ChicagoLeadPoisoining

2023-11-30

```r
# Read the CSV file
data <- read.csv('/Users/tejaswiniviswanath/Downloads/DPA/Chicago Lead Poisoning Dataset - Community are
```

```r
str(data)
```

```
## 'data.frame':    77 obs. of  43 variables:
##  $ Name                                      : chr  "Rogers Park" "Norwood Park" "Jefferson Park" "Fo
##  $ Population                                : int  55454 41069 26201 19579 17522 48549 63038 51911 4
##  $ Uninsured_Resident_2017_2021              : num  7504 1737 2113 1395 1835 ...
##  $ Num_Of_Children_With_EBLL_2022            : int  39 1 1 2 6 19 8 8 2 1 ...
##  $ Num_Of_Children_With_EBLL_2021            : int  30 1 1 2 7 20 12 13 3 1 ...
##  $ Num_Of_Children_With_EBLL_2020            : int  26 2 5 3 5 24 13 13 5 1 ...
##  $ Num_Of_Children_With_EBLL_2019            : int  37 1 9 0 5 35 16 17 5 2 ...
##  $ Num_Of_Children_With_EBLL_2018            : int  37 0 7 0 6 29 23 19 3 4 ...
##  $ Num_Of_Children_With_EBLL_2017            : int  37 6 7 0 7 29 20 13 4 0 ...
##  $ Social_Vulnerability_Index_2020           : num  80.6 40.2 43.9 23.8 73.5 ...
##  $ Social_Vulnerability_Index_2018           : num  74.3 28.2 41.1 17.6 64.8 ...
##  $ Social_Vulnerability_Index_2016           : num  73.9 32.1 39.1 19.3 68.4 ...
##  $ Particulate_.Concentration_.2021          : num  9.76 10.33 10.25 10.15 10.1 ...
##  $ Particulate_.Concentration_.2020          : num  8.98 9.41 9.36 9.28 9.24 ...
##  $ Particulate_.Concentration_.2019          : num  9.43 9.79 9.76 9.69 9.68 ...
##  $ Particulate_.Concentration_.2018          : num  12.5 12.8 12.8 12.7 12.8 ...
##  $ Particulate_.Concentration_.2017          : num  11 11.2 11.3 11.2 11.3 ...
##  $ Traffic_Intensity_2021                    : num  958 1778 2212 2185 826 ...
##  $ Traffic_Intensity_2020                    : num  445 1500 1906 1927 474 ...
##  $ Traffic_Intensity_2019                    : num  433 1486 1917 1920 464 ...
##  $ Traffic_Intensity_2018                    : num  15.8 1146.2 1482.8 1873.1 263.7 ...
##  $ Traffic_Intensity_2017                    : num  15.7 1115.1 1471.2 1837.7 249.8 ...
##  $ High_School_Grad_Rate_.2017_2021          : num  88.4 92.8 92.5 94.8 86.1 ...
##  $ High_School_Grad_Rate_Female_.2017_2021   : num  89.5 92.5 92.6 93.9 84.2 ...
##  $ High_School_Grad_Rate_Male_.2017_2021     : num  87.3 93 92.3 95.8 88.7 ...
##  $ College_Grad_.Rate_.2017_2021             : num  47.9 44.5 41.6 59.7 47.4 ...
##  $ College_Grad_.Rate_F_2017_2021            : num  52.5 45.3 43.4 62.5 49 ...
##  $ College_Grad_.Rate_M_2017_2021            : num  43.7 43.7 39.9 56.9 45.9 ...
##  $ Preschool_Enrol_.2017_2021                : num  50.2 50.4 64.5 59.9 56.5 ...
##  $ Preschool_Enrol_.F_2017_2021              : num  48.4 40.5 46.9 60.2 39.8 ...
##  $ Preschool_Enrol_.M_2017_2021              : num  57.6 55.4 76.7 64.9 65.8 ...
##  $ Unemployement_Rate_2017_2021              : num  5.09 4.46 5.73 4.39 3.58 ...
##  $ Unemployement_Rate_Young_Adults_2017_2021: num  5.92 5.56 6.07 3.51 4.35 ...
##  $ Hardship_Index_.2017_2021                 : num  57.3 30.3 38.1 16 59.9 ...
##  $ Median_Household_Income_.2017_2021        : num  51703 96763 84746 132774 63710 ...
##  $ Medicaid_Coverage_2017_2021               : num  23.44 10.28 14.54 7.83 24.27 ...
##  $ Per_Capita_Income_2017_2021               : num  33202 48392 39217 64378 34841 ...
##  $ Poverty_Rate_2017_2021                    : num  21.46 6.86 8.87 3.33 15.11 ...
```

```
## $ Population_2017_2021                      : num  55627 42502 27156 20293 18794 ...
## $ Population_Infants_.2017_2021             : num  3386 2779 1578 1488 1488 ...
## $ Population_Juveniles._2017_2021           : num  6126 6927 4189 3694 2427 ...
## $ Population_Females_Infants_.2017_2021     : num  1661 1268 804 594 843 ...
## $ Population_Males_Infants_2017_2021        : num  1725 1511 774 894 645 ...
```

```r
summary(data)
```

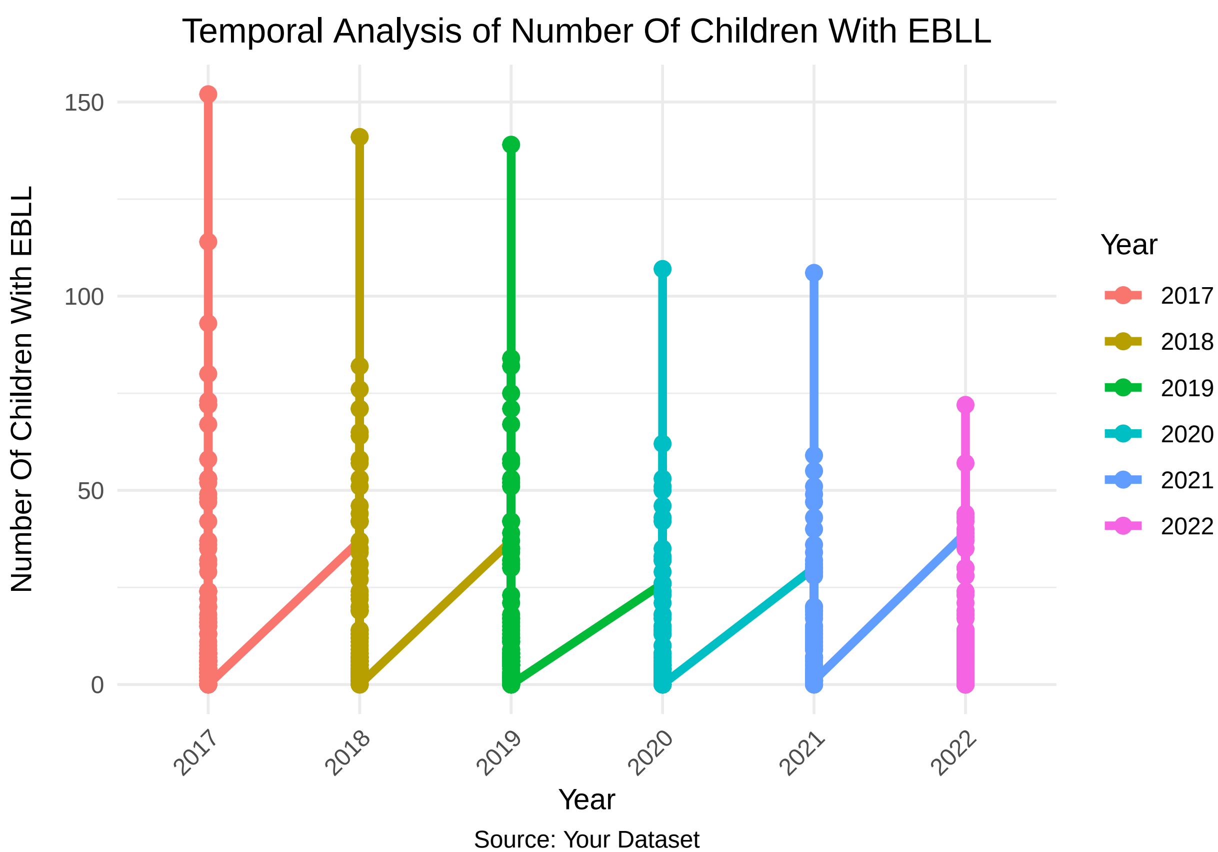
```
##      Name             Population     Uninsured_Resident_2017_2021
##  Length:77          Min.   :  2514   Min.   :   77.96
##  Class :character   1st Qu.: 18633   1st Qu.: 1394.53
##  Mode  :character   Median : 29899   Median : 2447.45
##                     Mean   : 35571   Mean   : 3447.41
##                     3rd Qu.: 45141   3rd Qu.: 4407.00
##                     Max.   :103048   Max.   :13630.80
##  Num_Of_Children_With_EBLL_2022 Num_Of_Children_With_EBLL_2021
##  Min.   : 0.00                  Min.   :  0.00
##  1st Qu.: 2.00                  1st Qu.:  2.00
##  Median : 6.00                  Median :  7.00
##  Mean   :12.84                  Mean   : 14.58
##  3rd Qu.:18.00                  3rd Qu.: 19.00
##  Max.   :72.00                  Max.   :106.00
##  Num_Of_Children_With_EBLL_2020 Num_Of_Children_With_EBLL_2019
##  Min.   :  0.00                 Min.   :  0.00
##  1st Qu.:  2.00                 1st Qu.:  3.00
##  Median :  5.00                 Median :  8.00
##  Mean   : 13.94                 Mean   : 19.84
##  3rd Qu.: 18.00                 3rd Qu.: 31.00
##  Max.   :107.00                 Max.   :139.00
##  Num_Of_Children_With_EBLL_2018 Num_Of_Children_With_EBLL_2017
##  Min.   :  0.00                 Min.   :  0.0
##  1st Qu.:  3.00                 1st Qu.:  4.0
##  Median :  9.00                 Median : 10.0
##  Mean   : 20.94                 Mean   : 22.9
##  3rd Qu.: 31.00                 3rd Qu.: 32.0
##  Max.   :141.00                 Max.   :152.0
##  Social_Vulnerability_Index_2020 Social_Vulnerability_Index_2018
##  Min.   : 8.162                  Min.   : 5.107
##  1st Qu.:52.297                  1st Qu.:49.970
##  Median :73.495                  Median :69.839
##  Mean   :65.010                  Mean   :62.142
##  3rd Qu.:82.319                  3rd Qu.:81.301
##  Max.   :90.800                  Max.   :93.237
##  Social_Vulnerability_Index_2016 Particulate_.Concentration_.2021
##  Min.   : 7.915                  Min.   : 9.765
##  1st Qu.:52.341                  1st Qu.:10.261
##  Median :72.999                  Median :10.373
##  Mean   :63.964                  Mean   :10.372
##  3rd Qu.:83.216                  3rd Qu.:10.536
##  Max.   :91.532                  Max.   :10.772
##  Particulate_.Concentration_.2020 Particulate_.Concentration_.2019
##  Min.   :8.982                    Min.   : 9.428
##  1st Qu.:9.343                    1st Qu.: 9.750
##  Median :9.418                    Median : 9.827
```

```
##  Mean   :9.419                   Mean   : 9.817
##  3rd Qu.:9.514                   3rd Qu.: 9.902
##  Max.   :9.706                   Max.   :10.055
##  Particulate_.Concentration_.2018 Particulate_.Concentration_.2017
##  Min.   :12.54                   Min.   :11.05
##  1st Qu.:12.96                   1st Qu.:11.45
##  Median :13.22                   Median :11.63
##  Mean   :13.11                   Mean   :11.55
##  3rd Qu.:13.27                   3rd Qu.:11.68
##  Max.   :13.30                   Max.   :11.71
##  Traffic_Intensity_2021 Traffic_Intensity_2020 Traffic_Intensity_2019
##  Min.   :  169.4        Min.   :   75.18       Min.   :   75.16
##  1st Qu.:  670.5        1st Qu.:  471.20       1st Qu.:  462.67
##  Median : 1068.3        Median :  888.48       Median :  875.20
##  Mean   : 1916.5        Mean   : 1660.26       Mean   : 1664.02
##  3rd Qu.: 2299.6        3rd Qu.: 1905.77       3rd Qu.: 1913.86
##  Max.   :11697.0        Max.   :12519.20       Max.   :12426.87
##  Traffic_Intensity_2018 Traffic_Intensity_2017 High_School_Grad_Rate_.2017_2021
##  Min.   :  15.75        Min.   :  15.71        Min.   :61.76
##  1st Qu.: 294.91        1st Qu.: 301.30        1st Qu.:79.56
##  Median : 789.16        Median : 793.38        Median :86.87
##  Mean   :1349.94        Mean   :1345.64        Mean   :85.11
##  3rd Qu.:1482.85        3rd Qu.:1471.21        3rd Qu.:92.49
##  Max.   :9826.27        Max.   :9600.16        Max.   :98.12
##  High_School_Grad_Rate_Female_.2017_2021 High_School_Grad_Rate_Male_.2017_2021
##  Min.   :59.75                           Min.   :60.82
##  1st Qu.:80.55                           1st Qu.:77.09
##  Median :87.53                           Median :86.51
##  Mean   :85.44                           Mean   :84.73
##  3rd Qu.:92.22                           3rd Qu.:92.58
##  Max.   :98.09                           Max.   :98.21
##  College_Grad_.Rate_.2017_2021 College_Grad_.Rate_F_2017_2021
##  Min.   : 6.568                Min.   : 6.384
##  1st Qu.:16.386                1st Qu.:17.550
##  Median :28.692                Median :29.957
##  Mean   :34.567                Mean   :36.270
##  3rd Qu.:45.238                3rd Qu.:48.483
##  Max.   :85.969                Max.   :86.146
##  College_Grad_.Rate_M_2017_2021 Preschool_Enrol_.2017_2021
##  Min.   : 5.433                 Min.   : 9.727
##  1st Qu.:15.279                 1st Qu.:43.883
##  Median :25.973                 Median :56.148
##  Mean   :32.656                 Mean   :55.184
##  3rd Qu.:43.652                 3rd Qu.:66.504
##  Max.   :85.622                 Max.   :99.957
##  Preschool_Enrol_.F_2017_2021 Preschool_Enrol_.M_2017_2021
##  Min.   : 0.6404              Min.   : 0.1579
##  1st Qu.: 39.3678             1st Qu.:43.4884
##  Median : 53.3766             Median :55.9757
##  Mean   : 55.1071             Mean   :55.7423
##  3rd Qu.: 72.7550             3rd Qu.:68.7167
##  Max.   : 99.9963             Max.   :99.9435
##  Unemployement_Rate_2017_2021 Unemployement_Rate_Young_Adults_2017_2021
##  Min.   : 2.039               Min.   : 1.63
```

```
##  1st Qu.: 5.483              1st Qu.: 6.07
##  Median : 9.140              Median :10.67
##  Mean   :11.128              Mean   :13.01
##  3rd Qu.:16.209              3rd Qu.:18.82
##  Max.   :32.519              Max.   :34.26
##  Hardship_Index_.2017_2021 Median_Household_Income_.2017_2021
##  Min.   : 2.458            Min.   : 18316
##  1st Qu.:39.701            1st Qu.: 41454
##  Median :72.119            Median : 59341
##  Mean   :61.637            Mean   : 64388
##  3rd Qu.:84.881            3rd Qu.: 78899
##  Max.   :97.745            Max.   :151342
##  Medicaid_Coverage_2017_2021 Per_Capita_Income_2017_2021 Poverty_Rate_2017_2021
##  Min.   : 3.694              Min.   : 13005              Min.   : 3.327
##  1st Qu.:15.708              1st Qu.: 22317              1st Qu.: 9.624
##  Median :26.494              Median : 29697              Median :16.346
##  Mean   :27.579              Mean   : 36513              Mean   :19.022
##  3rd Qu.:37.719              3rd Qu.: 41859              3rd Qu.:25.430
##  Max.   :76.572              Max.   :111614              Max.   :50.938
##  Population_2017_2021 Population_Infants_.2017_2021
##  Min.   :  2280       Min.   :  76.17
##  1st Qu.: 18794       1st Qu.:1130.32
##  Median : 29381       Median :1752.42
##  Mean   : 35524       Mean   :2118.07
##  3rd Qu.: 48110       3rd Qu.:2778.76
##  Max.   :102608       Max.   :6417.69
##  Population_Juveniles._2017_2021 Population_Females_Infants_.2017_2021
##  Min.   :  340.9                 Min.   :  43.25
##  1st Qu.: 2738.7                 1st Qu.: 554.71
##  Median : 4384.7                 Median : 819.27
##  Mean   : 5145.2                 Mean   :1046.96
##  3rd Qu.: 6647.8                 3rd Qu.:1414.25
##  Max.   :17230.8                 Max.   :3457.25
##  Population_Males_Infants_2017_2021
##  Min.   :  18.08
##  1st Qu.: 511.20
##  Median : 919.86
##  Mean   :1071.10
##  3rd Qu.:1437.65
##  Max.   :3275.72
```

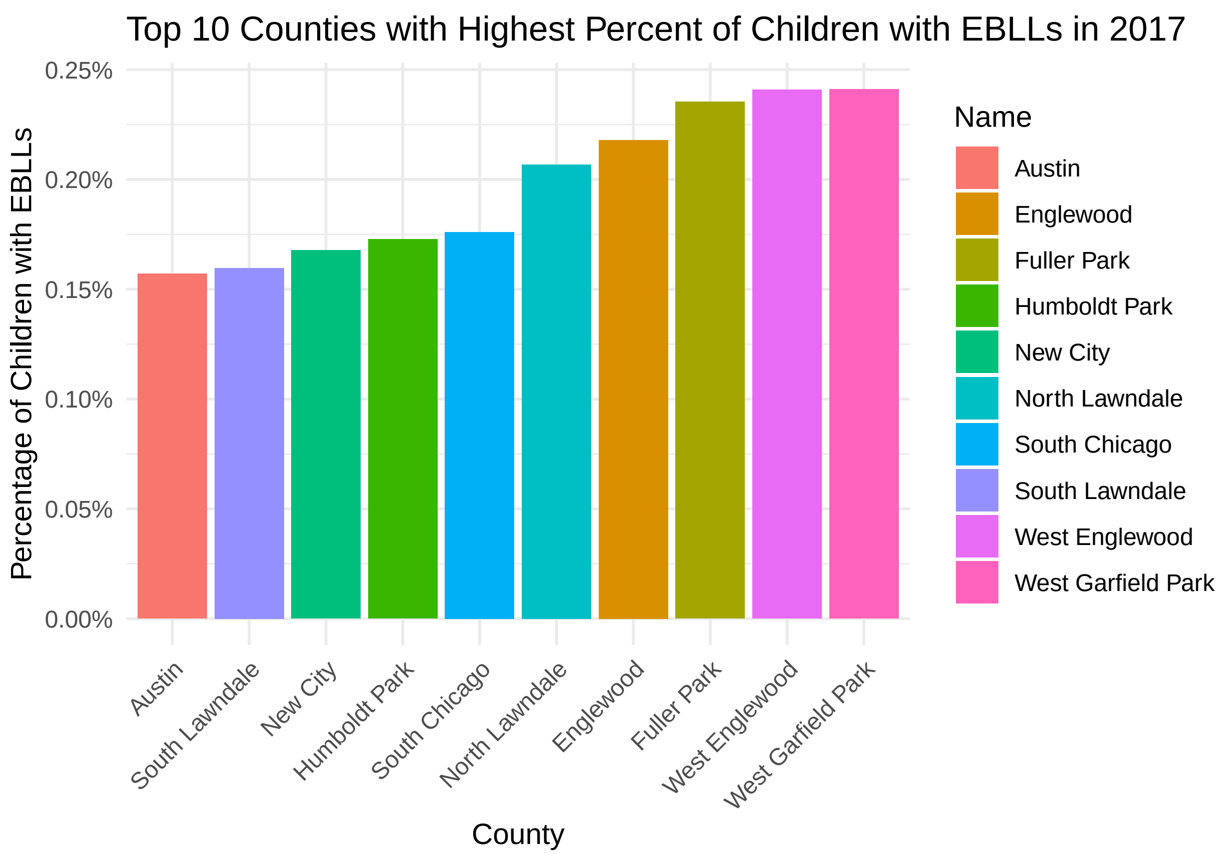
```r
# Load required libraries
library(ggplot2)
library(tidyr)

# Convert data to long format for easier plotting
data_long <- pivot_longer(data, cols = starts_with("Num_Of_Children_With_EBLL_"), names_to = "Year", val

# Convert 'Year' to numeric (assuming it's stored as a character)
data_long$Year <- as.factor(gsub("Num_Of_Children_With_EBLL_", "", data_long$Year))

# Generate a dynamic color palette based on the number of unique years
line_colors <- scales::hue_pal()(length(unique(data_long$Year)))
```

```r
# Plot the time series with enhanced aesthetics
ggplot(data_long, aes(x = Year, y = Num_Of_Children_With_EBLL_Level, group = 1)) +
  geom_line(aes(color = Year), linewidth = 1.5) +
  geom_point(aes(color = Year), size = 2.5) +
  scale_color_manual(values = line_colors) +
  labs(title = "Temporal Analysis of Number Of Children With EBLL",
       x = "Year",
       y = "Number Of Children With EBLL",
       caption = "Source: Your Dataset") +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5),
        plot.caption = element_text(hjust = 0.5),
        axis.title.y = element_text(margin = margin(r = 10)),
        axis.text.x = element_text(angle = 45, hjust = 1, vjust = 1))
```



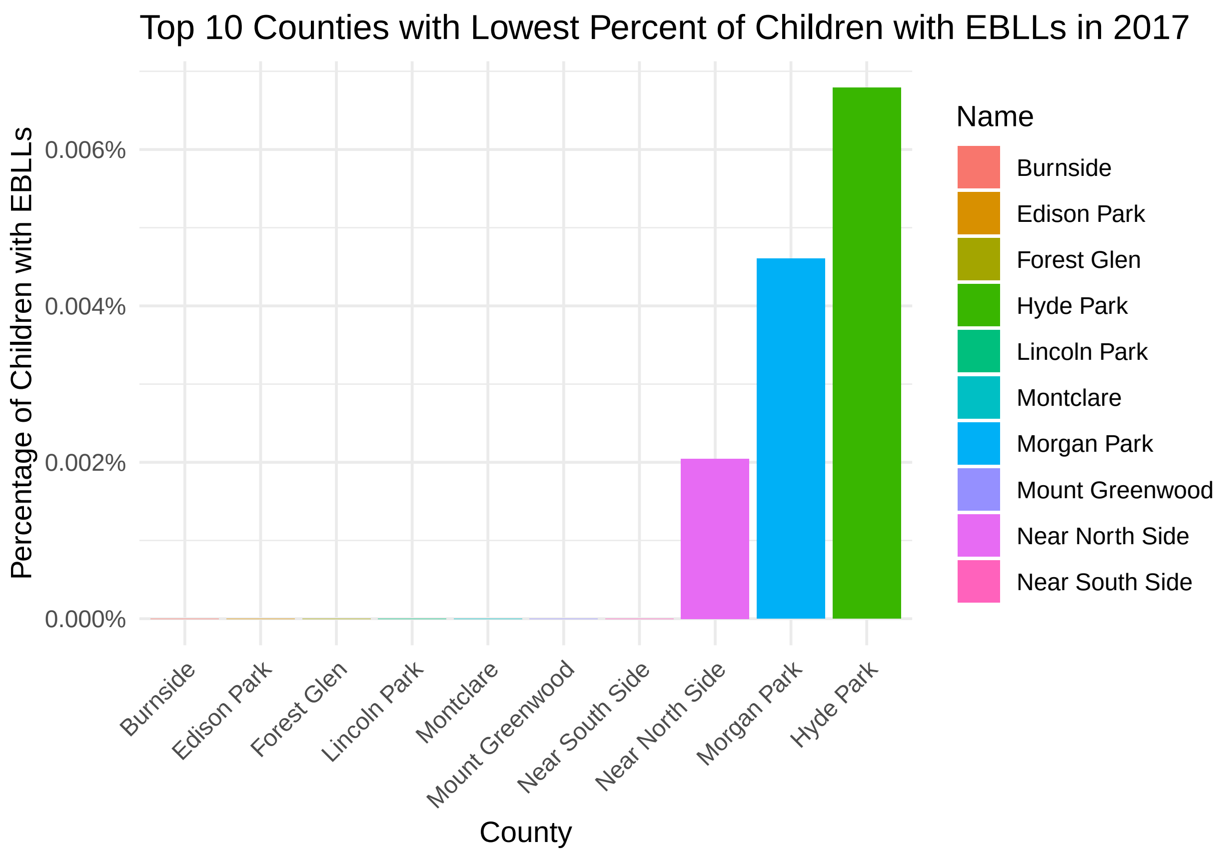

```r
library(ggplot2)

# Calculate the percentage of children with elevated blood lead levels for each county in 2017
data$Percent_Children_Elevated_BLL_2017 <-
  (data$Num_Of_Children_With_EBLL_2017 / data$Population) * 100

# Select the top 10 counties
top_10_counties <- head(data[order(-data$Percent_Children_Elevated_BLL_2017), ], 10)

# Create a ggplot bar plot with unique colors for each county
```

```r
ggplot(top_10_counties, aes(x = reorder(Name, Percent_Children_Elevated_BLL_2017),
                             y = Percent_Children_Elevated_BLL_2017, fill = Name)) +
  geom_bar(stat = "identity") +
  labs(title = "Top 10 Counties with Highest Percent of Children with EBLLs in 2017",
       x = "County",
       y = "Percentage of Children with EBLLs") +
  scale_y_continuous(labels = scales::percent_format(scale = 1)) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  scale_fill_discrete()  # This ensures unique colors for each county
```




```r
library(ggplot2)

# Assuming your dataset is named 'data'
# Calculate the percentage of children with elevated blood lead levels for each county in 2017
data$Percent_Children_Elevated_BLL_2017 <-
  (data$Num_Of_Children_With_EBLL_2017 / data$Population) * 100

# Select the bottom 10 counties (lowest percentage)
bottom_10_counties <- head(data[order(data$Percent_Children_Elevated_BLL_2017), ], 10)

# Create a ggplot bar plot with unique colors for each county
ggplot(bottom_10_counties, aes(x = reorder(Name, Percent_Children_Elevated_BLL_2017),
                                y = Percent_Children_Elevated_BLL_2017, fill = Name)) +
  geom_bar(stat = "identity") +
```

```r
  labs(title = "Top 10 Counties with Lowest Percent of Children with EBLLs in 2017",
       x = "County",
       y = "Percentage of Children with EBLLs") +
  scale_y_continuous(labels = scales::percent_format(scale = 1)) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  scale_fill_discrete()  # This ensures unique colors for each county
```



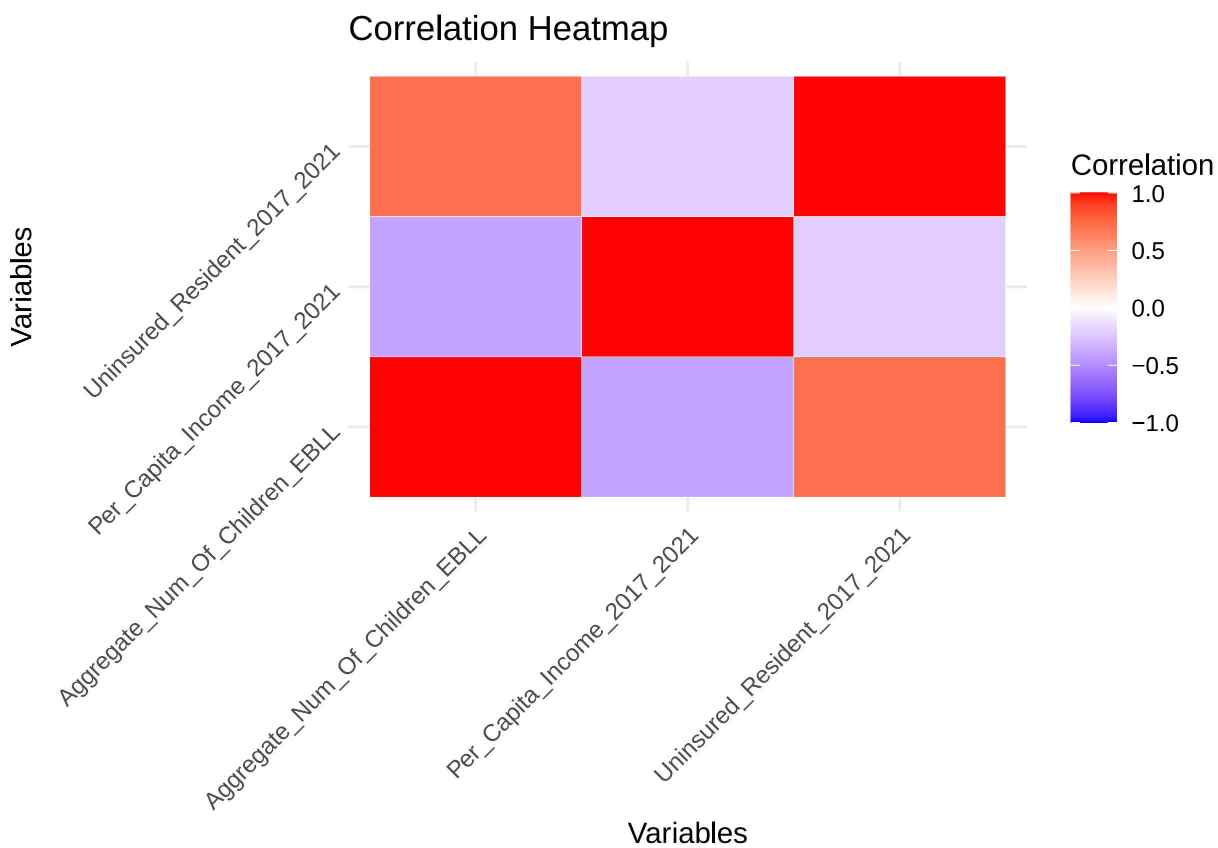

```r
# Load required libraries
library(dplyr)
```

```
##
## 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
```

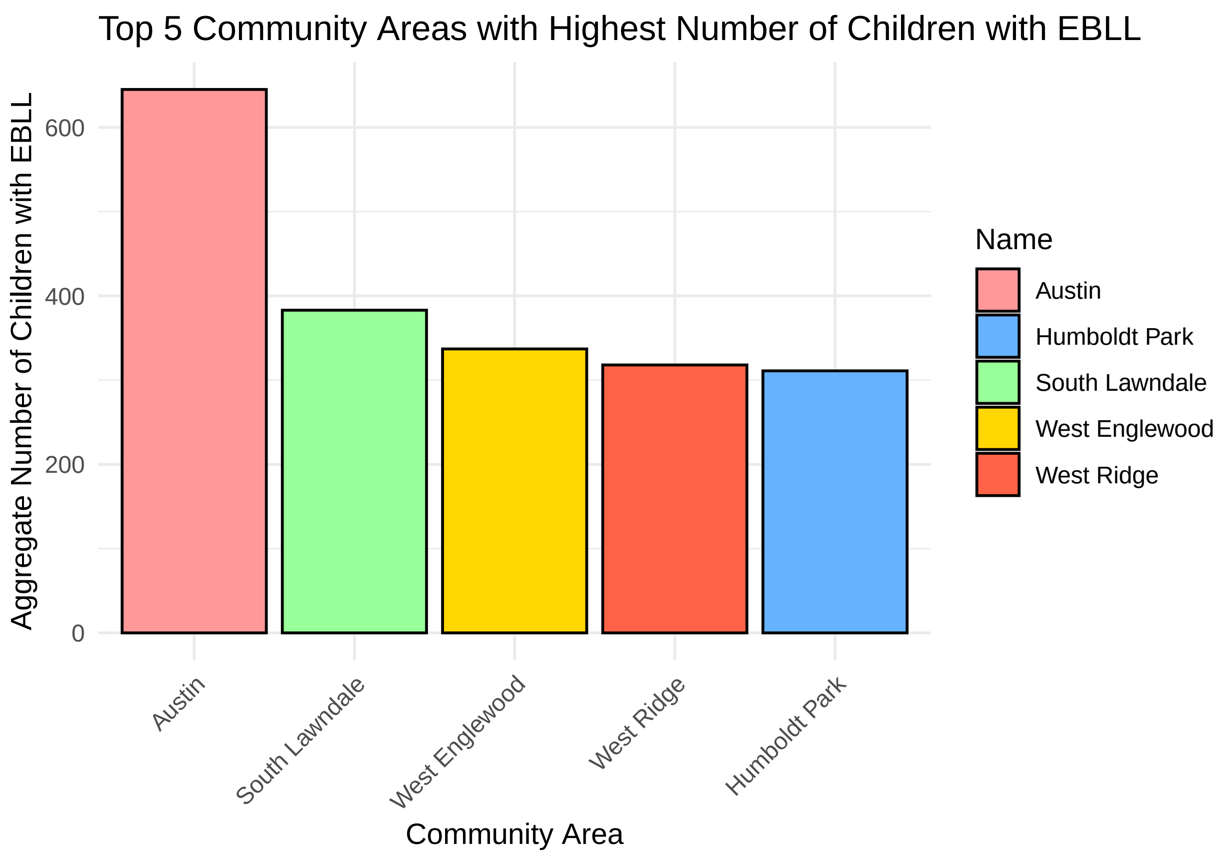
```r
library(ggplot2)
library(reshape2)
```

```
##
## Attaching package: 'reshape2'

## The following object is masked from 'package:tidyr':
##
##     smiths
```

```r
# Calculate the aggregate of 'Num_Of_Children_With_EBLL_' columns over the years
data$Aggregate_Num_Of_Children_EBLL <- rowSums(data[, grep("Num_Of_Children_With_EBLL_", colnames(data)

# Select relevant columns for correlation
correlation_data <- data %>%
  select(Aggregate_Num_Of_Children_EBLL, Per_Capita_Income_2017_2021, Uninsured_Resident_2017_2021)

# Calculate the correlation matrix
correlation_matrix <- cor(correlation_data)

# Melt the correlation matrix for ggplot
melted_correlation <- melt(correlation_matrix)

# Create a visually pleasing heatmap using ggplot2
ggplot(melted_correlation, aes(Var1, Var2, fill = value)) +
  geom_tile(color = "white") +
  scale_fill_gradient2(low = "blue", mid = "white", high = "red", midpoint = 0, limit = c(-1,1), space =
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1),
        axis.text.y = element_text(angle = 45, hjust = 1)) +
  labs(title = "Correlation Heatmap",
       x = "Variables",
       y = "Variables")
```




```r
# Assuming 'data' is your dataset
library(dplyr)
library(ggplot2)

# Create a new variable for the aggregate number of children with EBLL
data$Aggregate_Num_Of_Children_EBLL <- rowSums(data[, c("Num_Of_Children_With_EBLL_2017", "Num_Of_Child

# Find the top 5 community areas with the highest aggregate number of children with EBLL
top_5_areas <- data %>%
  arrange(desc(Aggregate_Num_Of_Children_EBLL)) %>%
  slice_head(n = 5) %>%
  select(Name, Aggregate_Num_Of_Children_EBLL)

# Custom color palette
custom_colors <- c("#FF9999", "#66B2FF", "#99FF99", "#FFD700", "#FF6347")

# Visualize the top 5 community areas
ggplot(top_5_areas, aes(x = reorder(Name, -Aggregate_Num_Of_Children_EBLL), y = Aggregate_Num_Of_Childre
  geom_bar(stat = "identity", color = "black") +
  scale_fill_manual(values = custom_colors) +
  labs(title = "Top 5 Community Areas with Highest Number of Children with EBLL",
       x = "Community Area",
       y = "Aggregate Number of Children with EBLL") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```




```
# Assuming 'data' is your dataset
# Replace 'data' with the actual name of your dataset if different

library(dplyr)
library(ggplot2)
library(ggpubr)
library(viridis)
```

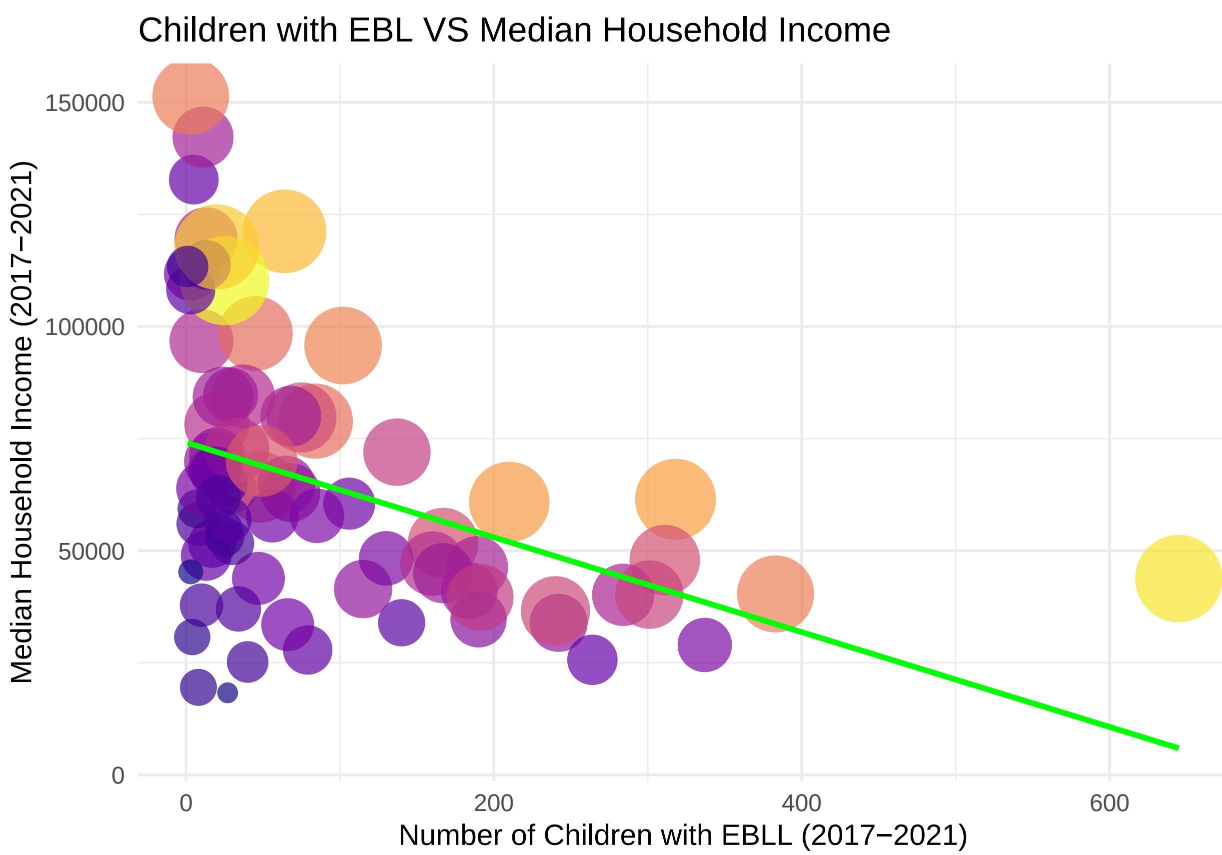
```
## Loading required package: viridisLite
```

```
# Define a custom color palette with bright and muted colors
color_palette <- viridis(5)

# Scatter plot with trend line
scatter_plot <- ggplot(data, aes(x = Aggregate_Num_Of_Children_EBLL, y = Median_Household_Income_.2017_2
  geom_point(alpha = 0.7) +
  geom_smooth(method = "lm", se = FALSE, color = "green", size = 1) +  # Add trend line
  scale_size_continuous(range = c(3, 15)) +
  labs(title = "Children with EBL VS Median Household Income",
       x = "Number of Children with EBLL (2017-2021)",
       y = "Median Household Income (2017-2021)",
       size = "Population (2017-2021)",
       color = "Population (2017-2021)") +
  theme_minimal() +
  scale_color_viridis_c(option = "plasma") +  # Apply viridis color palette for continuous variable
  theme(legend.position = "none")  # Remove legend for size
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
# Print scatter plot with trend line
print(scatter_plot)
```

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




```
# Linear regression analysis
linear_model <- lm(Median_Household_Income_.2017_2021 ~ Aggregate_Num_Of_Children_EBLL + Population_201

# Print regression analysis results
regression_results <- summary(linear_model)

# Extract p-value and R^2 value
p_value <- regression_results$coefficients[2, 4]
r_squared <- summary(linear_model)$r.squared

# Print p-value and R^2 value
cat("P-value:", p_value, "\n")
```

```
## P-value: 1.951442e-11
```

```r
cat("R-squared:", r_squared, "\n")
```

```
## R-squared: 0.5253801
```

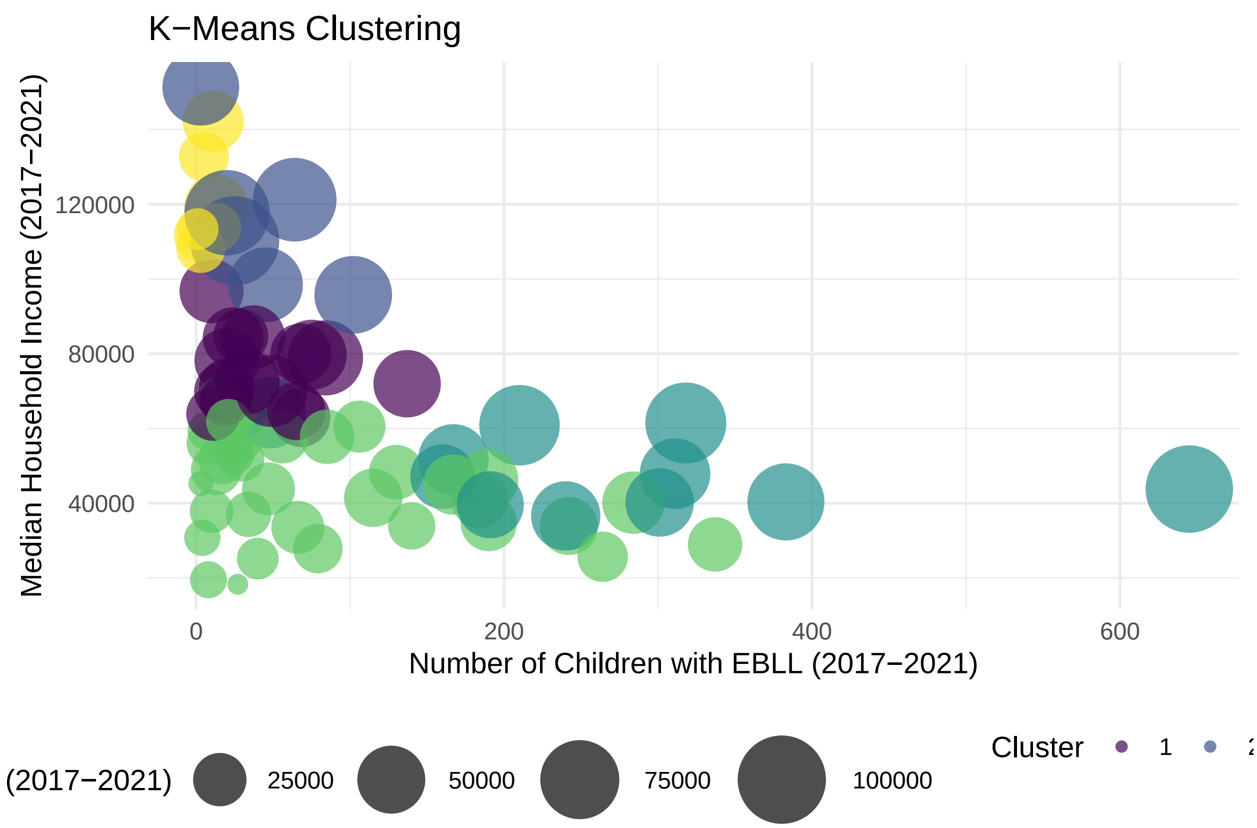
```r
# Perform k-means clustering
k <- 5  # Specify the number of clusters
kmeans_model <- kmeans(data[, c("Aggregate_Num_Of_Children_EBLL", "Median_Household_Income_.2017_2021",
data$cluster <- as.factor(kmeans_model$cluster)

# Print K-Means clustering results
cat("K-means clustering with", k, "clusters\n")
```

```
## K-means clustering with 5 clusters
```

```r
cat("Cluster means:\n", kmeans_model$centers, "\n")
```

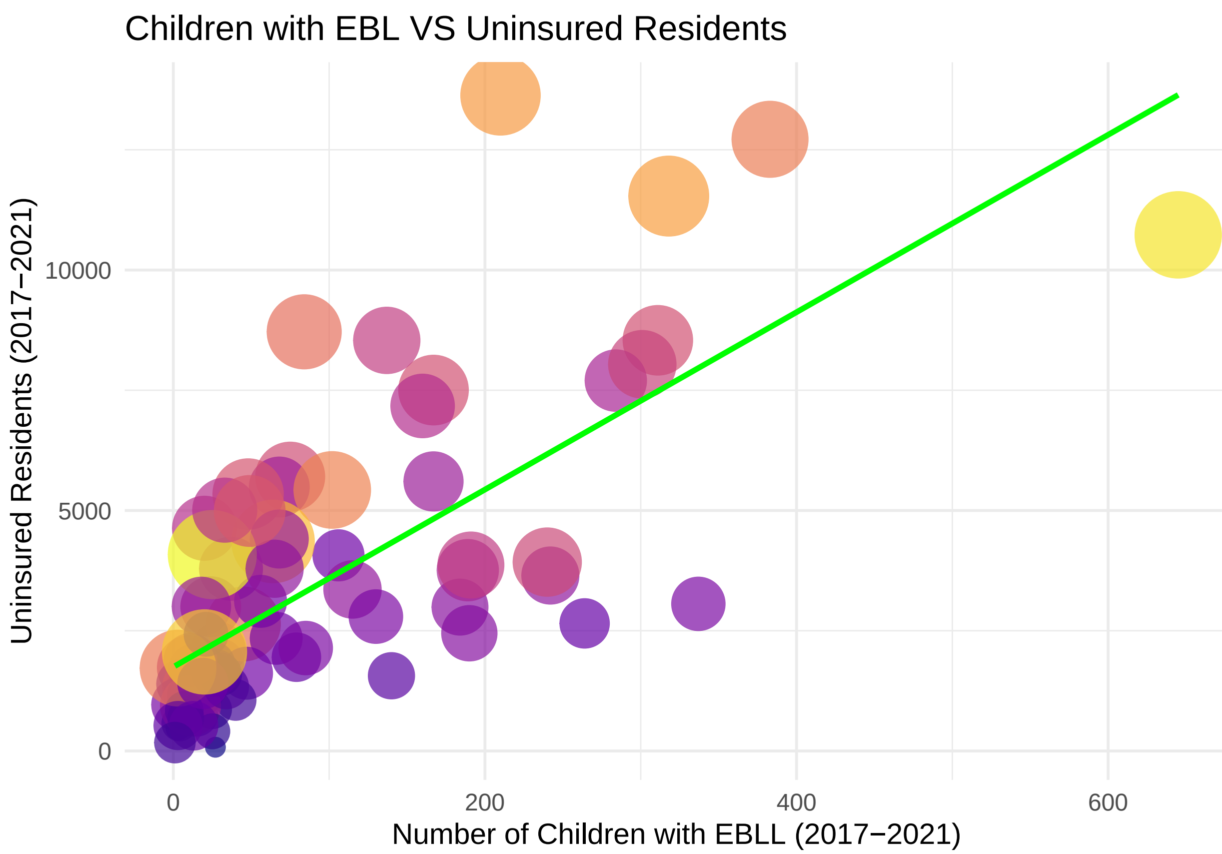
```
## Cluster means:
##  45.11765 43.16667 270.3636 84.72222 7 75395.51 115789.3 48533.61 44609.09 120235.2 39849 81196.29 6
```

```r
# Plot K-Means Clustering
kmeans_plot <- ggplot(data, aes(x = Aggregate_Num_Of_Children_EBLL, y = Median_Household_Income_.2017_2
  geom_point(alpha = 0.7) +
  scale_size_continuous(range = c(3, 15)) +
  labs(title = "K-Means Clustering",
       x = "Number of Children with EBLL (2017-2021)",
       y = "Median Household Income (2017-2021)",
       size = "Population (2017-2021)",
       color = "Cluster") +
  theme_minimal() +
  scale_color_manual(values = color_palette) +  # Apply custom color palette for k-means clusters
  theme(legend.position = "bottom")

# Print K-Means clustering plot
print(kmeans_plot)
```




```
library(dplyr)
library(ggplot2)
library(ggpubr)
library(viridis)

# Define a custom color palette with bright and muted colors
color_palette <- viridis(5)

# Scatter plot with trend line
scatter_plot <- ggplot(data, aes(x = Aggregate_Num_Of_Children_EBLL, y = Uninsured_Resident_2017_2021, s
  geom_point(alpha = 0.7) +
  geom_smooth(method = "lm", se = FALSE, color = "green", size = 1) +  # Add trend line
  scale_size_continuous(range = c(3, 15)) +
  labs(title = "Children with EBL VS Uninsured Residents",
       x = "Number of Children with EBLL (2017-2021)",
       y = "Uninsured Residents (2017-2021)",
       size = "Population (2017-2021)",
       color = "Population (2017-2021)") +
  theme_minimal() +
  scale_color_viridis_c(option = "plasma") +  # Apply viridis color palette for continuous variable
  theme(legend.position = "none")  # Remove legend for size

# Print scatter plot with trend line
print(scatter_plot)
```

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




```
# Linear regression analysis
linear_model <- lm(Uninsured_Resident_2017_2021 ~ Aggregate_Num_Of_Children_EBLL + Population_2017_2021

# Print regression analysis results
regression_results <- summary(linear_model)

# Extract p-value and R^2 value
p_value <- regression_results$coefficients[2, 4]
r_squared <- summary(linear_model)$r.squared

# Print p-value and R^2 value
cat("P-value:", p_value, "\n")
```

```
## P-value: 1.996655e-09
```

```
cat("R-squared:", r_squared, "\n")
```

```
## R-squared: 0.6905649
```

```
# Perform k-means clustering
k <- 5  # Specify the number of clusters
kmeans_model <- kmeans(data[, c("Aggregate_Num_Of_Children_EBLL", "Uninsured_Resident_2017_2021", "Popul
data$cluster <- as.factor(kmeans_model$cluster)
```

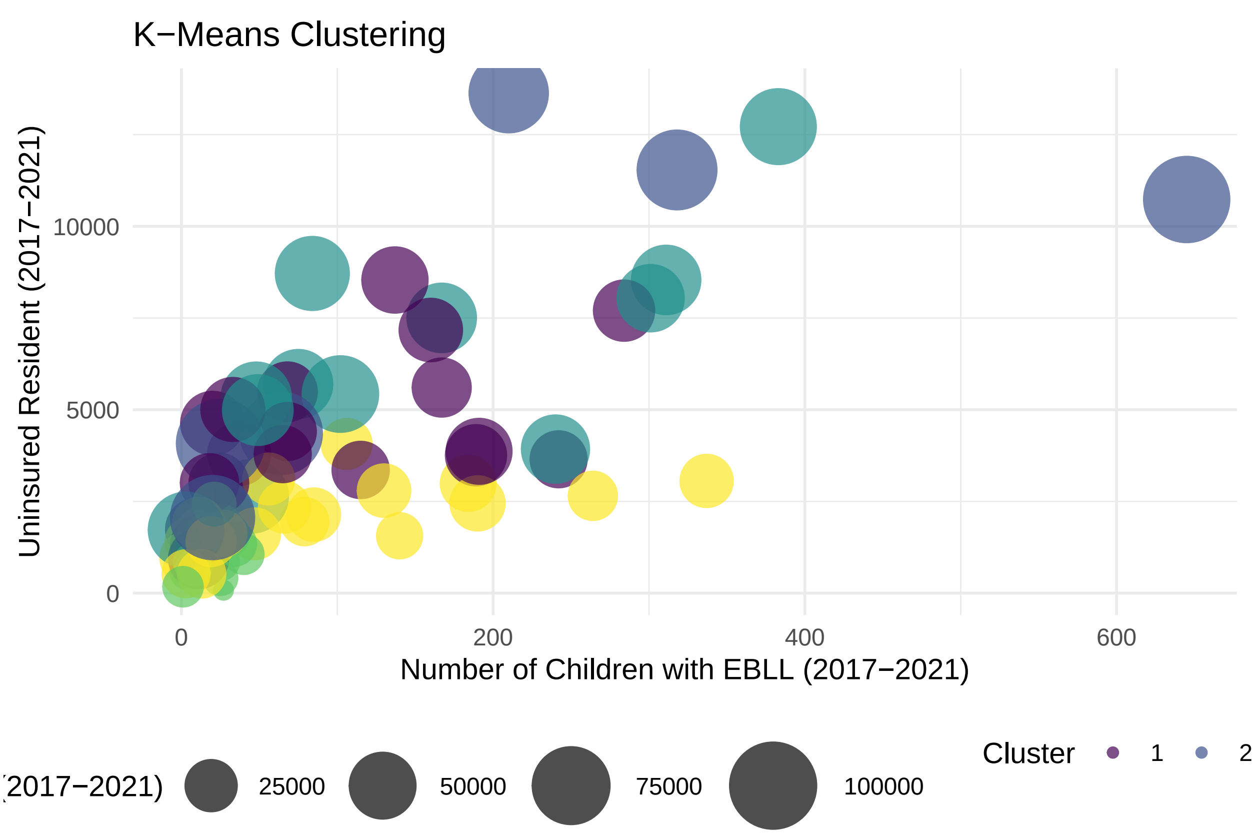
```r
# Print K-Means clustering results
cat("K-means clustering with", k, "clusters\n")
```

```
## K-means clustering with 5 clusters
```

```r
cat("Cluster means:\n", kmeans_model$centers, "\n")
```

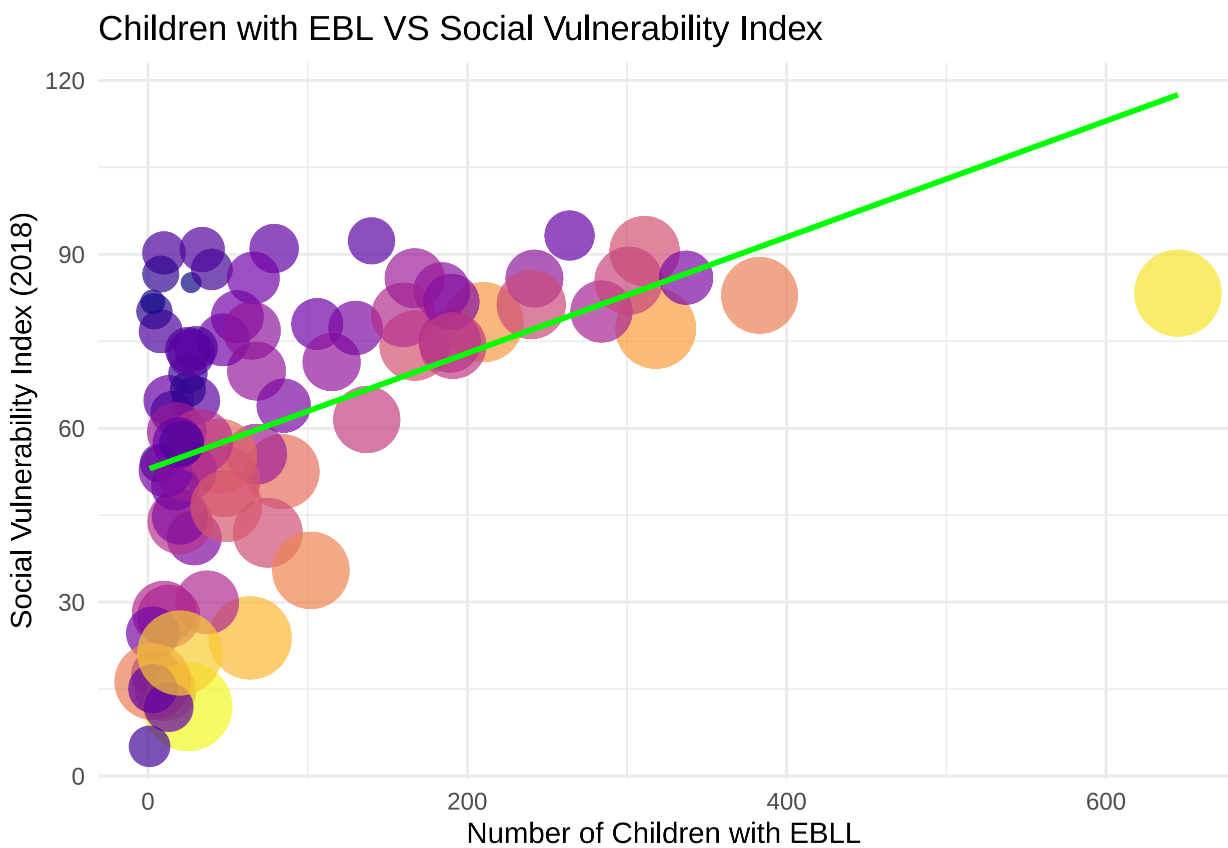
```
## Cluster means:
##  97.47368 213.6667 150.6667 17.6875 78.08333 4249.981 7733.179 6272.482 967.2506 1981.488 39169 8951
```

```r
# Plot K-Means Clustering
kmeans_plot <- ggplot(data, aes(x = Aggregate_Num_Of_Children_EBLL, y = Uninsured_Resident_2017_2021, s
  geom_point(alpha = 0.7) +
  scale_size_continuous(range = c(3, 15)) +
  labs(title = "K-Means Clustering",
       x = "Number of Children with EBLL (2017-2021)",
       y = "Uninsured Resident (2017-2021)",
       size = "Population (2017-2021)",
       color = "Cluster") +
  theme_minimal() +
  scale_color_manual(values = color_palette) +  # Apply custom color palette for k-means clusters
  theme(legend.position = "bottom")

# Print K-Means clustering plot
print(kmeans_plot)
```



```r
library(dplyr)
library(ggplot2)
library(ggpubr)
library(viridis)
library(cluster)

# Define a custom color palette with bright and muted colors
color_palette <- viridis(5)

# Scatter plot with trend line
scatter_plot <- ggplot(data, aes(x = Aggregate_Num_Of_Children_EBLL, y = Social_Vulnerability_Index_2018
  geom_point(alpha = 0.7) +
  geom_smooth(method = "lm", se = FALSE, color = "green", size = 1) +  # Add trend line
  scale_size_continuous(range = c(3, 15)) +
  labs(title = "Children with EBL VS Social Vulnerability Index",
       x = "Number of Children with EBLL",
       y = "Social Vulnerability Index (2018)",
       size = "Population (2017-2021)",
       color = "Population (2017-2021)") +
  theme_minimal() +
  scale_color_viridis_c(option = "plasma") +  # Apply viridis color palette for continuous variable
  theme(legend.position = "none")  # Remove legend for size

# Print scatter plot with trend line
print(scatter_plot)
```

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




```r
# Linear regression analysis
linear_model <- lm(Social_Vulnerability_Index_2018 ~ Aggregate_Num_Of_Children_EBLL + Population_2017_2

# Print regression analysis results
regression_results <- summary(linear_model)

# Extract p-value and R^2 value
p_value <- regression_results$coefficients[2, 4]
r_squared <- summary(linear_model)$r.squared

# Print p-value and R^2 value
cat("P-value:", p_value, "\n")
```

```
## P-value: 3.955259e-12
```

```r
cat("R-squared:", r_squared, "\n")
```

```
## R-squared: 0.5160807
```

```r
# Perform k-means clustering
k <- 5  # Specify the number of clusters
kmeans_model <- kmeans(data[, c("Aggregate_Num_Of_Children_EBLL", "Social_Vulnerability_Index_2018", "Po
data$cluster <- as.factor(kmeans_model$cluster)
```

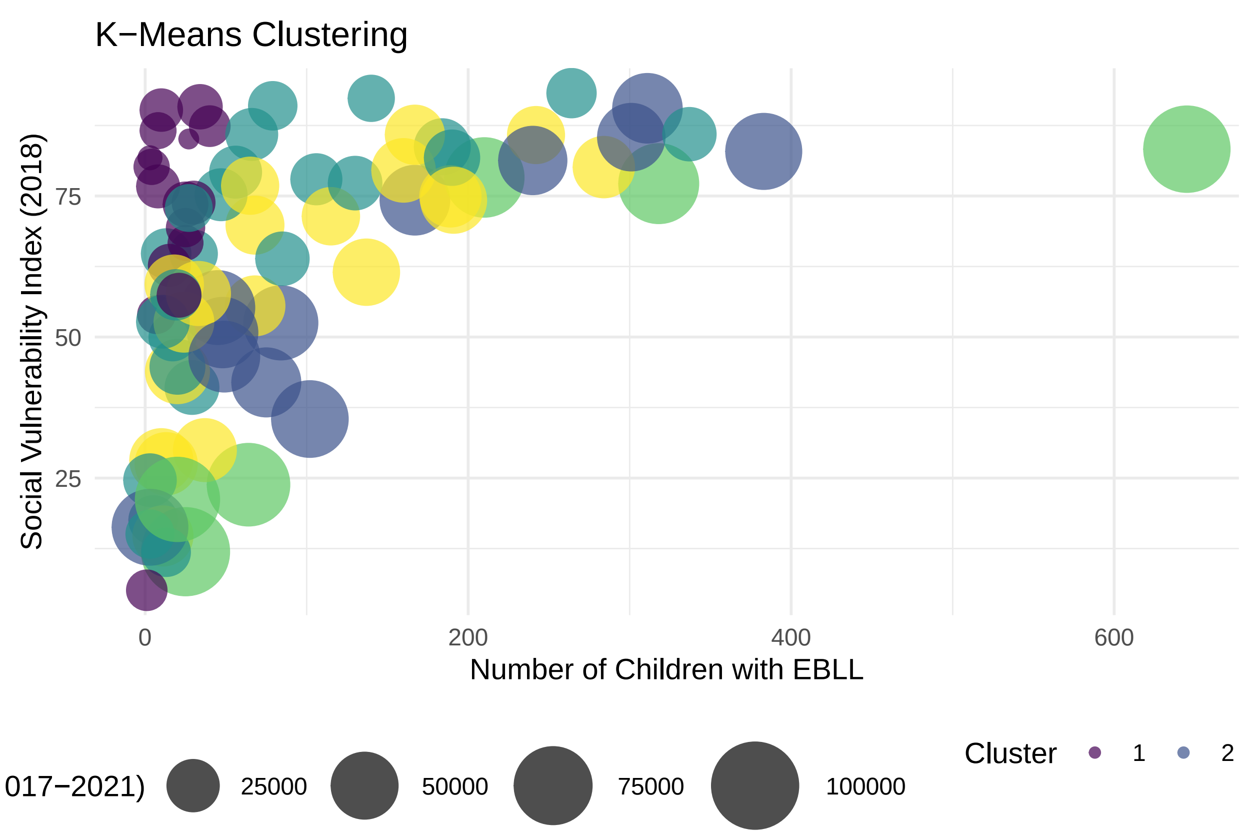
```r
# Print K-Means clustering results
cat("K-means clustering with", k, "clusters\n")
```

```
## K-means clustering with 5 clusters
```

```r
cat("Cluster means:\n", kmeans_model$centers, "\n")
```

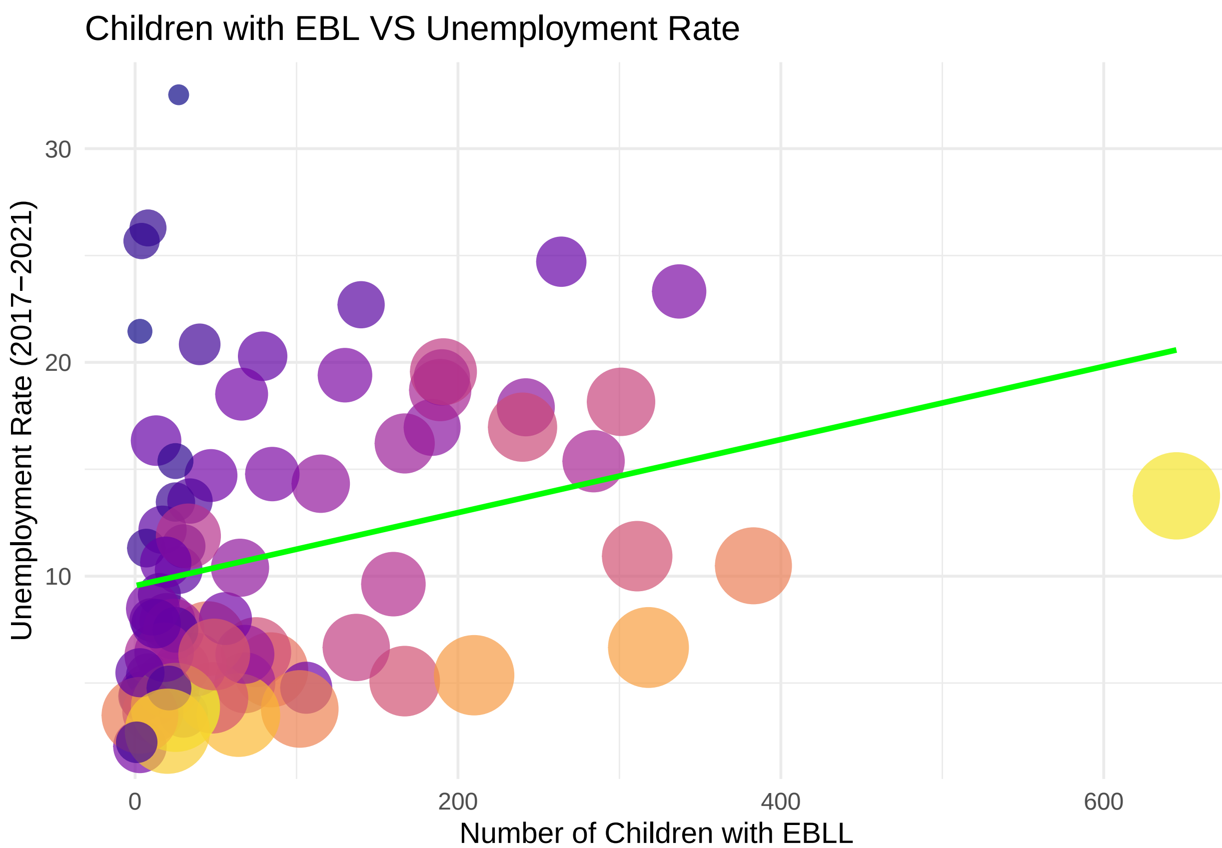
```
## Cluster means:
##  17.6875 150.6667 78.08333 213.6667 97.47368 71.33757 59.43226 62.70977 49.29484 59.45126 10518.56 60
```

```r
# Plot K-Means Clustering
kmeans_plot <- ggplot(data, aes(x = Aggregate_Num_Of_Children_EBLL, y = Social_Vulnerability_Index_2018
  geom_point(alpha = 0.7) +
  scale_size_continuous(range = c(3, 15)) +
  labs(title = "K-Means Clustering",
       x = "Number of Children with EBLL",
       y = "Social Vulnerability Index (2018)",
       size = "Population (2017-2021)",
       color = "Cluster") +
  theme_minimal() +
  scale_color_manual(values = color_palette) +  # Apply custom color palette for k-means clusters
  theme(legend.position = "bottom")

# Print K-Means clustering plot
print(kmeans_plot)
```



```r
library(dplyr)
library(ggplot2)
library(ggpubr)
library(viridis)
library(cluster)

# Define a custom color palette with bright and muted colors
color_palette <- viridis(5)

# Scatter plot with trend line
scatter_plot <- ggplot(data, aes(x = Aggregate_Num_Of_Children_EBLL, y = Unemployement_Rate_2017_2021, s
  geom_point(alpha = 0.7) +
  geom_smooth(method = "lm", se = FALSE, color = "green", size = 1) +  # Add trend line
  scale_size_continuous(range = c(3, 15)) +
  labs(title = "Children with EBL VS Unemployment Rate",
       x = "Number of Children with EBLL",
       y = "Unemployment Rate (2017-2021)",
       size = "Population (2017-2021)",
       color = "Population (2017-2021)") +
  theme_minimal() +
  scale_color_viridis_c(option = "plasma") +  # Apply viridis color palette for continuous variable
  theme(legend.position = "none")  # Remove legend for size

# Print scatter plot with trend line
print(scatter_plot)
```

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




```r
# Linear regression analysis
linear_model <- lm(Unemployement_Rate_2017_2021 ~ Aggregate_Num_Of_Children_EBLL + Population_2017_2021

# Print regression analysis results
regression_results <- summary(linear_model)

# Extract p-value and R^2 value
p_value <- regression_results$coefficients[2, 4]
r_squared <- summary(linear_model)$r.squared

# Print p-value and R^2 value
cat("P-value:", p_value, "\n")
```

```
## P-value: 7.063144e-08
```

```r
cat("R-squared:", r_squared, "\n")
```

```
## R-squared: 0.4355787
```

```r
# Perform k-means clustering
k <- 5  # Specify the number of clusters
kmeans_model <- kmeans(data[, c("Aggregate_Num_Of_Children_EBLL", "Unemployement_Rate_2017_2021", "Popul
data$cluster <- as.factor(kmeans_model$cluster)
```

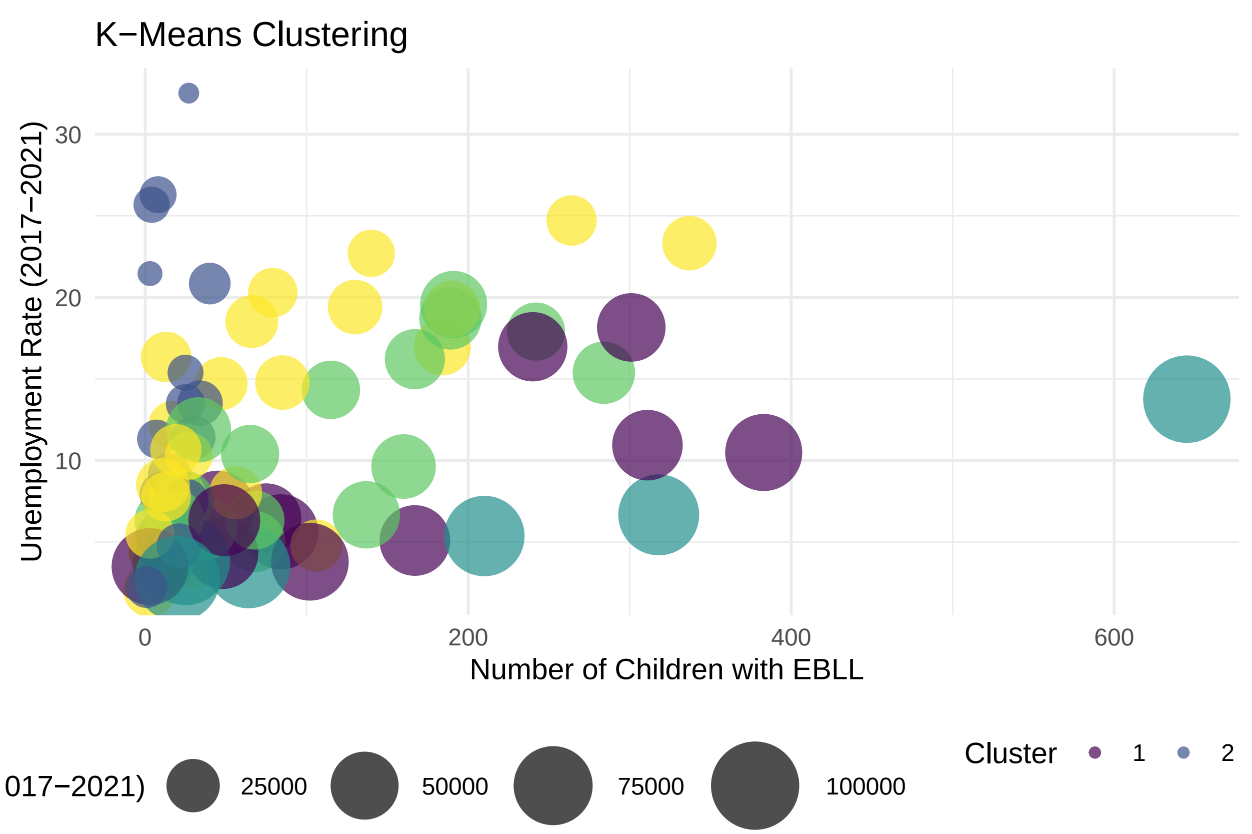
```r
# Print K-Means clustering results
cat("K-means clustering with", k, "clusters\n")
```

```
## K-means clustering with 5 clusters
```

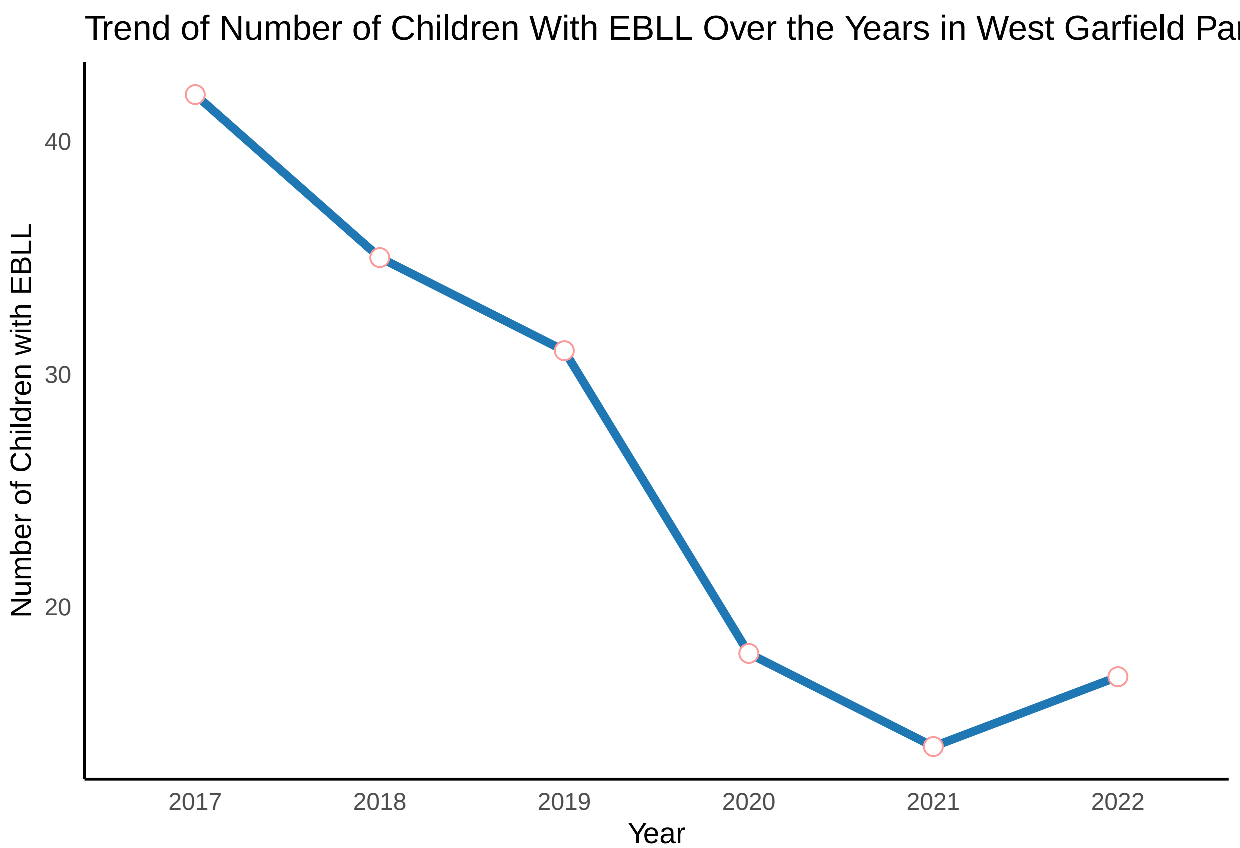
```r
cat("Cluster means:\n", kmeans_model$centers, "\n")
```

```
## Cluster means:
##  150.6667 17.6875 213.6667 97.47368 78.08333 8.224866 14.30069 5.982348 10.05994 12.59634 60522.85 1
```

```r
# Plot K-Means Clustering
kmeans_plot <- ggplot(data, aes(x = Aggregate_Num_Of_Children_EBLL, y = Unemployement_Rate_2017_2021, s
  geom_point(alpha = 0.7) +
  scale_size_continuous(range = c(3, 15)) +
  labs(title = "K-Means Clustering",
       x = "Number of Children with EBLL",
       y = "Unemployment Rate (2017-2021)",
       size = "Population (2017-2021)",
       color = "Cluster") +
  theme_minimal() +
  scale_color_manual(values = color_palette) +  # Apply custom color palette for k-means clusters
  theme(legend.position = "bottom")

# Print K-Means clustering plot
print(kmeans_plot)
```

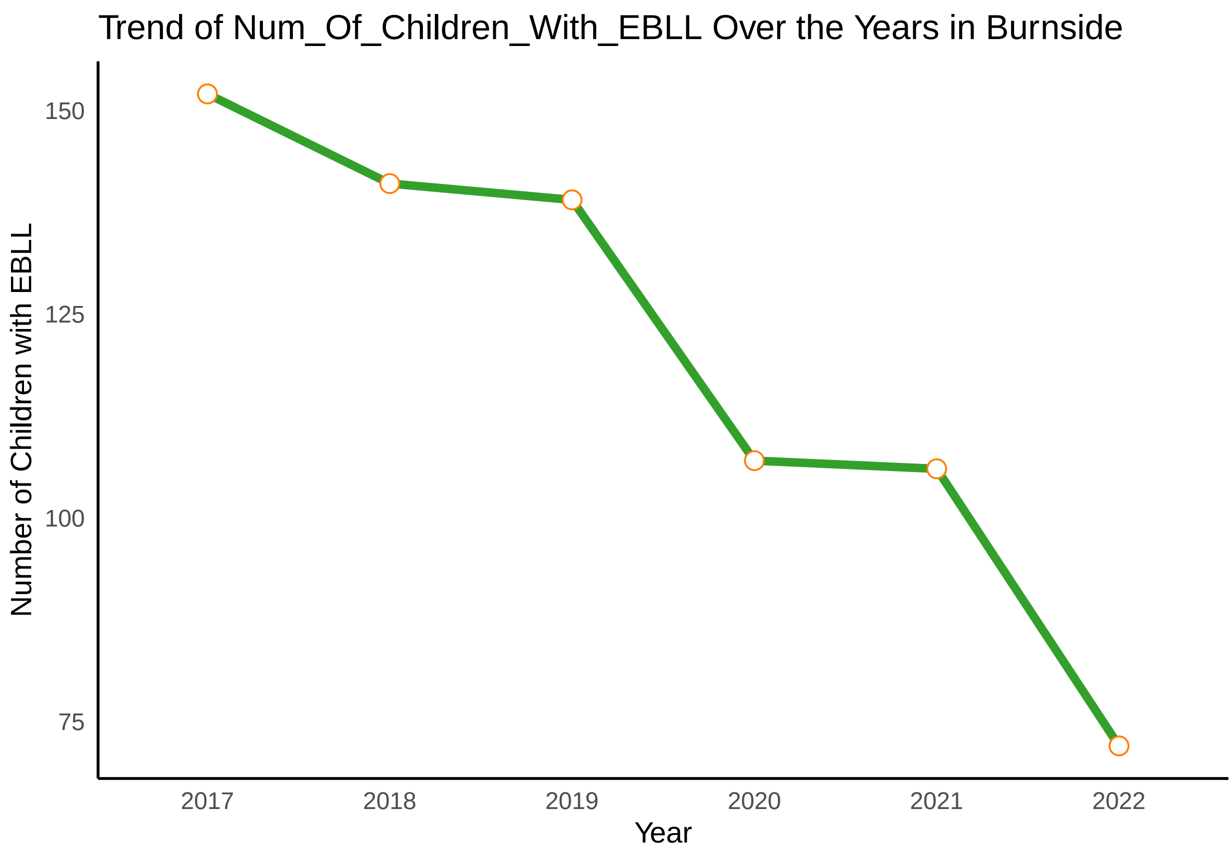


```r
library(ggplot2)

# Assuming you have columns like 'Num_Of_Children_With_EBLL_2017', 'Num_Of_Children_With_EBLL_2018', et

# Filter data for 'West Garfield Park' county
west_garfield_park_data <- data[data$Name == "West Garfield Park", ]

# Melt the data to long format for easy plotting
melted_data <- reshape2::melt(west_garfield_park_data,
                              id.vars = "Name",
                              measure.vars = c("Num_Of_Children_With_EBLL_2017",
                                               "Num_Of_Children_With_EBLL_2018",
                                               "Num_Of_Children_With_EBLL_2019",
                                               "Num_Of_Children_With_EBLL_2020",
                                               "Num_Of_Children_With_EBLL_2021",
                                               "Num_Of_Children_With_EBLL_2022"),
                              variable.name = "Year",
                              value.name = "Num_Of_Children_With_EBLL")
melted_data$Year <- gsub("Num_Of_Children_With_EBLL_", "", melted_data$Year)

# Create a time series plot
ggplot(melted_data, aes(x = as.factor(Year), y = Num_Of_Children_With_EBLL, group = 1)) +
  geom_line(color = "#1F78B4", size = 1.5) +  # Line color and thickness
  geom_point(color = "#FB9A99", size = 3, fill = "white", shape = 21) +  # Point color, size, and shape
  labs(title = "Trend of Number of Children With EBLL Over the Years in West Garfield Park County",
       x = "Year",
       y = "Number of Children with EBLL") +
  theme_minimal() +
  theme(
    panel.grid.major = element_blank(),  # Remove major grid lines
    panel.grid.minor = element_blank(),  # Remove minor grid lines
    panel.border = element_blank(),  # Remove panel border
    axis.line = element_line(colour = "black"),  # Axis line color
    legend.position = "none"  # Remove legend
  )
```




```
# Assuming your dataset is named 'data'
# Assuming you have columns like 'Num_Of_Children_With_EBLL_2017', 'Num_Of_Children_With_EBLL_2018', et

# Filter data for 'Burnside' county
burnside_data <- data[data$Name == "Austin", ]

# Melt the data to long format for easy plotting
melted_data_burnside <- reshape2::melt(burnside_data,
                                       id.vars = "Name",
                                       measure.vars = c("Num_Of_Children_With_EBLL_2017",
                                                        "Num_Of_Children_With_EBLL_2018",
                                                        "Num_Of_Children_With_EBLL_2019",
                                                        "Num_Of_Children_With_EBLL_2020",
                                                        "Num_Of_Children_With_EBLL_2021",
                                                        "Num_Of_Children_With_EBLL_2022"),
                                       variable.name = "Year",
                                       value.name = "Num_Of_Children_With_EBLL")

# Extract only the year from the variable names
melted_data_burnside$Year <- gsub("Num_Of_Children_With_EBLL_", "", melted_data_burnside$Year)

# Create a time series plot
ggplot(melted_data_burnside, aes(x = Year, y = Num_Of_Children_With_EBLL, group = 1)) +
  geom_line(color = "#33A02C", size = 1.5) +  # Line color and thickness
  geom_point(color = "#FF7F00", size = 3, fill = "white", shape = 21) +  # Point color, size, and shape
  labs(title = "Trend of Num_Of_Children_With_EBLL Over the Years in Burnside",
```

```
      x = "Year",
      y = "Number of Children with EBLL") +
  theme_minimal() +
  theme(
    panel.grid.major = element_blank(),  # Remove major grid lines
    panel.grid.minor = element_blank(),  # Remove minor grid lines
    panel.border = element_blank(),  # Remove panel border
    axis.line = element_line(colour = "black")  # Axis line color
  )
```




```
# Assuming 'historical_data' and 'data' are your datasets

# Extract numeric values from 'historical_data'
historical_data_numeric <- as.data.frame(lapply(data[, -1], as.numeric))

# Aggregate Number Of Children With EBLL columns
Num_Of_Children_With_EBLL_data <- rowSums(historical_data_numeric[, grepl("Num_Of_Children_With_EBLL_",

# Specify the columns to aggregate for each factor category
environmental_columns <- c(
  "Particulate_.Concentration_.2021",
  "Particulate_.Concentration_.2020",
  "Particulate_.Concentration_.2019",
  "Particulate_.Concentration_.2018",
  "Particulate_.Concentration_.2017",
```

```r
  "Traffic_Intensity_2021",
  "Traffic_Intensity_2020",
  "Traffic_Intensity_2019",
  "Traffic_Intensity_2018",
  "Traffic_Intensity_2017"
)

social_vulnerability_index_columns <- c(
  "Social_Vulnerability_Index_2020",
  "Social_Vulnerability_Index_2018",
  "Social_Vulnerability_Index_2016"
)

economic_columns <- c(
  "Unemployement_Rate_2017_2021",
  "Unemployement_Rate_Young_Adults_2017_2021",
  "Hardship_Index_.2017_2021",
  "Per_Capita_Income_2017_2021",
  "Poverty_Rate_2017_2021"
)

education_columns <- c(
  "High_School_Grad_Rate_.2017_2021",
  "High_School_Grad_Rate_Female_.2017_2021",
  "High_School_Grad_Rate_Male_.2017_2021",
  "College_Grad_.Rate_.2017_2021",
  "College_Grad_.Rate_F_2017_2021",
  "College_Grad_.Rate_M_2017_2021",
  "Preschool_Enrol_.2017_2021",
  "Preschool_Enrol_.F_2017_2021",
  "Preschool_Enrol_.M_2017_2021"
)

demographic_columns <- c(
  "Population",
  "Median_Household_Income_.2017_2021",
  "Per_Capita_Income_2017_2021",
  "Poverty_Rate_2017_2021"
)

# Aggregate environmental columns
aggregated_environmental_data <- data.frame(
  Particulate_Concentration = rowSums(historical_data_numeric[, grepl("Particulate_.Concentration_", col
  Traffic_Intensity = rowSums(historical_data_numeric[, grepl("Traffic_Intensity_", colnames(historical_
)

# Aggregate social vulnerability index columns
aggregated_social_vulnerability_index <- rowSums(data[, social_vulnerability_index_columns], na.rm = TRU

  economic_data <- data[, economic_columns]
  education_data <- data[, education_columns]
  demographic_data <- data[, demographic_columns]
```

```r
# Combine aggregated columns with Num_Of_Children_With_EBLL_data
combined_data <- cbind(
  Num_Of_Children_With_EBLL_data,
  aggregated_environmental_data,
  aggregated_social_vulnerability_index,
  economic_data,
  education_data,
  demographic_data
)

# Calculate the correlation matrix
correlation_matrix <- cor(combined_data, use = "complete.obs")

# Extract the correlation between Number Of Children With EBLL and aggregated factors
correlation_with_lead_poisoning <- correlation_matrix["Num_Of_Children_With_EBLL_data", -1]

# Find the top 5 factors with the highest correlation
top_5_factors <- names(sort(abs(correlation_with_lead_poisoning), decreasing = TRUE))[1:5]

# Print the correlation values
print("Correlation with Number Of Children With EBLL:")
```

```
## [1] "Correlation with Number Of Children With EBLL:"
```

```r
print(correlation_with_lead_poisoning)
```

```
##                 Particulate_Concentration
##                                 0.1313802
##                         Traffic_Intensity
##                                -0.1840913
##     aggregated_social_vulnerability_index
##                                 0.4898648
##              Unemployement_Rate_2017_2021
##                                 0.2855272
## Unemployement_Rate_Young_Adults_2017_2021
##                                 0.2823984
##                 Hardship_Index_.2017_2021
##                                 0.4506639
##               Per_Capita_Income_2017_2021
##                                -0.3963630
##                    Poverty_Rate_2017_2021
##                                 0.3382006
##          High_School_Grad_Rate_.2017_2021
##                                -0.4378644
##   High_School_Grad_Rate_Female_.2017_2021
##                                -0.3866806
##     High_School_Grad_Rate_Male_.2017_2021
##                                -0.4870875
##            College_Grad_.Rate_.2017_2021
##                                -0.4091323
##             College_Grad_.Rate_F_2017_2021
##                                -0.4037447
```

```
##            College_Grad_.Rate_M_2017_2021
##                                -0.4076867
##                Preschool_Enrol_.2017_2021
##                                -0.2327377
##              Preschool_Enrol_.F_2017_2021
##                                -0.1559582
##              Preschool_Enrol_.M_2017_2021
##                                -0.2047869
##                                Population
##                                 0.4603063
##        Median_Household_Income_.2017_2021
##                                -0.4077827
##               Per_Capita_Income_2017_2021
##                                -0.3963630
##                    Poverty_Rate_2017_2021
##                                 0.3382006
```

```r
# Print the top 5 factors
print("Top 5 Factors with Highest Correlation:")
```

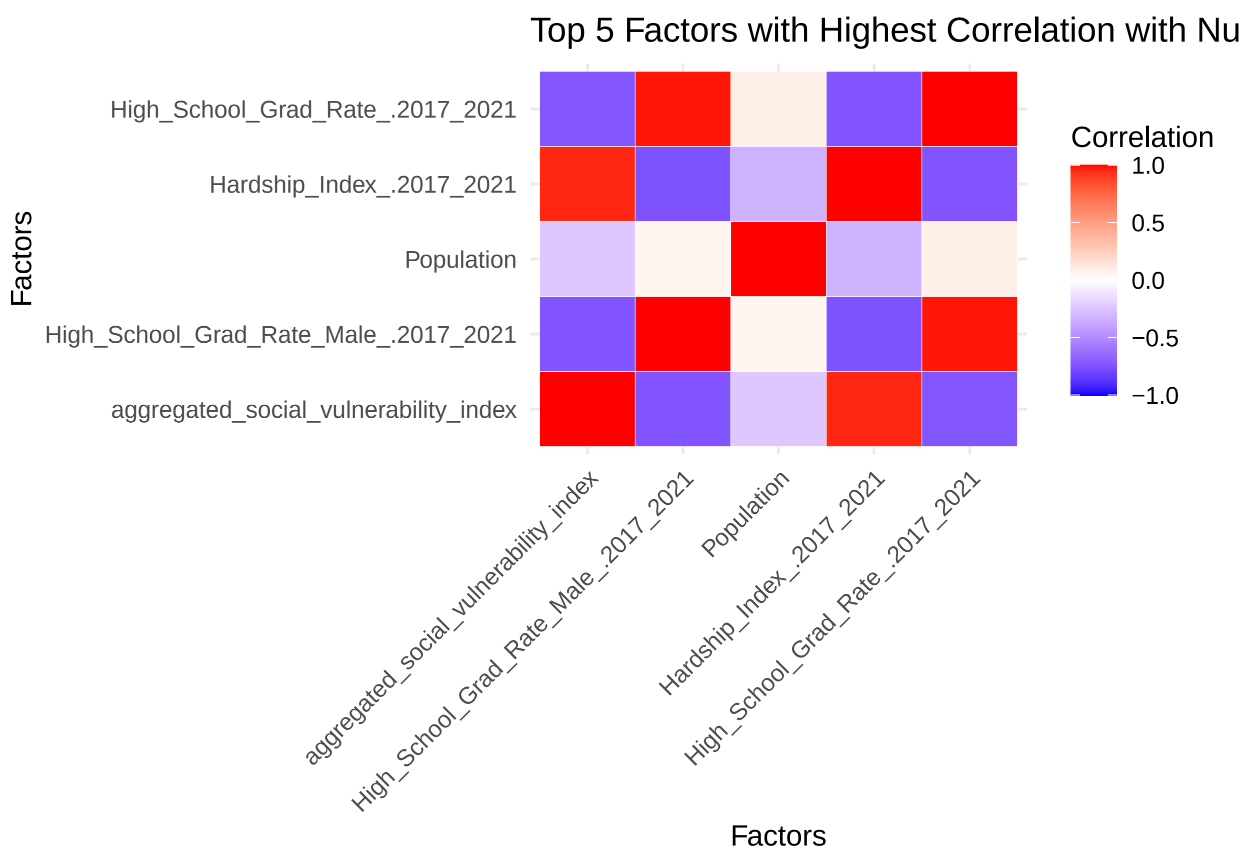
```
## [1] "Top 5 Factors with Highest Correlation:"
```

```r
print(top_5_factors)
```

```
## [1] "aggregated_social_vulnerability_index"
## [2] "High_School_Grad_Rate_Male_.2017_2021"
## [3] "Population"
## [4] "Hardship_Index_.2017_2021"
## [5] "High_School_Grad_Rate_.2017_2021"
```

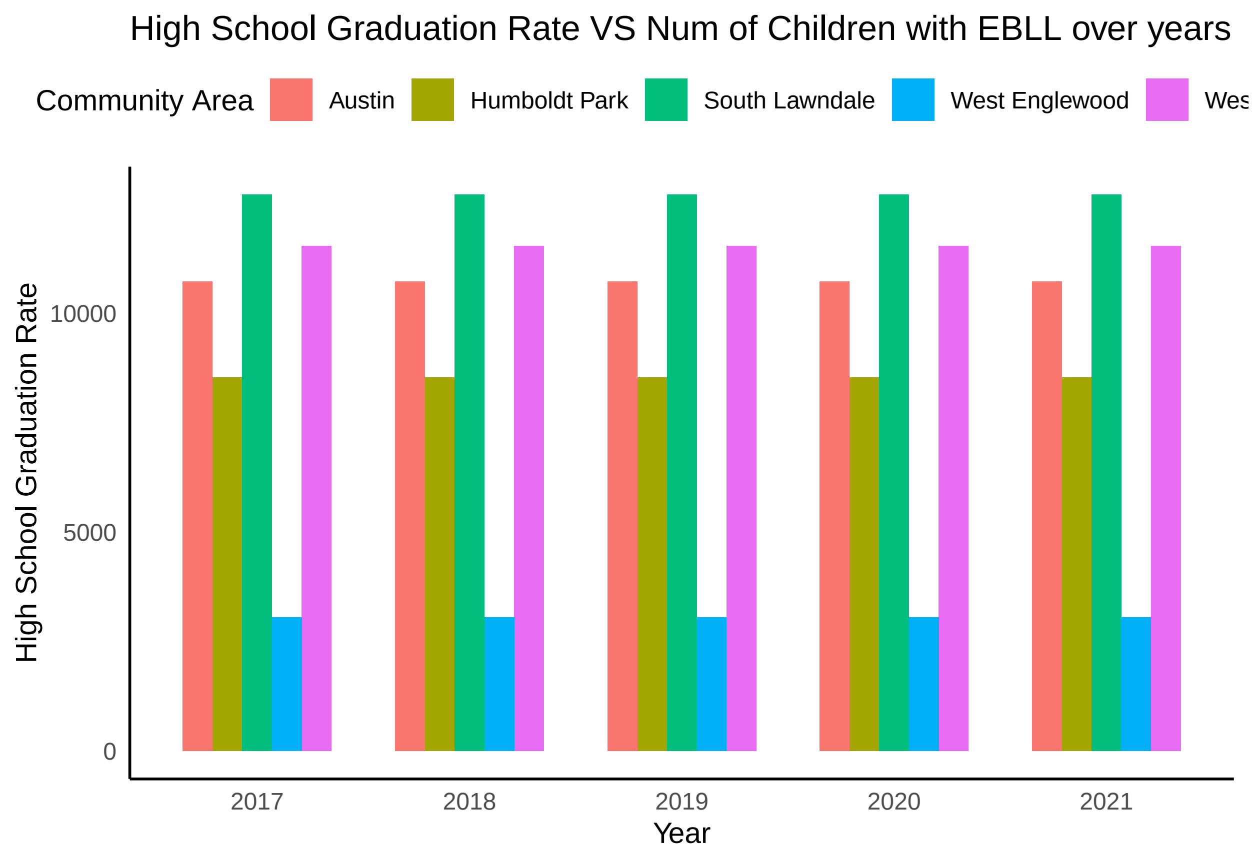
```r
# Visualization - Bar plot for the top 5 factors
library(ggplot2)


# Repeat Num_Of_Children_With_EBLL_data to match the length of top_5_factors
Num_Of_Children_With_EBLL_values <- rep(Num_Of_Children_With_EBLL_data, each = length(top_5_factors))

# Visualization - Bar plot for the top 5 factors
library(ggplot2)


# Assuming you have the correlation matrix stored in 'correlation_matrix'
# Assuming you have the top 5 factors stored in 'top_5_factors'

# Extract the correlation values for the top 5 factors
top_factors_correlation <- correlation_matrix["Num_Of_Children_With_EBLL_data", top_5_factors]

# Create a subset of the correlation matrix for the top 5 factors
subset_correlation_matrix <- correlation_matrix[top_5_factors, top_5_factors]

# Plot heatmap
library(ggplot2)
```

```r
# Convert the correlation matrix to long format
library(reshape2)
melted_correlation <- melt(subset_correlation_matrix)

# Plot heatmap
ggplot(melted_correlation, aes(Var1, Var2, fill = value)) +
  geom_tile(color = "white") +
  scale_fill_gradient2(low = "blue", mid = "white", high = "red",
                       midpoint = 0, limit = c(-1,1), space = "Lab",
                       name="Correlation") +
  theme_minimal() +
  labs(
    title = "Top 5 Factors with Highest Correlation with Number Of Children With EBLL",
    x = "Factors",
    y = "Factors"
  ) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



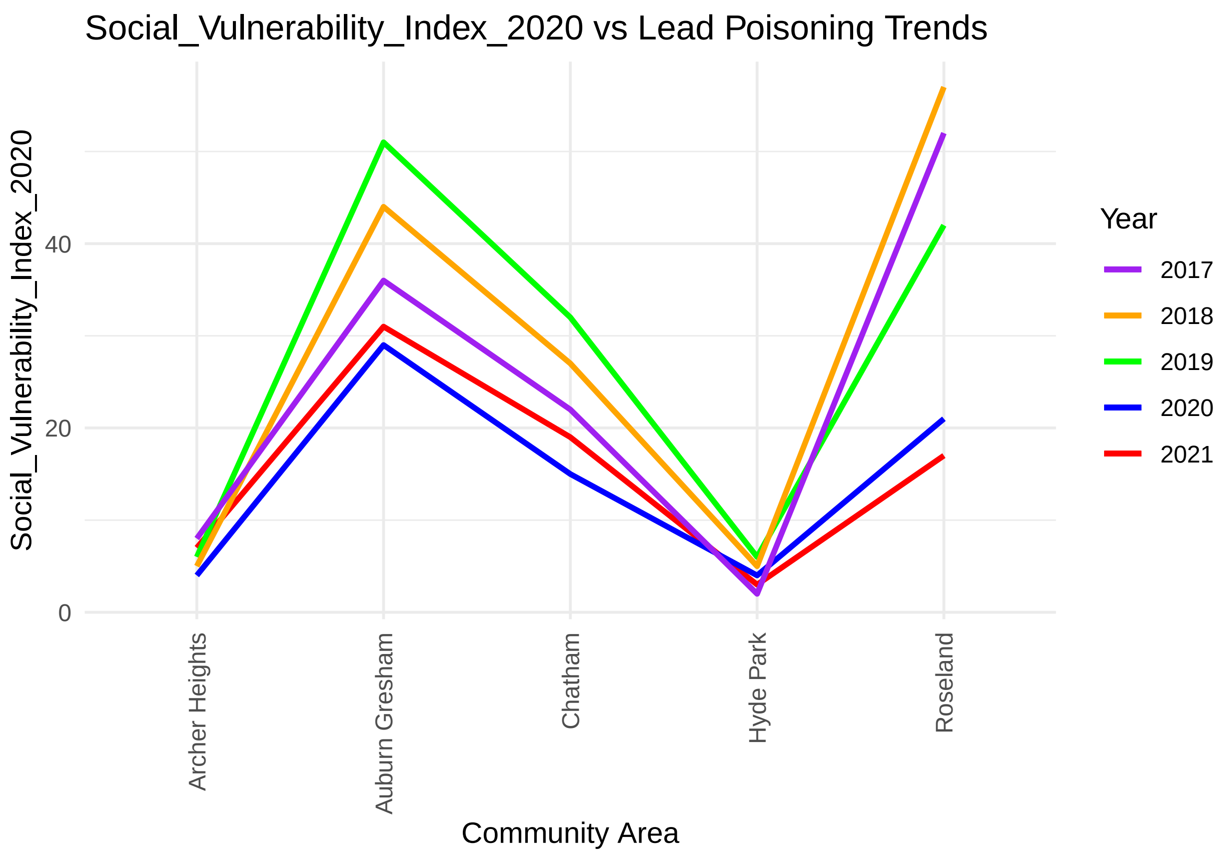

```r
# Assuming your dataset is named 'data'
# Assuming you have columns like 'Num_Of_Children_With_EBLL_2017', 'Num_Of_Children_With_EBLL_2018', et

# Melt the data to long format for easy plotting
melted_data <- reshape2::melt(data,
                              id.vars = c("Name", "Uninsured_Resident_2017_2021"),
                              measure.vars = c("Num_Of_Children_With_EBLL_2017",
```

```r
                                              "Num_Of_Children_With_EBLL_2018",
                                              "Num_Of_Children_With_EBLL_2019",
                                              "Num_Of_Children_With_EBLL_2020",
                                              "Num_Of_Children_With_EBLL_2021"),
                               variable.name = "Year",
                               value.name = "Num_Of_Children_With_EBLL")

# Extract only the year from the variable names
melted_data$Year <- as.numeric(gsub("Num_Of_Children_With_EBLL_", "", melted_data$Year))

# Filter and identify top 5 community areas with the highest Num_Of_Children_With_EBLL
top5_communities <- melted_data %>%
  group_by(Name) %>%
  summarize(Total_EBLL = sum(Num_Of_Children_With_EBLL, na.rm = TRUE)) %>%
  top_n(5, Total_EBLL) %>%
  select(Name)

# Filter data for the top 5 community areas
filtered_data <- melted_data %>%
  filter(Name %in% top5_communities$Name)

# Create a grouped bar plot
ggplot(filtered_data, aes(x = factor(Year), y = Uninsured_Resident_2017_2021, fill = Name)) +
  geom_bar(stat = "identity", position = "dodge", width = 0.7) +
  labs(title = "High School Graduation Rate VS Num of Children with EBLL over years",
       x = "Year",
       y = "High School Graduation Rate",
       fill = "Community Area") +
  theme_minimal() +
  theme(
    legend.position = "top",  # Change legend position
    panel.grid.major = element_blank(),  # Remove major grid lines
    panel.grid.minor = element_blank(),  # Remove minor grid lines
    panel.border = element_blank(),  # Remove panel border
    axis.line = element_line(colour = "black")  # Axis line color
  )
```



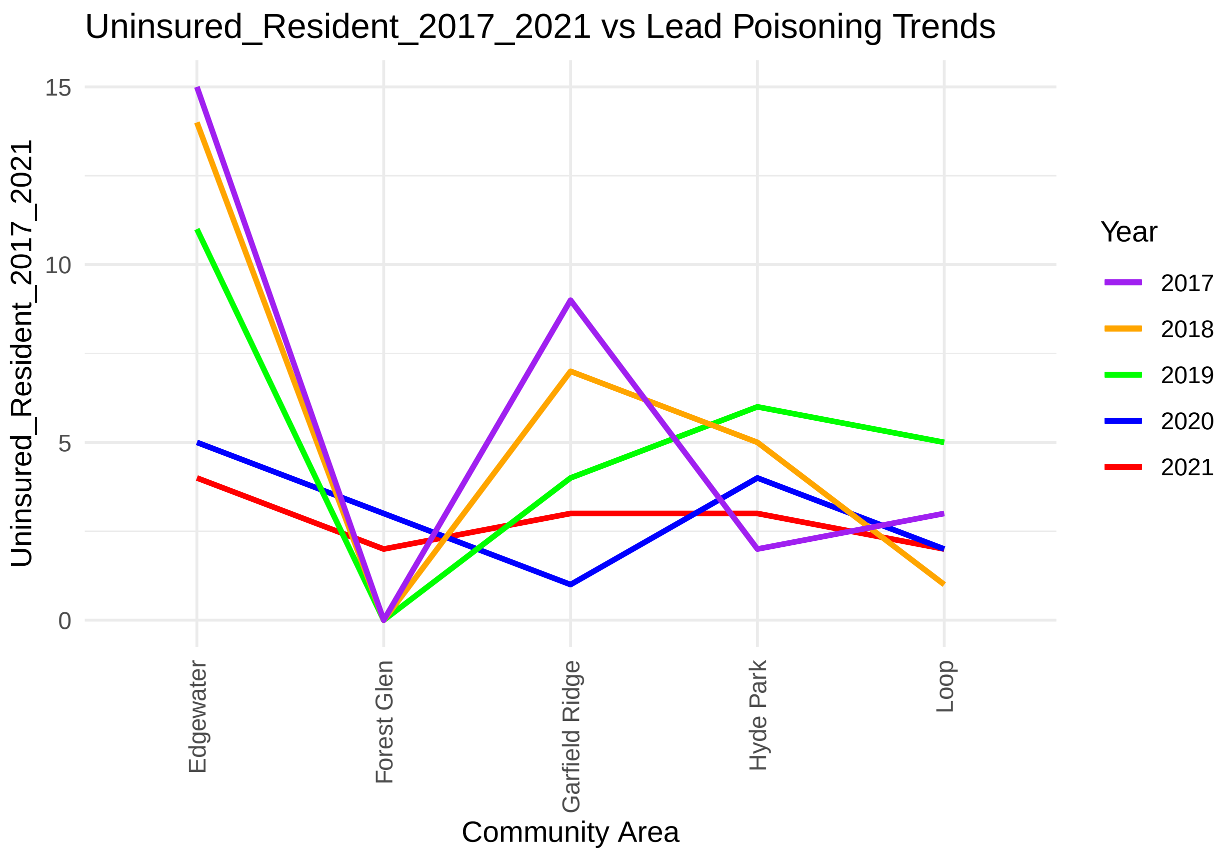

```r
library(ggplot2)
library(dplyr)

# Selecting important variables related to lead poisoning trends
lead_poisoning_columns <- c('Num_Of_Children_With_EBLL_2021', 'Num_Of_Children_With_EBLL_2020',
                            'Num_Of_Children_With_EBLL_2019', 'Num_Of_Children_With_EBLL_2018', 'Num_Of

other_variable <- 'Social_Vulnerability_Index_2020'

# Randomly select 5 community areas
selected_areas <- sample(data$Name, 5)

# Plotting lead poisoning trends over time for the selected variable and areas
plot_data <- data %>%
  filter(Name %in% selected_areas) %>%
  select(Name, all_of(c(lead_poisoning_columns, other_variable)))

ggplot(plot_data, aes(x = Name, group = 1)) +
  geom_line(aes(y = .data[[lead_poisoning_columns[1]]], color = "2021"), size = 1, linetype = "solid")
  geom_line(aes(y = .data[[lead_poisoning_columns[2]]], color = "2020"), size = 1, linetype = "solid")
  geom_line(aes(y = .data[[lead_poisoning_columns[3]]], color = "2019"), size = 1, linetype = "solid")
  geom_line(aes(y = .data[[lead_poisoning_columns[4]]], color = "2018"), size = 1, linetype = "solid")
  geom_line(aes(y = .data[[lead_poisoning_columns[5]]], color = "2017"), size = 1, linetype = "solid")
  labs(title = paste(other_variable, "vs Lead Poisoning Trends"),
       x = "Community Area",
       y = other_variable,
```

```r
      color = "Year") +
  scale_color_manual(values = c("2021" = "red", "2020" = "blue", "2019" = "green", "2018" = "orange", "2
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1))
```



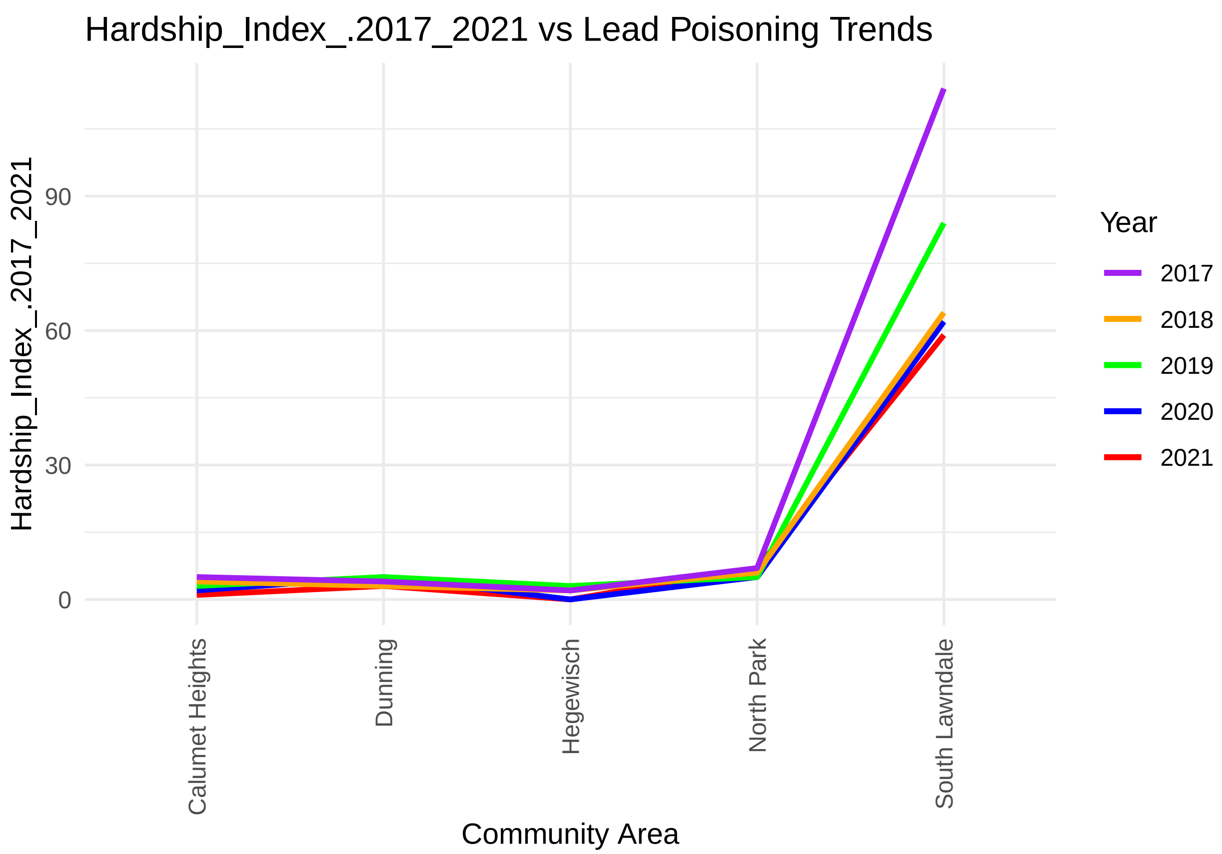

```r
uninsured_variable <- 'Uninsured_Resident_2017_2021'

# Randomly select 5 community areas
selected_areas <- sample(data$Name, 5)

# Plotting lead poisoning trends over time for the selected variable and areas
plot_data_uninsured <- data %>%
  filter(Name %in% selected_areas) %>%
  select(Name, all_of(c(lead_poisoning_columns, uninsured_variable)))

ggplot(plot_data_uninsured, aes(x = Name, group = 1)) +
  geom_line(aes(y = .data[[lead_poisoning_columns[1]]], color = "2021"), size = 1, linetype = "solid")
  geom_line(aes(y = .data[[lead_poisoning_columns[2]]], color = "2020"), size = 1, linetype = "solid")
  geom_line(aes(y = .data[[lead_poisoning_columns[3]]], color = "2019"), size = 1, linetype = "solid")
  geom_line(aes(y = .data[[lead_poisoning_columns[4]]], color = "2018"), size = 1, linetype = "solid")
  geom_line(aes(y = .data[[lead_poisoning_columns[5]]], color = "2017"), size = 1, linetype = "solid")
  labs(title = paste(uninsured_variable, "vs Lead Poisoning Trends"),
       x = "Community Area",
       y = uninsured_variable,
       color = "Year") +
```

```
  scale_color_manual(values = c("2021" = "red", "2020" = "blue", "2019" = "green", "2018" = "orange", "2
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1))
```




```
library(ggplot2)
library(dplyr)

# Selecting important variables related to lead poisoning trends
lead_poisoning_columns <- c('Num_Of_Children_With_EBLL_2021', 'Num_Of_Children_With_EBLL_2020',
                            'Num_Of_Children_With_EBLL_2019', 'Num_Of_Children_With_EBLL_2018', 'Num_Of

hardship_variable <- 'Hardship_Index_.2017_2021'

# Randomly select 5 community areas
selected_areas <- sample(data$Name, 5)

# Plotting lead poisoning trends over time for the selected variable and areas
plot_data_hardship <- data %>%
  filter(Name %in% selected_areas) %>%
  select(Name, all_of(c(lead_poisoning_columns, hardship_variable)))

ggplot(plot_data_hardship, aes(x = Name, group = 1)) +
  geom_line(aes(y = .data[[lead_poisoning_columns[1]]], color = "2021"), size = 1, linetype = "solid") +
  geom_line(aes(y = .data[[lead_poisoning_columns[2]]], color = "2020"), size = 1, linetype = "solid") +
  geom_line(aes(y = .data[[lead_poisoning_columns[3]]], color = "2019"), size = 1, linetype = "solid") +
```

```
  geom_line(aes(y = .data[[lead_poisoning_columns[4]]], color = "2018"), size = 1, linetype = "solid") +
  geom_line(aes(y = .data[[lead_poisoning_columns[5]]], color = "2017"), size = 1, linetype = "solid") +
  labs(title = paste(hardship_variable, "vs Lead Poisoning Trends"),
       x = "Community Area",
       y = hardship_variable,
       color = "Year") +
  scale_color_manual(values = c("2021" = "red", "2020" = "blue", "2019" = "green", "2018" = "orange", "2
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1))
```




```
# Read the CSV file
historical_data <- read.csv('/Users/tejaswiniviswanath/Downloads/Chicago Historical lead data.csv')
```

```
str(historical_data)
```

```
## 'data.frame':	77 obs. of  18 variables:
##  $ Name                             : chr  "Rogers Park" "Norwood Park" "Jefferson Park" "Forest Glen"
##  $ Num_Of_Children_With_EBLL_1996: int  337 5 4 5 22 232 60 132 5 1 ...
##  $ Num_Of_Children_With_EBLL_1997: int  300 6 11 0 17 234 57 113 8 5 ...
##  $ Num_Of_Children_With_EBLL_1998: int  252 6 4 7 12 211 53 103 10 10 ...
##  $ Num_Of_Children_With_EBLL_1999: int  229 3 5 0 10 193 42 84 6 8 ...
##  $ Num_Of_Children_With_EBLL_2000: int  150 2 5 1 21 138 39 77 5 5 ...
##  $ Num_Of_Children_With_EBLL_2001: int  123 4 3 0 9 126 32 82 3 5 ...
##  $ Num_Of_Children_With_EBLL_2002: int  92 1 5 0 12 97 33 68 4 4 ...
```

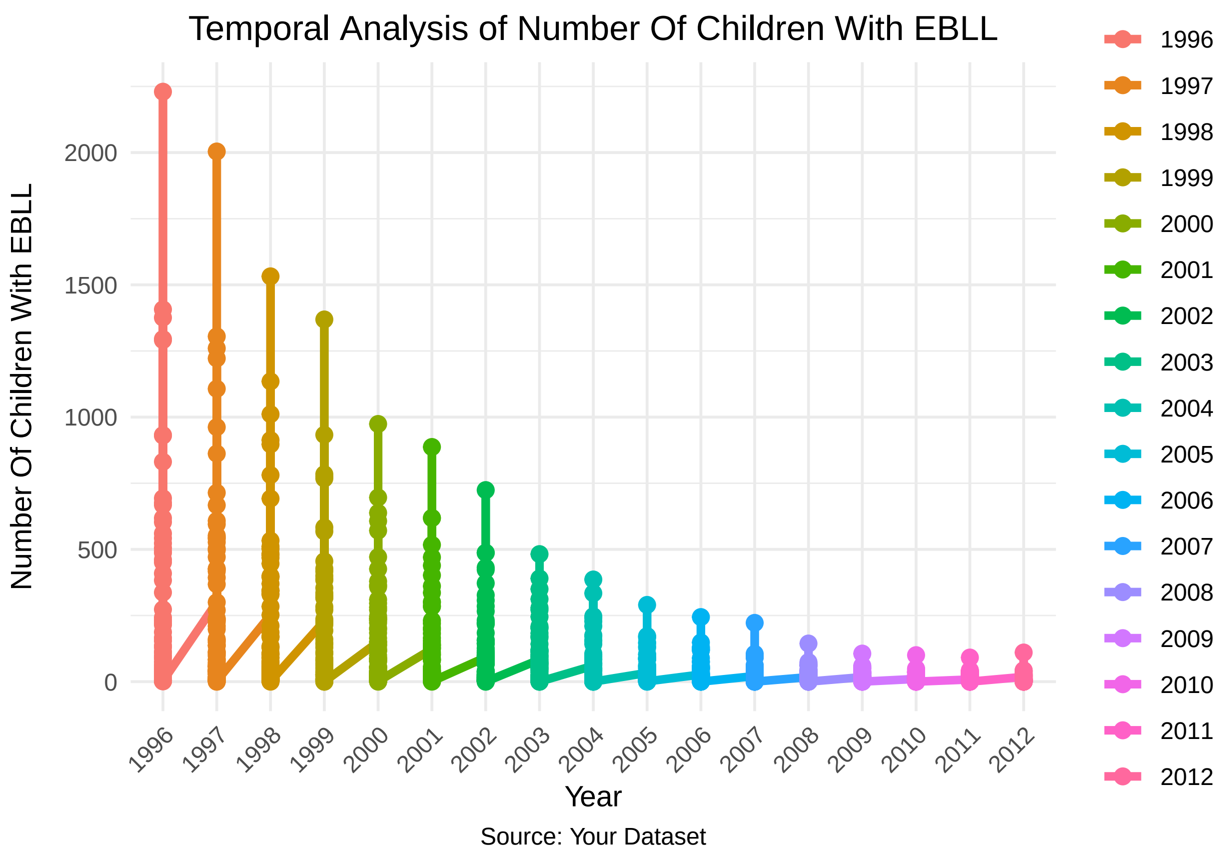
```
## $ Num_Of_Children_With_EBLL_2003: int 83 3 5 2 11 97 25 49 3 2 ...
## $ Num_Of_Children_With_EBLL_2004: int 60 1 0 0 7 52 29 48 4 5 ...
## $ Num_Of_Children_With_EBLL_2005: int 33 1 0 1 4 42 20 31 3 5 ...
## $ Num_Of_Children_With_EBLL_2006: int 27 2 4 0 4 28 17 11 3 4 ...
## $ Num_Of_Children_With_EBLL_2007: int 21 4 5 0 5 28 20 23 3 1 ...
## $ Num_Of_Children_With_EBLL_2008: int 17 3 4 0 1 26 12 10 4 1 ...
## $ Num_Of_Children_With_EBLL_2009: int 17 0 1 0 4 20 19 17 2 0 ...
## $ Num_Of_Children_With_EBLL_2010: int 10 2 0 1 1 13 13 18 2 3 ...
## $ Num_Of_Children_With_EBLL_2011: int 8 0 0 0 2 25 7 9 1 4 ...
## $ Num_Of_Children_With_EBLL_2012: int 18 0 1 0 4 16 4 13 0 4 ...
```

```r
summary(historical_data)
```

```
##      Name            Num_Of_Children_With_EBLL_1996
##  Length:77          Min.   :   1.0
##  Class :character   1st Qu.:  30.0
##  Mode  :character   Median :  95.0
##                     Mean   : 293.8
##                     3rd Qu.: 461.0
##                     Max.   :2230.0
##  Num_Of_Children_With_EBLL_1997 Num_Of_Children_With_EBLL_1998
##  Min.   :   0.0                 Min.   :   0.0
##  1st Qu.:  36.0                 1st Qu.:  26.0
##  Median : 104.0                 Median :  78.0
##  Mean   : 275.6                 Mean   : 222.1
##  3rd Qu.: 425.0                 3rd Qu.: 344.0
##  Max.   :2004.0                 Max.   :1532.0
##  Num_Of_Children_With_EBLL_1999 Num_Of_Children_With_EBLL_2000
##  Min.   :   0.0                 Min.   :  0.0
##  1st Qu.:  21.0                 1st Qu.: 16.0
##  Median :  70.0                 Median : 53.0
##  Mean   : 182.7                 Mean   :141.2
##  3rd Qu.: 283.0                 3rd Qu.:203.0
##  Max.   :1369.0                 Max.   :974.0
##  Num_Of_Children_With_EBLL_2001 Num_Of_Children_With_EBLL_2002
##  Min.   :  0.0                  Min.   :  0.0
##  1st Qu.: 15.0                  1st Qu.: 10.0
##  Median : 40.0                  Median : 36.0
##  Mean   :125.8                  Mean   :102.3
##  3rd Qu.:154.0                  3rd Qu.:126.0
##  Max.   :887.0                  Max.   :724.0
##  Num_Of_Children_With_EBLL_2003 Num_Of_Children_With_EBLL_2004
##  Min.   :  0.00                 Min.   :  0.00
##  1st Qu.:  8.00                 1st Qu.:  6.00
##  Median : 22.00                 Median : 19.00
##  Mean   : 74.71                 Mean   : 56.81
##  3rd Qu.:103.00                 3rd Qu.: 70.00
##  Max.   :482.00                 Max.   :386.00
##  Num_Of_Children_With_EBLL_2005 Num_Of_Children_With_EBLL_2006
##  Min.   :  0.00                 Min.   :  0.00
##  1st Qu.:  5.00                 1st Qu.:  4.00
##  Median : 14.00                 Median : 11.00
##  Mean   : 40.73                 Mean   : 29.69
##  3rd Qu.: 56.00                 3rd Qu.: 33.00
```

```
## Max.   :290.00                Max.   :244.00
## Num_Of_Children_With_EBLL_2007 Num_Of_Children_With_EBLL_2008
## Min.   :  0.00                Min.   :  0.00
## 1st Qu.:  3.00                1st Qu.:  2.00
## Median : 11.00                Median :  7.00
## Mean   : 25.38                Mean   : 17.71
## 3rd Qu.: 35.00                3rd Qu.: 25.00
## Max.   :222.00                Max.   :144.00
## Num_Of_Children_With_EBLL_2009 Num_Of_Children_With_EBLL_2010
## Min.   :  0.00                Min.   :  0.00
## 1st Qu.:  3.00                1st Qu.:  2.00
## Median :  8.00                Median :  6.00
## Mean   : 14.95                Mean   : 12.65
## 3rd Qu.: 20.00                3rd Qu.: 18.00
## Max.   :106.00                Max.   :100.00
## Num_Of_Children_With_EBLL_2011 Num_Of_Children_With_EBLL_2012
## Min.   : 0.00                 Min.   :  0.00
## 1st Qu.: 1.00                 1st Qu.:  2.00
## Median : 5.00                 Median :  6.00
## Mean   :11.31                 Mean   : 11.88
## 3rd Qu.:17.00                 3rd Qu.: 18.00
## Max.   :91.00                 Max.   :110.00
```

```r
# Assuming 'historical_data' is your dataset

# Load required libraries
library(ggplot2)
library(tidyr)

# Convert data to long format for easier plotting
historical_data_long <- pivot_longer(historical_data, cols = starts_with("Num_Of_Children_With_EBLL_"),

# Convert 'Year' to numeric (assuming it's stored as a character)
historical_data_long$Year <- as.factor(gsub("Num_Of_Children_With_EBLL_", "", historical_data_long$Year

# Generate a dynamic color palette based on the number of unique years
line_colors <- scales::hue_pal()(length(unique(historical_data_long$Year)))

# Plot the time series with enhanced aesthetics
ggplot(historical_data_long, aes(x = Year, y = Num_Of_Children_With_EBLL_Level, group = 1)) +
  geom_line(aes(color = Year), linewidth = 1.5) +
  geom_point(aes(color = Year), size = 2.5) +
  scale_color_manual(values = line_colors) +
  labs(title = "Temporal Analysis of Number Of Children With EBLL",
       x = "Year",
       y = "Number Of Children With EBLL",
       caption = "Source: Your Dataset") +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5),
        plot.caption = element_text(hjust = 0.5),
        axis.title.y = element_text(margin = margin(r = 10)),
        axis.text.x = element_text(angle = 45, hjust = 1, vjust = 1))
```



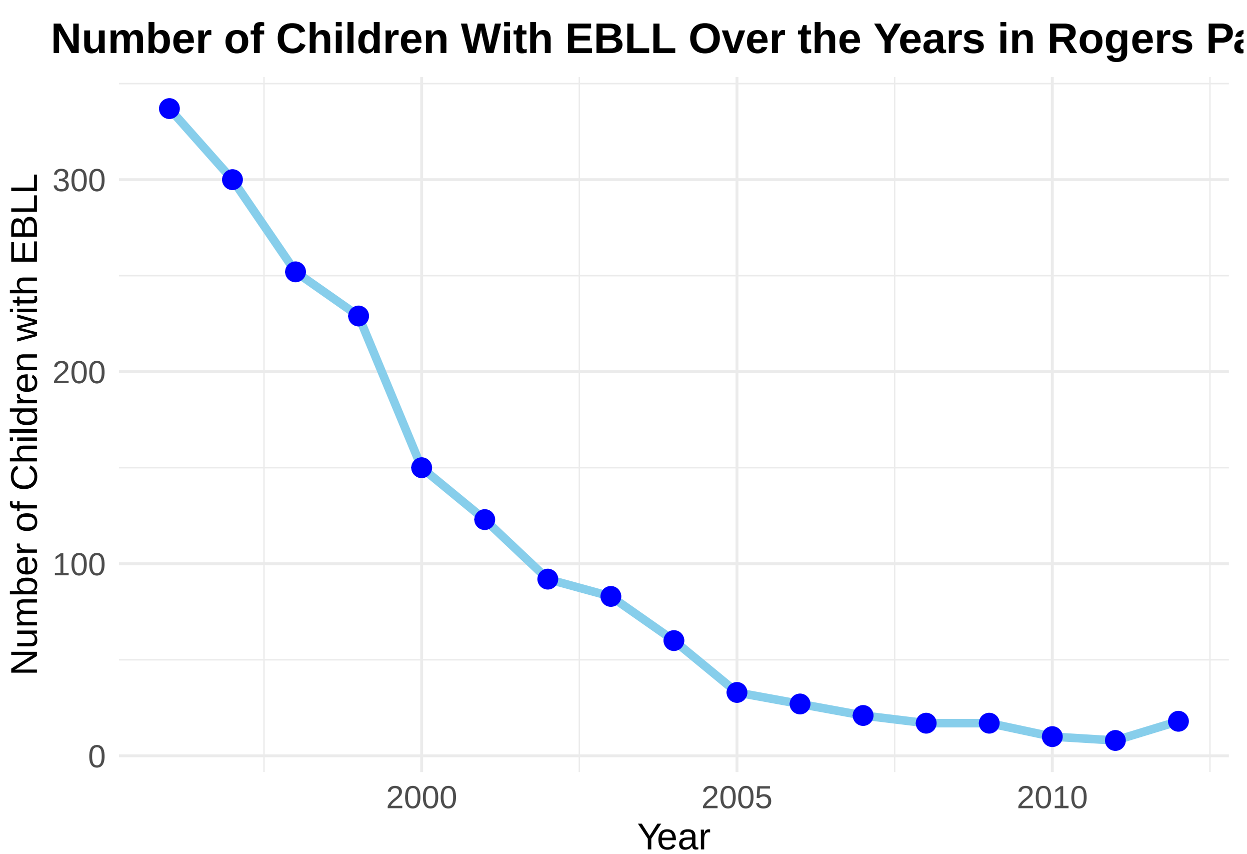

```
# Load required libraries
library(ggplot2)
library(dplyr)

# Assuming your dataset is named 'historical_data'
# Assuming you have columns like 'Num_Of_Children_With_EBLL_1996', 'Num_Of_Children_With_EBLL_1997', et

# Select one random community area
selected_area <- sample(historical_data$Name, 1)

# Filter data for the selected community area
selected_area_data <- historical_data %>%
  filter(Name == selected_area) %>%
  select(starts_with("Num_Of_Children_With_EBLL"))

# Melt the data to long format for easy plotting
melted_data <- reshape2::melt(selected_area_data,
                              variable.name = "Year",
                              value.name = "Num_Of_Children_With_EBLL")
```

```
## No id variables; using all as measure variables
```

```
# Extract only the year from the variable names
melted_data$Year <- as.numeric(gsub("Num_Of_Children_With_EBLL_", "", melted_data$Year))

print(selected_area)
```

```
## [1] "Rogers Park"
```

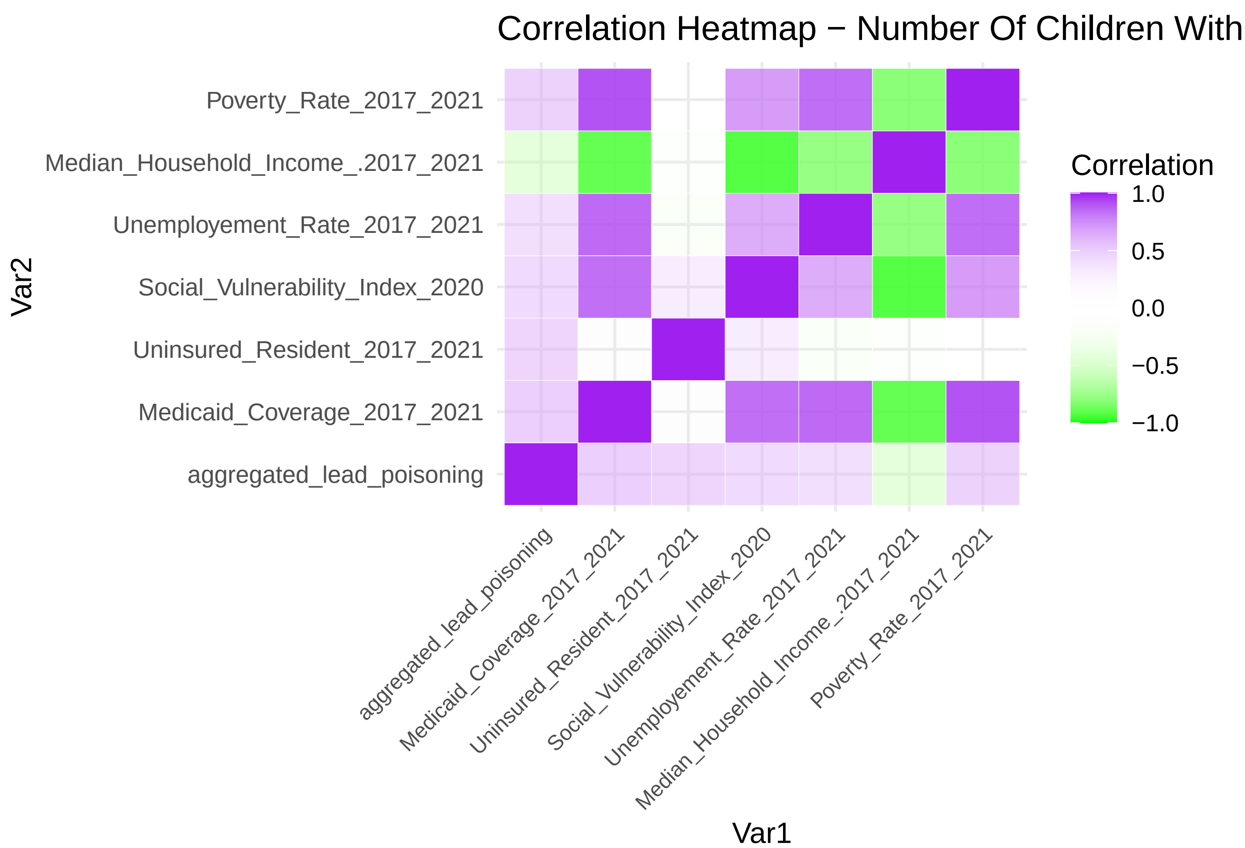
```r
# Create a visually appealing time series plot
ggplot(melted_data, aes(x = Year, y = Num_Of_Children_With_EBLL, group = 1)) +
  geom_line(color = "skyblue", size = 1.5) +
  geom_point(color = "blue", size = 3) +
  labs(title = paste("Number of Children With EBLL Over the Years in", selected_area),
       x = "Year",
       y = "Number of Children with EBLL") +
  theme_minimal() +
  theme(
    plot.title = element_text(size = 16, face = "bold", hjust = 0.5),
    axis.title.x = element_text(size = 14),
    axis.title.y = element_text(size = 14),
    axis.text = element_text(size = 12),
    legend.position = "none"  # Remove legend
  )
```




```r
# Assuming 'historical_data' and 'data' are your datasets

# Aggregate Number Of Children With EBLL data over the years
aggregated_lead_poisoning <- rowSums(historical_data[, -1], na.rm = TRUE)
```

```r
# Select economic and health-related columns for analysis
factors_columns <- c(
  "Medicaid_Coverage_2017_2021",
  "Uninsured_Resident_2017_2021",
  "Social_Vulnerability_Index_2020",
  "Unemployement_Rate_2017_2021",
  "Median_Household_Income_.2017_2021",
  "Poverty_Rate_2017_2021"
)

# Check if the specified factor columns exist in the 'data' dataframe
if (all(factors_columns %in% colnames(data))) {
  # Subsetting the data
  factors_data <- data[, factors_columns]

  # Combine aggregated_lead_poisoning with factors_data
  combined_data <- cbind(aggregated_lead_poisoning, factors_data)

  # Calculate the correlation matrix
  correlation_matrix <- cor(combined_data, use = "complete.obs")

  # Extract the correlation between Number Of Children With EBLL and economic/health factors
  correlation_with_lead_poisoning <- correlation_matrix[1, factors_columns]

  # Print the correlation values
  print(correlation_with_lead_poisoning)

  # Visualization - Heatmap
  library(ggplot2)
  library(reshape2)

  ggplot(melt(correlation_matrix), aes(x = Var1, y = Var2, fill = value)) +
    geom_tile(color = "white") +
    scale_fill_gradient2(low = "green", high = "purple", mid = 0, midpoint = 0, limit = c(-1, 1), space
                         name = "Correlation") +
    theme_minimal() +
    theme(axis.text.x = element_text(angle = 45, vjust = 1, size = 8, hjust = 1)) +
    labs(title = "Correlation Heatmap - Number Of Children With EBLL and Economic/Health Factors")
} else {
  cat("Error: Some factor columns are not present in the 'data' dataframe.\n")
}
```

```
##        Medicaid_Coverage_2017_2021       Uninsured_Resident_2017_2021
##                          0.4864369                          0.4553872
##    Social_Vulnerability_Index_2020       Unemployement_Rate_2017_2021
##                          0.4296922                          0.4056214
## Median_Household_Income_.2017_2021             Poverty_Rate_2017_2021
##                         -0.4137244                          0.4684106
```



R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

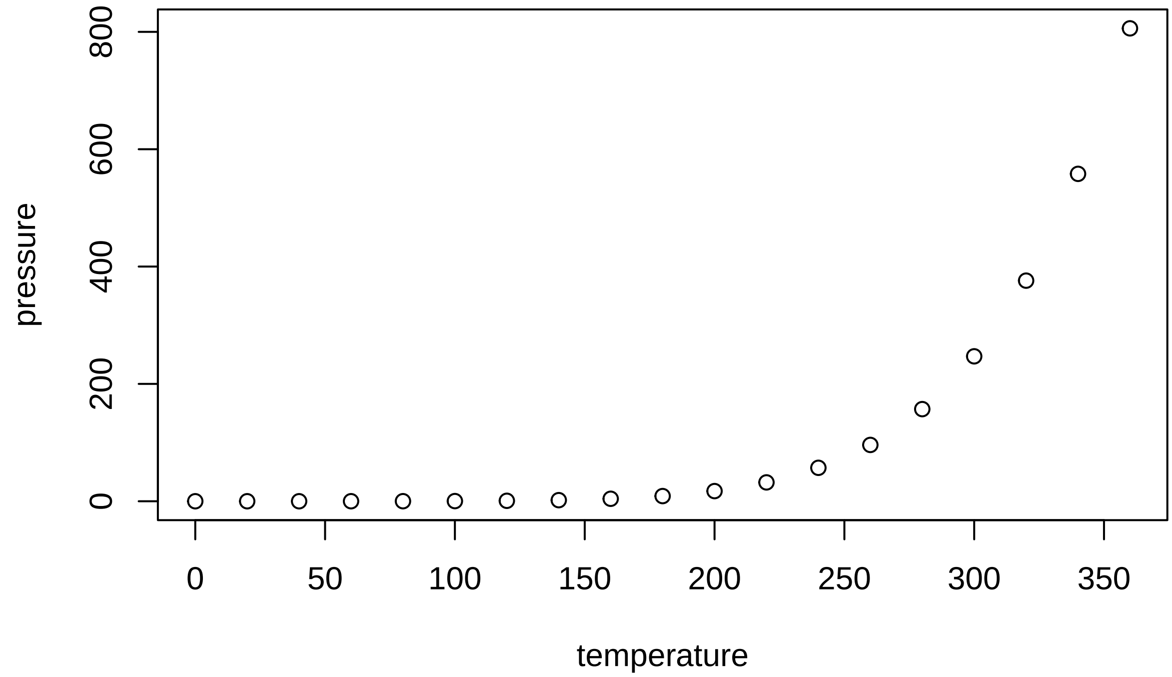
When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:teja2924

```
summary(cars)
```

```
##      speed           dist       
##  Min.   : 4.0   Min.   :  2.00  
##  1st Qu.:12.0   1st Qu.: 26.00  
##  Median :15.0   Median : 36.00  
##  Mean   :15.4   Mean   : 42.98  
##  3rd Qu.:19.0   3rd Qu.: 56.00  
##  Max.   :25.0   Max.   :120.00
```

Including Plots

You can also embed plots, for example:



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.