Supplementary Material

S.M.1. Python programming scripts for the IV2SLS test

```
import pandas as pd
import numpy as np
import statsmodels.api as sm
from statsmodels.sandbox.regression.gmm import IV2SLS

# 1. Load the data
file_path = "Dados juntos.xlsx"  # File name
sheet_name = "Plan3"  # Correct sheet

df = pd.read_excel(file_path, sheet_name=sheet_name)

# 2. Rename columns for easier manipulation
df = df.rename(columns={
    "Ano": "Year",
    "Empresa": "Firm",
    "ISE_B3": "ISE_B3",
    "DAV/Revenue": "DAV_Revenue",
    "Personal & exchanges/Revenues": "Personal_Exchanges_Revenue",
    "(Taxes, fees and contributions) / (Revenues)": "Taxes_Fees_Revenue",
    "Remuneration of third-party capital / (Revenues)": "Third_Party_Capital_Revenue",
    "Equity re-muneration / (Revenues)": "Equity_Remuneration_Revenue"
})

# 3. Set panel index
df = df.set_index(["Firm", "Year"])

# 4. Create lagged dependent variables
dependent_vars = ["DAV_Revenue", "Personal_Exchanges_Revenue", "Taxes_Fees_Revenue",
                  "Third_Party_Capital_Revenue", "Equity_Remuneration_Revenue"]
for var in dependent_vars:
    df[f"{var}_L1"] = df.groupby("Firm")[var].shift(1)  # First lag
    df[f"{var}_L2"] = df.groupby("Firm")[var].shift(2)  # Second lag (instrument)

# 5. Remove missing values (necessary for regression)
df = df.dropna()

# 6. Estimate IV2SLS for each dependent variable
results = {}
for var in dependent_vars:
    y = df[var]  # Dependent variable
    X = df[["ISE_B3", f"{var}_L1"]]  # Explanatory variables
    Z = df[["ISE_B3", f"{var}_L2"]]  # Instruments

    X = sm.add_constant(X)
    Z = sm.add_constant(Z)

    model = IV2SLS(y, X, Z).fit()
    results[var] = model

# 7. Print results
for var, model in results.items():
    print(f"\nResults for {var}:")
    print(model.summary())
```

S.M.2. IV2SLS Regression Results

Results for DAV_Revenue:

```
==============================================================================
Dep. Variable:          DAV_Revenue   R-squared:                       0.765
Model:                       IV2SLS   Adj. R-squared:                  0.741
Method:                 Two Stage    F-statistic:                     28.41
                  Least Squares      Prob (F-statistic):           1.43e-06
Date:                Sun, 23 Mar 2025
Time:                        19:55:01
No. Observations:                  23
Df Residuals:                      20
Df Model:                           2
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          0.0565      0.315      0.179      0.859      -0.600       0.713
ISE_B3        -0.0006      0.004     -0.132      0.896      -0.010       0.008
DAV_Revenue_L1     0.9628      0.164      5.870      0.000       0.621       1.305
==============================================================================
Omnibus:                       10.702   Durbin-Watson:                   2.289
Prob(Omnibus):                  0.005   Jarque-Bera (JB):                8.687
Skew:                          -1.239   Prob(JB):                       0.0130
Kurtosis:                       4.709   Cond. No.                     1.64e+03
==============================================================================
```

Results for Personal_ Expenses _Revenue:

```
==============================================================================
Dep. Variable:     Personal_ Expenses _Revenue   R-squared:                       0.898
Model:                            IV2SLS   Adj. R-squared:                  0.888
Method:                      Two Stage    F-statistic:                     89.52
                       Least Squares      Prob (F-statistic):           1.05e-10
Date:                     Sun, 23 Mar 2025
Time:                             19:55:01
No. Observations:                       23
Df Residuals:                           20
Df Model:                                2
==============================================================================
```

	coef	std err	t	P>|t|	[0.025	0.975]
const	0.0328	0.063	0.523	0.607	-0.098	0.164
ISE_B3	-0.0004	0.001	-0.470	0.644	-0.002	0.001
Personal_ Expenses _Revenue_L1	0.9986	0.076	13.136	0.000	0.840	1.157

Omnibus:	8.183	Durbin-Watson:	2.122
Prob(Omnibus):	0.017	Jarque-Bera (JB):	5.930
Skew:	-1.000	Prob(JB):	0.0515
Kurtosis:	4.479	Cond. No.	1.75e+03

Results for Taxes_Fees_Revenue:

Dep. Variable:	Taxes_Fees_Revenue	R-squared:	0.855
Model:	IV2SLS	Adj. R-squared:	0.840
Method:	Two Stage Least Squares	F-statistic:	47.26
		Prob (F-statistic):	2.64e-08
Date:	Sun, 23 Mar 2025		
Time:	19:55:01		
No. Observations:	23		
Df Residuals:	20		
Df Model:	2		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-0.0125	0.156	-0.080	0.937	-0.338	0.313
ISE_B3	0.0003	0.002	0.162	0.873	-0.004	0.005
Taxes_Fees_Revenue_L1	0.8295	0.117	7.110	0.000	0.586	1.073

Omnibus:	2.984	Durbin-Watson:	1.843
Prob(Omnibus):	0.225	Jarque-Bera (JB):	1.701
Skew:	-0.651	Prob(JB):	0.427
Kurtosis:	3.283	Cond. No.	1.88e+03

Results for Third_Party_Capital_Revenue:

```
Dep. Variable:    Third_Party_Capital_Revenue   R-squared:                  0.845
Model:                          IV2SLS   Adj. R-squared:                 0.830
Method:                       Two Stage   F-statistic:                    29.13
                        Least Squares   Prob (F-statistic):           1.19e-06
Date:                   Sun, 23 Mar 2025
Time:                           19:55:01
No. Observations:                     23
Df Residuals:                         20
Df Model:                              2
==============================================================================================
                          coef    std err          t      P>|t|      [0.025      0.975]
----------------------------------------------------------------------------------------------
const                     0.1986      0.137      1.449      0.163      -0.087       0.485
ISE_B3                   -0.0027      0.002     -1.597      0.126      -0.006       0.001
Third_Party_Capital_Revenue_L1     1.3578      0.178      7.632      0.000       0.987       1.729
==============================================================================
Omnibus:                    1.009   Durbin-Watson:                   2.142
Prob(Omnibus):              0.604   Jarque-Bera (JB):                0.146
Skew:                       0.050   Prob(JB):                        0.929
Kurtosis:                   3.378   Cond. No.                     1.42e+03
==============================================================================
```

Results for Equity_Remuneration_Revenue:

```
==================================================================================
Dep. Variable:    Equity_Remuneration_Revenue   R-squared:                  0.433
Model:                          IV2SLS   Adj. R-squared:                 0.376
Method:                       Two Stage   F-statistic:                    8.890
                        Least Squares   Prob (F-statistic):            0.00173
Date:                   Sun, 23 Mar 2025
Time:                           19:55:01
No. Observations:                     23
Df Residuals:                         20
Df Model:                              2
==============================================================================================
                          coef    std err          t      P>|t|      [0.025      0.975]
----------------------------------------------------------------------------------------------
const                    -0.2402      0.256     -0.937      0.360      -0.775       0.295
ISE_B3                    0.0027      0.003      0.819      0.422      -0.004       0.009
```

Equity_Remuneration_Revenue_L1	1.0226	0.315	3.245	0.004	0.365	1.680

Omnibus:	15.658	Durbin-Watson:	2.086
Prob(Omnibus):	0.000	Jarque-Bera (JB):	15.762
Skew:	-1.594	Prob(JB):	0.000378
Kurtosis:	5.507	Cond. No.	1.48e+03

S.M.3. First-stage F statistics

```
                            OLS Regression Results
==============================================================================
Dep. Variable:         ISE_B3_Current   R-squared:                       0.719
Model:                            OLS   Adj. R-squared:                  0.715
Method:                 Least Squares   F-statistic:                     174.2
Date:                Sun, 01 Jun 2025   Prob (F-statistic):           1.98e-20
Time:                        15:34:05   Log-Likelihood:                -183.52
No. Observations:                  70   AIC:                             371.0
Df Residuals:                      68   BIC:                             375.5
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const         25.9599      3.988      6.510      0.000      18.003      33.917
ISE_B3_Lagged  0.7122      0.054     13.199      0.000       0.605       0.820
==============================================================================
Omnibus:                        2.197   Durbin-Watson:                   2.227
Prob(Omnibus):                  0.333   Jarque-Bera (JB):                1.519
Skew:                           0.123   Prob(JB):                        0.468
Kurtosis:                       2.321   Cond. No.                         730.
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

F-statistic do primeiro estágio: 174.21002320689394
p-value do F-statistic: 1.977105466113459e-20
```

The test presented above was performed using the Python code below and run in IDLE.

```python
import pandas as pd
import statsmodels.api as sm

# 🔧 Carregar o arquivo Excel
file_path = r'C:.....xlsx'
sheet_name = 'F_statistic'  # Aba onde estão os dados

# 🔷 Carregar os dados da aba
df = pd.read_excel(file_path, sheet_name=sheet_name)

# 🔷 Verificar as colunas disponíveis
print("Colunas disponíveis:", df.columns.tolist())

# 🔷 Ordenar por Empresa e Ano para criar defasagem corretamente
df = df.sort_values(by=['Empresa', 'Ano'])

# 🔷 Criar a variável defasada ISE_B3_Lagged
df['ISE_B3_Lagged'] = df.groupby('Empresa')['ISE_B3_Current'].shift(1)

# 🔍 Remover linhas onde não há defasagem (primeiro ano de cada empresa)
df_reg = df.dropna(subset=['ISE_B3_Lagged'])

# 🔷 Definir variáveis da regressão do primeiro estágio
y = df_reg['ISE_B3_Current']                    # Variável dependente
X = df_reg[['ISE_B3_Lagged']]                    # Variável instrumental
X = sm.add_constant(X)                           # Adiciona intercepto

# 🔶 Rodar regressão OLS (primeiro estágio)
model = sm.OLS(y, X).fit()

# 🔎 Mostrar os resultados
print(model.summary())

# 🔥 Extrair o F-statistic
f_stat = model.fvalue
p_value = model.f_pvalue

print("\nF-statistic do primeiro estágio:", f_stat)
print("p-value do F-statistic:", p_value)
```

S.M.4. Robustness check using first-differenced regression

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  Δ_DAV   R-squared:                       0.014
Model:                            OLS   Adj. R-squared:                 -0.001
Method:                 Least Squares   F-statistic:                    0.9480
Date:                Tue, 10 Jun 2025   Prob (F-statistic):              0.334
Time:                        10:44:43   Log-Likelihood:                 24.261
No. Observations:                  70   AIC:                            -44.52
Df Residuals:                      68   BIC:                            -40.03
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const         -0.0596      0.033     -1.829      0.072      -0.125       0.005
Δ_ISE_B3       0.0051      0.005      0.974      0.334      -0.005       0.016
==============================================================================
Omnibus:                       85.460   Durbin-Watson:                   1.960
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              919.869
Skew:                          -3.740   Prob(JB):                    1.79e-200
Kurtosis:                      19.107   Cond. No.                         9.93
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

The test presented above was performed using the Python code below and run in IDLE.

```python
import pandas as pd
import statsmodels.api as sm

# Carregar os dados
file_path = 'C:\\.......xlsx'
sheet_name = 'F_statistic'
df = pd.read_excel(file_path, sheet_name=sheet_name)

# Ordenar por empresa e ano
df = df.sort_values(by=["Empresa", "Ano"])

# Calcular primeiras diferenças
df['Δ_DAV'] = df.groupby('Empresa')['DAV/Revenue'].diff()
df['Δ_ISE_B3'] = df.groupby('Empresa')['ISE_B3_Current'].diff()

# Remover linhas com NaN (início de cada grupo)
df_diff = df.dropna(subset=['Δ_DAV', 'Δ_ISE_B3'])

# Definir variável dependente e independente
Y = df_diff['Δ_DAV']
X = sm.add_constant(df_diff['Δ_ISE_B3'])

# Estimar modelo OLS
model = sm.OLS(Y, X).fit()

# Exibir resultados
print(model.summary())
```

S.M.5. Python programming scripts for the Hausman test

```
import pandas as pd
import numpy as np
import statsmodels.api as sm
import statsmodels.formula.api as smf
from linearmodels.panel import PanelOLS, RandomEffects
from scipy import stats

# Carregar os dados da aba "Plan3"
file_path = "Dados juntos.xlsx"  # Nome do arquivo
df = pd.read_excel(file_path, sheet_name="Plan3")

# Renomear colunas para remover espaços extras
df.columns = df.columns.str.strip()

# Converter "Ano" para numérico
df["Ano"] = pd.to_numeric(df["Ano"], errors="coerce")

# Remover possíveis valores nulos (caso algum "Ano" não tenha sido convertido corretamente)
df = df.dropna(subset=["Ano"])

# Definir índice de painel
df = df.set_index(["Empresa", "Ano"])  # Agora "Ano" é numérico

# Definir variável dependente e independentes
y_var = "ISE_B3"
x_vars = [
    "DAV/Revenue",
    "Personal & exchanges/Revenues",
    "(Taxes, fees and contributions) / (Revenues)",
    "Remuneration of third-party capital / (Revenues)",
    "Equity re-muneration / (Revenues)",
]

# Adicionar constante para FE e RE
df["const"] = 1

# Modelo de Efeitos Fixos (FE)
fe_model = PanelOLS(df[y_var], df[x_vars + ["const"]], entity_effects=True).fit()

# Modelo de Efeitos Aleatórios (RE)
re_model = RandomEffects(df[y_var], df[x_vars + ["const"]]).fit()

# Cálculo do Teste de Hausman para todas as variáveis
b_fe = fe_model.params[x_vars]  # Coeficientes FE
b_re = re_model.params[x_vars]  # Coeficientes RE
diff = b_fe - b_re

# Matriz de variâncias e covariâncias
V_fe = fe_model.cov.iloc[:-1, :-1]  # Matriz de variância-covariância FE
V_re = re_model.cov.iloc[:-1, :-1]  # Matriz de variância-covariância RE
V_diff = V_fe - V_re

# Estatística Qui-quadrado
hausman_stat = diff.T @ np.linalg.inv(V_diff) @ diff
p_value = stats.chi2.sf(hausman_stat, len(x_vars))
```

```python
# Criar DataFrame com os resultados
resultados_hausman = pd.DataFrame({
    "Variable": x_vars + ["Hausman Test"],
    "Coefficient FE": list(b_fe) + [np.nan],
    "Coefficient RE": list(b_re) + [np.nan],
    "Difference (FE - RE)": list(diff) + [np.nan],
    "Chi-square": [np.nan] * len(x_vars) + [hausman_stat],
    "p-value": [np.nan] * len(x_vars) + [p_value],
})

# Exibir tabela formatada
print("\nHausman Test Results:\n")
print(resultados_hausman.to_string(index=False))

# Decisão do modelo
print("\nHausman Test Decision:")
if p_value < 0.05:
    print("Reject H0: The Fixed Effects (FE) model should be used.")
else:
    print("Fail to reject H0: The Random Effects (RE) model can be used.")
```

S.M.6. Python programming scripts for the random effects (RE) tests.

```
import pandas as pd
import statsmodels.api as sm
from linearmodels.panel import RandomEffects

# 1. Carregar os dados
file_path = r"C:\Users\User\Downloads\Dados juntos.xlsx"  # Substitua pelo caminho correto do arquivo
sheet_name = "Plan3"  # Nome correto da aba

df = pd.read_excel(file_path, sheet_name=sheet_name)

# 2. Renomear colunas para facilitar a manipulação
df = df.rename(columns={
    "Ano": "Year",
    "Empresa": "Firm",
    "ISE_B3": "ISE_B3",
    "DAV/Revenue": "DAV_Revenue",
    "Personal & exchanges/Revenues": "Personal_Exchanges_Revenue",
    "(Taxes, fees and contributions) / (Revenues)": "Taxes_Fees_Revenue",
    "Remuneration of third-party capital / (Revenues)": "Third_Party_Capital_Revenue",
    "Equity re-muneration / (Revenues)": "Equity_Remuneration_Revenue"
})

# 3. Definir índice do painel
df = df.set_index(["Firm", "Year"])

# 4. Definir variáveis dependentes e independentes
X = df[["ISE_B3"]]
X = sm.add_constant(X)  # Adicionar constante

# 5. Estimar modelos de Efeitos Aleatórios (RE) para cada variável dependente
dependent_vars = ["DAV_Revenue", "Personal_Exchanges_Revenue",
                  "Taxes_Fees_Revenue", "Third_Party_Capital_Revenue",
                  "Equity_Remuneration_Revenue"]

results = {}

for var in dependent_vars:
    model = RandomEffects(df[var], X)
    results[var] = model.fit()

# 6. Exibir os resultados de forma resumida
for var, result in results.items():
    print(f"Resultados para {var}:")
    print(result.summary)
    print("\n" + "="*80 + "\n")  # Separação entre os resultados
```

S.M.7. Random Effects Estimation Summary

Results for DAV_Revenue:

==========

Dep. Variable: DAV_Revenue **R-squared:** 0.0986
Estimator: RandomEffects **R-squared (Between):** -0.0014
No. Observations: 149 **R-squared (Within):** 0.0037
Date: Sun, Mar 23 2025 **R-squared (Overall):** -0.0024
Time: 20:24:30 **Log-likelihood** 98.487
Cov. Estimator: Unadjusted
F-statistic: 16.081
Entities: 79 **P-value** 0.0001
Avg Obs: 1.8861 **Distribution:** F(1,147)
Min Obs: 1.0000
Max Obs: 3.0000 **F-statistic (robust):** 0.1373
P-value 0.7115
Time periods: 3 **Distribution:** F(1,147)
Avg Obs: 49.667
Min Obs: 35.000
Max Obs: 63.000

Parameter Estimates

	Parameter	Std. Err.	T-stat	P-value	Lower CI	Upper CI
const	0.4839	0.1507	3.2118	0.0016	0.1861	0.7816
ISE_B3	-0.0007	0.0020	-0.3706	0.7115	-0.0047	0.0032

==========

==========

Results for Personal_Expenses_Revenue:

==========

Dep. Variable: Personal_ Expenses _Revenue **R-squared:** 0.0956
Estimator: RandomEffects **R-squared (Between):** 0.0317
No. Observations: 149 **R-squared (Within):** 0.0051
Date: Sun, Mar 23 2025 **R-squared (Overall):** 0.0306
Time: 20:24:30 **Log-likelihood** 337.04

Cov. Estimator:	Unadjusted		
	F-statistic:	15.540	
Entities:	79	P-value	0.0001
Avg Obs:	1.8861	Distribution:	F(1,147)
Min Obs:	1.0000		
Max Obs:	3.0000	F-statistic (robust):	2.9161
	P-value	0.0898	
Time periods:	3	Distribution:	F(1,147)
Avg Obs:	49.667		
Min Obs:	35.000		
Max Obs:	63.000		

Parameter Estimates

	Parameter	Std. Err.	T-stat	P-value	Lower CI	Upper CI
const	0.1682	0.0385	4.3706	0.0000	0.0921	0.2442
ISE_B3	-0.0009	0.0005	-1.7077	0.0898	-0.0019	0.0001

Results for Taxes_Fees_Revenue:

Dep. Variable:	Taxes_Fees_Revenue	R-squared:	0.0258
Estimator:	RandomEffects	R-squared (Between):	0.0100
No. Observations:	149	R-squared (Within):	-0.0104
Date:	Sun, Mar 23 2025	R-squared (Overall):	0.0107
Time:	20:24:30	Log-likelihood	278.62
Cov. Estimator:	Unadjusted		
	F-statistic:	3.8894	
Entities:	79	P-value	0.0505
Avg Obs:	1.8861	Distribution:	F(1,147)
Min Obs:	1.0000		
Max Obs:	3.0000	F-statistic (robust):	0.1237
	P-value	0.7256	
Time periods:	3	Distribution:	F(1,147)
Avg Obs:	49.667		

Min Obs: 35.000
Max Obs: 63.000

Parameter Estimates

	Parameter	Std. Err.	T-stat	P-value	Lower CI	Upper CI
const	0.0920	0.0537	1.7139	0.0886	-0.0141	0.1981
ISE_B3	0.0003	0.0007	0.3517	0.7256	-0.0012	0.0017

Results for Third_Party_Capital_Revenue:

Dep. Variable:	Third_Party_Capital_Revenue	R-squared:	0.0452
Estimator:	RandomEffects	R-squared (Between):	0.0164
No. Observations:	149	R-squared (Within):	-0.0055
Date:	Sun, Mar 23 2025	R-squared (Overall):	0.0190
Time:	20:24:30	Log-likelihood	143.94
Cov. Estimator:	Unadjusted		
		F-statistic:	6.9614
Entities:	79	P-value	0.0092
Avg Obs:	1.8861	Distribution:	F(1,147)
Min Obs:	1.0000		
Max Obs:	3.0000	F-statistic (robust):	1.2943
		P-value	0.2571
Time periods:	3	Distribution:	F(1,147)
Avg Obs:	49.667		
Min Obs:	35.000		
Max Obs:	63.000		

Parameter Estimates

	Parameter	Std. Err.	T-stat	P-value	Lower CI	Upper CI
const	0.2605	0.1058	2.4613	0.0150	0.0513	0.4697
ISE_B3	-0.0016	0.0014	-1.1377	0.2571	-0.0044	0.0012

==========

==========

Results for Equity_Remuneration_Revenue:

==========

Dep. Variable:	Equity_Remuneration_Revenue	R-squared:	0.0041
Estimator:	RandomEffects	R-squared (Between):	0.0120
No. Observations:	149	R-squared (Within):	-0.0078
Date:	Sun, Mar 23 2025	R-squared (Overall):	0.0104
Time:	20:24:30	Log-likelihood	146.55
Cov. Estimator:	Unadjusted		
		F-statistic:	0.6050
Entities:	79	P-value	0.4379
Avg Obs:	1.8861	Distribution:	F(1,147)
Min Obs:	1.0000		
Max Obs:	3.0000	F-statistic (robust):	0.6328
		P-value	0.4276
Time periods:	3	Distribution:	F(1,147)
Avg Obs:	49.667		
Min Obs:	35.000		
Max Obs:	63.000		

Parameter Estimates

==========

	Parameter	Std. Err.	T-stat	P-value	Lower CI	Upper CI
const	-4.364e-05	0.0944	-0.0005	0.9996	-0.1867	0.1866
ISE_B3	0.0010	0.0013	0.7955	0.4276	-0.0015	0.0035

==========

==========