Economic Trends Analysis- Inelastic Goods, Population, and GDP

Tony, Au

2025-11-25

Introduction

This project aims to predict economic trends using alternative data sources. Specifically, we modeled population growth using the consumption of a daily commodity, rice, and explored the causal relationship between population growth and GDP growth.

This project consisted of 5 parts:

1. Data exploration and cleaning
2. Regression analysis of national rice consumption against population, starting with China
3. Regression of per capita rice consumption against a binary country classification, developed versus developing countries, including models with time-series errors
4. IV (Instrumental Variable) methods to estimate the causal effect of rice consumption on economic growth, measured by GDP, through population dynamics
5. Three-stage recursive VAR (Vector Autoregressive) model with shock decomposition analysis

- Empirical results showed non-significant statistical evidence in support of the relationship between inelastic good consumption, population, and economic development.

1. Data Exploration and Cleaning

Setup

Load necessary libraries.

```r
# install.packages(c("tidyverse", "WDI", "tseries", "AER", "stargazer", "countrycode"))
library(tidyverse)
library(WDI)          # World Bank Data
library(tseries)      # Time series tests (ADF)
library(AER)          # For IV regression
library(stargazer)    # For regression tables
library(countrycode)  # For country names/codes standardization
library(mgcv)
library(MASS)
library(dplyr)
suppressPackageStartupMessages(library(quantmod))  # Data download (FRED) and xts helpers
suppressPackageStartupMessages(library(ggpubr))    # Plot arrangements
suppressPackageStartupMessages(library(lubridate)) # Date utilities
suppressPackageStartupMessages(library(dynlm))     # Time-series OLS with L() lag operator
suppressPackageStartupMessages(library(strucchange)) # Stability tests (CUSUM)
suppressPackageStartupMessages(library(forecast))  # fanchart and forecasting utils
library(vars)       # VAR estimation, IRF, FEVD, diagnostics
library(readxl)     # Read Excel input (.xlsx)
library(readr)
library(ggrepel)
library(tidyr)
library(nlme)   # For GLS with time-series correlation
library(stargazer) # For nice tables
library(ggplot2)
library(purrr)
library(forecast)
```

1.1 Data Sourcing: China

We gathered data for: 1. **Population**: World Bank indicator *SP.POP.TOTL*. 2. **GDP**: World Bank indicator *NY.GDP.MKTP.CD* (Current US) or *NY.GDP.MKTP.KD* (Constant 2015 US). 3. **Rice Consumption**: FAOSTAT.

```
# 1. Load Population and GDP data for China from World Bank
wb_data_china <- read_csv("World_bank_China_population_GDP.csv")
head(wb_data_china)
```

```
## # A tibble: 6 x 7
##    ...1 country iso2c iso3c  year        pop     gdp
##   <dbl> <chr>   <chr> <chr> <dbl>      <dbl>   <dbl>
## 1     1 China   CN    CHN    1990 1135185000 1.04e12
## 2     2 China   CN    CHN    1991 1150780000 1.14e12
## 3     3 China   CN    CHN    1992 1164970000 1.30e12
## 4     4 China   CN    CHN    1993 1178440000 1.48e12
## 5     5 China   CN    CHN    1994 1191835000 1.68e12
## 6     6 China   CN    CHN    1995 1204855000 1.86e12
```

```
# 2. Load Soy Sauce Consumption Data
rice_data_china <- read_csv("FAOSTAT_data_China_rice.csv")
```

1.2 Data Cleaning: China

```
# 3. Select year and Value, rename for clarity
rice_data_china <- rice_data_china %>%
  dplyr::select(year = Year, rice_quantity = Value)
wb_data_china <- wb_data_china %>%
  dplyr::select(year, pop, gdp)

# 4. Join the datasets by year
china_combined <- left_join(wb_data_china, rice_data_china, by = "year")
china_combined <- na.omit(china_combined) # omit N/A value
```

```
# 5. Look at result
head(china_combined)
```

```
## # A tibble: 6 x 4
##    year        pop     gdp rice_quantity
##   <dbl>      <dbl>   <dbl>         <dbl>
## 1  1990 1135185000 1.04e12     191614680
## 2  1991 1150780000 1.14e12     186124638
## 3  1992 1164970000 1.30e12     188291880
## 4  1993 1178440000 1.48e12     179746933
## 5  1994 1191835000 1.68e12     177994395
## 6  1995 1204855000 1.86e12     187297968
```

Data Description: China

The china_combined data frame contains 3 columns of annual data: population, GDP, and Rice production (assuming supply is approximately equal to local demand, given that Asia consumes the most rice globally) for China from 1990 to 2022. The units are people, USD, and tonnes.

```
# Loading data
countries <- c("US", "JP", "FR", "CN", "IN", "BR")
wb_data_multi <- read_csv("World_bank_multi_country_population_GDP.csv")
head(wb_data_multi)
```

```
## # A tibble: 6 x 7
##    ...1 country iso2c iso3c  year       pop     gdp
##   <dbl> <chr>   <chr> <chr> <dbl>     <dbl>   <dbl>
## 1     1 Brazil  BR    BRA    1990 149143223 9.17e11
## 2     2 Brazil  BR    BRA    1991 151724256 9.27e11
## 3     3 Brazil  BR    BRA    1992 154275079 9.22e11
## 4     4 Brazil  BR    BRA    1993 156794577 9.67e11
## 5     5 Brazil  BR    BRA    1994 159265006 1.02e12
## 6     6 Brazil  BR    BRA    1995 161735073 1.07e12
```

```
rice_data_multi <- read_csv("FAOSTAT_data_multi_rice.csv")
head(rice_data_multi)
```

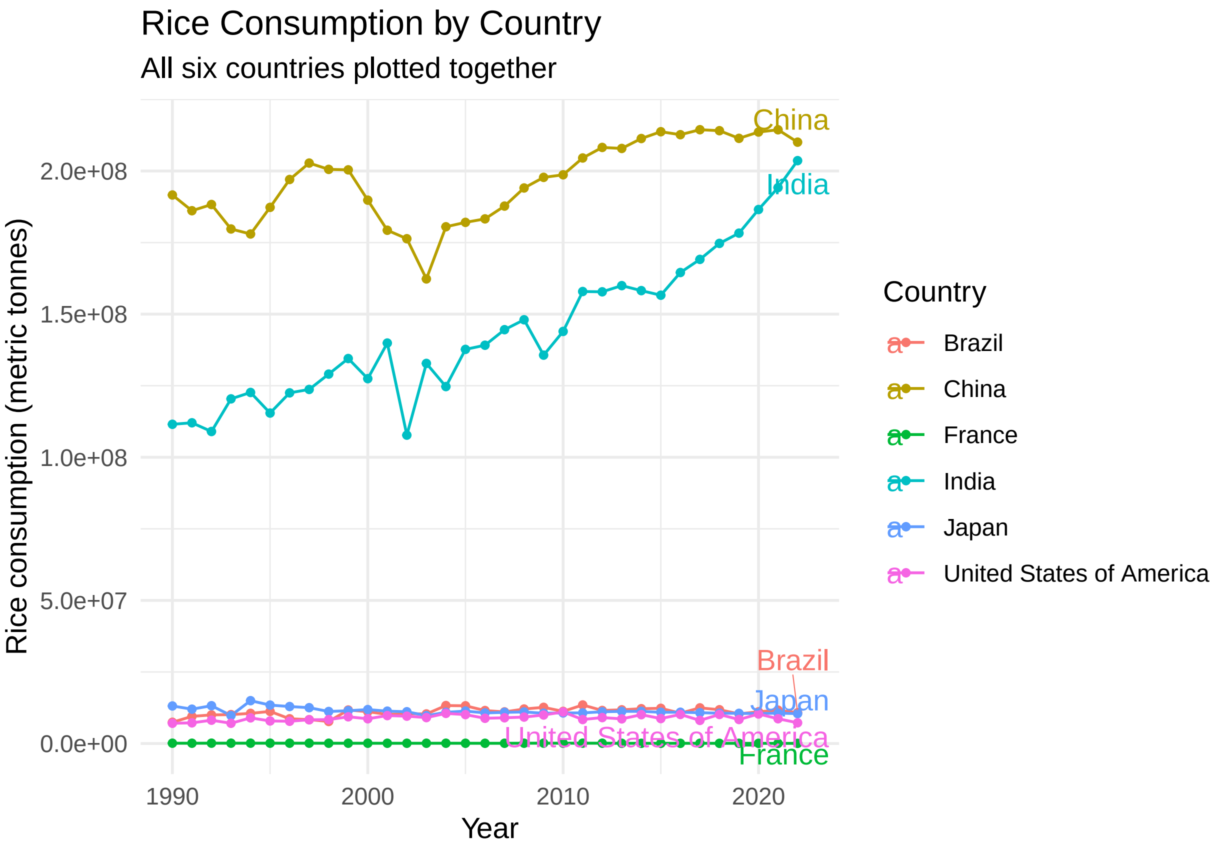
```
## # A tibble: 6 x 15
##   `Domain Code` Domain            `Area Code (M49)` Area  `Element Code` Element
##   <chr>         <chr>             <chr>             <chr>          <dbl> <chr>
## 1 QCL           Crops and livest~ 076               Braz~           5510 Produc~
## 2 QCL           Crops and livest~ 076               Braz~           5510 Produc~
## 3 QCL           Crops and livest~ 076               Braz~           5510 Produc~
## 4 QCL           Crops and livest~ 076               Braz~           5510 Produc~
## 5 QCL           Crops and livest~ 076               Braz~           5510 Produc~
## 6 QCL           Crops and livest~ 076               Braz~           5510 Produc~
## # i 9 more variables: `Item Code (CPC)` <chr>, Item <chr>, `Year Code` <dbl>,
## #   Year <dbl>, Unit <lgl>, Value <dbl>, Flag <chr>, `Flag Description` <chr>,
## #   Note <lgl>
```

1.3 Data Cleaning: Multiple Countries

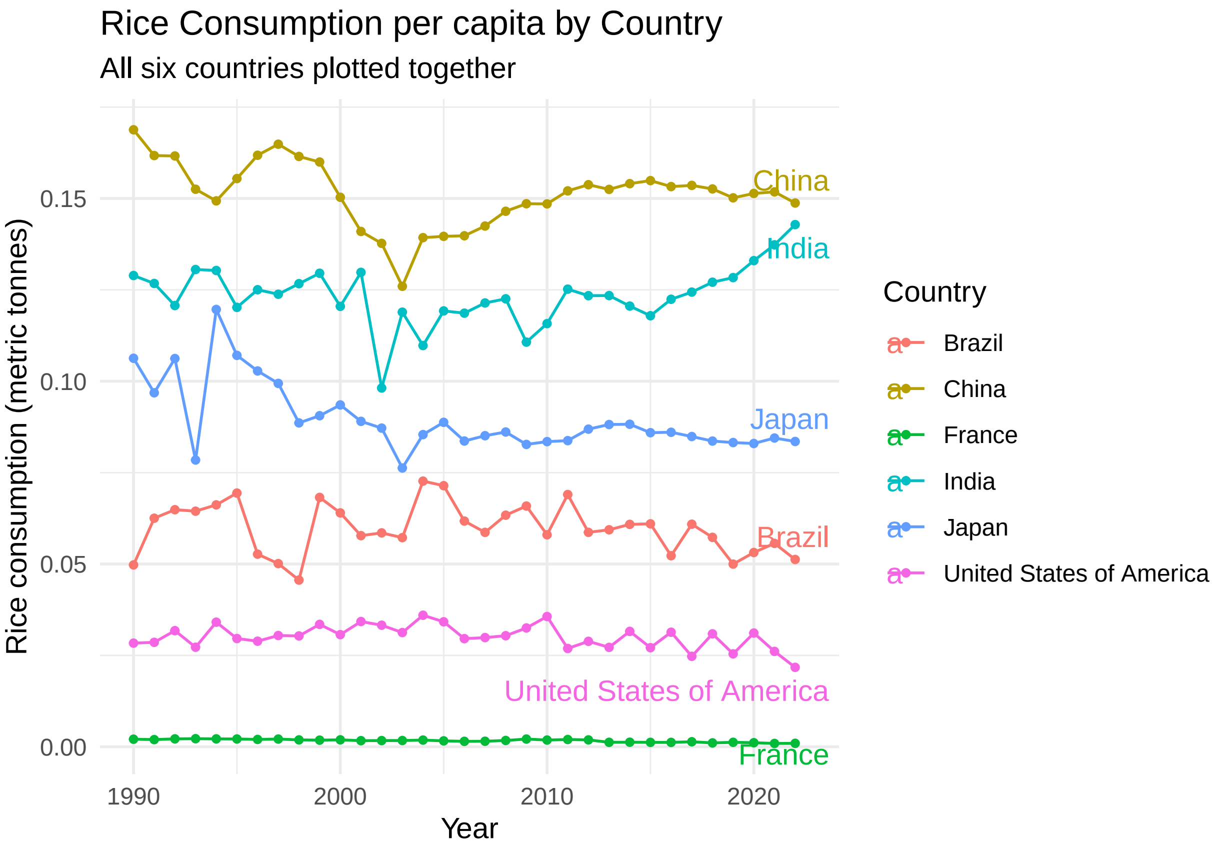
```r
countries <- c("United States of America", "Japan", "France", "China", "India", "Brazil")
developed_countries <- c("United States of America", "Japan", "France")

wb_data_multi <- read_csv("World_bank_multi_country_population_GDP.csv") %>%
  rename(country = country, year = year, population = pop, GDP = gdp) %>%
  mutate(country = case_when(
    country == "United States" ~"United States of America",
    TRUE ~ country
  )) %>%
  filter(country %in% countries)

rice_data_multi <- read_csv("FAOSTAT_data_multi_rice.csv") %>%
  rename(country = Area, year = Year, rice_consumption = Value) %>%
  filter(country %in% countries)

multi_combined <- wb_data_multi %>%
  inner_join(rice_data_multi, by = c("country", "year")) %>%
  mutate(
    developed = if_else(country %in% developed_countries, 1L, 0L)
  ) %>%
  arrange(country, year) %>%
  dplyr::select(year, country, developed, population, GDP, rice_consumption)
```

1.4 Data Description: Multiple Countries

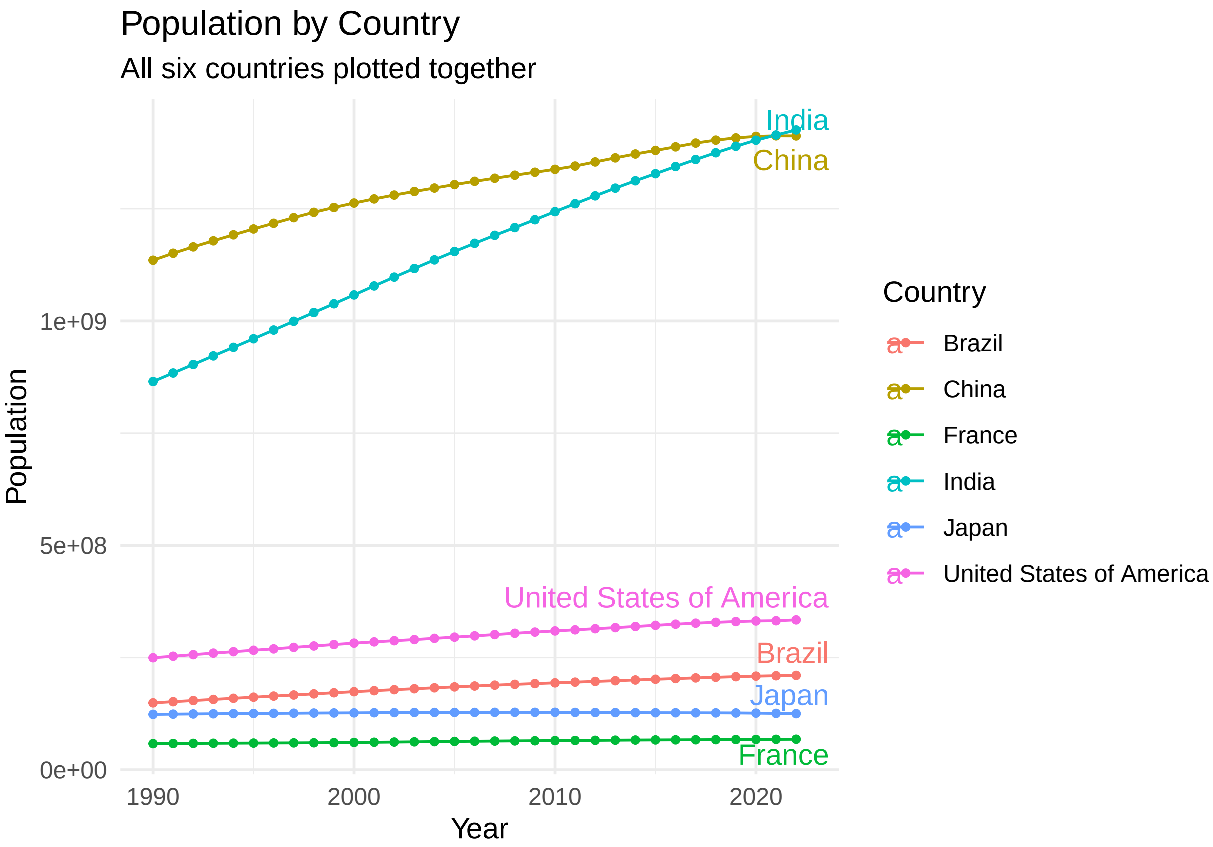
```r
# Plot all six countries on the same axes with a label for each series
last_year_point <- multi_combined %>%
  group_by(country) %>%
  filter(year == max(year))

# Plot for rice consumption
ggplot(multi_combined, aes(x = year, y = rice_consumption, color = country, group = country)) +
  geom_line() +
  geom_point(size = 1) +
  geom_text_repel(
    data = last_year_point,
    aes(label = country),
    nudge_x = 0.5,
    direction = "y",
    hjust = 0,
    segment.size = 0.2
  ) +
  theme_minimal() +
  labs(
    title = "Rice Consumption by Country",
    subtitle = "All six countries plotted together",
    x = "Year",
    y = "Rice consumption (metric tonnes)",
    color = "Country"
  )
```



```
# Plot for rice consumption per c
ggplot(multi_combined, aes(x = year, y = rice_consumption/population, color = country, group = countr
  geom_line() +
  geom_point(size = 1) +
  geom_text_repel(
    data = last_year_point,
    aes(label = country),
    nudge_x = 0.5,
    direction = "y",
    hjust = 0,
    segment.size = 0.2
  ) +
  theme_minimal() +
  labs(
    title = "Rice Consumption per capita by Country",
    subtitle = "All six countries plotted together",
    x = "Year",
    y = "Rice consumption (metric tonnes)",
    color = "Country"
  )
```

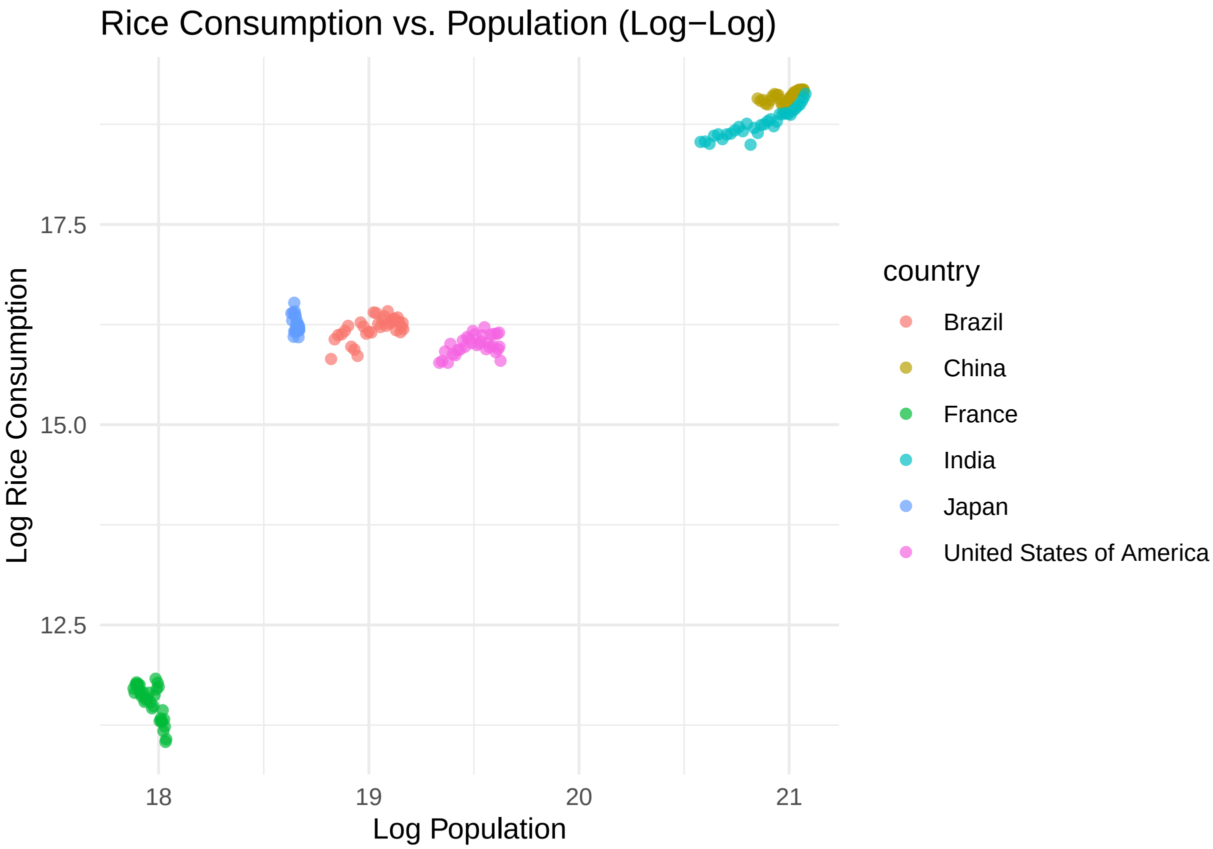


```
# Plot for population
ggplot(multi_combined, aes(x = year, y = population, color = country, group = country)) +
  geom_line() +
  geom_point(size = 1) +
  geom_text_repel(
    data = last_year_point,
    aes(label = country),
    nudge_x = 0.5,
    direction = "y",
    hjust = 0,
    segment.size = 0.2
  ) +
  theme_minimal() +
  labs(
    title = "Population by Country",
    subtitle = "All six countries plotted together",
    x = "Year",
    y = "Population",
    color = "Country"
  )
```



- We observed that China and India consume significantly more rice compared to other countries, likely due to their large population sizes.
- The upward trend suggests an association with population growth. China and India recorded an upward trend in population which synchronizes with the rising consumption of rice, indicating daily inelastic demand.

```
# Base scatter plot
ggplot(multi_combined, aes(x = log(population), y = log(rice_consumption), color = country)) +
  geom_point(size = 1.5, alpha = 0.7) +
  labs(title = "Rice Consumption vs. Population (Log-Log)",
       x = "Log Population", y = "Log Rice Consumption") +
  theme_minimal()
```



The scatterplot shows clustering by country. However, the data for China and India cluster together, suggesting a possibility to model the association using a binary classification of the country's development status.

2. Regression Analysis: National Rice Consumption vs. Population

2.1 OLS and GAM Regression (China)

We analyzed the relationship between Rice Production and Population.

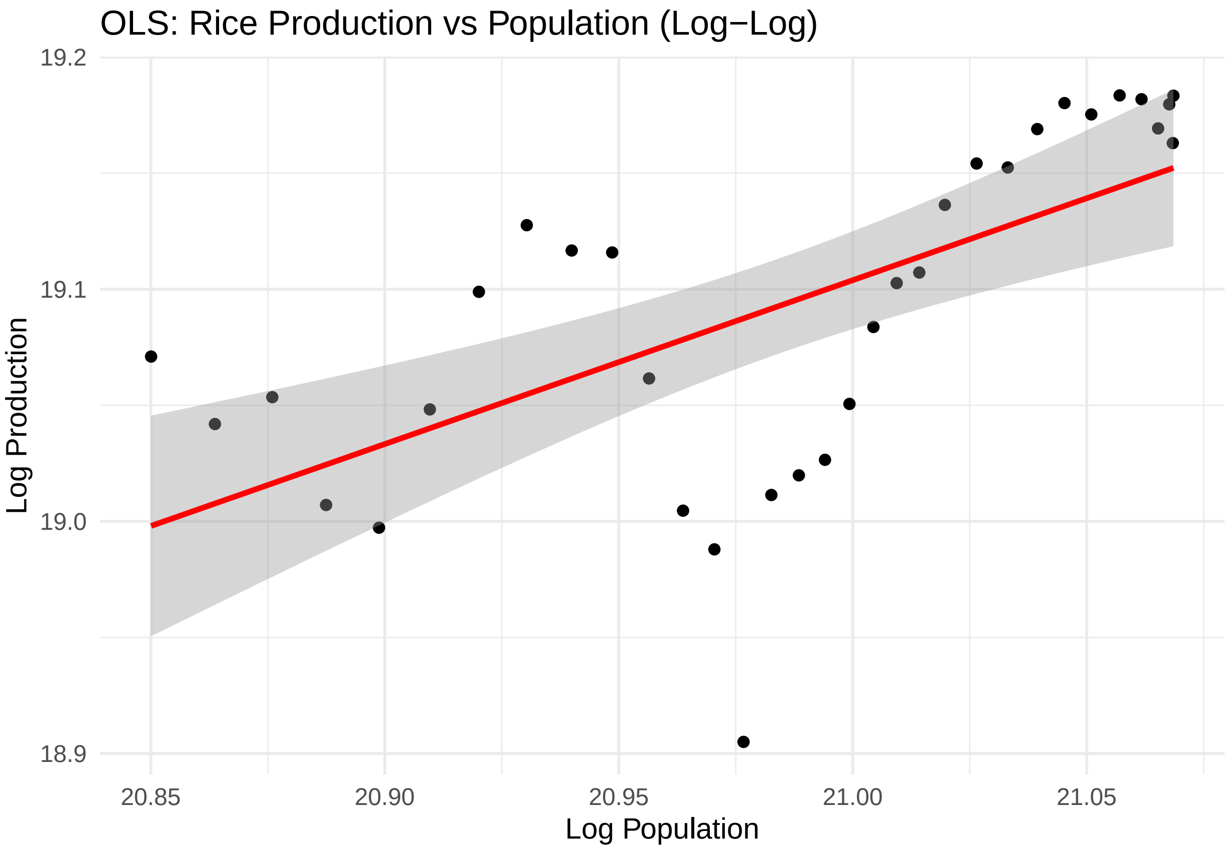
Hypothesis: Population size is linearly associated with the consumption of inelastic goods.

OLS Modeling

```
model_step2 <- lm(log(china_combined$rice_quantity) ~ log(china_combined$pop))
model_nonlin_step2 <- gam(log(china_combined$rice_quantity) ~ s(log(china_combined$pop)))

print(stargazer(model_step2, type = "text", title = "OLS Interaction Model Results (China)"))
```

```
##
## OLS Interaction Model Results (China)
## ==================================================
##                           Dependent variable:
##                       ----------------------------
##                           rice_quantity)
## -----------------------------------------------
## pop)                         0.706***
##                              (0.156)
##
## Constant                      4.277
##                              (3.283)
##
## -----------------------------------------------
## Observations                    33
## R2                            0.397
## Adjusted R2                   0.377
## Residual Std. Error       0.058 (df = 31)
## F Statistic           20.370*** (df = 1; 31)
## ===============================================
## Note:               *p<0.1; **p<0.05; ***p<0.01
##  [1] ""
##  [2] "OLS Interaction Model Results (China)"
##  [3] "==============================================="
##  [4] "                        Dependent variable:    "
##  [5] "                    ---------------------------"
##  [6] "                          rice_quantity)       "
##  [7] "-----------------------------------------------"
##  [8] "pop)                         0.706***          "
##  [9] "                             (0.156)           "
## [10] "                                               "
## [11] "Constant                      4.277            "
## [12] "                             (3.283)           "
## [13] "                                               "
## [14] "-----------------------------------------------"
## [15] "Observations                    33             "
## [16] "R2                            0.397            "
## [17] "Adjusted R2                   0.377            "
## [18] "Residual Std. Error       0.058 (df = 31)      "
## [19] "F Statistic           20.370*** (df = 1; 31)   "
## [20] "==============================================="
## [21] "Note:               *p<0.1; **p<0.05; ***p<0.01"
```

```r
# Plot
ggplot(china_combined, aes(x = log(pop), y = log(rice_quantity))) +
  geom_point() +
  geom_smooth(method = "lm", col = "red") +
  labs(title = "OLS: Rice Production vs Population (Log-Log)",
       x = "Log Population", y = "Log Production") +
  theme_minimal()
```

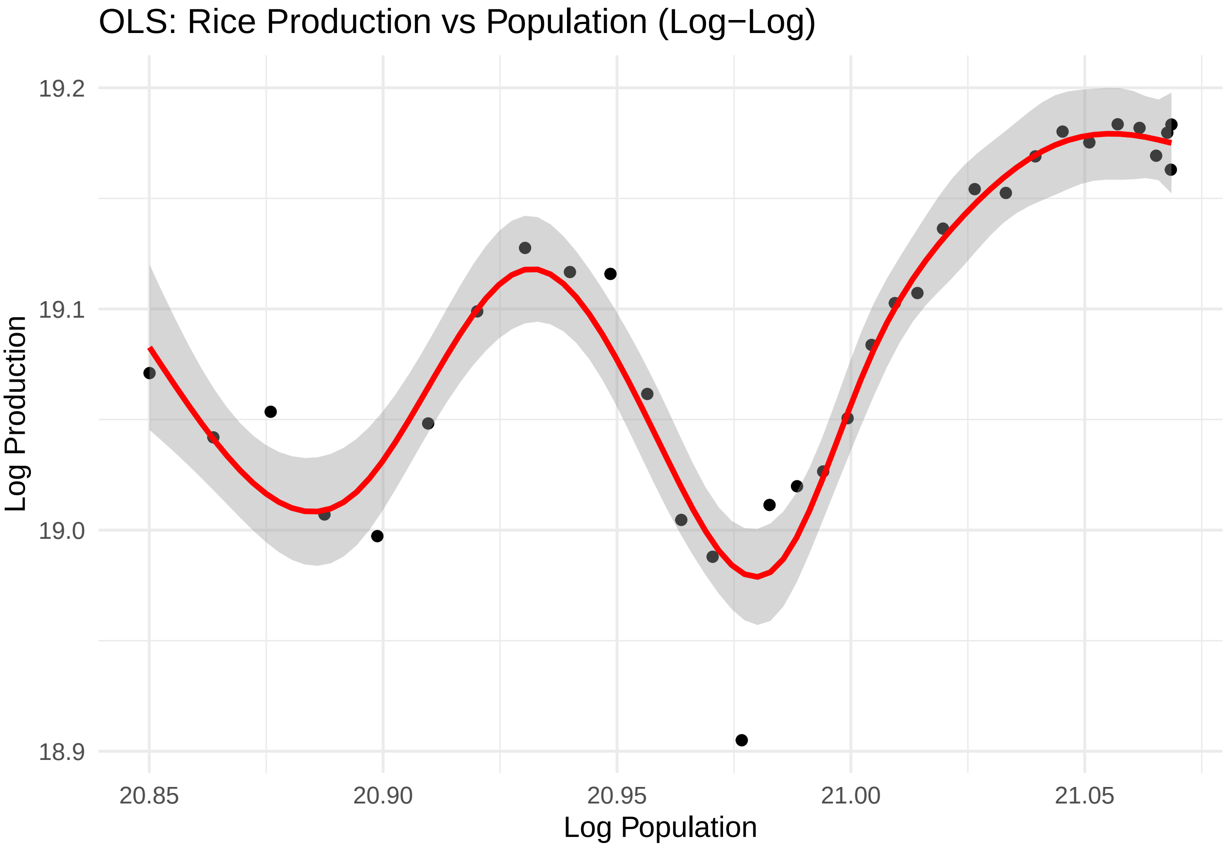


GAM Modeling

```
model_nonlin_step2 <- gam(log(china_combined$rice_quantity) ~ s(log(china_combined$pop)))

print(summary(model_nonlin_step2, title = "GAM Interaction Model Results (China)"))
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## log(china_combined$rice_quantity) ~ s(log(china_combined$pop))
##
## Parametric coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 19.092937   0.003202    5963   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                              edf Ref.df     F p-value
## s(log(china_combined$pop)) 8.446  8.912 53.36  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.937   Deviance explained = 95.3%
## GCV = 0.00047403  Scale est. = 0.00033835  n = 33
```

```r
# Plot
ggplot(china_combined, aes(x = log(pop), y = log(rice_quantity))) +
  geom_point() +
  geom_smooth(method = "gam", col = "red") +
  labs(title = "OLS: Rice Production vs Population (Log-Log)",
       x = "Log Population", y = "Log Production") +
  theme_minimal()
```



The parameters were found to be highly significant. GAM can fit the non-linear data relationship between log rice consumption and log population better visually.

3. Regression of Per Capita Rice Consumption and Binary Classification

We classified countries based on income levels to compare “Developed” vs “Developing” trends.

1. **Classify**: Developed vs Underdeveloped (Developing).
2. **Model**: $Consumption = \beta_0 + \beta_1 Population + \beta_2 Developed + \beta_3(Population \times Developed) + \epsilon$

3.1 Model 1: Simple Regression

Description:
This model investigates how rice consumption varies with population size across all countries and years. The relationship is estimated using three approaches:

- **OLS (Ordinary Least Squares)**: Fits a straight-line relationship (on the log scale).
- **GAM (Generalized Additive Model)**: Allows the relationship to be nonlinear by modeling it as a smooth curve.
- **GLS (Generalized Least Squares) with AR(1) errors**: Accounts for time-series correlation in residuals using a country-specific autoregressive process.

Model Formulae:

- **OLS / GLS:**

  $$\log(\text{RiceConsumption}_{it}) = \beta_0 + \beta_1 \log(\text{Population}_{it}) + \varepsilon_{it}$$

  Where i indexes country, t indexes year, and in the GLS version, ε_{it} may follow an AR(1) process.

- **GAM:**

  $$\log(\text{RiceConsumption}_{it}) = \beta_0 + f(\log(\text{Population}_{it})) + \varepsilon_{it}$$

  Where $f()$ denotes a smooth, potentially nonlinear function estimated from the data.

Analysis (Consumption ~ Population)

Hypothesis: Rice consumption scales with population size.

```
# Ensure data is sorted for time-series models
multi_combined <- multi_combined %>%
  arrange(country, year)

# 1.1 OLS
m1_ols <- lm(log(rice_consumption) ~ log(population), data = multi_combined)

# 1.2 GAM (Generalized Additive Model) - allows non-linear relationship
m1_gam <- gam(log(rice_consumption) ~ s(log(population)), data = multi_combined)

# 1.3 GLS with AR(1) errors (Time-Series Error Term)
# We use correlation = corAR1(form = ~ year | country) to account for serial correlation within each
m1_gls <- gls(log(rice_consumption) ~ log(population),
              data = multi_combined,
              correlation = corAR1(form = ~ year | country),
              method = "REML")

summary(m1_ols)
```

```
##
## Call:
## lm(formula = log(rice_consumption) ~ log(population), data = multi_combined)
##
## Residuals:
```

```
##     Min      1Q  Median      3Q     Max 
## -2.3045 -0.4192 -0.2232  0.8402  1.9404 
## 
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)    
## (Intercept)     -23.06420    1.31617  -17.52   <2e-16 ***
## log(population)   2.01906    0.06739   29.96   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 1.054 on 196 degrees of freedom
## Multiple R-squared:  0.8208, Adjusted R-squared:  0.8199 
## F-statistic: 897.7 on 1 and 196 DF,  p-value: < 2.2e-16
```

```r
summary(m1_gam)
```

```
## 
## Family: gaussian 
## Link function: identity 
## 
## Formula:
## log(rice_consumption) ~ s(log(population))
## 
## Parametric coefficients:
##              Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 16.306289   0.009347    1744   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Approximate significance of smooth terms:
##                      edf Ref.df    F p-value    
## s(log(population)) 8.974      9 7779  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## R-sq.(adj) =  0.997   Deviance explained = 99.7%
## GCV = 0.018218  Scale est. = 0.0173    n = 198
```

```r
summary(m1_gls)
```

```
## Generalized least squares fit by REML
##   Model: log(rice_consumption) ~ log(population) 
##   Data: multi_combined 
##         AIC       BIC   logLik
##   -280.2811 -267.1686 144.1405
## 
## Correlation Structure: AR(1)
##  Formula: ~year | country 
##  Parameter estimate(s):
```

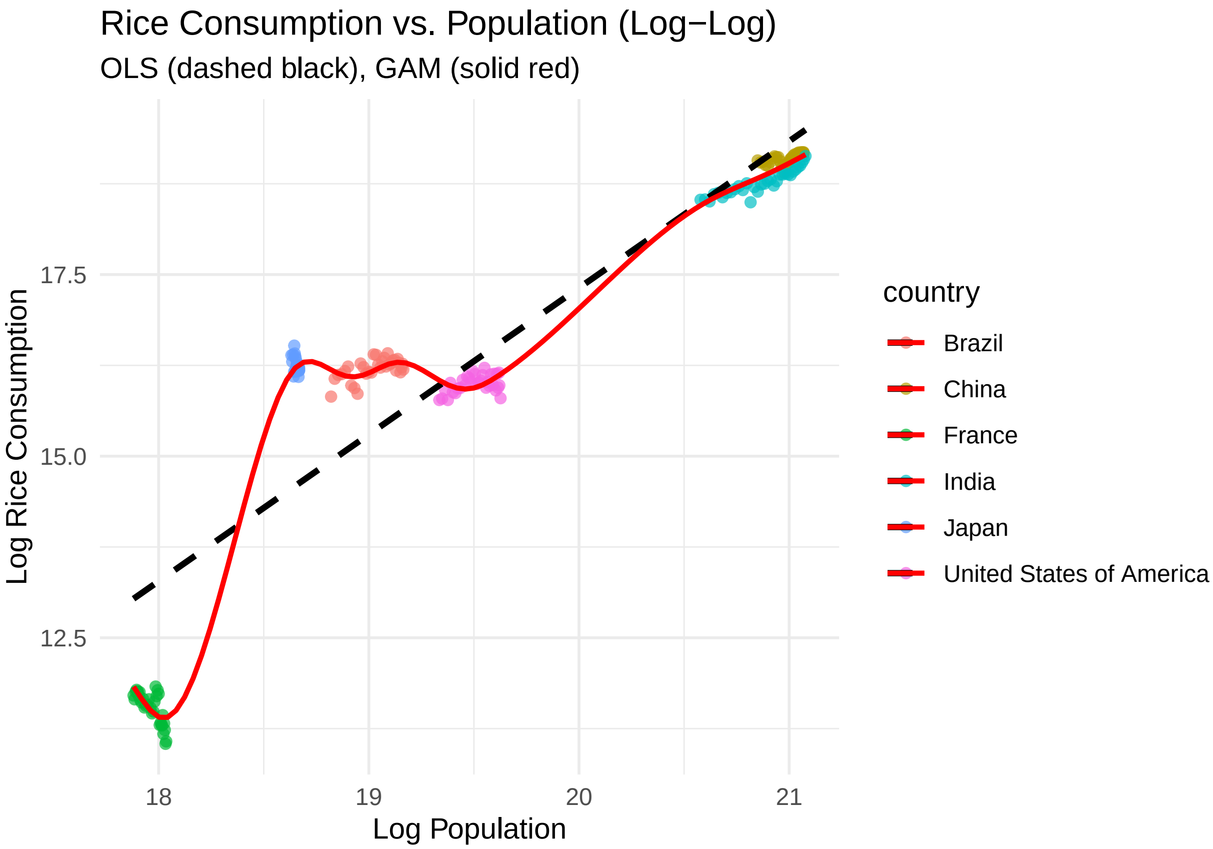
```
##       Phi
## 0.9968294
##
## Coefficients:
##                      Value Std.Error  t-value p-value
## (Intercept)     -17.306010  8.199646 -2.11058  0.0361
## log(population)   1.722201  0.420052  4.09997  0.0001
##
##  Correlation:
##                 (Intr)
## log(population) -0.998
##
## Standardized residuals:
##        Min         Q1        Med         Q3        Max
## -1.9785852 -0.1834846  0.1541567  0.5200010  1.2538735
##
## Residual standard error: 1.369446
## Degrees of freedom: 198 total; 196 residual
```

- The model parameters were all statistically significant.

```r
# Base scatter plot
ggplot(multi_combined, aes(x = log(population), y = log(rice_consumption), color = country)) +
  geom_point(size = 1.5, alpha = 0.7) +
  # OLS line
  geom_smooth(method = "lm", se = FALSE, color = "black", size = 1.1, linetype = "dashed",
              show.legend = TRUE, aes(group = 1)) +
  # GAM line
  geom_smooth(method = "gam", formula = y ~ s(x), se = FALSE, color = "red", size = 0.9, linetype = '
              show.legend = TRUE, aes(group = 5)) +
  labs(title = "Rice Consumption vs. Population (Log-Log)",
       subtitle = "OLS (dashed black), GAM (solid red)",
       x = "Log Population", y = "Log Rice Consumption") +
  theme_minimal()
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



- The plot indicates an upward trend in rice consumption and population.

3.2 Model 2: Interaction Regression

Description:
This model tests if the relationship between rice consumption and population differs between Developed and Developing countries by introducing a binary indicator for "developed" and an interaction term.

- **Full Model**: Consumption ~ Pop + Developed + Pop*Developed
- **Short Model**: Consumption ~ Pop + Developed (no interaction)

Model Formulae (LaTeX):

- **Full Model (with interaction):**

  $$\log(\text{RiceConsumption}_{it}) = \beta_0 + \beta_1 \log(\text{Population}_{it}) + \beta_2 \text{Developed}_i + \beta_3 \left[\log(\text{Population}_{it}) \times \text{Developed}_i\right] + \varepsilon_{it}$$

  Where $\text{Developed}_i = 1$ for developed countries, 0 otherwise.

- **Short Model (without interaction):**

  $$\log(\text{RiceConsumption}_{it}) = \beta_0 + \beta_1 \log(\text{Population}_{it}) + \beta_2 \text{Developed}_i + \varepsilon_{it}$$

- **GAM Full Model:**

  $$\log(\text{RiceConsumption}_{it}) = \beta_0 + f_0\big(\log(\text{Population}_{it})\big) \cdot \mathbb{1}[\text{Developed}_i = 0] + f_1\big(\log(\text{Population}_{it})\big) \cdot \mathbb{1}[\text{Developed}_i = 1] + \varepsilon$$

Analysis

```r
# 2.1 OLS
m2_ols_full <- lm(log(rice_consumption) ~ log(population) * factor(developed), data = multi_combined)
m2_ols_short <- lm(log(rice_consumption) ~ log(population) + factor(developed), data = multi_combined
print(stargazer(m2_ols_full, type = "text", title = "OLS Interaction Model Results"))
```

```
##
## OLS Interaction Model Results
## ==============================================================
##                                        Dependent variable:
##                                    ---------------------------
##                                       log(rice_consumption)
## --------------------------------------------------------------
## log(population)                             1.434***
##                                              (0.103)
##
## factor(developed)1                         -25.752***
##                                              (3.455)
##
## log(population):factor(developed)1          1.315***
##                                              (0.179)
##
## Constant                                   -11.074***
##                                              (2.092)
##
## --------------------------------------------------------------
## Observations                                   198
## R2                                            0.863
## Adjusted R2                                   0.861
## Residual Std. Error                     0.926 (df = 194)
## F Statistic                         407.914*** (df = 3; 194)
## ==============================================================
## Note:                              *p<0.1; **p<0.05; ***p<0.01
##  [1] ""
##  [2] "OLS Interaction Model Results"
##  [3] "=============================================================="
##  [4] "                                       Dependent variable:    "
##  [5] "                                   ---------------------------"
##  [6] "                                      log(rice_consumption)   "
##  [7] "--------------------------------------------------------------"
##  [8] "log(population)                             1.434***          "
##  [9] "                                             (0.103)          "
## [10] "                                                              "
## [11] "factor(developed)1                         -25.752***         "
## [12] "                                             (3.455)          "
## [13] "                                                              "
## [14] "log(population):factor(developed)1          1.315***          "
## [15] "                                             (0.179)          "
## [16] "                                                              "
## [17] "Constant                                   -11.074***         "
## [18] "                                             (2.092)          "
## [19] "                                                              "
## [20] "--------------------------------------------------------------"
## [21] "Observations                                   198            "
## [22] "R2                                            0.863           "
## [23] "Adjusted R2                                   0.861           "
## [24] "Residual Std. Error                     0.926 (df = 194)      "
## [25] "F Statistic                         407.914*** (df = 3; 194)  "
## [26] "=============================================================="
## [27] "Note:                              *p<0.1; **p<0.05; ***p<0.01"
```

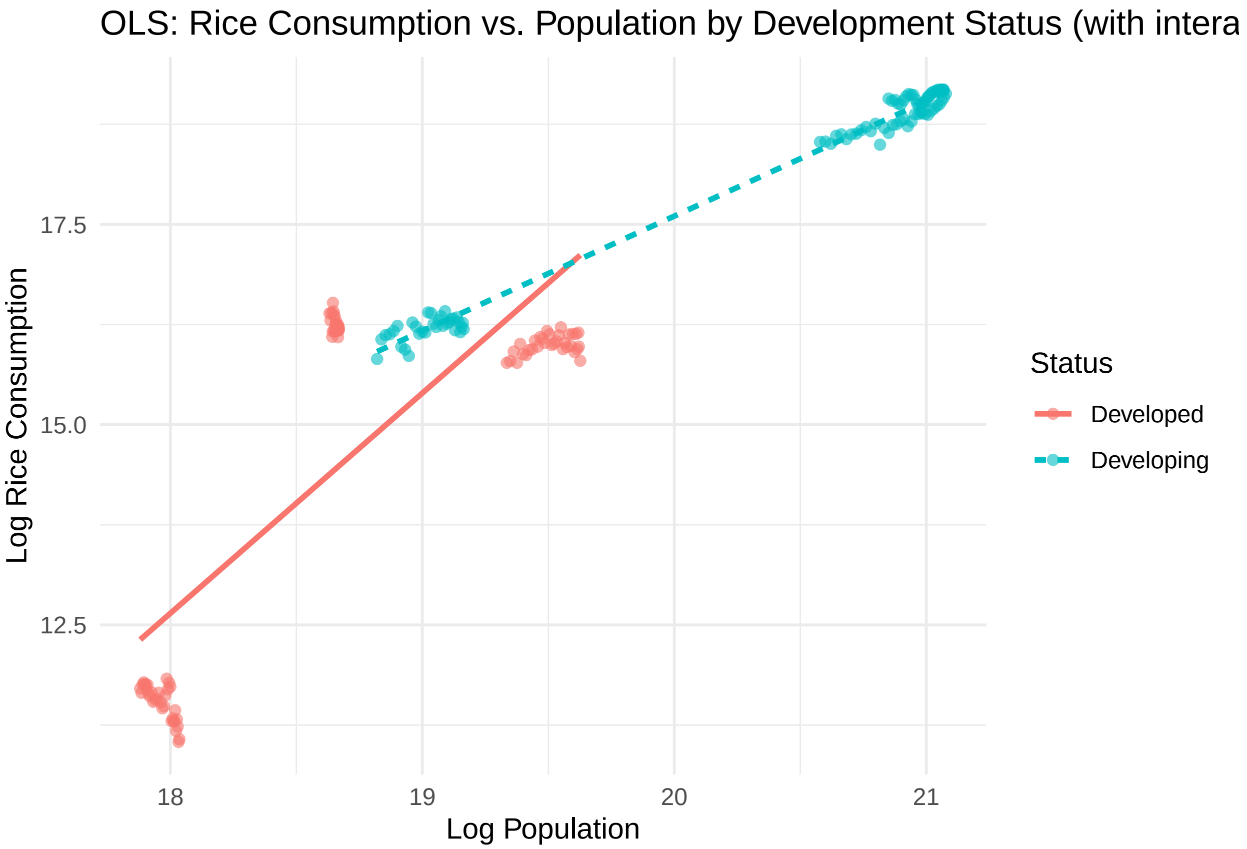
```
print(stargazer(m2_ols_short, type = "text", title = "OLS Interaction Model Results"))
```

```
## 
## OLS Interaction Model Results
## ===============================================
##                         Dependent variable:    
##                     ---------------------------
##                        log(rice_consumption)   
## -----------------------------------------------
## log(population)              1.867***          
##                               (0.095)          
##                                                
## factor(developed)1           -0.474**          
##                               (0.211)          
##                                                
## Constant                    -19.865***         
##                               (1.931)          
##                                                
## -----------------------------------------------
## Observations                    198            
## R2                             0.825           
## Adjusted R2                    0.824           
## Residual Std. Error      1.043 (df = 195)      
## F Statistic          460.596*** (df = 2; 195)  
## ===============================================
## Note:               *p<0.1; **p<0.05; ***p<0.01
##  [1] ""                                               
##  [2] "OLS Interaction Model Results"                  
##  [3] "==============================================="
##  [4] "                        Dependent variable:    "
##  [5] "                    ---------------------------"
##  [6] "                       log(rice_consumption)   "
##  [7] "-----------------------------------------------"
##  [8] "log(population)              1.867***          "
##  [9] "                              (0.095)          "
## [10] "                                               "
## [11] "factor(developed)1           -0.474**          "
## [12] "                              (0.211)          "
## [13] "                                               "
## [14] "Constant                    -19.865***         "
## [15] "                              (1.931)          "
## [16] "                                               "
## [17] "-----------------------------------------------"
## [18] "Observations                    198            "
## [19] "R2                             0.825           "
## [20] "Adjusted R2                    0.824           "
## [21] "Residual Std. Error      1.043 (df = 195)      "
## [22] "F Statistic          460.596*** (df = 2; 195)  "
## [23] "==============================================="
## [24] "Note:               *p<0.1; **p<0.05; ***p<0.01"
```

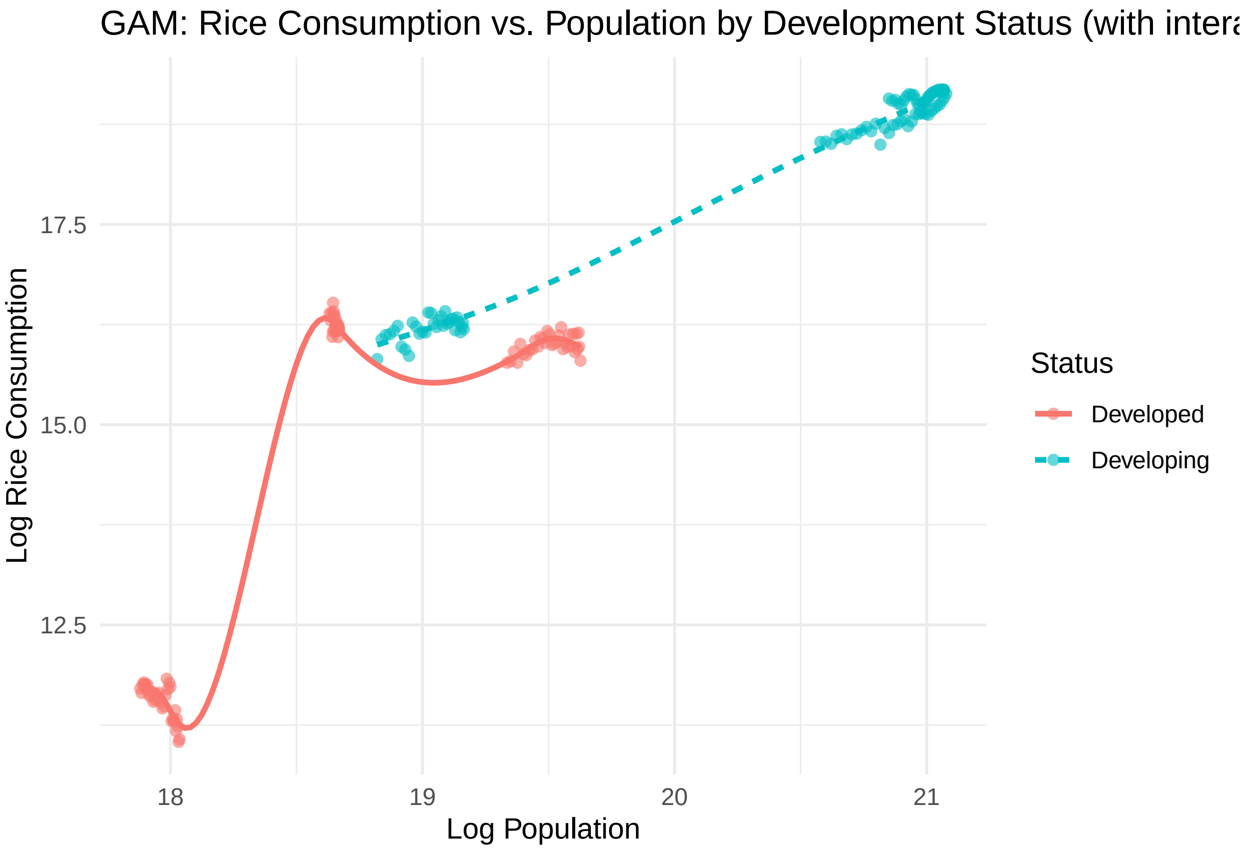
Both the long and short models were highly significant; adding the interaction term yielded little difference.

```
# 2.2 GAM
# We fit separate smooths for each level of 'developed' to capture interaction non-linearly.
m2_gam_full <- gam(log(rice_consumption) ~ s(log(population), by = factor(developed)) + factor(develo
m2_gam_short <- gam(log(rice_consumption) ~ s(log(population)) + factor(developed), data = multi_comb
```

```
multi_combined$dev_label <- ifelse(multi_combined$developed == 1, "Developed", "Developing")

# OLS interaction plot
ggplot(multi_combined, aes(x = log(population), y = log(rice_consumption), color = dev_label)) +
  geom_point(alpha=0.6) +
  # OLS (interaction, by developed)
  geom_smooth(method = "lm", se = FALSE, aes(linetype = dev_label), size = 1) +
  labs(title = "OLS: Rice Consumption vs. Population by Development Status (with interaction)",
       x = "Log Population", y = "Log Rice Consumption", color = "Status", linetype = "Status") +
  theme_minimal()
```

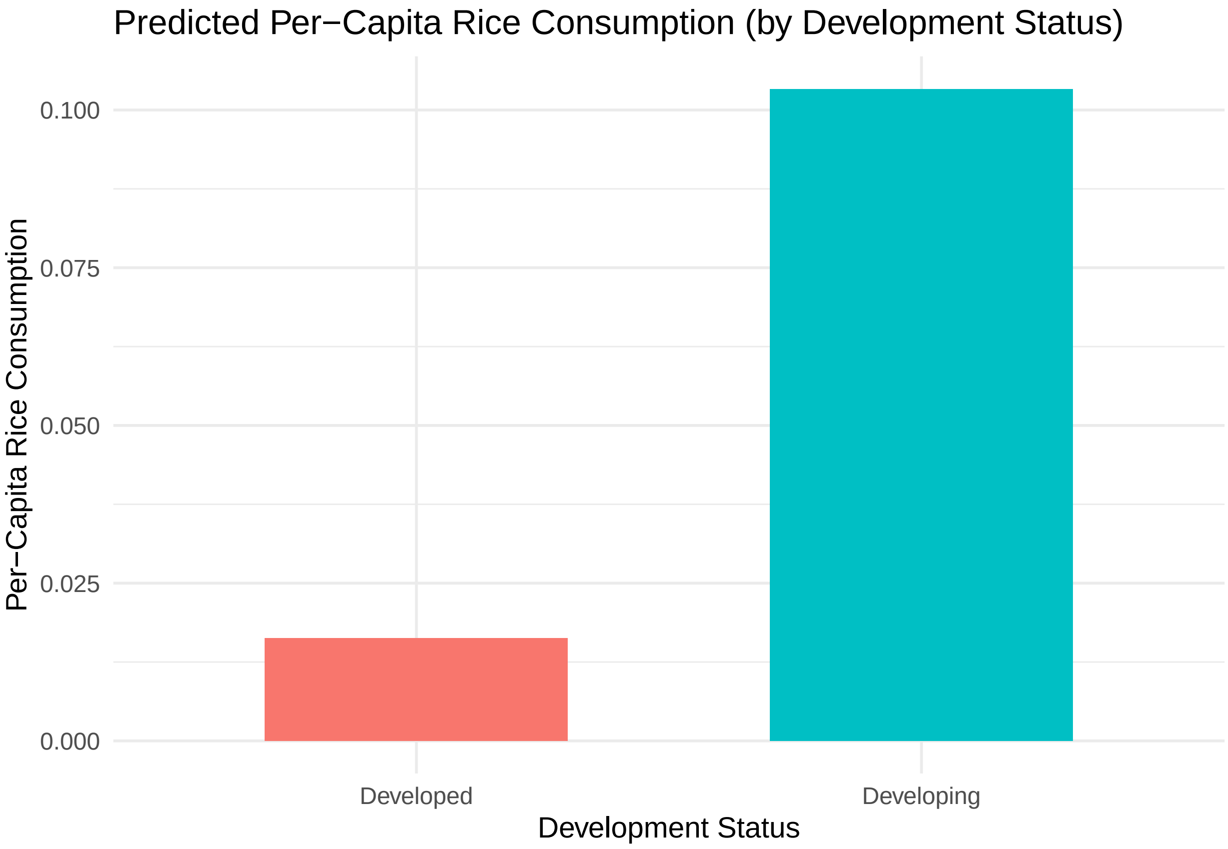


```
# GAM interaction plot (separate smooth for each group)
ggplot(multi_combined, aes(x = log(population), y = log(rice_consumption), color = dev_label)) +
  geom_point(alpha=0.6) +
  geom_smooth(method = "gam", formula = y ~ s(x), se = FALSE, aes(linetype = dev_label), size = 1) +
  labs(title = "GAM: Rice Consumption vs. Population by Development Status (with interaction)",
       x = "Log Population", y = "Log Rice Consumption", color = "Status", linetype = "Status") +
  theme_minimal()
```

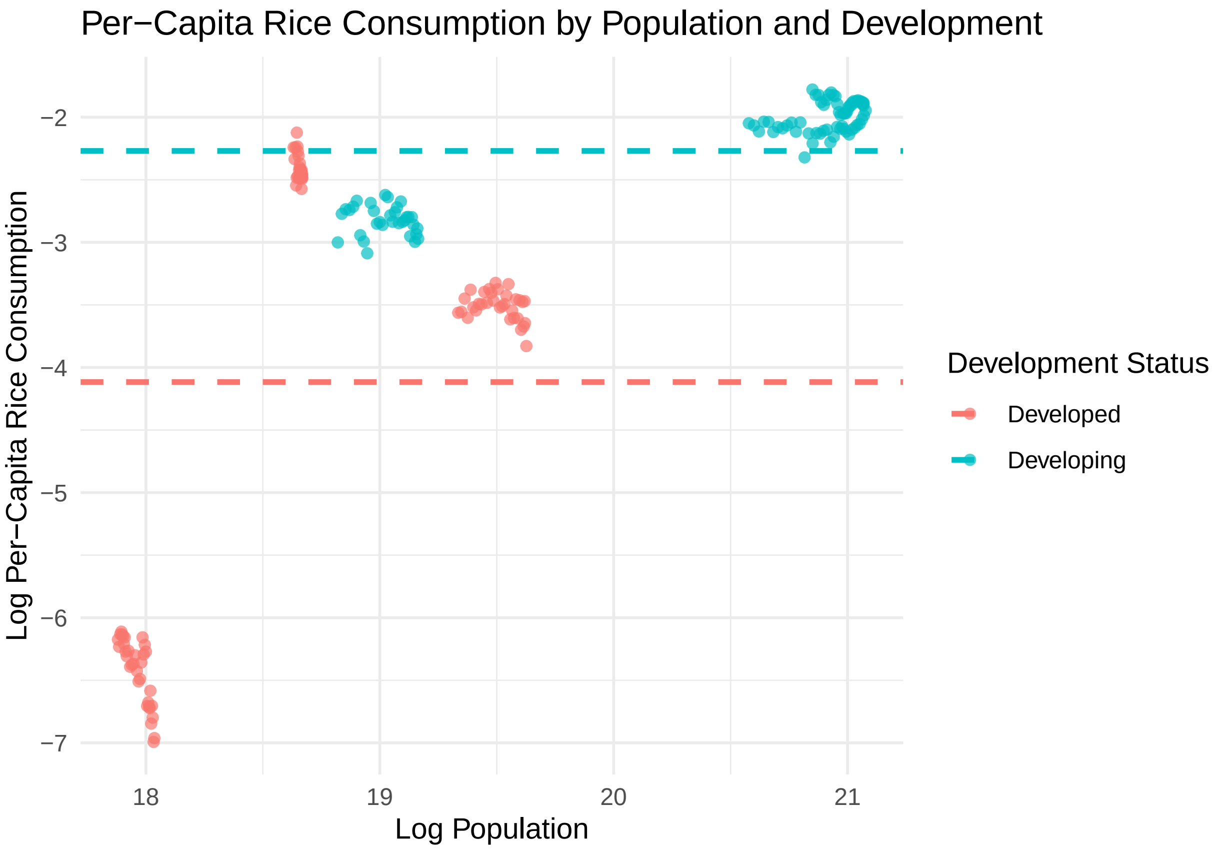


- The plot shows data clustering by country, and labeling them by development status does not significantly clarify the relationship. There exists an intuitive explanation: “The more people in your country, the higher the rice consumption.”
- To investigate this hypothesis, we constructed a new variable “rice consumption per capita” to normalize the differences across countries and achieve a standardized comparison.

```
# 2.3 GLM on consumption per capita

m2_glm_test <- glm(log(rice_consumption/population) ~ factor(developed), data = multi_combined)
multi_combined$dev_label <- ifelse(multi_combined$developed == 1, "Developed", "Developing")

# Compute predicted group means on original scale (per-capita, not log)
pred <- data.frame(
  developed = c(0, 1),
  dev_label = c("Developing", "Developed")
)
pred$log_pc_rice <- predict(m2_glm_test, newdata = pred)
pred$per_capita_rice <- exp(pred$log_pc_rice)

ggplot(pred, aes(x = dev_label, y = per_capita_rice, fill = dev_label)) +
  geom_col(width = 0.6, show.legend = FALSE) +
  labs(
    title = "Predicted Per-Capita Rice Consumption (by Development Status)",
    x = "Development Status",
    y = "Per-Capita Rice Consumption"
  ) +
  theme_minimal()
```



```
# Add predicted value column for each observation
multi_combined$pred_GAM <- predict(m2_glm_test)

ggplot(multi_combined, aes(x = log(population), y = log(rice_consumption/population), color = dev_lab
  geom_point(alpha = 0.7) +
  # Add horizontal line for each group mean (from GAM prediction)
  geom_hline(data = pred, aes(yintercept = log_pc_rice, color = dev_label), linetype = "dashed", size
  labs(
    title = "Per-Capita Rice Consumption by Population and Development",
    x = "Log Population",
    y = "Log Per-Capita Rice Consumption",
    color = "Development Status"
  ) +
  theme_minimal()
```



```
summary(m2_glm_test)
```

```
##
## Call:
## glm(formula = log(rice_consumption/population) ~ factor(developed),
##     data = multi_combined)
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -2.2697     0.1249  -18.17   <2e-16 ***
## factor(developed)1  -1.8469     0.1766  -10.46   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1.544484)
##
##     Null deviance: 471.56  on 197  degrees of freedom
## Residual deviance: 302.72  on 196  degrees of freedom
## AIC: 651.96
##
## Number of Fisher Scoring iterations: 2
```

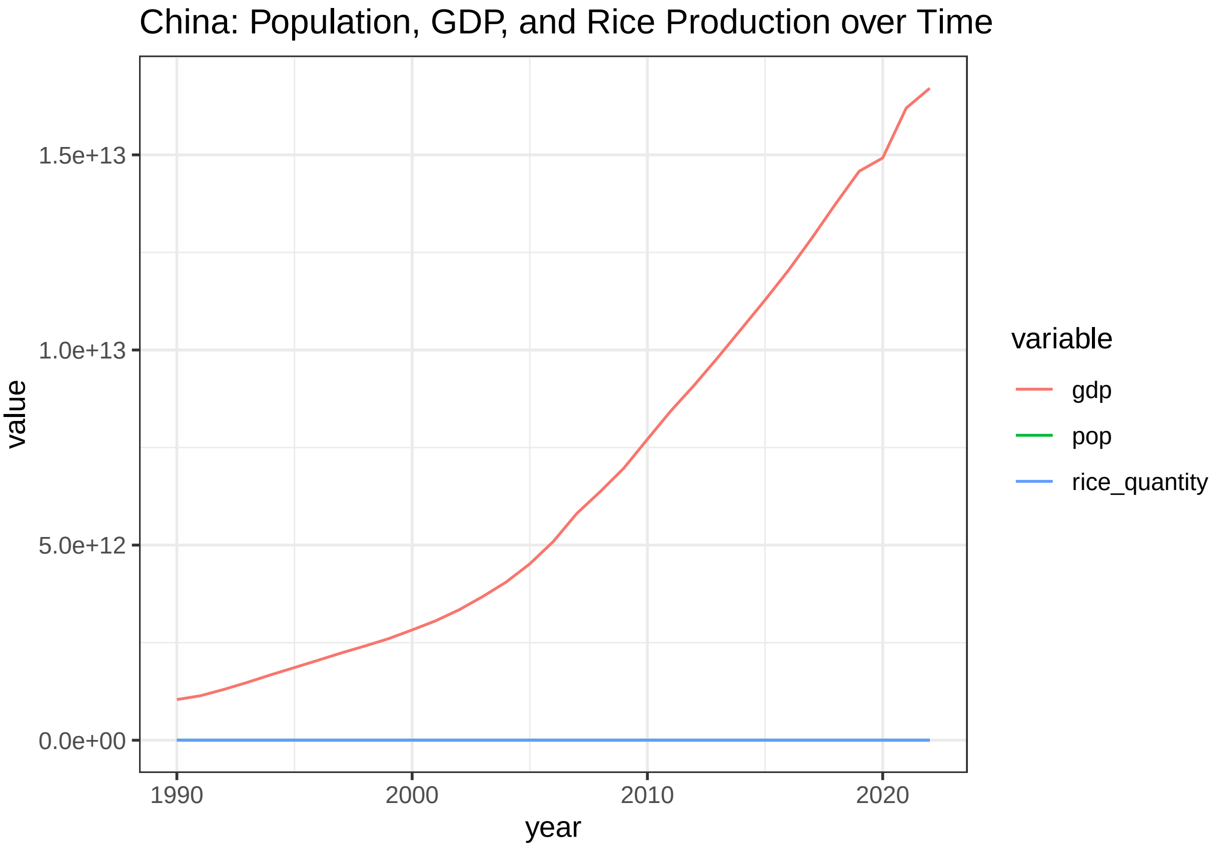
- The bar chart shows that developing countries consume more than four times the amount of developed countries. The plot indicates the regression line for developing countries is higher than for developed countries, but the slope is zero, indicating two groups of average consumption. This hints at a significant effect of the binary label, which was verified by the model summary.

3.3 Regression Model with Time-Series Errors

Suspecting autocorrelation in the data, we verified this through the Augmented Dickey-Fuller (ADF) test and constructed a regression model with time-series errors.

Time-series Data Stationarity Test: ADF Test

```r
# Plot the series
china_long <- china_combined %>%
  pivot_longer(cols = -year, names_to = "variable", values_to = "value")

ggplot(china_long, aes(x = year, y = value, color = variable)) +
  geom_line() +
  theme_bw() +
  labs(title = "China: Population, GDP, and Rice Production over Time")
```



```r
if(length(china_combined$pop) > 10) {
  adf_pop <- adf.test(china_combined$pop)
  adf_rice <- adf.test(china_combined$rice_quantity)

  print(paste("ADF p-value for Population:", adf_pop$p.value))
  print(paste("ADF p-value for Rice Consumption:", adf_rice$p.value))
}
```

```
## [1] "ADF p-value for Population: 0.975369655986486"
## [1] "ADF p-value for Rice Consumption: 0.401397560003098"
```

```r
# If p-value > 0.05, data is non-stationary. We might need differencing or log-transformation.

if(length(china_combined$pop) > 10) {
  adf_pop <- adf.test(diff(log(china_combined$pop)))
  adf_rice <- adf.test(diff(log(china_combined$rice_quantity)))

  print(paste("ADF p-value for (log diff) Population:", adf_pop$p.value))
  print(paste("ADF p-value for (log diff) Rice Consumption:", adf_rice$p.value))
}
```

```
## [1] "ADF p-value for (log diff) Population: 0.504480752960819"
## [1] "ADF p-value for (log diff) Rice Consumption: 0.146616219070101"
```

```r
# consumption per capita
if(length(china_combined$pop) > 10) {
  adf_rice_pc<- adf.test(diff(log(china_combined$rice_quantity/china_combined$pop)))

  print(paste("ADF p-value for (log diff) Rice Consumption per capita:", adf_rice_pc$p.value))
}
```

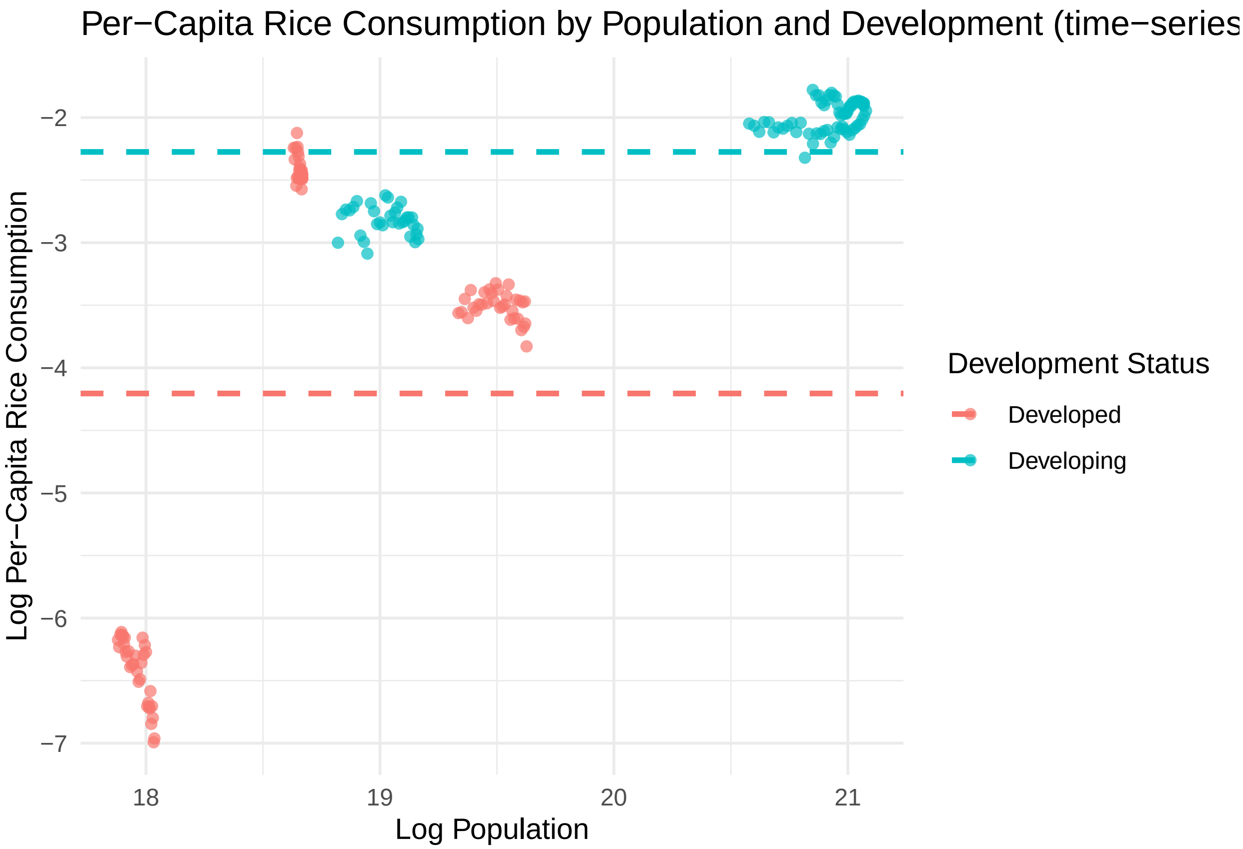
```
## [1] "ADF p-value for (log diff) Rice Consumption per capita: 0.13108681683182"
```

The results were not significant, even after log-differencing, indicating the presence of a unit root in the series (i.e., non-stationary). We now construct a regression model with time-series errors to test that.

GLS with AR(1) Error Term

```r
# 2.3 GLS
m2_gls_test <- gls(log(rice_consumption/population) ~ factor(developed),
                   data = multi_combined,
                   correlation = corAR1(form = ~ year | country),
                   method = "REML")

multi_combined$dev_label <- ifelse(multi_combined$developed == 1, "Developed", "Developing")

# Compute predicted group means on original scale (per-capita, not log)
pred <- data.frame(
  developed = c(0, 1),
  dev_label = c("Developing", "Developed")
)
pred$log_pc_rice <- predict(m2_gls_test, newdata = pred)
pred$per_capita_rice <- exp(pred$log_pc_rice)

# Add predicted value column for each observation
```

```
multi_combined$pred_GAM <- predict(m2_gls_test)

ggplot(multi_combined, aes(x = log(population), y = log(rice_consumption/population), color = dev_lab
  geom_point(alpha = 0.7) +
  # Add horizontal line for each group mean (from GAM prediction)
  geom_hline(data = pred, aes(yintercept = log_pc_rice, color = dev_label), linetype = "dashed", size
  labs(
    title = "Per-Capita Rice Consumption by Population and Development (time-series error)",
    x = "Log Population",
    y = "Log Per-Capita Rice Consumption",
    color = "Development Status"
  ) +
  theme_minimal()
```



```
print(stargazer(m2_gls_test, type = "text", title = "GLS Interaction Model Results"))
```

```
##
## GLS Interaction Model Results
## ===================================================
##                          Dependent variable:
##                     -------------------------------
##                     log(rice_consumption/population)
## ---------------------------------------------------
## factor(developed)1              -1.930
##                                 (1.264)
##
## Constant                        -2.275**
##                                 (0.894)
##
## -----------------------------------------------------
## Observations                      198
## Log Likelihood                  145.279
## Akaike Inf. Crit.               -282.559
## Bayesian Inf. Crit.             -269.446
## =====================================================
## Note:                     *p<0.1; **p<0.05; ***p<0.01
##  [1] ""
##  [2] "GLS Interaction Model Results"
##  [3] "====================================================="
##  [4] "                          Dependent variable:        "
##  [5] "                    ---------------------------------"
##  [6] "                    log(rice_consumption/population) "
##  [7] "-----------------------------------------------------"
##  [8] "factor(developed)1               -1.930              "
##  [9] "                                (1.264)              "
## [10] "                                                     "
## [11] "Constant                        -2.275**             "
## [12] "                                (0.894)              "
## [13] "                                                     "
## [14] "-----------------------------------------------------"
## [15] "Observations                      198                "
## [16] "Log Likelihood                  145.279              "
## [17] "Akaike Inf. Crit.               -282.559             "
## [18] "Bayesian Inf. Crit.             -269.446             "
## [19] "====================================================="
## [20] "Note:                     *p<0.1; **p<0.05; ***p<0.01"
```

GLS Result Interpretation

The model parameters lost significance, suggesting the previous association was spurious and driven by autocorrelation. This demonstrates that despite the regression model looking very similar visually, the violation of assumptions can invalidate the relationship.

We then investigated more deeply into the data to see the time-series properties.

Unit-root Stationarity Test

```r
# ADF for population and rice_consumption by country
adf_results <- multi_combined %>%
  group_by(country) %>%
  summarise(
    adf_p_pop = tryCatch(adf.test(population)$p.value, error = function(e) NA),
    adf_p_rice = tryCatch(adf.test(rice_consumption)$p.value, error = function(e) NA)
  )

print(adf_results)
```

```
## # A tibble: 6 x 3
##   country                  adf_p_pop adf_p_rice
##   <chr>                        <dbl>      <dbl>
## 1 Brazil                       0.99       0.764
## 2 China                        0.975      0.401
## 3 France                       0.413      0.296
## 4 India                        0.99       0.952
## 5 Japan                        0.99       0.441
## 6 United States of America     0.99       0.815
```

```
# Check if (first differencing) population growth rates and rice consumption growth rates are station
adf_results_diff <- multi_combined %>%
  group_by(country) %>%
  summarise(
    adf_p_pop_diff = tryCatch(adf.test(diff(log(population)))$p.value, error = function(e) NA),
    adf_p_rice_diff = tryCatch(adf.test(diff(log(rice_consumption)))$p.value, error = function(e) NA)
  )

print(adf_results_diff)
```

```
## # A tibble: 6 x 3
##   country                  adf_p_pop_diff adf_p_rice_diff
##   <chr>                             <dbl>           <dbl>
## 1 Brazil                           0.173           0.0105
## 2 China                            0.504           0.147
## 3 France                           0.468           0.0165
## 4 India                            0.350           0.01
## 5 Japan                            0.103           0.01
## 6 United States of America         0.0737          0.01
```

Some countries exhibited unit-root non-stationarity.

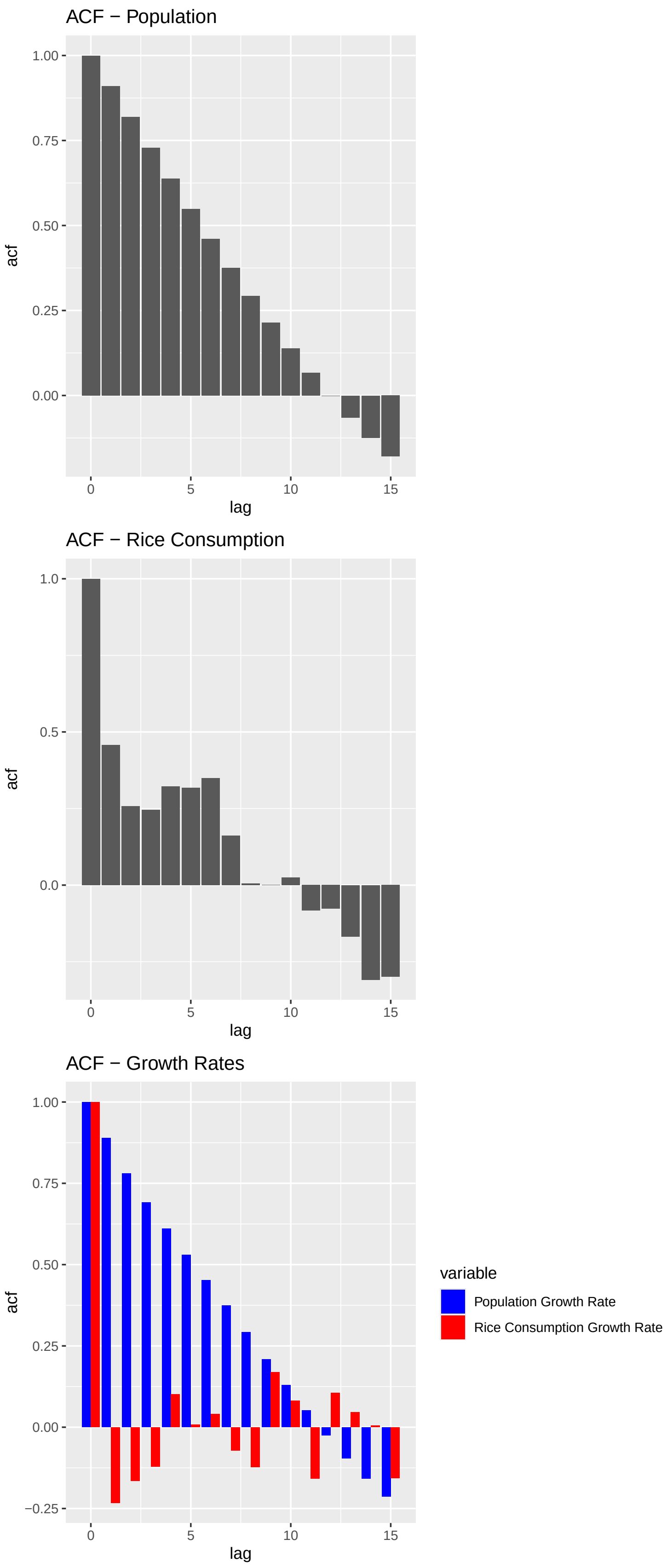
```
# Prepare wide data with all variables
plot_data <- multi_combined %>%
  group_by(country) %>%
  mutate(
    pop_growth = c(NA, diff(log(population))),
    rice_growth = c(NA, diff(log(rice_consumption)))
  ) %>%
  dplyr::select(country, population, rice_consumption, pop_growth, rice_growth) %>%
  pivot_longer(
    cols = c(population, rice_consumption, pop_growth, rice_growth),
    names_to = "variable",
    values_to = "value"
```

```r
  ) %>%
  filter(!is.na(value))

# List of variable sets to plot for each country
variables_to_plot <- list(
  population = "population",
  rice_consumption = "rice_consumption",
  growth_rate = c("pop_growth", "rice_growth")
)

# To plot: By country, show 3 ACFs: pop, rice, growth_rate (overlay pop/rice growth rates)
plot_acf_country <- function(df, country_name) {
  # ACF for population
  acf_pop <- ggAcf(df$value[df$variable == "population"], plot = FALSE)
  data_pop <- with(acf_pop, data.frame(lag, acf, variable = "Population"))

  # ACF for rice consumption
  acf_rice <- ggAcf(df$value[df$variable == "rice_consumption"], plot = FALSE)
  data_rice <- with(acf_rice, data.frame(lag, acf, variable = "Rice Consumption"))

  # ACF for pop growth
  acf_popg <- ggAcf(df$value[df$variable == "pop_growth"], plot = FALSE)
  data_popg <- with(acf_popg, data.frame(lag, acf, variable = "Population Growth Rate"))

  # ACF for rice growth
  acf_riceg <- ggAcf(df$value[df$variable == "rice_growth"], plot = FALSE)
  data_riceg <- with(acf_riceg, data.frame(lag, acf, variable = "Rice Consumption Growth Rate"))

  # Put growth rates together
  data_growth <- rbind(data_popg, data_riceg)
  data_growth$variable <- factor(data_growth$variable)

  # Three panels:
  p1 <- ggplot(data_pop, aes(lag, acf)) + geom_bar(stat="identity") +
    ggtitle("ACF - Population")
  p2 <- ggplot(data_rice, aes(lag, acf)) + geom_bar(stat="identity") +
    ggtitle("ACF - Rice Consumption")
  p3 <- ggplot(data_growth, aes(lag, acf, fill=variable)) +
    geom_bar(stat="identity", position = "dodge") +
    ggtitle("ACF - Growth Rates") +
    scale_fill_manual(values = c("Population Growth Rate"="blue", "Rice Consumption Growth Rate"="red

  library(patchwork) # for nice composition
  combined <- (p1 / p2 / p3) + plot_annotation(title = paste("ACF plots for", country_name))
  print(combined)
}
```

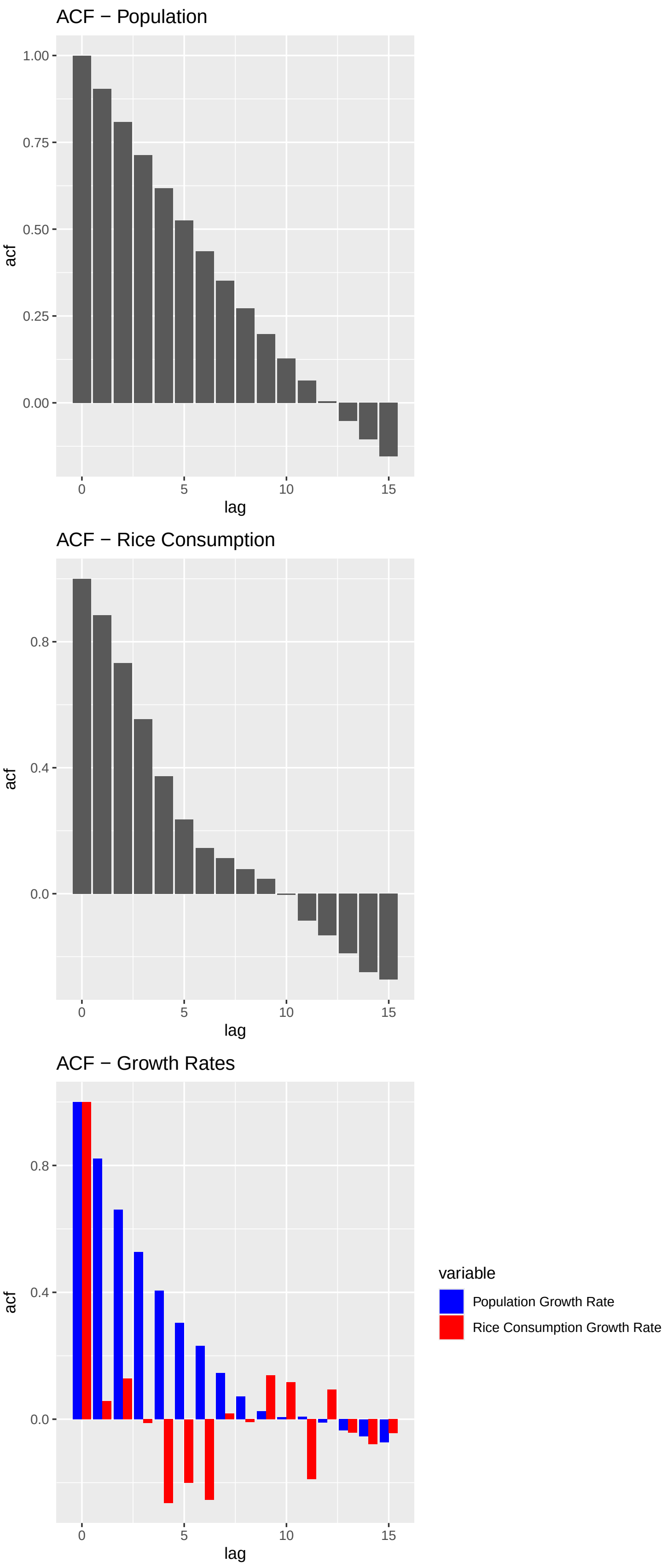
```r
# Apply for each country
multi_combined %>%
  group_split(country) %>%
  walk(~ plot_acf_country(
    df = plot_data %>% filter(country == unique(.x$country)),
    country_name = unique(.x$country)
  ))
```

```
##
## Attaching package: 'patchwork'

## The following object is masked from 'package:MASS':
##
##     area
```

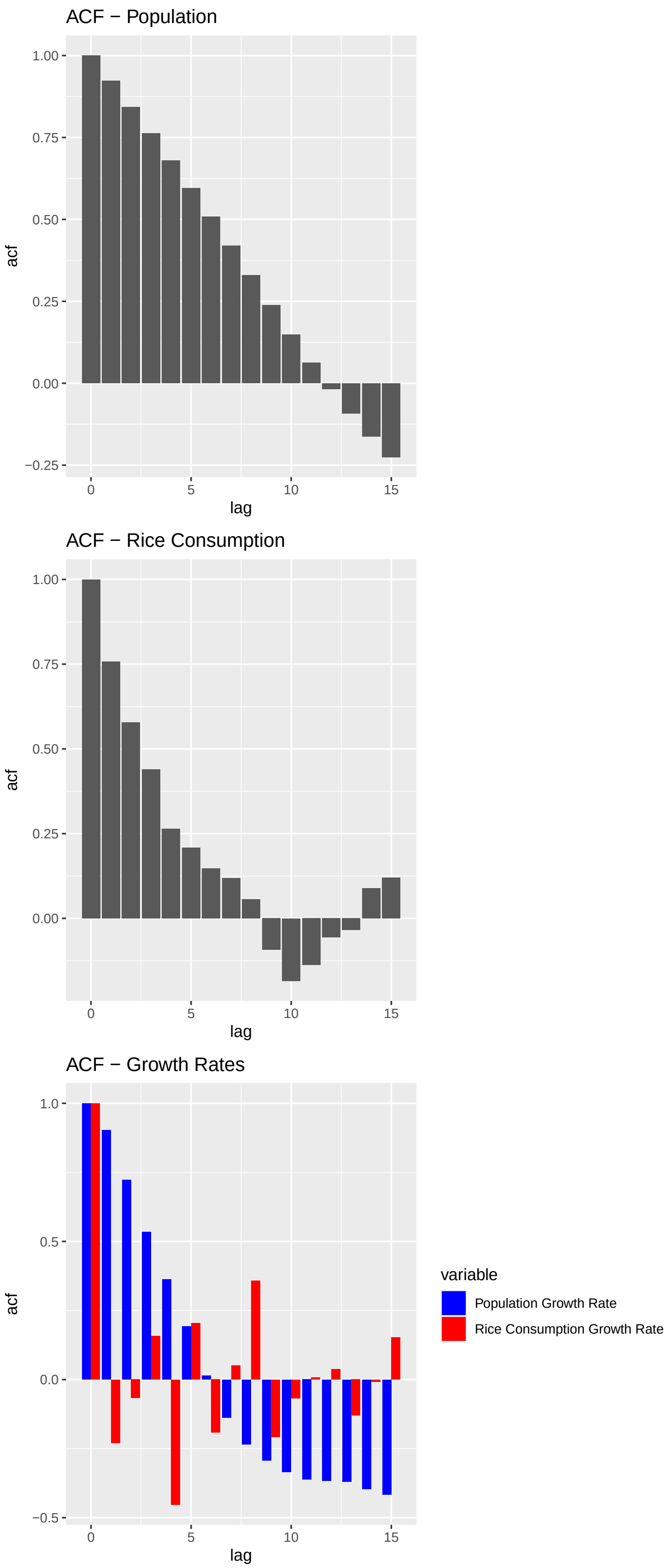
ACF plots for Brazil



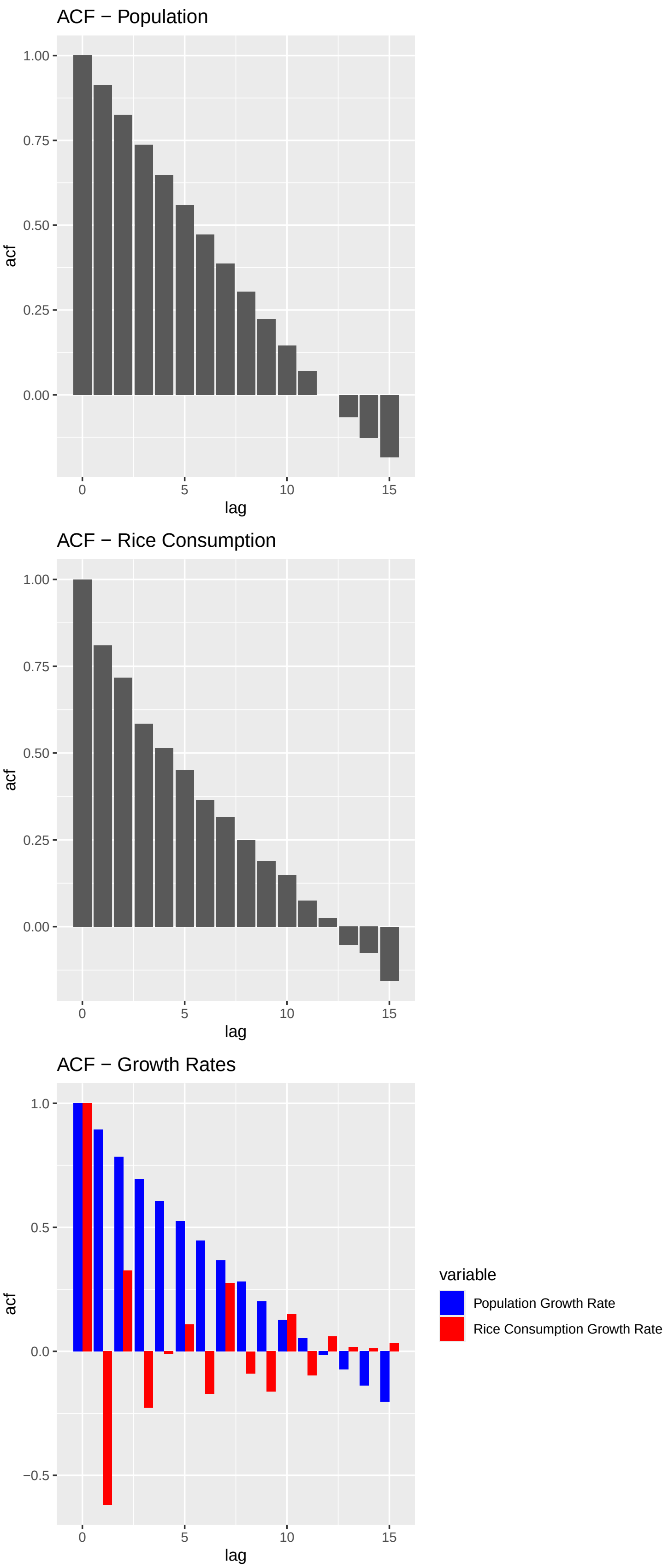
ACF plots for China



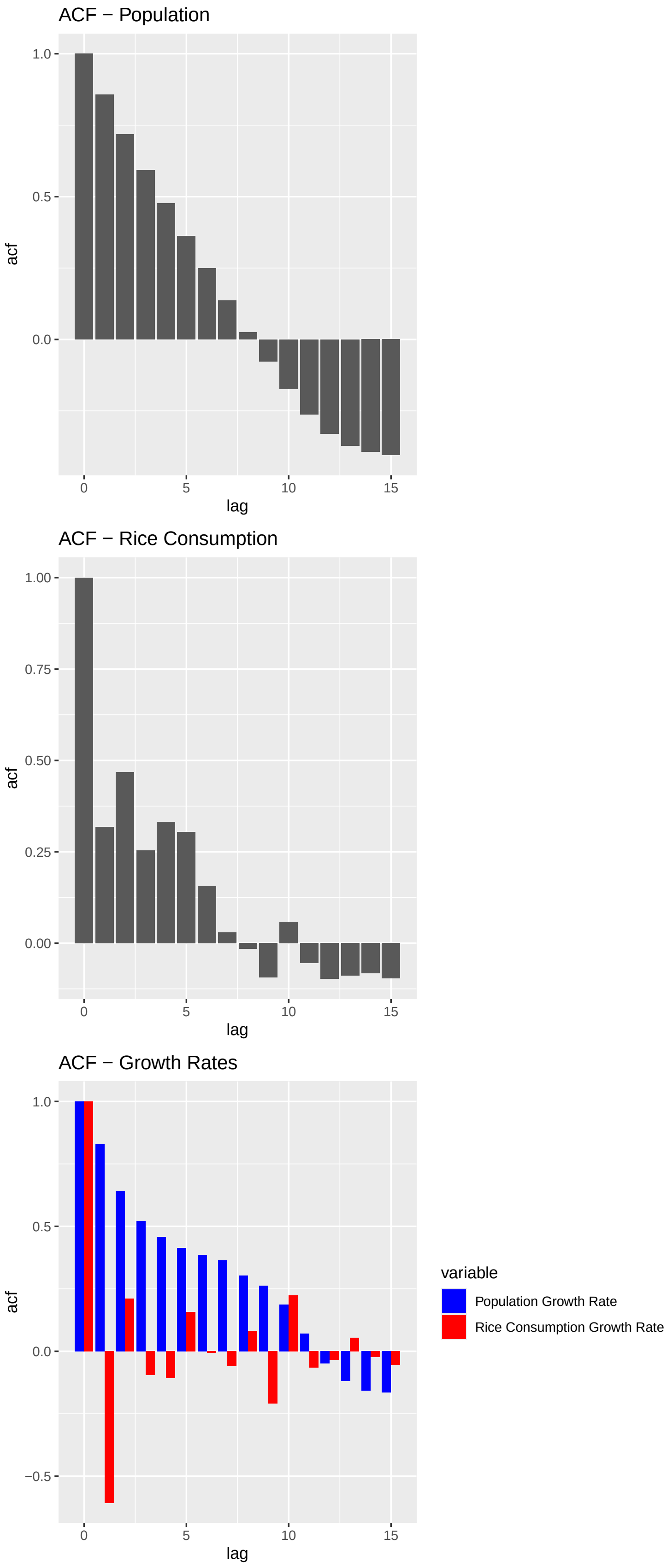
ACF plots for France



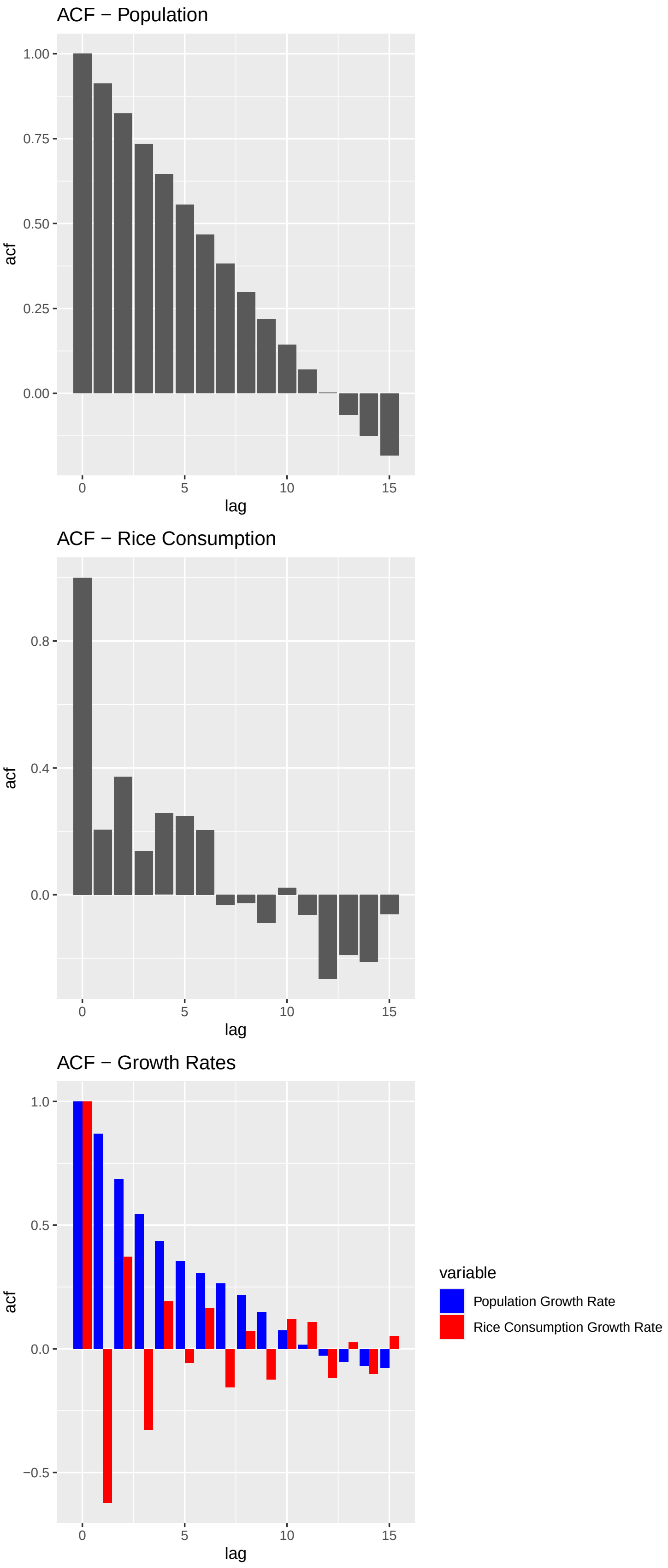
ACF plots for India



ACF plots for Japan



ACF plots for United States of America



The ACF takes roughly 10 lags (10 years) to go to zero, and there exist some cycles in autocorrelation. Even after taking the log transform, the data still exhibit alternating autocorrelation, indicating the data is not stationary and the statistical significance obtained earlier is spurious.

4. Instrumental Variable (IV) Methods

Model Description & Assumptions: We explore using **Rice Consumption** as an instrumental variable (IV) for **Population** to estimate the causal effect of population growth on **GDP**.

- **Structural Equation**:

$$\Delta \log(\text{GDP}_t) = \beta_0 + \beta_1 \Delta \log(\text{Population}_t) + \epsilon_t$$

- **First Stage**:

$$\Delta \log(\text{Population}_t) = \gamma_0 + \gamma_1 \Delta \log(\text{RiceConsumption}_t) + \nu_t$$

- **Assumptions**:
  1. **Relevance**: Rice consumption is correlated with population (more people consume more staples).
  2. **Exclusion**: Rice consumption affects GDP *only* through its effect on population size.

R Code & Analysis:

```r
# Calculate Growth Rates (Log differences)
iv_data <- multi_combined %>%
  group_by(country) %>%
  mutate(
    gdp_growth = c(NA, diff(log(GDP))),
    pop_growth = c(NA, diff(log(population))),
    rice_growth = c(NA, diff(log(rice_consumption)))
  ) %>%
  drop_na() %>%
  ungroup()

# 1. Naive OLS (Population Growth -> GDP Growth)
ols_naive <- lm(gdp_growth ~ pop_growth, data = iv_data)

# 2. IV Regression (Instrumenting Pop Growth with Rice Growth)
iv_model <- ivreg(gdp_growth ~ pop_growth | rice_growth, data = iv_data)

# Compare results
stargazer(ols_naive, iv_model, type = "text",
          title = "IV Regression Results: Effect of Population on GDP",
          column.labels = c("OLS (Naive)", "IV (Rice as Instr)"),
          model.names = FALSE)
```

```
##
## IV Regression Results: Effect of Population on GDP
## ==========================================================================
##                                           Dependent variable:
```

```
##                                 ------------------------------------------
##                                                 gdp_growth
##                                       OLS (Naive)       IV (Rice as Instr)
##                                           (1)                  (2)
## -------------------------------------------------------------------------
## pop_growth                             2.159***               2.166
##                                         (0.442)              (4.179)
##
## Constant                               0.019***               0.019
##                                         (0.004)              (0.033)
##
## -------------------------------------------------------------------------
## Observations                             192                  192
## R2                                      0.112                0.112
## Adjusted R2                             0.107                0.107
## Residual Std. Error (df = 190)          0.034                0.034
## F Statistic                    23.897*** (df = 1; 190)
## =========================================================================
## Note:                                         *p<0.1; **p<0.05; ***p<0.01
```

```r
# Diagnostic Tests for IV
cat("\n--- IV Diagnostics ---\n")
```

```
##
## --- IV Diagnostics ---
```

```r
summary(iv_model, diagnostics = TRUE)
```

```
##
## Call:
## ivreg(formula = gdp_growth ~ pop_growth | rice_growth, data = iv_data)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.103066 -0.017847 -0.005322  0.019555  0.102315
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.01872    0.03321   0.564    0.574
## pop_growth   2.16587    4.17919   0.518    0.605
##
## Diagnostic tests:
##                  df1 df2 statistic p-value
## Weak instruments   1 190     2.146   0.145
## Wu-Hausman         1 189     0.000   0.999
## Sargan             0  NA        NA      NA
##
## Residual standard error: 0.0344 on 190 degrees of freedom
## Multiple R-Squared: 0.1117,  Adjusted R-squared: 0.107
## Wald test: 0.2686 on 1 and 190 DF,  p-value: 0.6049
```

Interpretation:

- **Weak Instruments**: The diagnostic test checks if rice consumption is a strong predictor of population. Here, the non-significant test statistic ($p > 0.05$) suggests relevance doesn't hold.
- **Wu-Hausman**: Tests whether the OLS and IV estimates are significantly different. Here, $p > 0.05$; endogeneity is not present, and OLS is not inconsistent.
- **Coefficient**: The IV coefficient for population is larger than OLS; this suggests OLS underestimated the effect (possibly due to measurement error or omitted variable bias). However, due to the exclusion restriction violation, the IV estimate may capture the direct effect of consumption on GDP, inflating the result.

5. Three-Stage Recursive VAR Model

Model Description & Assumptions: We use a **Vector Autoregression (VAR)** model to capture the dynamic interrelationships between Rice Consumption, GDP, and Population. Unlike single-equation models, VAR treats all variables as endogenous.

- **VAR System (Lag p)**: **Reduced-Form VAR**

Let

$$Y_t = \begin{bmatrix} \Delta\log(\text{Rice}_t) \\ \Delta\log(\text{GDP}_t) \\ \Delta\log(\text{Pop}_t) \end{bmatrix}$$

.

The reduced-form VAR(p) is:

$$Y_t = A_0 + A_1 Y_{t-1} + \cdots + A_p Y_{t-p} + u_t$$

Where $Y_t = [\Delta\log(\text{Rice}_t), \Delta\log(\text{GDP}_t), \Delta\log(\text{Pop}_t)]'$,$ u_t $ is the vector of reduced-form (correlated) errors.

Structural VAR (SVAR) Transformation

The structural VAR can be written as:

$$BY_t = C_0 + C_1 Y_{t-1} + \cdots + C_p Y_{t-p} + \epsilon_t$$

where: - $ B $ is a 3 x 3 contemporaneous impact matrix (with 1's on the diagonal and off-diagonal elements representing contemporaneous effects). - $ _t $ are **structural shocks** (mutually uncorrelated, often economically interpretable). solving for $ Y_t $:

$$Y_t = B^{-1}C_0 + B^{-1}C_1 Y_{t-1} + \cdots + B^{-1}C_p Y_{t-p} + B^{-1}\epsilon_t$$

Relation between Reduced and Structural Errors

$$u_t = B^{-1}\epsilon_t$$

where $ u_t $ are reduced-form errors and $ _t $ are orthogonalized "structural" shocks.

- **Causal Ordering (Cholesky Decomposition)**: To identify structural shocks in Impulse Response Functions (IRF), we assume the following recursive ordering for contemporaneous effects:
  1. **Rice Consumption** (Fastest adjustment)
  2. **GDP** (Responds to Rice, but not Population immediately)
  3. **Population** (Slowest/Lagged response; responds to Rice & GDP only with a lag)

  Path: Rice → GDP → Population.

R Code & Analysis:

```
# Prepare Data: Growth Rates for VAR (Focusing on China as a representative case)
country_var_data <- iv_data %>%
  filter(country == "China") %>%
  dplyr::select(rice_growth, gdp_growth, pop_growth) %>%
  as.data.frame()

# 1. Lag Selection
# We test information criteria (AIC, HQ, SC) to find the optimal lag length
lag_selection <- VARselect(country_var_data, lag.max = 5, type = "const")
print(lag_selection$selection)
```

```
## AIC(n)  HQ(n)  SC(n) FPE(n)
##      2      1      1      2
```

```
best_lag <- lag_selection$selection["AIC(n)"]

cat("\nSelected Lag based on AIC:", best_lag, "\n")
```

```
##
## Selected Lag based on AIC: 2
```

```
# 2. Fit the VAR Model
var_model <- VAR(country_var_data, p = best_lag, type = "const")
summary(var_model)
```

```
##
## VAR Estimation Results:
## =========================
## Endogenous variables: rice_growth, gdp_growth, pop_growth
## Deterministic variables: const
## Sample size: 30
## Log Likelihood: 336.884
## Roots of the characteristic polynomial:
## 0.9962 0.6747 0.6747 0.4825 0.4825 0.3618
## Call:
## VAR(y = country_var_data, p = best_lag, type = "const")
##
##
## Estimation results for equation rice_growth:
## ============================================
## rice_growth = rice_growth.l1 + gdp_growth.l1 + pop_growth.l1 + rice_growth.l2 + gdp_growth.l2 + po
```

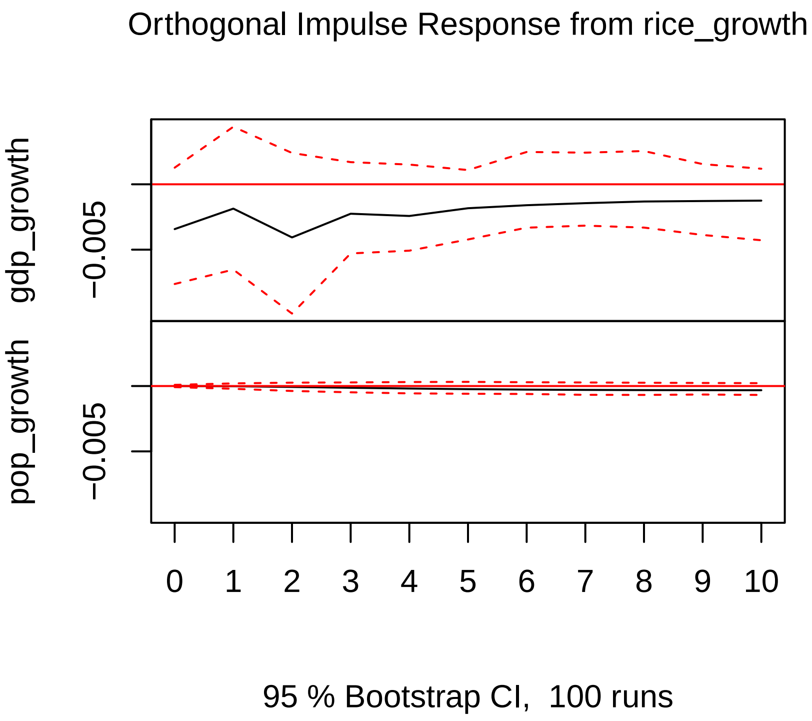
```
##
##                 Estimate Std. Error t value Pr(>|t|)
## rice_growth.l1 -0.08830    0.18690  -0.472   0.6410
## gdp_growth.l1  -0.07278    0.38364  -0.190   0.8512
## pop_growth.l1   2.18957   13.27061   0.165   0.8704
## rice_growth.l2  0.05511    0.18138   0.304   0.7640
## gdp_growth.l2   0.89383    0.44542   2.007   0.0567 .
## pop_growth.l2  -7.19005   12.55640  -0.573   0.5725
## const          -0.03182    0.02855  -1.115   0.2766
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.03411 on 23 degrees of freedom
## Multiple R-Squared:  0.28,   Adjusted R-squared: 0.09222
## F-statistic: 1.491 on 6 and 23 DF,  p-value: 0.2253
##
##
## Estimation results for equation gdp_growth:
## ===========================================
## gdp_growth = rice_growth.l1 + gdp_growth.l1 + pop_growth.l1 + rice_growth.l2 + gdp_growth.l2 + pop
##
##                 Estimate Std. Error t value Pr(>|t|)
## rice_growth.l1 -0.01179    0.10024  -0.118   0.9074
## gdp_growth.l1   0.42583    0.20576   2.070   0.0499 *
## pop_growth.l1   0.29697    7.11744   0.042   0.9671
## rice_growth.l2 -0.06064    0.09728  -0.623   0.5392
## gdp_growth.l2   0.36007    0.23889   1.507   0.1454
## pop_growth.l2   0.43893    6.73439   0.065   0.9486
## const           0.01050    0.01531   0.686   0.4998
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.01829 on 23 degrees of freedom
## Multiple R-Squared: 0.5732,  Adjusted R-squared: 0.4618
## F-statistic: 5.148 on 6 and 23 DF,  p-value: 0.001746
##
##
## Estimation results for equation pop_growth:
## ===========================================
## pop_growth = rice_growth.l1 + gdp_growth.l1 + pop_growth.l1 + rice_growth.l2 + gdp_growth.l2 + pop
##
##                   Estimate Std. Error t value Pr(>|t|)
## rice_growth.l1  0.0002995  0.0025090   0.119  0.90602
## gdp_growth.l1   0.0090022  0.0051502   1.748  0.09381 .
## pop_growth.l1   1.4608106  0.1781525   8.200  2.8e-08 ***
## rice_growth.l2 -0.0004907  0.0024349  -0.202  0.84205
## gdp_growth.l2   0.0026949  0.0059796   0.451  0.65643
## pop_growth.l2  -0.5108832  0.1685646  -3.031  0.00594 **
## const          -0.0008893  0.0003833  -2.321  0.02954 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.0004579 on 23 degrees of freedom
## Multiple R-Squared: 0.9791,  Adjusted R-squared: 0.9736
## F-statistic: 179.5 on 6 and 23 DF,  p-value: < 2.2e-16
##
##
##
## Covariance matrix of residuals:
##             rice_growth gdp_growth pop_growth
## rice_growth   1.163e-03 -1.169e-04  1.013e-07
## gdp_growth   -1.169e-04  3.346e-04 -4.263e-07
## pop_growth    1.013e-07 -4.263e-07  2.096e-07
##
## Correlation matrix of residuals:
##             rice_growth gdp_growth pop_growth
## rice_growth    1.000000    -0.1874   0.006485
## gdp_growth    -0.187384     1.0000  -0.050902
## pop_growth     0.006485    -0.0509   1.000000
```

```
# 3. Granger Causality Tests
# Does Rice Growth Granger-cause GDP Growth?
granger_rice_gdp <- causality(var_model, cause = "rice_growth")
print(granger_rice_gdp)
```

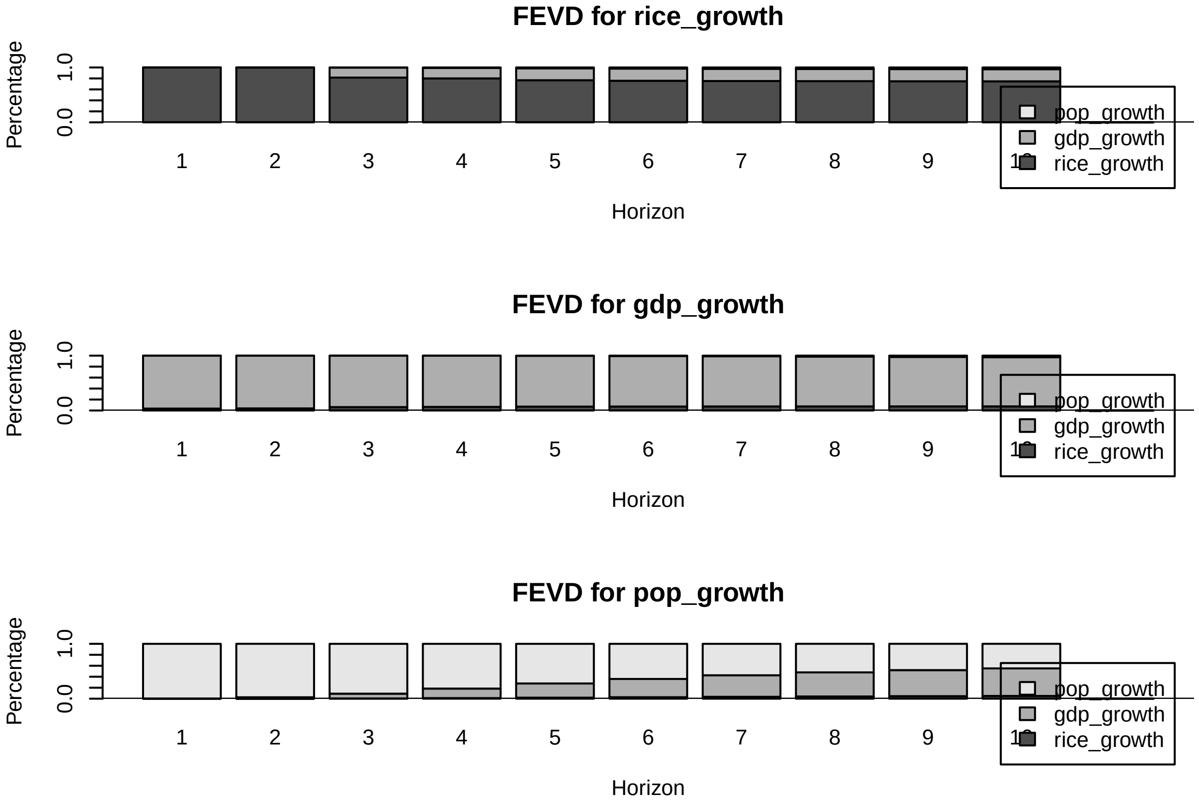
```
## $Granger
##
##  Granger causality H0: rice_growth do not Granger-cause gdp_growth
##  pop_growth
##
## data:  VAR object var_model
## F-Test = 0.11738, df1 = 4, df2 = 69, p-value = 0.9759
##
##
## $Instant
##
##  H0: No instantaneous causality between: rice_growth and gdp_growth
##  pop_growth
##
## data:  VAR object var_model
## Chi-squared = 1.0179, df = 2, p-value = 0.6011
```

```
# 4. Impulse Response Analysis (IRF)
# We use the specified causal ordering: Rice -> GDP -> Pop
```

```r
irf_result <- irf(var_model,
                  impulse = "rice_growth",
                  response = c("gdp_growth", "pop_growth"),
                  boot = TRUE, runs = 100, n.ahead = 10,
                  ortho = TRUE) # Cholesky orthogonalization used

# Plot IRF
plot(irf_result)
```



```r
# 5. Variance Decomposition (FEVD)
# Shows how much of the forecast error variance of each variable can be explained by shocks to the o
fevd_res <- fevd(var_model, n.ahead = 10)
plot(fevd_res)
```



Interpretation:

- **Lag Selection**: We choose the lag p (e.g., 1 or 2 years) that minimizes the AIC to capture dynamics without overfitting.
- **Granger Causality**: The p-value is > 0.05; past values of Rice Consumption do not contain statistically significant information to predict current GDP, supporting a temporal link.
- **Impulse Response (IRF)**:
  - The plots show the reaction of GDP and Population over 10 years to a one-standard-deviation shock in Rice Consumption.
  - We expect Population to respond slowly (lagged positive effect) while GDP might respond faster, which is weakly represented in the plot.
  - The confidence interval (dotted lines) includes zero; thus, the response is not significant.
- **Variance Decomposition**: Tells us "what drives what." Population was explained by a growing share of GDP variance over time, confirming a strong linkage between economic growth and demographic trends. On the other hand, rice consumption was weakly explained by a small but growing share of population variance over time, confirming a mild linkage between inelastic daily commodities and demographic trends. Whereas the rice consumption shock did not explain any variable.

6. Conclusion

We notice the data exhibited clustering and non-homogeneity across countries and autocorrelation; traditional regression methods result in spurious regression. The data clustering cannot be handled by simply labeling with binary development status, and the autocorrelation still persists after taking log differencing.

After accounting for the time series properties (GLS model using AR(1) error term), the result is not significant anymore. To conclude, we did not find any statistically significant evidence supporting the association between rice consumption and demographic or economic trends.

- Empirical results showed non-significant statistical evidence in support of the relationship between inelastic good consumption, population, and economic development.

7. Areas for Improvement

1. **Non-independent data**: Countries like China experienced significant change in the past 50 years relative to other countries. It might not be sufficient to only classify them based on development status; other categorical labels might be needed. Moreover, we might try to replace China with another country with more "stable" data.
2. **Explicit link between rice consumption and GDP**: In the formula for calculating GDP ($GDP = C+I+G+NX$), consumption is a direct component, which makes the IV econometric model exclusion restriction likely violated. Other econometric approaches should be considered.
3. **Time-series lagged relationship**: The VAR model assumed a recursive relationship from consumption to GDP, then to population. However, this project's data is annual, meaning the lagged response might happen within a year that we can't capture using this annual VAR model. Additionally, there is a question about convergence rate from consumption to GDP since GDP is directly related to consumption. One possible alternative data source would be using demand data instead of consumption data to forecast market movement, similar to using the VIX index in the stock market (volatility index).