# Data Science I Final Project

Ayaka, Martina, and Rosie

## Set-Up

```
# Libraries
library(tidyr)
library(tidycensus)
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(purrr)
library(here)
```

```
here() starts at /Users/ayakasanu/Desktop/HBDS5018/dsi-final
```

```
library(tidyverse)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v forcats   1.0.1     v readr     2.1.5
v ggplot2   4.0.0     v stringr   1.5.2
v lubridate 1.9.4     v tibble    3.3.0
```

```
-- 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 becor
```

```
library(psych)
```

```
Attaching package: 'psych'

The following objects are masked from 'package:ggplot2':

    %+%, alpha
```

```
library(ggplot2)
library(cluster)
library(GGally)
library(factoextra)
```

```
Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa
```

```
library(ggcorrplot)
library(maps)
```

```
Attaching package: 'maps'

The following object is masked from 'package:cluster':

    votes.repub

The following object is masked from 'package:purrr':

    map
```

```
library(gtsummary)
library(gt)
library(clustertend)
```

```
Package `clustertend` is deprecated.  Use package `hopkins` instead.
```

```r
library(NbClust)
library(clValid)
library(mclust)
```

```
Package 'mclust' version 6.1.2
Type 'citation("mclust")' for citing this R package in publications.

Attaching package: 'mclust'

The following object is masked from 'package:maps':

    map

The following object is masked from 'package:psych':

    sim

The following object is masked from 'package:purrr':

    map

The following object is masked from 'package:dplyr':

    count
```

```r
library(sf)
```

```
Linking to GEOS 3.13.0, GDAL 3.8.5, PROJ 9.5.1; sf_use_s2() is TRUE
```

```r
library(here)
```

## Data Construction

### ACS Tables:

- B02001: Race
  - Includes total population and counts for major racial groups (White alone, Black alone, Asian alone, Two or more races)

- B15003: Educational Attainment (Population 25 Years and Older)
  - Includes high school diploma, bachelors degree, and doctorate attainment counts
- B25003: Housing Tenure
  - Includes counts of owner-occupied vs renter-occupied housing units
- B07007: Geographic Mobility by Citizenship Status
  - Includes total population for mobility, native population, and foreign-born population
- B17001: Poverty Status in the Past 12 Months
  - Includes total poverty universe and individuals below the poverty level
- B08006: Means of Transportation to Work
  - Includes total workers and counts of individuals using public transportation to commute
- B08303: Travel Time to Work
  - Includes total workers and those with long commute times (90+ minutes)
- B09002: Family Type and Subfamilies
  - Includes total family population and married-couple families
- B23025: Employment Status
  - Includes total civilian population age 16+ and unemployed individuals
- B28002: Presence and Types of Internet Subscriptions in Household
- Includes total and those with an internet subscription

### Pulling Data and Variables

```
# Variable list for ACS pull with explanation of variable
variables <-
  c(
    # Race
    'B02001_001', # Total Population for Races
    'B02001_002', # White Alone
    'B02001_003', # Black or African American Alone
    'B02001_005', # Asian Alone
    'B02001_009', # Two or more races
```

```
    # Educational Attainment
    'B15003_001', # Total Education Population
    'B15003_017', # High School Diploma
    'B15003_022', # bachelors Degree
    'B15003_025', # Doctorate Degree

    # Housing Status
    'B25003_001', # Total Housing Units
    'B25003_002', # Owner Occupied
    'B25003_003', # Renter Occupied

    # Geographic Mobility by Citizenship
    'B07007_001', # Total Population for Mobility
    'B07007_002', # Native Population
    'B07007_003', # Foreign Born Population

    # Poverty Status
    "B17001_001", # Total
    "B17001_002", # Income in the past 12 months below poverty level

    # Transportation to Work
    "B08006_001",  # Total workers
    "B08006_008",  # Public transportation

    # Travel time to work
    "B08303_001", # Total workers
    "B08303_013", # Workers with commute time 90 or more minutes

    # Family Characteristics/Structure
    "B09002_001",  # Total population in families
    "B09002_002",  # In married-couple families

    # Employment Status
    "B23025_001", # Total Population 16+
    "B23025_005",  # Unemployed

    # Internet
    "B28002_001", # Total
    "B28002_002" # With Internet Subscription in Household
    )
```

```r
# Retrieves ACS estimates for counties in TX (2019-2023 5-year ACS)
tx_counties <- get_acs(
  geography = "county",
  state = "TX",
  variables = variables,
  year = 2023,
  survey = "acs5"
)
```

```
Getting data from the 2019-2023 5-year ACS
```

```r
tx_counties <- tx_counties %>%
  select(GEOID, NAME, variable, estimate) %>%
  pivot_wider(names_from = variable, values_from = estimate)
```

**Converting variables to percentgaes**

```r
acs_clean <- tx_counties %>%
  mutate(
    # Race
    percent_white = B02001_002 / B02001_001 * 100,
    percent_black = B02001_003 / B02001_001 * 100,
    percent_asian = B02001_005 / B02001_001 * 100,
    percent_two_or_more  = B02001_009 / B02001_001 * 100,

    # Education
    percent_hs_diploma = B15003_017 / B15003_001 * 100,
    percent_bachelors = B15003_022 / B15003_001 * 100,
    percent_doctorate = B15003_025 / B15003_001 * 100,

    # Housing
    percent_owner_occupied = B25003_002 / B25003_001 * 100,
    percent_renter = B25003_003 / B25003_001 * 100,

    # Mobility
    percent_native_us = B07007_002 / B07007_001 * 100,
    percent_foreign_born = B07007_003 / B07007_001 * 100,

    # Poverty
```

```
    percent_poverty = B17001_002 / B17001_001 * 100,

    # Public Transit
    percent_public_transit = B08006_008 / B08006_001 * 100,

    # Long Commute to Work
    percent_long_commute = B08303_013 / B08303_001 * 100,

    # Family Structure
    percent_married_family = B09002_002 / B09002_001 * 100,

    # Unemployment
    percent_unemployed = B23025_005 / B23025_001 * 100,

    # Internet
    percent_internet = B28002_002 / B28002_001 * 100
  )
```

**Dataset with Percentage Variables**

```
# Dataset With the Percentage Variables/Data
# Rename to cleaner names
data <- acs_clean %>%
  select(NAME, percent_white, percent_black, percent_asian, percent_two_or_more,
         percent_hs_diploma, percent_bachelors, percent_doctorate, percent_owner_occupied,
         percent_renter, percent_native_us, percent_foreign_born,
         percent_poverty, percent_public_transit, percent_long_commute,
         percent_married_family, percent_unemployed, percent_internet) %>%
  rename(
    White = percent_white,
    Black = percent_black,
    Asian = percent_asian,
    Two_Races = percent_two_or_more,
    HS_Diploma = percent_hs_diploma,
    Bachelors = percent_bachelors,
    Doctorate = percent_doctorate,
    Owner_Occupied = percent_owner_occupied,
    Renter = percent_renter,
    Native = percent_native_us,
    Foreign_Born = percent_foreign_born,
```

```
    Poverty = percent_poverty,
    Public_Transit = percent_public_transit,
    Long_Commute = percent_long_commute,
    Married_Family = percent_married_family,
    Unemployed = percent_unemployed,
    Internet_Subscription = percent_internet
  )
```

## EDA

### Summary Table

```
# Create summary table with tbl_summary
summary_table <- data %>%
  select(-NAME) %>%
  tbl_summary(
    statistic = all_continuous() ~ "{mean} ({sd}) [{min}, {max}]", # Summary Stats
    digits = all_continuous() ~ 2, # 2 decimal places
    label = list(
      White ~ "White (%)",
      Black ~ "Black (%)",
      Asian ~ "Asian (%)",
      Two_Races ~ "Two or More Races (%)",
      HS_Diploma ~ "High School Diploma (%)",
      Bachelors ~ "Bachelors Degree (%)",
      Doctorate ~ "Doctorate Degree (%)",
      Owner_Occupied ~ "Owner-Occupied Housing (%)",
      Renter ~ "Renter-Occupied Housing (%)",
      Native ~ "Native Population (%)",
      Foreign_Born ~ "Foreign-Born Population (%)",
      Poverty ~ "Below Poverty Line (%)",
      Public_Transit ~ "Public Transit Use (%)",
      Long_Commute ~ "Long Commute (90+ min) (%)",
      Married_Family ~ "Married-Couple Family (%)",
      Unemployed ~ "Unemployment Rate (%)",
      Internet_Subscription ~ "Home Internet Subscription (%)")
    ) %>%
  modify_caption(
    "Table 1: Socioeconomic Indicators at County Level")
```

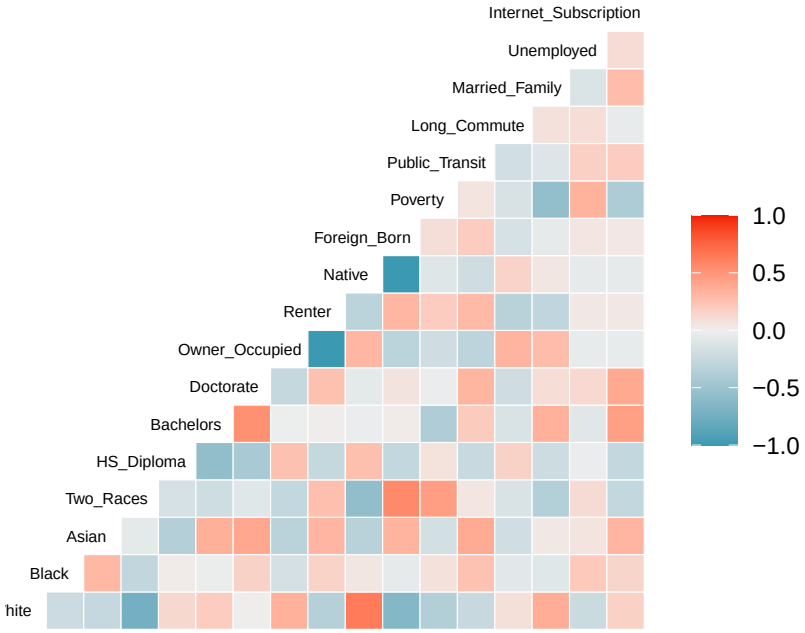
| Characteristic | N = 254[1] |
|---|---|
| White (%) | 68.36 (13.98) [19.23, 95.16] |
| Black (%) | 6.06 (6.23) [0.00, 33.14] |
| Asian (%) | 1.23 (2.29) [0.00, 21.83] |
| Two or More Races (%) | 13.63 (11.61) [0.00, 73.08] |
| High School Diploma (%) | 26.07 (5.76) [10.41, 45.90] |
| Bachelors Degree (%) | 14.83 (5.72) [0.00, 34.71] |
| Doctorate Degree (%) | 0.67 (0.67) [0.00, 5.86] |
| Owner-Occupied Housing (%) | 72.53 (9.25) [11.11, 94.51] |
| Renter-Occupied Housing (%) | 27.47 (9.25) [5.49, 88.89] |
| Native Population (%) | 90.79 (6.81) [61.54, 100.00] |
| Foreign-Born Population (%) | 9.21 (6.81) [0.00, 38.46] |
| Below Poverty Line (%) | 14.93 (6.39) [2.73, 44.83] |
| Public Transit Use (%) | 0.19 (0.34) [0.00, 2.10] |
| Long Commute (90+ min) (%) | 3.64 (2.53) [0.00, 18.40] |
| Married-Couple Family (%) | 71.16 (11.79) [17.28, 100.00] |
| Unknown | 2 |
| Unemployment Rate (%) | 2.62 (1.34) [0.00, 7.87] |
| Home Internet Subscription (%) | 83.61 (6.76) [52.94, 100.00] |

[1]Mean (SD) [Min, Max]

```
# Output
summary_table
```

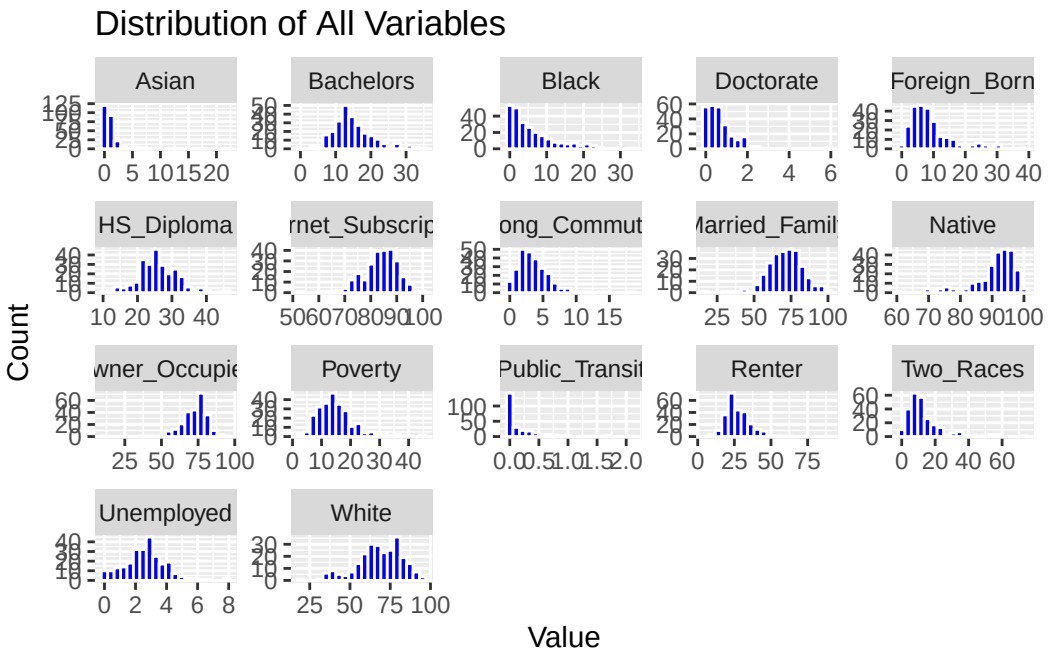
### Correlation Heatmap

```
ggcorr(data %>%
         select(-NAME),
       hjust = .9,
       size = 2)
```




```
# Histogram Plot
data %>%
  select(-NAME) %>%
  pivot_longer(everything(), names_to = "variable", values_to = "value") %>%
  ggplot(aes(value)) +
  geom_histogram(bins = 20, fill = "blue", color = "white") +
  facet_wrap(~ variable, scales = "free") +
  labs(title = "Distribution of All Variables", x = "Value", y = "Count")
```

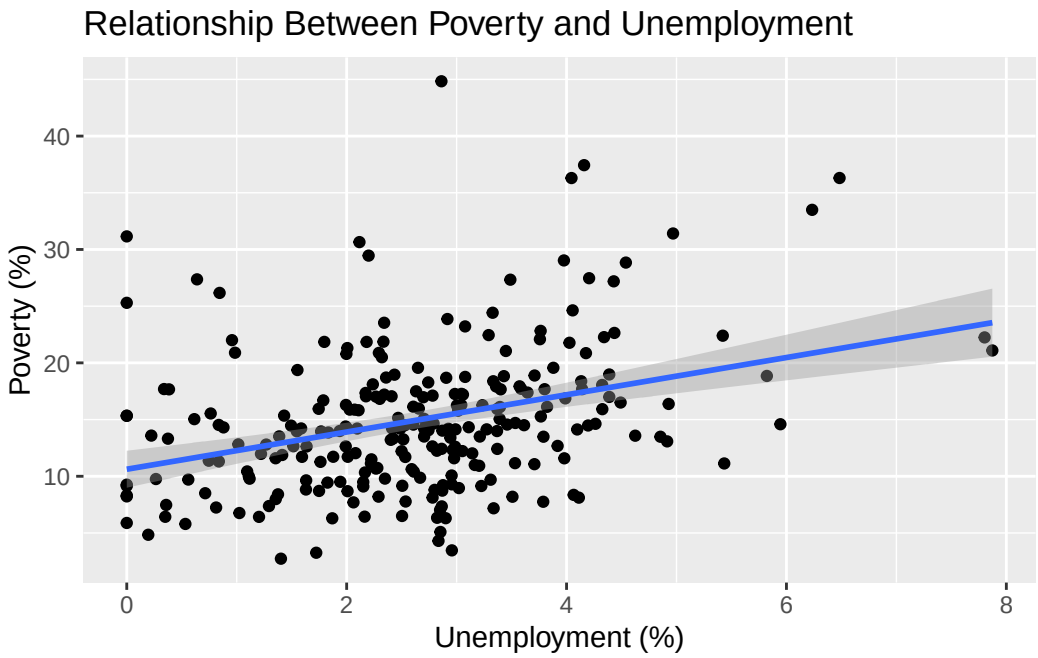
```
Warning: Removed 2 rows containing non-finite outside the scale range
(`stat_bin()`).
```




**Scatterplot: Poverty vs. Unemployment**

```
ggplot(data, aes(Unemployed, Poverty)) +
geom_point() +
geom_smooth(method = "lm") +
labs(title = "Relationship Between Poverty and Unemployment",
x = "Unemployment (%)",
y = "Poverty (%)")
```

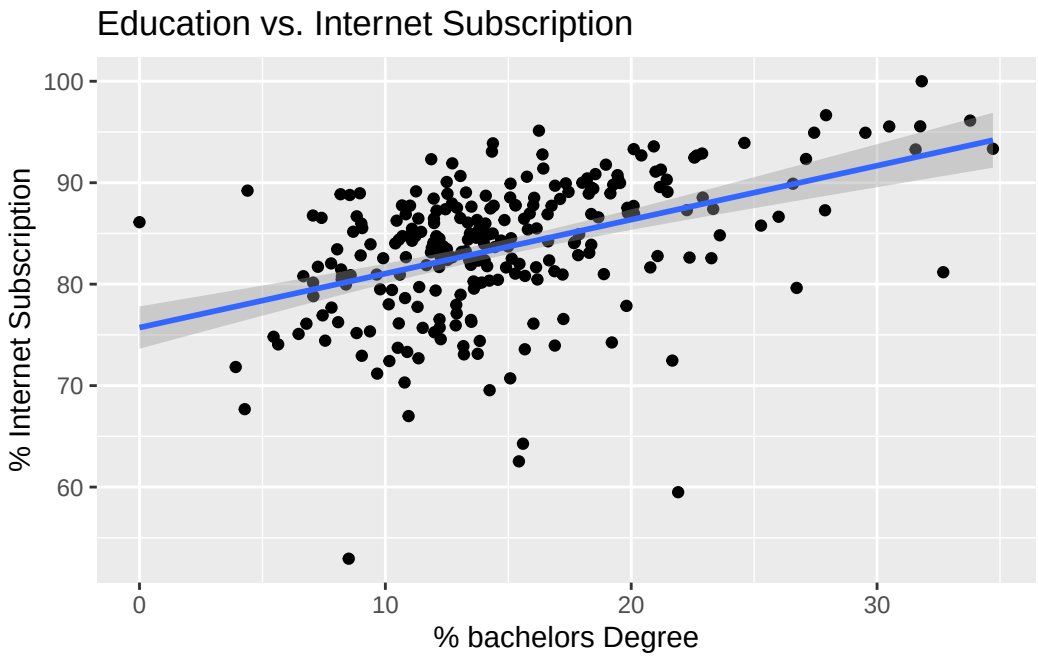
```
`geom_smooth()` using formula = 'y ~ x'
```




**Scatterplot: Education vs. Internet Subscription**

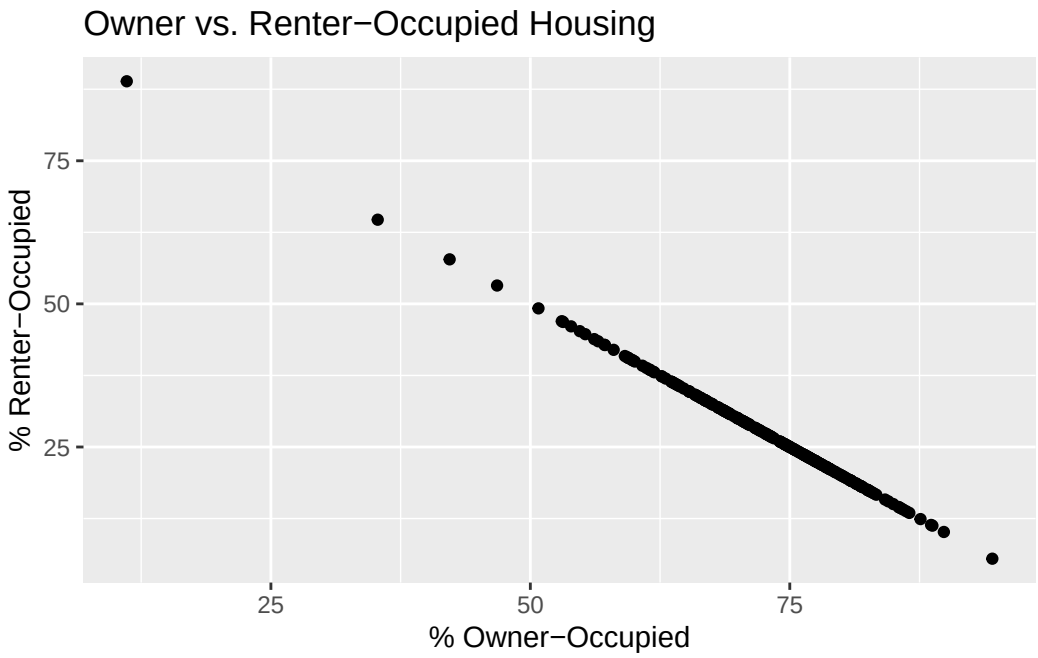
```
ggplot(data, aes(Bachelors, Internet_Subscription)) +
geom_point() +
geom_smooth(method = "lm") +
labs(title = "Education vs. Internet Subscription",
x = "% bachelors Degree",
y = "% Internet Subscription")
```

```
`geom_smooth()` using formula = 'y ~ x'
```




**Scatterplot: Owner-Occupied vs. Rented Housing**

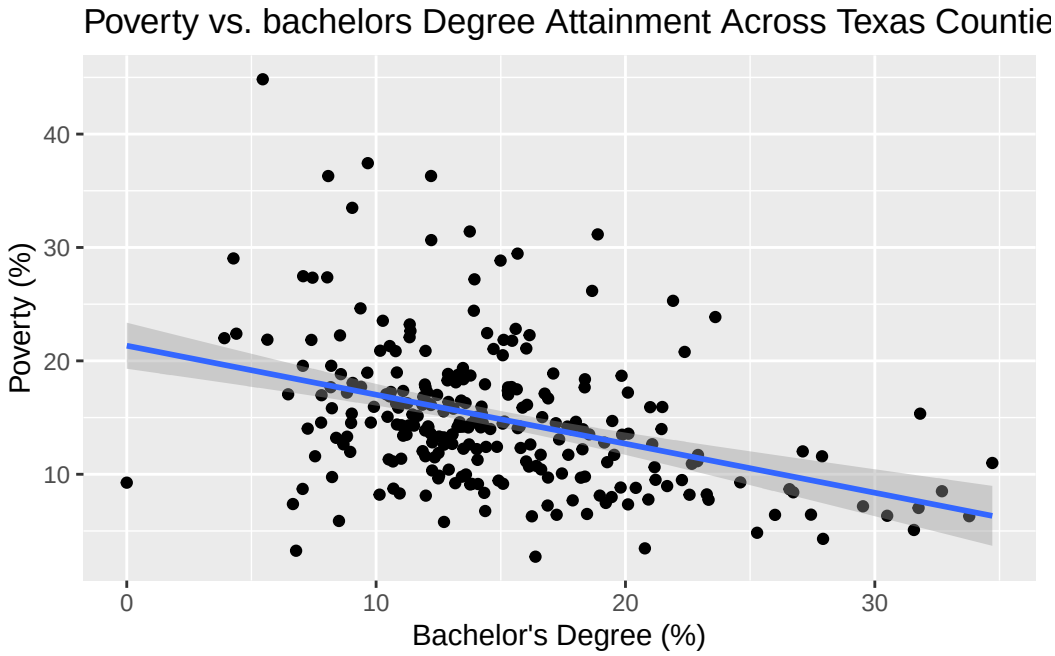
```
ggplot(data, aes(Owner_Occupied, Renter)) +
geom_point() +
labs(title = "Owner vs. Renter-Occupied Housing",
x = "% Owner-Occupied",
y = "% Renter-Occupied")
```




**Scatterplot of Education vs. Poverty**

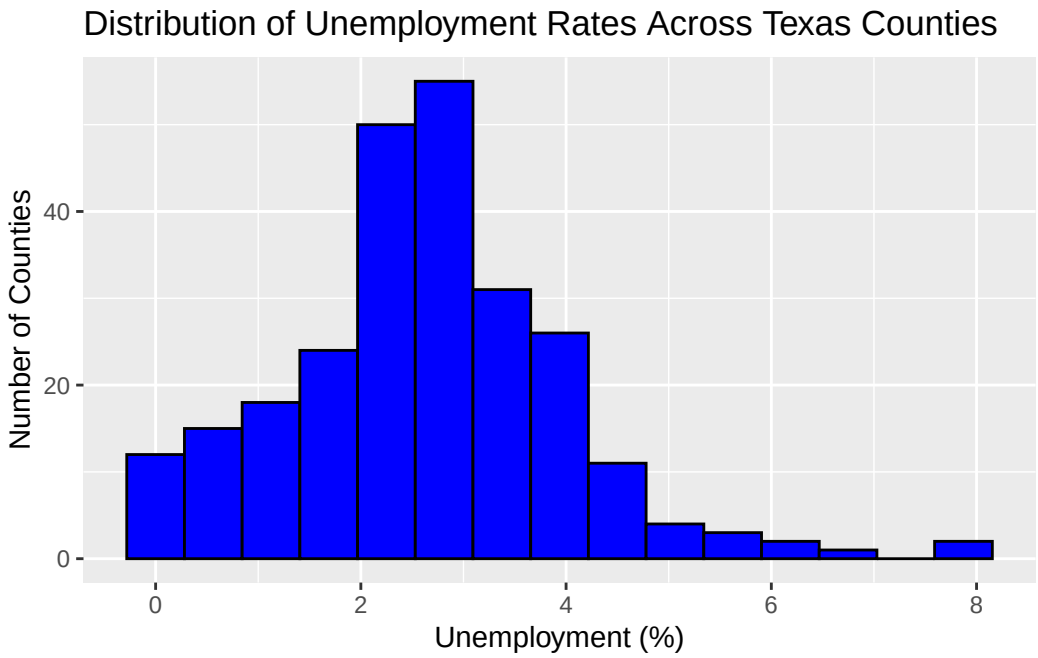
```
ggplot(data, aes(x = Bachelors, y = Poverty)) +
  geom_point() +
  geom_smooth(method = "lm") +
  labs(title = "Poverty vs. bachelors Degree Attainment Across Texas Counties",
    x = "Bachelor's Degree (%)",
    y = "Poverty (%)")
```

```
`geom_smooth()` using formula = 'y ~ x'
```




**Unemployment Histogram**

```
ggplot(data, aes(Unemployed)) +
  geom_histogram(bins = 15, fill = "blue", col = "black") +
  labs(
    title = "Distribution of Unemployment Rates Across Texas Counties",
    x = "Unemployment (%)",
    y = "Number of Counties"
  )
```




### Maps for all Variables

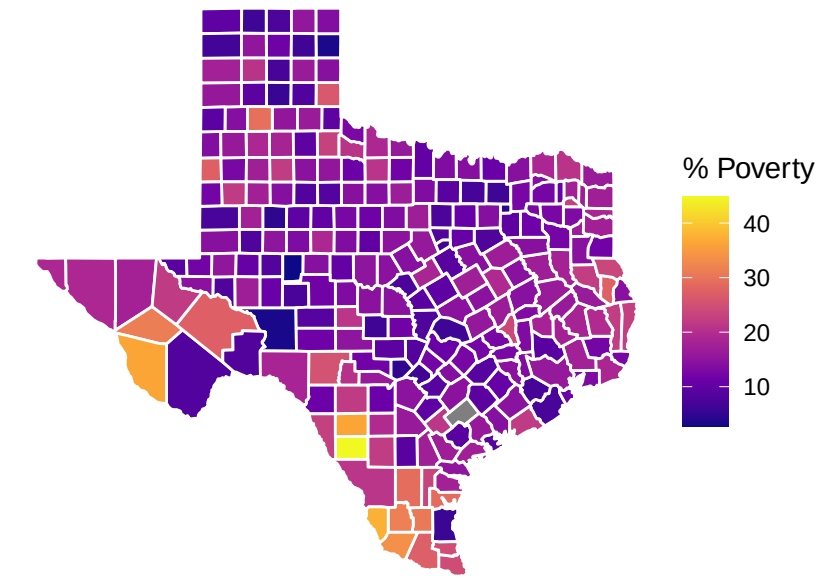
```
# Set-Up
# Extract Shape File
tx_shapes <- st_read("tl_rd22_us_county.shp")

# Read Texas Only
tx_shapes <- tx_shapes %>%
  filter(STATEFP == "48")

# Join to acs_clean
map_data <- tx_shapes %>%
  left_join(acs_clean, by = "GEOID")
```

### Variable Maps

```
county_map <- map_data("county") %>%
  filter(region == "texas")

education_plot <- data %>%
```

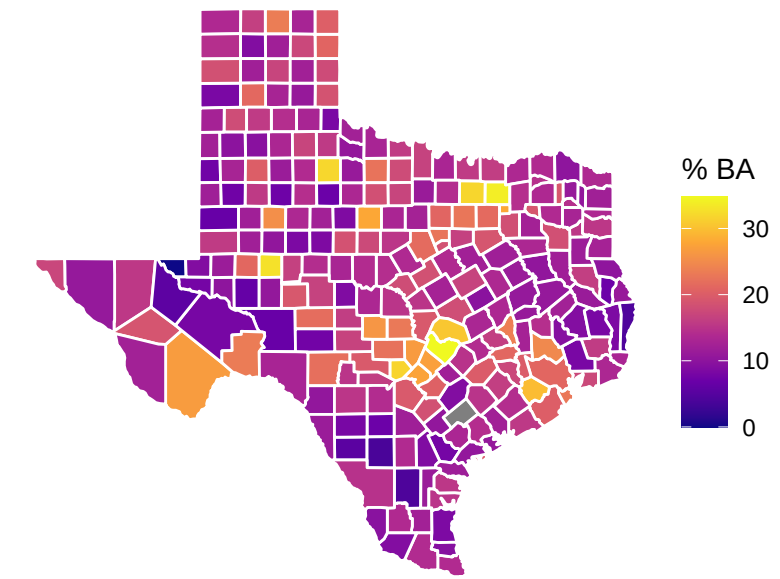
```
  mutate(
    county = tolower(NAME),
    county = gsub(",.*", "", county), # remove everything after a comma
    county = gsub(" county$", "", county), # remove " county"
    county = trimws(county)
  )

merged_map <- county_map %>%
  left_join(education_plot, by = c("subregion" = "county"))
```

```
# Map for Poverty
ggplot(merged_map, aes(long, lat, group = group, fill = Poverty)) +
  geom_polygon(color = "white") +
  coord_map() +
  scale_fill_viridis_c(option = "plasma") +
  labs(
    title = "Poverty Rates Across Texas Counties",
    fill = "% Poverty"
  ) +
  theme_void()
```



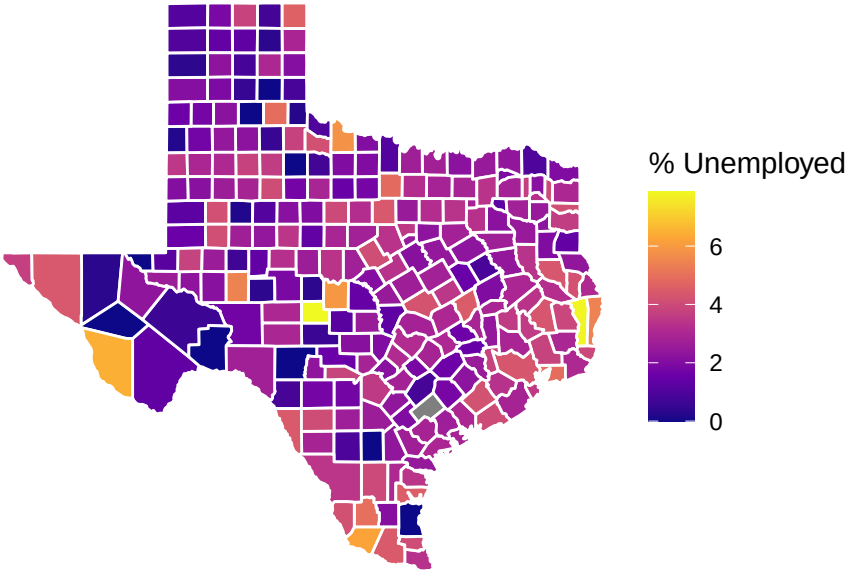

```
# Map for Bachelors Degree
ggplot(merged_map, aes(long, lat, group = group, fill = Bachelors)) +
```

```
  geom_polygon(color = "white") +
  coord_map() +
  scale_fill_viridis_c(option = "plasma") +
  labs(
    title = "bachelors Degree Attainment Across Texas Counties",
    fill = "% BA"
  ) +
  theme_void()
```

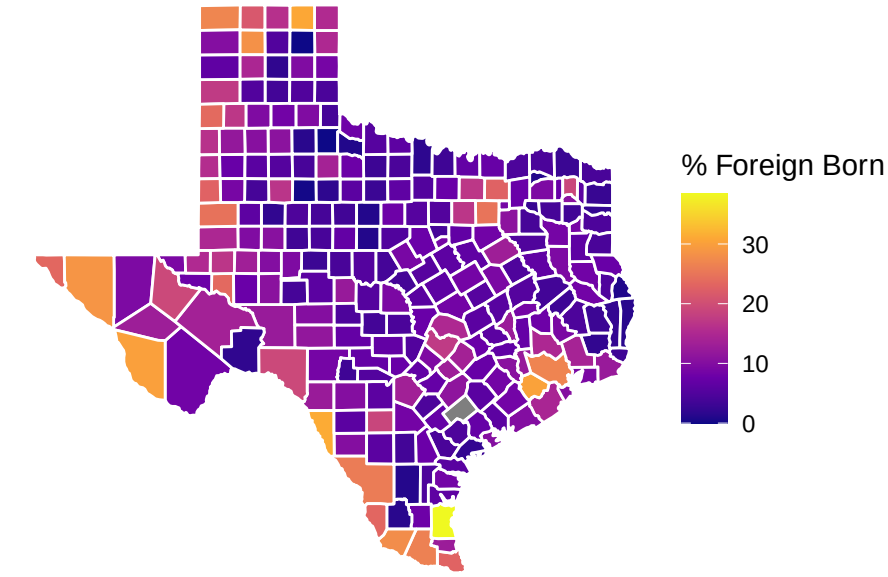



```
# Map for Unemployment
ggplot(merged_map, aes(long, lat, group = group, fill = Unemployed)) +
  geom_polygon(color = "white") +
  coord_map() +
  scale_fill_viridis_c(option = "plasma") +
  labs(
    title = "Unemployment Rates Across Texas Counties",
    fill = "% Unemployed"
  ) +
  theme_void()
```

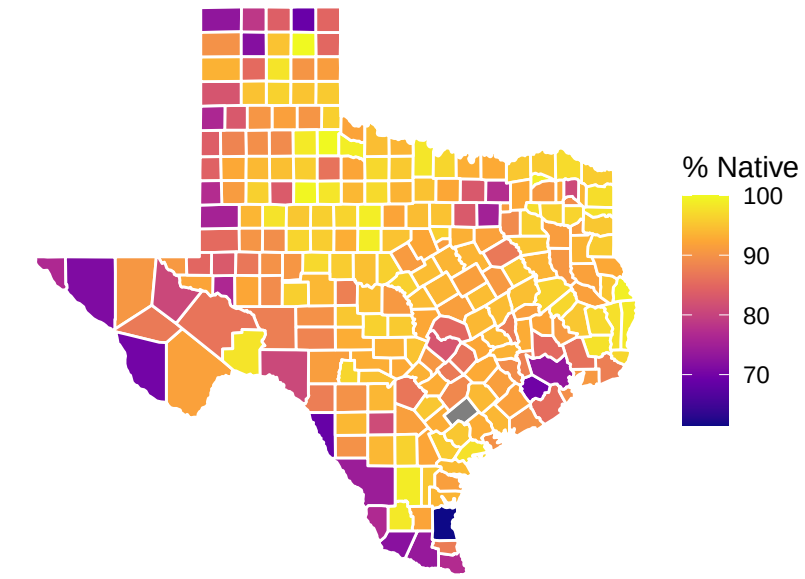
Unemployment Rates Across Texas Counties




```
# Map for Foreign-Born Population
ggplot(merged_map, aes(long, lat, group = group, fill = Foreign_Born)) +
  geom_polygon(color = "white") +
  coord_map() +
  scale_fill_viridis_c(option = "plasma") +
  labs(
    title = "Foreign-Born Population Across Texas Counties",
    fill = "% Foreign Born"
  ) +
  theme_void()
```

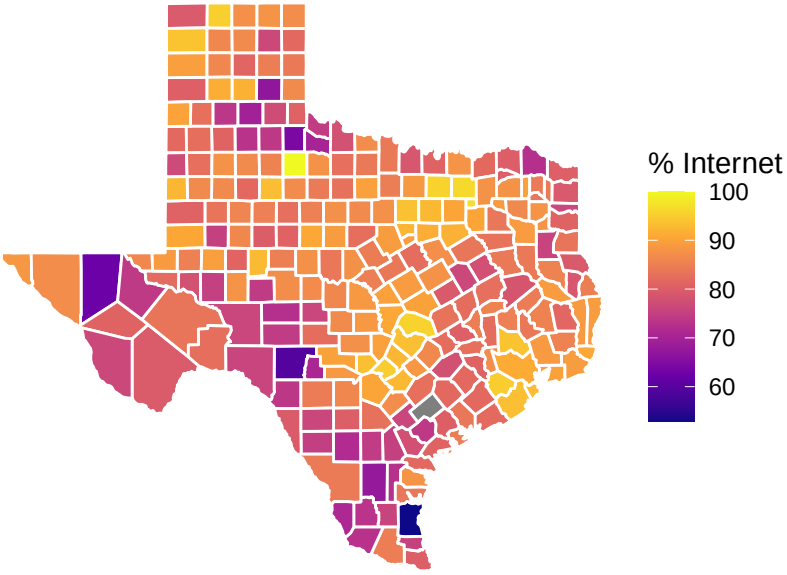



```
# Map for Native-Born Population
ggplot(merged_map, aes(long, lat, group = group, fill = Native)) +
  geom_polygon(color = "white") +
  coord_map() +
  scale_fill_viridis_c(option = "plasma") +
  labs(
    title = "Native-Born Population Across Texas Counties",
    fill = "% Native"
  ) +
  theme_void()
```

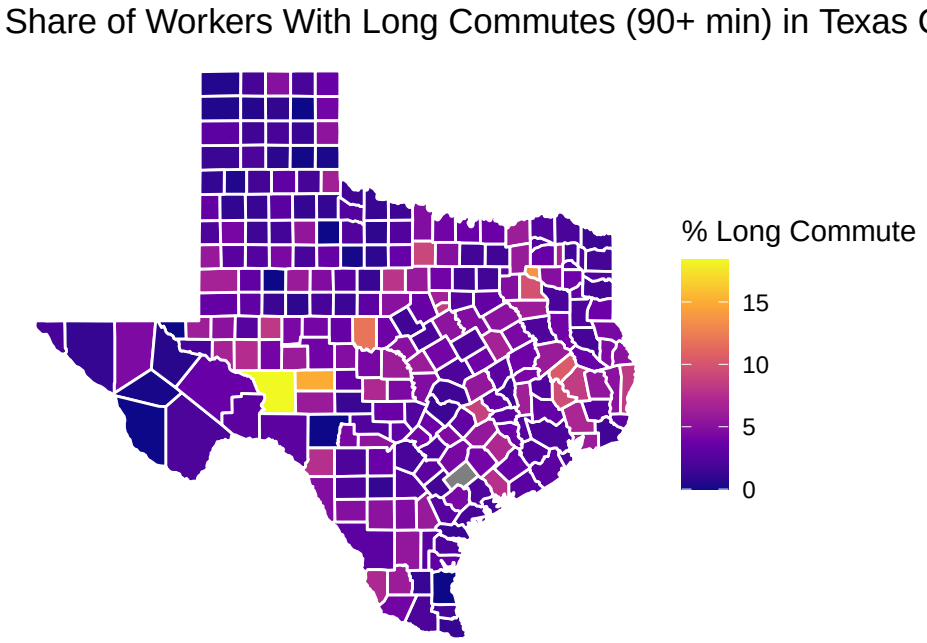



```
# Map for Internet Subscription
ggplot(merged_map, aes(long, lat, group = group, fill = Internet_Subscription)) +
  geom_polygon(color = "white") +
  coord_map() +
  scale_fill_viridis_c(option = "plasma") +
  labs(
    title = "Home Internet Subscription Across Texas Counties",
    fill = "% Internet"
  ) +
  theme_void()
```

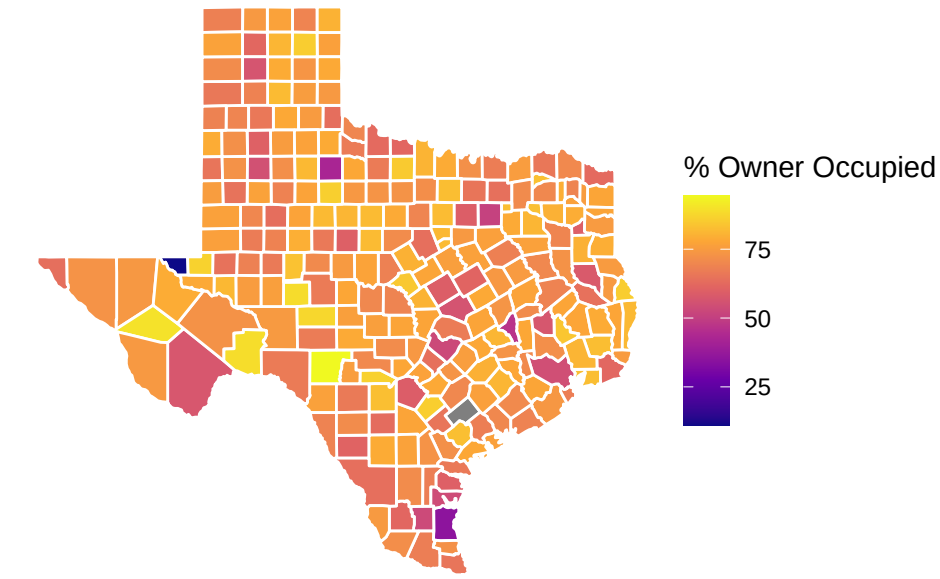
Home Internet Subscription Across Texas Counties



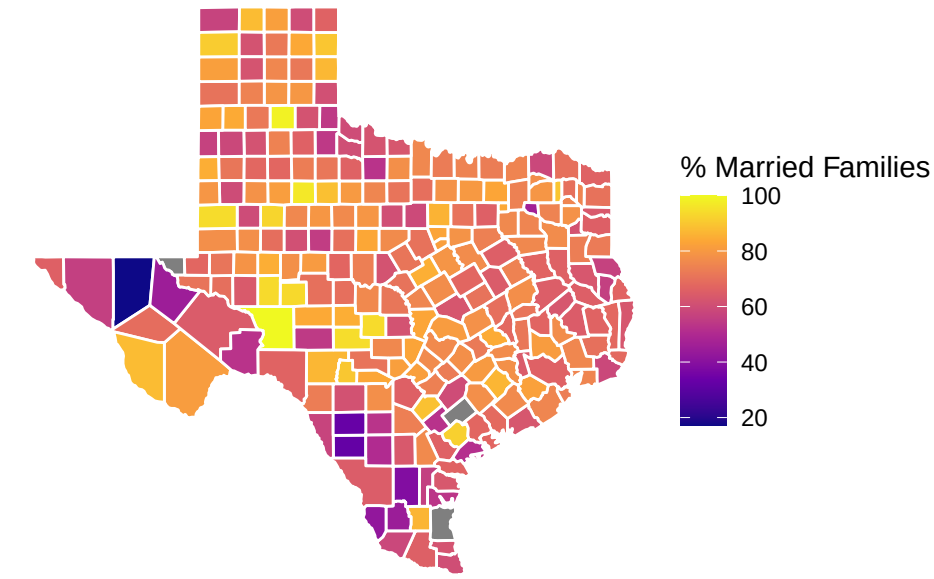

```
# Map for Long Commute
ggplot(merged_map, aes(long, lat, group = group, fill = Long_Commute)) +
  geom_polygon(color = "white") +
  coord_map() +
  scale_fill_viridis_c(option = "plasma") +
  labs(
    title = "Share of Workers With Long Commutes (90+ min) in Texas Counties",
    fill = "% Long Commute"
  ) +
  theme_void()
```




```
# Owning House Map
ggplot(merged_map, aes(long, lat, group = group, fill = Owner_Occupied)) +
  geom_polygon(color = "white") +
  coord_map() +
  scale_fill_viridis_c(option = "plasma") +
  labs(
    title = "Owner-Occupied Housing Across Texas Counties",
    fill = "% Owner Occupied"
  ) +
  theme_void()
```




```
# Married-Couple Family Map
ggplot(merged_map, aes(long, lat, group = group, fill = Married_Family)) +
  geom_polygon(color = "white") +
  coord_map() +
  scale_fill_viridis_c(option = "plasma") +
  labs(
    title = "Married-Couple Families Across Texas Counties",
    fill = "% Married Families"
  ) +
  theme_void()
```

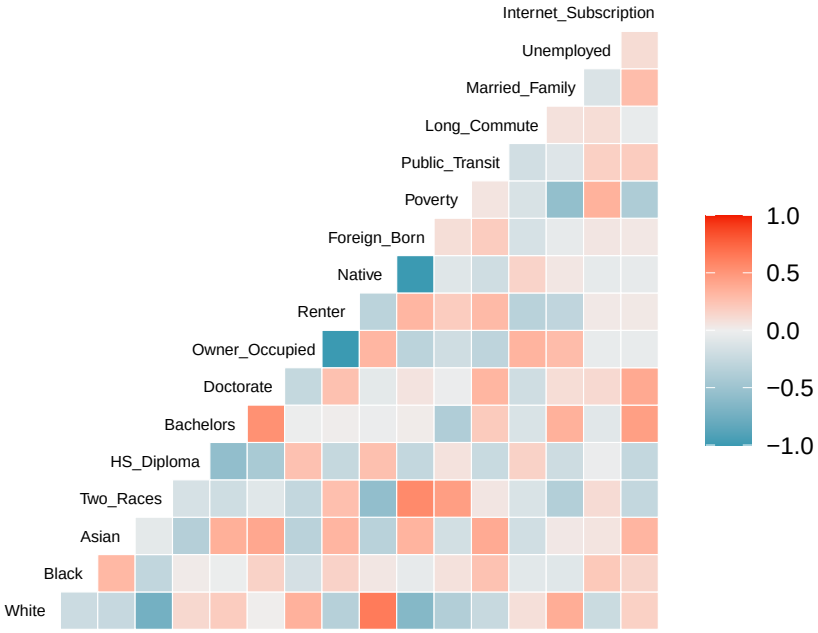



## PCA

### Correlation Heatmap

```
# Correlation Heatmap
ggcorr(data, hjust = .9, size = 2, layout.exp = 2)
```

```
Warning: data in column NAME is not numeric and was ignored
```




## NA

```
# Check which have NAs
colSums(is.na(data %>% select(-NAME)))
```

```
             White              Black              Asian
                 0                  0                  0
         Two_Races          HS_Diploma          Bachelors
                 0                  0                  0
         Doctorate       Owner_Occupied             Renter
                 0                  0                  0
            Native        Foreign_Born            Poverty
                 0                  0                  0
      Public_Transit       Long_Commute     Married_Family
                 0                  0                  2
        Unemployed Internet_Subscription
                 0                  0
```

```
# Remove NAs for PCA
clean_data <- data %>%
  select(-NAME) %>%
  na.omit() # Removes states with missing values
```

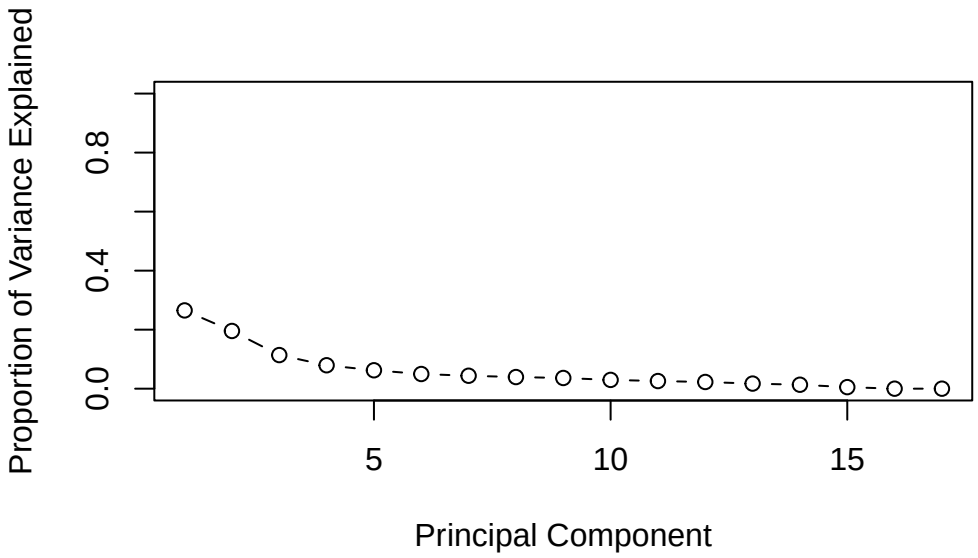
### Run PCA

```r
# Run PCA
pca_result <- prcomp(clean_data, center = TRUE, scale. = TRUE)

# Results as df
pc_scores <- as.data.frame(pca_result$x)

# PCA Summary
summary(pca_result)
```

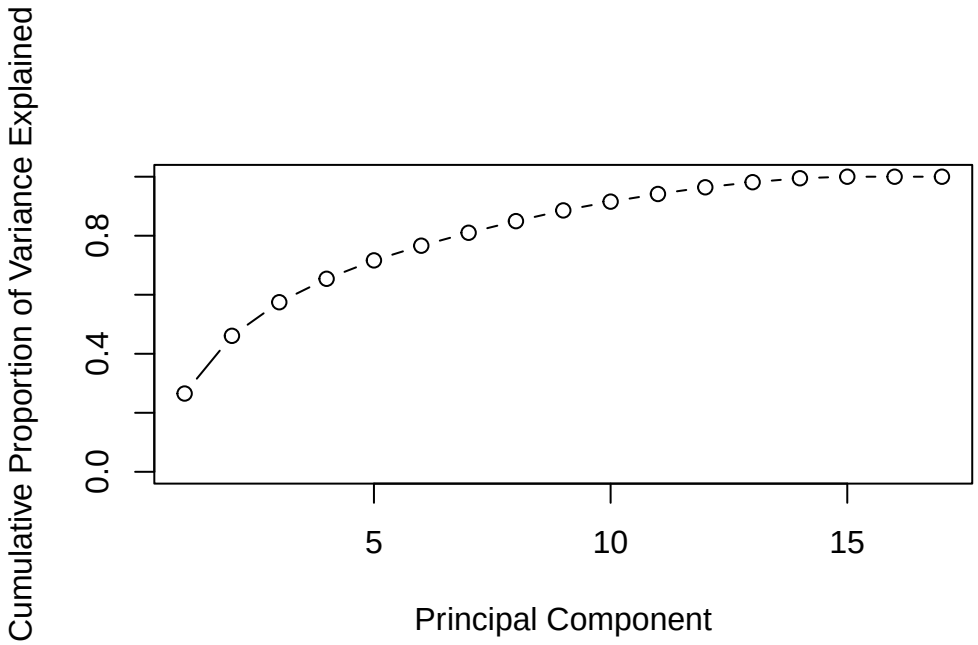
```
Importance of components:
                          PC1    PC2    PC3     PC4     PC5     PC6     PC7
Standard deviation     2.1235 1.8233 1.3915 1.16150 1.03015 0.91980 0.86320
Proportion of Variance 0.2652 0.1956 0.1139 0.07936 0.06242 0.04977 0.04383
Cumulative Proportion  0.2652 0.4608 0.5747 0.65404 0.71647 0.76623 0.81006
                           PC8     PC9    PC10    PC11   PC12    PC13    PC14
Standard deviation     0.81827 0.78561 0.71121 0.66364 0.6212 0.54151 0.47695
Proportion of Variance 0.03939 0.03631 0.02975 0.02591 0.0227 0.01725 0.01338
Cumulative Proportion  0.84945 0.88575 0.91551 0.94141 0.9641 0.98136 0.99475
                          PC15      PC16      PC17
Standard deviation     0.29888 9.801e-15 5.469e-16
Proportion of Variance 0.00525 0.000e+00 0.000e+00
Cumulative Proportion  1.00000 1.000e+00 1.000e+00
```

### Proportion of Variance and Cummulative Variance

```r
## Compute manually
explained_variance <- pca_result$sdev^2 / sum(pca_result$sdev^2)

## Use summary output
explained_variance <- summary(pca_result)$importance[2,]

# Proportion of Variance explained (PVE) by each PC
plot(explained_variance, xlab = "Principal Component",
     ylab = "Proportion of Variance Explained", ylim = c(0, 1),
     type = "b")
```



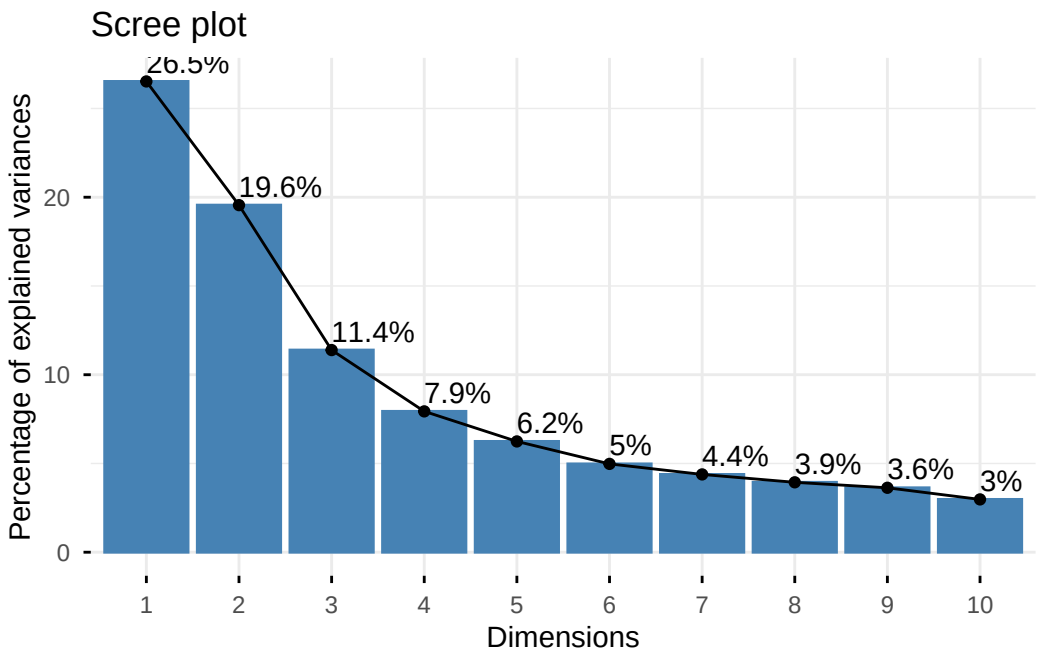

```
# Cumulative PVE across all PCs
plot(cumsum(explained_variance), xlab = "Principal Component",
     ylab = "Cumulative Proportion of Variance Explained",
     ylim = c(0, 1), type = "b")
```



### Scree Plot

```
# Scree Plot through Function
fviz_eig(pca_result, addlabels = TRUE)
```

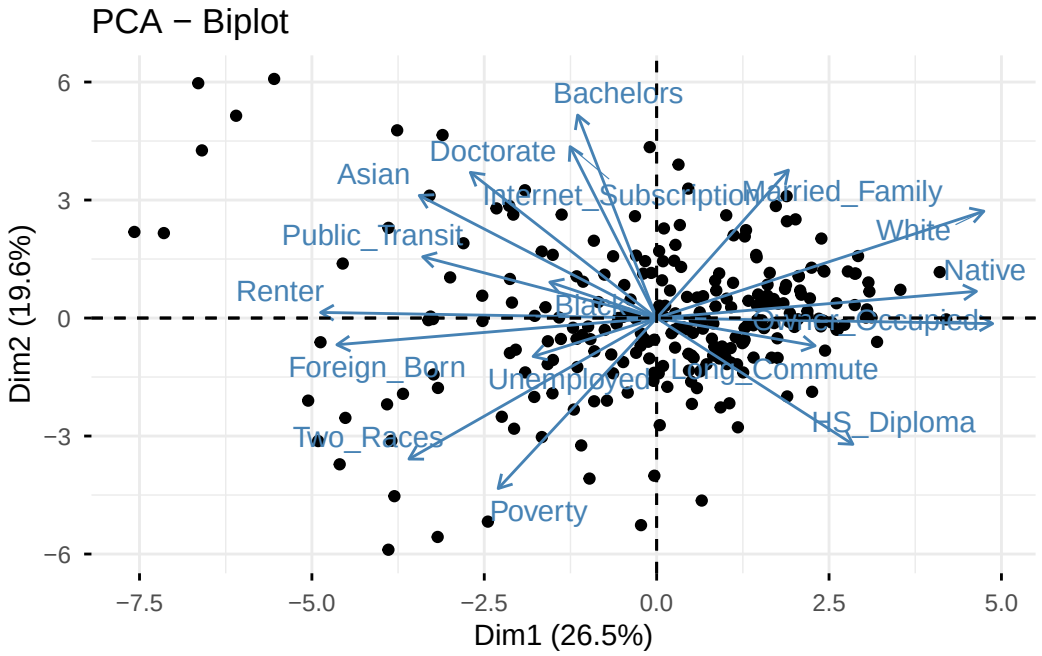
```
Warning in geom_bar(stat = "identity", fill = barfill, color = barcolor, :
Ignoring empty aesthetic: `width`.
```




### Biplot

```
# Biplot
fviz_pca_biplot(pca_result, geom.ind = "point", repel = TRUE)
```

```
Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
i Please use `linewidth` instead.
i The deprecated feature was likely used in the ggpubr package.
  Please report the issue at <https://github.com/kassambara/ggpubr/issues>.
```

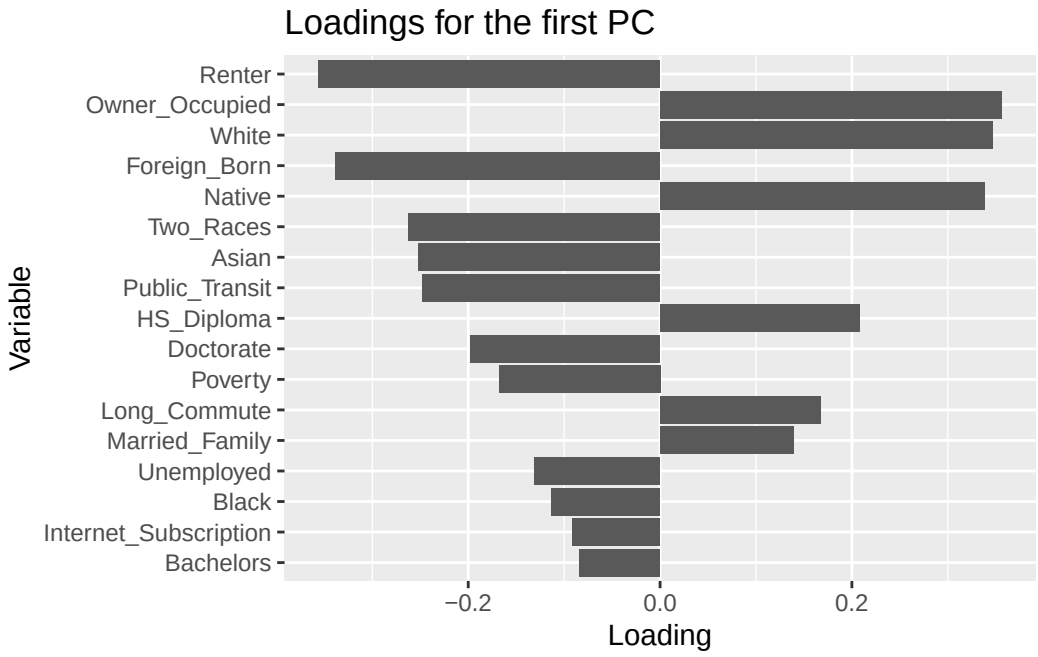



**Loadings and Scores**

```
# Extract loadings from the PCA result
loadings <- pca_result$rotation[, 1]

loadings
```

```
          White           Black           Asian
     0.34664484    -0.11322051    -0.25149658
      Two_Races     HS_Diploma      Bachelors
    -0.26238298     0.20789380    -0.08370410
     Doctorate  Owner_Occupied         Renter
    -0.19737288     0.35595877    -0.35595877
         Native   Foreign_Born        Poverty
     0.33875706    -0.33875706    -0.16789253
  Public_Transit   Long_Commute  Married_Family
    -0.24787041     0.16785548     0.13955437
    Unemployed Internet_Subscription
    -0.13089104    -0.09167275
```

```r
# Plot
ggplot() +
  geom_bar(
    aes(
      x = reorder(names(loadings),
                  abs(loadings)),
      y = loadings), stat = "identity") +
  coord_flip() +
  labs(title = paste("Loadings for the first PC"), x = "Variable", y = "Loading")
```



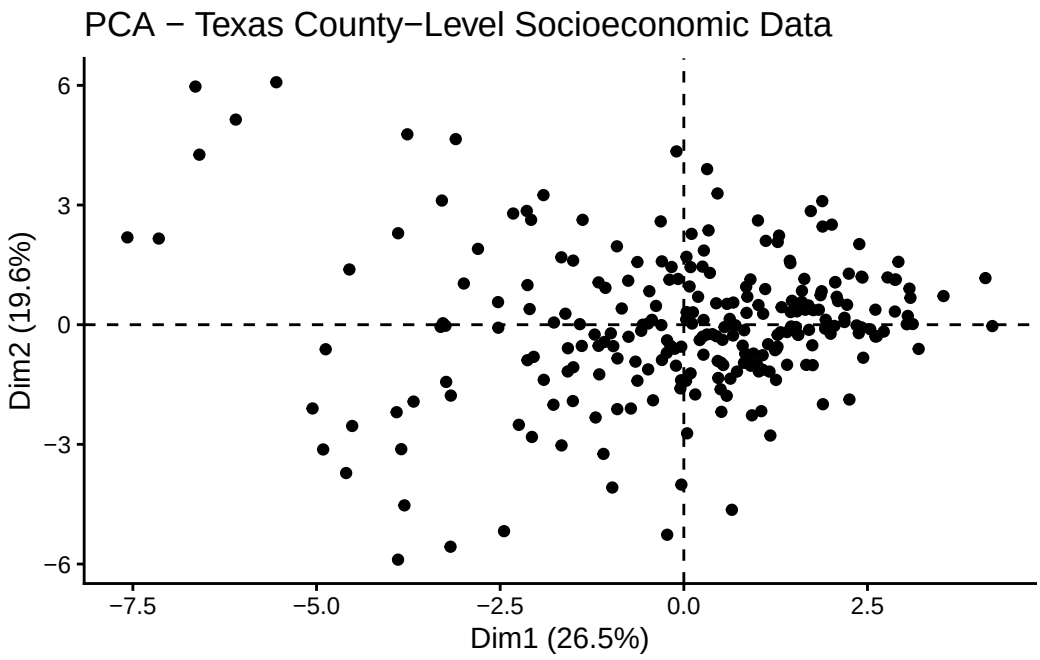

## Clustering

### Read and Visaulize Data

```r
# Identify which rows have no missing values
rows_used <- complete.cases(data %>% select(-NAME))

# Standardize the dataset (Removing name and scaling)
df <- data %>%
  select(-NAME) %>%
```

```
  na.omit() %>%
  scale()
```

```
# Plot the Dataset (Lab 3)
fviz_pca_ind(
  prcomp(df),
  title = "PCA - Texas County-Level Socioeconomic Data",
  geom = "point",
  ggtheme = theme_classic()
)
```

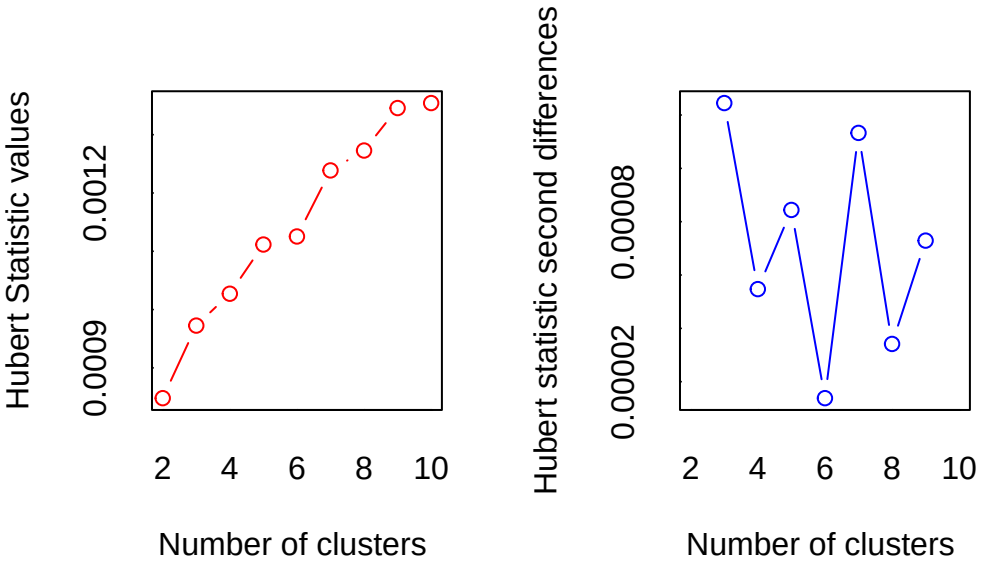



**Compute the Hopkins statistic**

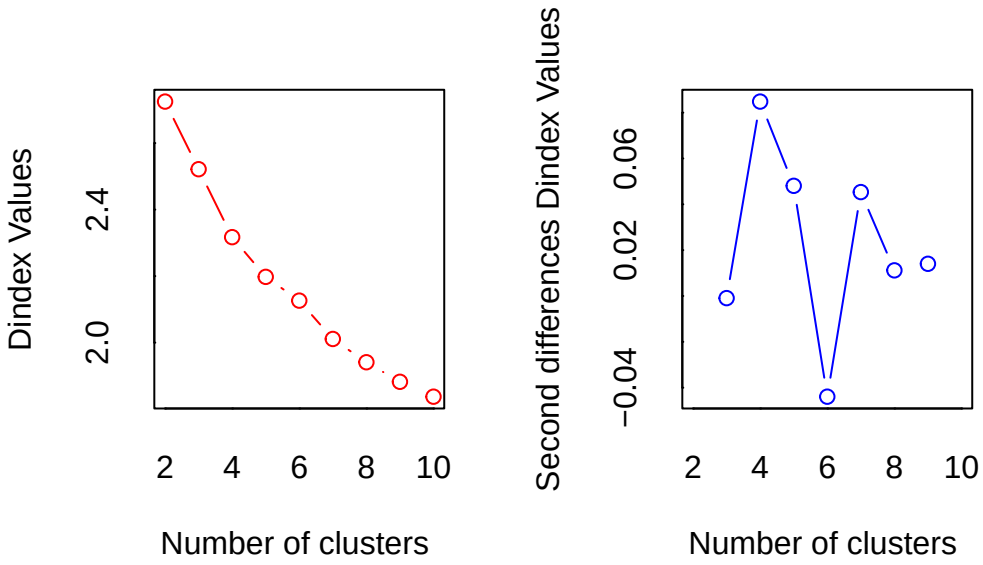
```
# Compute Hopkins statistic (Way from lab would not work)
set.seed(2025)

# m <- round(nrow(df) * 0.1)  # 10% of rows
# hopkins(df, m = m)
```

**NbClust() for best number of clusters**

```
pc <- prcomp(df, scale.=TRUE)
df_pc <- pc$x[, 1:5]  # use enough to explain ~80% variance

nb <- NbClust(df_pc, distance="euclidean", min.nc=2, max.nc=10, method="kmeans")
```



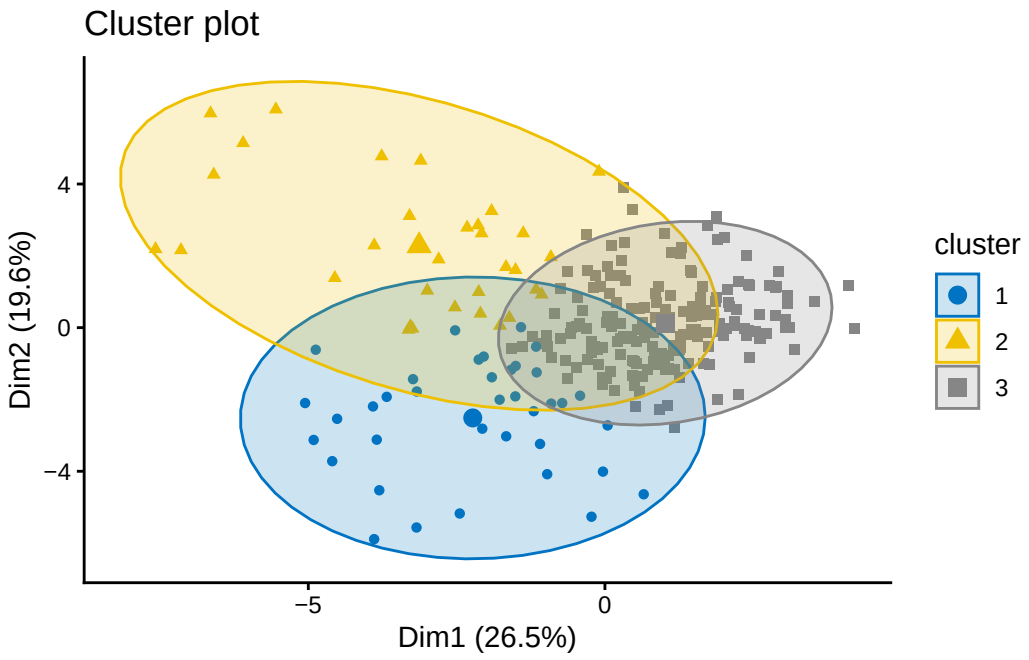

```
*** : The Hubert index is a graphical method of determining the number of clusters.
                In the plot of Hubert index, we seek a significant knee that corresponds to a
                significant increase of the value of the measure i.e the significant peak in
                index second differences plot.
```




```
*** : The D index is a graphical method of determining the number of clusters.
                In the plot of D index, we seek a significant knee (the significant peak in t
                second differences plot) that corresponds to a significant increase of the va
                the measure.

*****************************************************************
* Among all indices:
* 6 proposed 2 as the best number of clusters
* 9 proposed 3 as the best number of clusters
* 1 proposed 4 as the best number of clusters
* 1 proposed 5 as the best number of clusters
* 3 proposed 7 as the best number of clusters
* 2 proposed 9 as the best number of clusters
* 1 proposed 10 as the best number of clusters

                  ***** Conclusion *****

* According to the majority rule, the best number of clusters is  3


*****************************************************************
```

### Performing K-Means Clustering

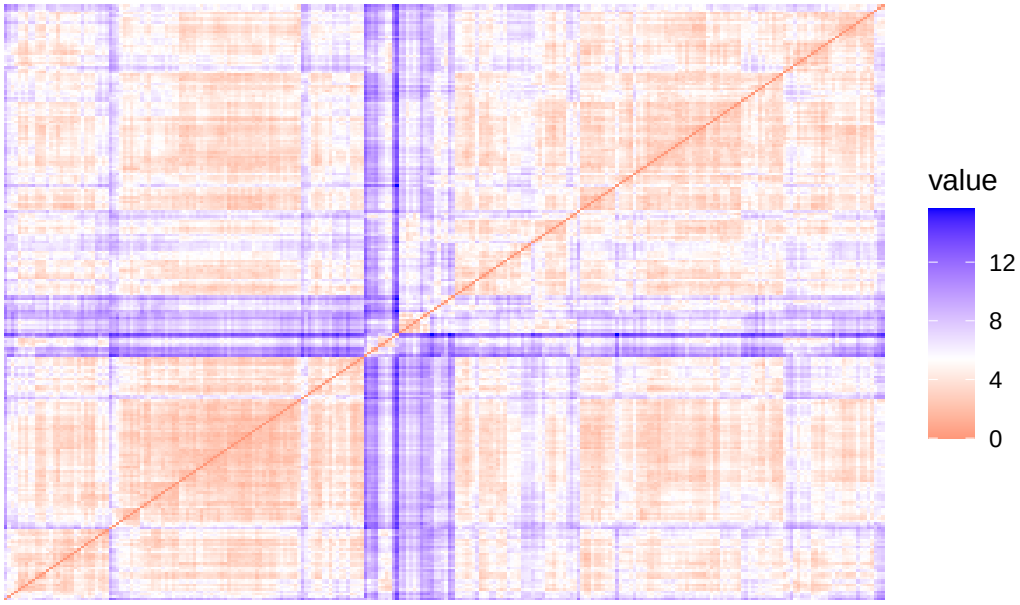
```
set.seed(2025)

# K-means
km.res1 <- kmeans(df, 3)
fviz_cluster(list(data = df, cluster = km.res1$cluster), ellipse.type = "norm", geom = "poin
```




### Visaulize Pairwise Distance

```
fviz_dist(dist(df), show_labels = FALSE) +
  labs(title = "Distance Matrix - Texas County Data")
```

```
Warning: `aes_string()` was deprecated in ggplot2 3.0.0.
i Please use tidy evaluation idioms with `aes()`.
i See also `vignette("ggplot2-in-packages")` for more information.
i The deprecated feature was likely used in the factoextra package.
  Please report the issue at <https://github.com/kassambara/factoextra/issues>.
```




**Compare Clustering Methods**

```
set.seed(2025)
cl_methods <- c("hierarchical", "kmeans", "pam", "model")

internal_valid <- clValid(df,
                          nClust = 2:6,
                          clMethods = cl_methods,
                          validation = "internal")
```

```
Warning in clValid(df, nClust = 2:6, clMethods = cl_methods, validation =
"internal"): rownames for data not specified, using 1:nrow(data)
```

```
summary(internal_valid)
```

```
Clustering Methods:
 hierarchical kmeans pam model

Cluster sizes:
 2 3 4 5 6
```

```
Validation Measures:
                                 2        3        4        5        6

hierarchical Connectivity    7.6480   7.6480  10.5770  15.2349  18.3067
             Dunn            0.2691   0.2691   0.2691   0.2691   0.2739
             Silhouette      0.5266   0.5061   0.3660   0.2761   0.2397
kmeans       Connectivity   74.0845  92.5718 143.8512 170.4325 196.5548
             Dunn            0.0887   0.1238   0.1063   0.1155   0.1155
             Silhouette      0.2351   0.2137   0.1366   0.1278   0.1269
pam          Connectivity  113.7246 144.7548 178.4647 241.9690 251.6361
             Dunn            0.1010   0.1010   0.0970   0.0825   0.0825
             Silhouette      0.1142   0.1213   0.1032   0.0672   0.0730
model        Connectivity   86.5734 156.5853 184.6230 228.0595 215.5524
             Dunn            0.0955   0.0861   0.0932   0.0880   0.0957
             Silhouette      0.2394   0.0719   0.0603   0.0462   0.0734

Optimal Scores:

             Score  Method       Clusters
Connectivity 7.6480 hierarchical 2
Dunn         0.2739 hierarchical 6
Silhouette   0.5266 hierarchical 2
```

```r
stab_valid <- clValid(df,
                      nClust = 2:6,
                      clMethods = cl_methods,
                      validation = "stability")
```

```
Warning in clValid(df, nClust = 2:6, clMethods = cl_methods, validation =
"stability"): rownames for data not specified, using 1:nrow(data)
```

```r
optimalScores(stab_valid)
```

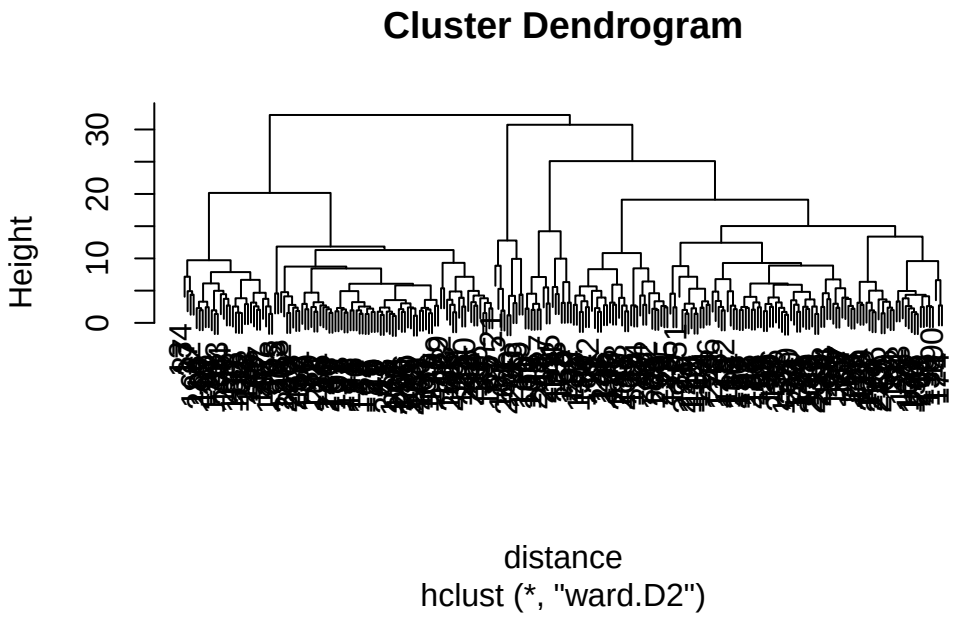
```
          Score       Method Clusters
APN 0.006738438 hierarchical        2
AD  4.434315620          pam        6
ADM 0.095130577 hierarchical        3
FOM 0.854884876          pam        6
```

### Hierarchical Clustering

```
Xsc <- scale(df)

set.seed(100)

m <- c( "average", "single", "complete", "ward")
names(m) <- c( "average", "single", "complete", "ward")
ac <- function(x) {
  agnes(Xsc, method = x)$ac
  }
purrr::map_dbl(m, ac)
```

```
  average    single  complete      ward
0.7752391 0.6501157 0.8350339 0.9200759
```

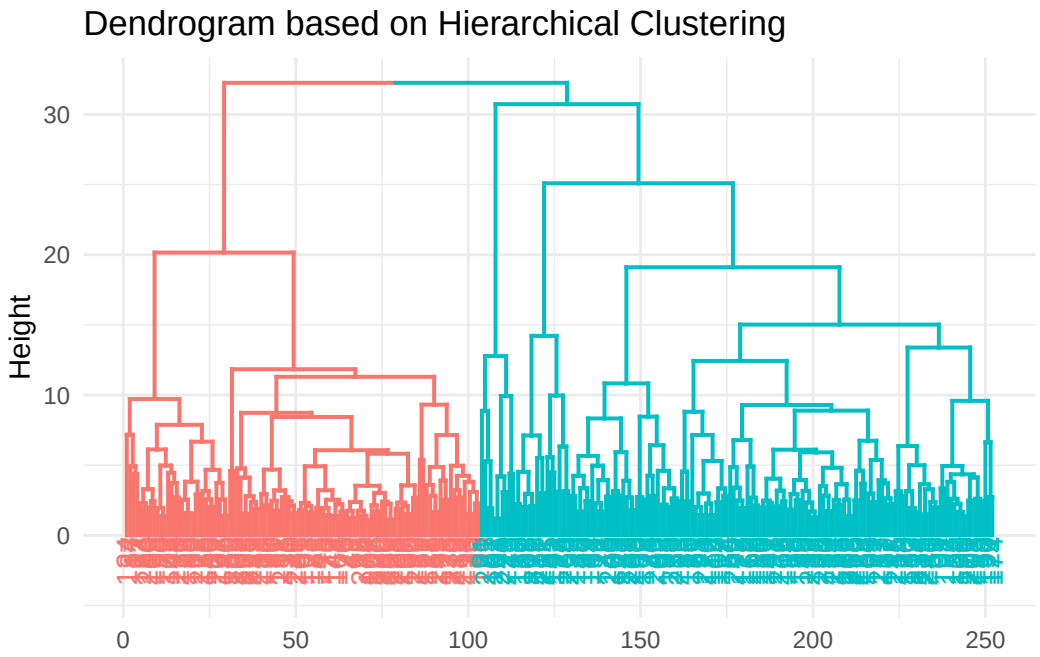
```
distance <- dist(Xsc, method = "euclidean")
hc_complete <- hclust(distance, method = 'ward.D2')
plot(hc_complete)
```




```
plot <- fviz_dend(hc_complete, k = 2)
```

```
Warning: The `<scale>` argument of `guides()` cannot be `FALSE`. Use "none" instead as
of ggplot2 3.3.4.
i The deprecated feature was likely used in the factoextra package.
  Please report the issue at <https://github.com/kassambara/factoextra/issues>.
```

```
plot +
  theme_minimal() +
  labs(title = "Dendrogram based on Hierarchical Clustering")
```




```
sub_grp <- cutree(hc_complete, 2)
```

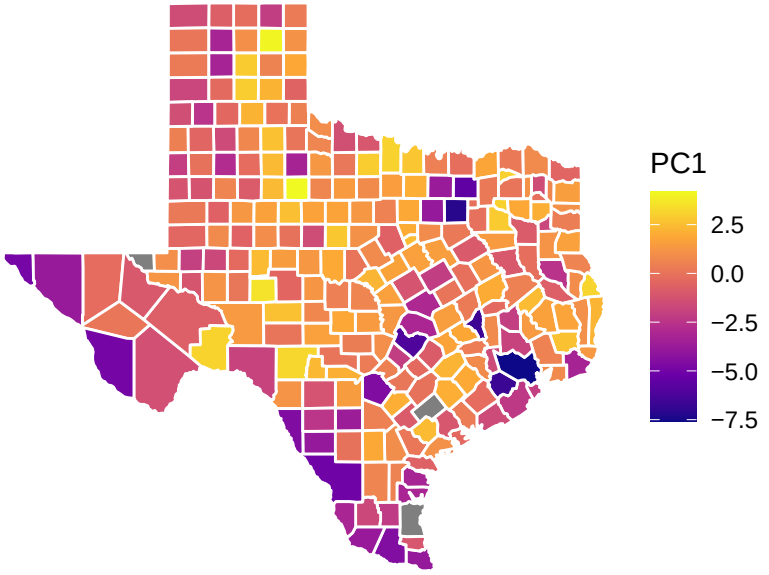
## Maps for PCA and Clustering

### PCA Map

```
rows_used_pca <- complete.cases(data %>% select(-NAME))


PC1_full <- rep(NA, nrow(data))
PC2_full <- rep(NA, nrow(data))
```

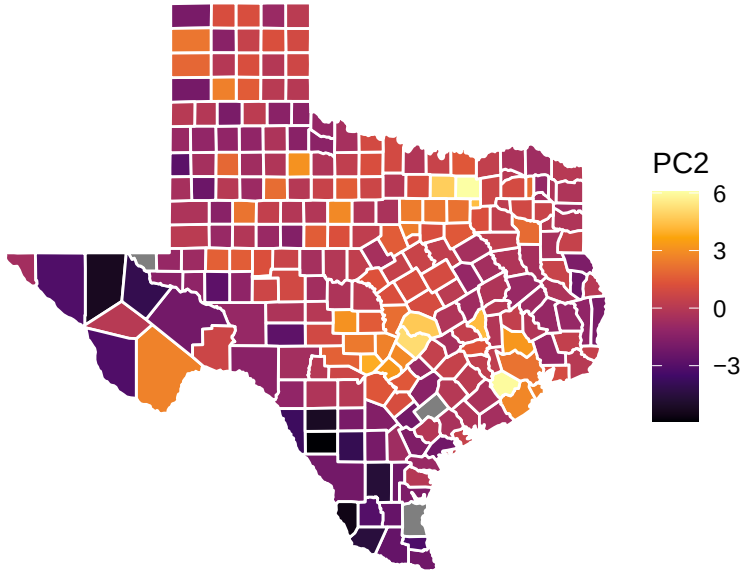
```r
PC1_full[rows_used_pca] <- pc_scores$PC1
PC2_full[rows_used_pca] <- pc_scores$PC2

pca_df <- data %>%
  mutate(
    PC1 = PC1_full,
    PC2 = PC2_full,
    county = tolower(NAME),
    county = gsub(",.*", "", county),
    county = gsub(" county$", "", county),
    county = trimws(county)
  )

# Merge with map
merged_map_pca <- county_map %>%
  left_join(pca_df, by = c("subregion" = "county"))

# PC
ggplot(merged_map_pca, aes(long, lat, group = group, fill = PC1)) +
  geom_polygon(color = "white") +
  coord_map() +
  scale_fill_viridis_c(option = "plasma") +
  labs(
    title = "PCA Component 1 Across Texas Counties",
    fill = "PC1"
  ) +
  theme_void()
```

PCA Component 1 Across Texas Counties




```
ggplot(merged_map_pca, aes(long, lat, group = group, fill = PC2)) +
  geom_polygon(color = "white") +
  coord_map() +
  scale_fill_viridis_c(option = "inferno") +
  labs(
    title = "PCA Component 2 Across Texas Counties",
    fill = "PC2"
  ) +
  theme_void()
```

PCA Component 2 Across Texas Counties




**Cluster Map**

```
# Full-length cluster vector to match data
cluster_vec <- rep(NA, nrow(data))

# Insert clusters for rows used in k-means
cluster_vec[rows_used] <- km.res1$cluster

# Attach the clusters and also clean county names
cluster_df <- data %>%
  mutate(
    cluster = cluster_vec,
    county = tolower(NAME),
    county = gsub(",.*", "", county),
    county = gsub(" county$", "", county),
    county = trimws(county)
  )

# Texas counties
county_map <- map_data("county") %>%
  filter(region == "texas")
```

```r
# Merge cluster info with map
merged_map_cluster <- county_map %>%
  left_join(cluster_df, by = c("subregion" = "county"))

# Plot cluster map
ggplot(merged_map_cluster, aes(long, lat, group = group, fill = factor(cluster))) +
  geom_polygon(color = "white") +
  coord_map() +
  scale_fill_viridis_d(option = "plasma") +
  labs(
    title = "K-Means Clusters of Texas Counties Based on Socioeconomic Indicators",
    fill = "Cluster"
  ) +
  theme_void()
```