

STAT 107 Project:PHC

Marianne, Donnie, Uriel

2025-10-14

Introduction

Contributions: Donnie: Created and finished a RMD file, helped with smoothing out the structure of all the writing, helped created the sub-question, and brain stormed ideas for analysis.

Marianne: finished up visualization section, helped eliminating non-essential data sets, added/did 00.requirement.R, and introduced Population growth as a potential factor.

Uriel: Work mainly on the code itself, helped set up the main structure of 'STAT 107 Project:PHC', find the majority of the data set and original question, and communicated the most. # Purpose: The purpose of this experiment was to create and train a model to not only establish a connection with the Poverty headcount ratio between the Inflation and Population Growth. But potential create a means to see trends that will eventually help foresee potential spikes worldwide. This analysis should be to benefit government institutions that not only track but try to influence how much inflation would inflict upon society as a whole. Moreover to see as if the world's population continues to rise, the poverty line would be swayed by it. This would implicate federal policies that end up being passed or being killed in the senate. Which in turn would affect local business all the way to international ones and even affect industries on multiple levels. # Question -Main Question: Is there a statistically significant correlation between inflation rate and the Poverty headcount ratio across countries worldwide? -Sub-Question: Which of the two effects; inflation rate or population growth, have a bigger influenced when it comes to affecting the Poverty headcount ratio?

Benefit:

This analysis should be to benefit government institutions that not only track but try to influence how much inflation would inflict upon society as a whole. This would implicate federal policies that end up being passed or killed in the senate. Which in turn would affect local business all the way to international ones.

Plan:

We hope to answer our question since, if we are able to establish a connection between inflation to PHC. Then going forward it would be possible to train a model that could potential forecast the trends of inflation which would help government bodies worldwide create plans to counteract these shifts. However for the sub question, it is more of means of determining which of the two effects have a bigger influence if they do.

Data

The data that is available is NEW_DDAY and NEW_DDAY dos are from the same set (Poverty Head Count at \$3.00 a day (2021 PPP)), API_1 and API_2 are from the same data set (Inflation consumer prices (annual %) for the World, Grow (Population growth (annual%))). However API_2 and NEW_DDAY dos is less about data and its importance is more of the information it can present since it contains notes left behind from the makers of the csv file. That would explain why certain values appeared. ## Data sets:

```
ND <- read.csv("New_DDAY.csv")
ND2 <- read.csv("NEW_DDAY_dos.csv")
API <- read.csv("API_A.csv")
SN_API <- read.csv("API_B.csv")
Pop_Growth <- read.csv("Pop_Growth.csv")
```

Cleaning the Data:

For each of the three data sets, it was needed to recreate a data frame since the original csv file showed each excel file to have their rows and columns in the wrong place. Since basic R function like summary() or mean() would grab information from the file's columns, it would output an answer. However the number result would be wrong since it would be grabbing all 270 countries info for 1960. So df_ND,API,PG are just data frames with all the numeric information of the orginal csv file. So it looks more visually more appealing and so it be a ton easier to use any built in graphing function that will not produce the wrong result. # Code for New_DDAY

```

#Item names: row_ND, not_imp, ind_row, RC, row_ND, vari_RC.
row_ND <- c()
not_imp <- c() #useless vector
for (ind_row in rownames(ND)){
  for (i in ND[ind_row,]){
    RC <- as.character(i)
    if (is.na(RC)){
      RC <- 0
      row_ND <- c(row_ND, RC)

    } else {
      vari_RC <- suppressWarnings(as.numeric(RC))

      if (!is.na(vari_RC)){
        row_ND <- c(row_ND, RC)
      } else if (RC != ""){
        not_imp <- c(not_imp, RC)
      }
    }
  }
}

```

`find_ND`: Get groups of 270 and each group has a length of 65. Then use `split()` to apply that `row_ND`

```

a <- row_ND
ND_by_65 <- rep(1:270, each = 65)
fin_ND <- split(a, ND_by_65)

```

Data Frame for ND with rows and columns switched: `df_ND`

```
# must have this chunk running first if you want the other similar section to function!!
col_nm <- ND[,1][5:270]
df_ND <- data.frame(fin_ND[5:270])

colnames(df_ND) <- c(col_nm)
years <- c(1960:2024)
rownames(df_ND) <- c(years)
```

Supporting data sets: API and Pop_Growth

```
#Item names: row_API, not_imp_2, ind_API, RC_2, vari_RC_2.
row_API <- c()
not_imp_2 <- c() #useless vector
for (ind_API in rownames(API)){
  for (i in API[ind_API,]){
    RC_2 <- as.character(i)
    if (is.na(RC_2)){
      RC_2 <- 0
      row_API <- c(row_API, RC_2)

    } else {
      vari_RC_2 <- suppressWarnings(as.numeric(RC_2))

      if (!is.na(vari_RC_2)){
        row_API <- c(row_API, RC_2)
        } else if (RC_2 != ""){
          not_imp_2 <- c(not_imp_2, RC_2)
        }
      }
    }
}
```

Look at main data set grouping section for explanation

```
b <- row_API  
API_by_65 <- rep(1:270, each = 65)  
fin_API <- split(b, API_by_65)
```

Data Frame for API with rows and columns switched: df_API

```
col_nm_2 <- API[,1][5:270]  
df_API <- data.frame(col_nm = fin_API[5:270])  
  
colnames(df_API) <- c(col_nm_2)  
rownames(df_API) <- c(years)
```

Pop_Growth

```
#Item names: row_PG, NI, ind_PG, RC_3, vari_RC_3.  
row_PG <- c()  
NI <- c() #useless vector  
for (ind_PG in rownames(Pop_Growth)){  
  for (i in Pop_Growth[ind_PG, ]){  
    RC_3 <- as.character(i)  
    if (is.na(RC_3)){  
      RC_3 <- 0  
      row_PG <- c(row_PG, RC_3)  
  
    } else {  
      vari_RC_3 <- suppressWarnings(as.numeric(RC_3))  
  
      if (!is.na(vari_RC_3)){  
        row_PG <- c(row_PG, RC_3)  
      } else if (RC_3 != ""){  
        NI <- c(NI, RC_3)  
      }  
    }  
  }  
}
```

Look at main data set grouping section for explanation

```
c <- row_PG  
PG_by_65 <- rep(1:270, each = 65)  
fin_PG <- split(c, PG_by_65)
```

Data Frame for Pop_Growth with rows and columns switched:

df_PG

```
col_nm_3 <- Pop_Growth[,1][5:270]
df_PG <- data.frame(col_nm = fin_PG[5:270])

colnames(df_PG) <- c(col_nm_3)
rownames(df_PG) <- c(years)
```

Variables and number of observations:

1. How many observations: `nrow()` function.
2. How many variables: `ncol()` function.
3. Dimensions of data frame: `dim()` function.

```
print("All three Data Sets have the same rows, columns and dimensions")
```

```
## [1] "All three Data Sets have the same rows, columns and dimensions"
```

```
nrow(df_ND) # No. of observations.
```

```
## [1] 65
```

```
ncol(df_ND) # No. of variables of interest.
```

```
## [1] 266
```

```
dim(df_ND) # both.
```

```
## [1] 65 266
```

Will the data be generated through a randomized simulation?

In our case no, unless forecast counts as randomzied simulations.

Visualization:

```
##df_ND[,266] is the last country (Zimbabwe). Because I set fin_ND[5:270], so we can skip the first 3 rows of zeros and country years
i = 1
while ( i < 267 ){
  sum_ND <- summary(as.double(df_ND[,i]))
  sum_API <- summary(as.double(df_API[,i]))
  sum_PG <- summary(as.double(df_PG[,i]))
  i = i + 1

}
```

Preliminary visualization:

```
library("ggplot2")
library("stargazer")
```

```
##
## Please cite as:
```

```
## Hlavac, Marek (2022). stargazer: Well-Formatted Regression and Summary Statistics Tables.
```

```
## R package version 5.2.3. https://CRAN.R-project.org/package=stargazer
```

#If we want a linear regression to run, we must keep each df_ND\$ as a double type.

```
A1 <- lm(as.double(df_API$Australia) ~ as.double(df_API$Austria), data = df_API)

B1 <- lm(as.double(df_API`Viet Nam`) ~ as.double(df_API$Vanuatu), data = df_API)

C1 <- lm(as.double(df_ND$Australia) ~ as.double(df_ND$Austria), data = df_ND)

D1 <- lm(as.double(df_ND`Viet Nam`) ~ as.double(df_ND$Vanuatu), data = df_ND)

E1 <- lm(as.double(df_ND$Spain) ~ as.double(df_ND$Estonia), data = df_ND)
```

#four is the max amount of lm() we can have per stargazer(), because 5 becomes a crowded.

```
stargazer(A1,B1,C1,D1,
          header=FALSE,
          font.size = "tiny",
          type = "text",
          algin = TRUE,
          single = FALSE,
          column.sep.width = "1pt",
          digits =2)
```

```

##  

## =====  

## Dependent variable:  

## -----  

##          Australia) `Viet Nam` ) Australia) `Viet Nam` )  

##          (1)        (2)        (3)        (4)  

## -----  

## Austria)           1.17***  

##                   (0.16)  

##  

## Vanuatu)          0.90***  

##                   (0.25)  

##  

## Austria)           0.20  

##                   (0.17)  

##  

## Vanuatu)          -0.03  

##                   (0.35)  

##  

## Constant           0.73      1.13*     0.12**    3.57**  

##                   (0.64)     (0.57)     (0.05)    (1.39)  

##  

## -----  

## Observations       65        65        65        65  

## R2                 0.45      0.17      0.02      0.0001  

## Adjusted R2         0.44      0.16      0.01      -0.02  

## Residual Std. Error (df = 63) 2.75      3.97      0.32      11.06  

## F Statistic (df = 1; 63)   52.12***  13.25***  1.50      0.01  

## =====  

## Note:                  *p<0.1; **p<0.05; ***p<0.01  

##  

## ====  

## TRUE  

## ----  

##  

## ====  

## FALSE  

## -----

```

```
stargazer(E1,
  header=FALSE,
  font.size = "tiny",
  type = "text",
  algin = TRUE,
  single = FALSE,
  column.sep.width = "1pt",
  digits =2)
```

```
##  
## ======  
##           Dependent variable:  
##           -----  
##           Spain)  
## -----  
## Estonia)          0.24***  
##                  (0.07)  
##  
## Constant          0.36***  
##                  (0.07)  
##  
## -----  
## Observations      65  
## R2                 0.14  
## Adjusted R2        0.12  
## Residual Std. Error    0.48 (df = 63)  
## F Statistic         10.11*** (df = 1; 63)  
## ======  
## Note:             *p<0.1; **p<0.05; ***p<0.01  
##  
## ====  
## TRUE  
## ----  
##  
## =====  
## FALSE  
## -----
```

Analysis:

associated relocation of p-values of each Linear regression for each country would be placed here.