modelo econometrico

Ana Munoz

Pre procesamiento de datos

1. Lectura de datos y formato panel

```r
df <- read.csv('df_17_21_noclean.csv') %>%
  dplyr::select(year, country, total_earnings, # orden de datos panel
         total_players,
         -iso, -code, #no aplica el modelo
         pbicap,
         gdp_gr,
         CPI, # corrupcion
         internet, elect_acc,
         exp_tech, # tech access
         -age_work,
         life_exp, # edades
         poblacion# people
        ) %>%
  arrange(country, decreasing = FALSE)



head(df, 3)
```

```
##         year country total_earnings total_players   pbicap   gdp_gr CPI
## 1 2017-01-01 Albania        2868.16             2 4531.032 3.898112  38
## 2 2018-01-01 Albania        1346.55             3 5287.661 4.276312  36
## 3 2019-01-01 Albania       37459.64            14 5396.214 2.523541  35
##   internet elect_acc exp_tech life_exp poblacion
## 1 62.40000     99.89 31.52551   79.047   2873457
## 2 65.40000    100.00 30.75579   79.184   2866376
## 3 68.55039    100.00 31.18889   79.282   2854191
```

2. Valores faltantes

- Numero de Valores faltantes por variable

```r
sapply(df, function(x) sum(is.na(x)))
```

```
##           year        country total_earnings  total_players         pbicap
##              0              0              0              0             10
```

```
##         gdp_gr            CPI       internet      elect_acc       exp_tech
##              8              6             54             99             38
##       life_exp      poblacion
##             99              0
```

- corriegiendo los NAs

```
# Pbi faltantes
## "Cuba"  "Lebanon"  "Syrian Arab Republic"  "Venezuela"
pbicap_faltantes <- unique(df[is.na(df$pbicap), ]$country)
df <- df[!df$country %in% pbicap_faltantes, ]

# Internet: 2 faltantes -> 2018 cambodia y trinidad y tobago

### Cambodia, hueco en 2018, reemplazdo por el promedio

df[df$country=='Cambodia', 'internet'][2] <-
          (df[df$country=='Cambodia', 'internet'][1] +
             df[df$country=='Cambodia', 'internet'][3])/2

### trinidad y tobago, reemplazdo por el promedio
df[df$country=='Trinidad and Tobago', 'internet'][2] <-
          (df[df$country=='Trinidad and Tobago', 'internet'][1]+
              df[df$country=='Trinidad and Tobago', 'internet'][3])/2

# Acceso a electricidad y life expectanci solo antes del 2022
df <- df %>%
  filter(year < as.Date("2022-01-01"))

# EXportacion tecnologica voy a quitar a los paises que no tiene exportacion por temas politicos
##"Iran, Islamic Republic of" "United Arab Emirates" "Viet Nam"
exp_faltantes <- unique(df[is.na(df$exp_tech), ]$country)
df <- df[!df$country %in% exp_faltantes, ]


# CPI macao no tiene por temas politicos
df <- df[df$country != 'Macao', ]


### Jugadores por poblacion por millon
df$players_ppl <- (df$total_players/df$poblacion)*1000000

###########

# verificamos NAs, ahora no tengo NAS
sapply(df, function(x) sum(is.na(x)))
```

```
##           year        country total_earnings  total_players         pbicap
##              0              0              0              0              0
##         gdp_gr            CPI       internet      elect_acc       exp_tech
##              0              0              0              0              0
##       life_exp      poblacion    players_ppl
##              0              0              0
```

3. Normalizacion con logaritmo

- valores con varianzas muy grandes

```
summary(df)
```

```
##      year             country          total_earnings     total_players
##  Length:455         Length:455         Min.   :      55   Min.   :   1.0
##  Class :character   Class :character   1st Qu.:   48243   1st Qu.:  20.0
##  Mode  :character   Mode  :character   Median :  289199   Median :  83.0
##                                        Mean   : 1948524   Mean   : 275.7
##                                        3rd Qu.: 1358358   3rd Qu.: 269.5
##                                        Max.   :51416470   Max.   :6280.0
##      pbicap           gdp_gr               CPI           internet
##  Min.   :  1243   Min.   :-18.8544   Min.   :18.00   Min.   : 13.78
##  1st Qu.:  4732   1st Qu.: -0.4916   1st Qu.:35.00   1st Qu.: 64.76
##  Median : 12532   Median :  1.9630   Median :44.00   Median : 79.17
##  Mean   : 22561   Mean   :  1.5517   Mean   :50.57   Mean   : 74.82
##  3rd Qu.: 34148   3rd Qu.:  4.4185   3rd Qu.:67.00   3rd Qu.: 88.97
##  Max.   :133712   Max.   : 18.7329   Max.   :89.00   Max.   :100.00
##    elect_acc          exp_tech         life_exp       poblacion
##  Min.   : 80.70   Min.   : 4.167   Min.   :62.34   Min.   :3.434e+05
##  1st Qu.: 99.80   1st Qu.:24.468   1st Qu.:73.02   1st Qu.:5.139e+06
##  Median :100.00   Median :43.495   Median :76.60   Median :1.100e+07
##  Mean   : 98.91   Mean   :42.225   Mean   :76.68   Mean   :6.720e+07
##  3rd Qu.:100.00   3rd Qu.:57.264   3rd Qu.:81.40   3rd Qu.:4.473e+07
##  Max.   :100.00   Max.   :92.665   Max.   :85.50   Max.   :1.412e+09
##   players_ppl
##  Min.   :  0.00604
##  1st Qu.:  1.58751
##  Median :  5.99318
##  Mean   : 12.89707
##  3rd Qu.: 15.43862
##  Max.   :127.57826
```

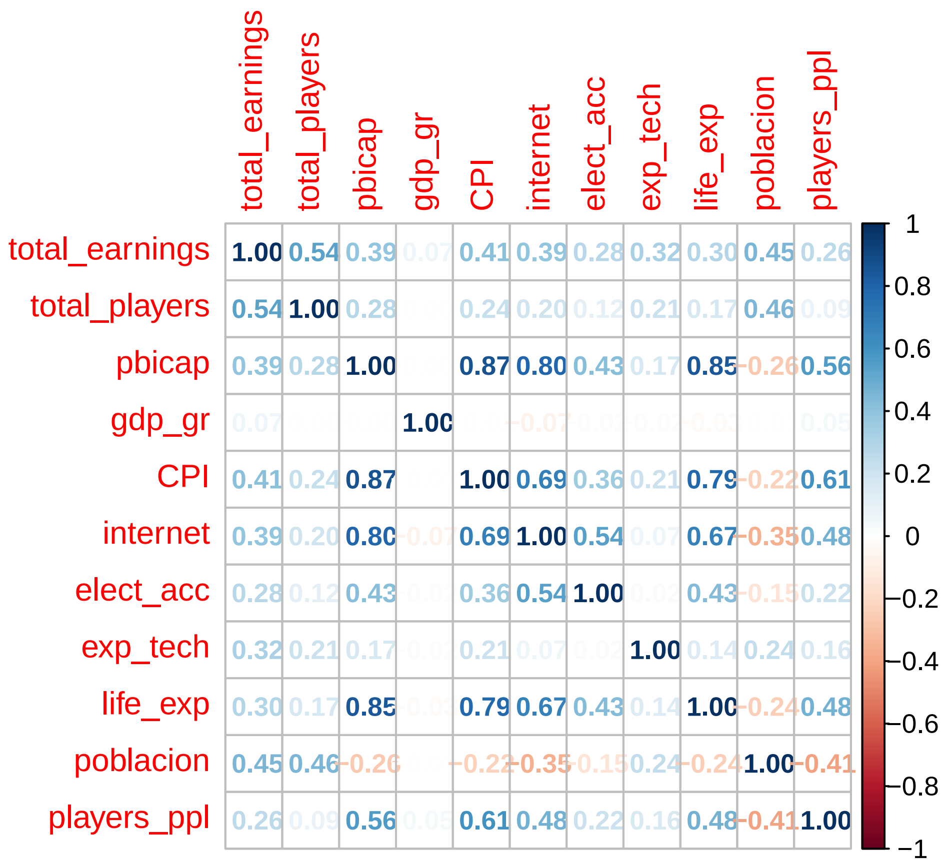
- aplico normalizacion logaritmica en algunas variables

```
df_standar <- df %>%
   mutate(across(c("total_earnings", "pbicap", "poblacion"), ~log(.) %>% as.vector))
```

4. Correlacion y eliminacion de variables

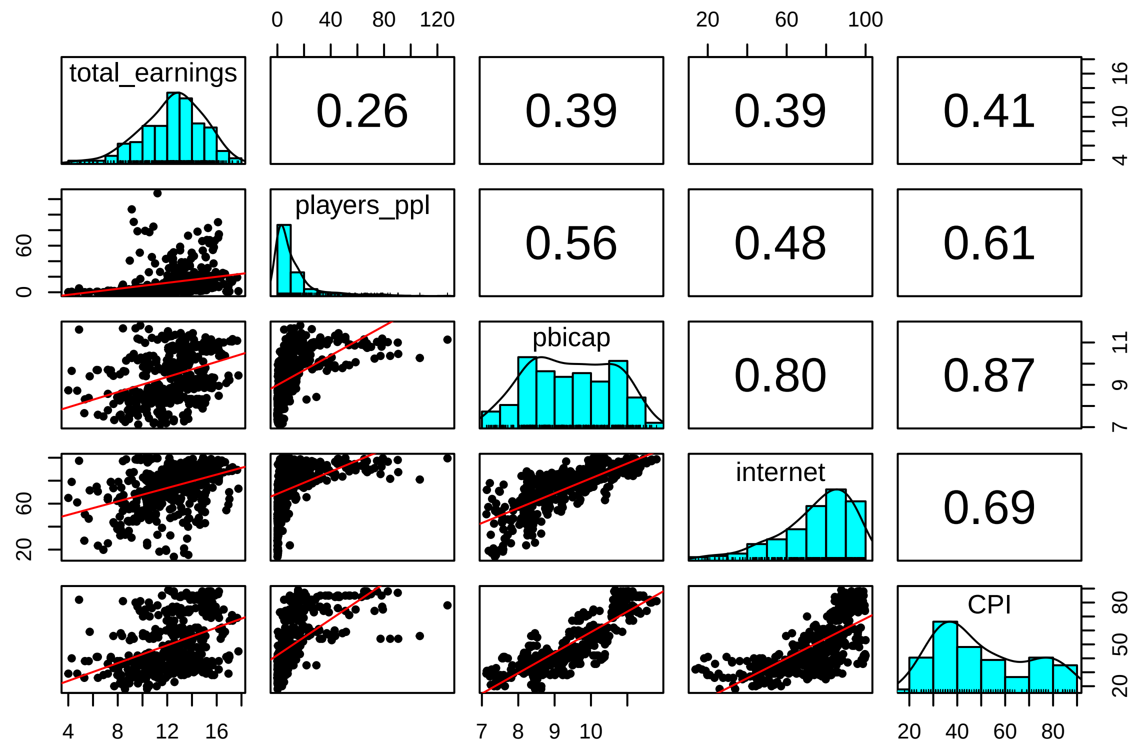
- verificacion de correlaciones

```
matriz_corr <- cor(df_standar[3:13])
corrplot(matriz_corr, method = 'number', number.cex = 0.8)
```



- Veamos mas a detalle las correlaciones encontradas, pbippa, cpi e acceso internet Se debe crear otra variable para eliminar la correlacion? estas correlaciones no implican causalidad, pero como debo tratarlas?

```
pairs.panels(df_standar%>%dplyr::select(total_earnings,players_ppl, pbicap, internet,CPI),
             smooth = FALSE,      # Si TRUE, dibuja ajuste suavizados de tipo loess
             scale = FALSE,       # Si TRUE, escala la fuente al grado de correlación
             density = TRUE,      # Si TRUE, añade histogramas y curvas de densidad
             ellipses = FALSE,    # Si TRUE, dibuja elipses
             method = "pearson", # Método de correlación (también "spearman" o "kendall")
             lm = TRUE,           # Si TRUE, dibuja un ajuste lineal en lugar de un ajuste LOESS
             cor = TRUE,          # Si TRUE, agrega correlaciones
             jiggle = FALSE,      # Si TRUE, se añade ruido a los dato
)
```



Implementando modelos

0. Preparando los datos

- tenemos datos panel con la siguente forma 90 paises 5 anios y estas cols

```
# df_panel <- df_standar %>% select(-c(total_players))
# por si sera necesario quitar algunas var
dim(table(df_standar$country,df_standar$year))
```

```
## [1] 91  5
```

```
colnames(df_standar)
```

```
##  [1] "year"           "country"        "total_earnings" "total_players"
##  [5] "pbicap"         "gdp_gr"         "CPI"            "internet"
##  [9] "elect_acc"      "exp_tech"       "life_exp"       "poblacion"
## [13] "players_ppl"
```

- definimos las variables para el modelo

```r
attach(df_standar)
Y <- cbind(total_earnings)
X <- cbind(pbicap, gdp_gr, CPI, internet,exp_tech,
           elect_acc, life_exp, poblacion, players_ppl)

df_panel <- pdata.frame(df_standar,
                        index=c('country','year'))

head(df_panel,3)
```

```
##                           year country total_earnings total_players   pbicap
## Albania-2017-01-01 2017-01-01 Albania       7.961426             2 8.418705
## Albania-2018-01-01 2018-01-01 Albania       7.205301             3 8.573131
## Albania-2019-01-01 2019-01-01 Albania      10.531019            14 8.593453
##                      gdp_gr CPI internet elect_acc exp_tech life_exp poblacion
## Albania-2017-01-01 3.898112  38 62.40000     99.89 31.52551   79.047  14.87103
## Albania-2018-01-01 4.276312  36 65.40000    100.00 30.75579   79.184  14.86856
## Albania-2019-01-01 2.523541  35 68.55039    100.00 31.18889   79.282  14.86430
##                    players_ppl
## Albania-2017-01-01   0.6960257
## Albania-2018-01-01   1.0466178
## Albania-2019-01-01   4.9050677
```

4. Efectos Fijos

```r
fijos <- plm(Y ~ X, data=df_panel, model= "within")
summary(fijos)
```

```
## Oneway (individual) effect Within Model
##
## Call:
## plm(formula = Y ~ X, data = df_panel, model = "within")
##
## Balanced Panel: n = 91, T = 5, N = 455
##
## Residuals:
##      Min.   1st Qu.    Median   3rd Qu.      Max.
## -3.149927 -0.375976 -0.020633  0.453314  2.345430
##
## Coefficients:
##               Estimate Std. Error t-value  Pr(>|t|)
## Xpbicap       0.9989354  0.6252323  1.5977 0.1109988
## Xgdp_gr       0.0185386  0.0109947  1.6861 0.0926462 .
## XCPI          0.0401443  0.0258401  1.5536 0.1211792
## Xinternet     0.0360802  0.0106407  3.3908 0.0007754 ***
## Xexp_tech     0.0053487  0.0072910  0.7336 0.4636768
## Xelect_acc    0.1402941  0.0480284  2.9211 0.0037114 **
## Xlife_exp    -0.0530382  0.0549139 -0.9658 0.3347810
## Xpoblacion    7.9269261  3.2059174  2.4726 0.0138816 *
## Xplayers_ppl  0.0267594  0.0079393  3.3705 0.0008329 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    359.09
## Residual Sum of Squares: 248.23
## R-Squared:      0.30872
## Adj. R-Squared: 0.11595
## F-statistic: 17.6159 on 9 and 355 DF, p-value: < 2.22e-16
```

5. Efectos aleatorios

```r
random <- plm(Y ~ X, data=df_panel, model= "random")
summary(random)
```

```
## Oneway (individual) effect Random Effect Model
##    (Swamy-Arora's transformation)
##
## Call:
## plm(formula = Y ~ X, data = df_panel, model = "random")
##
## Balanced Panel: n = 91, T = 5, N = 455
##
## Effects:
##                  var std.dev share
## idiosyncratic 0.6992  0.8362 0.311
## individual    1.5506  1.2452 0.689
## theta: 0.7124
##
## Residuals:
##      Min.   1st Qu.    Median   3rd Qu.      Max.
## -3.952302 -0.376243  0.054623  0.463462  2.395561
##
## Coefficients:
##                 Estimate  Std. Error z-value  Pr(>|z|)
## (Intercept)  -18.6311506   4.2744727 -4.3587 1.308e-05 ***
## Xpbicap        0.0400381   0.2598427  0.1541  0.877542
## Xgdp_gr        0.0269529   0.0096654  2.7886  0.005294 **
## XCPI           0.0232307   0.0123315  1.8839  0.059585 .
## Xinternet      0.0522383   0.0074169  7.0431 1.880e-12 ***
## Xexp_tech      0.0069262   0.0049339  1.4038  0.160381
## Xelect_acc     0.1114421   0.0364746  3.0553  0.002248 **
## Xlife_exp     -0.0571051   0.0375214 -1.5219  0.128025
## Xpoblacion     1.0996936   0.0943987 11.6495 < 2.2e-16 ***
## Xplayers_ppl   0.0306102   0.0064179  4.7695 1.847e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    563.01
## Residual Sum of Squares: 314.1
## R-Squared:      0.44211
## Adj. R-Squared: 0.43082
## Chisq: 352.642 on 9 DF, p-value: < 2.22e-16
```

Test

Hausman test for fijos versus random effects model

```r
phtest(fijos, random)
```

```
##
##  Hausman Test
##
## data:  Y ~ X
## chisq = 19.337, df = 9, p-value = 0.02247
## alternative hypothesis: one model is inconsistent
```