Assignment 5 - Part 1: Project Organization (In-Class)

AQM2 - UC3M

Spring 2026

1. Folder structure

a) Create the folder structure from the terminal:

```
mkdir -p assignment5/data assignment5/analysis/output assignment5/plots/output
touch assignment5/Makefile assignment5/README.md
touch assignment5/analysis/models.R assignment5/plots/figures.R
```

b) Download `corruption.dta` and place it in `assignment5/data/`.

c) Example `README.md`:

```
# Assignment 5: Corruption and Wealth Analysis

This mini-project explores the relationship between GDP per capita
and perceived corruption using cross-country data.

- `data/` contains the corruption dataset (`corruption.dta`)
- `analysis/` contains the regression script and outputs a LaTeX table
- `plots/` contains the scatter plot script and outputs a PDF figure

Run `make` from the project root to reproduce all outputs.
```

2. Analysis script

The complete `analysis/models.R`:

```
# setwd("~/path/to/assignment5")
options(stringsAsFactors = FALSE)

# ============================================================
# Load packages
# ============================================================

library(readstata13)
library(modelsummary)

# ============================================================
# Load data
```

```r
# ============================================================

df = read.dta13("data/corruption.dta")

# ============================================================
# Define constants
# ============================================================

dep_var = "ti_cpi"
indep_var = "undp_gdp"

# ============================================================
# Clean data
# ============================================================

df = df[!is.na(df[[dep_var]]) & !is.na(df[[indep_var]]), ]
cat("Observations:", nrow(df), "\n")

# Assertion: check minimum sample size
if(nrow(df) < 10) stop("Too few observations")

# ============================================================
# Estimate models
# ============================================================

m1 = lm(ti_cpi ~ undp_gdp, data = df)
m2 = lm(ti_cpi ~ log(undp_gdp), data = df)

# ============================================================
# Save output
# ============================================================

modelsummary(
  list("Level" = m1, "Log" = m2),
  output = "analysis/output/table_models.tex",
  stars = TRUE,
  gof_map = c("r.squared", "nobs"))

cat("Analysis complete. Table saved.\n")
```

Verification that the code works (loading data from URL and running models inline):

```r
library(readstata13)
library(modelsummary)

df = read.dta13("https://raw.githubusercontent.com/franvillamil/AQM2/refs/heads/master/datasets/other/corr

dep_var = "ti_cpi"
```

```
indep_var = "undp_gdp"

df = df[!is.na(df[[dep_var]]) & !is.na(df[[indep_var]]), ]
cat("Observations:", nrow(df), "\n")
```

```
## Observations: 170
```

```
if(nrow(df) < 10) stop("Too few observations")

m1 = lm(ti_cpi ~ undp_gdp, data = df)
m2 = lm(ti_cpi ~ log(undp_gdp), data = df)

modelsummary(
  list("Level" = m1, "Log" = m2),
  stars = TRUE,
  gof_map = c("r.squared", "nobs"),
  output = "markdown")
```

	Level	Log
(Intercept)	2.502***	-8.114***
	(0.124)	(0.769)
undp_gdp	0.000***	
	(0.000)	
log(undp_gdp)		1.431***
		(0.090)
R2	0.673	0.603
Num.Obs.	170	170

Note: + p < 0.1, * p < 0.05, ** p < 0.01, *** p < 0.001

3. Plots script

The complete `plots/figures.R`:

```
# setwd("~/path/to/assignment5")
options(stringsAsFactors = FALSE)

# ============================================================
# Load packages
# ============================================================

library(readstata13)
library(ggplot2)

# ============================================================
# Load and clean data
# ============================================================
```

```r
df = read.dta13("data/corruption.dta")
df = df[!is.na(df$ti_cpi) & !is.na(df$undp_gdp), ]

# ============================================================
# Scatter plot: corruption vs log GDP
# ============================================================

p = ggplot(df, aes(x = log(undp_gdp), y = ti_cpi)) +
  geom_point() +
  geom_smooth(method = "lm") +
  labs(
    x = "Log GDP per capita (PPP)",
    y = "Corruption Perceptions Index",
    title = "Corruption and Wealth") +
  theme_minimal()

ggsave("plots/output/scatter_corruption.pdf", p, width = 7, height = 5)

cat("Scatter plot saved.\n")
```

4. Makefile

The complete `Makefile` (note: recipe lines must use literal tab characters):

```
all: analysis/output/table_models.tex \
     plots/output/scatter_corruption.pdf

analysis/output/table_models.tex: analysis/models.R \
                                  data/corruption.dta
    Rscript --no-save analysis/models.R

plots/output/scatter_corruption.pdf: plots/figures.R \
                                     data/corruption.dta
    Rscript --no-save plots/figures.R
```

Testing: Running `make` from the `assignment5/` directory should execute both scripts and produce `analysis/output/table_model` and `plots/output/scatter_corruption.pdf`. If a file already exists and its dependencies have not changed, `make` will skip it (this is the benefit of a Makefile over running scripts manually).