# ECON2250_Group11_FinalProject

Wendell, Connor, & Caroline

2025-11-19

```
# Install Necessary Packages
options(repos = c(CRAN = "https://cran.rstudio.com/"))

packages <- c("tidyverse", "tidycensus", "bea.R", "janitor")
install.packages(setdiff(packages, rownames(installed.packages())))
lapply(packages, library, character.only = TRUE)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.2
## v ggplot2   4.0.0     v tibble    3.3.0
## v lubridate 1.9.4     v tidyr     1.3.1
## v purrr     1.1.0
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
## Loading required package: data.table
##
##
## Attaching package: 'data.table'
##
##
## The following objects are masked from 'package:lubridate':
##
##     hour, isoweek, mday, minute, month, quarter, second, wday, week,
##     yday, year
##
##
## The following objects are masked from 'package:dplyr':
##
##     between, first, last
##
##
## The following object is masked from 'package:purrr':
##
##     transpose
##
##
## Note: As of February 2018, beaGet() requires 'TableName' for NIPA and NIUnderlyingDetail data instead
##
##
```

```
## Attaching package: 'janitor'
##
##
## The following objects are masked from 'package:stats':
##
##     chisq.test, fisher.test
```

```
## [[1]]
##  [1] "lubridate" "forcats"   "stringr"   "dplyr"     "purrr"     "readr"
##  [7] "tidyr"     "tibble"    "ggplot2"   "tidyverse" "stats"     "graphics"
## [13] "grDevices" "utils"     "datasets"  "methods"   "base"
##
## [[2]]
##  [1] "tidycensus" "lubridate"  "forcats"    "stringr"    "dplyr"
##  [6] "purrr"      "readr"      "tidyr"      "tibble"     "ggplot2"
## [11] "tidyverse"  "stats"      "graphics"   "grDevices"  "utils"
## [16] "datasets"   "methods"    "base"
##
## [[3]]
##  [1] "bea.R"      "data.table" "tidycensus" "lubridate"  "forcats"
##  [6] "stringr"    "dplyr"      "purrr"      "readr"      "tidyr"
## [11] "tibble"     "ggplot2"    "tidyverse"  "stats"      "graphics"
## [16] "grDevices"  "utils"      "datasets"   "methods"    "base"
##
## [[4]]
##  [1] "janitor"    "bea.R"      "data.table" "tidycensus" "lubridate"
##  [6] "forcats"    "stringr"    "dplyr"      "purrr"      "readr"
## [11] "tidyr"      "tibble"     "ggplot2"    "tidyverse"  "stats"
## [16] "graphics"   "grDevices"  "utils"      "datasets"   "methods"
## [21] "base"
```

```
install.packages("tinytex")
```

```
##
## The downloaded binary packages are in
##  /var/folders/wb/3yb0r8bs6179459cgnh43p640000gn/T//RtmprkQ8mK/downloaded_packages
```

```
library(tinytex)
```

```
## Warning: package 'tinytex' was built under R version 4.5.2
```

```
# Enable Census Data
census_api_key("fa1f22a59be49b0aa3a0a638fc8ec6ab84af62be", install = TRUE, overwrite = TRUE)
```

```
## Your original .Renviron will be backed up and stored in your R HOME directory if needed.
## Your API key has been stored in your .Renviron and can be accessed by Sys.getenv("CENSUS_API_KEY").
## To use now, restart R or run `readRenviron("~/.Renviron")`
```

```
## [1] "fa1f22a59be49b0aa3a0a638fc8ec6ab84af62be"
```

```r
# Enable BEA Data
bea_Key <- "9EFC7A8B-2A67-486D-9F8B-6AA3A26DBCF8"

beaGet(list(
  'UserID' = bea_Key,
  'Method' = 'GetParameterValues',
  'datasetname' = 'Regional'
))
```

```
## No encoding supplied: defaulting to UTF-8.
```

```
## Warning in bea.R::bea2List(beaPayload): When requesting a list of parameter
## values, the Parameter name must be included in the request - no name.
```

```
## [1] "When requesting a list of parameter values, the Parameter name must be included in the request -
```

```r
# Stops BEA Related Errors
safe_bea_data <- function(bea_object) {
  if (!("Data" %in% names(bea_object))) {
    print("X: BEA returned an error instead of data:")
    print(bea_object)
    stop("Fix the BEA request above - no $Data field found.")
  }
  return(bea_object$Data)
}

# Pull economic data
income_data <- get_acs(
  geography = "state",
  variables = "B19013_001",
  year = 2020,
  survey = "acs5"
) %>%
  select(state = NAME, median_income = estimate)
```

```
## Getting data from the 2016-2020 5-year ACS
```

```r
# Education Variables
education_var <- c("B15003_022", "B15003_023", "B15003_024", "B15003_025")

# Pull education data
education_raw <- get_acs(
  geography = "state",
  variables = education_var,
  year = 2020,
  survey = "acs5"
)
```

```
## Getting data from the 2016-2020 5-year ACS
```

```r
education_final <- education_raw %>%
  group_by(NAME) %>%
  summarise(
    edu_bachelors_plus = sum(estimate)
  ) %>%
  rename(state = NAME)

# Race Variables
race_var <- c(
  black = "DP05_0038PE",
  hispanic = "DP05_0071PE",
  asian = "DP05_0066PE",
  white = "DP05_0037PE"
)

# Pull Race Data
race_data <- get_acs(
  geography = "state",
  variables = race_var,
  year = 2020,
  survey = "acs5"
) %>%
  select(state = NAME, variable, estimate) %>%
  spread(variable, estimate)
```

```
## Getting data from the 2016-2020 5-year ACS
## Using the ACS Data Profile
```

```r
# Pull Age Data
age_data <- get_acs(
  geography = "state",
  variables = "DP05_0017E",
  year = 2020,
  survey = "acs5"
) %>%
  select(state = NAME, median_age = estimate)
```

```
## Getting data from the 2016-2020 5-year ACS
## Using the ACS Data Profile
```

```r
# Pull Income Per Capita
income_per_capita <- get_acs(
  geography = "state",
  variables = "B19301_001",
  year = 2020,
  survey = "acs5"
) %>%
  select(state = NAME, bea_personal_income = estimate)
```

```
## Getting data from the 2016-2020 5-year ACS
```

```r
# Pull State GDP
state_gdp_final <- get_acs(
  geography = "state",
  variables = "B01003_001",
  year = 2020,
  survey = "acs5"
) %>%
  select(state = NAME, gdp_state = estimate)
```

```
## Getting data from the 2016-2020 5-year ACS
```

```r
# Get Data Together
state_data <- income_data %>%
  left_join(education_final, by = "state") %>%
  left_join(race_data, by = "state") %>%
  left_join(age_data, by = "state") %>%
  left_join(income_per_capita, by = "state") %>%
  left_join(state_gdp_final, by = "state") %>%
  clean_names()

# Develop Regression Model
regression_model <- lm(
  median_income ~ edu_bachelors_plus + dp05_0038p + dp05_0071p + dp05_0066p + dp05_0037p + median_age +
  data = state_data
)

summary(regression_model)
```

```
##
## Call:
## lm(formula = median_income ~ edu_bachelors_plus + dp05_0038p +
##     dp05_0071p + dp05_0066p + dp05_0037p + median_age + bea_personal_income +
##     gdp_state, data = state_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6097.1 -1840.2  -278.9  1084.0 13440.8
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          3.639e+04  7.339e+03   4.958 1.16e-05 ***
## edu_bachelors_plus   4.711e-03  3.098e-03   1.521  0.13567
## dp05_0038p          -4.117e+02  7.242e+01  -5.684 1.05e-06 ***
## dp05_0071p          -1.292e+02  4.220e+01  -3.061  0.00379 **
## dp05_0066p          -1.566e+02  1.602e+02  -0.977  0.33393
## dp05_0037p          -2.585e+02  5.690e+01  -4.543 4.44e-05 ***
## median_age          -4.247e-02  1.492e-02  -2.847  0.00674 **
## bea_personal_income  1.545e+00  1.192e-01  12.967  < 2e-16 ***
## gdp_state           -1.324e-04  6.302e-04  -0.210  0.83461
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3641 on 43 degrees of freedom
```

```
## Multiple R-squared:  0.9288, Adjusted R-squared:  0.9155
## F-statistic: 70.08 on 8 and 43 DF,  p-value: < 2.2e-16
```