

README file for
“URAP Spring 2025 - Economic Analysis for Public Policy”
By Finn Tangvorakasem

This repository includes all scripts and data-processing steps used to generate the figures, tables, and analysis presented in the research report for the Undergraduate Research Apprentice Program (URAP) in Spring 2025.

Data. The project analyzes public procurement outcomes across 32 European countries using TED (Tenders Electronic Daily) data between 2018 and 2023, normalized by procurement type and contract structure.

Notes on Figures. Table 4 and 5 were re-made to enhance a clearer picture, using the data from Figure 1 and 2, hence no designated output and script. Figure 1 and 2, likewise, was an indirect product of the data analyses done in Section 6.

Library. The analyses comprise of the following Python libraries:

pandas (import pandas as pd)	Core data manipulation and tabular operations (e.g., filtering, grouping, merging, exporting CSVs).
numpy (import numpy as np)	Numerical operations such as sorting, array manipulation, and mathematical computations (e.g., cumulative sums).
matplotlib.pyplot (import matplotlib.pyplot as plt)	Plotting library for creating static visualizations including line charts, scatter plots, and bar graphs.
seaborn (import seaborn as sns)	High-level interface built on top of matplotlib; used for styling and bar plots.
RegEx (import re)	Splitting and manipulating data within each cell.

Guide to scripts/code inside /Official Code

Section 4 Winner Segmentation and Equal Contract Splitting

highest	Table1
lowest	Table 2
middle	Table 3

Section 6 Analysis: Vague CPVs Contracts

def calculate_vague_percentage(df_input, label)	Table1
vague_raw	Figure 3, Table 4
vague_nodup	Figure 3, Table 5
vague_compare	Figure 3 (input)
def plot_vague(vague_compare)	Figure 3
def generate_vague_summary_table(vague_raw)	Table 4
def generate_vague_summary_table(vague_nodup)	Table 5

Section 7 Analysis: Thresholds and Granularity

Compute_granularity_score_single(df, digits=...,threshold=..., procurement_type='...')	Figure 4 - 11 (Data Processing, DataFrame)
plot_competitiveness_comparison(df1, df2, label_x='...', label_y='...', procurement_type='...')	Figure 4 - 11 (Visualizations)