Tarea2_Auil_Cabezas

October 14, 2022

Tarea N°2: Data de panel

Lucas Auil - Mario Cabezas / 05-10-2022

0.1 Lectura y reconocimiento de bibliotecas

[1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import statsmodels.api as sm
import statsmodels.formula.api as smf
import sklearn
import scipy
import linearmodels.panel as lmp

%matplotlib inline
```

0.2 Lectura de Datos

[2]:
```
df = pd.read_csv("..\TAREA 2 LAB MAA\charls.csv")
df
```

[2]:

	cesd	child	drinkly	female	hrsusu	hsize	inid	intmonth	\
0	6	2	0.None	1	0.0	4	1.010410e+10	7	
1	7	2	0.None	1	49.0	4	1.010410e+10	7	
2	5	2	0.None	1	56.0	7	1.010410e+10	8	
3	0	2	1.Yes	0	63.0	4	1.010410e+10	7	
4	5	2	1.Yes	0	49.0	4	1.010410e+10	7	
…	…	…	…	…	…	…	…	…	
34366	5	2	0.None	0	0.0	2	3.477630e+11	7	
34367	0	2	1.Yes	0	15.0	2	3.477630e+11	8	
34368	6	1	0.None	1	112.0	2	3.477630e+11	8	
34369	15	2	0.None	1	56.0	2	3.477630e+11	7	
34370	22	2	0.None	1	0.0	2	3.477630e+11	8	

	married	retired	schadj	urban	wave	wealth	age
0	1	0	0	0	1	-5800.0	46
1	1	0	0	0	2	100.0	46

```
2            1        0       0      0     3  -59970.0   46
3            1        0       4      0     1   -5800.0   48
4            1        0       4      0     2     100.0   48
…          …      …       …     …    …       …   …
34366        0        1      16      1     2   68200.0   70
34367        1        0      16      1     3  105400.0   70
34368        1        0       4      1     1       0.0   49
34369        1        0       4      1     2       0.0   49
34370        1        0       4      1     3       0.0   49

[34371 rows x 15 columns]
```

0.2.1 Reconocimiento de la naturaleza de las variables

[3]:
```
df.dtypes
```

[3]:
```
cesd          int64
child         int64
drinkly      object
female        int64
hrsusu      float64
hsize         int64
inid        float64
intmonth      int64
married       int64
retired       int64
schadj        int64
urban         int64
wave          int64
wealth      float64
age           int64
dtype: object
```

0.3 Limpieza de Datos

Al revisar la base de datos entregada, nos damos cuenta que la variable inid viene expuesto en notación cientifica como un numero flotante, lo que implica que existan varios individuos que compartan la misma inid cuando en realidad podría no ser asi. Es por esto que se procede a transformar esta variable a un numero entero.

[4]:
```
df["inid"] = df["inid"].astype("int64")
df
```

[4]:
```
       cesd  child drinkly  female  hrsusu  hsize          inid  intmonth  \
0         6      2  0.None       1     0.0      4   10104101001         7
1         7      2  0.None       1    49.0      4   10104101001         7
2         5      2  0.None       1    56.0      7   10104101001         8
```

```
3         0      2   1.Yes       0    63.0      4   10104101002         7
4         5      2   1.Yes       0    49.0      4   10104101002         7
…     …    …     …     …     …    …           …       …
34366     5      2  0.None       0     0.0      2  347763000000         7
34367     0      2   1.Yes       0    15.0      2  347763000000         8
34368     6      1  0.None       1   112.0      2  347763000000         8
34369    15      2  0.None       1    56.0      2  347763000000         7
34370    22      2  0.None       1     0.0      2  347763000000         8

       married  retired  schadj  urban  wave    wealth  age
0            1        0       0      0     1   -5800.0   46
1            1        0       0      0     2     100.0   46
2            1        0       0      0     3  -59970.0   46
3            1        0       4      0     1   -5800.0   48
4            1        0       4      0     2     100.0   48
…        …       …     …     …    …       …   …
34366        0        1      16      1     2   68200.0   70
34367        1        0      16      1     3  105400.0   70
34368        1        0       4      1     1       0.0   49
34369        1        0       4      1     2       0.0   49
34370        1        0       4      1     3       0.0   49

[34371 rows x 15 columns]
```

Se procede a eliminar aquellos inid que aparezcan más de tres veces(periodos de encuesta) dentro de nuestro dataframe, dado que si esto ocurre, significa que la data está erronea. Tal como pudimos corroborar en archivo de Excel, desde la fila 10.059 en adelante, todos los inid se repiten más de tres veces, por lo que se procede a eliminarlos.

[5]:
```
df.drop(range(10059, 34371, 1),axis=0,inplace= True)
df
```

[5]:
```
       cesd  child drinkly  female  hrsusu  hsize          inid  intmonth  \
0         6      2  0.None       1     0.0      4   10104101001         7
1         7      2  0.None       1    49.0      4   10104101001         7
2         5      2  0.None       1    56.0      7   10104101001         8
3         0      2   1.Yes       0    63.0      4   10104101002         7
4         5      2   1.Yes       0    49.0      4   10104101002         7
…     …    …     …     …     …    …           …       …
10054     5      2  0.None       1     0.0      2   94004308001         7
10055     5      2  0.None       1     0.0      4   94004308001         8
10056     4      4   1.Yes       1    70.0      3  101791000000        10
10057     1      4   1.Yes       1    84.0      2  101791000000        10
10058     7      4   1.Yes       1    28.0      1  101791000000         7

       married  retired  schadj  urban  wave    wealth  age
0            1        0       0      0     1   -5800.0   46
```

```
1            1        0       0      0     2     100.0   46
2            1        0       0      0     3  -59970.0   46
3            1        0       4      0     1   -5800.0   48
4            1        0       4      0     2     100.0   48
…          …      …       …     …    …       …   …
10054        0        1       8      1     2   32500.0   61
10055        0        1       8      1     3       0.0   61
10056        0        0       4      0     1    4000.0   62
10057        0        0       4      0     2  100100.0   62
10058        0        0       4      0     3       0.0   62

[10059 rows x 15 columns]
```

Como podemos notar, la variable drinkly está definida como un object(0.None;1.Yes;.m:missing), sin embargo, se debe trabajar como una variable binaria que tome valores 0 y 1 dependiendo de la respuesta del individuo. Es por esto que se procede tanto a eliminar aquellas observaciones que contengan .m:missing, como a transformar las cadenas de texto en valores numericos 0 o 1

[6]:
```
def parse_values(x):
    if x== "0.None":
       return 0
    elif x=="1.Yes":
       return 1
    else:
       return np.NaN

df['drinkly'] = df['drinkly'].apply(parse_values)

df = df.dropna()
df.reset_index(drop=True, inplace=True)
df["drinkly"] = df["drinkly"].astype("int64")

df
```

```
C:\Users\Hpp\AppData\Local\Temp\ipykernel_7700\401987670.py:13:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  df["drinkly"] = df["drinkly"].astype("int64")
```

[6]:
```
      cesd  child  drinkly  female  hrsusu  hsize          inid  intmonth  \
0        6      2        0       1     0.0      4   10104101001         7
1        7      2        0       1    49.0      4   10104101001         7
2        5      2        0       1    56.0      7   10104101001         8
```

```
3          0      2        1       0    63.0      4   10104101002          7
4          5      2        1       0    49.0      4   10104101002          7
…      …    …      …      …      …    …           …       …
10047      5      2        0       1     0.0      2   94004308001          7
10048      5      2        0       1     0.0      4   94004308001          8
10049      4      4        1       1    70.0      3  101791000000         10
10050      1      4        1       1    84.0      2  101791000000         10
10051      7      4        1       1    28.0      1  101791000000          7

       married  retired  schadj  urban  wave    wealth  age
0            1        0       0      0     1   -5800.0   46
1            1        0       0      0     2     100.0   46
2            1        0       0      0     3  -59970.0   46
3            1        0       4      0     1   -5800.0   48
4            1        0       4      0     2     100.0   48
…        …      …      …    …   …       …  …
10047        0        1       8      1     2   32500.0   61
10048        0        1       8      1     3       0.0   61
10049        0        0       4      0     1    4000.0   62
10050        0        0       4      0     2  100100.0   62
10051        0        0       4      0     3       0.0   62

[10052 rows x 15 columns]
```

[7]:
```
df.set_index(["inid","wave"])
```

[7]:
```
                    cesd  child  drinkly  female  hrsusu  hsize  intmonth  \
inid         wave
10104101001  1         6      2        0       1     0.0      4         7
             2         7      2        0       1    49.0      4         7
             3         5      2        0       1    56.0      7         8
10104101002  1         0      2        1       0    63.0      4         7
             2         5      2        1       0    49.0      4         7
…                  …    …      …      …    …       …       …
94004308001  2         5      2        0       1     0.0      2         7
             3         5      2        0       1     0.0      4         8
101791000000 1         4      4        1       1    70.0      3        10
             2         1      4        1       1    84.0      2        10
             3         7      4        1       1    28.0      1         7

                    married  retired  schadj  urban    wealth  age
inid         wave
10104101001  1            1        0       0      0   -5800.0   46
             2            1        0       0      0     100.0   46
             3            1        0       0      0  -59970.0   46
10104101002  1            1        0       4      0   -5800.0   48
             2            1        0       4      0     100.0   48
```

```
…                  …      …      …      …      …    …
94004308001  2          0      1      8      1   32500.0   61
             3          0      1      8      1       0.0   61
101791000000 1          0      0      4      0    4000.0   62
             2          0      0      4      0  100100.0   62
             3          0      0      4      0       0.0   62

[10052 rows x 13 columns]
```

[8]:
```
#variable construction
X=df[['child','drinkly','female','hrsusu','hsize','intmonth','married','retired','schadj','urb
Xm=(X.groupby(df['inid']).transform('mean'))
Xid=df[['inid','wave','cesd','child','drinkly','female','hrsusu','hsize','intmonth','married',
Xc=pd.DataFrame(np.c_[Xid, Xm],␣
 ↪columns=['inid','wave','cesd','child','drinkly','female','hrsusu','hsize','intmonth','marri

#set panel structure
Xc = Xc.set_index(["inid","wave"])
Xc.describe()
```

[8]:
```
               cesd         child       drinkly        female        hrsusu  \
count  10052.000000  10052.000000  10052.000000  10052.000000  10052.000000
mean       8.865997      2.769200      0.324413      0.542280     27.979457
std        6.289159      1.436222      0.468178      0.498234     27.254901
min        0.000000      0.000000      0.000000      0.000000      0.000000
25%        4.000000      2.000000      0.000000      0.000000      0.000000
50%        8.000000      2.000000      0.000000      1.000000     24.000000
75%       13.000000      3.000000      1.000000      1.000000     49.000000
max       30.000000     10.000000      1.000000      1.000000    168.000000

              hsize      intmonth       married       retired        schadj  \
count  10052.000000  10052.000000  10052.000000  10052.000000  10052.000000
mean       3.652010      7.592917      0.858138      0.268504      4.093315
std        1.784554      1.100565      0.348926      0.443203      3.603198
min        1.000000      1.000000      0.000000      0.000000      0.000000
25%        2.000000      7.000000      1.000000      0.000000      0.000000
50%        3.000000      7.000000      1.000000      0.000000      4.000000
75%        5.000000      8.000000      1.000000      1.000000      4.000000
max       13.000000     12.000000      1.000000      1.000000     16.000000

       …       mfemale       mhrsusu        mhsize     mintmonth  \
count  …  10052.000000  10052.000000  10052.000000  10052.000000
mean   …      0.542280     27.979457      3.652010      7.592917
std    …      0.498234     21.281012      1.459955      0.630721
min    …      0.000000      0.000000      1.000000      5.000000
25%    …      0.000000      8.333333      2.333333      7.333333
50%    …      1.000000     28.000000      3.666667      7.666667
```

```
75%    …       1.000000     43.666667      4.666667      8.000000
max    …       1.000000    119.000000     10.000000     10.000000

           mmarried      mretired       mschadj        murban       mwealth  \
count  10052.000000  10052.000000  10052.000000  10052.000000  1.005200e+04
mean       0.858138      0.268504      4.093315      0.315559  1.020345e+04
std        0.332874      0.365762      3.603198      0.464761  6.280248e+04
min        0.000000      0.000000      0.000000      0.000000 -3.250000e+05
25%        1.000000      0.000000      0.000000      0.000000  8.333333e+01
50%        1.000000      0.000000      4.000000      0.000000  1.073333e+03
75%        1.000000      0.333333      4.000000      1.000000  8.666667e+03
max        1.000000      1.000000     16.000000      1.000000  2.672550e+06

               mage
count  10052.000000
mean      58.225627
std        9.234432
min       16.000000
25%       51.000000
50%       58.000000
75%       64.000000
max       89.000000

[8 rows x 25 columns]
```

[9]:
```
corr_df = df.corr()
print(corr_df)
```

```
             cesd     child   drinkly    female    hrsusu     hsize  \
cesd      1.000000  0.069174 -0.072919  0.160518 -0.008703 -0.020662
child     0.069174  1.000000 -0.078771  0.043810 -0.150272  0.037097
drinkly  -0.072919 -0.078771  1.000000 -0.419436  0.152972 -0.015267
female    0.160518  0.043810 -0.419436  1.000000 -0.112730  0.010619
hrsusu   -0.008703 -0.150272  0.152972 -0.112730  1.000000  0.038030
hsize    -0.020662  0.037097 -0.015267  0.010619  0.038030  1.000000
inid     -0.046739 -0.006969  0.018528  0.013948 -0.025834 -0.083519
intmonth  0.009928  0.028297 -0.004538  0.005445 -0.006881  0.039464
married  -0.105279 -0.115968  0.058841 -0.099413  0.170328  0.183552
retired   0.013802  0.157542 -0.146527  0.097495 -0.621993 -0.075573
schadj   -0.186930 -0.223085  0.153444 -0.286892  0.031213 -0.033322
urban    -0.147690 -0.116395 -0.025166  0.033895 -0.119775 -0.045365
wave     -0.002236  0.068482 -0.011197  0.000153 -0.085929 -0.154840
wealth   -0.057655 -0.026605  0.013471 -0.012055 -0.021799 -0.019001
age       0.027725  0.451290 -0.013411 -0.112337 -0.281442 -0.184091

             inid  intmonth   married   retired    schadj     urban  \
cesd     -0.046739  0.009928 -0.105279  0.013802 -0.186930 -0.147690
```

```
child    -0.006969  0.028297 -0.115968  0.157542 -0.223085 -0.116395
drinkly   0.018528 -0.004538  0.058841 -0.146527  0.153444 -0.025166
female    0.013948  0.005445 -0.099413  0.097495 -0.286892  0.033895
hrsusu   -0.025834 -0.006881  0.170328 -0.621993  0.031213 -0.119775
hsize    -0.083519  0.039464  0.183552 -0.075573 -0.033322 -0.045365
inid      1.000000 -0.084800 -0.030872  0.104465  0.179127  0.125267
intmonth -0.084800  1.000000  0.002202  0.008101 -0.029258  0.131736
married  -0.030872  0.002202  1.000000 -0.198228  0.143319 -0.028843
retired   0.104465  0.008101 -0.198228  1.000000  0.015211  0.210258
schadj    0.179127 -0.029258  0.143319  0.015211  1.000000  0.189405
urban     0.125267  0.131736 -0.028843  0.210258  0.189405  1.000000
wave     -0.000228 -0.187438 -0.037160  0.079664  0.000009 -0.000014
wealth    0.057163 -0.016191  0.034759  0.039538  0.084674  0.053689
age       0.117847 -0.000350 -0.291062  0.316317 -0.196876  0.007518

                wave    wealth       age
cesd     -0.002236 -0.057655  0.027725
child     0.068482 -0.026605  0.451290
drinkly  -0.011197  0.013471 -0.013411
female    0.000153 -0.012055 -0.112337
hrsusu   -0.085929 -0.021799 -0.281442
hsize    -0.154840 -0.019001 -0.184091
inid     -0.000228  0.057163  0.117847
intmonth -0.187438 -0.016191 -0.000350
married  -0.037160  0.034759 -0.291062
retired   0.079664  0.039538  0.316317
schadj    0.000009  0.084674 -0.196876
urban    -0.000014  0.053689  0.007518
wave      1.000000  0.042499 -0.000031
wealth    0.042499  1.000000  0.004422
age      -0.000031  0.004422  1.000000
```

0.4 Pooled OLS

Dentro de las variables que decidimos no incluir en nuestro modelo se encuentra la variable “retired”, dado que posee una alta correlacion con la variable “hrsusu”, lo cual tiene sentido ya que, si una persona está pensionada, es más probable que tenga menos horas de trabajo semanal.

[10]:
```
y=Xc['cesd']
X=Xc[['child','drinkly','female','hrsusu','hsize','intmonth','married','schadj','urban','wealt
X=sm.add_constant(X)

model = sm.OLS(y, X)
results = model.fit()
print(results.summary())
```

```
                            OLS Regression Results
==============================================================================
```

```
Dep. Variable:                   cesd   R-squared:                       0.072
Model:                           OLS   Adj. R-squared:                  0.071
Method:                Least Squares   F-statistic:                     70.83
Date:               Wed, 05 Oct 2022   Prob (F-statistic):          7.07e-154
Time:                       19:54:12   Log-Likelihood:                -32371.
No. Observations:              10052   AIC:                         6.477e+04
Df Residuals:                  10040   BIC:                         6.485e+04
Df Model:                         11
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const         10.1590      0.732     13.887      0.000       8.725      11.593
child          0.0913      0.049      1.876      0.061      -0.004       0.187
drinkly       -0.0347      0.144     -0.241      0.809      -0.316       0.247
female         1.5442      0.143     10.810      0.000       1.264       1.824
hrsusu         0.0014      0.002      0.571      0.568      -0.003       0.006
hsize         -0.0792      0.035     -2.245      0.025      -0.148      -0.010
intmonth       0.1334      0.056      2.398      0.016       0.024       0.242
married       -1.4098      0.186     -7.582      0.000      -1.774      -1.045
schadj        -0.1928      0.019    -10.266      0.000      -0.230      -0.156
urban         -1.7883      0.137    -13.087      0.000      -2.056      -1.520
wealth     -2.294e-06   6.11e-07     -3.753      0.000   -3.49e-06    -1.1e-06
age           -0.0094      0.008     -1.140      0.254      -0.026       0.007
==============================================================================
Omnibus:                      671.434   Durbin-Watson:                   1.335
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              813.921
Skew:                           0.695   Prob(JB):                    1.82e-177
Kurtosis:                       3.100   Cond. No.                     1.22e+06
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
[2] The condition number is large, 1.22e+06. This might indicate that there are
strong multicollinearity or other numerical problems.
```

[11]:
```
model=lmp.PooledOLS(y,X)
OLS=model.fit(cov_type="robust")
print(OLS)
```

```
                          PooledOLS Estimation Summary
==============================================================================
Dep. Variable:                   cesd   R-squared:                      0.0720
Estimator:                  PooledOLS   R-squared (Between):            0.1087
No. Observations:              10052   R-squared (Within):            -0.0004
Date:               Wed, Oct 05 2022   R-squared (Overall):            0.0720
Time:                       19:54:12   Log-likelihood              -3.237e+04
```

Cov. Estimator:	Robust		
		F-statistic:	70.833
Entities:	3353	P-value	0.0000
Avg Obs:	2.9979	Distribution:	F(11,10040)
Min Obs:	2.0000		
Max Obs:	3.0000	F-statistic (robust):	69.495
		P-value	0.0000
Time periods:	3	Distribution:	F(11,10040)
Avg Obs:	3350.7		
Min Obs:	3349.0		
Max Obs:	3353.0		

Parameter Estimates

	Parameter	Std. Err.	T-stat	P-value	Lower CI	Upper CI
const	10.159	0.7392	13.742	0.0000	8.7100	11.608
child	0.0913	0.0498	1.8313	0.0671	-0.0064	0.1889
drinkly	-0.0347	0.1420	-0.2441	0.8072	-0.3129	0.2436
female	1.5442	0.1432	10.784	0.0000	1.2635	1.8249
hrsusu	0.0014	0.0024	0.5646	0.5723	-0.0034	0.0061
hsize	-0.0792	0.0346	-2.2882	0.0221	-0.1470	-0.0113
intmonth	0.1334	0.0542	2.4597	0.0139	0.0271	0.2397
married	-1.4098	0.2015	-6.9951	0.0000	-1.8049	-1.0148
schadj	-0.1928	0.0182	-10.564	0.0000	-0.2285	-0.1570
urban	-1.7883	0.1310	-13.653	0.0000	-2.0450	-1.5315
wealth	-2.294e-06	1.93e-06	-1.1890	0.2345	-6.076e-06	1.488e-06
age	-0.0094	0.0084	-1.1158	0.2645	-0.0259	0.0071

0.5 Interpretación de datos Pooled OLS

Los resultados mediante la estimacion Pooled OLS nos entrega que las variables que predicen el comportamiento del puntaje en la escala de salud mental de manera significativa son female(binaria), married(binaria),schadj y urban(binaria).

Para la variable female tenemos que si el individuo encuestado es mujer(female=1), aumenta el puntaje en la escala de su salud mental, en comparacion al caso contrario de los hombres(female=0). Esto se puede explicar porque generalmente son las mujeres quienes tienen más normalizado pedir ayuda cuando se sienten mal o tienen problemas de cualquier indole, mientras que los hombres generalmente se reservan un poco más sus emociones en situaciones dificiles y no buscan ayuda por su propia cuenta, lo que podría afectar en mayor medida su salud mental.

Para el caso de la variable urban(1=urbano,0=rural) notamos que afecta en gran medida el puntaje en la escala de salud mental de los encuestados. Si el individuo vive en una zona urbana, se encontrará sometido a circunstancias más estresantes que la gente que vive en zonas rurales, esto debido a muchos factores tales como el tráfico, la contaminacion acustica, la delincuencia, el comportamiento de las personas que conviven en zonas urbanas, etc). Caso similar ocurre con la variable married(1=casado,0=no casado), ya que el individuo que se encuentre casado podría estar

mas sometido a situaciones de estres por el hecho de compartir su vida con una persona o por tener hijos con su pareja, mientras que una persona no casada posee más libertad al no tener mas preocupaciones que su propia vida.

En el caso de la variable schadj, existe una correlación inversa con respecto a la variable de puntaje en la escala de salud mental. Esto dado a que mientras una persona posea menor años de escolaridad(individuo generalmente más joven) tiene menos responsabilidades y no poseen mayores preocupaciones lo que provocaría un mayor puntaje en la escala de salud mental. Lo mismo ocurre para el caso contrario una persona con una cantidad mayor de años de escolaridad poseerá un menor puntaje en la escala de salud mental, debido a las nuevas responsabilidades que va adquiriendo a través del tiempo

0.6 Fixed Effects

[12]:
```
pd.options.display.max_columns = None
df.groupby(["wave"]).describe()
```

[12]:

wave	cesd								child								\
	count	mean	std	min	25%	50%	75%	max	count	mean							
1	3353.0	9.024456	6.418574	0.0	4.0	8.0	13.0	30.0	3353.0	2.651655							
2	3349.0	8.583159	5.821829	0.0	4.0	7.0	12.0	30.0	3349.0	2.763511							
3	3350.0	8.990149	6.593080	0.0	4.0	8.0	13.0	30.0	3350.0	2.892537							

wave	drinkly										\
	std	min	25%	50%	75%	max	count	mean	std	min	
1	1.412856	0.0	2.0	2.0	3.0	10.0	3353.0	0.329556	0.470122	0.0	
2	1.427112	0.0	2.0	2.0	3.0	10.0	3349.0	0.326963	0.469174	0.0	
3	1.458628	0.0	2.0	3.0	4.0	10.0	3350.0	0.316716	0.465265	0.0	

wave	female											\
	25%	50%	75%	max	count	mean	std	min	25%	50%	75%	max
1	0.0	0.0	1.0	1.0	3353.0	0.542201	0.498290	0.0	0.0	1.0	1.0	1.0
2	0.0	0.0	1.0	1.0	3349.0	0.542251	0.498286	0.0	0.0	1.0	1.0	1.0
3	0.0	0.0	1.0	1.0	3350.0	0.542388	0.498274	0.0	0.0	1.0	1.0	1.0

wave	hrsusu								hsize	\
	count	mean	std	min	25%	50%	75%	max	count	
1	3353.0	30.642410	27.894781	0.0	0.0	30.0	56.0	144.0	3353.0	
2	3349.0	28.387130	26.808306	0.0	0.0	25.0	49.0	140.0	3349.0	
3	3350.0	24.906567	26.746553	0.0	0.0	18.0	42.0	168.0	3350.0	

wave	inid							\
	mean	std	min	25%	50%	75%	max	
	count	mean						

```
1     3.838950  1.858271  1.0  2.0  4.0  5.0  13.0  3353.0  4.891483e+10
2     3.954912  1.946123  1.0  2.0  4.0  5.0  13.0  3349.0  4.889959e+10
3     3.162090  1.395829  1.0  2.0  3.0  4.0  11.0  3350.0  4.890203e+10

                                                                        \
              std           min           25%           50%           75%
wave
1     2.297229e+10  1.010410e+10  3.110611e+10  5.630230e+10  6.403312e+10
2     2.298022e+10  1.010410e+10  3.110611e+10  5.630230e+10  6.403312e+10
3     2.297713e+10  1.010410e+10  3.110611e+10  5.630230e+10  6.403312e+10

                  intmonth                                              \
              max    count      mean       std  min  25%  50%  75%   max
wave
1     1.017910e+11   3353.0  7.801074  1.374451  1.0  7.0  8.0  8.0  12.0
2     1.017910e+11   3349.0  7.681696  0.911674  7.0  7.0  7.0  8.0  12.0
3     1.017910e+11   3350.0  7.295821  0.879724  1.0  7.0  7.0  8.0   9.0

     married                                              retired           \
       count      mean       std  min  25%  50%  75%  max   count      mean
wave
1     3353.0  0.874441  0.331401  0.0  1.0  1.0  1.0  1.0  3353.0  0.226961
2     3349.0  0.857271  0.349849  0.0  1.0  1.0  1.0  1.0  3349.0  0.265154
3     3350.0  0.842687  0.364150  0.0  1.0  1.0  1.0  1.0  3350.0  0.313433

                                        schadj                            \
           std  min  25%  50%  75%  max   count      mean       std  min  25%
wave
1     0.418930  0.0  0.0  0.0  0.0  1.0  3353.0  4.092455  3.603206  0.0  0.0
2     0.441481  0.0  0.0  0.0  1.0  1.0  3349.0  4.094954  3.603969  0.0  0.0
3     0.463958  0.0  0.0  0.0  1.0  1.0  3350.0  4.092537  3.603493  0.0  0.0

                    urban                                               \
      50%  75%   max   count      mean       std  min  25%  50%  75%  max
wave
1     4.0  4.0  16.0  3353.0  0.315538  0.464799  0.0  0.0  0.0  1.0  1.0
2     4.0  4.0  16.0  3349.0  0.315617  0.464830  0.0  0.0  0.0  1.0  1.0
3     4.0  4.0  16.0  3350.0  0.315522  0.464793  0.0  0.0  0.0  1.0  1.0

      wealth                                                            \
       count          mean            std       min  25%    50%     75%
wave
1     3353.0   5802.358789   54503.111517 -975000.0  0.0  300.0  2200.0
2     3349.0   8654.986115   51929.828046 -499500.0  0.0  300.0  4300.0
3     3350.0  16156.485672  154820.770071 -596000.0  0.0  500.0  7000.0

                   age
```

	max	count	mean	std	min	25%	50%	75%	max
wave									
1	900100.0	3353.0	58.226663	9.236190	16.0	51.0	58.0	64.0	89.0
2	1001500.0	3349.0	58.224246	9.235091	16.0	51.0	58.0	64.0	89.0
3	8001500.0	3350.0	58.225970	9.234769	16.0	51.0	58.0	64.0	89.0

Al analizar las medias de cada variable en los 3 periodos de estudio(wave), podemos ver que las variables que cambian en el tiempo son: cesd,child,drinkly,hrsusu,hsize,intmonth,married,retired,wealth y aquellas que no cambian en el tiempo(o su cambio es constante) son female,schadj,urban,age, las cuales fueron eliminadas al aplicar el modelo de efectos fijos

[13]:
```
X=Xc[['wealth','child','drinkly','hrsusu','hsize','intmonth','married','retired']]
X=sm.add_constant(X)
model=lmp.PanelOLS(y,X, entity_effects=True)
fe=model.fit(cov_type="robust")
print(fe)
```

```
                          PanelOLS Estimation Summary
================================================================================
Dep. Variable:                   cesd   R-squared:                        0.0040
Estimator:                   PanelOLS   R-squared (Between):              0.0124
No. Observations:               10052   R-squared (Within):               0.0040
Date:                Wed, Oct 05 2022   R-squared (Overall):              0.0096
Time:                        19:54:13   Log-likelihood                -2.724e+04
Cov. Estimator:                Robust
                                        F-statistic:                      3.3229
Entities:                        3353   P-value                           0.0008
Avg Obs:                       2.9979   Distribution:                  F(8,6691)
Min Obs:                       2.0000
Max Obs:                       3.0000   F-statistic (robust):             2.8125
                                        P-value                           0.0041
Time periods:                       3   Distribution:                  F(8,6691)
Avg Obs:                       3350.7
Min Obs:                       3349.0
Max Obs:                       3353.0


                             Parameter Estimates
================================================================================
            Parameter  Std. Err.     T-stat    P-value    Lower CI    Upper CI
--------------------------------------------------------------------------------
const          9.9040     0.6695     14.792     0.0000      8.5915      11.217
wealth     -5.209e-07  8.434e-07    -0.6176     0.5368  -2.174e-06  1.132e-06
child          0.1468     0.0960     1.5290     0.1263     -0.0414      0.3350
drinkly        0.2027     0.1887     1.0746     0.2826     -0.1671      0.5725
hrsusu        -0.0003     0.0029    -0.0997     0.9206     -0.0060      0.0054
hsize         -0.1207     0.0442    -2.7326     0.0063     -0.2073     -0.0341
intmonth      -0.0184     0.0507    -0.3629     0.7167     -0.1177      0.0810
married       -1.1814     0.5066    -2.3321     0.0197     -2.1744     -0.1884
```

```
retired        0.3623     0.2020     1.7935     0.0729     -0.0337      0.7583
==============================================================================

F-test for Poolability: 3.8461
P-value: 0.0000
Distribution: F(3352,6691)

Included effects: Entity
```

0.7 Interpretación de datos Fixed Effects

Al realizar la estimación mediante Fixed Effects notamos que la prueba F-statics arroja un valor de 3,6365 el cual nos indica la capacidad explicativa que tiene el grupo de variables independientes seleccionadas sobre la variación de la variable dependiente cesd. Este valor nos indica que la estimación hecha es significativa, lo cual también se corrobora con el el valor P-value de 0,0006.

En cuanto a la interpretación de los resultados, podemos observar que al sacar las variables age, urban, schadj y female (eliminadas por no variar en el tiempo) las nuevas variables significativas corresponden a hsize(tamaño del hogar) y married(1=casado, 0= no casado). Se puede decir nuevamente que la variable married afecta de manera siginificativa a el puntaje en la escala de salud mental, puesto que el hecho de compartir nuestras vidas con una persona puede generar estrés o muchas veces afectar nuestra salud mental, no asi el caso de las personas que no se encuentran casadas, las cuales deberían poseer un puntaje mayor en la escala de sauld mental.

0.8 Random Effects

[14]:
```
model=lmp.RandomEffects(y,X)
re=model.fit(cov_type="robust")
print(re)
```

```
                          RandomEffects Estimation Summary
================================================================================
Dep. Variable:                    cesd   R-squared:                        0.0104
Estimator:                RandomEffects   R-squared (Between):              0.0283
No. Observations:                 10052   R-squared (Within):               0.0009
Date:                 Wed, Oct 05 2022   R-squared (Overall):              0.0191
Time:                         19:54:14   Log-likelihood                -2.931e+04
Cov. Estimator:                 Robust
                                         F-statistic:                      13.143
Entities:                         3353   P-value                           0.0000
Avg Obs:                        2.9979   Distribution:                  F(8,10043)
Min Obs:                        2.0000
Max Obs:                        3.0000   F-statistic (robust):             11.115
                                         P-value                           0.0000
Time periods:                        3   Distribution:                  F(8,10043)
Avg Obs:                        3350.7
Min Obs:                        3349.0
Max Obs:                        3353.0
```

Parameter Estimates

	Parameter	Std. Err.	T-stat	P-value	Lower CI	Upper CI
const	9.9183	0.4652	21.323	0.0000	9.0065	10.830
wealth	-1.544e-06	1.128e-06	-1.3689	0.1711	-3.754e-06	6.668e-07
child	0.2278	0.0547	4.1649	0.0000	0.1206	0.3350
drinkly	-0.4273	0.1428	-2.9932	0.0028	-0.7072	-0.1475
hrsusu	0.0017	0.0026	0.6620	0.5080	-0.0033	0.0068
hsize	-0.0720	0.0352	-2.0434	0.0410	-0.1410	-0.0029
intmonth	0.0022	0.0463	0.0474	0.9622	-0.0886	0.0930
married	-1.5809	0.2486	-6.3596	0.0000	-2.0682	-1.0937
retired	0.1003	0.1726	0.5810	0.5612	-0.2381	0.4387

0.9 Interpretacion de datos Random Effects

Al revisar los resultados de esta estimación, nos podemos dar cuenta que la prueba F-statistic nos entrega un valor de $p<0.05$, lo que nos indica que los estimadores o variables explicativas del modelo son significativas, por lo tanto la regresion también lo es. En cuanto a la interpretacion de los parametros, notamos que las variable intmonth no es significativa dado su valor $p=0.9622>0.05$, no asi la variable child, la cual es una variable altamente significativa debido a su valor p. Esta significancia tiene sentido si pensamos que probablemente aquellos individuos que tienen mayor cantidad de hijos estan expuestos a mayores niveles de estres por factores economicos(mientras mas hijos mas dificil mantenerlos economicamente), por preocupacion de que les pase algo, problemas que pueden dar, entre otros.

[15]:
```
re.variance_decomposition
```

[15]:
```
Effects                   18.631084
Residual                  19.857687
Percent due to Effects     0.484065
Name: Variance Decomposition, dtype: float64
```

0.10 Model comparison

[16]:
```
print(lmp.compare({"FE": fe, "RE": re, "Pooled": OLS}))
```

Model Comparison

	FE	RE	Pooled
Dep. Variable	cesd	cesd	cesd
Estimator	PanelOLS	RandomEffects	PooledOLS
No. Observations	10052	10052	10052
Cov. Est.	Robust	Robust	Robust
R-squared	0.0040	0.0104	0.0720
R-Squared (Within)	0.0040	0.0009	-0.0004

R-Squared (Between)	0.0124	0.0283	0.1087
R-Squared (Overall)	0.0096	0.0191	0.0720
F-statistic	3.3229	13.143	70.833
P-value (F-stat)	0.0008	0.0000	0.0000
const	9.9040	9.9183	10.159
	(14.792)	(21.323)	(13.742)
wealth	-5.209e-07	-1.544e-06	-2.294e-06
	(-0.6176)	(-1.3689)	(-1.1890)
child	0.1468	0.2278	0.0913
	(1.5290)	(4.1649)	(1.8313)
drinkly	0.2027	-0.4273	-0.0347
	(1.0746)	(-2.9932)	(-0.2441)
hrsusu	-0.0003	0.0017	0.0014
	(-0.0997)	(0.6620)	(0.5646)
hsize	-0.1207	-0.0720	-0.0792
	(-2.7326)	(-2.0434)	(-2.2882)
intmonth	-0.0184	0.0022	0.1334
	(-0.3629)	(0.0474)	(2.4597)
married	-1.1814	-1.5809	-1.4098
	(-2.3321)	(-6.3596)	(-6.9951)
retired	0.3623	0.1003	
	(1.7935)	(0.5810)	
female			1.5442
			(10.784)
schadj			-0.1928
			(-10.564)
urban			-1.7883
			(-13.653)
age			-0.0094
			(-1.1158)
Effects	Entity		

T-stats reported in parentheses

0.11 Comparación resultados obtenidos en 2,3 y 4

La principal diferencia entre los resultados entregados tanto en Pooled OLS, Fixed Effects y Random Effects se encuentra en las variables que se utilizaron para desarrollar el modelo. Comparando los valores de la prueba F y los valores p de significancia, encontramos que los mayores valores los posee la estimacion Pooled OLS, por ende es el modelo que mejor se ajusta a la data existente, dado que toma en cuenta la variabilidad existente tanto por los efectos(variables)fijos como los aleatorios. Además que este modelo permite estimar el efecto promedio, asumiendo que el tiempo afecta a todas las unidades de la misma forma y la heterogeneidad individual no impacta la relación de interes.

[17]:
```
import numpy.linalg as la
from scipy import stats

def hausman(fe, re):
 diff = fe.params-re.params
 psi = fe.cov - re.cov
 dof = diff.size -1
 W = diff.dot(la.inv(psi)).dot(diff)
 pval = stats.chi2.sf(W, dof)
 return W, dof, pval
```

[21]:
```
htest = hausman(fe, re)
print("Hausman Test: chi-2 = {0}, df = {1}, p-value = {2}".format(htest[0],␣
 ↪htest[1], htest[2]))
```

```
Hausman Test: chi-2 = 42.44907334158654, df = 8, p-value =
1.1152088096649905e-06
```

Como el valor p del test de Haussman es muy cercano a cero, es mejor utilizar el modelo de efectos fijos sobre los efectos aleatorios. Esto tiene sentido debido a la existencia de las diferencias entre los coeficientes de cada muestra.

0.12 Correlated Random Effects

[20]:
```
X=Xc[['wealth','child','drinkly','hrsusu','hsize','intmonth','married','retired','mwealth','mo
X=sm.add_constant(X)
model=lmp.RandomEffects(y,X)
cre=model.fit(cov_type="robust")
print(cre)
```

```
                        RandomEffects Estimation Summary
================================================================================
Dep. Variable:                   cesd   R-squared:                        0.0168
Estimator:                RandomEffects   R-squared (Between):              0.0414
No. Observations:                 10052   R-squared (Within):               0.0040
Date:                Wed, Oct 05 2022   R-squared (Overall):              0.0289
Time:                        20:18:50   Log-likelihood                -2.927e+04
Cov. Estimator:                Robust
                                        F-statistic:                      10.743
Entities:                        3353   P-value                           0.0000
Avg Obs:                       2.9979   Distribution:                F(16,10035)
Min Obs:                       2.0000
Max Obs:                       3.0000   F-statistic (robust):             8.9074
                                        P-value                           0.0000
Time periods:                       3   Distribution:                F(16,10035)
Avg Obs:                       3350.7
Min Obs:                       3349.0
Max Obs:                       3353.0
```

Parameter Estimates

	Parameter	Std. Err.	T-stat	P-value	Lower CI	Upper CI
const	8.6765	1.1272	7.6974	0.0000	6.4669	10.886
wealth	-5.209e-07	1.533e-06	-0.3397	0.7341	-3.527e-06	2.485e-06
child	0.1468	0.1006	1.4586	0.1447	-0.0505	0.3441
drinkly	0.2027	0.1907	1.0630	0.2878	-0.1711	0.5765
hrsusu	-0.0003	0.0029	-0.0993	0.9209	-0.0060	0.0054
hsize	-0.1207	0.0445	-2.7134	0.0067	-0.2079	-0.0335
intmonth	-0.0184	0.0493	-0.3732	0.7090	-0.1150	0.0782
married	-1.1814	0.4871	-2.4255	0.0153	-2.1361	-0.2266
retired	0.3623	0.2023	1.7909	0.0733	-0.0342	0.7588
mwealth	-6.708e-06	3.904e-06	-1.7185	0.0857	-1.436e-05	9.436e-07
mchild	0.0965	0.1218	0.7921	0.4283	-0.1423	0.3352
mdrinkly	-1.5083	0.2866	-5.2629	0.0000	-2.0701	-0.9465
mhrsusu	0.0088	0.0066	1.3480	0.1777	-0.0040	0.0217
mhsize	0.1325	0.0769	1.7239	0.0848	-0.0182	0.2832
mintmonth	0.2052	0.1478	1.3886	0.1650	-0.0845	0.4949
mmarried	-0.6785	0.5740	-1.1822	0.2372	-1.8036	0.4465
mretired	-0.7012	0.4047	-1.7325	0.0832	-1.4946	0.0922

Este modelo no es adecuado para capturar la heterogeneidad no observada, debido a que los valores p de las medias de las variables incluidas en este modelo no son significativas.

[22]:
```
print(lmp.compare({"FE": fe, "RE": re, "CRE": cre}))
```

Model Comparison

	FE	RE	CRE
Dep. Variable	cesd	cesd	cesd
Estimator	PanelOLS	RandomEffects	RandomEffects
No. Observations	10052	10052	10052
Cov. Est.	Robust	Robust	Robust
R-squared	0.0040	0.0104	0.0168
R-Squared (Within)	0.0040	0.0009	0.0040
R-Squared (Between)	0.0124	0.0283	0.0414
R-Squared (Overall)	0.0096	0.0191	0.0289
F-statistic	3.3229	13.143	10.743
P-value (F-stat)	0.0008	0.0000	0.0000
const	9.9040	9.9183	8.6765
	(14.792)	(21.323)	(7.6974)
wealth	-5.209e-07	-1.544e-06	-5.209e-07
	(-0.6176)	(-1.3689)	(-0.3397)

```
child                        0.1468          0.2278          0.1468
                            (1.5290)        (4.1649)        (1.4586)
drinkly                      0.2027         -0.4273          0.2027
                            (1.0746)        (-2.9932)       (1.0630)
hrsusu                      -0.0003          0.0017         -0.0003
                            (-0.0997)       (0.6620)        (-0.0993)
hsize                       -0.1207         -0.0720         -0.1207
                            (-2.7326)       (-2.0434)       (-2.7134)
intmonth                    -0.0184          0.0022         -0.0184
                            (-0.3629)       (0.0474)        (-0.3732)
married                     -1.1814         -1.5809         -1.1814
                            (-2.3321)       (-6.3596)       (-2.4255)
retired                      0.3623          0.1003          0.3623
                            (1.7935)        (0.5810)        (1.7909)
mwealth                                                    -6.708e-06
                                                           (-1.7185)
mchild                                                      0.0965
                                                           (0.7921)
mdrinkly                                                   -1.5083
                                                           (-5.2629)
mhrsusu                                                     0.0088
                                                           (1.3480)
mhsize                                                      0.1325
                                                           (1.7239)
mintmonth                                                   0.2052
                                                           (1.3886)
mmarried                                                   -0.6785
                                                           (-1.1822)
mretired                                                   -0.7012
                                                           (-1.7325)
======================= ============== ================= =================
Effects                        Entity
-------------------------------------------------------------------------

T-stats reported in parentheses
```

Preferimos el modelo CRE debido a que es el más utilizado y es el que se puede agregar mayor informacion, debido a que se puede integrar las variables que son constantes en el tiempo, las cuales no se pueden incluir en el modelo de Efectos Fijos

Tarea 2

Instrucciones

Los resultados de los ejericicios propuestos se deben entregar como un notebook por correo electronico a *juan.caro@uni.lu* el dia 3/10 hasta las 21:00.

Es importante considerar que el código debe poder ejecutarse en cualquier computadora con la data original del repositorio. Recordar la convencion para el nombre de archivo ademas de incluir en su documento titulos y encabezados por seccion. La data a utilizar es **charls.csv**.

Las variables tienen la siguiente descripcion:

- INID: identificador unico
- wave: periodo de la encuesta (1-3)
- cesd: puntaje en la escala de salud mental (0-30)
- child: numero de hijos
- drinkly: bebio alcohol en el ultimo mes (binario)
- hrsusu: horas promedio trabajo semanal
- hsize: tamano del hogar
- intmonth: mes en que fue encuestado/a (1-12)
- married: si esta casado/a (binario)
- retired: si esta pensionado/a (binario)
- schadj: años de escolaridad
- urban: zona urbana (binario)
- wealth: riqueza neta (miles RMB)
- age: edad al entrar a la encuesta (no varia entre periodos)

Preguntas:

1. Cargar la base de datos *charls.csv* en el ambiente. Identifique los tipos de datos que se encuentran en la base, realice estadisticas descriptivas sobre las variables importantes (Hint: Revisar la distribuciones, datos faltantes, outliers, etc.) y limpie las variables cuando sea necesario.

2. Ejecute un modelo Pooled OLS para explicar el puntaje en la escala de salud mental (CESD). Seleccione las variables dependientes a incluir en el modelo final e interprete su significado.

3. Ejecute un modelo de efectos fijos para explicar el puntaje en la escala de salud mental (CESD). Seleccione las variables dependientes a incluir en el modelo final e interprete su significado.

4. Ejecute un modelo de efectos aleatorios para explicar el puntaje en la escala de salud mental (CESD). Seleccione las variables dependientes a incluir en el modelo final e interprete su significado.

5. Comente los resultados obtenidos en 2, 3 y 4. ¿Cuáles y por qué existen las diferencias entre los resultados?. En su opinión, ¿Cuál sería el más adecuado para responder la pregunta de investgación y por qué? ¿Qué variables resultaron ser robustas a la especificación?

6. Ejecute un modelo de efectos aleatorios correlacionados (CRE) para explicar el puntaje en la escala de salud mental (CESD). Seleccione las variables dependientes a incluir en el modelo final e interprete su significado. Es este modelo adecuado, dada la data disponible, para modelar el componente no observado?

7. Usando el modelo CRE, prediga la distribucion del componente no observado. Que puede inferir respecto de la heterogeneidad fija en el tiempo y su impacto en el puntaje CESD?

8. Usando sus respuestas anteriores, que modelo prefiere? que se puede inferir en general respecto del efecto de las variables explicativas sobre el puntaje CESD?