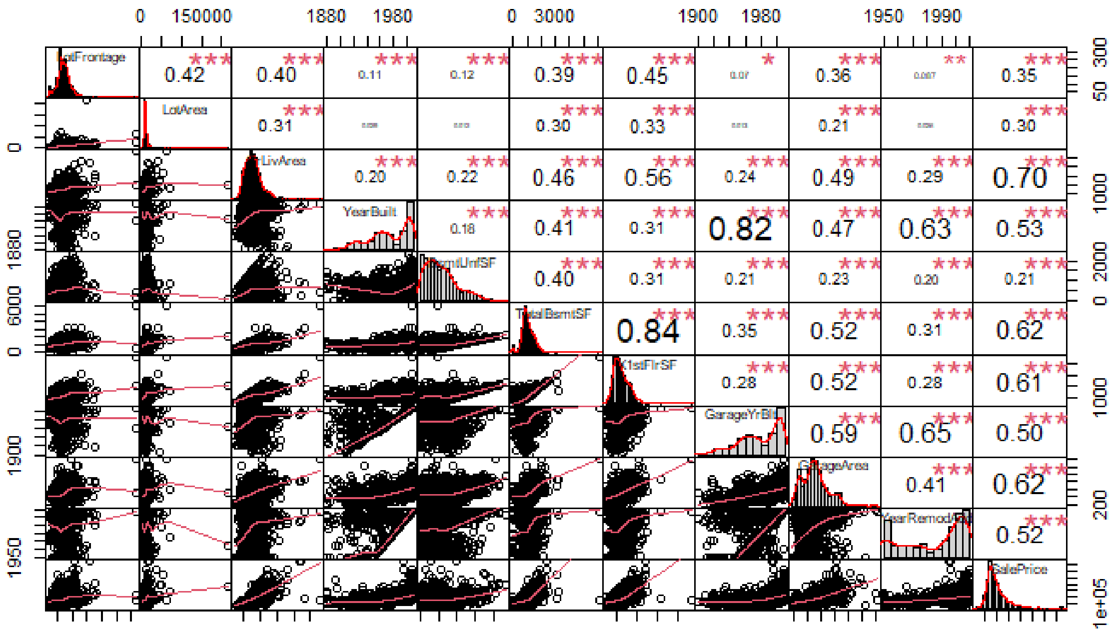

Regresion logistica: Casas Intermedias
Probabilidad default
1.00
0.75
0.50
0.25
0.00
1000
2000
3000
4000
5000
GrLivArea



Se están creando tres gráficas de regresión logística utilizando el paquete ggplot. Cada gráfica muestra la relación entre la variable GrLivArea y una variable objetivo de clasificación llamada "datos1", "datos2" y "datos3" respectivamente.

Cada gráfica tiene puntos de datos que representan la variable GrLivArea en el eje x y la variable objetivo en el eje y, con diferentes colores para indicar la clasificación. Además, cada gráfica incluye una línea de regresión logística ajustada utilizando el método glm con argumentos de familia binomial, y se utiliza el color rosa para resaltar la línea.

4. Analice el modelo. Determine si hay multicolinealidad en las variables, y cuáles son las que aportan al modelo, por su valor de significación. Haga un análisis de correlación de las variables del modelo y especifique si el modelo se adapta bien a los datos.



La observación de la tabla de correlación revela la existencia de variables que presentan altos índices de correlación, lo que indica que estas variables son relevantes para el modelo de regresión lineal y pueden ser útiles para predecir el valor de la variable dependiente. Asimismo, la presencia de estas variables con altos índices de correlación sugiere que el modelo de regresión lineal se ajusta bien a los datos asociados con estas variables.

5. Utilice el modelo con el conjunto de prueba y determine la eficiencia del algoritmo para clasificar.

```
```{r}
prediction1 <- predict(modelo, test)
prediction1
```
```

```
2 3 9 22 26 27 29 34
-8.24470545 -2.88931095 -13.50814450 -15.65852386 -1.43934844 -7.73499406 -4.35682942 -8.73369368
35 52 53 54 55 56 58 61
-1.48557076 -15.23806937 -12.69241886 -0.31937020 -11.02847296 -9.87362996 -3.56534858 -3.60206646
63 64 70 75 80 81 97 98
-3.96368549 -13.48231411 -8.54391006 -15.29568265 -12.10888394 -7.06722218 -2.47856660 -12.19921173
103 106 107 110 111 116 119 122
-6.56688398 -2.30288342 -11.59510907 -5.40246262 -8.66665119 -5.48571552 2.31659216 -15.95878152
124 130 131 138 139 140 144 146
-6.45171751 -7.59687669 -7.32253410 -3.23611552 -2.58647580 -3.50833957 -3.27486785 -3.94409347
155 159 172 173 175 177 180 198
-15.74142767 -1.88923184 -8.13264323 -5.40432027 -3.32110501 -3.62462173 -9.25074706 -0.99481662
202 210 213 216 218 227 229 232
-7.92806242 -11.29038123 -0.93142732 -7.30288754 -15.90350286 -0.85072660 -11.92223317 1.15611471
237 240 248 256 261 264 271 277
-3.99400810 -14.57147820 -12.74633841 -2.74111209 -7.96816247 -7.18928567 -0.98228984 -3.29496297
283 284 287 291 295 298 302 304
-2.42029881 -1.93032646 -7.30216027 -1.82306464 -9.82981604 -2.93828577 -0.55539863 -10.53129157
313 316 321 323 332 335 341 345
-15.11934105 -4.47027780 1.04898655 -0.09946956 -8.52518411 -3.92516387 -3.32006719 -11.17851708
353 360 366 372 373 375 391 392
-15.64819783 -2.20900803 -8.87472468 -9.58885205 -10.17651590 -3.38428884 -14.99348127 -2.29620076
397 400 403 409 410 414 415 419
-10.59536714 -2.58599617 -11.71086521 -0.39693208 0.26369689 -16.33005339 -2.82579097 -15.75446238
421 426 431 433 436 437 439 444
-2.97911977 -9.87009271 -12.69091123 -10.48091306 -4.22220249 -16.21795653 -10.80797000 -4.06888832
445 450 451 461 462 464 468 473
-3.80279967 -8.33751340 -10.49707147 -1.46313113 -7.49347851 -8.06807220 -7.55735390 -4.46153993
475 476 478 483 486 487 490 495
-1.90933413 -11.74285044 1.88007251 -7.62688246 -7.05131064 -11.04868021 -9.48133644 -10.33256200
500 503 504 511 512 513 516 524
-10.77553155 -10.60849805 -2.80597855 -5.86990292 -4.06100777 -13.15547619 2.78541982 14.89837434
525 531 533 544 552 553 554 562
-0.15951197 -4.75464768 -7.94925048 -7.56287786 -14.52245032 -1.13639209 -6.39867372 -7.64575419
564 566 569 574 575 578 580 582
-13.67053946 -15.46274400 0.57620402 -5.42864888 -7.30112748 -9.33468961 -12.07261241 2.11986465
584 585 586 589 592 603 618 625
-2.61994783 -8.80094208 2.36188310 -3.11619443 2.84786769 -3.98205626 -15.22185647 -8.21435660
626 629 631 632 634 635 640 647
-12.68885849 -7.35484630 -10.12757574 -2.79788150 -6.96821207 -8.23226214 -1.12395537 -14.23373028
656 659 660 662 665 668 670 676
-12.81194702 -14.02139253 -5.11067997 3.40419153 3.85808491 -1.98382041 -15.57681994 -9.11288559
678 684 690 695 697 702 705 709
-10.67426824 -1.08631776 -4.07103112 -13.30419135 -18.19767869 -11.06346021 -2.32219946 -4.60844851
716 730 733 734 736 741 744 754
-9.17297112 -15.25402309 -2.73625580 -9.27214783 -11.62686065 -8.95625095 -4.36755718 -1.15865506
```


```
656 659 660 662 665 668 670 676
-12.81194702 -14.02139253 -5.11067997 3.40419153 3.85808491 -1.98382041 -15.57681994 -9.11288559
678 684 690 695 697 702 705 709
-10.67426824 -1.08631776 -4.07103112 -13.30419135 -18.19767869 -11.06346021 -2.32219946 -4.60844851
716 730 733 734 736 741 744 754
-9.17297112 -15.25402309 -2.73625580 -9.27214783 -11.62686065 -8.95625095 -4.36755718 -1.15865506
```


```
756 761 772 781 789 791 797 801
-5.46382294 -13.63584996 -12.72602014 -6.43812740 -9.68990880 -5.40671429 -8.59813359 -4.45955724
802 803 815 830 831 832 840 850
-10.29342692 -4.07006460 -16.49327539 -6.10612734 -10.71045036 -6.16041020 -9.31580253 -4.91774911
851 865 871 874 876 879 882 884
-6.31054722 -3.49747710 -14.18114336 -14.00352060 0.29726870 -6.48501301 -4.43714240 -12.37026452
885 890 893 896 897 906 907 913
-12.78252221 -10.75290981 -7.18772799 -9.74648541 -16.03654351 -13.82048281 -1.45887775 -15.47210417
914 919 925 931 933 934 939 946
-10.71018006 -2.90640737 -4.19199914 -2.95433799 -0.53313618 -3.41265610 -0.36626901 -6.74684759
950 965 973 974 988 990 991 993
-3.25941589 -2.55284566 -11.23727993 -3.26067543 2.02502080 -3.80368924 0.52785281 -5.29494280
1008 1009 1013 1016 1017 1020 1028 1041
-12.13652452 -2.24561430 -9.36229687 -3.65438440 -3.96148010 -3.59040693 0.56829819 -2.77747767
1051 1052 1053 1054 1055 1064 1070 1075
-4.18303265 -4.03459616 -6.59552562 -5.24781686 -1.35219788 -11.26737713 -8.13480613 -3.69384537
1077 1080 1083 1086 1088 1090 1093 1095
-7.64174280 -7.53844558 -4.07302559 -6.79778308 -1.02736055 -3.27602798 -12.43583510 -8.71367630
1106 1107 1126 1127 1128 1131 1135 1136
0.57349588 -4.04678345 -13.68135861 -2.43599216 -1.17926342 -10.88469116 -5.42448552 -17.10420862
1140 1146 1148 1158 1159 1160 1164 1167
-14.22462378 -14.91655706 -13.94678112 -2.42234891 -1.65569911 -7.55018903 -9.14170826 -1.26178127
1168 1170 1171 1173 1175 1176 1197 1203
-3.45016398 5.89049016 -9.60076820 -5.08284431 -10.48203189 2.32033194 -2.41305377 -9.79766611
1205 1215 1221 1222 1226 1229 1230 1233
-9.44335874 -12.44960022 -12.52448749 -10.31936052 -12.49127698 1.49943891 -10.03136792 -11.00131559
1240 1241 1247 1250 1255 1257 1265 1268
-2.81041957 -2.96517123 -3.95807381 -14.87982108 -4.52334555 -0.55453260 -5.09991799 0.43941715
1283 1289 1290 1293 1295 1307 1314 1320
-5.52277825 -1.91679887 -1.54533478 -9.94666027 -10.07412014 -4.05561799 -0.05418656 -14.57113907
1323 1330 1336 1337 1346 1348 1360 1373
-3.49849488 -5.05851256 -7.72139152 -8.11617373 -16.37455999 -0.93523550 1.70112863 -1.68153848
1375 1376 1377 1378 1380 1383 1386 1391
-1.48535366 -2.13213385 -17.42367112 -10.49407085 -4.44496350 -13.00763273 -8.57941439 -2.15030781
1394 1400 1402 1403 1404 1409 1411 1412
-5.55027740 -9.12352960 -3.47426526 -3.89686327 -1.17454110 -15.55245463 -2.48748145 -5.90180869
1413 1415 1416 1423 1427 1437 1438 1441
-14.27762210 -7.01642161 -1.02643744 -6.27382132 -2.16678414 -11.30803323 1.74371622 -3.26101990
1446 1449 1452
-12.17186853 -9.32141039 -1.55985134
```

```{r}
prediction2 <- predict(modelo2, test)
prediction2

```

```
            2             3             9            22            26            27            29 
-1.9057958913  0.1434361470 -2.4589050959 -3.0508052548  0.7880919328 -2.3859984235 -1.1214078418 
           34            35            52            53            54            55            56 
-2.3460652672 -0.1917285718 -3.6480195564 -3.0923245226 -1.2399254807 -2.8713058616 -1.9839702314 
           58            61            63            64            70            75            80 
 0.6177350273 -0.5811735745  0.5914038655 -3.5795188833 -1.9313250618 -2.8693875831 -2.9197592230 
           81            97            98           103           106           107           110 
-1.2039164543 -0.4073621902 -2.4766797588 -1.0396212501  0.8588948183 -2.9146851019 -0.8887010698 
          111           116           119           122           124           130           131 
-1.4245682822 -0.2938994981  0.3306024900 -3.3652700066 -0.5714195306 -2.2420085700 -1.2586565390 
          138           139           140           144           146           155           159 
 0.3136465899 -0.0716958676 -0.2861324614 -0.1813675216  0.7537836697 -2.8776357931  0.6707201081 
          172           173           175           177           180           198           202 
-1.9758027792 -0.7786343642 -1.4683994742 -0.4533611148 -1.3543129344 -0.8329730912 -1.8454061372 
          210           213           216           218           227           229           232 
-2.6463533476  0.2607849084 -1.0164080205 -3.0531766999  0.4274887740 -2.7502144425  0.7112802715 
          237           240           248           256           261           264           271 
 0.5578598969 -2.5430945093 -2.0901638583  0.9757753047 -3.3224245529 -2.3364615994  0.7238583882 
          277           283           284           287           291           295           298 
 0.6742893295 -0.2372144455  0.6694392327 -1.3053401846  0.9592920871 -2.9315591057 -0.0500662191 
          302           304           313           316           321           323           332 
 0.8243084200 -2.6201095213 -3.3773219524  0.0213393505  1.5605841928 -0.4066935722 -1.6774538865 
          335           341           345           353           360           366           372 
-0.4061502210  0.8300309960 -1.8082553953 -3.3798330960 -0.2386289396 -2.2439937768 -2.5357805136 
          373           375           391           392           397           400           403 
-1.8639853042  0.8326079263 -4.7272164092  0.1221071822 -2.6902683419  0.0523402544 -1.5142269718 
          409           410           414           415           419           421           426 
 1.0174118303  0.3730802043 -3.0963805382 -0.0924618593 -3.0950683805 -1.1331412623 -1.0275576770 
          431           433           436           437           439           444           445 
-2.1976394180 -1.9105342109 -0.4954714329 -4.1697221256 -3.1302534987  0.3707678792 -0.2039347927 
          450           451           461           462           464           468           473 
-1.4809237325 -1.2698489861  0.5877069762 -1.9278467358 -1.0684615686 -1.0052259055 -0.4318221792 
          475           476           478           483           486           487           490 
-0.4078422284 -3.1822727547  2.0235004411 -0.7371842646 -0.9297408582 -2.6473923141 -1.4431649719 
          495           500           503           504           511           512           513 
-1.5779904680 -1.7455019924 -2.4048384155 -0.9608128151 -1.1138799398  0.6020422733 -3.3218032236 
          516           524           525           531           533           544           552 
 0.2255682683  2.6049288216  0.2980280838 -0.8326835984 -1.2129642319 -1.1809241100 -2.7264649827 
          553           554           562           564           566           569           574 
 0.8587482346 -1.6972021483 -1.9721749029 -2.9269437405 -2.7945350968 -0.1259420702  0.0654759530 
          575           578           580           582           584           585           586 
-0.7449040583 -2.3692246105 -1.7697043904  1.1518791857 -1.4302034144 -0.7873084222 -0.0004466716 
          589           592           603           618           625           626           629 
-0.9036834201  0.4364793978 -0.4634884051 -2.9777774329 -1.2190452051 -2.0079322153 -1.3542261758 
          631           632           634           635           640           647           656 
-2.3561967488  0.7329481296 -1.3027876181 -2.1211149082 -0.2300620296 -3.0770512780 -1.7602911915 
          659           660           662           665           668           670           676 
-2.3168238310 -0.7112790115  0.4117286538 -0.1493262193 -0.5485519427 -3.0321329655 -1.2946159503 
          678           684           690           695           697           702           705 
-1.2919376033 -0.1954409760  0.5926421557 -3.3500141452 -4.1797666990 -1.6273390245  0.0377820800 
          709           716           730           733           734           736           741 
 0.5566562825 -1.1713222192 -3.7404963569  0.5383927290 -1.6842238279 -2.2203082479 -1.3043982536 
          744           754           756           761           772           781           789 
```


```
-2.3561967488  0.7329481296 -1.3027876181 -2.1211149082 -0.2300620296 -3.0770512780 -1.7602911915 
          659           660           662           665           668           670           676 
-2.3168238310 -0.7112790115  0.4117286538 -0.1493262193 -0.5485519427 -3.0321329655 -1.2946159503 
          678           684           690           695           697           702           705 
-1.2919376033 -0.1954409760  0.5926421557 -3.3500141452 -4.1797666990 -1.6273390245  0.0377820800 
          709           716           730           733           734           736           741 
 0.5566562825 -1.1713222192 -3.7404963569  0.5383927290 -1.6842238279 -2.2203082479 -1.3043982536 
          744           754           756           761           772           781           789 
```


```
-1.2128384231  0.8887599926 -0.2489777122 -3.5104451072 -3.2619523170 -0.0347717822 -0.9401788187 
          791           797           801           802           803           815           830 
 0.3079503238 -1.2498403894 -0.2821293593 -2.2768483498 -0.1064912415 -3.5596701868  0.0734084302 
          831           832           840           850           851           865           871 
-2.7765400119  0.0581168815 -1.2541445079 -0.1467251604 -0.7684231173  0.6863656132 -2.2530705720 
          874           876           879           882           884           885           890 
-2.8259049405  1.6521009182 -1.8749448290 -0.6112721723 -3.3745651935 -2.3350642574 -2.8526513769 
          893           896           897           906           907           913           914 
-0.9925230224 -1.8381599739 -3.2699921042 -2.9525898728  0.7441221402 -3.7046042234 -2.4254193877 
          919           925           931           933           934           939           946 
-0.3382315772 -1.8120053242  0.7259290330  1.1839892935  0.6271819350  0.5663145401 -2.9795818976 
          950           965           973           974           988           990           991 
-0.8926465360  0.1968162018 -1.2788229644  0.8274651439  0.1254547435  0.4968083278  0.0134798714 
          993          1008          1009          1013          1016          1017          1020 
-0.7970354109 -2.2710702876  1.0021044476 -1.0260455924 -0.1589778164 -0.2708610455  0.6740487557 
         1028          1041          1051          1052          1053          1054          1055 
-0.3757012469 -0.6823256235  0.6263578221  0.6620053963 -1.1188899178 -0.4388559535  0.0454708034 
         1064          1070          1075          1077          1080          1083          1086 
-2.9038465452 -1.6788635733  0.2582818475 -1.9491654356 -0.8364612442  0.5111166578 -1.2949426284 
         1088          1090          1093          1095          1106          1107          1126 
 0.7200155886 -0.5418093887 -2.9198448313 -1.2886624222  0.2463705201 -0.9963592703 -2.9161201250 
         1127          1128          1131          1135          1136          1140          1146 
 0.7232740310 -0.3874683093 -3.4252768058  0.1940420988 -3.0885249344 -4.3263482501 -2.5133957006 
         1148          1158          1159          1160          1164          1167          1168 
-2.6244191432 -0.2377704843  0.8127460738 -1.1955045640 -2.4221542249  1.0090501173 -0.2116198755 
         1170          1171          1173          1175          1176          1197          1203 
 1.2503192932 -1.6040955983  0.1340315920 -2.1729862562  1.1206443436  0.8281861809 -1.4818345915 
         1205          1215          1221          1222          1226          1229          1230 
-2.0685520957 -2.7584013644 -2.8651868067 -2.2128524833 -2.6811059620 -0.4494686374 -2.2907089328 
         1233          1240          1241          1247          1250          1255          1257 
-2.2623557668  0.4305039651  0.1384087066  0.4529701610 -3.3074035795  0.5552007190 -0.3686486440 
         1265          1268          1283          1289          1290          1293          1295 
-0.0595889817  1.4984840896 -0.7542571449 -0.3637638731  0.9307156751 -2.9860984128 -1.6617485866 
         1307          1314          1320          1323          1330          1336          1337 
 0.6035041170  1.2739068718 -2.9790979430 -0.4296498714  0.2876080203 -1.6073877625 -0.7568988490 
         1346          1348          1360          1373          1375          1376          1377 
-4.3952987324  1.0480004546  0.0561362455 -0.1027475469  0.7279701028  0.8302554802 -3.5327037203 
         1378          1380          1383          1386          1391          1394          1400 
-3.2588756407  0.3738870230 -3.6171557403 -1.5953034467 -0.6876504036 -2.2276260792 -1.2634471529 
         1402          1403          1404          1409          1411          1412          1413 
-0.0806590982  0.4072019144 -0.0636931170 -3.9706413035  0.0163527041 -0.1739138751 -3.1826763548 
         1415          1416          1423          1427          1437          1438          1441 
-0.5525589924 -0.0876095727 -0.7910660101 -0.2180930804 -2.4727764938  0.2207957611 -1.2217924550 
         1446          1449          1452 
-2.9526830142 -1.8242885496  0.8436478138 
```

````
```{r}
prediction3 <- predict(modelo3, test)
prediction3
```
````

```
          2          3          9         22         26         27         29         34         35
 2.25227063 -2.27601583 3.50313959 7.29193946 -3.73494459 4.58579517 -0.29995540 1.98381465 -2.06378503
         52         53         54         55         56         58         61         63         64
 7.40162295 5.67776967 -5.16329688 5.39448819 2.09933637 -2.04509459 -0.16587591 -0.91062773 6.46722022
         70         75         80         81         97         98        103        106        107
-0.82209120 4.90996822 6.38744828 -0.32810976 -1.59918746 4.83071787 3.55797163 -3.46886653 7.43706950
        110        111        116        119        122        124        130        131        138
-1.28087873 1.43249384 0.45862124 -7.96754606 8.04127818 0.95118982 3.28324199 -0.24672711 -3.40840857
        139        140        144        146        155        159        172        173        175
-2.48798890 -1.61304339 -1.42643821 -1.23685102 5.80848874 -3.55936366 1.18810448 -0.42390622 -1.20589480
        177        180        198        202        210        213        216        218        227
-2.18096643 3.31865370 -6.86086055 2.24768232 4.17506147 -3.09900760 1.76208228 6.78056331 -5.23317945
        229        232        237        240        248        256        261        264        271
 5.53320382 -7.26130946 -1.12348755 5.29534570 3.55299228 -3.49203099 2.07861013 3.36361462 -4.72092662
        277        283        284        287        291        295        298        302        304
-2.05182673 -1.04757551 -3.01737074 0.87739201 -3.86130582 2.82422665 -2.34107216 -5.23394806 4.37796554
        313        316        321        323        332        335        341        345        353
 6.21969381 -0.16983694 -7.28836006 -5.26214153 3.16579918 -0.85581821 -2.51248032 4.48670516 6.94090489
        360        366        372        373        375        391        392        397        400
-2.69407910 3.74519670 4.56976669 4.92932893 -2.28850463 8.51297333 -2.75907252 4.69880071 -1.27685700
        403        409        410        414        415        419        421        426        431
 5.32019407 -5.51738130 -4.70828857 6.66413337 -3.14412817 6.63235776 -0.89144246 3.08894180 5.76084178
        433        436        437        439        444        445        450        451        461
 3.37419144 -0.88950951 9.14353795 8.33923311 -0.67412249 -1.66117162 3.85588256 4.48913469 -3.08014547
        462        464        468        473        475        476        478        483        486
 3.32541621 1.66265721 1.33347217 1.74147288 -1.54159862 5.58349539 -8.08291985 1.01618539 2.06960883
        487        490        495        500        503        504        511        512        513
 4.04093104 4.90871626 5.70248695 4.45865865 3.50971139 -1.50092286 1.49020403 -0.86993073 5.71779155
        516        524        525        531        533        544        552        553        554
-6.06159952 -19.63502799 -5.53417426 -0.53999862 5.08992425 3.54637035 6.39483400 -4.18717238 5.27419232
        562        564        566        569        574        575        578        580        582
 1.82217669 4.36893644 5.35436546 -4.51783892 0.85135267 2.65900561 2.70415230 2.46556687 -7.57685422
        584        585        586        589        592        603        618        625        626
-1.93611361 1.93913250 -5.45137796 -1.28717808 -6.65627483 -1.54070678 6.67146327 1.15315889 4.29585370
        629        631        632        634        635        640        647        656        659
-0.40760835 4.81929370 -2.18065279 2.45369108 2.80588521 -2.17175024 7.81975365 5.27705023 4.16496423
        660        662        665        668        670        676        678        684        690
-0.17336763 -8.11180793 -7.30866904 -1.58632345 6.94935601 2.30488574 4.54247765 -2.91890330 -0.84278513
        695        697        702        705        709        716        730        733        734
 5.94766563 9.55111822 3.19043202 -1.82379379 -0.46907666 2.38134247 7.08940939 -3.11179122 3.75907290
        736        741        744        754        756        761        772        781        789
 2.76287339 3.04506854 2.41863785 -4.67270070 0.41255867 7.87492504 5.61906187 0.80355642 3.71896448
        791        797        801        802        803        815        830        831        832
 0.72879490 3.03192974 -1.15252362 4.57472964 -0.26142604 7.76360796 0.90771745 3.14398905 0.96517678
        840        850        851        865        871        874        876        879        882
 2.97995151 -0.30914266 2.42043348 -1.29774813 6.03173130 5.55210905 -6.13000177 2.30970726 -0.83389981
```

```
        562        564        566        569        574        575        578        580        582
 1.82217669 4.36893644 5.35436546 -4.51783892 0.85135267 2.65900561 2.70415230 2.46556687 -7.57685422
        584        585        586        589        592        603        618        625        626
-1.93611361 1.93913250 -5.45137796 -1.28717808 -6.65627483 -1.54070678 6.67146327 1.15315889 4.29585370
        629        631        632        634        635        640        647        656        659
-0.40760835 4.81929370 -2.18065279 2.45369108 2.80588521 -2.17175024 7.81975365 5.27705023 4.16496423
        660        662        665        668        670        676        678        684        690
-0.17336763 -8.11180793 -7.30866904 -1.58632345 6.94935601 2.30488574 4.54247765 -2.91890330 -0.84278513
        695        697        702        705        709        716        730        733        734
 5.94766563 9.55111822 3.19043202 -1.82379379 -0.46907666 2.38134247 7.08940939 -3.11179122 3.75907290
        736        741        744        754        756        761        772        781        789
 2.76287339 3.04506854 2.41863785 -4.67270070 0.41255867 7.87492504 5.61906187 0.80355642 3.71896448
        791        797        801        802        803        815        830        831        832
 0.72879490 3.03192974 -1.15252362 4.57472964 -0.26142604 7.76360796 0.90771745 3.14398905 0.96517678
        840        850        851        865        871        874        876        879        882
 2.97995151 -0.30914266 2.42043348 -1.29774813 6.03173130 5.55210905 -6.13000177 2.30970726 -0.83389981
        884        885        890        893        896        897        906        907        913
 5.03532489 5.62663128 2.75801344 2.37327522 2.02648536 6.94442992 5.76953583 -3.76092077 7.09843531
        914        919        925        931        933        934        939        946        950
 0.94430252 -3.76445034 -0.79889059 -2.00166801 -4.88759959 -1.77555019 -4.54006404 2.94315393 -0.51077667
        965        973        974        988        990        991        993       1008       1009
-2.39922143 4.43300447 -1.65615815 -4.73668898 -1.46435155 -5.58025193 -0.32366945 5.25754809 -3.10171189
       1013       1016       1017       1020       1028       1041       1051       1052       1053
 2.79680620 -1.19465059 -1.00215850 -0.96552367 -3.51829768 -0.87596022 -0.65001077 -0.85818413 -0.02227627
       1054       1055       1064       1070       1075       1077       1080       1083       1086
 0.39751075 -3.22287228 5.78815976 3.84740096 -1.30383829 2.23909302 2.63099663 -1.27396135 2.06655171
       1088       1090       1093       1095       1106       1107       1126       1127       1128
-4.70281860 -0.16382457 3.33163658 3.42914611 -6.02702129 0.15734502 5.51772203 -1.99183202 -2.42485951
       1131       1135       1136       1140       1146       1148       1158       1159       1160
 3.85761647 -0.54341187 7.58136388 7.76132803 4.89875666 4.81730961 -1.04408245 -3.42836017 0.20200038
       1164       1167       1168       1170       1171       1173       1175       1176       1197
 2.99399997 -3.85038712 -1.29114011 -12.42884347 3.22667609 0.16488075 0.97120583 -6.81143923 -3.23026221
       1203       1205       1215       1221       1222       1226       1229       1230       1233
 4.04249355 3.23596778 5.44119223 5.76075571 4.16193286 5.99773705 -4.54713873 3.41105901 5.86274666
       1240       1241       1247       1250       1255       1257       1265       1268       1283
-1.60485837 -1.91258332 -1.33899879 6.87643737 -0.87774262 -4.09899789 -0.14652017 -5.64071436 1.21844875
       1289       1290       1293       1295       1307       1314       1320       1323       1330
-1.59803056 -3.99087687 4.02090427 3.69170696 -0.87911410 -6.57931953 6.58605177 -1.38830580 -0.79808259
       1336       1337       1346       1348       1360       1373       1375       1376       1377
 1.26832094 0.31327192 9.85510906 -4.18444154 -5.38386692 -3.26986863 -4.28904897 -2.98801165 8.42221773
       1378       1380       1383       1386       1391       1394       1400       1402       1403
 3.53749370 0.42549613 6.20134036 3.21701867 -1.48018930 3.31180524 2.35563248 -0.77036414 -1.01310918
       1404       1409       1411       1412       1413       1415       1416       1423       1427
-2.77520676 7.55063393 -2.55947102 0.38519524 8.08762917 0.09231155 -2.04532992 2.42086946 -3.07859559
       1437       1438       1441       1446       1449       1452
 4.73412423 -5.19215815 -0.22545632 6.59963358 4.06956570 -3.40718919
```

Se puede observar el resultado de las variables de respuesta a través de la predicción realizada.

6. Explique si hay sobreajuste (overfitting) o no (recuerde usar para esto los errores del conjunto de prueba y de entrenamiento). Muestre las curvas de aprendizaje usando los errores de los conjuntos de entrenamiento y prueba.

Cuando se presenta el fenómeno del sobreajuste (overfitting), es posible que esto se deba a que ciertas variables, en particular aquellas que tienen una correlación baja entre sí, estén influyendo demasiado en el modelo.

7. Haga otros dos modelos cambiando las variables predictoras de acuerdo con la significación de los coeficientes en el primer modelo. Explique por qué seleccionó las variables que uso para cada modelo.

8. Haga un análisis de la eficiencia del algoritmo usando una matriz de confusión. Tenga en cuenta la efectividad, donde el algoritmo se equivocó más, donde se equivocó menos y la importancia que tienen los errores, el tiempo y la memoria consumida. Para esto último puede usar "profvis" si trabaja con R y "cProfile" en Python.

````
```{r}
print("Tiempo de entrenamiento modelo casas caras: ")
totalTm1
```

[1] "Tiempo de entrenamiento modelo casas caras: "
Time difference of 0.1425271 secs
````

````
```{r}
print("Tiempo de entrenamiento modelo casas intermedias: ")
totalTm2
```

[1] "Tiempo de entrenamiento modelo casas intermedias: "
Time difference of 0.1168048 secs
````

```{r}
print("Tiempo de entrenamiento modelo casas baratas: ")
totalTm3
```

```
[1] "Tiempo de entrenamiento modelo casas baratas: "
Time difference of 0.1213999 secs
```

En cuanto a los resultados, se puede observar que el tiempo de entrenamiento de los modelos es bastante rápido, con un promedio de alrededor de 0.12 a 0.14 segundos por modelo. Esto indica que los modelos son bastante eficientes en términos de tiempo.

En cuanto a la evaluación de la efectividad de los modelos, esto se realiza utilizando una matriz de confusión.

9. Determine cual de todos los modelos es mejor, puede usar AIC y BIC para esto, además de los parámetros de la matriz de confusión y los del profiler.

```{r}
AICmodelo <- AIC(modelo)
BICmodelo <- BIC(modelo)

print("Modelo 1 AIC: ")
AICmodelo
print("Modelo 1 BIC: ")
BICmodelo
```

```
[1] "Modelo 1 AIC: "
[1] 221.5876
[1] "Modelo 1 BIC: "
[1] 268.2826
```

```{r}
AICmodelo2 <- AIC(modelo2)
BICmodelo2 <- BIC(modelo2)

print("Modelo 2 AIC: ")
AICmodelo2
print("Modelo 2 BIC: ")
BICmodelo2
```

```
[1] "Modelo 2 AIC: "
[1] 778.6515
[1] "Modelo 2 BIC: "
[1] 825.3465
```

```{r}
AICmodelo3 <- AIC(modelo3)
BICmodelo3 <- BIC(modelo3)

print("Modelo 3 AIC: ")
AICmodelo3
print("Modelo 3 BIC: ")
BICmodelo3
```

```
[1] "Modelo 3 AIC: "
[1] 436.2621
[1] "Modelo 3 BIC: "
[1] 482.9571
```

Después de realizar el análisis utilizando AIC y BIC, se puede concluir que el Modelo 1 es el mejor modelo ya que tiene el menor valor de AIC y BIC. En este caso se recomienda utilizar el Modelo 1 para predecir y clasificar los datos.

10. Haga un modelo de árbol de decisión, uno de Random Forest y uno de Naive Bayes usando la misma variable respuesta y los mismos predictores que el mejor de los modelos de Regresión Logística.

11. Compare la eficiencia de los 3 modelos que creó en el punto anterior y el mejor de los de regresión logística ¿Cuál se demoró más en procesar?¿Cuál se equivocó más?¿Cuál se equivocó menos?¿por qué?