

# 学部生のための重力モデル

Y.V. Yotov. Gravity for Undergrads.

Code by Blaize Giangiulio (2025)

2026-03-28

## A. 準備

## 必要なパッケージ

パッケージ	用途
<code>readr</code>	CSV ファイルの読み込み
<code>dplyr</code>	データ操作
<code>modelsummary</code>	推定結果の表作成
<code>fixest</code>	高次元固定効果 OLS
<code>penppml</code>	ポアソン擬似最尤推定 (PPML)

## パッケージのインストールと読み込み

```
# Rのメモリ上にある全オブジェクトを削除し、クリーンな状態で分析を始める  
rm(list = ls())
```

```
# 未インストールの場合は自動的にインストールする
```

```
if (!require("dplyr"))      { install.packages("dplyr") }  
if (!require("readr"))      { install.packages("readr") }  
if (!require("modelsummary")) { install.packages("modelsummary") }  
if (!require("stargazer"))  { install.packages("stargazer") }  
if (!require("penppml"))    { install.packages("penppml") }  
if (!require("fixest"))     { install.packages("fixest") }
```

```
library(readr); library(dplyr); library(modelsummary)  
library(stargazer); library(penppml); library(fixest)
```

## B. データ

## データの概要

- ▶ ファイル：Gravity\_Undergrads\_Data.csv
- ▶ 観測数：約 32 万件（1990～2023 年、二国間貿易データ）

変数	内容
Exporter / Importer	輸出国・輸入国の ISO コード
Trade	二国間貿易額
GDP_Exporter / GDP_Importer	輸出国・輸入国の GDP
Distance	二国間距離
RTA	地域貿易協定ダミー
EU	EU 加盟ダミー

## データの読み込み

```
# CSVファイルを読み込み、データフレームとして保存する  
data <- readr::read_csv("Gravity_Undergrads_Data.csv")
```

## 対数変換

重力方程式は対数線形形式で推定する：

$$\ln X_{ij} = \beta_0 + \beta_1 \ln D_{ij} + \beta_2 \ln Y_i + \beta_3 \ln Y_j + \dots$$

係数の解釈は**弾力性 (elasticity)** となる。

# 貿易額・距離・GDPの対数変数を作成する

```
data <- data %>%  
  dplyr::mutate(  
    ln_trade    = log(Trade),  
    ln_dist     = log(Distance),  
    ln_gdp_exp  = log(GDP_Exporter),  
    ln_gdp_imp  = log(GDP_Importer)  
  )
```

**注意：**  $\log(0) = -\infty$  となるゼロ貿易はOLS推定から除外する。

## C. 重力モデルの推定

## 5つのアプローチ

モデル	データ	固定効果	推定法
Naive	2023年	なし	OLS
Structural	2023年	輸出国・輸入国	OLS
Panel	全年	輸出国 × 年・輸入国 × 年	OLS
HDFE	全年	上記 + 二国間ペア	OLS
Multiplicative	全年	上記と同じ	PPML

## C-1. 単純重力モデル

## 単純重力モデル (Naive Gravity)

$$\ln X_{ij} = \beta_0 + \beta_1 \ln D_{ij} + \beta_2 \text{RTA}_{ij} + \beta_3 \text{EU}_{ij} + \beta_4 \ln Y_i + \beta_5 \ln Y_j + \varepsilon_{ij}$$

- ▶ 2023年のクロスセクションデータのみ使用
- ▶ **問題点**：多角的貿易抵抗指数が考慮されず、推定量にバイアスが生じる

```
# stats::lm() で OLS を実行する (2023年・ゼロ貿易を除外)
naive <- lm(
  ln_trade ~ ln_dist + RTA + EU +
    ln_gdp_exp + ln_gdp_imp,
  data = dplyr::filter(data,
    Year == 2023,
    ln_trade != -Inf)
)
```

## C-2. 構造的重力モデル

## 構造的重力モデル (Structural Gravity)

$$\ln X_{ij} = \beta_1 \ln D_{ij} + \beta_2 \text{RTA}_{ij} + \beta_3 \text{EU}_{ij} + \pi_i + \chi_j + \varepsilon_{ij}$$

- ▶  $\pi_i$  : 輸出国固定効果、 $\chi_j$  : 輸入国固定効果
- ▶ **改善点** : 多角的貿易抵抗指数を固定効果で吸収できる

```
# fixest::feols() で固定効果 OLS を実行する
# | の右側に固定効果 (Exporter・Importer) を指定する
structural <- fixest::feols(
  ln_trade ~ ln_dist + RTA + EU |
            Exporter + Importer,
  data = dplyr::filter(data,
                        Year == 2023,
                        ln_trade != -Inf)
)
```

## C-3. パネル重力モデル

## パネル重力モデル (Panel Gravity)

$$\ln X_{ijt} = \beta_1 \ln D_{ij} + \beta_2 \text{RTA}_{ijt} + \beta_3 \text{EU}_{ijt} + \pi_{it} + \chi_{jt} + \varepsilon_{ijt}$$

- ▶  $\pi_{it}$  : 輸出国 × 年固定効果、 $\chi_{jt}$  : 輸入国 × 年固定効果
- ▶ **改善点** : 全年データを使用し、各国の時間変化をコントロールできる

```
# fixest::feols() でパネル固定効果 OLS を実行する
# Exporter^Year: 輸出国 × 年の交差固定効果 (fixest の記法)
panel <- fixest::feols(
  ln_trade ~ ln_dist + RTA + EU |
              Exporter^Year + Importer^Year,
  data = dplyr::filter(data, ln_trade != -Inf)
)
```

## C-4. 高次元固定効果モデル

## 高次元固定効果モデル (HDFE Gravity)

$$\ln X_{ijt} = \beta_1 \text{RTA}_{ijt} + \beta_2 \text{EU}_{ijt} + \pi_{it} + \chi_{jt} + \mu_{ij} + \varepsilon_{ijt}$$

- ▶  $\mu_{ij}$  : 二国間ペア固定効果
- ▶ **改善点** : 距離など時不変の二国間変数を固定効果が吸収するため、`ln_dist` が不要になる

```
# fixest::feols() で高次元固定効果 OLS を実行する  
# Exporter^Importer: 二国間ペア固定効果  
hdfe <- fixest::feols(  
  ln_trade ~ RTA + EU |  
    Exporter^Year + Importer^Year +  
    Exporter^Importer,  
  data = dplyr::filter(data, ln_trade != -Inf)  
)
```

## C-5. 乗法型重力モデル (PPML)

## OLS の2つの問題点

1. **ゼロ貿易問題**：  $\log(0) = -\infty$  のため、ゼロ貿易の観測値を除外しなければならない
2. **Jensen's inequality 問題**：不均一分散のもとで対数線形 OLS の推定量は一致性を失う (Santos Silva & Tenreyro, 2006)

**解決策**：ポアソン擬似最尤推定 (PPML)

$$X_{ijt} = \exp(\beta_1 \text{RTA}_{ijt} + \beta_2 \text{EU}_{ijt} + \pi_{it} + \chi_{jt} + \mu_{ij}) \cdot \varepsilon_{ijt}$$

- ▶ レベル変数をそのまま使用 → ゼロ貿易を含む全観測値を利用可能
- ▶ 不均一分散に対してロバスト

## PPML の推定 (前処理)

`penppml::hdfppml_int()` の仕様に合わせて変数の型を変換する。

```
# Exporter・Importer・Year: 因子型へ (固定効果用)
data$Exporter <- as.factor(data$Exporter)
data$Importer <- as.factor(data$Importer)
data$Year     <- as.factor(data$Year)
# RTA・EU・Trade: 数値型へ (行列・被説明変数用)
data$RTA      <- as.numeric(data$RTA)
data$EU       <- as.numeric(data$EU)
data$Trade    <- as.numeric(data$Trade)
```

## PPML の推定 (固定効果・変数の設定)

```
# 固定効果リスト (hdfeppml_int の仕様)
fes <- list(
  interaction(data$Exporter, data$Year), # 輸出国 × 年
  interaction(data$Importer, data$Year), # 輸入国 × 年
  interaction(data$Exporter, data$Importer) # 二国間ペア
)
# 説明変数行列 (データフレーム不可)
x <- data.matrix(data[, c("RTA", "EU")])
# 被説明変数 (レベル変数・対数ではない)
y <- data$Trade
```

## PPML の推定 (実行)

```
# penppml::hdfeppl_int() で PPML を実行する  
multiplicative <- penppml::hdfeppl_int(  
  y = y, x = x, fes = fes  
)
```

## PPML の結果

出力構造が OLS/feols と異なるため、係数と標準誤差を手動で取り出す。

```
print(paste0("Coefficient on RTA: ", round(multiplicative$coefficients[1], 4)))
```

```
## [1] "Coefficient on RTA: 0.0756"
```

```
print(paste0("Coefficient on EU: ", round(multiplicative$coefficients[2], 4)))
```

```
## [1] "Coefficient on EU: 0.2726"
```

```
print(paste0("SE on RTA:          ", round(multiplicative$se[1], 4)))
```

```
## [1] "SE on RTA:          0.0116"
```

```
print(paste0("SE on EU:          ", round(multiplicative$se[2], 4)))
```

```
## [1] "SE on EU:          0.0236"
```

## D. 結果の可視化

## 表1のコード

```
# 4つの OLS 系モデルを並べた比較表を作成する
# output = "kableExtra" で beamer と互換性のある表を生成する
modelsummary::modelsummary(
  list("Naive" = naive, "Structural" = structural,
       "Panel" = panel, "HDFE" = hdfc),
  output = "kableExtra",
  stars = TRUE,
  title = "Table 1: Gravity Model Estimates"
)
```

## 表 1：推定結果の比較 (Table 1)

Table 4: Table 1: Gravity Model Estimates

	Naive	Structural	Panel	HDFE
(Intercept)	-37.865*** (0.600)			
ln_dist	-1.179*** (0.032)	-1.473*** (0.034)	-1.583*** (0.006)	
RTA	0.664*** (0.049)	0.236*** (0.051)	0.147*** (0.010)	0.103*** (0.013)
EU	0.900*** (0.112)	-0.261* (0.117)	-0.526*** (0.022)	0.427*** (0.030)
ln_gdp_exp	1.352*** (0.015)			
ln_gdp_imp	1.143*** (0.015)			
Num.Obs.	9564	9564	288085	288085
R2	0.643	0.759	0.772	0.897

+ p &lt; 0.1, \* p &lt; 0.05, \*\* p &lt; 0.01, \*\*\* p &lt; 0.001

## 図3のコード

```
# 散布図を描画する (x軸: PPML 予測値、y軸: 実際の貿易額)
plot(
  multiplicative$mu, data$Trade,
  xlab = "Trade, Predicted from Gravity",
  ylab = "Actual Trade",
  main = "Scatterplot of Trade vs Predicted",
  pch = 19,
  col = adjustcolor("steelblue", alpha.f = 0.3)
)
# stats::lm() で近似直線を赤色で追加する
fit <- lm(data$Trade ~ multiplicative$mu)
abline(fit, col = "red", lwd = 2)
```

## 図3：実際の貿易額 vs 予測値 (Figure 3)

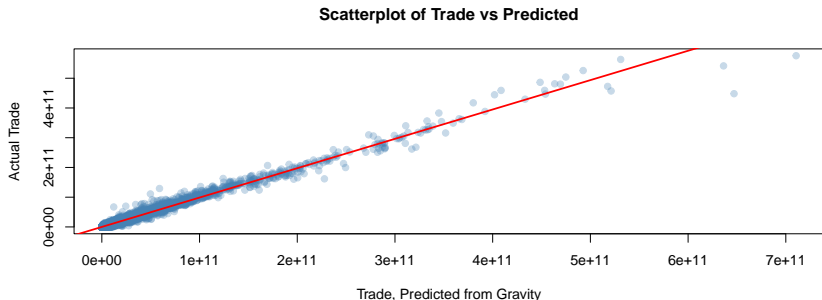


Figure 1: Figure 3: Scatterplot of Actual Trade vs. Gravity Predicted Trade

## まとめ

## まとめ

- ▶ **Naive Gravity (OLS)** : 最もシンプルだが、多角的貿易抵抗指数を無視しバイアスが生じる
- ▶ **Structural Gravity** : 輸出国・輸入国固定効果で多角的貿易抵抗指数を吸収する
- ▶ **Panel Gravity** : 輸出国 × 年・輸入国 × 年固定効果で時間変化をコントロールする
- ▶ **HDFE Gravity** : 二国間ペア固定効果をさらに追加し、時不変の二国間要因を吸収する
- ▶ **Multiplicative Gravity (PPML)** : ゼロ貿易を扱え、不均一分散にロバストである

## 参考文献

- ▶ Yotov, Y.V. “Gravity for Undergrads.”
- ▶ Santos Silva, J.M.C. & Tenreyro, S. (2006). “The Log of Gravity.” *Review of Economics and Statistics*, 88(4), 641–658.
- ▶ Bergstrand, J.H. (1985). “The Gravity Equation in International Trade.” *Review of Economics and Statistics*, 67(3), 474–481.