STA 445 Chapter 13

Dr. Robert Buscaglia

October 25, 2023

Exercise 3

We will fully clean the government data and create some interesting analysis results.

Load the data. Viewing the original excel file is a good idea!

```
Budget <- readxl::read_xlsx('US_Gov_Budget_1962_2020.xlsx', skip=2)
```

We will want to create columns for the functions and subfunctions within the document. Let us rename these Department for now.

```
Budget.2 <- Budget %>% rename(`Department` = `Function and Subfunction`)
```

We should remove any totals or budget information. There are also two rows that inform the user of what NAs indicate.

```
Budget.3 <- Budget.2 %>%
  filter(!str_detect(Department,
                     'Total|Subtotal|[oO]n-budget|[oO]ff-budget|Not available'))
```

Create an ID column for the functions and subfunctions by extracting it from the department name. I then relocate the rows to keep the ID at the beginning, and then remove the ID number from the Department column.

```
Budget.4 <- Budget.3 %>%
  mutate(ID_number = str_extract(Department, '\\d+')) %>%
  relocate(ID_number) %>%
  mutate(Department = str_remove_all(Department, '\\d+')) %>%
  fill(ID_number)
```

Functions correspond to rows which have no numerical entries. We want to split the Department column into Functions and Subfunctions by recognizing this element. We can use an `ifelse` statement to allow us to identify which Department headings are functions, and create two corresponding new columns to indicate the function and subfunction groupings. We should finish cleaning by ensuring we no longer keep the Department column when finished.

```
Budget.5 <- Budget.4 %>%
  mutate(Function = ifelse(is.na(`2015`), Department, NA)) %>%
  relocate(Function) %>%
  fill(Function) %>%
  mutate(Subfunction = ifelse(!is.na(`2015`), Department, NA)) %>%
```

```
  relocate(Subfunction) %>%
  filter(!is.na(`2015`)) %>%
  relocate(ID_number, Function) %>%
  select(-Department) %>%
  mutate(Function = str_replace_all(Function, '\\:', ''))
```

We are ready to start calculating some summary values. We can pivot the data to clean a bit more. Let us create a long version of this data set that has only four columns: Function, Subfunction, Year, Amount.

```
Budget.long <- Budget.5 %>%
  pivot_longer(names_to = 'Year', values_to = 'Amount', `2001`:`2015`) %>%
  select(Function, Subfunction, Year, Amount)
```

There are some entries for the Amounts that are non-numeric. Clean these observations out using a filter.

```
Budget.long.2 <- Budget.long %>%
  mutate(Amount = as.numeric(Amount)) %>%
  filter(!is.na(Amount))
```

```
## Warning in mask$eval_all_mutate(quo): NAs introduced by coercion
```

```
Budget.long %>% filter(!str_detect(Amount, '\\.'))
```

Before making any calculations, let us change the variable type of Year and Amount to be numeric rather than strings.

```
Budget.long.3 <- Budget.long.2 %>%
  mutate(Year = as.numeric(Year), Amount = as.numeric(Amount))
```

Now consider information you may like to know. How about the mean Amount of funding for each subgroup over the duration available.

```
Budget.long.3 %>%
  group_by(Subfunction) %>%
  summarize(Mean.Budget = mean(Amount)) %>%
  arrange(desc(Mean.Budget)) %>% slice(1:10) %>%
  kable(align='lc', caption='Top 10 Mean Budgets for Subfunctions.')
```

Table 1: Top 10 Mean Budgets for Subfunctions.

Subfunction	Mean.Budget
Social security	638655.87
Interest on Treasury debt securities (gross)	388598.93
Medicare	383772.93
Health care services	270862.40

Subfunction	Mean.Budget
Operation and Maintenance	218720.87
Other income security	143081.53
Military Personnel	130963.47
Federal employee retirement and disability	108874.40
Procurement	99334.27
Food and nutrition assistance	72155.47

How about the mean Amount of funding for each Function over the duration available.

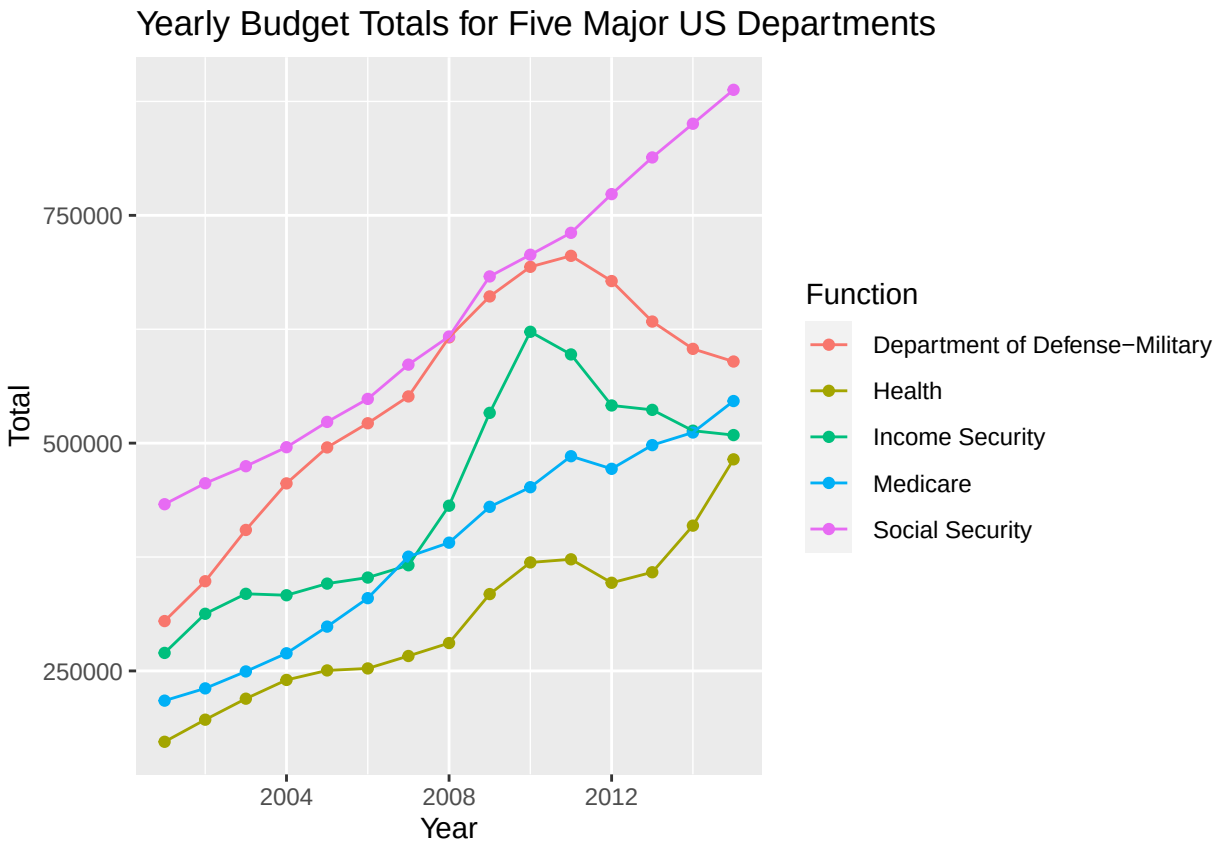
```
Budget.long.3 %>%
  group_by(Function) %>%
  summarize(Mean.Budget = mean(Amount)) %>%
  arrange(desc(Mean.Budget)) %>% slice(1:10) %>%
  kable(align='lc', caption='Top 10 Mean Budgets for Functions')
```

Table 2: Top 10 Mean Budgets for Functions

Function	Mean.Budget
Social Security	638655.87
Medicare	383772.93
Net Interest	125993.20
Health	101139.93
Income Security	73321.36
Department of Defense-Military	61204.01
Transportation	19533.75
Veterans Benefits and Services	18852.32
Education, Training, Employment, and Social Services	15367.64
General Science, Space, and Technology	12888.03

Let us extract the top five Functions and create a graph of their total budget. I chose to omit the `Net Interest`.

```
Budget.long.3 %>%
  filter(str_detect(`Function`, 'Defense|Health|Military|Security|Medicare')) %>%
  group_by(Function, Year) %>%
  summarize(Total = sum(Amount)) %>%
  ggplot(aes(x = Year, y = Total)) +
  geom_point(aes(color = Function)) +
  geom_line(aes(color=Function)) +
  labs(title = 'Yearly Budget Totals for Five Major US Departments')
```


Yearly Budget Totals for Five Major US Departments
Total
750000
500000
250000
2004
2008
2012
Year
Function
Department of Defense−Military
Health
Income Security
Medicare
Social Security