Modelo econometrico suspuestos

Ana Munoz

Pre procesamiento de datos

1. Lectura de datos y formato panel

```
df <- read.csv('df_17_21_noclean.csv') %>%
  dplyr::select(year, country, total_earnings, # orden de datos panel
         total_players,
         pbicap,
         internet,
         desempleo,
         pea, # tech access
         life_exp, # edades
         poblacion,# people
         inflacion
        ) %>%
  arrange(country, decreasing = FALSE)

head(df, 3)
```

```
##         year country total_earnings total_players   pbicap internet desempleo
## 1 2017-01-01 Albania        2868.16             2 4531.032 62.40000     13.62
## 2 2018-01-01 Albania        1346.55             3 5287.661 65.40000     12.30
## 3 2019-01-01 Albania       37459.64            14 5396.214 68.55039     11.47
##       pea life_exp poblacion inflacion
## 1 1958423   79.047   2873457  1.450732
## 2 1951044   79.184   2866376  1.472953
## 3 1937930   79.282   2854191  1.257025
```

2. Valores faltantes

- Numero de Valores faltantes por variable

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

```
##           year        country total_earnings  total_players         pbicap
##              0              0              0              0             10
##       internet      desempleo            pea       life_exp      poblacion
##             54              1              0              0              0
##      inflacion
##              8
```

- corriegiendo los NAs

```r
# 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"))

##"Iran, Islamic Republic of" "United Arab Emirates" "Viet Nam"
# exp_faltantes <- unique(df[is.na(df$exp_tech), ]$country)
# df <- df[!df$country %in% exp_faltantes, ]
df <- df[!df$country %in% c("Iran", "Islamic Republic of", "United Arab Emirates", "Viet Nam"), ]

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

# Migation Hong Kong considerado dentro del gobierno de cina
df <- df[df$country != 'Hong Kong', ]

###########

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

```
##          year        country total_earnings  total_players         pbicap
##             0              0              0              0              0
##      internet      desempleo            pea       life_exp      poblacion
##             0              0              0              0              0
##     inflacion
##             0
```

3. Normalizacion con logaritmo

- valores con varianzas muy grandes
- aplico normalizacion logaritmica en algunas variables

```r
df_standar <- df %>%
  mutate(across(c("total_earnings", "pbicap", "poblacion", "pea", "total_players"), ~log(.)))%>%
  # mutate(across(c("total_earnings", "pbicap", "poblacion", "pea", "total_players"), ~sqrt(.))) %>%
  mutate(year = year(df$year))
```

```r
df_standar$players_ppl <- (df_standar$total_players/df_standar$poblacion)

summary(df_standar)
```

```
##       year        country          total_earnings   total_players
##  Min.   :2017   Length:455         Min.   : 4.006   Min.   :0.000
##  1st Qu.:2018   Class :character   1st Qu.:10.719   1st Qu.:2.944
##  Median :2019   Mode  :character   Median :12.557   Median :4.382
##  Mean   :2019                      Mean   :12.356   Mean   :4.276
##  3rd Qu.:2020                      3rd Qu.:14.102   3rd Qu.:5.598
##  Max.   :2021                      Max.   :17.758   Max.   :8.745
##      pbicap          internet        desempleo            pea
##  Min.   : 7.125   Min.   : 13.78   Min.   : 0.116   Min.   :12.34
##  1st Qu.: 8.440   1st Qu.: 64.71   1st Qu.: 4.050   1st Qu.:15.05
##  Median : 9.402   Median : 78.99   Median : 5.572   Median :15.81
##  Mean   : 9.413   Mean   : 74.60   Mean   : 7.059   Mean   :16.15
##  3rd Qu.:10.372   3rd Qu.: 88.70   3rd Qu.: 8.770   3rd Qu.:17.24
##  Max.   :11.803   Max.   :100.00   Max.   :28.770   Max.   :20.71
##     life_exp       poblacion       inflacion          players_ppl
##  Min.   :62.34   Min.   :12.75   Min.   :-13.911   Min.   :0.0000
##  1st Qu.:72.98   1st Qu.:15.45   1st Qu.:  1.548   1st Qu.:0.1845
##  Median :76.40   Median :16.25   Median :  2.926   Median :0.2755
##  Mean   :76.56   Mean   :16.56   Mean   :  4.998   Mean   :0.2551
##  3rd Qu.:81.30   3rd Qu.:17.66   3rd Qu.:  5.459   3rd Qu.:0.3291
##  Max.   :84.56   Max.   :21.07   Max.   : 56.320   Max.   :0.4460
```

0. Preparando los datos

```r
df_standar %>%
  group_by(year) %>%
  summarise(count = n())
```

```
## # A tibble: 5 x 2
##    year count
##   <dbl> <int>
## 1  2017    91
## 2  2018    91
## 3  2019    91
## 4  2020    91
## 5  2021    91
```

- Tenemos datos panel con la siguente forma 90 paises 5 anios y estas columnas
- Nuestro panel es balanceado y corto

```r
dim(table(df_standar$country,df_standar$year))
```

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

```r
colnames(df_standar)
```

```
##  [1] "year"           "country"        "total_earnings" "total_players"
##  [5] "pbicap"         "internet"       "desempleo"      "pea"
##  [9] "life_exp"       "poblacion"      "inflacion"      "players_ppl"
```

- definimos las variables para el modelo

```r
attach(df_standar)
Y <- cbind(total_earnings)
X <- cbind(pbicap,
           internet,
           life_exp,
           pea,
           desempleo,
           poblacion,
           inflacion,
           # total_players,
           players_ppl
           )

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

head(df_panel,3)
```

```
##             year country total_earnings total_players   pbicap internet
## Albania-2017 2017 Albania       7.961426     0.6931472 8.418705 62.40000
## Albania-2018 2018 Albania       7.205301     1.0986123 8.573131 65.40000
## Albania-2019 2019 Albania      10.531019     2.6390573 8.593453 68.55039
##             desempleo      pea life_exp poblacion inflacion players_ppl
## Albania-2017     13.62 14.48765   79.047  14.87103  1.450732  0.04661058
## Albania-2018     12.30 14.48388   79.184  14.86856  1.472953  0.07388828
## Albania-2019     11.47 14.47713   79.282  14.86430  1.257025  0.17754334
```

1. Efectos Fijos

```r
fijos <- plm(Y ~ X, data=df_panel, index=c('country','year'), model= "within")
summary(fijos)
```

```
## Oneway (individual) effect Within Model
##
## Call:
## plm(formula = Y ~ X, data = df_panel, model = "within", index = c("country",
##     "year"))
```

```
##
## Balanced Panel: n = 91, T = 5, N = 455
##
## Residuals:
##      Min.   1st Qu.    Median   3rd Qu.      Max.
## -2.359274 -0.351382 -0.011891  0.341615  2.818692
##
## Coefficients:
##                Estimate Std. Error t-value  Pr(>|t|)
## Xpbicap      -0.1978908  0.4688713 -0.4221  0.673238
## Xinternet     0.0230482  0.0088170  2.6141  0.009327 **
## Xlife_exp    -0.0446374  0.0452524 -0.9864  0.324603
## Xpea         -1.4899036  6.0467712 -0.2464  0.805517
## Xdesempleo   -0.0835953  0.0361942 -2.3096  0.021480 *
## Xpoblacion    7.3774921  7.4229415  0.9939  0.320958
## Xinflacion    0.0319766  0.0079987  3.9977 7.779e-05 ***
## Xplayers_ppl 16.9586459  1.3315634 12.7359 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    382.45
## Residual Sum of Squares: 185.95
## R-Squared:      0.51379
## Adj. R-Squared: 0.37995
## F-statistic: 47.025 on 8 and 356 DF, p-value: < 2.22e-16
```

2. Efectos aleatorios

```r
random <- plm(Y ~ X, data=df_panel, index=c('country','year'), model= "random")
summary(random)
```

```
## Oneway (individual) effect Random Effect Model
##    (Swamy-Arora's transformation)
##
## Call:
## plm(formula = Y ~ X, data = df_panel, model = "random", index = c("country",
##     "year"))
##
## Balanced Panel: n = 91, T = 5, N = 455
##
## Effects:
##                  var std.dev share
## idiosyncratic 0.5223  0.7227 0.471
## individual    0.5865  0.7658 0.529
## theta: 0.6112
##
## Residuals:
##       Min.    1st Qu.     Median    3rd Qu.       Max.
## -2.8200408 -0.4013407  0.0012484  0.3796440  3.0846382
##
## Coefficients:
##                Estimate Std. Error z-value  Pr(>|z|)
```

```
## (Intercept)   0.6052363  2.1614744  0.2800  0.779469
## Xpbicap      -0.2060340  0.1549485 -1.3297  0.183619
## Xinternet     0.0285038  0.0056996  5.0010 5.704e-07 ***
## Xlife_exp    -0.0072069  0.0271275 -0.2657  0.790494
## Xpea         -2.9089937  1.7626647 -1.6503  0.098874 .
## Xdesempleo   -0.0162704  0.0172916 -0.9409  0.346734
## Xpoblacion    3.2739519  1.7646902  1.8553  0.063560 .
## Xinflacion    0.0192464  0.0071845  2.6789  0.007387 **
## Xplayers_ppl 19.1038529  1.0203692 18.7225 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    760.25
## Residual Sum of Squares: 240.84
## R-Squared:      0.68321
## Adj. R-Squared: 0.67753
## Chisq: 961.864 on 8 DF, p-value: < 2.22e-16
```

3. MCO

```
mco_pool = plm(Y ~ X, data=df_panel,index=c("state", "year"), model="pooling")
mco = lm(Y ~ X, data=df_panel)
# summary(mco)
```

Test para escoger el mejor modelo

1. F test

- H0: modelo (MCO) vs H1: efectos fijos
- p<0.05 entonces rechazo Ho, el mejor modelo seria efectos fijos

```
pFtest(fijos, mco)
```

```
##
##  F test for individual effects
##
## data:  Y ~ X
## F = 6.6253, df1 = 90, df2 = 356, p-value < 2.2e-16
## alternative hypothesis: significant effects
```

2. Breusch-Pagan

- H0: modelo agrupado (MCO) vs H1: efectos aleatorios
- p<0.05 entonces rechazo la Ho, por ahora el mejor modelo seria aleatorios

```
plmtest(mco_pool, type=c("bp"))
```

```
##
##  Lagrange Multiplier Test - (Breusch-Pagan)
##
## data:  Y ~ X
## chisq = 212.47, df = 1, p-value < 2.2e-16
## alternative hypothesis: significant effects
```

3. Hausman test

- H0: efectos aleatorios vs H1: efectos fijos
- p<0.05 entonces rechazo Ho y decido que efectos fijos es mejor

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

```
##
##  Hausman Test
##
## data:  Y ~ X
## chisq = 25.309, df = 8, p-value = 0.001378
## alternative hypothesis: one model is inconsistent
```

Regresiones

Regresieon con efectos fijos

by Country Spain

```r
df_panel$country <- relevel(df_panel$country, ref = "Spain")
regresion_country_sp = lm(Y ~ X + factor(country), data = df_panel)

# summary(regresion_country)


p_values <- summary(regresion_country_sp)$coefficients[,4]
coeficiente <- summary(regresion_country_sp)$coefficients[,1]

no_significativo <- names(p_values)[which(p_values > 0.05)]

significativo_positivos <- names(p_values)[which(p_values < 0.05 & coeficiente>0)]

significativo_negativos <- names(p_values)[which(p_values < 0.05 & coeficiente<0)]
```

Analizando significancias

```r
no_significativo
```

```
##  [1] "(Intercept)"
##  [2] "Xpbicap"
##  [3] "Xlife_exp"
##  [4] "Xpea"
##  [5] "Xpoblacion"
##  [6] "factor(country)Albania"
##  [7] "factor(country)Algeria"
##  [8] "factor(country)Armenia"
##  [9] "factor(country)Australia"
## [10] "factor(country)Austria"
## [11] "factor(country)Azerbaijan"
## [12] "factor(country)Bahrain"
## [13] "factor(country)Belarus"
## [14] "factor(country)Belgium"
## [15] "factor(country)Bolivia"
## [16] "factor(country)Brazil"
## [17] "factor(country)Cambodia"
## [18] "factor(country)Canada"
## [19] "factor(country)Chile"
## [20] "factor(country)China"
## [21] "factor(country)Colombia"
## [22] "factor(country)Costa Rica"
## [23] "factor(country)Croatia"
## [24] "factor(country)Czech Republic"
## [25] "factor(country)Dominican Republic"
## [26] "factor(country)Ecuador"
## [27] "factor(country)Estonia"
## [28] "factor(country)France"
## [29] "factor(country)Georgia"
## [30] "factor(country)Germany"
## [31] "factor(country)Greece"
## [32] "factor(country)Guatemala"
## [33] "factor(country)Hungary"
## [34] "factor(country)Iceland"
## [35] "factor(country)India"
## [36] "factor(country)Indonesia"
## [37] "factor(country)Iraq"
## [38] "factor(country)Ireland"
## [39] "factor(country)Italy"
## [40] "factor(country)Japan"
## [41] "factor(country)Kazakhstan"
## [42] "factor(country)Korea, Republic of"
## [43] "factor(country)Kuwait"
## [44] "factor(country)Kyrgyzstan"
## [45] "factor(country)Lao People's Democratic Republic"
## [46] "factor(country)Latvia"
## [47] "factor(country)Lithuania"
## [48] "factor(country)Luxembourg"
## [49] "factor(country)Malaysia"
## [50] "factor(country)Malta"
## [51] "factor(country)Mexico"
## [52] "factor(country)Moldova, Republic of"
## [53] "factor(country)Mongolia"
## [54] "factor(country)Morocco"
```

```
## [55] "factor(country)Netherlands"
## [56] "factor(country)New Zealand"
## [57] "factor(country)Nicaragua"
## [58] "factor(country)Norway"
## [59] "factor(country)Pakistan"
## [60] "factor(country)Panama"
## [61] "factor(country)Paraguay"
## [62] "factor(country)Peru"
## [63] "factor(country)Philippines"
## [64] "factor(country)Poland"
## [65] "factor(country)Portugal"
## [66] "factor(country)Romania"
## [67] "factor(country)Russian Federation"
## [68] "factor(country)Saudi Arabia"
## [69] "factor(country)Singapore"
## [70] "factor(country)Slovakia"
## [71] "factor(country)Slovenia"
## [72] "factor(country)South Africa"
## [73] "factor(country)Sri Lanka"
## [74] "factor(country)Switzerland"
## [75] "factor(country)Thailand"
## [76] "factor(country)Trinidad and Tobago"
## [77] "factor(country)Tunisia"
## [78] "factor(country)Turkey"
## [79] "factor(country)Ukraine"
## [80] "factor(country)United Kingdom"
## [81] "factor(country)United States"
## [82] "factor(country)Uruguay"
## [83] "factor(country)Uzbekistan"
```

```
significativo_positivos
```

```
##  [1] "Xinternet"
##  [2] "Xinflacion"
##  [3] "Xplayers_ppl"
##  [4] "factor(country)Bosnia and Herzegovina"
##  [5] "factor(country)Bulgaria"
##  [6] "factor(country)Denmark"
##  [7] "factor(country)Finland"
##  [8] "factor(country)Israel"
##  [9] "factor(country)Jordan"
## [10] "factor(country)North Macedonia"
## [11] "factor(country)Sweden"
```

```
significativo_negativos
```

```
## [1] "Xdesempleo"
## [2] "factor(country)Argentina"
## [3] "factor(country)Bangladesh"
## [4] "factor(country)Egypt"
## [5] "factor(country)Iran, Islamic Republic of"
```

Supuestos

- normalidad

```r
# summary(regresion_country_sp)
shapiro.test(resid(regresion_country_sp))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  resid(regresion_country_sp)
## W = 0.98234, p-value = 2.437e-05
```

*homocedasticidad

```r
bptest(regresion_country_sp)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  regresion_country_sp
## BP = 173.14, df = 98, p-value = 4.381e-06
```

```r
dwtest(regresion_country_sp)
```

```
##
##  Durbin-Watson test
##
## data:  regresion_country_sp
## DW = 1.8408, p-value = 3.579e-11
## alternative hypothesis: true autocorrelation is greater than 0
```

- multicolinealidad

```r
vif(regresion_country_sp)
```

```
##                         GVIF Df GVIF^(1/(2*Df))
## X               2.801298e+12  8        5.997358
## factor(country) 2.801298e+12 90        1.172606
```

By year

- El año en sí mismo no parece tener un efecto significativo en Y después de ajustar por X

```r
regresion_years = lm(Y~X+factor(year))
summary(regresion_years)
```

```
##
## Call:
## lm(formula = Y ~ X + factor(year))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.8974 -0.5317 -0.0130  0.5283  3.7601
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       2.033313   1.420147   1.432 0.152919
## Xpbicap          -0.381351   0.111325  -3.426 0.000671 ***
## Xinternet         0.031752   0.005725   5.546 5.04e-08 ***
## Xlife_exp        -0.005552   0.020005  -0.278 0.781517
## Xpea             -2.885339   1.109636  -2.600 0.009627 **
## Xdesempleo       -0.001742   0.011115  -0.157 0.875547
## Xpoblacion        3.221221   1.116247   2.886 0.004096 **
## Xinflacion       -0.010608   0.007218  -1.470 0.142390
## Xplayers_ppl     20.675964   0.809349  25.546  < 2e-16 ***
## factor(year)2018  0.052972   0.156927   0.338 0.735856
## factor(year)2019  0.025855   0.161649   0.160 0.872996
## factor(year)2020 -0.266534   0.168605  -1.581 0.114633
## factor(year)2021  0.138438   0.173222   0.799 0.424607
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.052 on 442 degrees of freedom
## Multiple R-squared:  0.8301, Adjusted R-squared:  0.8255
## F-statistic:   180 on 12 and 442 DF,  p-value: < 2.2e-16
```