

## Start Here

```
# Load necessary libraries
library(dplyr)
```

Attaching package: 'dplyr'

The following objects are masked from 'package:stats':

filter, lag

The following objects are masked from 'package:base':

intersect, setdiff, setequal, union

```
library(corrplot) # correlation matrix visualization
```

corrplot 0.92 loaded

```
library(ggcorrplot) # ggplot2 style visualization of correlation matrix
```

Loading required package: ggplot2

```
# Read datasets
data <- read.csv("../Data/data.csv")
data2 <- read.csv("../Data/500cities.csv")

# Prepare 'data' by padding and merging columns to create a unique identifier
data$STATEFP <- sprintf("%02d", as.numeric(data$STATEFP))
data$COUNTYFP <- sprintf("%03d", as.numeric(data$COUNTYFP))
data$TRACTCE <- sprintf("%06d", as.numeric(data$TRACTCE))
data$TractFIPS <- as.numeric(paste0(data$STATEFP, data$COUNTYFP, data$TRACTCE))

# Summarize 'data' to get mean of 'NatWalkInd' by 'TractFIPS'
averaged_data <- data %>%
  group_by(TractFIPS) %>%
  summarise(NatWalkInd = mean(NatWalkInd, na.rm = TRUE)) %>%
  ungroup()

# Select relevant columns from 'data2'
data2_relevant <- data2 %>%
  select(TractFIPS, StateAbbr, DIABETES_CrudePrev, BPHIGH_CrudePrev, OBESITY_CrudePrev, LPA_CrudePrev, C

# Merge 'averaged_data' with 'data2_relevant' on 'TractFIPS'
merged_data <- merge(averaged_data, data2_relevant, by = "TractFIPS", all.x = TRUE, all.y = TRUE)

# Filter for specific condition
merged_data <- merged_data %>% filter(StateAbbr == "CA")
```

```
# Calculate correlation matrix
cor_matrix <- cor(merged_data %>% select(-TractFIPS, -StateAbbr), use = "complete.obs") # Handling miss

# Output variable names (column names) used for correlation matrix
cat("Variables used for correlation matrix:\n")
```

Variables used for correlation matrix:

```
print(names(merged_data %>% select(-TractFIPS, -StateAbbr)))
```

```
[1] "NatWalkInd"          "DIABETES_CrudePrev" "BPHIGH_CrudePrev"
[4] "OBESITY_CrudePrev"   "LPA_CrudePrev"      "CSMOKING_CrudePrev"
```

```
# Make correlation matrix
corrplot(cor_matrix, method = "circle", type = "upper", order = "hclust",
          tl.col = "black", tl.srt = 45, # Text label color and rotation
          addCoef.col = "orange") # Add correlation coefficients to plot
```

