

¿Cómo crear un mapa de probabilidades para predecir el departamento de origen de una familia?

*Brahian Cano Urrego
Yeison Yovany Ocampo
Daniel Alexander Cano
Sebastian Pino*

4 de abril de 2018

Introducción

En este documento se presentara una aplicación de la regresión multinomial, selección de variables y filtrado de una base de datos con el objetivo de predecir el departamento de proveniencia de una familia con respecto a ciertas características de las mismas.

Usuario

Hemos definido nuestro usuario como un estudiante practicante de ingeniería y desea conocer un poco como realizar un modelamiento multinomial y además asociarlo a un ejemplo cotidiano de tratar de predecir la procedencia de una familia dado cierto conjunto de características de esa familia, además de mostrar un poco el proceso de selección de variables adecuadas y una correcta obtención de los datos, esta persona tiene una poco o inexistente dominio del programa R y necesita tener un control de las características que él desea incorporar para encontrar una mayor probabilidad de encontrarlo ya sea en un departamento o en un región del país. Además permite interactuar de una manera didactica y esta persona lo puede exponer o interpretar de una manera fácil y rápida.

Filtrado

Los datos aqui propuestos estaran disponibles en este [link](#) el cual nos llevará a las bases de datos del DANE(departamento Administrativo Nacional de Estadísticas) sobre calidad de vida; con fines prácticos se considera sólo la base calidad de vida,la cual contiene informacion sobre los hogares y sobre cada individuo que lo compone, de acuerdo a unas especificaciones de la encuesta; esto se vera reflejado en los directorios. Estas bases luego de descargadas se leen de la siguiente manera:

```
#asignaremos al objeto "datos" el data frame correspondiente a  
#caracteristicas y composicion del hogar  
  
datos <- read.csv("Caracteristicas y composicion del hogar.csv")
```

mostraremos una breve descripción de la base de datos:

```
#para la base datos datos  
dimensiones <- dim(datos)  
names(dimensiones)<- c("Observaciones","Variables")  
dimensiones
```

Observaciones	Variables
40359	39

Debido a que nuestro interés es predecir el departamento de origen sobre los hogares, esta variable sera fundamental para nuestro análisis, por lo tanto no nos interesara los registros que no la posean.

#filtraremos por los registros que no esten vacios en la variable departamento (P6076S1)

```
datos<-datos[!is.na(datos$P6076S1),]
```

#veremos si hay mas problemas en la base de datos

```
apply(datos[,c(21,24,27)],2,table)
```

\$P6081S1

1	2	3	4	5	6	7	8	Inf
1099	167	36	29	16	6	1	2	8049

\$P6083S1

1	2	3	4	5	6	7	9	Inf
963	1231	108	52	48	13	7	3	6980

\$P5667

	010_01	010_02	030_02	040_02	040_03	040_04	040_08	200_01	200_02
9099	1	2	8	1	1	4	1	17	1
200_03	210_01	220_04	280_01	281_03	282_01	290_01	290_02	290_03	290_08
6	1	2	3	2	9	3	3	3	1
292_01	310_02	340_01	350_03	350_05	460_01	500_01	500_02	500_03	560_01
1	1	15	1	2	1	24	15	15	15
560_02	565_01	565_03	565_04	650_01	660_02	720_01	720_02	720_06	730_05
1	2	3	2	21	1	5	1	4	14
730_06	750_01	780_01	800_01	800_02	855_01	870_01	950_01	999_01	
1	6	2	27	3	1	1	1	51	

se observan gran cantidad de valores registrados como infinitos en las primeras dos variables y en la otra una considerable cantidad de NA's, así que se tomó la decisión de omitirlas en el proceso que consideramos a continuación

```
datos<- datos[-c(21,24,27)]
```

ya que los valores resultantes como infinitos son un gran problema a la hora de realizar estimaciones sobre un modelo, ya que todos los cálculos aritméticos se verían altamente afectados y sus valores darían como resultados valores infinitos, por lo tanto procedemos a eliminar aquellas observaciones para nuestro propósito.

#haciendo un analisis con la función "apply(datos,2,table)" nos

#damos cuenta las variables que poseen infinitos y procedemos a eliminar esos registros

```
datos<-datos[which(is.finite(datos$P6071)
                    & is.finite(datos$P6071S1)
                    & is.finite(datos$P6088)
                    & is.finite(datos$P1903)
                    & is.finite(datos$P6087)), ]
```

Luego de esto tenemos una base "limpia" con la cual poder hacer modelamiento sin problemas y tener unos resultados más consistentes.

Por último eliminaremos aquellas variables que no se consideraron importantes para nuestra meta que es el modelo de los departamentos.

```
datos<- datos[, -c(1,2,3,4,5,6,11,13,15,17)]
```

Modelamiento

El primero paso será descargar e instalar los paquetes necesarios:

```
require(nnet)
library ( MASS )
```

Para no incurrir en sesgo recodificaremos las variables para que sean categóricas o numéricas dependiendo su respectivo caso, ya que las variables categóricas serán tratadas diferente por el modelo, usando variables indicadoras para cada una de las categorías.

```
#como la predominancia son las variables categoricas,
#es mejor volver categorico todo y luego revertir lo numerico
```

```
for(i in 1:24){
  datos[,i]<- as.factor(datos[,i])
}
```

```
datos$P6040<- as.integer(datos$P6040)
datos$P767<- as.integer(datos$P767)
```

Ya tenemos todos los elementos listo para la elaboración de nuestro modelo.

una forma sencilla de obtener la expresión para un modelo saturado se consigue de la siguiente manera:

```
paste0(names(datos)[-7],collapse = "+")
#el "-7" es para quitar la variable respuesta de la ecuacion
```

Ya con esta fórmula podremos incluirla en nuestro modelado

```
#creamos un modelo vacio para poder hacer una seleccion de variables hacia adelante
mod<-multinom(P6076S1~1,data=datos)
```

```
# weights: 64 (31 variable)
initial value 15044.759554
iter 10 value 13520.659357
iter 20 value 12969.396722
iter 30 value 12912.976053
iter 40 value 12899.899130
final value 12899.830379
converged
```

```
#la seleccion hacia adelante es favorable en este caso debido a la cantidad de variable
```

```
#que abra por las clases de las variables categoricas
```

```
stepAIC(mod,direction = "forward",
  scope=list(upper=~P6020+P6040+P6051+P5502+P6071S1+P767+P6077
    +P6096+P6081+P6087+P6083+P6088+P6080+P1895+P1896
    +P1897+P1898+P1899+P1901+P1902+P1903+P1904+P1905+LLAVEHOG+FEX_C
    ,lower=~1))
```

```
# el modelo final es:
```

```
modelo.final<-multinom(formula = P6076S1 ~ P6077 + P6080 + FEX_C + P6096 +
  P767 + P6081 + P5502, data = datos)
```

```
# weights: 640 (589 variable)
initial value 15044.759554
iter 10 value 14432.650918
```

```

iter 20 value 14211.272801
iter 30 value 14155.541240
iter 40 value 13785.615477
iter 50 value 13244.905393
iter 60 value 12783.392736
iter 70 value 12484.391700
iter 80 value 11903.911139
iter 90 value 11669.616685
iter 100 value 11590.346731
final value 11590.346731
stopped after 100 iterations

```

Tasa de buena clasificación

la sensibilidad del modelo se verá como la tasa de bien clasificados que la podemos definir así:

```

#definicion de variables auxiliares para poder comparar
obs<-as.vector(datos$P6076S1)
pre<-as.vector(predict(modelo.final,type = "class"))

# Ciclo para tasa de bien clasificados
cont<-0
for (i in 1:4341){
  if(pre[i]==obs[i]){
    cont<-cont+1
  }
}
cont/4341

```

```
[1] 0.1988021
```

la cual nos dice que el modelo tiene una capacidad predictiva del 20% aproximadamente la cual es considerable dado el caso de tener 32 opciones en este caso (departamentos) a las cuales asignar una familia.

Método alternativo

Dada la complejidad de predecir dentro de una de las 32 posibles categorías (departamentos) se ofrece una alternativa al tratar de reducir el número de categorías, lo cual permitirá tener una tasa de clasificación correcta mayor y podremos predecir de una mejor manera.

Vamos a considerar la función `Recode()` del paquete `car` la cual nos permite recodificar la variable que contiene los departamentos y organizarlos según la región a la que pertenecen, para observar cuál es la composición de las regiones de Colombia observe **Regiones**.

```

library(car)

#lectura de base de datos completa
datos1<- read.csv("baseUnida.csv")

#eliminar los NA's en departamento
datos2<-datos1[which(!is.na(datos1$P6076S1)),]

# Creamos la base de datos empírica
datos3<- data.frame(P1070=datos2$P1070,CLASE=datos2$CLASE,P5010=datos2$P5010,CANT_PERSONAS_HOGAR=datos2$
PERCAPITA=datos2$PERCAPITA,P1894=datos2$P1894,P6020=datos2$P6020,P6040=datos2$P6040

```

```

P767=datos2$P767,P6080=datos2$P6080,P6240=datos2$P6240,P6081=datos2$P6081,
P6077=datos2$P6077,P6096=datos2$P6096,P6076S1=datos2$P6076S1,P4567=datos2$P4567,P85
P5661S4=datos2$P5661S4)

# Libre de NA's
datos3<-na.omit(datos3)

datos3<-datos3[which(is.finite(datos3$P5502)
&is.finite(datos3$P8520S1A1)),]

# Recodificamos las variables necesarias a categóricas
datos3$P1070<- as.factor(datos3$P1070)
datos3$CLASE<- as.factor(datos3$CLASE)
datos3$P1894<- as.factor(datos3$P1894)
datos3$P6020<- as.factor(datos3$P6020)
datos3$P5502<- as.factor(datos3$P5502)
datos3$P6080<- as.factor(datos3$P6080)
datos3$P6240<- as.factor(datos3$P6240)
datos3$P6076S1<- as.factor(datos3$P6076S1)
datos3$P4567<- as.factor(datos3$P4567)
datos3$P8520S3<- as.factor(datos3$P8520S3)
datos3$P8520S1A1<- as.factor(datos3$P8520S1A1)
datos3$P5661S4<- as.factor(datos3$P5661S4)

# Creamos una variable alternativa por si nos queremos devolver al caso anterior
#
departamentos<- as.factor(datos3$P6076S1)

# Función para convertir departamento en regiones

regiones<- Recode(departamentos,
  "c(5,11,15,17,18,25,41,54,63,66,68,73)='ANDINA';
  c(8,13,20,23,44,47,70,88)='CARIBE';
  c(19,27,52,76)='PACIFICA';
  c(50,81,85,99)='ORINOQUIA';
  c(86,91,94,95,97)='AMAZONIA'")
levels(regiones)

```

```
## [1] "AMAZONIA" "ANDINA" "CARIBE" "ORINOQUIA" "PACIFICA"
```

Ya tenemos lista nuestra nueva variables y verificamos que las categorías sean las que establecimos

Modelamiento alternativo por región

```

#creamos un modelo vacío para poder hacer una selección de variables hacia adelante
mod.alternativo1<-multinom(regiones~1,data=datos)

```

```

# weights: 10 (4 variable)
initial value 6228.524721
iter 10 value 4392.962937
final value 4392.960720
converged

```

```

#la selección hacia adelante es favorable en este caso debido
#a la cantidad de variable que abra por las clases de las variables categóricas

```

```

stepAIC(mod.alternativo1,direction = "forward",
        scope=list(upper=~P6020+P6040+P6051+P5502+P6071S1+P767+P6077
                    +P6096+P6081+P6087+P6083+P6088+P6080+P1895+P1896
                    +P1897+P1898+P1899+P1901+P1902+P1903+P1904+P1905
                    ,lower=~1))

# el modelo final es:

modelo.alternativofinal<-multinom(formula = regiones ~ P6080+ P4567 + P8520S3 +P8520S1A1 +
                                CLASE + P767 + P6081 + P5661S4 + P6077 + P1070,
                                data = datos3)

```

```

# weights: 160 (124 variable)
initial value 6228.524721
iter 10 value 4293.082667
iter 20 value 4046.338886
iter 30 value 3937.107636
iter 40 value 3884.777077
iter 50 value 3862.680237
iter 60 value 3857.318940
iter 70 value 3855.251348
iter 80 value 3854.264781
iter 90 value 3853.912027
iter 100 value 3853.852409
final value 3853.852409
stopped after 100 iterations

```

Ahora vamos a observar que tan bien está clasificando este modelo alternativo

```

# Creamos los valores observados y los que predice el modelo para comparar
observados<-as.vector(regiones)
predichos<-as.vector(predict(modelo.alternativofinal,type = "class"))

cont<-0
for (i in 1:3870){

  if(predichos[i]==observados[i]){
    cont<-cont+1
  }
}
cont/3870

```

```
[1] 0.6082687
```

Ahora como podemos apreciar hemos encontrado una capacidad predictiva mayor ya que hemos reducido la cantidad de categorías, por lo tanto hemos de considerar como una mejor opción la alternativa y tratados de encontrar una ahora predicciones basadas en la familia y sus características y así poder determinar la probabilidad de ubicarlos en una región.