finalprojectEDA

DominiqueBarnes

2024-04-07

```r
library(tidyverse)
```

```
## -- 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   1.5.1
## v ggplot2   3.5.0     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- 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
```

```r
library(haven)
library(readxl)
library(MASS)
```

```
##
## Attaching package: 'MASS'
##
## The following object is masked from 'package:dplyr':
##
##     select
```

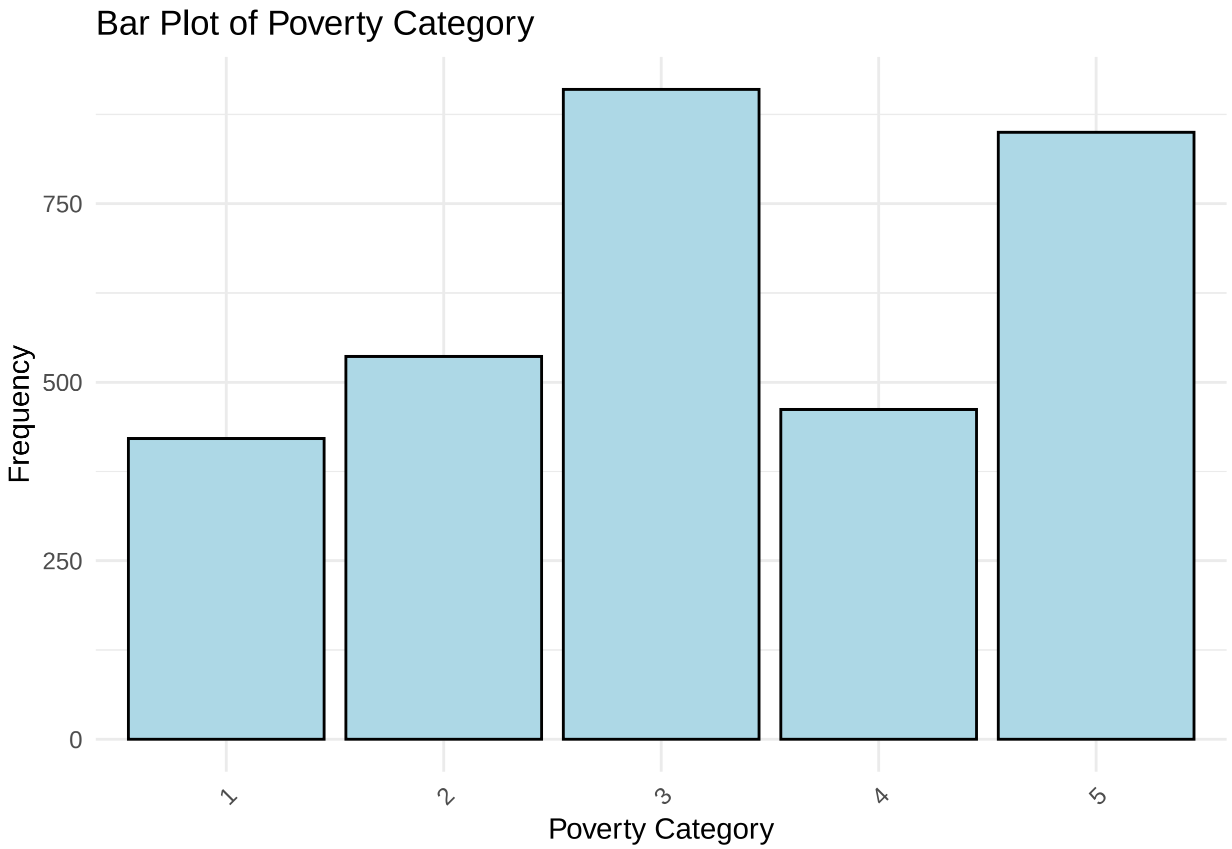
Load Data Wave 6

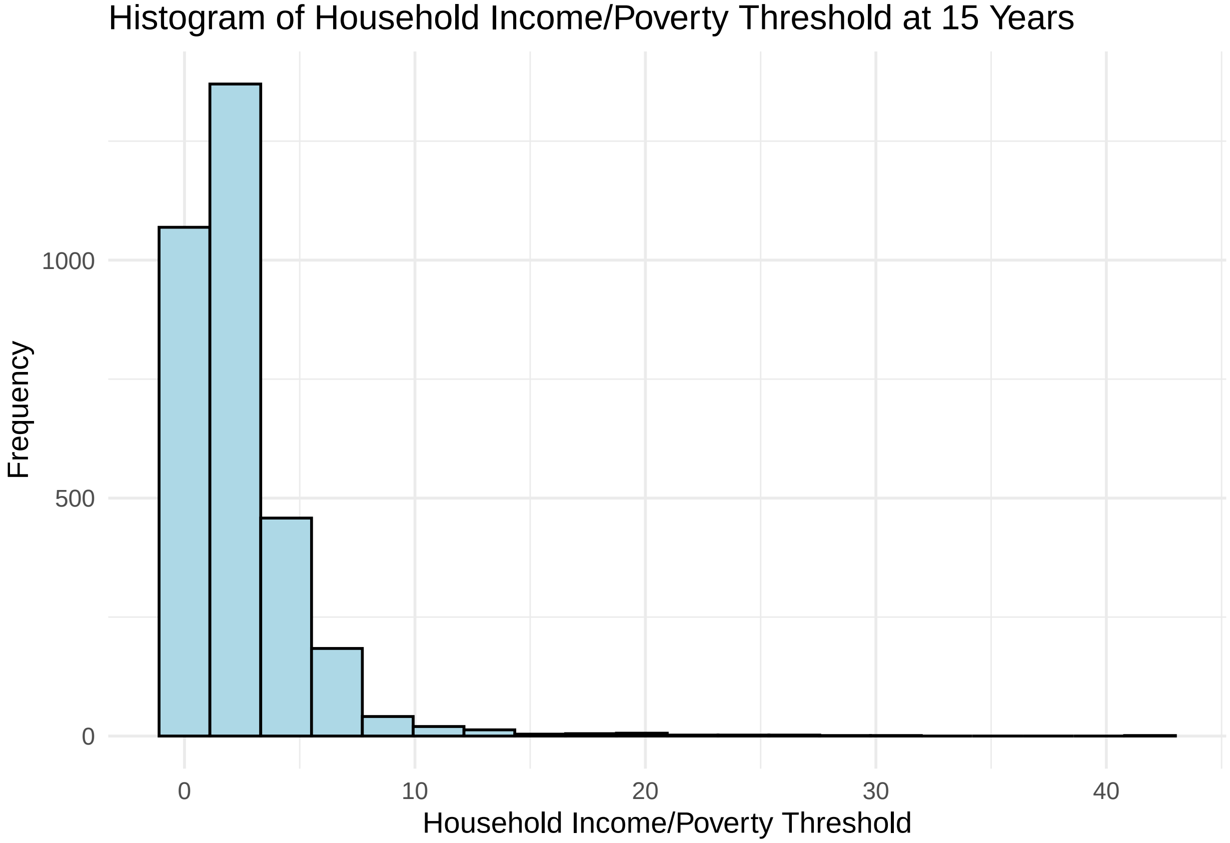
```r
wave6 <- as.data.frame(read_dta("/Users/dominiquebarnes/Desktop/SPR24_Coursework/DATA 2020/FFdata/wave6/
fin_var <- read.csv("/Users/dominiquebarnes/Desktop/SPR24_Coursework/DATA 2020/Final_Project/FinancialVa
fin_df <- as.data.frame(fin_var)
fin_var_code <- fin_df$Variable
df_selectCols <- wave6 %>% dplyr::select(all_of(fin_var_code))
```

```r
# Remove any rows that have values of -9(Not in Wave), -3 (Missing), -6(skip)
df_filtered <- df_selectCols %>%
  filter_all(all_vars(. !=-9 &. !=-3))
```

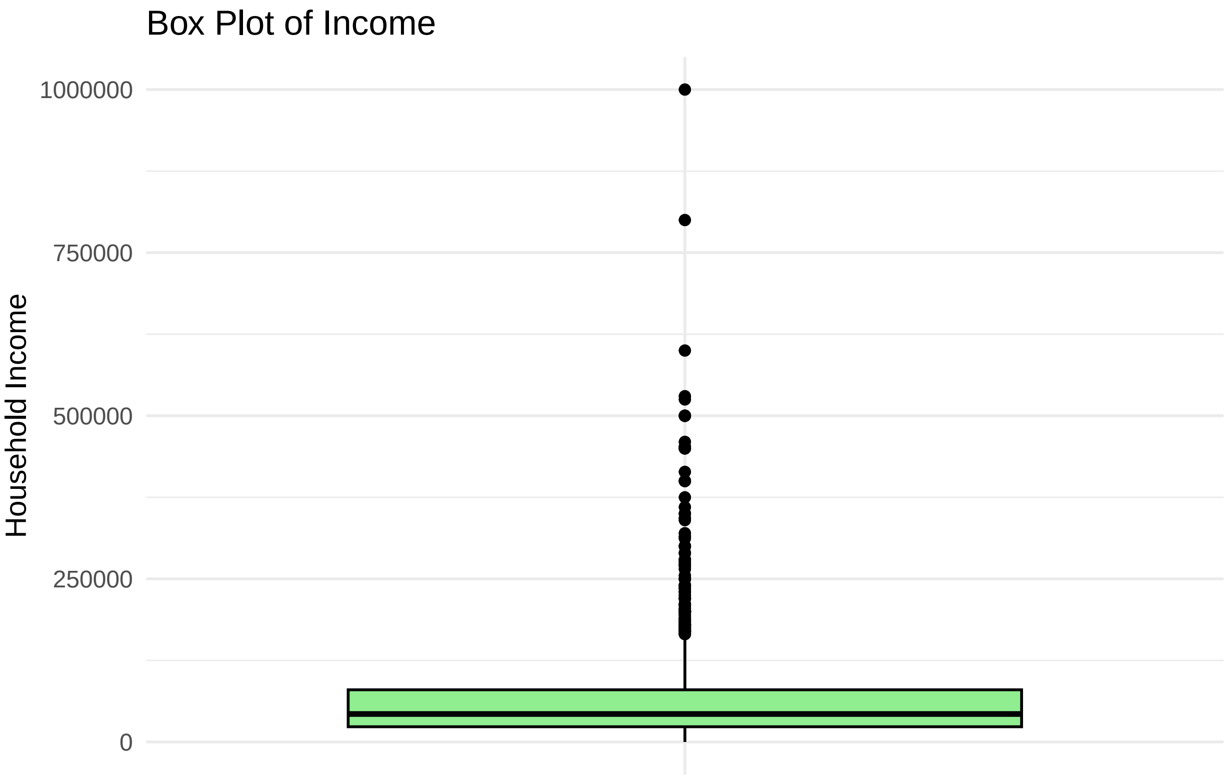
Including Plots

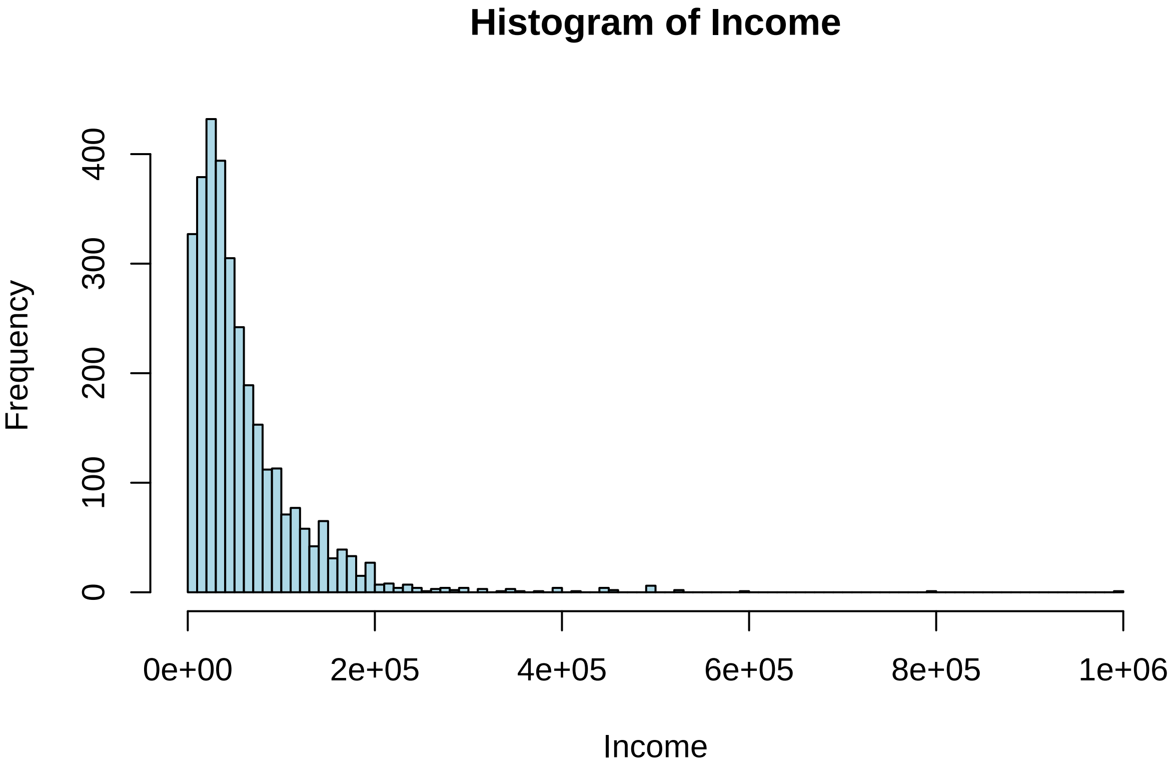
You can also embed plots, for example:


Bar Plot of Poverty Category

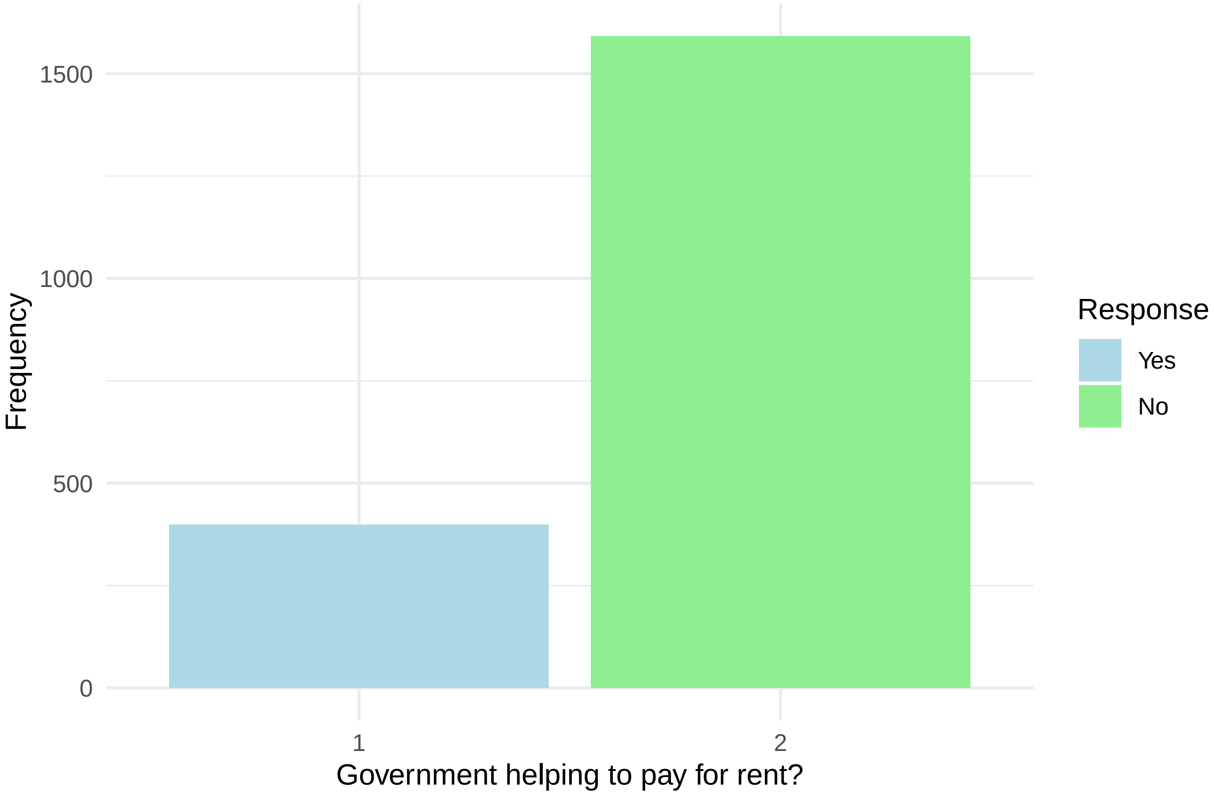

Histogram of Household Income/Poverty Threshold at 15 Years

## EDA

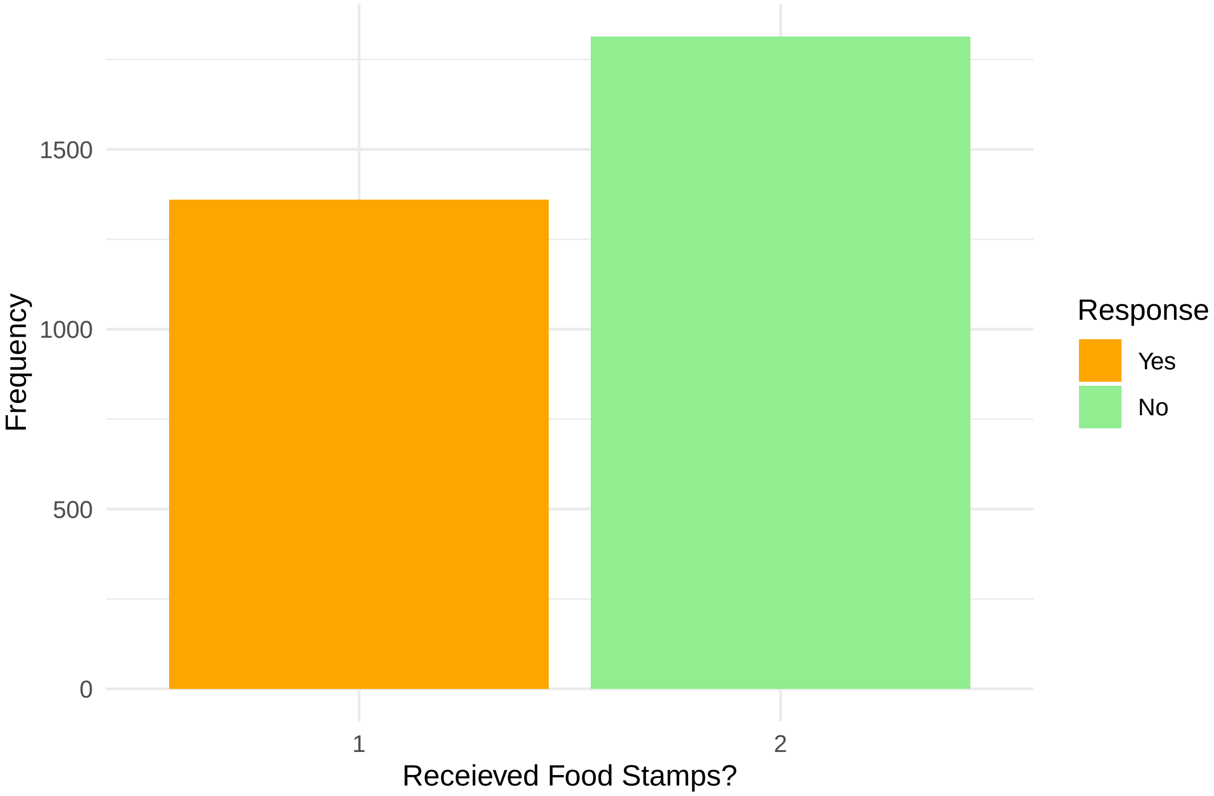

Box Plot of Income
Household Income
1000000
750000
500000
250000
0

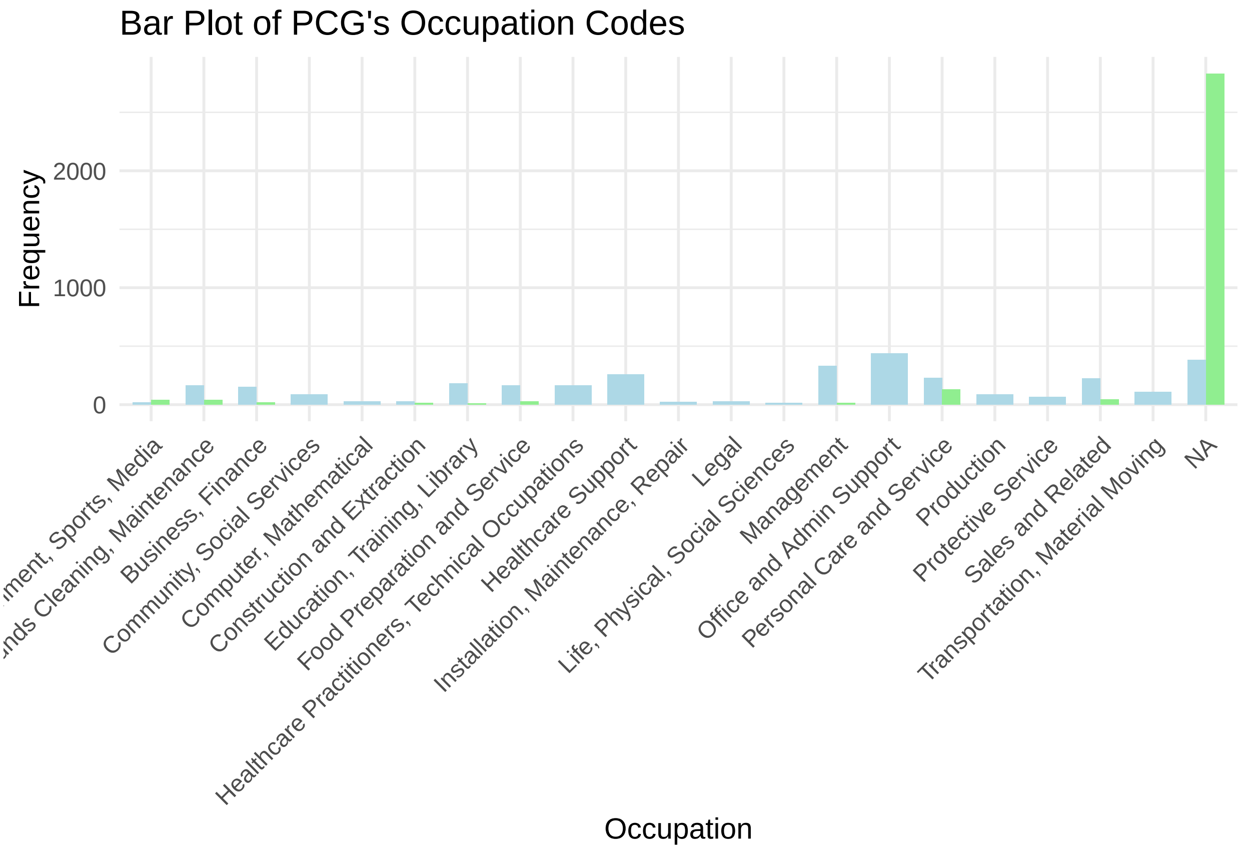

Histogram of Income
Frequency
0
100
200
300
400
0e+00
2e+05
4e+05
6e+05
8e+05
1e+06
Income

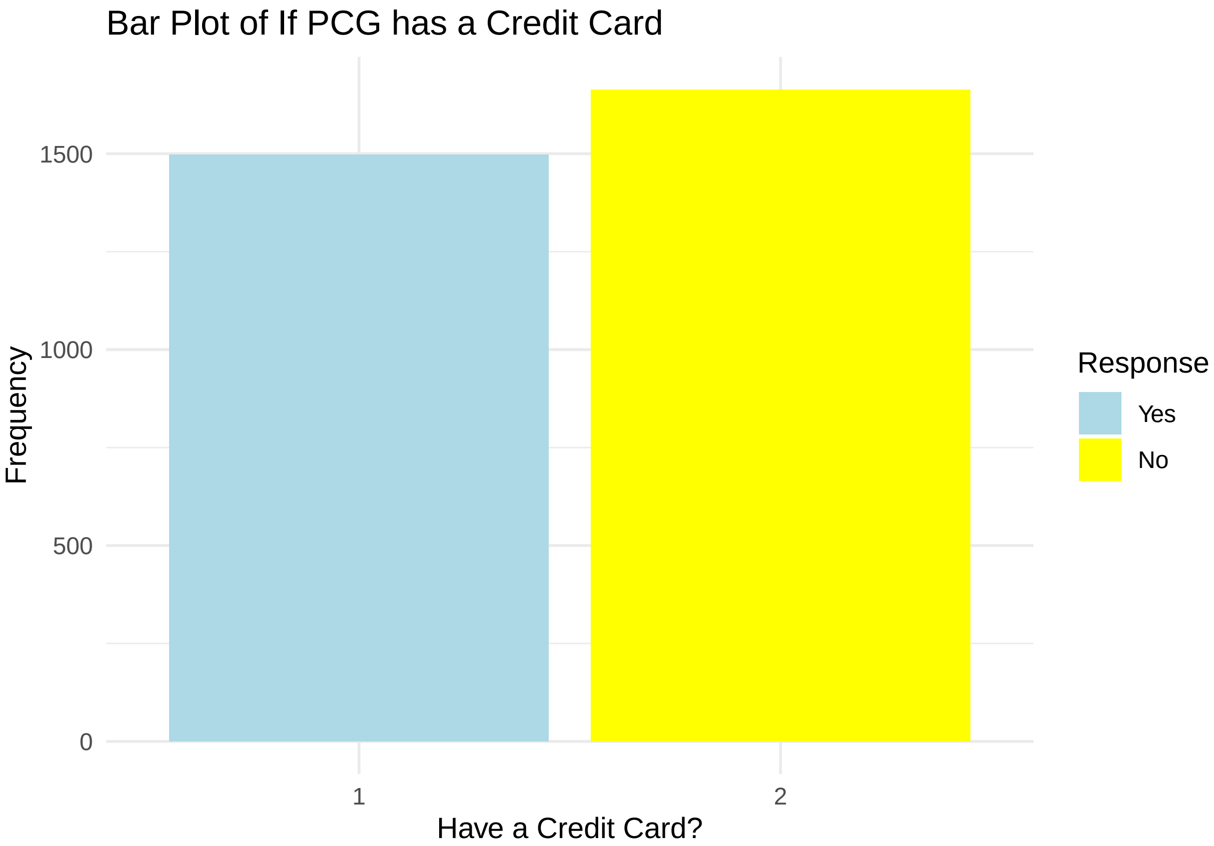
Bar Plot of Government Help for Rent

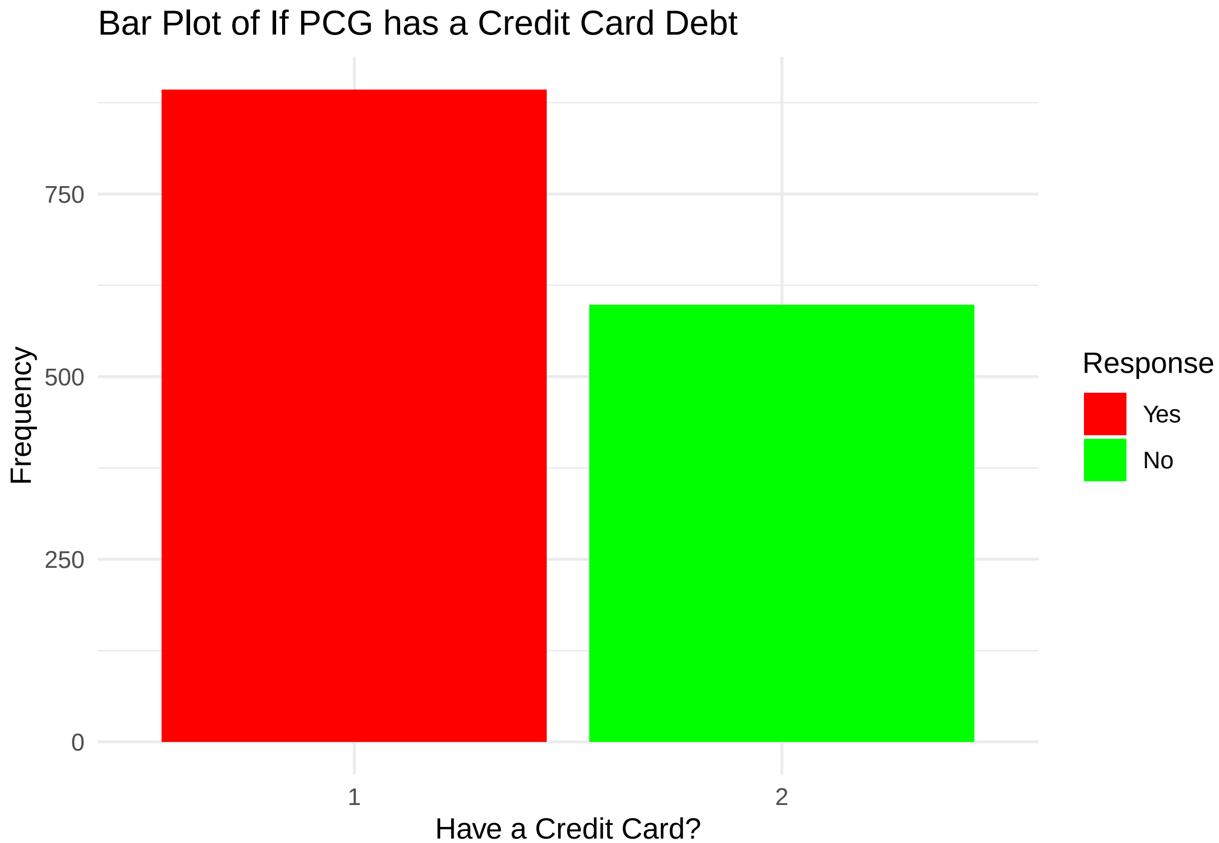


Bar Plot of Received Food Stamps

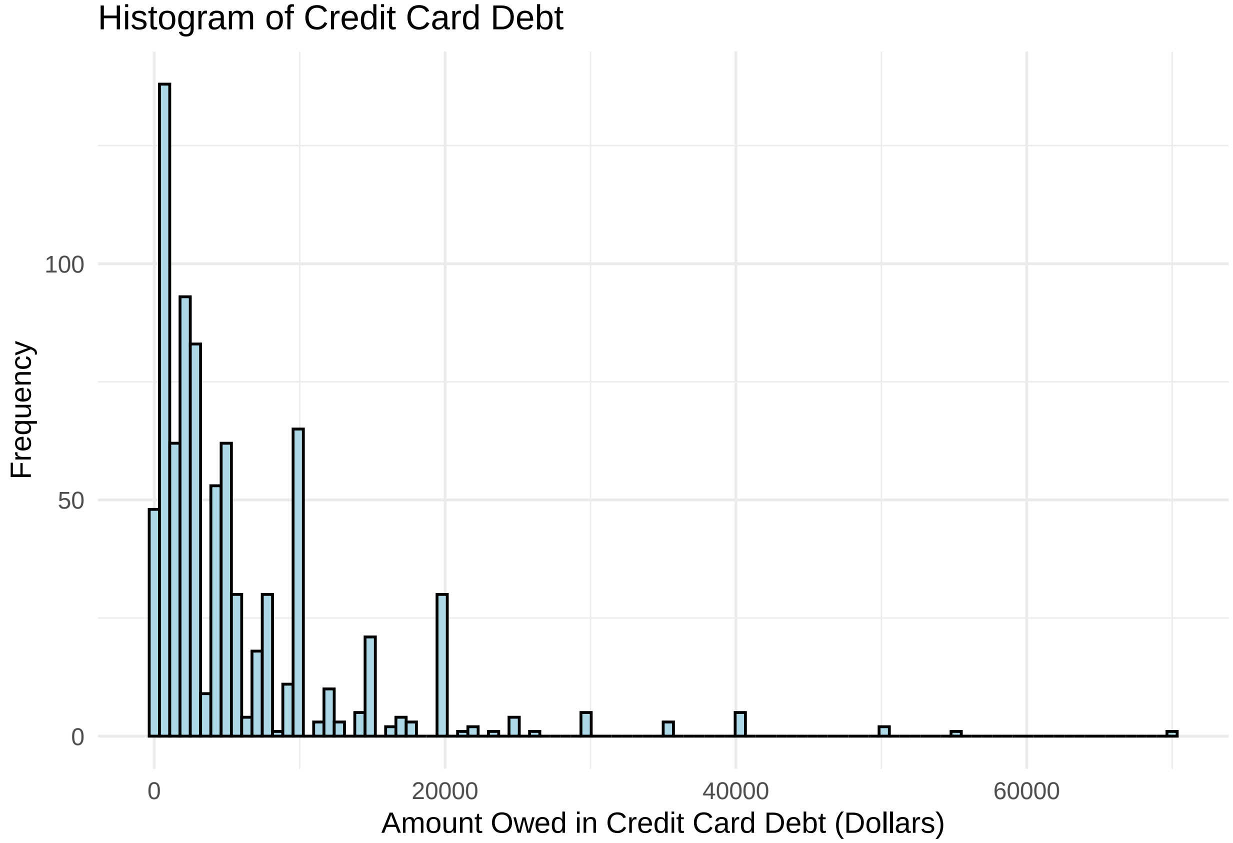



Bar Plot of PCG's Occupation Codes


Bar Plot of If PCG has a Credit Card



Bar Plot of If PCG has a Credit Card Debt



Histogram of Credit Card Debt

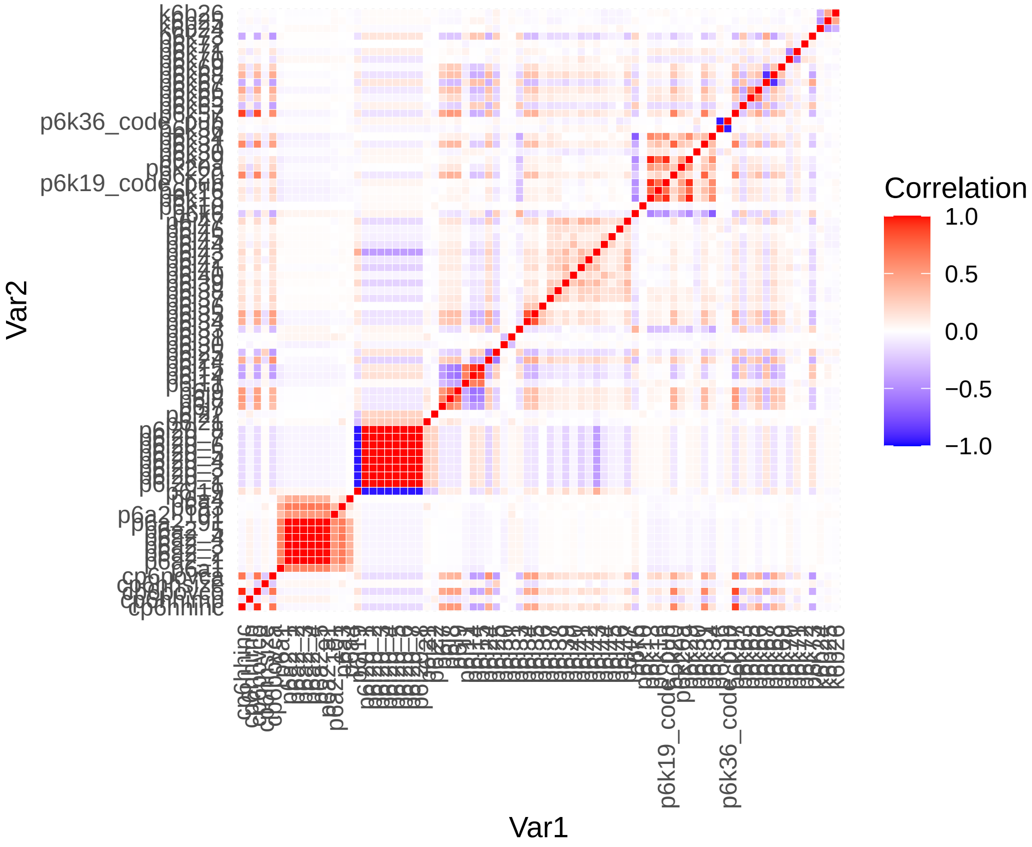
## Correlation MAtrix

```
# Filter out non-numeric columns
df_filtered <- df_filtered[, sapply(df_filtered, is.numeric)]

# Calculate correlation matrix
correlation_matrix <- cor(df_filtered)

# Convert correlation matrix to data frame for plotting
cor_df <- reshape2::melt(correlation_matrix)

# Plot heatmap
ggplot(cor_df, aes(Var1, Var2, fill = value)) +
  geom_tile(color = "white") +
  scale_fill_gradient2(low = "blue", high = "red", mid = "white",
                       midpoint = 0, limit = c(-1,1),
                       name="Correlation") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1)) +
  coord_fixed()
```



```
# Set correlation threshold
threshold <- 0.7

# Initialize an empty list to store variable pairs
high_correlation_pairs <- list()

# Iterate through the correlation matrix
for(i in 1:nrow(correlation_matrix)) {
  for(j in 1:ncol(correlation_matrix)) {
    # Exclude diagonal elements and redundant correlations
    if(i != j && j > i) {
      # Check if correlation is above the threshold
      if(correlation_matrix[i, j] > threshold) {
        # Add variable names to the list
        high_correlation_pairs <- c(high_correlation_pairs, list(c(rownames(correlation_matrix)[i], col
      }
    }
  }
}

# Print the list of variable pairs with correlations above the threshold
cat("Variables with correlations > 0.7:\n")
```

```
## Variables with correlations > 0.7:
```

```
for(pair in high_correlation_pairs) {
  cat(pair[[1]], "and", pair[[2]], "\n")
}
```

```
## cp6hhinc and cp6povco
## cp6hhinc and p6k57
## cp6povco and p6k57
## p6a2_1 and p6a2_2
## p6a2_1 and p6a2_3
## p6a2_1 and p6a2_4
## p6a2_1 and p6a2_5
## p6a2_1 and p6a2_91
## p6a2_2 and p6a2_3
## p6a2_2 and p6a2_4
## p6a2_2 and p6a2_5
## p6a2_2 and p6a2_91
## p6a2_3 and p6a2_4
## p6a2_3 and p6a2_5
## p6a2_3 and p6a2_91
## p6a2_4 and p6a2_5
## p6a2_4 and p6a2_91
## p6a2_5 and p6a2_91
## p6i20_1 and p6i20_2
## p6i20_1 and p6i20_3
## p6i20_1 and p6i20_4
## p6i20_1 and p6i20_5
## p6i20_1 and p6i20_6
## p6i20_1 and p6i20_7
## p6i20_1 and p6i20_8
## p6i20_2 and p6i20_3
## p6i20_2 and p6i20_4
## p6i20_2 and p6i20_5
## p6i20_2 and p6i20_6
## p6i20_2 and p6i20_7
## p6i20_2 and p6i20_8
## p6i20_3 and p6i20_4
## p6i20_3 and p6i20_5
## p6i20_3 and p6i20_6
## p6i20_3 and p6i20_7
## p6i20_3 and p6i20_8
## p6i20_4 and p6i20_5
## p6i20_4 and p6i20_6
## p6i20_4 and p6i20_7
## p6i20_4 and p6i20_8
## p6i20_5 and p6i20_6
## p6i20_5 and p6i20_7
## p6i20_5 and p6i20_8
## p6i20_6 and p6i20_7
## p6i20_6 and p6i20_8
## p6i20_7 and p6i20_8
## p6j8 and p6j9
## p6j14 and p6j15
## p6j34 and p6j35
## p6k13 and p6k16
## p6k13 and p6k19_code_pub
## p6k13 and p6k29
## p6k16 and p6k19_code_pub
## p6k16 and p6k29
## p6k19_code_pub and p6k29
## p6k20 and p6k31
```