Modelo generativo-predictivo de matrículas a partir del PIB

Introducción:

En el marco del concurso **Datathonlatam** de innovación con datos abiertos ... descripción del objetivo:

vamos a elaborar modelos que calculen el número de matrículas un dado nivel educativo para cada departamento a partir del algunos de los valores del PIB del mismo año. Esto permitirá establecer la cantidad y magnitud de la afectación de las diferentes industrias (subvalores del PIB departamental) sobre el número de matrículas en los diferentes niveles educativos.

Objetivo general:

Objetivos específicos:

1. Obtención, transformación, limpieza y ordenado de datos.
2. Análisis exploratorio de datos.
3. Creación de un modelo lineal regresivo por cada nivel educativo.
4. Análisis de los pesos de las variables y sus p-values.
5. Reflexiones.
6. Predicciones de los PIB necesarios para alimentar los modelos lineales en tres (3) departamentos.
7. Análisis de los pesos de los tn para las predicciones.
8. Alimentación de los modelos previamente creados para calcular la demanda de matrículas por nivel académico para 2016 y 2017 en los tres (3) departamentos.
9. Conclusiones.

Estado de la cuestión:

1. Obteniendo datos:

Los datos que vamos a tomar provienen de la página del MEN, del DANE y de datos abiertos.

En su estado original no tenían la disposición que vamos a darle, por eso reseñamos los links en los que pueden encontrarlos en su estado original.

El principal trabajo consistió en diseñar la disposición del set de datos de manera que sea útil para alimentar un modelo lineal regresivo, series de tiempo, modelos de aprendizaje automático e incluso modelos neuronales.

1.1. Datos sobre matrículas:

Las matrículas que en cada año en cada departamento se hacen. Esto por cada nivel educativo:

```
setwd("~/Documents/datathonlatam") ## setting dir
library(readr)
arc <- list.files("./datasets/educacion_superior", pattern = ".csv") ## creating list files csv

matricula <- read_csv(paste("./datasets/educacion_superior", arc[4], sep = "/"))
names(matricula) <- c("departamento", "anio",
                      paste("matr", gsub("tecnica profesional", "tecpro", tolower(names(matricula)[3:8]]
```

Tenemos un set de datos de 198 filas por 8 columnas que contiene los 33 departamentos: AMAZONAS, ANTIOQUIA, ARAUCA, ATLANTICO, BOGOTA D.C., BOLIVAR, BOYACA, CALDAS, CAQUETA, CASANARE, CAUCA, CESAR, CHOCO, CORDOBA, CUNDINAMARCA, GUAINIA, GUAVIARE, HUILA, LA GUAJIRA, MAGDALENA, META, NARINO, NORTE DE SANTANDER, PUTUMAYO, QUINDIO, RISARALDA, SAN ANDRES Y PROVIDENCIA, SANTANDER, SUCRE, TOLIMA, VALLE DEL CAUCA, VAUPES, VICHADA. Esto para los años 2010, 2011, 2012, 2013, 2014, 2015 y para los niveles educativos matr_tecpro, matr_tecnologica, matr_universitaria, matr_especializacion, matr_maestria, matr_doctorado. Una muestra:

```
head(matricula)
```

```
## # A tibble: 6 × 8
##   departamento  anio matr_tecpro matr_tecnologica matr_universitaria
##          <chr> <int>       <int>            <int>              <int>
## 1    ATLANTICO  2010        5297            12083              63339
## 2      BOLIVAR  2010        2603            18802              31854
## 3        CESAR  2010         376             4924              16067
## 4      CORDOBA  2010        1362             4210              21658
## 5   LA GUAJIRA  2010        1059             4579               8474
## 6    MAGDALENA  2010        1636             5599              15815
## # ... with 3 more variables: matr_especializacion <int>,
## #   matr_maestria <int>, matr_doctorado <int>
```

1.2. Datos sobre PIB:

El Producto Interno Bruto por departamento, por año, desagregado en sus 47 componentes:

```
totalpib <- data.frame() ## empty data frame to fill
k <- 0 ## a counter for naming by department from departamentos var
departamentos <- readRDS("./departamentosRDS")
library(readxl) ## library needed to read xls
for(i in seq(from = 2, to = 66, by = 2)) { ## We'll read even sheets (by number, not by name)

pib <- read_excel("~/Documents/datathonlatam/datasets/Copia de PIB_Departamentos_2015provisional.xls",
                  sheet = i, skip = 5, range = "A7:Q59") ## read just a range of the sheet
pib <- pib[!is.na(pib[, 1]) & !is.na(pib[, 2]), c(1, 2:16)] ## delete NAs and keep 2010 to 2015
pib <- data.frame(t(pib)) ## transpose
k <- k + 1 ## increase 'iterator' of departamentos names
pib$departamento <- departamentos[k] ## namig the departamento
pib$anio <- rownames(pib) ## anio as value not just as rowname
pib <- pib[c(2:nrow(pib)), c(48:49, 1:47)] ## erasing first row and reordering cols-vars
totalpib <- rbind(totalpib, pib) ## stacking results in one data frame
}

rm(pib) ## erase inecessary data frame

totalpib[, 3:ncol(totalpib)] <- apply(totalpib[, 3:ncol(totalpib)], 2, as.integer) ## data as numeric

library(readr)
dummy_names_PIB <- as.data.frame(read_csv("~/Documents/datathonlatam/dummy_names_PIB.csv")[, c(1:2)])
```

Los nombres de cada uno de las subvariables del PIB por departamento anual tuvieron que ser sustituidas en su orden por variables X1, X2, X3, . . . , X47. Sus nombres reales se listan:

```
dummy_names_PIB[, 2]
```

```
##  [1] "AGRICULTURA, GANADERIA, CAZA, SILVICULTURA Y PESCA"
##  [2] "Cultivo de café"
##  [3] "Cultivo de otros productos agrícolas"
##  [4] "Producción pecuaria y caza incluyendo las actividades veterinarias"
##  [5] "Silvicultura, extracción de madera y actividades conexas"
##  [6] "Pesca, produccion de peces en criaderos y granjas piscícolas; actividades de servicios relacio
##  [7] "EXPLOTACION DE MINAS Y CANTERAS"
##  [8] "Extracción de carbón, carbón lignítico y turba"
##  [9] "Extracción de petróleo crudo y de gas natural; actividades de servicios relacionadas con la ex
## [10] "Extracción de minerales metáliferos"
## [11] "Extracción de minerales no metálicos"
## [12] "INDUSTRIA MANUFACTURERA"
## [13] "Alimentos, bebidas y tabaco"
## [14] "Resto de la Industria"
## [15] "SUMINISTRO DE ELECTRICIDAD, GAS Y AGUA"
## [16] "Generación, captación y distribución de energía eléctrica"
## [17] "Fabricación de gas; distribución de combustibles gaseosos por tuberías; suministro de vapor y a
## [18] "Captación, depuración y distribución de agua"
## [19] "Eliminación de desperdicios y aguas residuales, saneamiento y actividades similares"
## [20] "CONSTRUCCION"
## [21] "Construcción de edificaciones completas y de partes de edificaciones;  acondicionamiento de edi
## [22] "Construcción de obras de ingeniería civil"
## [23] "COMERCIO, REPARACIÓN, RESTAURANTES Y HOTELES"
## [24] "Comercio"
## [25] "Mantenimiento y reparación de vehículos automotores; reparación de efectos personales y enseres
## [26] "Hoteles, restaurantes, bares y similares"
## [27] "TRANSPORTE, ALMACENAMIENTO Y COMUNICACIONES"
## [28] "Transporte por vía terrestre"
## [29] "Transporte por vía acuática"
## [30] "Transporte por vía aérea"
## [31] "Actividades complementarias y auxiliares al transporte; actividades de agencias de viajes"
## [32] "Correo y telecomunicaciones"
## [33] "ESTABLECIMIENTOS FINANCIEROS, SEGUROS, ACTIVIDADES INMOBILIARIAS Y SERVICIOS A LAS EMPRESAS"
## [34] "Intermediación financiera"
## [35] "Actividades inmobiliarias y alquiler de vivienda"
## [36] "Actividades de servicios a las empresas excepto servicios financieros e inmobiliarios"
## [37] "ACTIVIDADES DE SERVICIOS SOCIALES, COMUNALES Y PERSONALES"
## [38] "Administración pública y defensa; seguridad social de afiliación obligatoria"
## [39] "Educación de mercado"
## [40] "Educación de no mercado"
## [41] "Servicios sociales y de salud de mercado"
## [42] "Actividades de asociaciones n.c.p.; actividades de esparcimiento y actividades culturales y dep
## [43] "Actividades de asociaciones n.c.p.; actividades de esparcimiento y actividades culturales y dep
## [44] "Hogares privados con servicio doméstico"
## [45] "Subtotal Valor Agregado"
## [46] "Impuestos"
## [47] "PIB TOTAL DEPARTAMENTAL"
```

Tenemos un set de datos de 495 filas por 49 columnas que contiene los 33 departamentos: AMAZONAS, ANTIOQUIA, ARAUCA, ATLANTICO, BOGOTA D.C., BOLIVAR, BOYACA, CALDAS, CAQUETA, CASANARE, CAUCA, CESAR, CHOCO, CORDOBA, CUNDINAMARCA, GUAINIA, GUAVIARE, HUILA, LA GUAJIRA, MAGDALENA, META, NARINO, NORTE DE SANTANDER, PUTUMAYO, QUINDIO,

RISARALDA, SAN ANDRES Y PROVIDENCIA, SANTANDER, SUCRE, TOLIMA, VALLE DEL CAUCA, VAUPES, VICHADA. Esto para los años 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014 y para las 47 variables reseñadas antes.

1.3. Datos sobre proyecciones poblacionales:

El DANE, a partir del CENSO 2005 hizo proyecciones sobre la población en cada uno de los departamentos. Estos datos son los que tomamos.

```
library(readr)
dane <- read_csv("./datasets/censo 2005/proyeccion poblacion DANE 2005.csv")

totaldane <- data.frame()
for( i in 1:nrow(dane)){
    temp <- data.frame(departamento = dane[i, 1],
                                    anio = seq(1985, 2020),
                                    proy = as.numeric(dane[i, 2:ncol(dane)]))
    totaldane <- rbind(totaldane, temp)
}

rm(list = c("dane", "temp"))
```

Tenemos un set de datos de 1188 filas por 3 columnas que contiene los 33 departamentos: AMAZONAS, ANTIOQUIA, ARAUCA, ATLANTICO, BOGOTA D.C., BOLIVAR, BOYACA, CALDAS, CAQUETA, CASANARE, CAUCA, CESAR, CHOCO, CORDOBA, CUNDINAMARCA, GUAINIA, GUAVIARE, HUILA, LA GUAJIRA, MAGDALENA, META, NARINO, NORTE DE SANTANDER, PUTUMAYO, QUINDIO, RISARALDA, SAN ANDRES Y PROVIDENCIA, SANTANDER, SUCRE, TOLIMA, VALLE DEL CAUCA, VAUPES, VICHADA. Esto para los años 1985, 1986, 1987, 1988, 1989, 1990, 1991, 1992, 1993, 1994, 1995, 1996, 1997, 1998, 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020.

1.4. Juntando todos los datos:

Dado que para cada set de datos tenemos siempre los 33 departamentos pero no el mismo periodo de años, vamos a juntarlos durante los años comunes a todos (de 2010 a 2015) y conservando todas las variables. El nombre de los departamentos lo vamos a conservar como factor, y el resto como numérico.

```
df <- merge(merge(totaldane, totalpib), matricula)
df$departamento <- as.factor(df$departamento)
df$proy <- as.integer(df$proy)
str(df)
```

```
## 'data.frame':    165 obs. of  56 variables:
##  $ departamento        : Factor w/ 33 levels "AMAZONAS","ANTIOQUIA",..: 1 1 1 1 1 2 2 2 2 2 ...
##  $ anio                : int  2010 2011 2012 2013 2014 2010 2011 2012 2013 2014 ...
##  $ proy                : int  45509584 46044601 46581823 47121089 47661787 4383277 4428675 4474369 45
##  $ X1                  : int  53 50 51 52 61 4495 4859 4874 5206 5420 ...
##  $ X2                  : int  0 0 0 0 0 645 754 562 690 761 ...
##  $ X3                  : int  0 1 0 0 0 2201 2354 2471 2592 2520 ...
##  $ X4                  : int  2 2 2 2 2 1544 1627 1702 1766 1924 ...
##  $ X5                  : int  7 7 8 10 9 89 103 114 132 183 ...
##  $ X6                  : int  44 40 41 40 50 16 21 25 26 32 ...
##  $ X7                  : int  0 0 0 0 0 1718 2434 2985 2853 2645 ...
##  $ X8                  : int  0 0 0 0 0 17 43 19 11 25 ...
```

```
##  $ X9                 : int  0 0 0 0 0 749 1155 1432 1553 1232 ...
##  $ X10                : int  0 0 0 0 0 697 931 1218 928 927 ...
##  $ X11                : int  0 0 0 0 0 255 305 316 361 461 ...
##  $ X12                : int  8 9 9 10 10 9916 10950 11827 12100 12330 ...
##  $ X13                : int  4 4 4 5 5 2161 2206 2475 2643 2763 ...
##  $ X14                : int  4 5 5 5 5 7755 8744 9352 9457 9567 ...
##  $ X15                : int  7 7 7 7 7 3981 4427 4748 4849 4845 ...
##  $ X16                : int  5 5 5 5 5 2880 3276 3499 3563 3541 ...
##  $ X17                : int  0 0 0 0 0 151 166 198 224 222 ...
##  $ X18                : int  1 1 1 1 1 493 515 568 569 580 ...
##  $ X19                : int  1 1 1 1 1 457 470 483 493 502 ...
##  $ X20                : int  0 0 0 0 0 5699 7081 7577 9388 12511 ...
##  $ X21                : int  0 0 0 0 0 2701 3532 4138 5119 5938 ...
##  $ X22                : int  0 0 0 0 0 2998 3549 3439 4269 6573 ...
##  $ X23                : int  73 80 88 96 100 9700 10813 11366 12279 13405 ...
##  $ X24                : int  47 52 54 60 62 6577 7414 7523 8035 8763 ...
##  $ X25                : int  0 0 0 0 0 915 984 1117 1212 1318 ...
##  $ X26                : int  26 28 34 36 38 2208 2415 2726 3032 3324 ...
##  $ X27                : int  40 42 42 44 50 4297 4508 4659 5347 5858 ...
##  $ X28                : int  0 0 0 0 0 2101 2197 2223 2667 2984 ...
##  $ X29                : int  0 0 1 0 0 90 60 53 59 69 ...
##  $ X30                : int  18 19 18 19 22 219 228 252 312 348 ...
##  $ X31                : int  2 2 2 2 3 260 310 331 365 419 ...
##  $ X32                : int  20 21 21 23 25 1627 1713 1800 1944 2038 ...
##  $ X33                : int  27 28 33 34 38 15526 17165 18664 19900 21620 ...
##  $ X34                : int  18 19 23 24 26 3622 4019 4560 4939 5381 ...
##  $ X35                : int  7 7 8 8 10 6015 6398 6845 7307 7780 ...
##  $ X36                : int  2 2 2 2 2 5889 6748 7259 7654 8459 ...
##  $ X37                : int  158 173 191 210 229 10124 11014 12105 13326 14535 ...
##  $ X38                : int  80 87 96 106 116 3111 3406 3746 4263 4657 ...
##  $ X39                : int  1 1 1 1 1 1665 1820 1997 2199 2435 ...
##  $ X40                : int  49 53 58 65 69 1792 1898 2127 2314 2507 ...
##  $ X41                : int  16 18 21 24 28 1617 1766 2020 2261 2534 ...
##  $ X42                : int  10 12 13 12 13 1174 1305 1342 1358 1405 ...
##  $ X43                : int  1 1 1 1 1 179 184 192 207 223 ...
##  $ X44                : int  1 1 1 1 1 586 635 681 724 774 ...
##  $ X45                : int  366 389 421 453 495 65456 73251 78805 85248 93169 ...
##  $ X46                : int  16 20 22 24 28 5881 7225 7562 7365 8490 ...
##  $ X47                : int  382 409 443 477 523 71337 80476 86367 92613 101659 ...
##  $ matr_tecpro        : int  60 0 0 82 88 7400 3327 2622 4269 3910 ...
##  $ matr_tecnologica   : int  758 665 642 292 204 94171 107739 102893 111224 110533 ...
##  $ matr_universitaria : int  311 307 231 329 329 132554 143741 155416 164092 175701 ...
##  $ matr_especializacion: int  17 27 13 10 14 7821 9065 8550 9051 8874 ...
##  $ matr_maestria      : int  0 27 27 17 15 3322 4176 4185 4862 5520 ...
##  $ matr_doctorado     : int  0 0 0 3 6 611 696 787 967 1147 ...
```

2. Análisis exploratorio de datos.

Ahora que hemos definido nuestro set de datos vamos a explorarlo. Lo primero que necesitamos es remover algunas variables del PIB que son suma de otras (X1 es la suma de X2 a X6, X7 es la suma de X8 a X11 ... Las que aparecen relacionadas en mayúsculas son la suma de las siguientes en minúsculas, de acuerdo con la tabla presentada en el apartado 1.2. Datos sobre PIB:) ya que son redundantes.

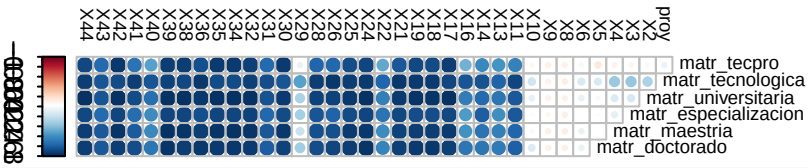
```
df_limpio <- df[, -c(4, 10, 15, 18, 23, 26, 30, 36, 40, 48:50)]
correlaciones <- cor(df_limpio[, c(2:ncol(df_limpio))])
```

Pasamos de tener un set de datos de 165 por 56 a tener uno de 165 por 44

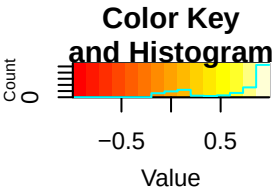
2.2. Revisando correlaciones:

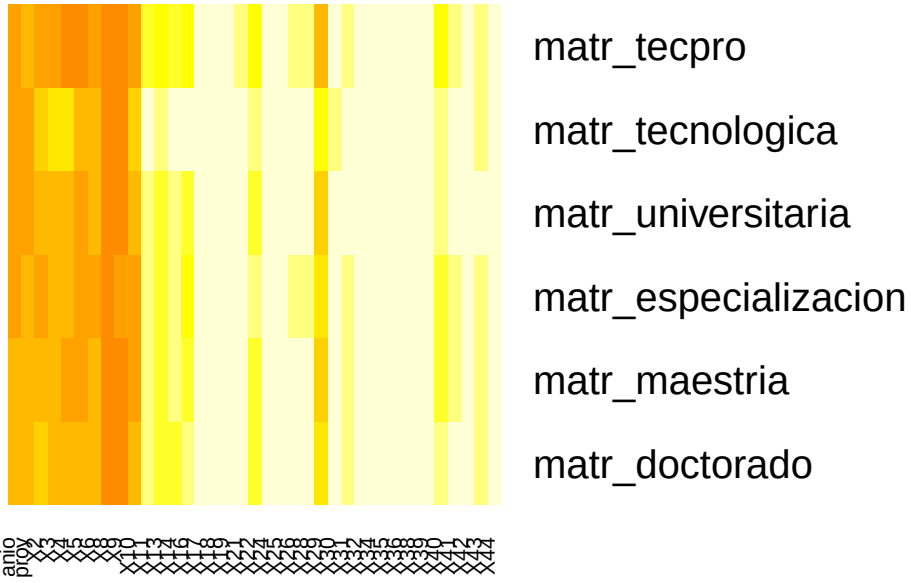
Una vez retiradas las variables podemos revisar las correlaciones entre todas las variables del PIB y las de matrículas en educación superior por nivel educativo; dejando por fuera los datos no numéricos (los nombres de los departamentos):

```
library(corrplot)
corrplot(correlaciones[1:37, 38:43], order = "original", tl.cex = 0.6, tl.col = "black", type = "lower"
```




```
correlate <- cor(df_limpio[, c(39:44)], df_limpio[, c(2:38)])
library(gplots)
heatmap.2(correlate, dendrogram='none', Rowv=FALSE, Colv=FALSE,trace='none', margins = c(4, 13))
```





3. Creación de un modelo lineal regresivo por cada nivel educativo.

Nos referimos a un modelo generativo como aquel que permite a establecer la medida en que las variables independientes o conocidas determinan la variable dependiente o a predecir. En el presente caso, un modelo generativo permite predecir el número de matrículas para un nivel educativo a partir de algunos de los valores del PIB del departamento: si conocemos algunos datos del PIB del departamento, podemos predecir o inferir el número de matrículas de cierto nivel académico en el departamento.

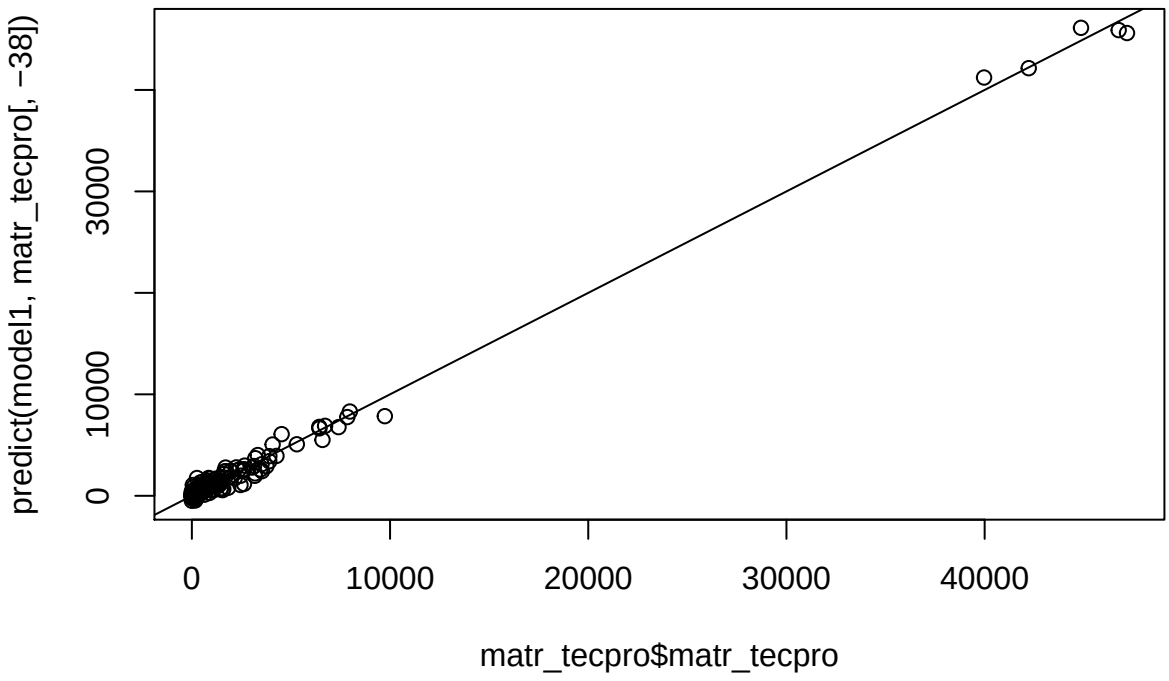
Lo importante de este tipo de modelo es que permite extraer la medida en que cada uno de los rubros del PIB contribuye al aumento o decremento del número de matrículas, y nos delimita el número de variables del PIB sobre las que tendremos que llevar a cabo predicciones para calcular las matrículas en años futuros.

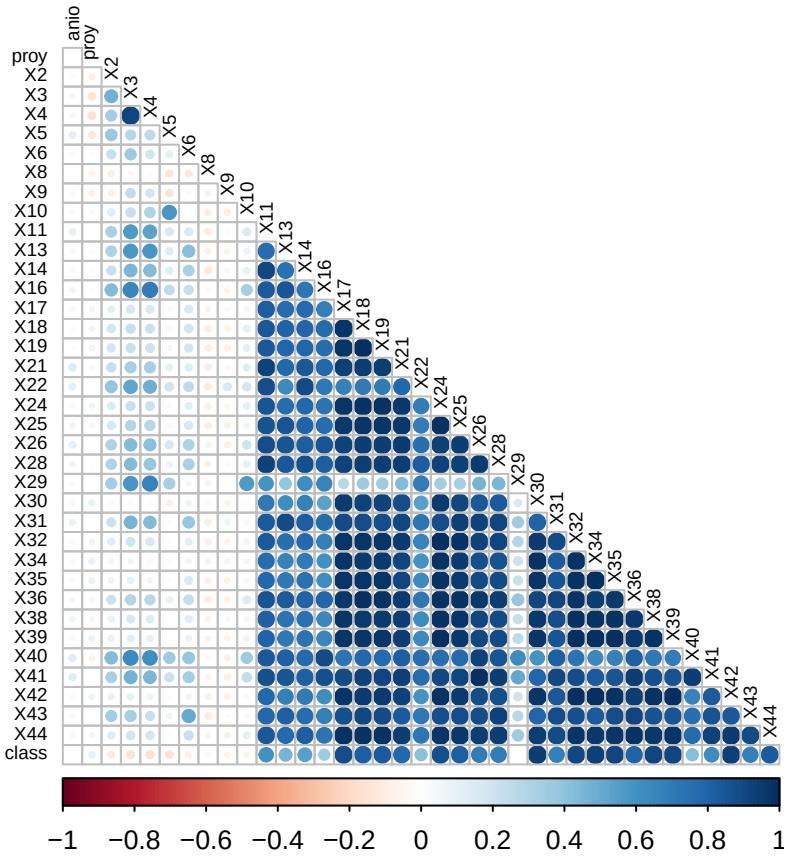
```
matr_tecpro <- df_limpio[, c(2:39)]
matr_tecpro$class <- ifelse(df_limpio$departamento == "BOGOTA D.C.", 1, 0)
model1 <- lm(matr_tecpro ~ .,data = matr_tecpro)
summary(model1)
```

```
##
## Call:
## lm(formula = matr_tecpro ~ ., data = matr_tecpro)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1545.4  -286.5    -7.8   230.0  1888.5
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.607e+04  9.631e+04   0.582 0.561490
## anio        -2.778e+01  4.789e+01  -0.580 0.562968
## proy         2.273e-05  9.844e-06   2.309 0.022554 *
## X2           3.921e+00  9.846e-01   3.982 0.000115 ***
## X3           3.270e-01  7.152e-01   0.457 0.648284
## X4          -1.304e+00  7.275e-01  -1.792 0.075506 .
## X5           3.522e+00  2.927e+00   1.204 0.231010
## X6          -1.727e+01  6.824e+00  -2.530 0.012627 *
## X8          -3.011e-01  9.441e-02  -3.189 0.001803 **
## X9           3.355e-03  3.609e-02   0.093 0.926088
## X10         -5.440e-01  4.093e-01  -1.329 0.186229
## X11          3.142e+01  5.715e+00   5.498 2.04e-07 ***
## X13          9.275e-01  8.518e-01   1.089 0.278270
## X14         -5.999e-01  1.598e-01  -3.753 0.000265 ***
## X16          7.729e-01  8.288e-01   0.933 0.352846
## X17         -1.475e+01  1.059e+01  -1.392 0.166332
## X18         -2.786e+01  2.222e+01  -1.254 0.212226
## X19          2.770e+01  2.291e+01   1.209 0.228902
## X21         -1.172e+00  4.243e-01  -2.762 0.006603 **
## X22         -1.392e+00  3.586e-01  -3.881 0.000167 ***
## X24         -1.232e+00  7.510e-01  -1.641 0.103302
## X25         -6.170e+00  3.568e+00  -1.729 0.086185 .
## X26          1.063e+01  2.089e+00   5.088 1.28e-06 ***
## X28         -9.403e-01  7.399e-01  -1.271 0.206106
## X29          5.914e+01  1.677e+01   3.527 0.000587 ***
## X30         -1.765e+01  5.483e+00  -3.218 0.001639 **
## X31          1.574e+01  3.946e+00   3.990 0.000111 ***
## X32          4.137e+00  3.897e+00   1.062 0.290361
```

```
## X34          4.439e-01  1.551e+00   0.286 0.775179
## X35          2.533e+00  8.468e-01   2.991 0.003341 **
## X36         -2.589e+00  5.221e-01  -4.958 2.25e-06 ***
## X38         -5.150e-01  8.530e-01  -0.604 0.547081
## X39         -8.239e+00  4.010e+00  -2.055 0.041973 *
## X40         -1.079e+01  3.031e+00  -3.558 0.000527 ***
## X41          8.224e+00  3.705e+00   2.220 0.028235 *
## X42         -8.040e+00  2.511e+00  -3.202 0.001730 **
## X43         -9.544e+00  8.669e+00  -1.101 0.273057
## X44          3.320e+01  6.821e+00   4.867 3.32e-06 ***
## class        5.975e+04  6.080e+03   9.827  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 618.8 on 126 degrees of freedom
## Multiple R-squared:  0.9949, Adjusted R-squared:  0.9934
## F-statistic:   648 on 38 and 126 DF,  p-value: < 2.2e-16
```

```r
plot(matr_tecpro$matr_tecpro, predict(model1, matr_tecpro[, -38]))
abline(a = 0, b = 1)
```





Debemos tener cuidado con las variables independientes que tienen altas correlaciones entre ellas

```
library(caret)
```

```
## Loading required package: lattice
```

```
## Loading required package: ggplot2
```

```
# Checking Variables that are highly correlated
highlyCorrelated <- findCorrelation(descrCor, cutoff=0.8)

#Identifying Variable Names of Highly Correlated Variables
highlyCorCol <- colnames(numericData)[highlyCorrelated]

#Print highly correlated attributes
highlyCorCol
```

```
##  [1] "X26"   "X36"   "X41"   "X28"   "X21"   "X25"   "X18"   "X19"
##  [9] "X44"   "X24"   "X32"   "X17"   "X31"   "X38"   "X11"   "X39"
## [17] "X35"   "X43"   "X42"   "X34"   "X40"   "X14"   "X13"   "class"
## [25] "X3"
```

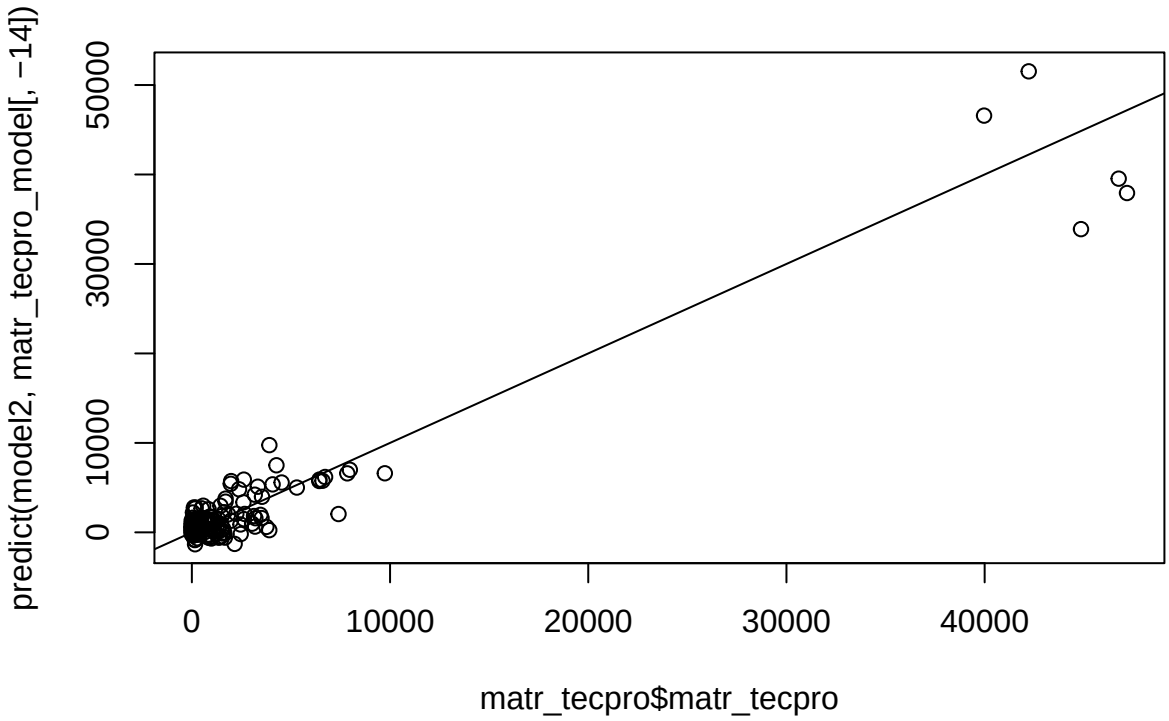
```
#Remove highly correlated variables and create a new dataset
matr_tecpro_model <- matr_tecpro[, -which(colnames(matr_tecpro) %in% highlyCorCol)]
dim(matr_tecpro_model)
```

```
## [1] 165  14
```

```
model2 <- lm(matr_tecpro ~ ., data = matr_tecpro_model)
summary(model2)
```

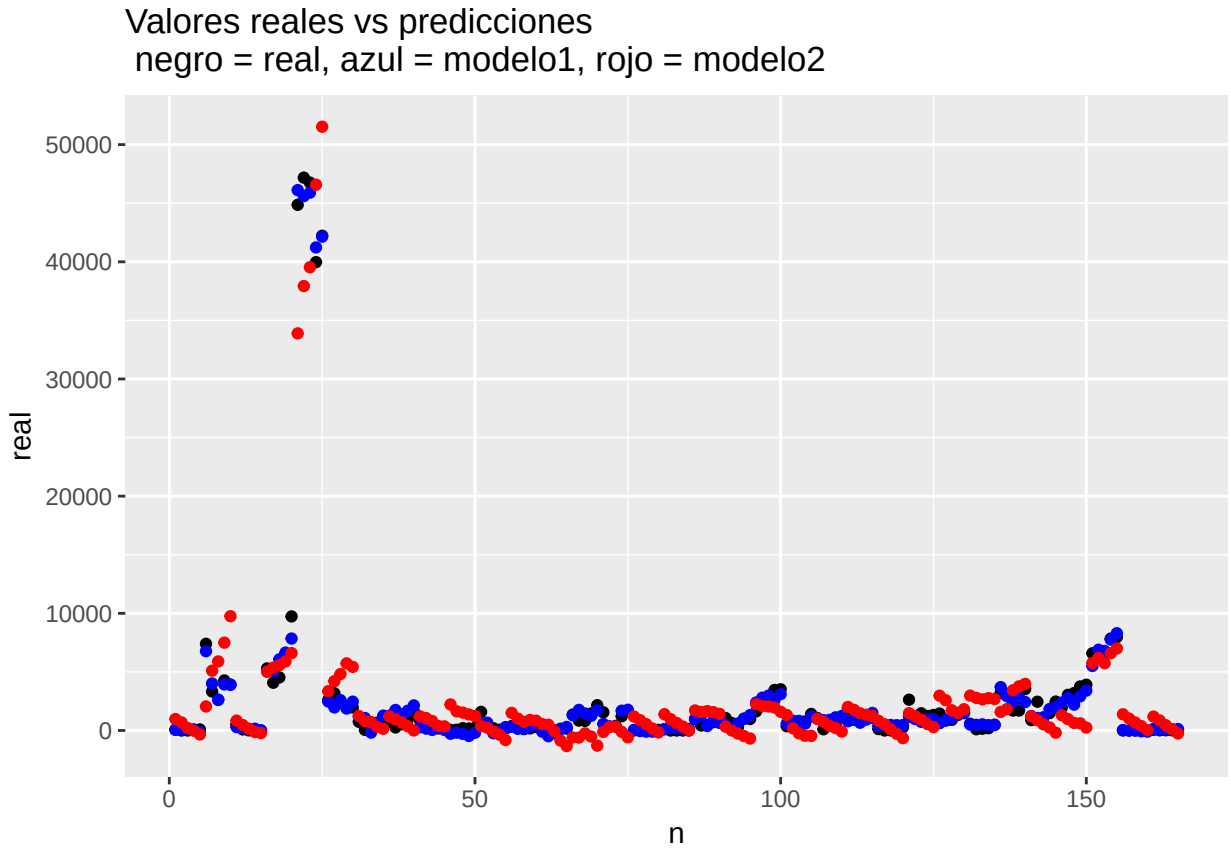
```
##
## Call:
## lm(formula = matr_tecpro ~ ., data = matr_tecpro_model)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9303.9  -840.4  -158.5   731.6 10971.1
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.813e+05  2.538e+05   2.684  0.00808 **
## anio        -3.384e+02  1.262e+02  -2.682  0.00814 **
## proy        -9.219e-06  2.251e-05  -0.410  0.68271
## X2           3.482e-02  1.151e+00   0.030  0.97591
## X4           6.959e-01  8.606e-01   0.809  0.42002
## X5          -7.218e+00  5.987e+00  -1.205  0.22991
## X6          -6.750e+00  3.170e+00  -2.129  0.03486 *
## X8          -2.488e-01  1.664e-01  -1.495  0.13696
## X9          -1.215e-01  4.826e-02  -2.518  0.01283 *
## X10          4.191e-01  9.326e-01   0.449  0.65382
## X16         -5.112e-01  6.727e-01  -0.760  0.44847
## X22          1.166e+00  3.496e-01   3.335  0.00107 **
## X29         -9.058e+01  2.996e+01  -3.023  0.00294 **
## X30          2.952e+01  1.515e+00  19.485  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2193 on 151 degrees of freedom
## Multiple R-squared:  0.9234, Adjusted R-squared:  0.9168
## F-statistic: 139.9 on 13 and 151 DF,  p-value: < 2.2e-16
```

```
plot(matr_tecpro$matr_tecpro, predict(model2, matr_tecpro_model[, -14]))
abline(a = 0, b = 1)
```



Vamos a ver entre los dos modelos cuáles variables conservar

```
p <- data.frame(real = matr_tecpro$matr_tecpro,
                m1 = predict(model1, matr_tecpro_pred),
                m2 = predict(model2, matr_tecpro_model[, -14]),
                n = seq(1, 165, 1))
library(ggplot2)
ggplot(data = p, aes(x = n, y = real), colour = "black") + geom_point() + geom_point(aes(x = n, y = m1)
```



4. Análisis de los pesos de las variables y sus p-values.

5. Reflexiones.

6. Predicciones de los PIB necesarios para alimentar los modelos lineales en tres (3) departamentos.

7. Análisis de los pesos de los tn para las predicciones.

8. Alimentación de los modelos previamente creados para calcular la demanda de matrículas por nivel académico para 2016 y 2017 en los tres (3) departamentos.

9. Conclusiones.