Salarios y Empleos de Puerto Rico 2001-2022

2025-09-03

Salarios y Empleos de Puerto Rico 2001-2022

The purpose of this exercise is to see and have a better understanding of the job market in Puerto Rico for the year 2022. The data used for this exercise from the Occupational Employment and Wage Statistics of the Bureu of Labor Statistics of the United States. For more information please visit https://www.bls.gov/oes/oes_emp.htm. All the data was exported from May 2022.

One of the main exercises which is goung to be presented here is the concept of Elasticity of Wages. Here we will be using Linear Regression to estimate the elasticity of wages from Puerto Rico.

First we load the libraries needed

```
library(readxl)
library(tidyverse)
library(janitor)
library(broom)
library(gridExtra)
library(grid)
library(ggpubr)
library(xts)
library(data.table)
library(kableExtra)
```

Data Wrangling

The data set brings all 50 states and some territories of the Unites States as well as different names column names. let's clean it up a little.

Creating a function to read all the files from a local folder

```
process_grouped_file <- function(year, file_path, group) {
  df <- suppressWarnings(read_excel(file_path)) %>% clean_names()

  # Define possible column names for each target
  rename_map <- list(
    a_mean      = c("a_mean"),
    area_title  = c("area_title", "state"),
    prim_state  = c("prim_state", "st"),
    occ_title   = c("occ_title", "occ_titl"),
```

```r
    occ_code    = c("occ_code"),
    tot_emp     = c("tot_emp")
  )

# Build correct rename list: new_name = old_name
  actual_rename <- list()
  for (std_name in names(rename_map)) {
    found <- intersect(rename_map[[std_name]], names(df))
    if (length(found) > 0) {
      actual_rename[[std_name]] <- found[1]  # this is the correct order: new = old
    }
  }

# Only proceed if all required final names are in the rename list
  required <- c("a_mean", "area_title", "prim_state", "occ_title", "occ_code", "tot_emp")
  if (!all(required %in% names(actual_rename))) {
    warning(paste("Skipping year", year, "- missing required columns"))
    return(NULL)
  }

# Rename columns
  df <- df %>%
    rename(!!!actual_rename) %>%
    select(all_of(required)) %>%
    mutate(
      A_MEAN = as.numeric(gsub("\\*", "", a_mean)),
      TOT_EMP = as.numeric(gsub("\\*", "", tot_emp)),
      YEAR = year
    ) %>%
    filter(!is.na(A_MEAN), !is.na(TOT_EMP)) %>%
    select(A_MEAN, AREA_TITLE = area_title, PRIM_STATE = prim_state,
           OCC_TITLE = occ_title, OCC_CODE = occ_code, TOT_EMP, YEAR) %>%
    arrange(A_MEAN)

  return(df)
}
```

##Setting the paramters for the process_grouped_file function

We can use the setwd() with the folder path to read the data *"C:/Users/. . . "*

```r
file_info <- data.frame(
  year = 2001:2021,
  file = c(
    "state_2001_dl.xls", "state_2002_dl.xls",
    "state_May2003_dl.xls", "state_May2004_dl.xls", "state_May2005_dl.xls",
    "state_May2006_dl.xls", "state_May2007_dl.xls", "state__M2008_dl.xls",
    "state_2009_dl.xls", "state_M2010_dl.xls", "state_M2011_dl.xls", "state_M2012_dl.xls",
    "state_M2013_dl.xls", "state_M2014_dl.xlsx", "state_M2015_dl.xlsx",
    "state_M2016_dl.xlsx", "state_M2017_dl.xlsx", "state_M2018_dl.xlsx",
    "state_M2019_dl.xlsx", "state_M2020_dl.xlsx", "state_M2021_dl.xlsx"
  ),
  group = c(
    rep("C", 12),  # 1999-2012
```

```r
    rep("B", 7),    # 2013-2019
    rep("A", 2)     # 2020-2021
  ),
  stringsAsFactors = FALSE
)
```

Exectuing the process_grouped_file Function

```r
setwd("C:/Users/yadel/OneDrive/Documents/1999-2021/")

Salarios_all <- bind_rows(lapply(1:nrow(file_info), function(i) {
  process_grouped_file(
    year = file_info$year[i],
    file_path = file_info$file[i],
    group = file_info$group[i]
  )
}))
```

```r
Salarios_PR <- Salarios_all%>%
  filter(PRIM_STATE == "PR", !is.na(OCC_TITLE),
         !is.na(OCC_CODE), !is.na(A_MEAN), !is.na(TOT_EMP))
```

Estimating Elasticity with a Log-Log Regression Model

Elasticity measures how responsive one variable is to changes in another.
In this case, we evaluate how **total employment** (`TOT_EMP`) responds to changes in **average wages** (`A_MEAN`).

To estimate this relationship, we use a **log-log linear regression model**:

$$\log(Y) = \beta_0 + \beta_1 \cdot \log(X) + \varepsilon$$

Where:
- Y is the dependent variable (e.g., total employment)
- X is the independent variable (e.g., average wage)
- β_1 is the **elasticity coefficient**
- ε is the error term

In this model, β_1 represents the **percentage change in** Y for a **1% change in** X:

$$\frac{d\log(Y)}{d\log(X)} = \beta_1$$

Interpretation:

- If $|\beta_1| > 1$: **Elastic** — employment responds strongly to wage changes
- If $|\beta_1| < 1$: **Inelastic** — employment responds weakly to wage changes

This modeling approach is commonly used in economics because it: - Handles non-linear relationships more effectively - Allows for easier interpretation in **percentage terms** - Normalizes variable scales, making comparisons more meaningful

Elasticity Models for each occupation

```
NNN1<-Salarios_PR %>% nest(data = -OCC_CODE)%>%
  mutate(model = map(data, ~lm(log(TOT_EMP)~log(A_MEAN), data = .)),
           tidied = map(model, glance)) %>%
  unnest(tidied)
```

Here we see the output for each ocupation.

```
title<- Salarios_PR%>%
  select(OCC_CODE, OCC_TITLE)%>%
  distinct(OCC_CODE, .keep_all = TRUE)

NNN<-Salarios_PR %>% nest(data = -OCC_CODE)%>%
  mutate(model = map(data, ~lm(log(TOT_EMP)~log(A_MEAN),, data = .)),
         tidied = map(model, tidy)) %>%
  unnest(tidied)

NNN<-NNN%>%
  filter(term == "log(A_MEAN)")%>%
  select(OCC_CODE,term, estimate,model, p_valor = p.value)


NNN1<-NNN1%>%
  inner_join(title,by = "OCC_CODE" )%>%
  distinct(OCC_CODE, .keep_all = TRUE)

NNN1<-NNN1%>%
  inner_join(NNN, by = "OCC_CODE")%>%
  distinct(OCC_CODE, .keep_all = TRUE)

NNN2<-Salarios_PR %>% nest(data = -OCC_CODE)%>%
  mutate(model = map(data, ~lm(log(TOT_EMP)~log(A_MEAN), data = .)),
         resid = map(model, residuals))%>%
  select(OCC_CODE,resid)


NNN1<-NNN1%>%
  inner_join(NNN2, by = "OCC_CODE")%>%
  distinct(OCC_CODE, .keep_all = TRUE)%>%
  mutate(Elasticidad = ifelse(abs(estimate)>1,"Elastica", "Inelastica"))

NNN1%>%
  select(-resid,-model.x,-model.y,-data,-term)%>%
  head(10)%>%
```

```
  kable(caption = "Elasticity Estimate No verification (Showing 10)", digits = 2) %>%
  kable_styling(full_width = FALSE, bootstrap_options = c("striped", "hover", "condensed"))
```

Table 1: Elasticity Estimate N

OCC_CODE	r.squared	adj.r.squared	sigma	statistic	p.value	df	logLik	AIC	BIC	deviance	df.resid
35-3022	0.33	0.28	0.15	6.38	0.03	1	8.18	-10.36	-8.24	0.30	
35-2021	0.03	-0.02	0.36	0.56	0.46	1	-6.69	19.39	22.37	2.29	
31-1011	0.02	-0.04	0.43	0.29	0.60	1	-9.32	24.65	27.32	2.97	
35-2011	0.02	-0.05	0.46	0.25	0.62	1	-10.60	27.19	29.87	3.42	
51-6031	0.34	0.30	0.18	9.17	0.01	1	7.09	-8.19	-5.20	0.58	
47-2041	0.33	0.00	0.21	1.00	0.42	1	1.86	2.28	0.44	0.09	
35-3021	0.06	0.00	0.20	1.01	0.33	1	4.66	-3.31	-0.64	0.63	
35-9021	0.36	0.33	0.12	10.24	0.00	1	14.30	-22.60	-19.61	0.28	
47-3014	0.28	0.18	0.81	2.73	0.14	1	-9.75	25.51	26.10	4.60	
53-6021	0.13	0.08	0.21	2.63	0.12	1	3.58	-1.16	1.83	0.82	

Now that we have our models we can verify the model assumptions.

Evaluating Elasticity Models

We validate the R-squares

```
NNN1<-NNN1%>%
  filter(p.value<0.01, p_valor<0.01)

Elasticidades<- NNN1%>%
  select(OCC_CODE, OCC_TITLE=  OCC_TITLE,resid, estimate,Elasticidad
         ,p.value,p_valor,r.squared,adj.r.squared,model.x)%>%
  arrange(OCC_CODE)%>%
  filter(r.squared> .70, adj.r.squared>.70)
```

We get the names of the occupations

```
ff<-map_df(Elasticidades$resid, ~as.data.frame(t(.)))
rownames(ff)<- Elasticidades$OCC_CODE
```

Normality test for the Data

```
# Get OCC_CODEs directly from Elasticidades
occ_codes <- Elasticidades$OCC_CODE
fff<-t(ff)

fff<- as.data.frame(fff)
```

```r
# Apply Shapiro-Wilk test and collect p-values
shapiro_test <- sapply(occ_codes, function(code) {
  x <- fff[[code]]
  if (all(is.na(x))) return(NA)  # handle NA-only columns safely
  shapiro.test(x)$p.value
})

# Convert to a clean data frame
shapiro_test <- data.frame(
  OCC_CODE = names(shapiro_test),
  p_value = unname(shapiro_test)
)

shapiro_test = shapiro_test %>%
  rename(shapiro_p = p_value)
```

Heteroskedasticity Test for the models

```r
library(lmtest)

bp_test <- sapply(Elasticidades[[10]], function(modelo) {
    tryCatch(bptest(modelo)$p.value, error = function(e) NA)
})

bp_test <- data.frame(matrix(unlist(bp_test)), nrow=length(bp_test), byrow=TRUE)

colnames(bp_test)<- c("bp_test", "nrow", "byrow")

bp_test<-bp_test%>%
  mutate(OCC_CODE = occ_codes)

bp_test <- bp_test %>%
  rename(bp_p = bp_test)
```

##Evaluating Hypothesis Tests

```r
Elasticidades<-Elasticidades%>%
  inner_join(shapiro_test, by = "OCC_CODE")%>%
  distinct(OCC_CODE, .keep_all = TRUE)

Elasticidades<-Elasticidades%>%
  inner_join(bp_test, by = "OCC_CODE")%>%
  distinct(OCC_CODE, .keep_all = TRUE)%>%
  select(-nrow,-byrow)

Elas<- Elasticidades%>%
  filter(bp_p>0.05,shapiro_p>0.05)%>%
  arrange(desc(estimate))%>%
  rename(modelo = model.x)

table(Elas$Elasticidad)
```

```
##
##   Elastica Inelastica
##         54          4
```

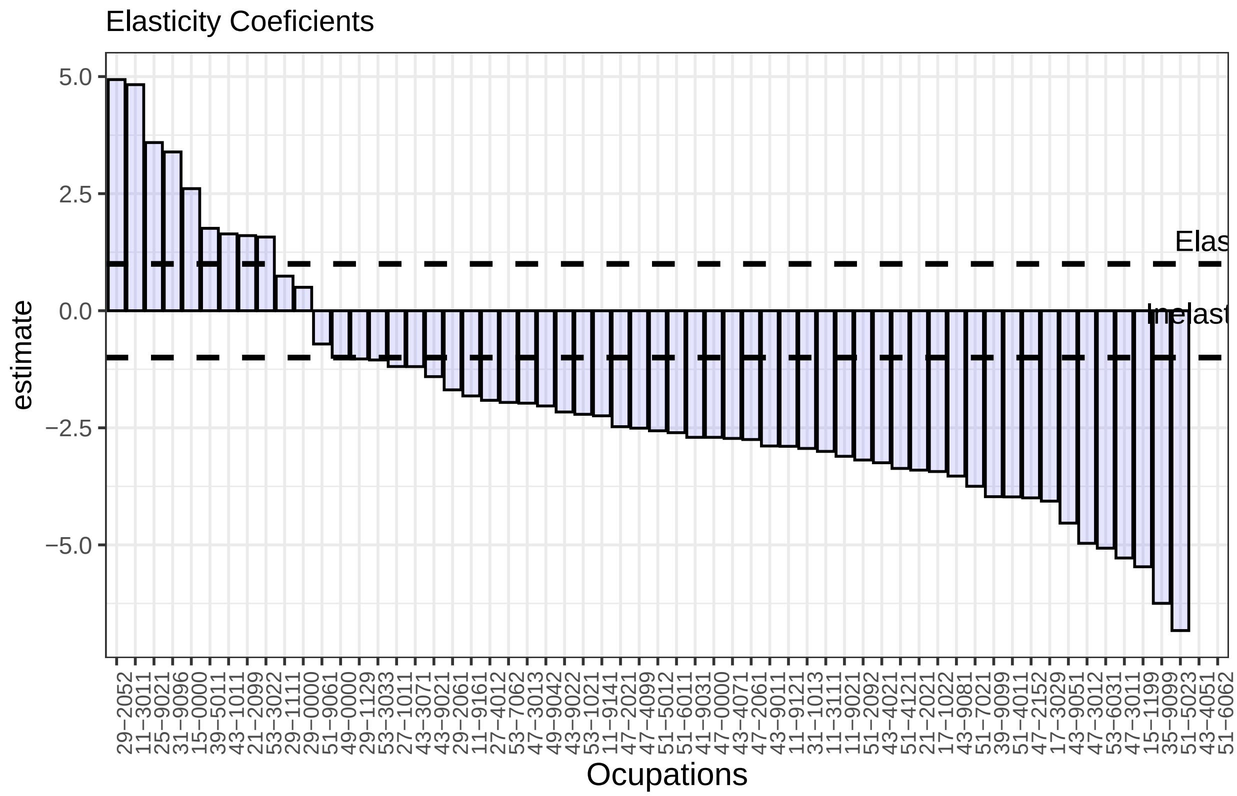
```
El<- Elas%>%
  select(-resid,-modelo)


Elasticidades%>%
  filter(bp_p>0.05,shapiro_p>0.05)%>%
  arrange(desc(estimate))%>%
  select(-model.x,-resid)%>%
  head(10)%>%
  kable(caption = "Elasticity Estimate Verified (Showing 10)", digits = 2) %>%
  kable_styling(full_width = FALSE, bootstrap_options = c("striped", "hover", "condensed"))
```

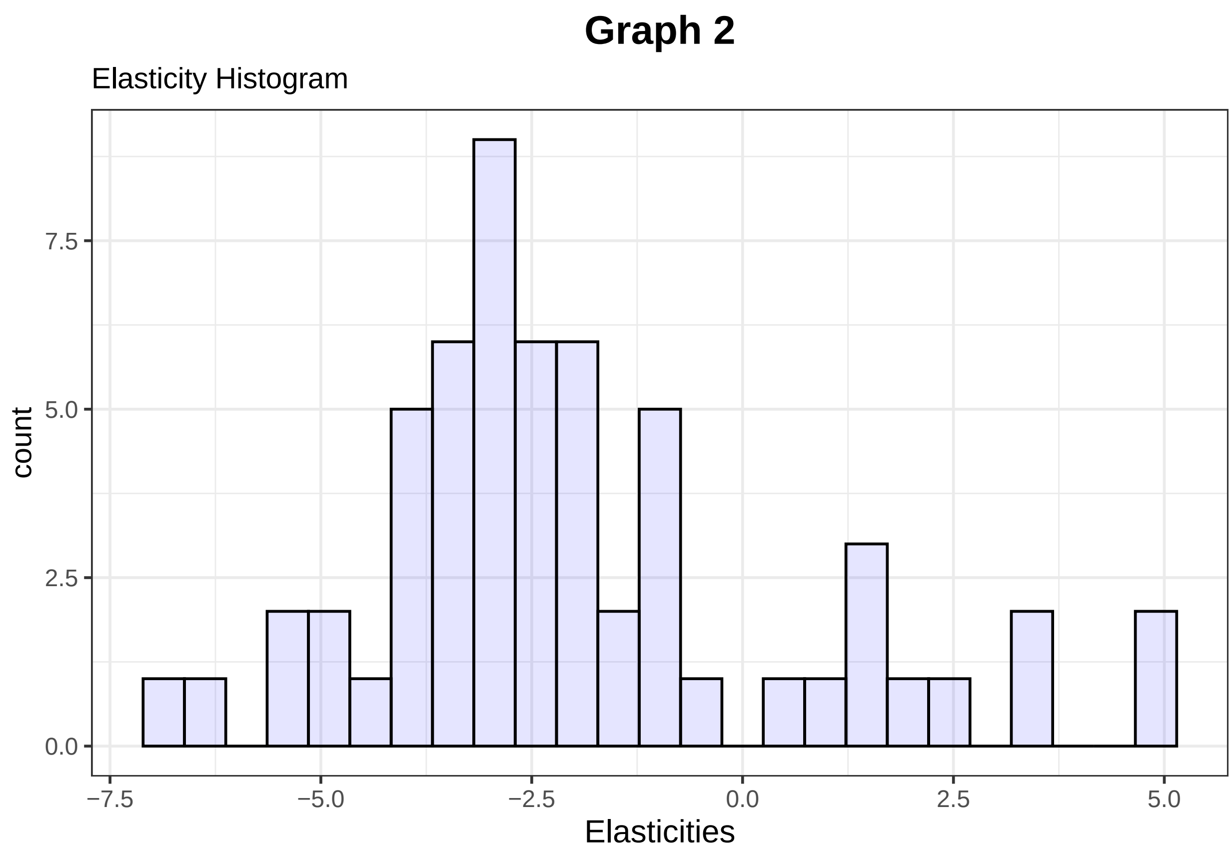
Table 2: Elasticity Estimate Verified (Showing 10)

OCC_CODE	OCC_TITLE	estimate	Elasticidad
29-2052	Pharmacy Technicians	4.94	Elastica
11-3011	Administrative Services Managers	4.83	Elastica
25-9021	Farm and Home Management Advisors	3.59	Elastica
31-9096	Veterinary Assistants and Laboratory Animal Caretakers	3.39	Elastica
15-0000	Computer and Mathematical Occupations	2.61	Elastica
39-5011	Barbers	1.76	Elastica
43-1011	First-Line Supervisors/Managers of Office and Administrative Support Workers	1.64	Elastica
21-2099	Religious workers, all other	1.60	Elastica
53-3022	Bus Drivers, School	1.57	Elastica
29-1111	Registered Nurses	0.74	Inelastica

We now have the Elasticities that confine with the linear regression model assupmtions.

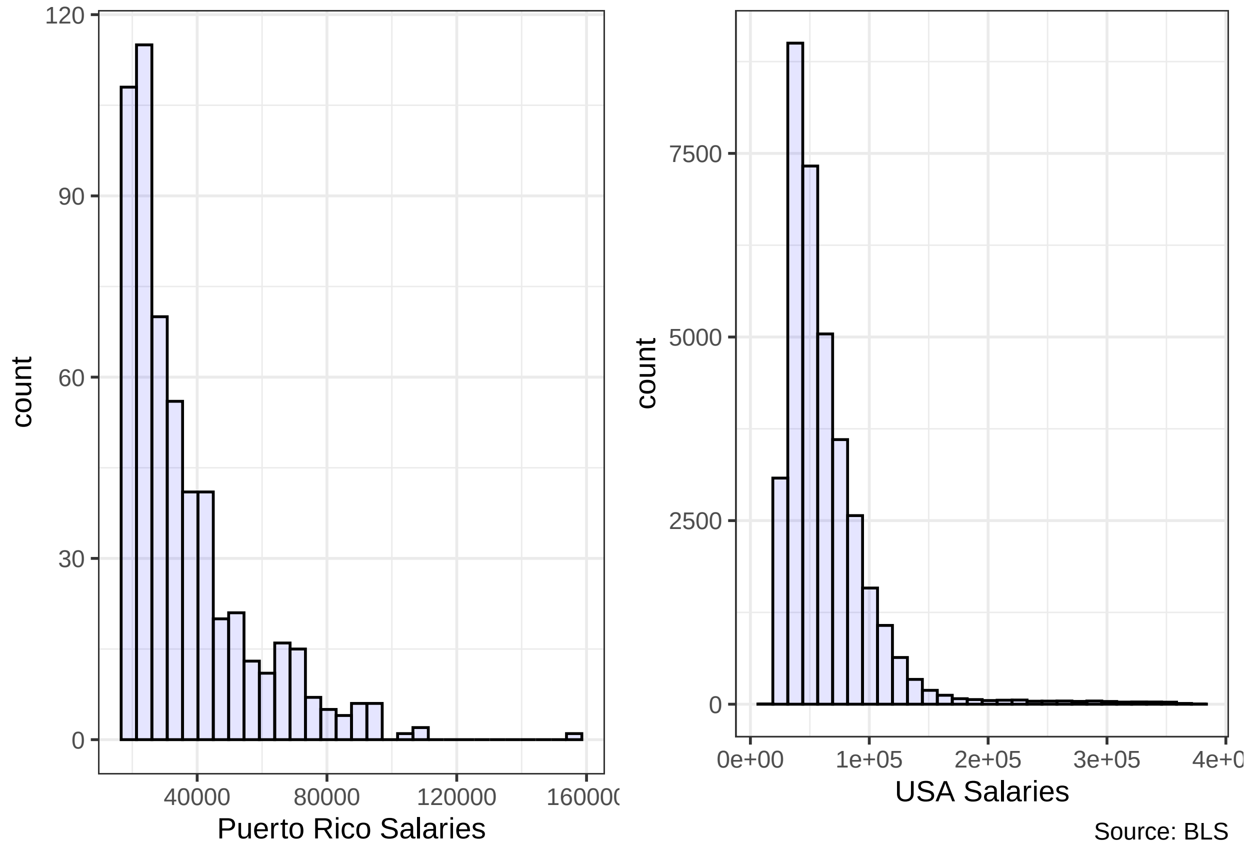
Graph 1



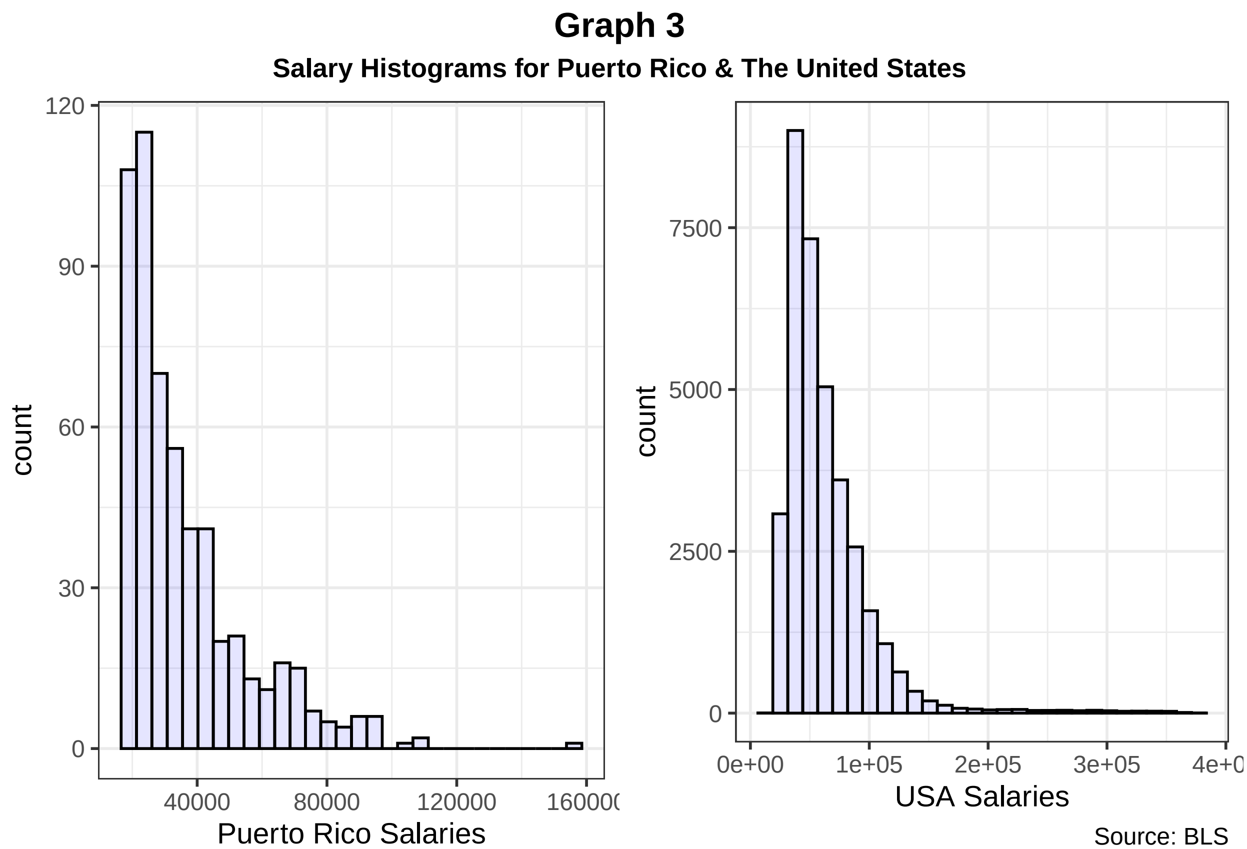



```r
h1<-ggplot(Salarios_PR_20_21, aes(A_MEAN))+
  geom_histogram(color="black", fill = "blue", alpha = 0.1)+
  labs(x = "Puerto Rico Salaries")+
  theme_bw()
h2<-ggplot(Salarios_USA_2021, aes(A_MEAN))+
  geom_histogram(color="black", fill = "blue", alpha = 0.1)+
  labs(x = "USA Salaries",caption = "Source: BLS")+
  theme_bw()

tg <- textGrob('Graph 3', gp = gpar(fontsize = 13, fontface = 'bold'))
sg <- textGrob("Salary Histograms for Puerto Rico & The United States",
             gp = gpar(fontsize = 10, fontface = 'bold'))



grided <-grid.arrange(h1,h2, nrow = 1, ncol = 2)
```

```
## 'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.
## 'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.
```



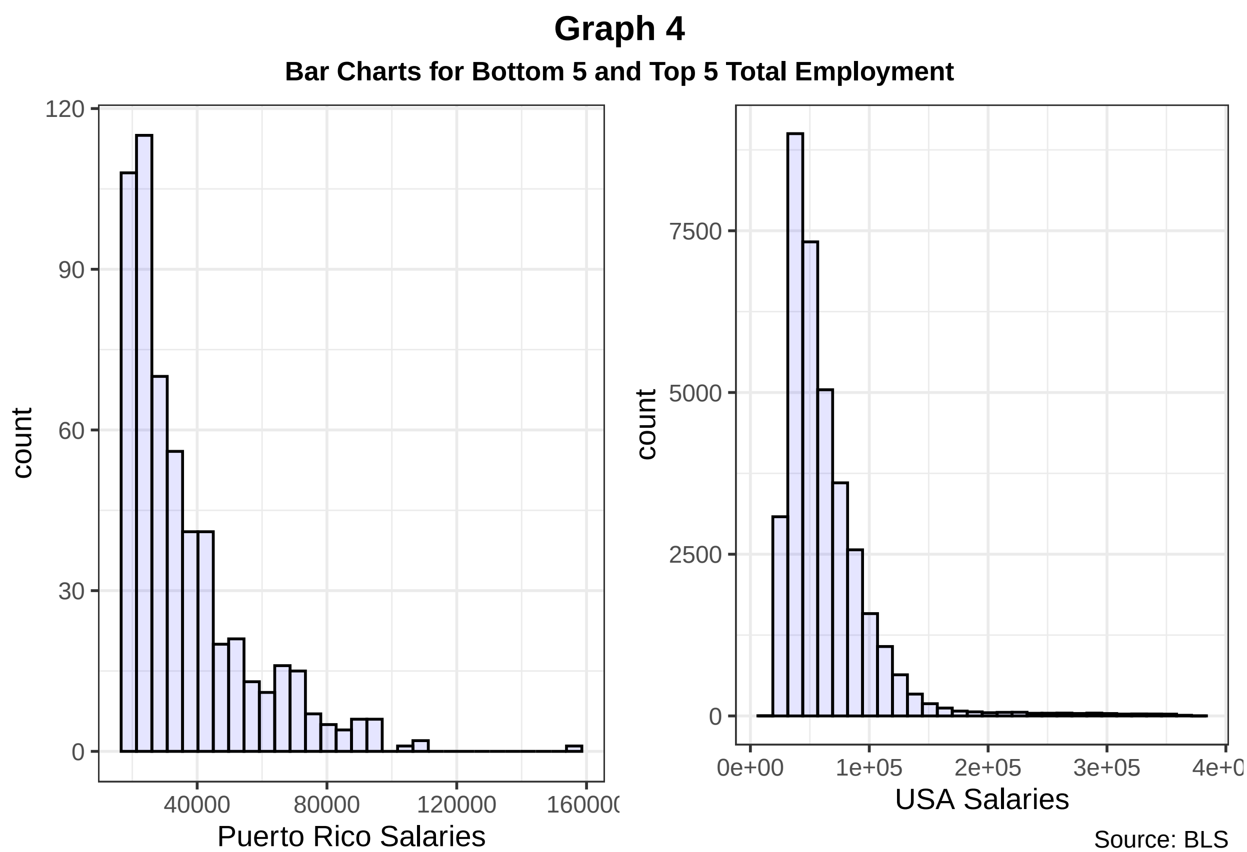

```
margin <- unit(0.5, "line")

grid.arrange(tg, sg, grided,
             heights = unit.c(grobHeight(tg) + 1.2*margin,
                              grobHeight(sg) + margin,
                              unit(1,"null")))
```



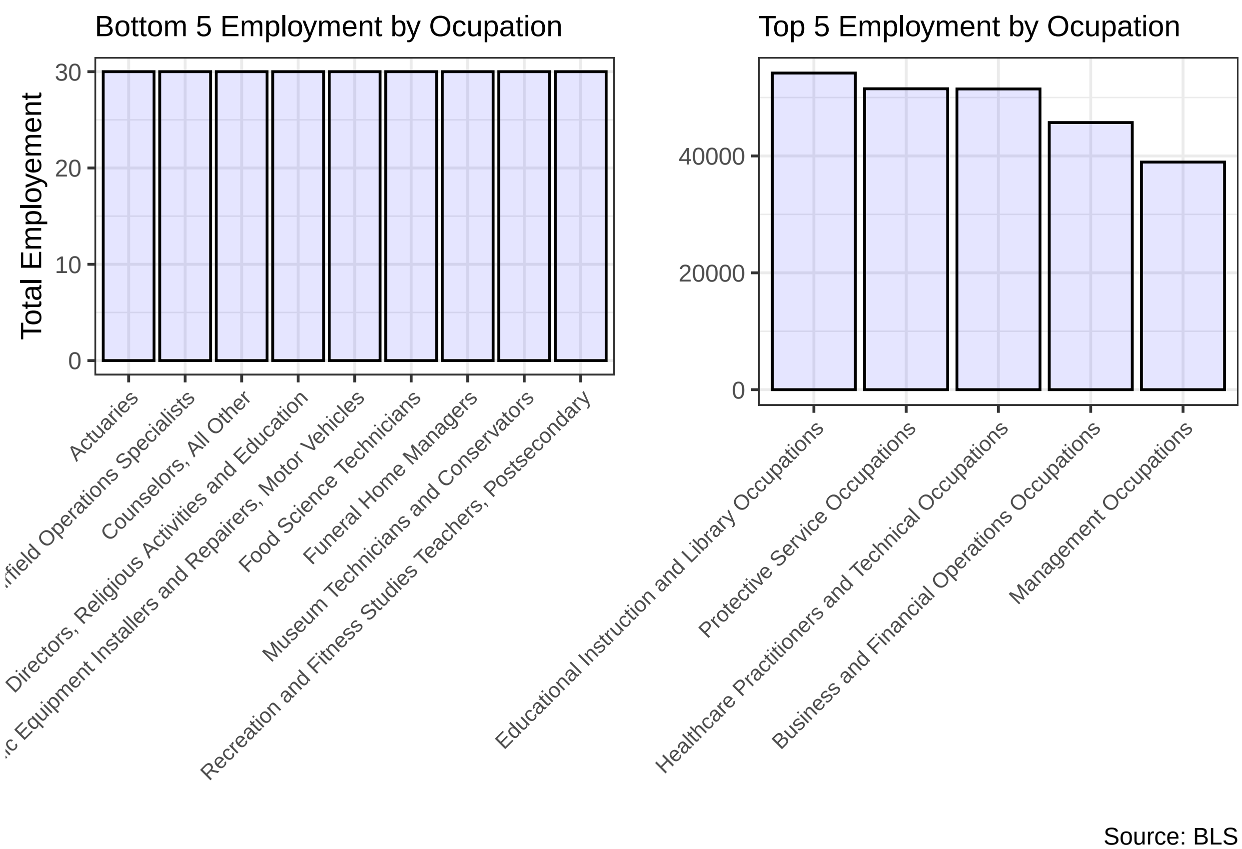
Int ths part we will see some date related to employment of Puerto Rico and the United States.

```
margin <- unit(0.5, "line")


low5 <- S_20_21_pp %>%
  slice_min(order_by = TOT_EMP, n = 5)

top5 = S_20_21_e %>%
  slice_min(order_by = TOT_EMP, n = 5)

empleo_min<-ggplot(low5, aes(x = reorder(OCC_TITLE,-TOT_EMP), y = TOT_EMP))+geom_col(color="black", fil
  theme(plot.title = element_text(size=10),
        axis.text.x = element_text(size = 8, angle = 65, hjust = 1))+
  labs(subtitle  = "Bottom 5 Employment by Ocupation",x = "", y = 'Total Employement')  +
  theme(plot.title = element_text(size=10 ,face = "bold", hjust = 0.5),
        axis.text.x = element_text(size = 8, angle = 45, hjust = 1))


empleo_max <- ggplot(top5, aes(x = reorder(OCC_TITLE,-TOT_EMP), y = TOT_EMP))+geom_col(color="black", f
    labs(subtitle  = "Top 5 Employment by Ocupation",x = "",y = '', caption = "Source: BLS")+
  theme(plot.title = element_text(size=10 ,face = "bold", hjust = 0.5),
        axis.text.x = element_text(size = 8, angle = 45, hjust = 1))

#empleo_min
#empleo_max
tg <- textGrob('Graph 4', gp = gpar(fontsize = 13, fontface = 'bold'))
```

```r
sg <- textGrob("Bar Charts for Bottom 5 and Top 5 Total Employment",
               gp = gpar(fontsize = 10, fontface = 'bold'))
margin <- unit(0.5, "line")

grid.arrange(tg, sg, grided,
             heights = unit.c(grobHeight(tg) + 1.2*margin,
                              grobHeight(sg) + margin,
                              unit(1,"null")))
```

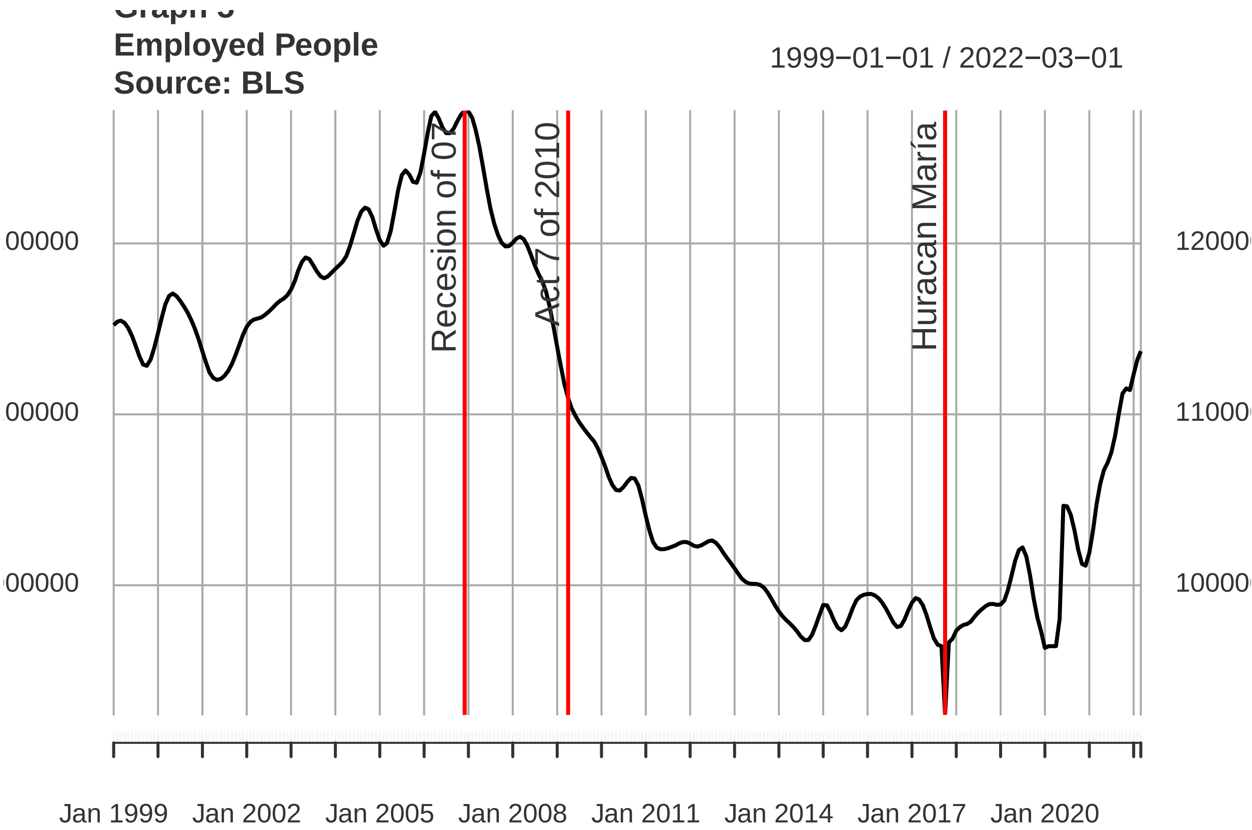



```r
grided <-grid.arrange(empleo_min,empleo_max, nrow = 1, ncol = 2)
```




```r
capture.output({
  library(readxl)
  library(dplyr)
  library(xts)

  # Load and prep data
  Empleo <- read_excel("C:/Users/yadel/OneDrive/Documents/1999-2021/Empleo.xlsx")

  Empleo_ <- Empleo %>%
    select(Year, Period, employment) %>%
    summarise(Empleo = as.numeric(employment))

  dates <- seq(as.Date("1999-01-01"), length = 279, by = "month")
  empleos <- as.matrix(Empleo_)
  empleo_xts <- xts(empleos, order.by = dates)

  # Define events
  events <- xts(
    c("Recesion of 07", "Huracan María", "Act 7 of 2010"),
    as.Date(c("2006-12-01", "2017-09-01", "2009-03-09"))
  )

  # Plot all at once
  par(mfrow = c(1, 1))
  plot(empleo_xts, main = "Graph 5\nEmployed People\nSource: BLS")
  addEventLines(events, pos = 2, offset = 0.5, cex = 1.2, col = "red", lwd = 2, srt = 90)
```

```
})
```


Graph 5
Employed People
Source: BLS
1999−01−01 / 2022−03−01
Recesion of 07
Act 7 of 2010
Huracan María
00000
00000
00000
12000
11000
10000
Jan 1999
Jan 2002
Jan 2005
Jan 2008
Jan 2011
Jan 2014
Jan 2017
Jan 2020

```
## character(0)
```