Exploratory Analysis

Contents

Overview 1

Descriptive Statistics 2

Individual Variable Plots 3

Chi-squared Tests of Association 59

Correlation Scatter Plots and Correlation Tests 59

Social Networks 90

Overview

This section conducts an exploratory analysis of the raw data. Specifically for visualizing and deriving descriptive statistics on the data In order to understand the underlying the structures and patterns in the data. A variety of descriptive statistics will be generated for both the numeric data and the categorical data. Similarly a variety of visualizations will be generated for both the numeric data and the categorical data. Further more some visualization on the patterns of diaspora will be generated using Social Networks

```
# Load in relevant libraries
library(rmarkdown)
library(knitr)
library(igraph)
```

```
##
## Attaching package: 'igraph'

## The following objects are masked from 'package:stats':
##
##     decompose, spectrum

## The following object is masked from 'package:base':
##
##     union
```

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

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:igraph':
##
##     as_data_frame, groups, union

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
# load in custom utility functions
source("./scripts/utilities/factor_descriptive_statistics.R")
source("./scripts/utilities/numeric_descriptive_statistics.R")
source("./scripts/utilities/plot_bar_chart.R")
source("./scripts/utilities/plot_histogram.R")
source("./scripts/utilities/plot_scatter.R")
source("./scripts/utilities/chisq_assoc_test.R")
source("./scripts/utilities/corr_tests.R")
source("./scripts/utilities/pred_corr_tests.R")
source("./scripts/utilities/graph_network.R")


# load in the raw ESC 2016 data for the analysis
ESCdata <- read.csv(file = "./data/ESC_2016_voting_data.csv", header = T)
```

```
# define vectors for all numeric columns
avg_point_num <- c('Average_Points')
mig_num <- c('FC_NonCOB', 'FC_NonCitzens', 'FC_COB', 'FC_Citizens', 'FC_Population', 'METRIC_COB', 'METR
mus_num <- c('danceability', 'loudiness', 'liveness', 'energy', 'speechiness', 'acousticness', 'instrume
demo_num <- c('TC_NumNeigh', 'FC_CAP_LON', 'FC_CAP_LAT', 'TC_CAP_LON', 'TC_CAP_LAT', 'FC_GDP_mil', 'TC_C
comp_num <- c('OOA')
# consolidate all numeric columns
all_num <- c(avg_point_num, mig_num, mus_num, demo_num, comp_num)

# define vectors for all categorical columns
voting_factors <- c('From_country', 'To_country', 'Points')
comp_factors <- c('Round', 'Voting_Method', 'Host_Nation', 'OOA')
ext_factors <- c('VBlocs1_TC', 'VBlocs2_TC', 'VBlocs1_FC', 'VBlocs2_FC', 'ComVBlocs1', 'ComVBlocs2', 'FC
perf_factors <- c('FC_SONGLANG', 'TC_SONGLANG','ComSONGLAN', 'TC_PerfType', 'TC_SingerGender', 'key', 'm
# consolidate all factor columns
all_factors <- c(voting_factors, comp_factors, ext_factors, perf_factors)
```

Descriptive Statistics

Here relevant descriptive statistics are derived for the exploratory analysis. Two custom function are used to generate the descriptive statistics in twp separate data frames. specifically for categorical data and numeric data.

```r
# run the categorical descriptive statistics function
raw_data_cat_desc_stats <- factor_descriptive_statistics(dataset = ESCdata, col_names = all_factors)
# write the categorical descriptive statistics to disk
write.csv(x = raw_data_cat_desc_stats, file = "./report/stats/categorical_descriptive_statistics.csv")

# generate numeric descriptive statistics
raw_data_num_desc_stats <- numeric_descriptive_statistics(dataset = ESCdata, col_names = all_num)
# write the numeric descriptive statistics to disk
write.csv(x = raw_data_num_desc_stats, file = "./report/stats/numeric_descriptive_statistics.csv")
```

Individual Variable Plots

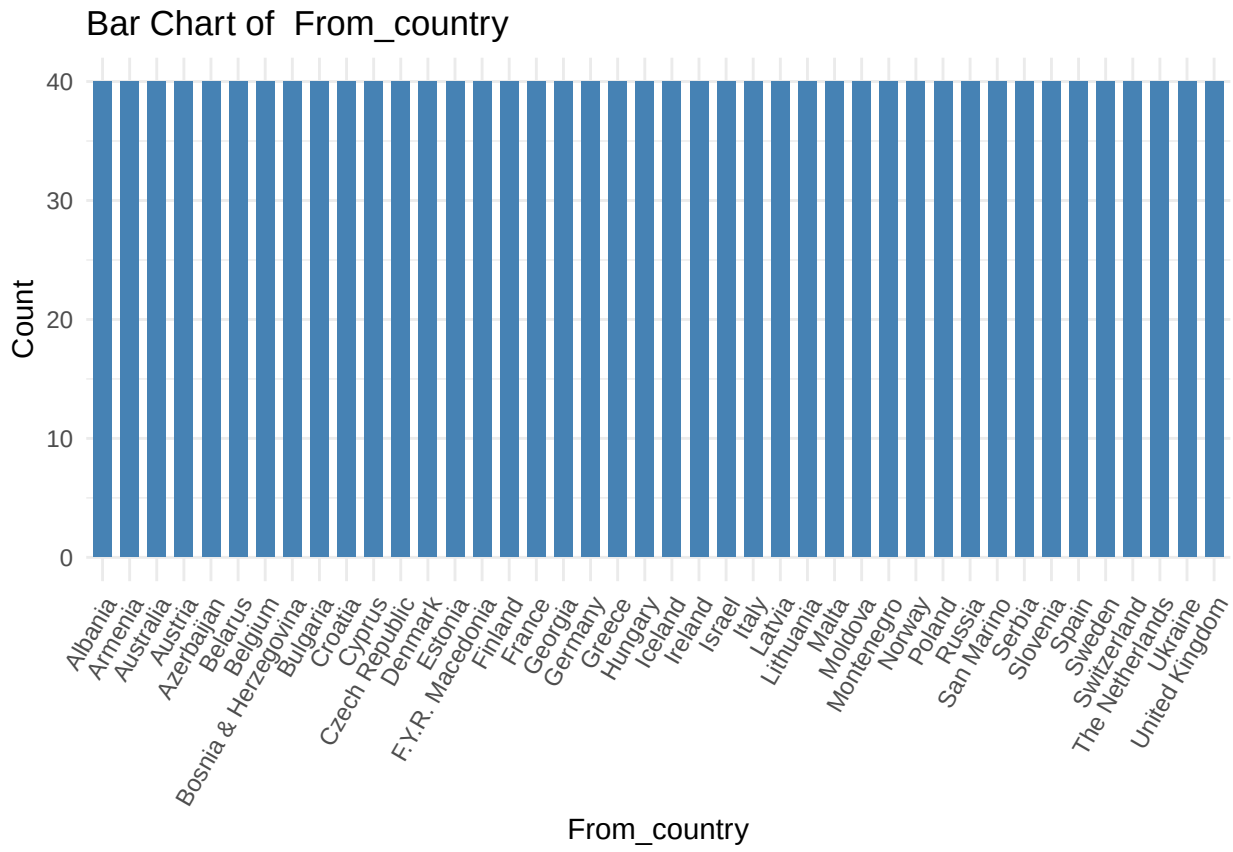
This section generates a variety of data visualizations. To display the underlying structures within each variable. Bar charts are generated for each categorical variable. histograms are generated for each numeric variable.

```r
# call bar chart plotting function
plot_bar_chart(dataset = ESCdata, col_names = all_factors, output_dir = './report/plots/bar_charts')
```

```
## Warning: ‘guides(<scale> = FALSE)‘ is deprecated. Please use ‘guides(<scale> =
## "none")‘ instead.
```

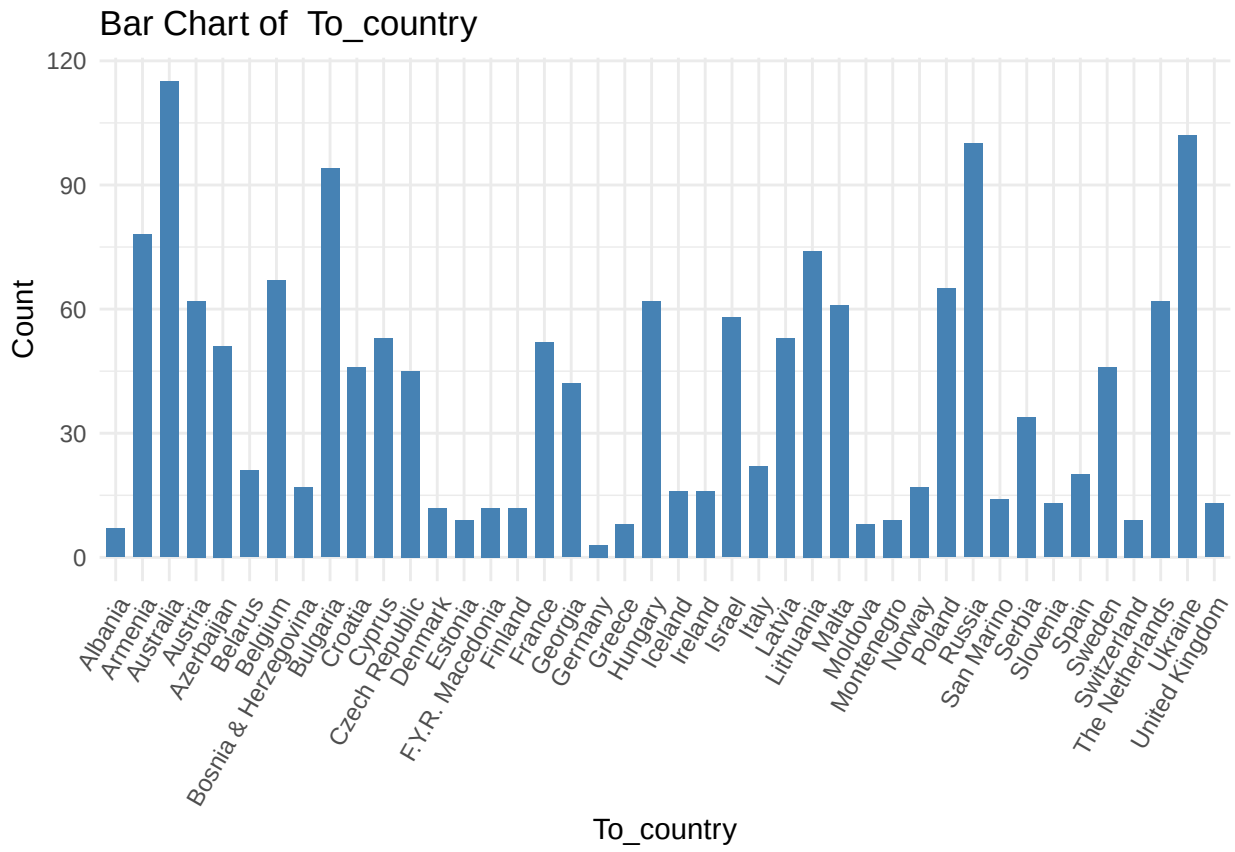
```
## Saving 6.5 x 4.5 in image
```

```
## Warning: ‘guides(<scale> = FALSE)‘ is deprecated. Please use ‘guides(<scale> =
## "none")‘ instead.
```



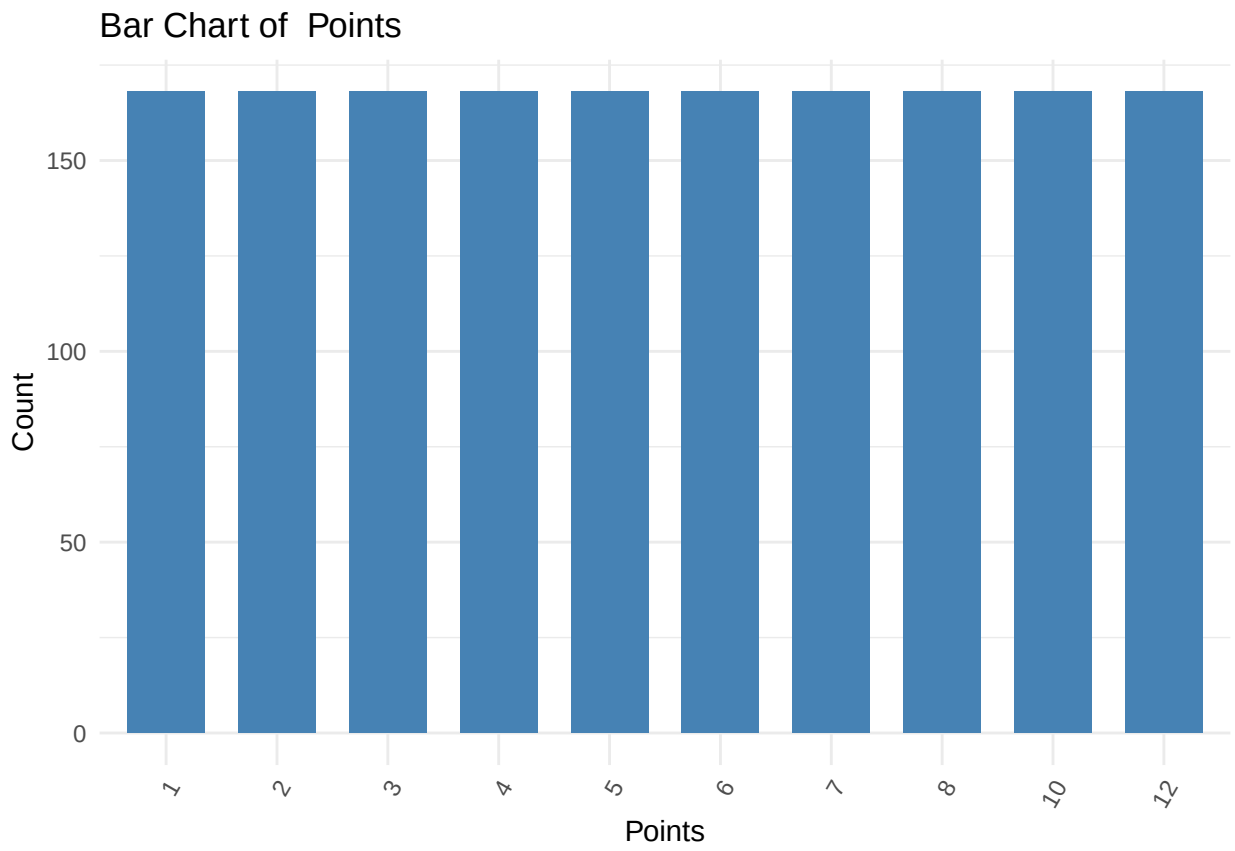

```
## Saving 6.5 x 4.5 in image
```

```
## Warning: ‘guides(<scale> = FALSE)‘ is deprecated. Please use ‘guides(<scale> =
## "none")‘ instead.
```



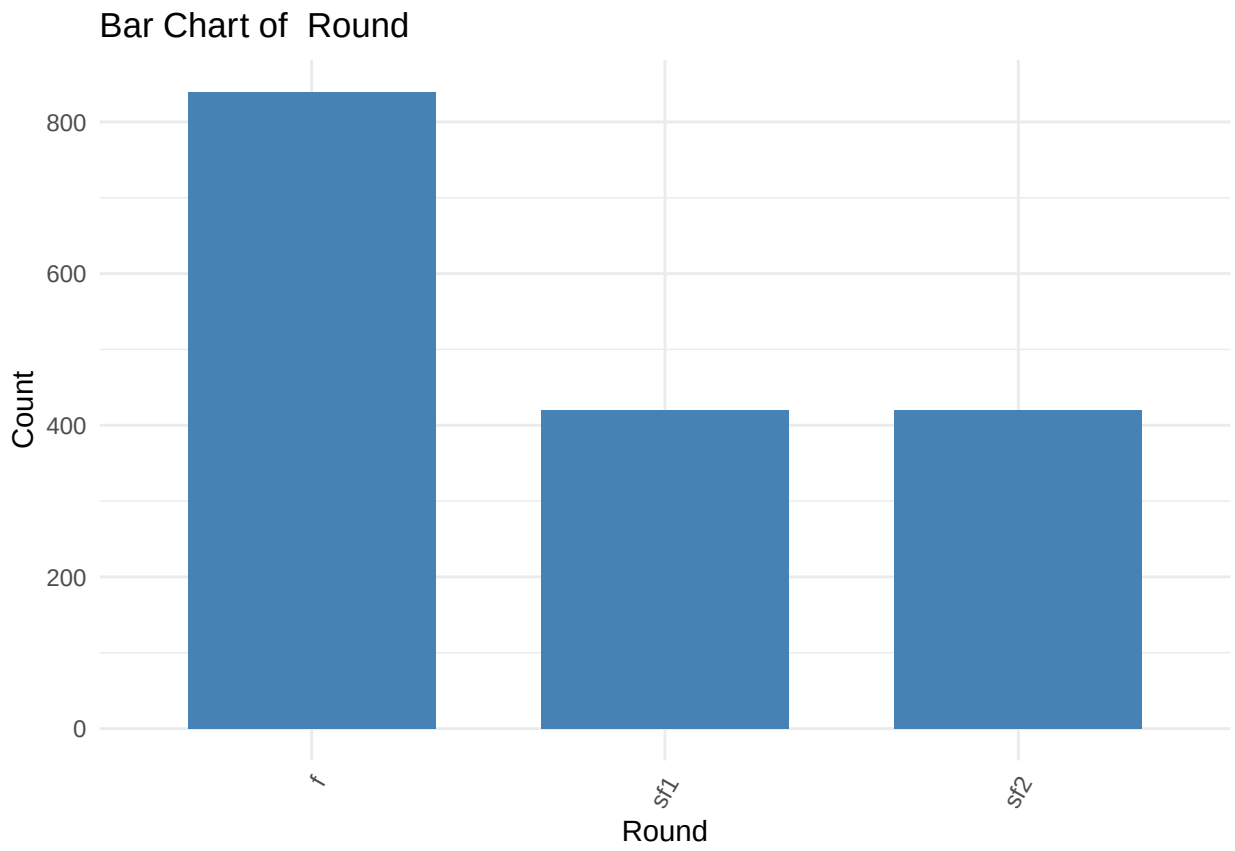

```
## Saving 6.5 x 4.5 in image
```

```
## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =
## "none")` instead.
```



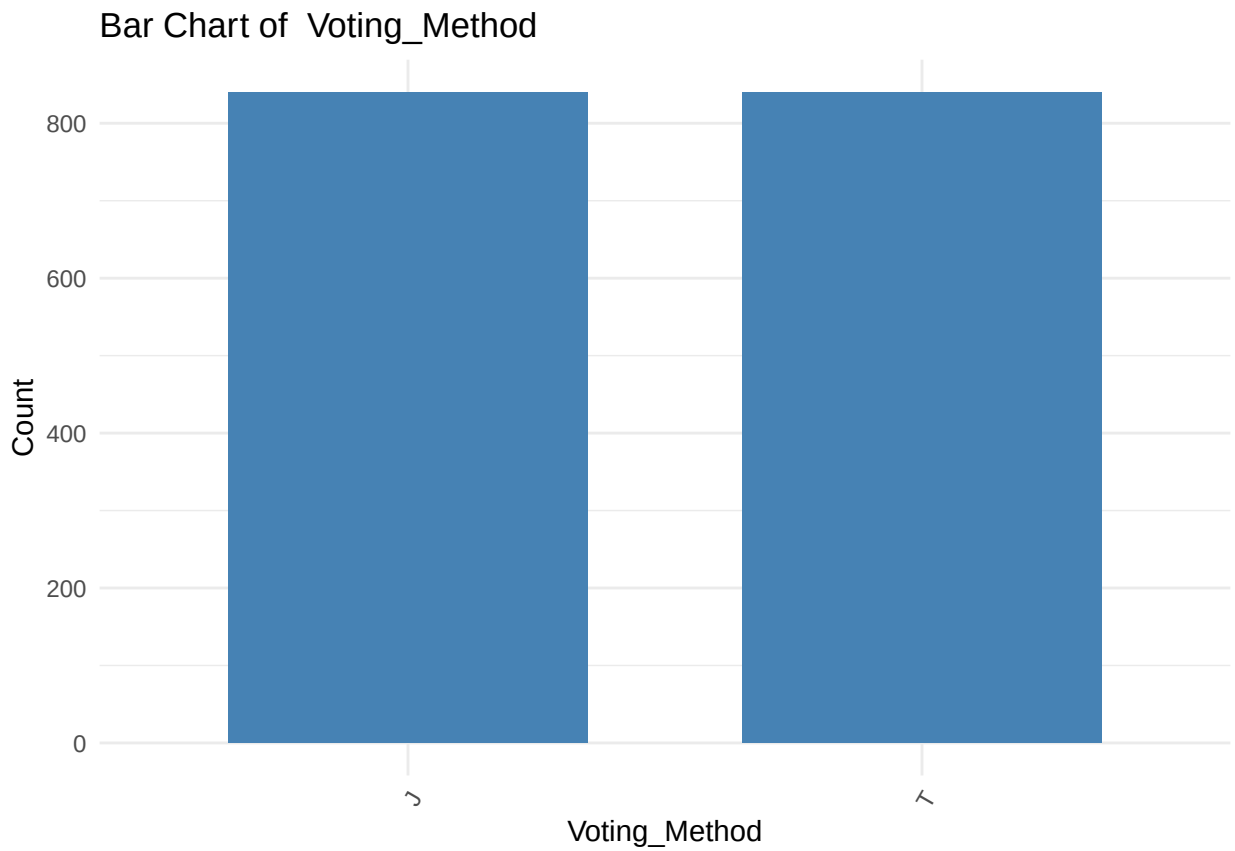

```
## Saving 6.5 x 4.5 in image
```

```
## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =
## "none")` instead.
```



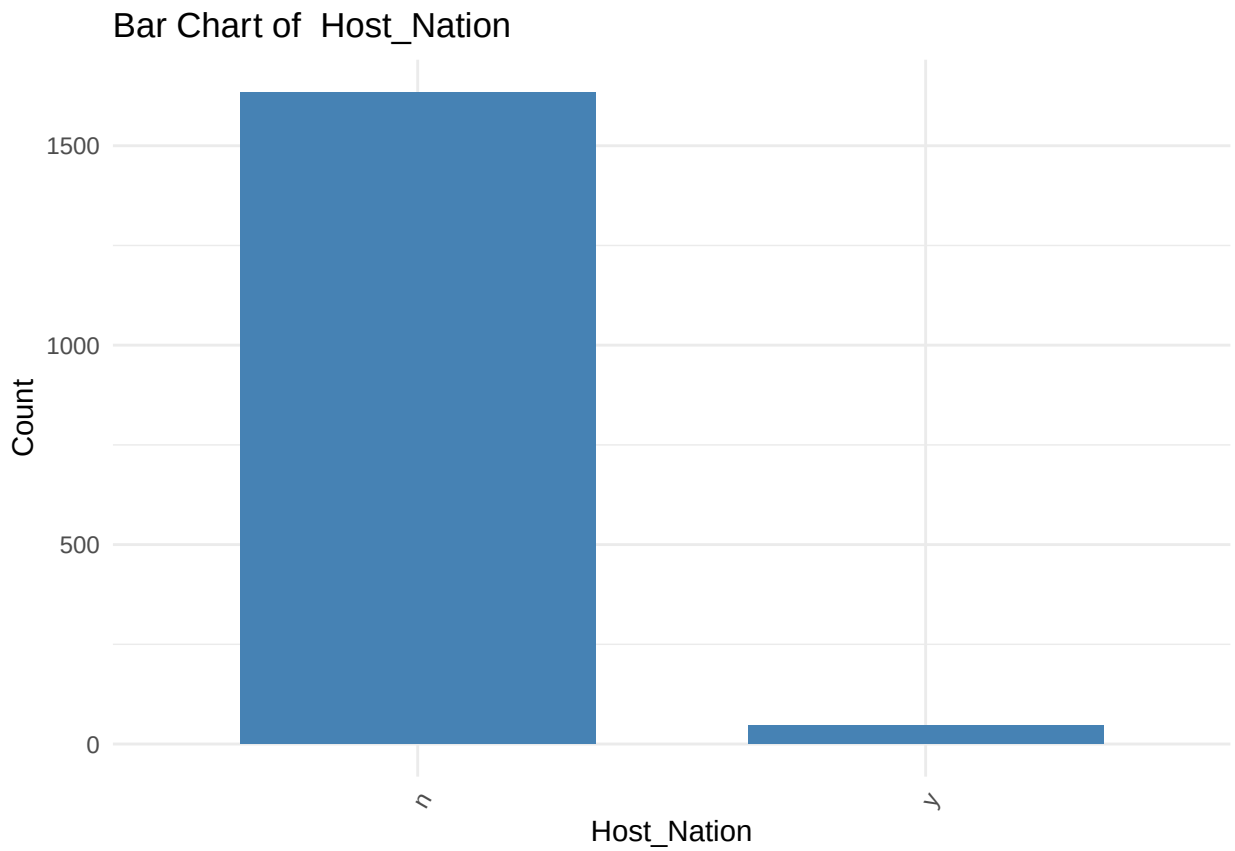

```
## Saving 6.5 x 4.5 in image
```

```
## Warning: ‘guides(<scale> = FALSE)‘ is deprecated. Please use ‘guides(<scale> =
## "none")‘ instead.
```



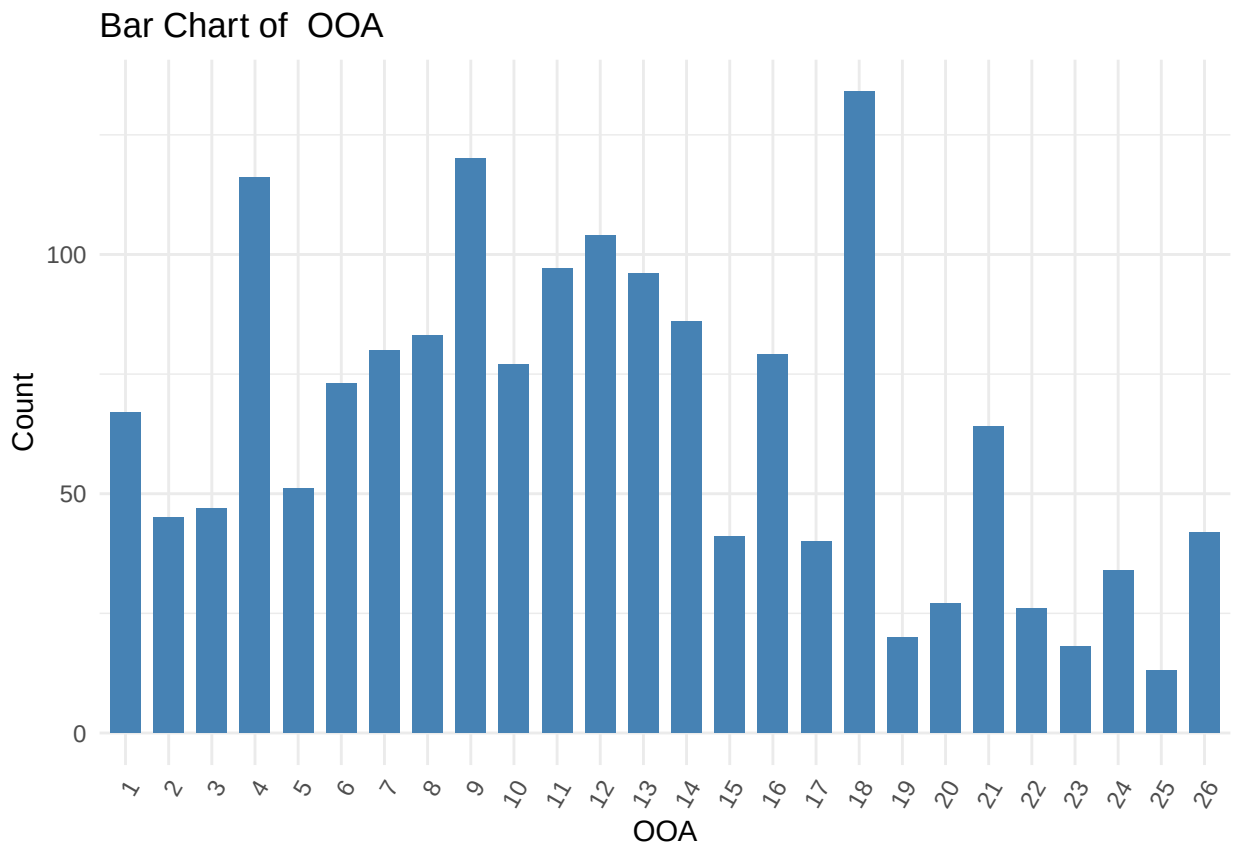

```
## Saving 6.5 x 4.5 in image
```

```
## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =
## "none")` instead.
```



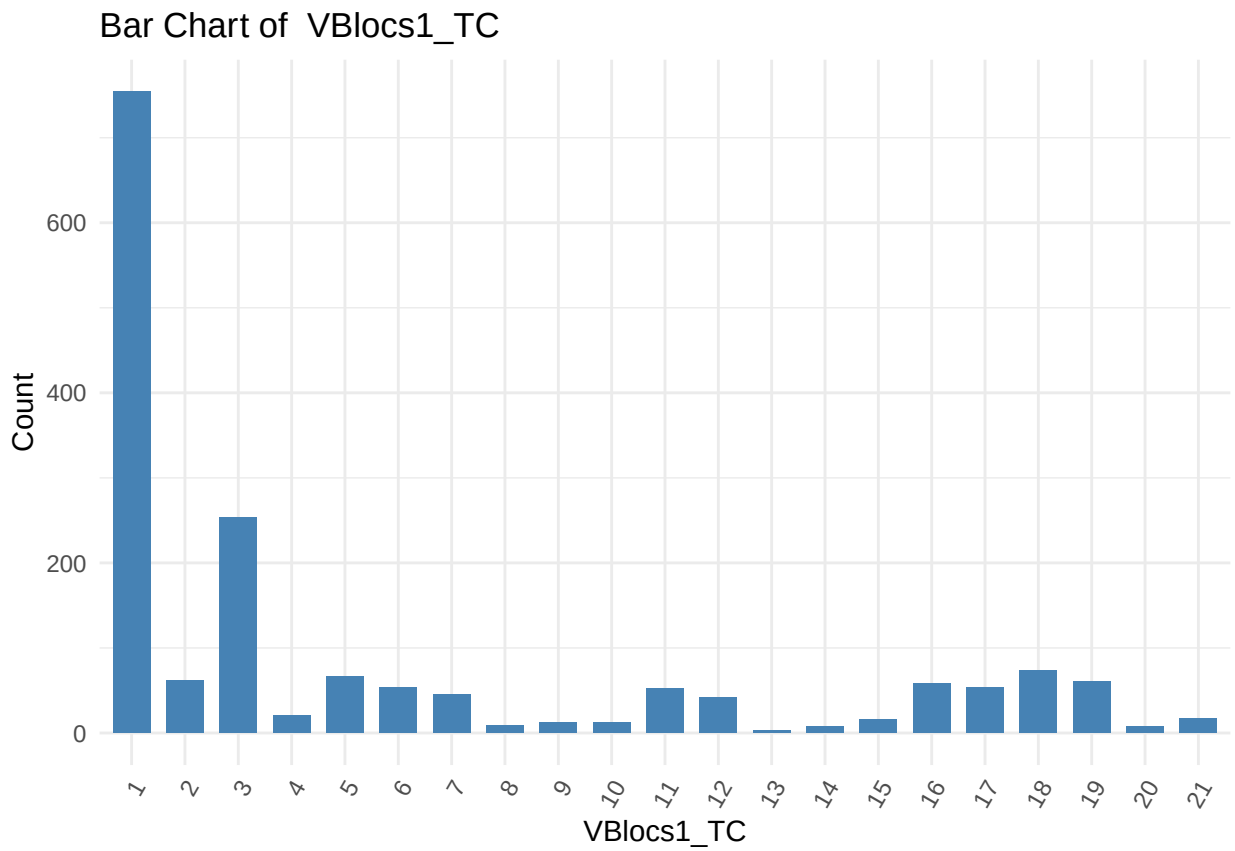

```
## Saving 6.5 x 4.5 in image
```

```
## Warning: ‘guides(<scale> = FALSE)‘ is deprecated. Please use ‘guides(<scale> =
## "none")‘ instead.
```



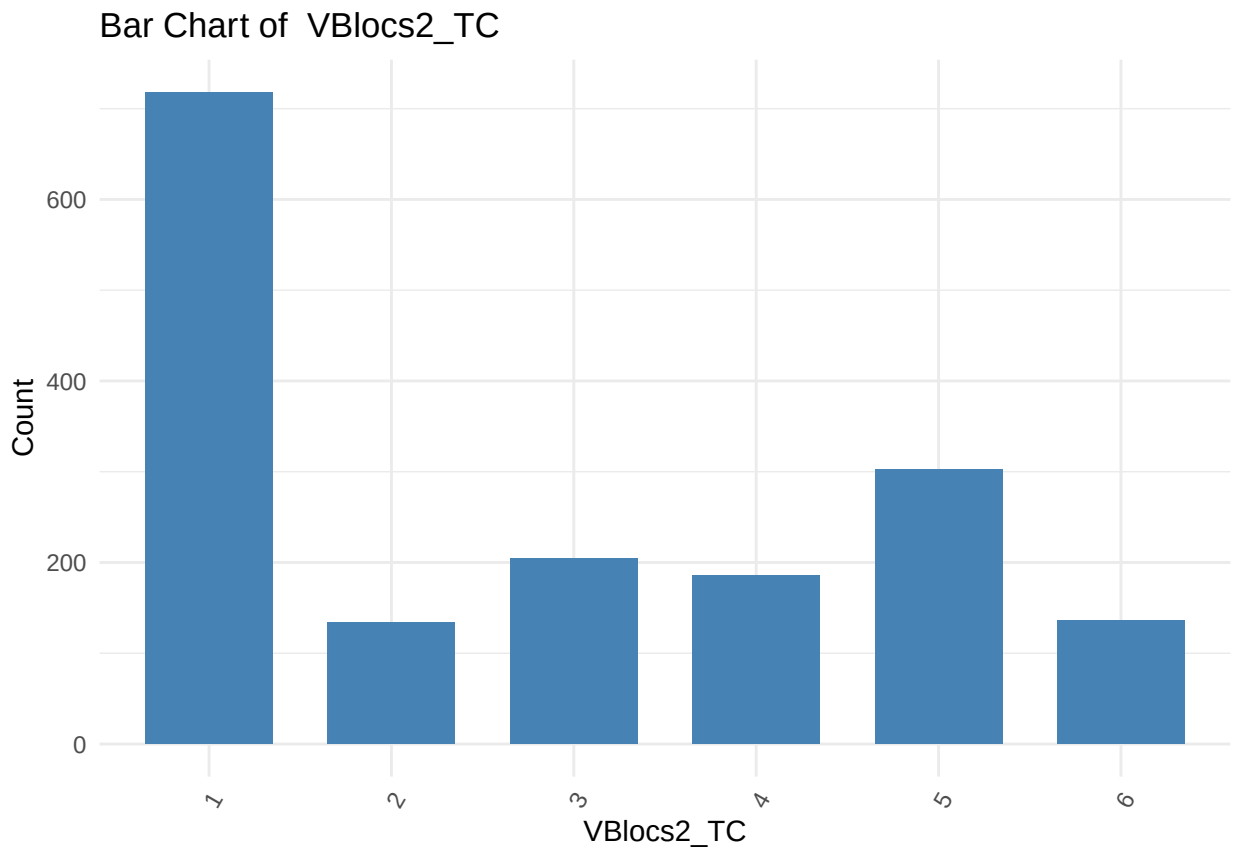

```
## Saving 6.5 x 4.5 in image
```

```
## Warning: ‘guides(<scale> = FALSE)‘ is deprecated. Please use ‘guides(<scale> =
## "none")‘ instead.
```



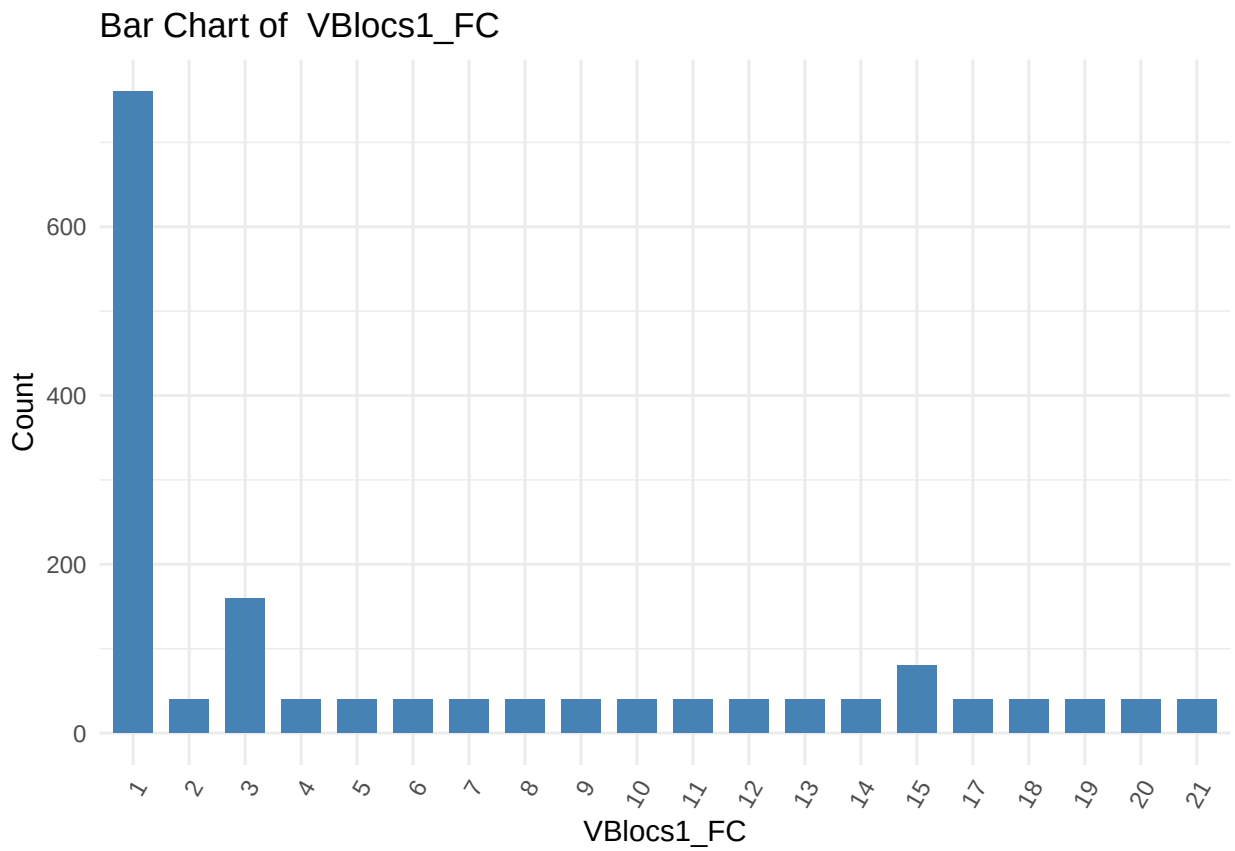

```
## Saving 6.5 x 4.5 in image
```

```
## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =
## "none")` instead.
```



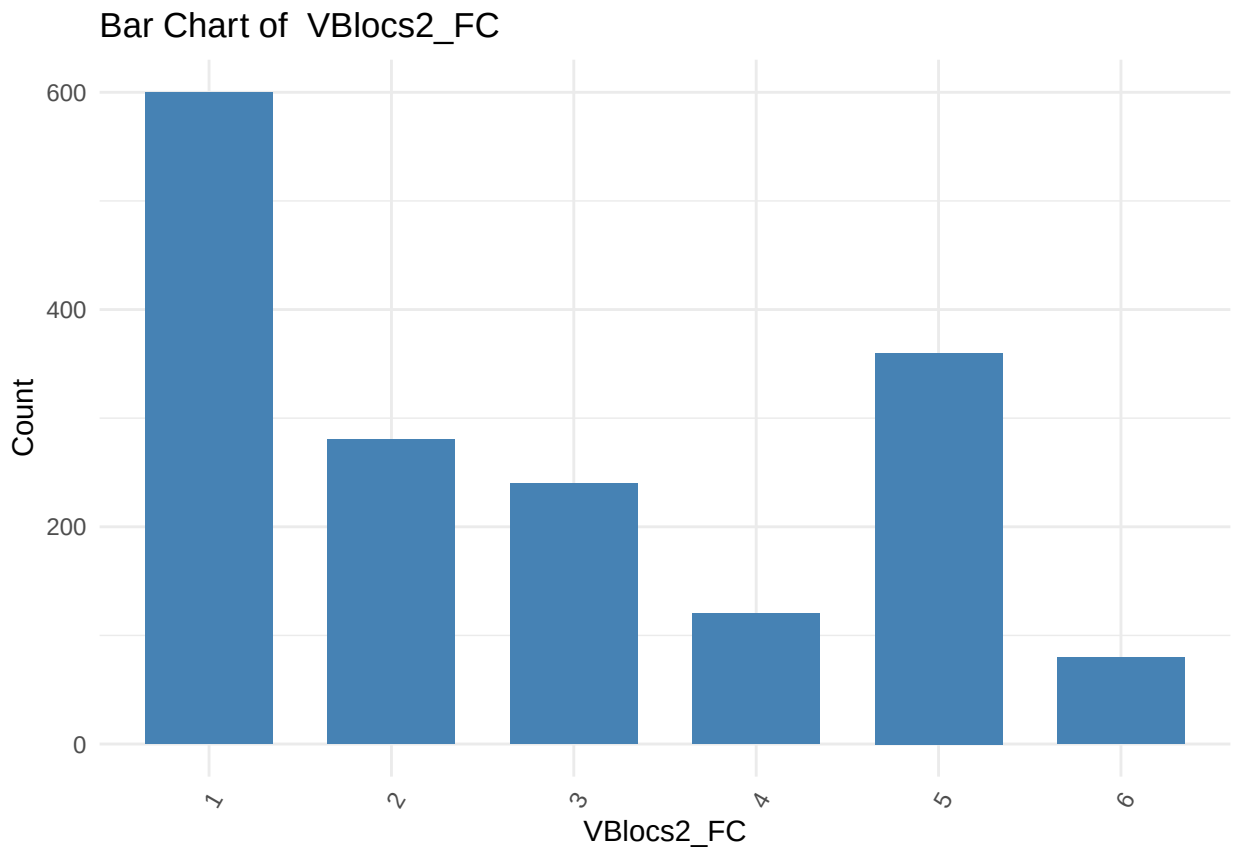

```
## Saving 6.5 x 4.5 in image
```

```
## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =
## "none")` instead.
```



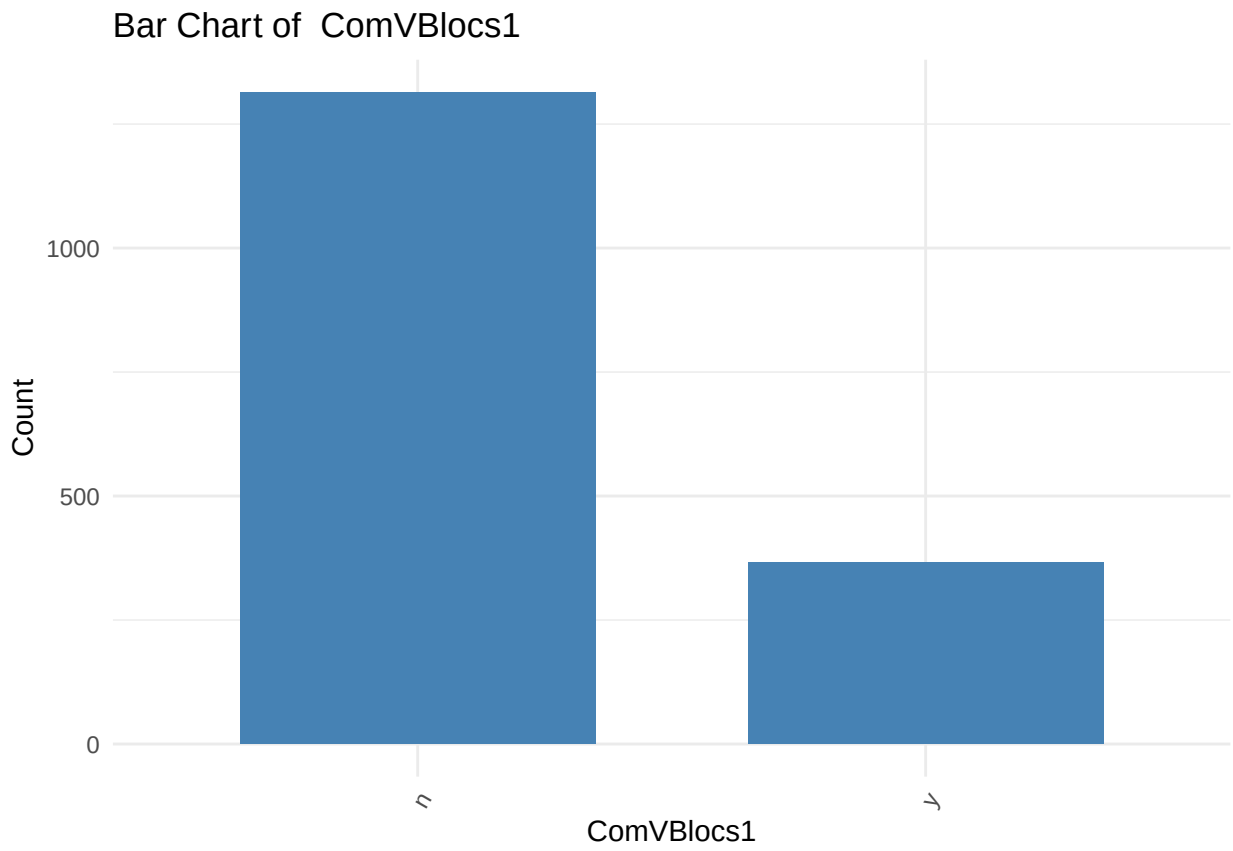

```
## Saving 6.5 x 4.5 in image
```

```
## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =
## "none")` instead.
```

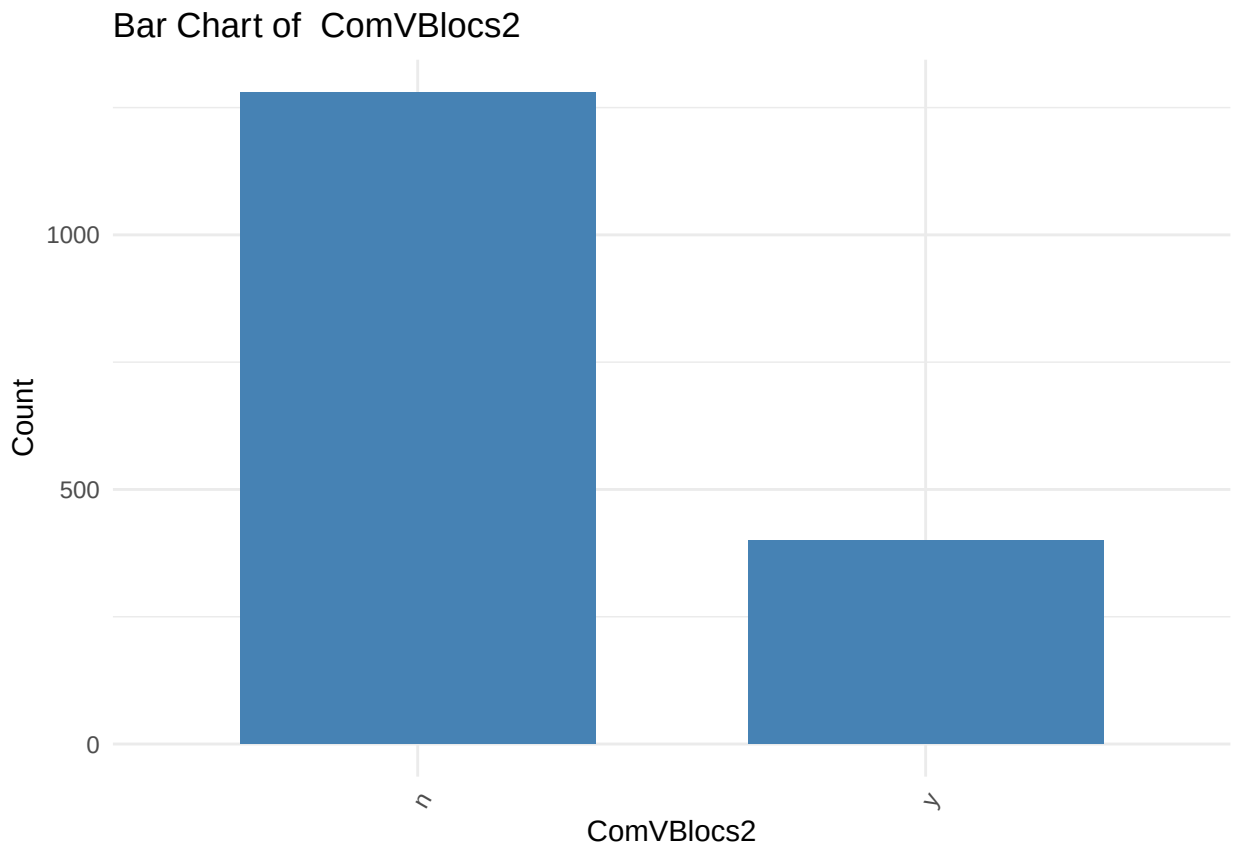



```
## Saving 6.5 x 4.5 in image
```

```
## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =
## "none")` instead.
```

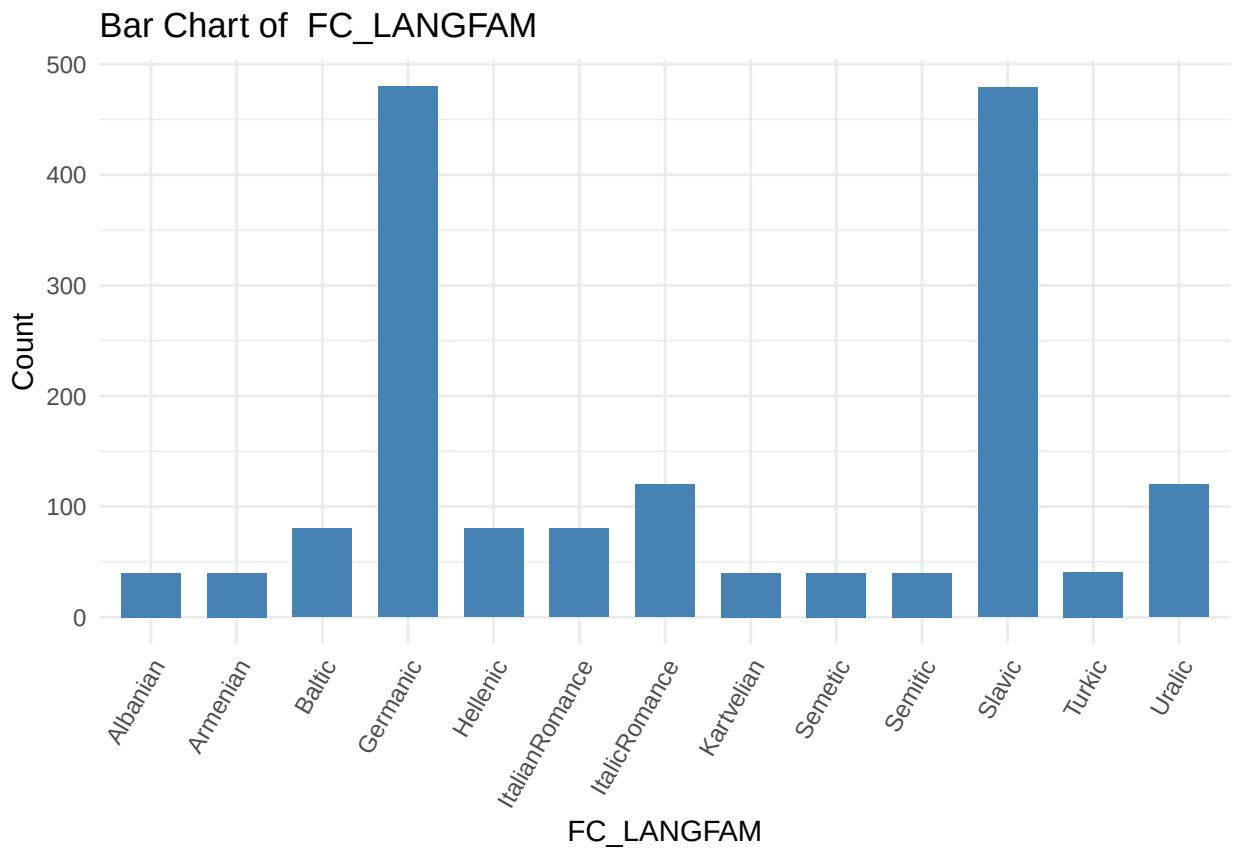



```
## Saving 6.5 x 4.5 in image

## Warning: ‘guides(<scale> = FALSE)‘ is deprecated. Please use ‘guides(<scale> =
## "none")‘ instead.
```



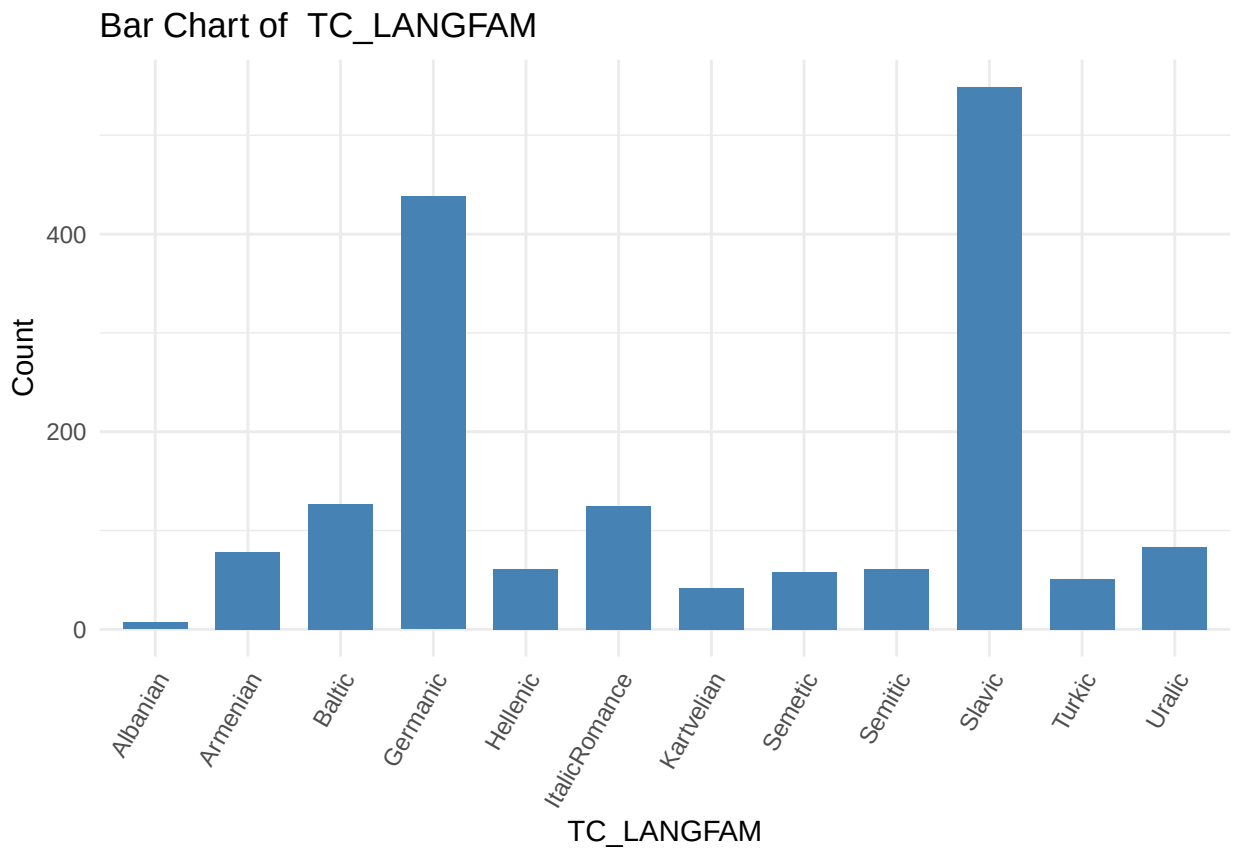

```
## Saving 6.5 x 4.5 in image
```

```
## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =
## "none")` instead.
```



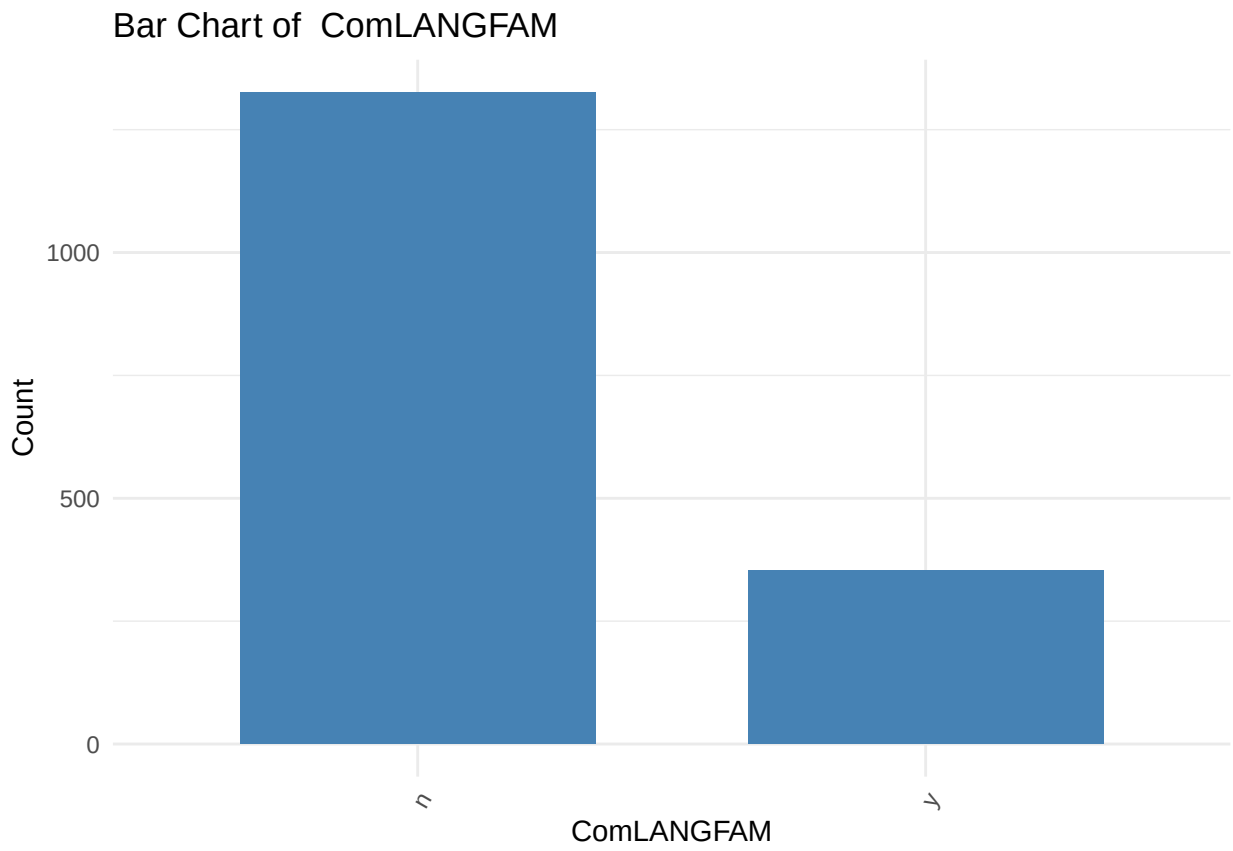

```
## Saving 6.5 x 4.5 in image
```

```
## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =
## "none")` instead.
```

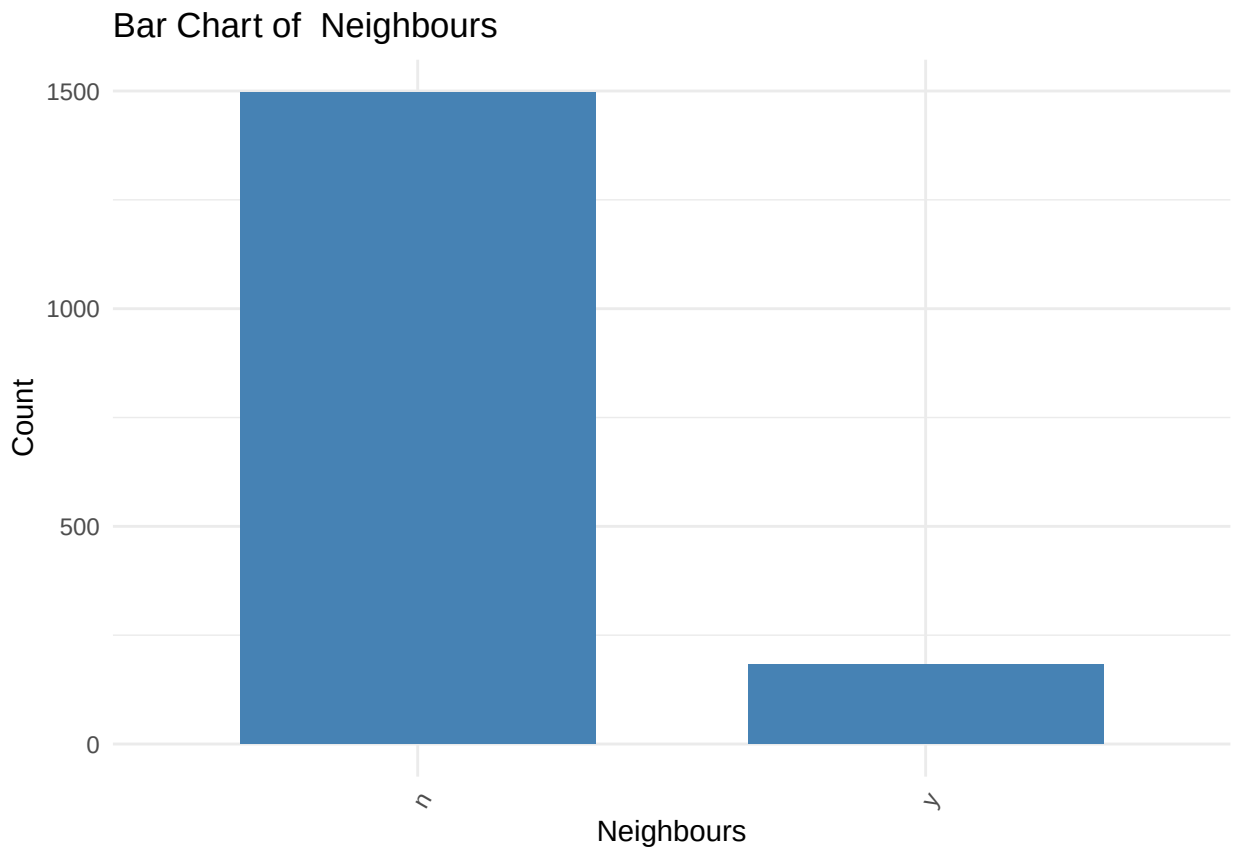



```
## Saving 6.5 x 4.5 in image
```

```
## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =
## "none")` instead.
```

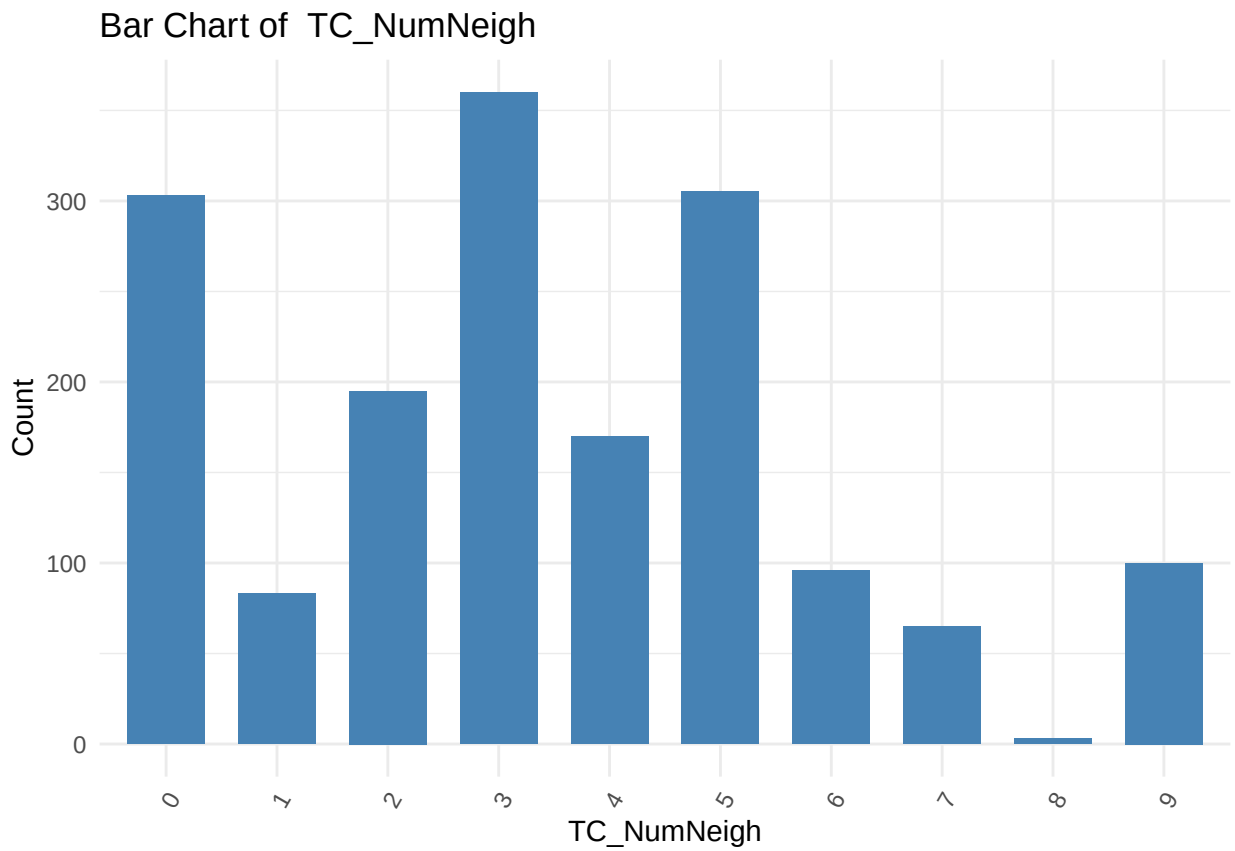



```
## Saving 6.5 x 4.5 in image

## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =
## "none")` instead.
```



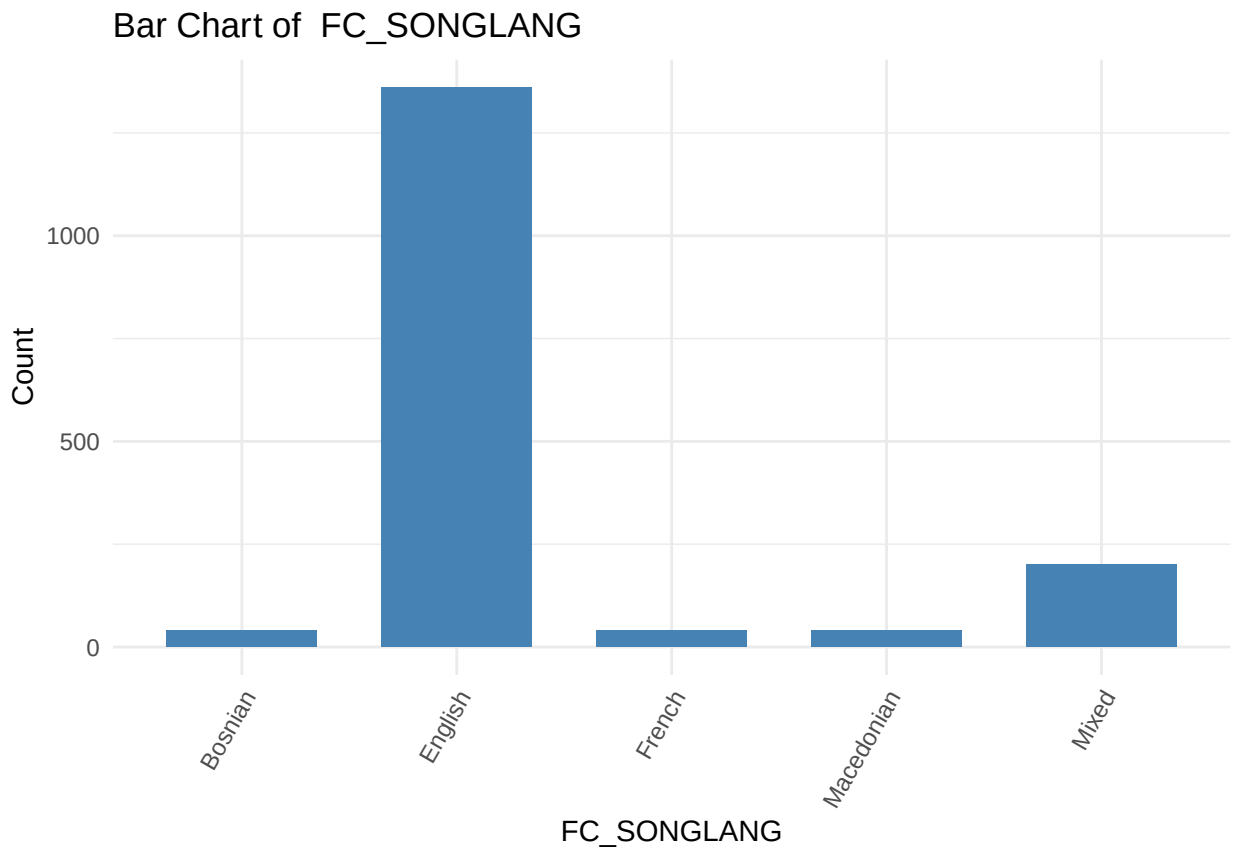

```
## Saving 6.5 x 4.5 in image
```

```
## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =
## "none")` instead.
```



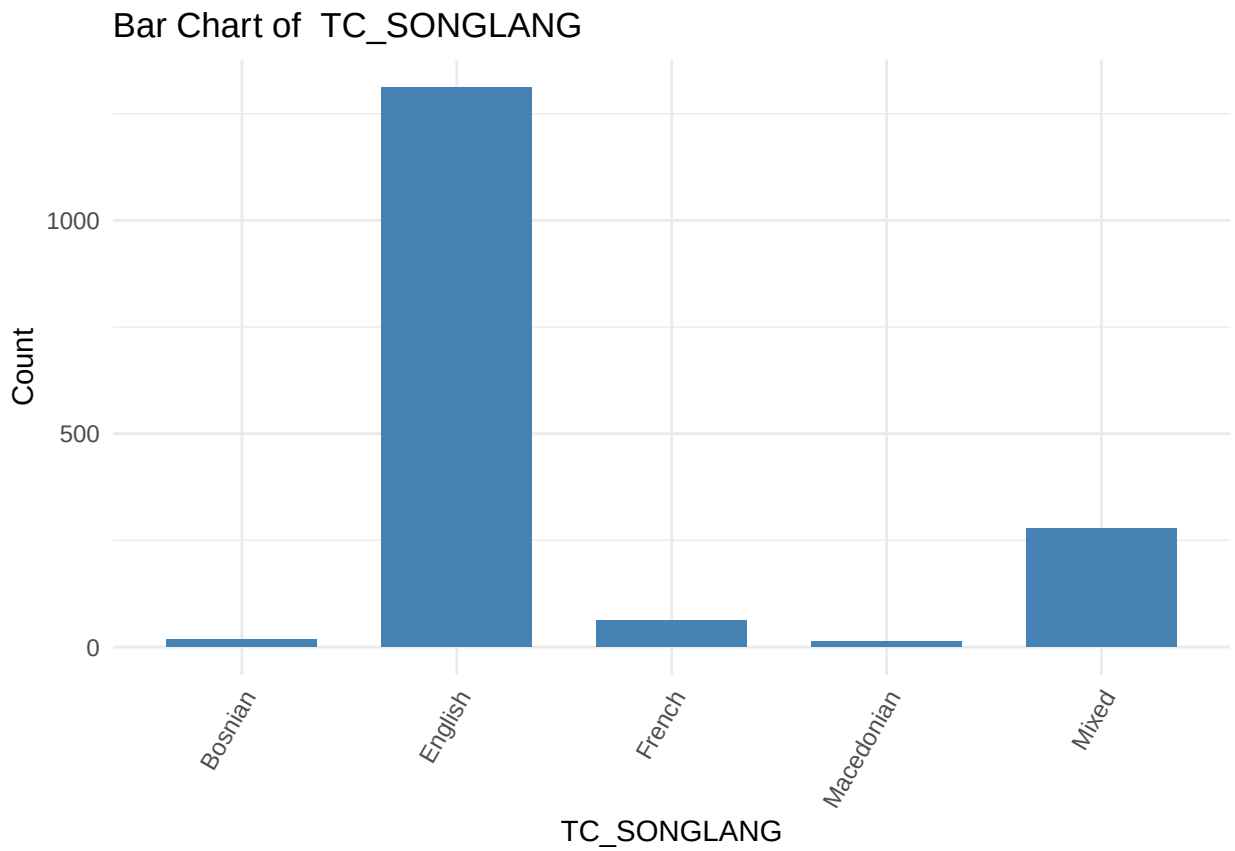

```
## Saving 6.5 x 4.5 in image
```

```
## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =
## "none")` instead.
```



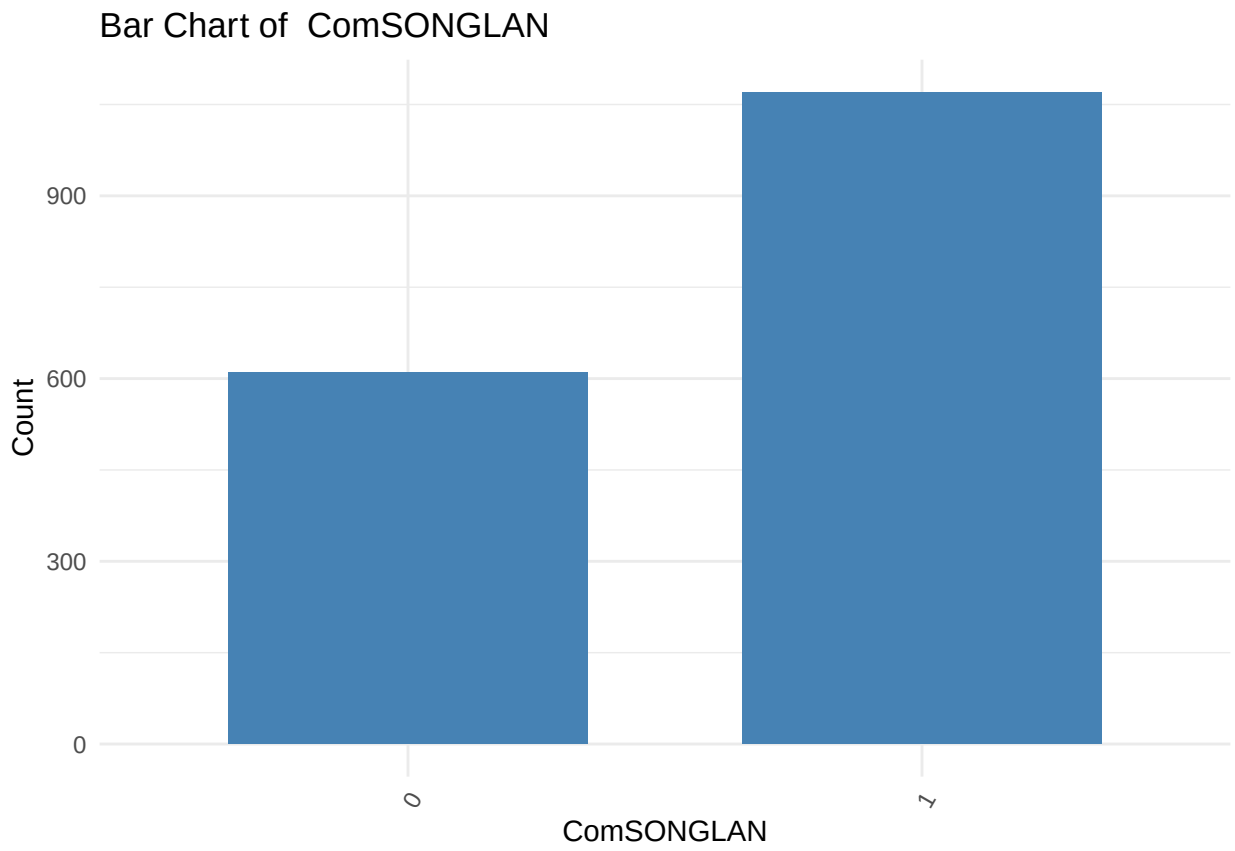

```
## Saving 6.5 x 4.5 in image
```

```
## Warning: ‘guides(<scale> = FALSE)‘ is deprecated. Please use ‘guides(<scale> =
## "none")‘ instead.
```

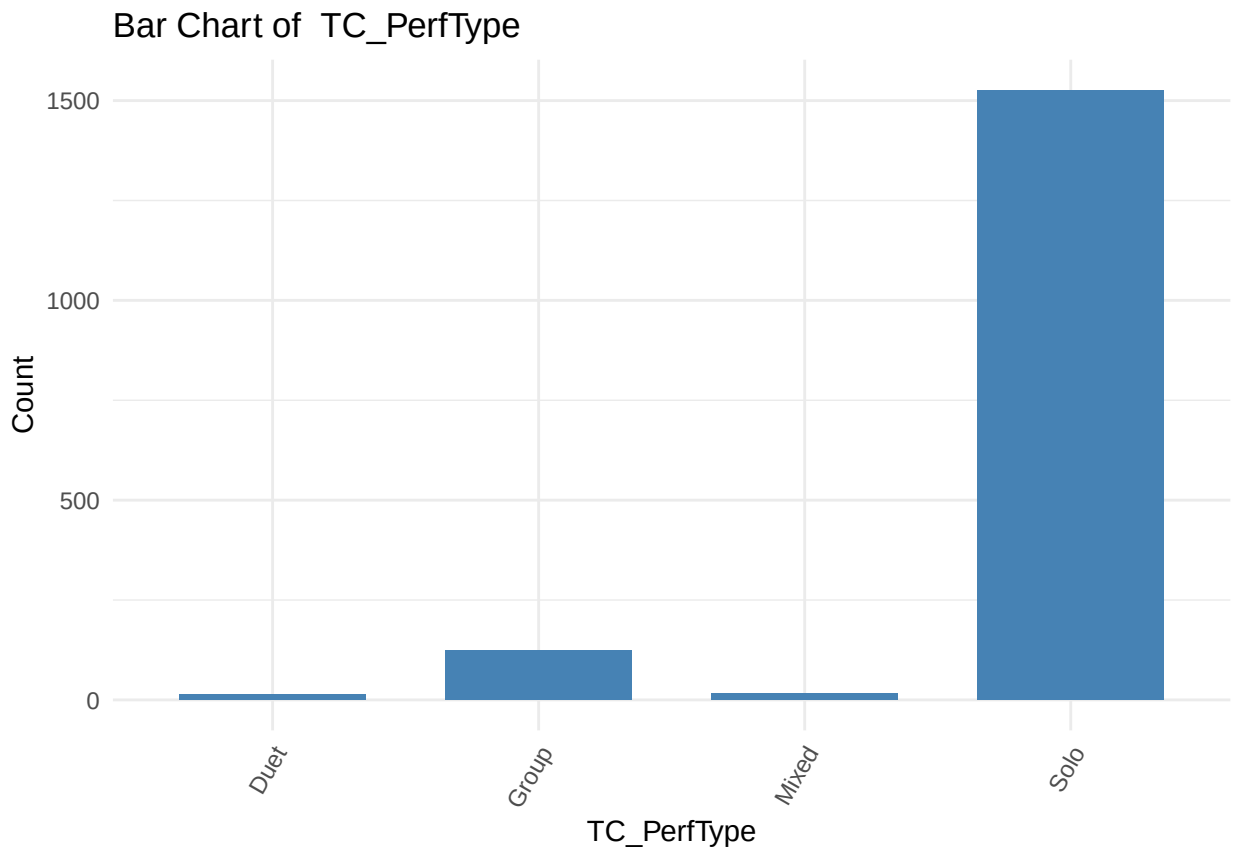



```
## Saving 6.5 x 4.5 in image
```

```
## Warning: ‘guides(<scale> = FALSE)‘ is deprecated. Please use ‘guides(<scale> =
## "none")‘ instead.
```

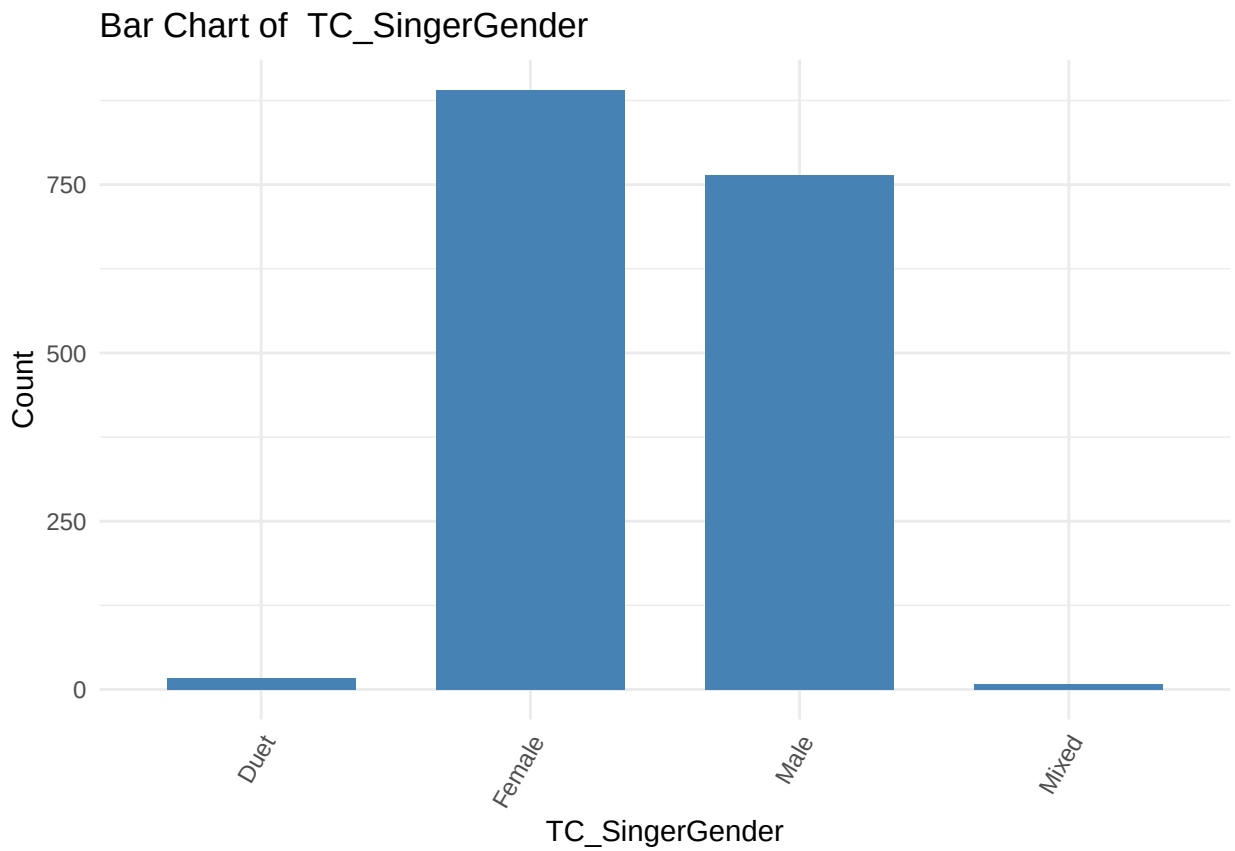



```
## Saving 6.5 x 4.5 in image

## Warning: ‘guides(<scale> = FALSE)‘ is deprecated. Please use ‘guides(<scale> =
## "none")‘ instead.
```

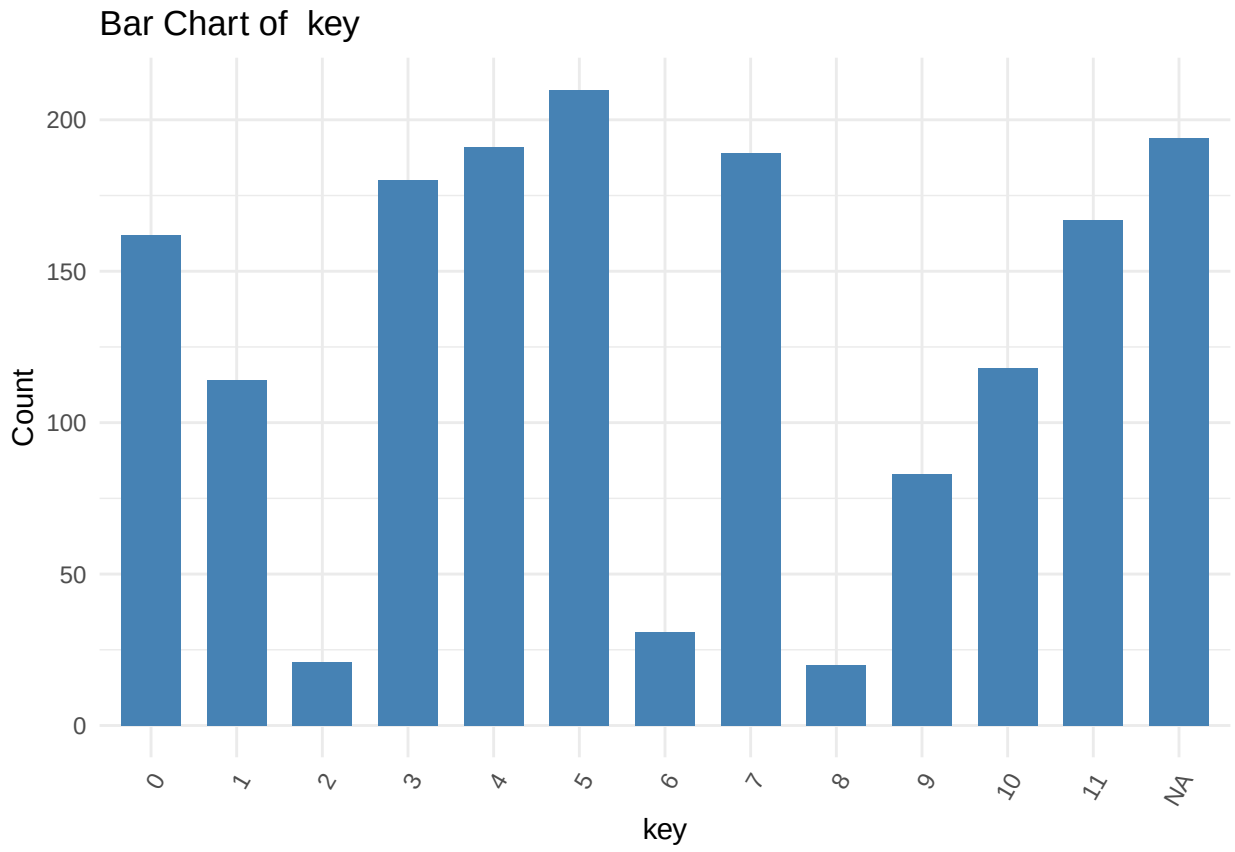



```
## Saving 6.5 x 4.5 in image
```

```
## Warning: ‘guides(<scale> = FALSE)‘ is deprecated. Please use ‘guides(<scale> =
## "none")‘ instead.
```

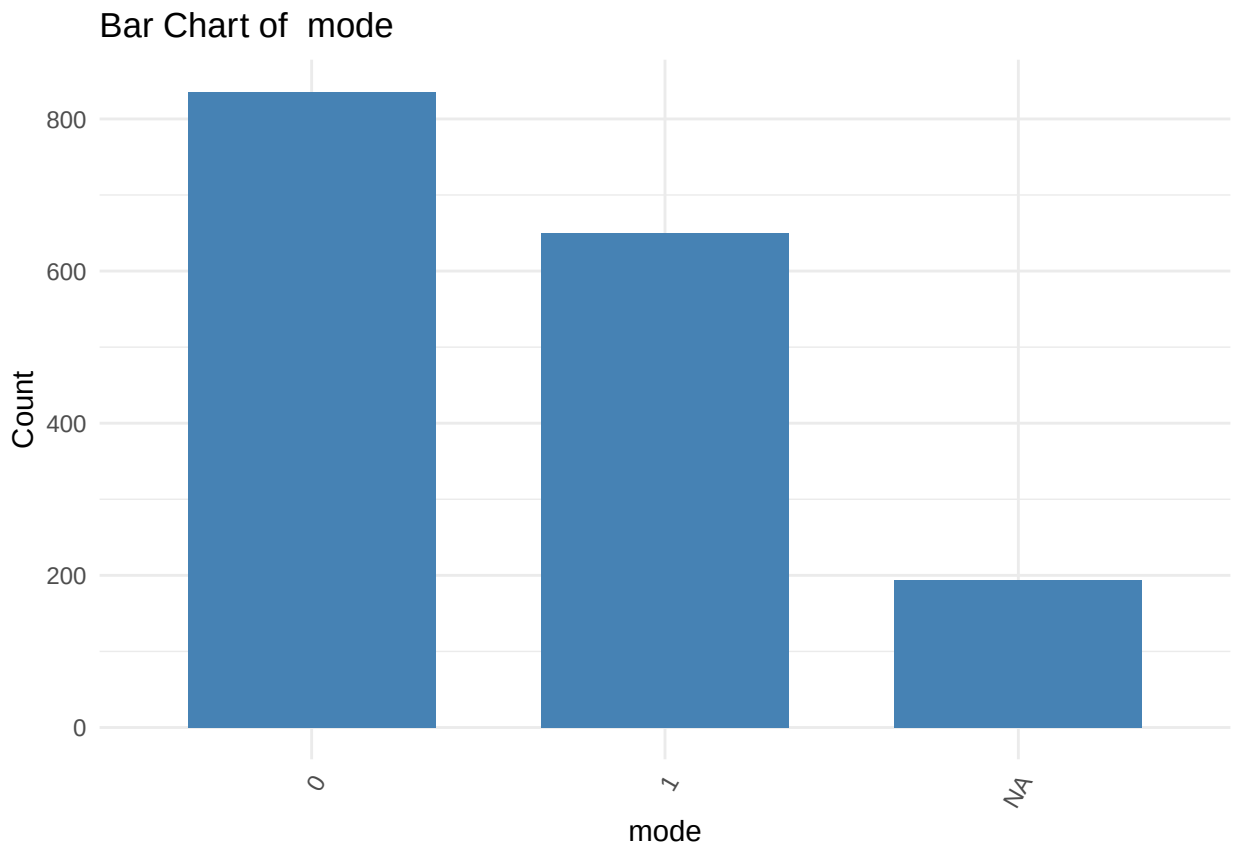



```
## Saving 6.5 x 4.5 in image

## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =
## "none")` instead.
```

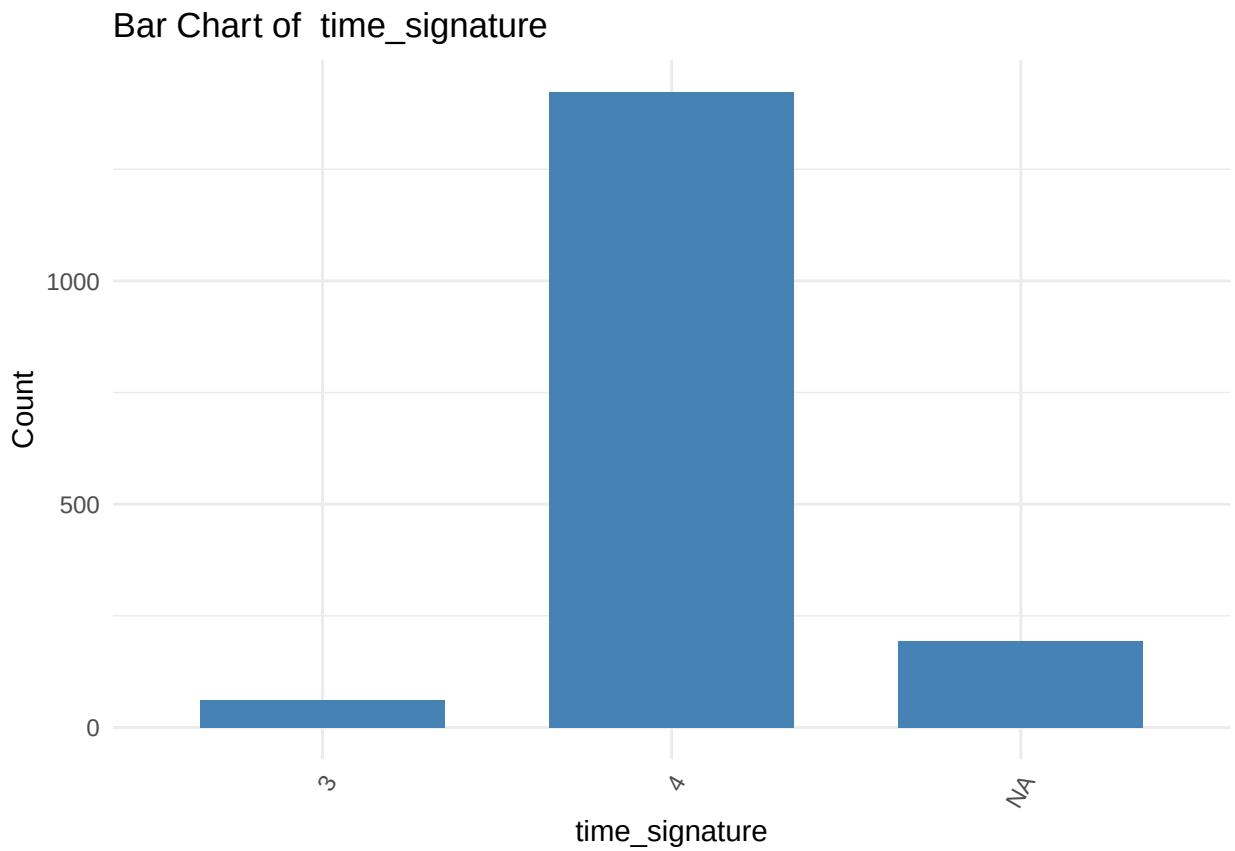



```
## Saving 6.5 x 4.5 in image
```

```
## Warning: ‘guides(<scale> = FALSE)‘ is deprecated. Please use ‘guides(<scale> =
## "none")‘ instead.
```



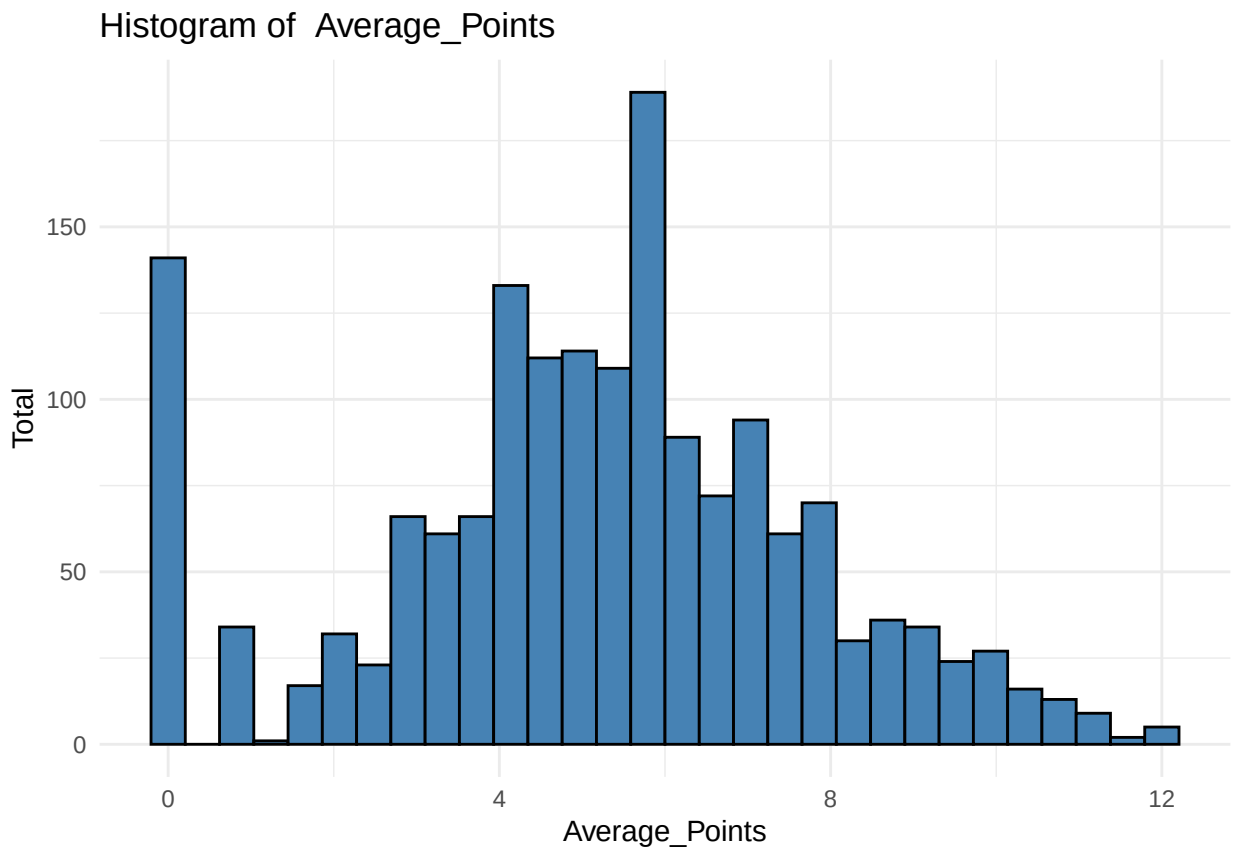

```
## Saving 6.5 x 4.5 in image
```




```
## [1] 0
```

```
# call histogram plotting function
plot_histogram(dataset = ESCdata, col_names = all_num, output_dir = './report/plots/histograms')
```

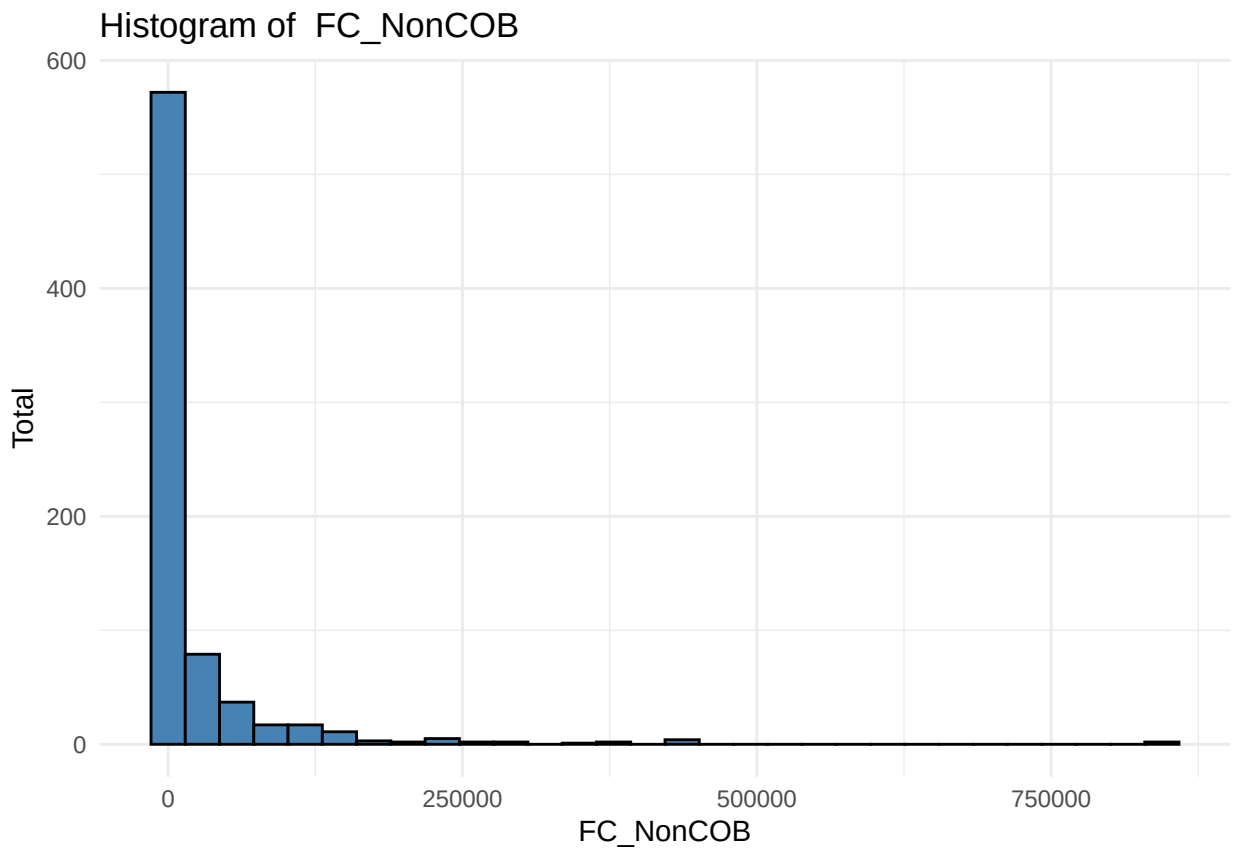
```
## Saving 6.5 x 4.5 in image
```



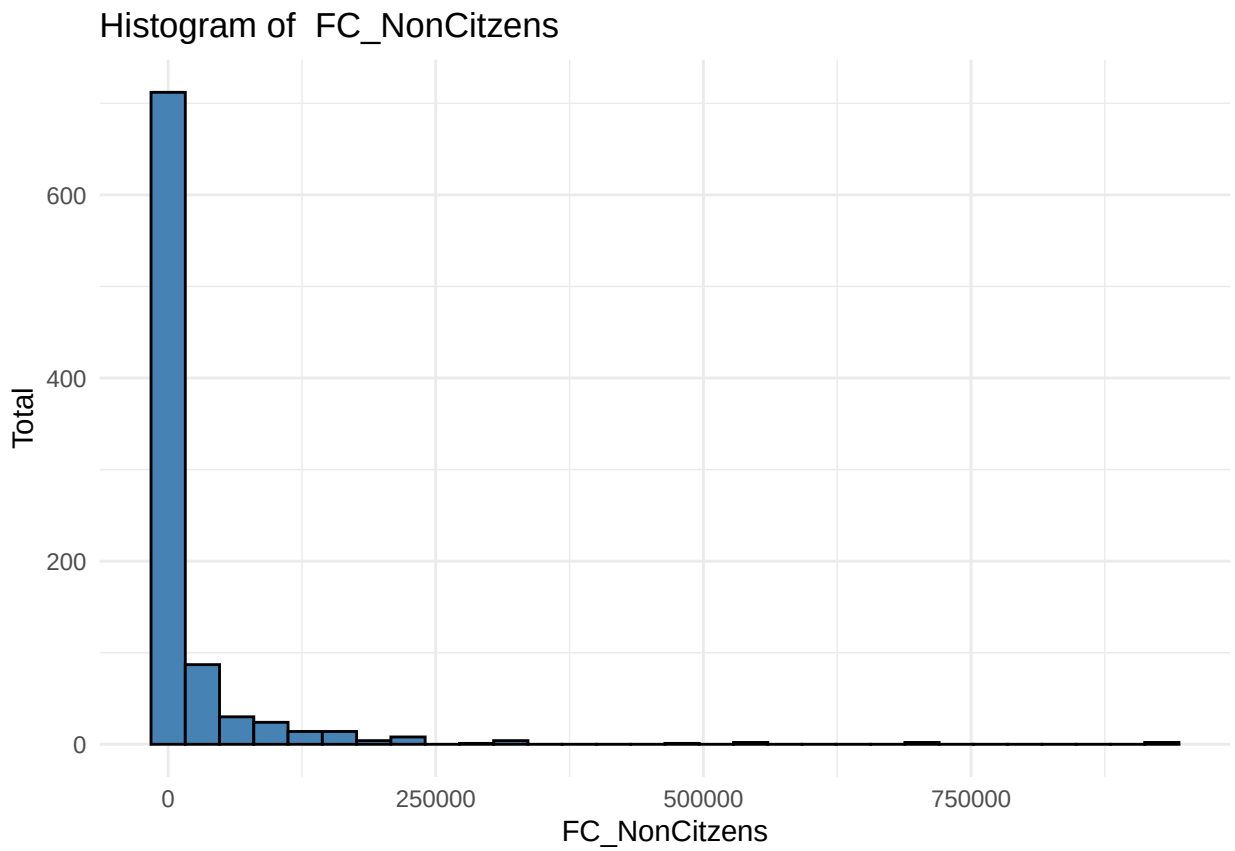

```
## Warning: Removed 924 rows containing non-finite values (stat_bin).
```

```
## Saving 6.5 x 4.5 in image
```

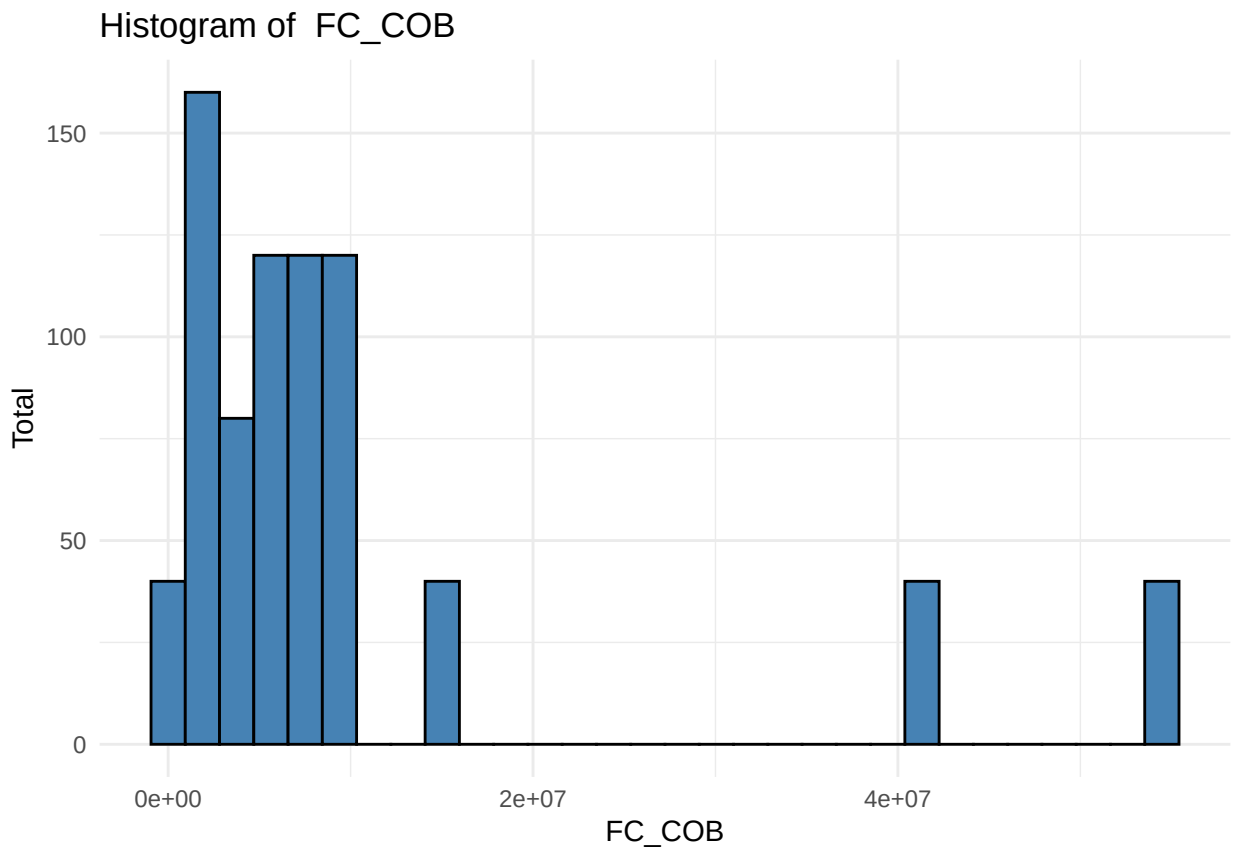
```
## Warning: Removed 924 rows containing non-finite values (stat_bin).
```



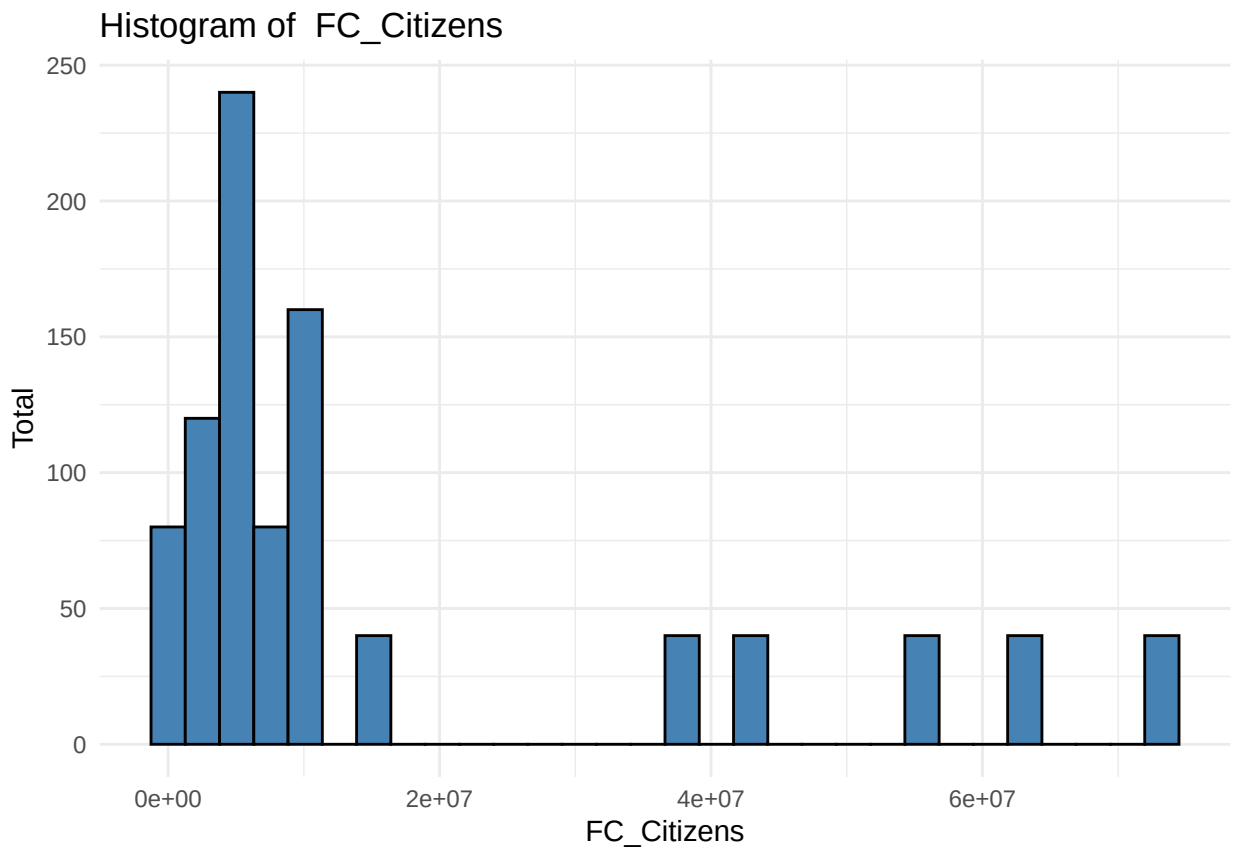

```
## Warning: Removed 775 rows containing non-finite values (stat_bin).

## Saving 6.5 x 4.5 in image

## Warning: Removed 775 rows containing non-finite values (stat_bin).
```



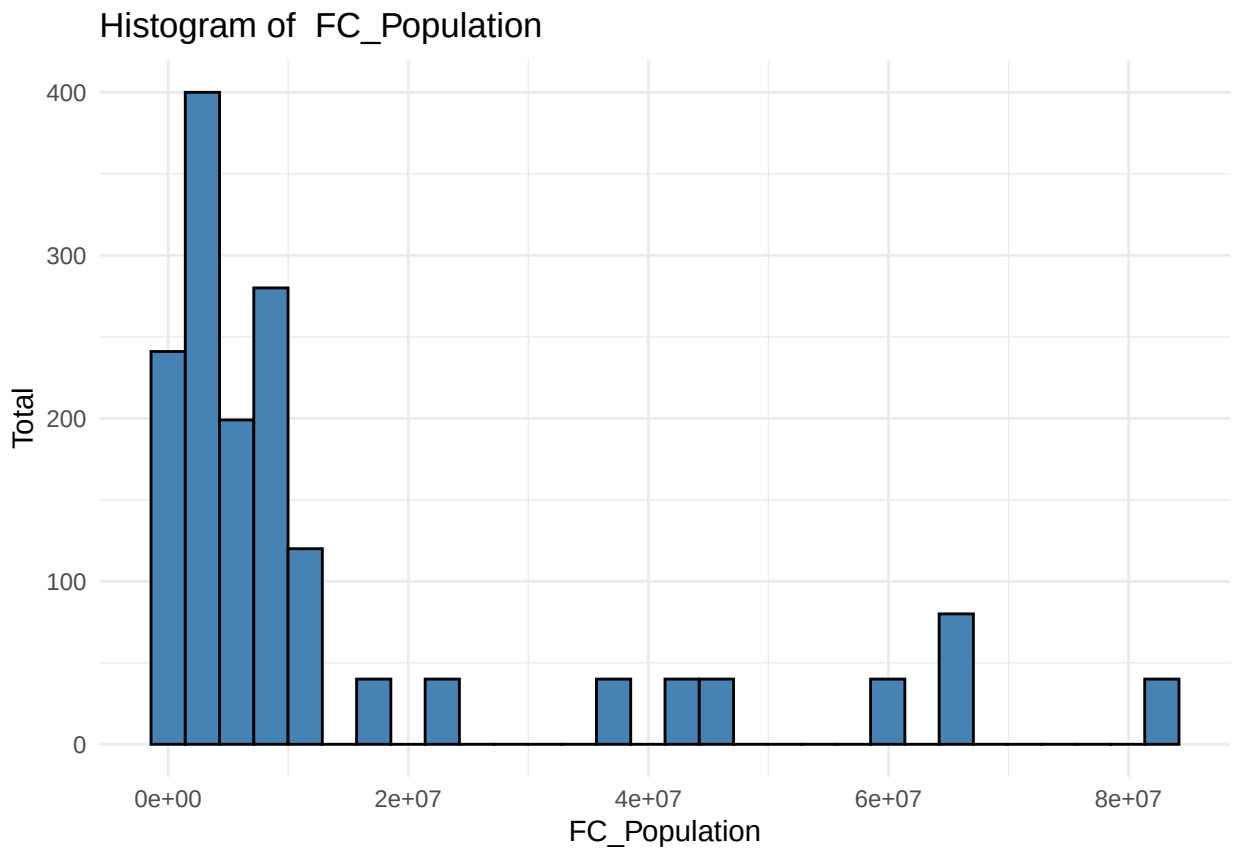

```
## Warning: Removed 920 rows containing non-finite values (stat_bin).

## Saving 6.5 x 4.5 in image

## Warning: Removed 920 rows containing non-finite values (stat_bin).
```




```
## Warning: Removed 760 rows containing non-finite values (stat_bin).

## Saving 6.5 x 4.5 in image

## Warning: Removed 760 rows containing non-finite values (stat_bin).
```



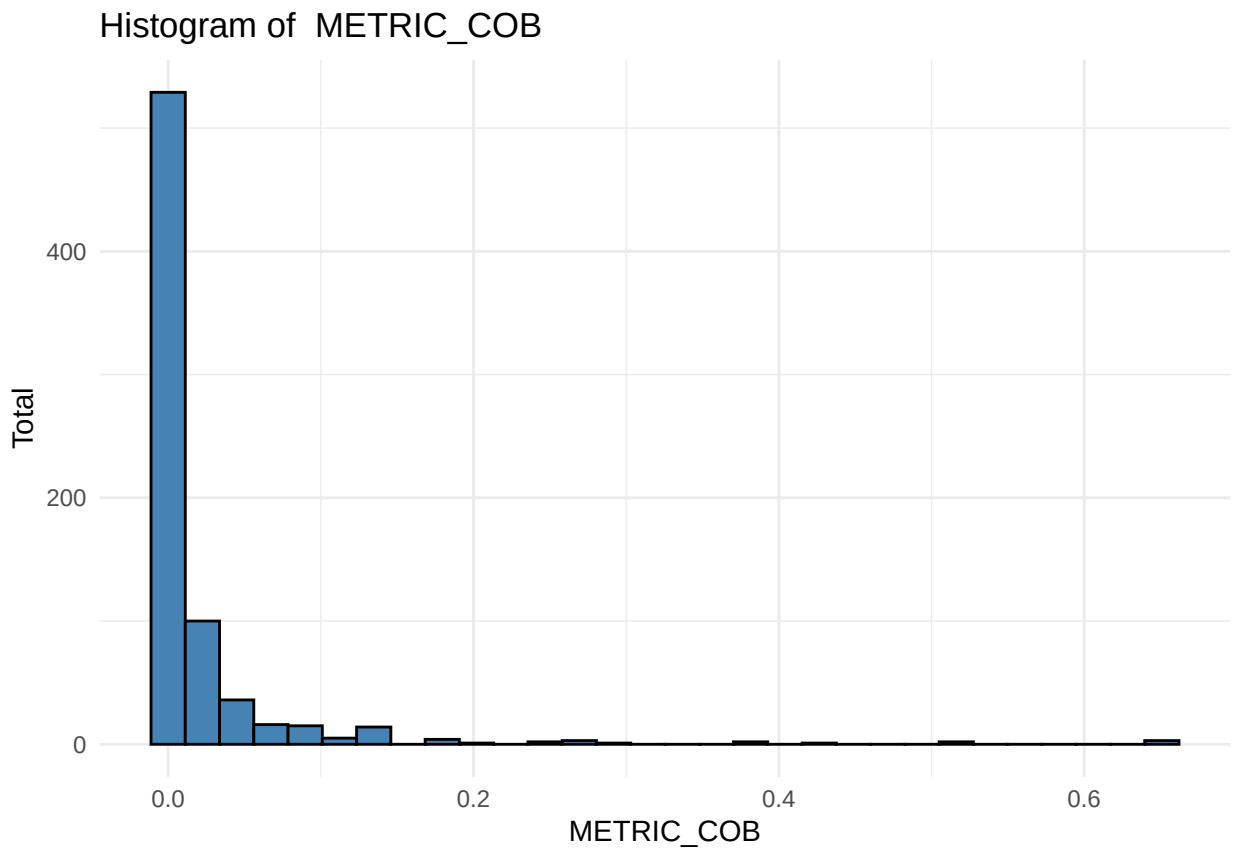

```
## Warning: Removed 80 rows containing non-finite values (stat_bin).

## Saving 6.5 x 4.5 in image

## Warning: Removed 80 rows containing non-finite values (stat_bin).
```



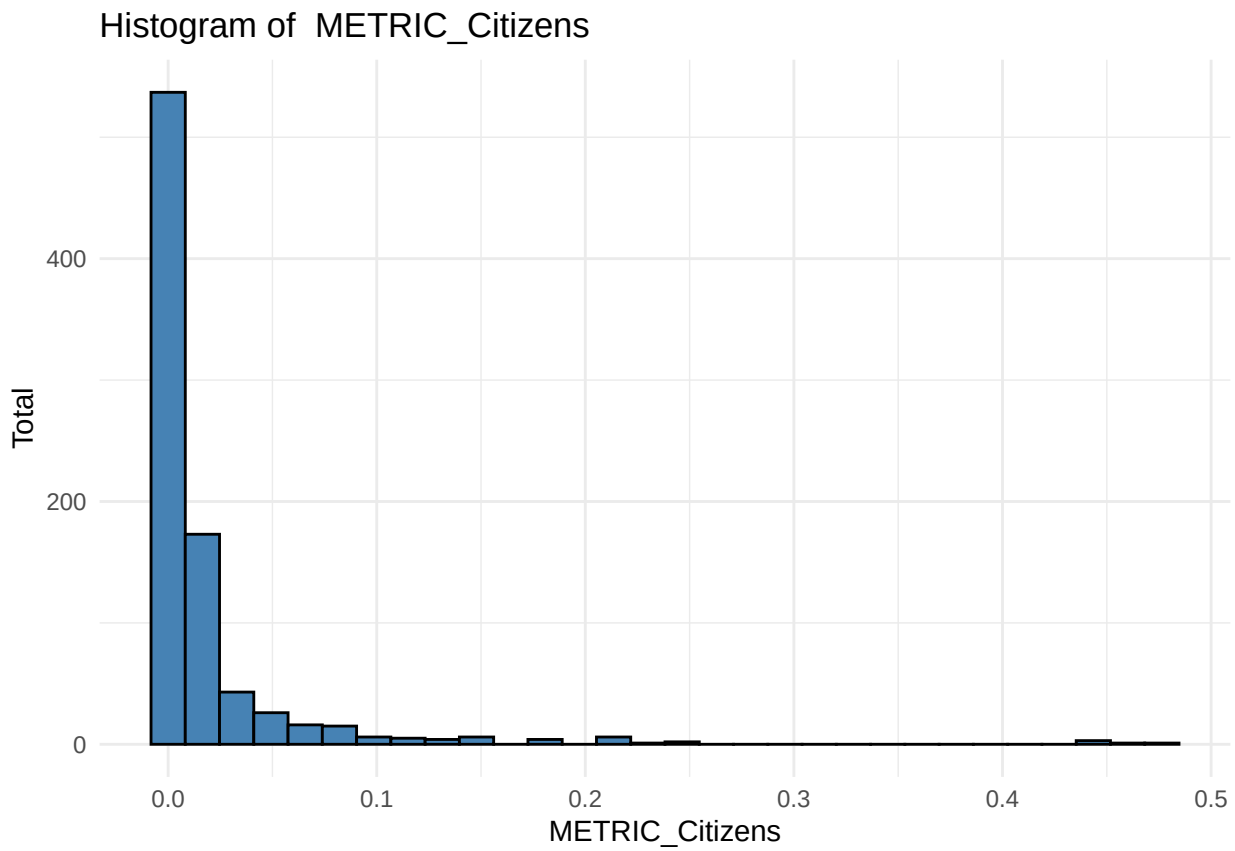

```
## Warning: Removed 946 rows containing non-finite values (stat_bin).
```

```
## Saving 6.5 x 4.5 in image
```

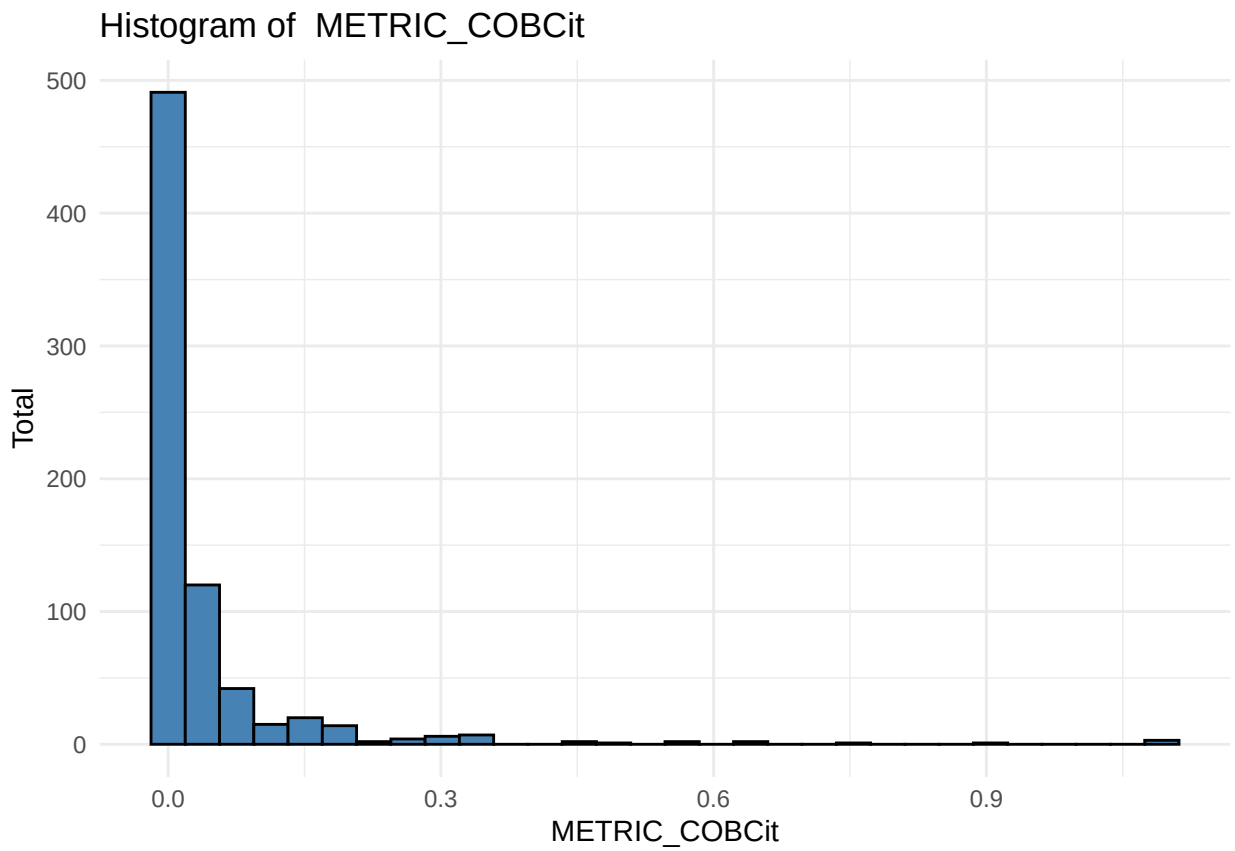
```
## Warning: Removed 946 rows containing non-finite values (stat_bin).
```



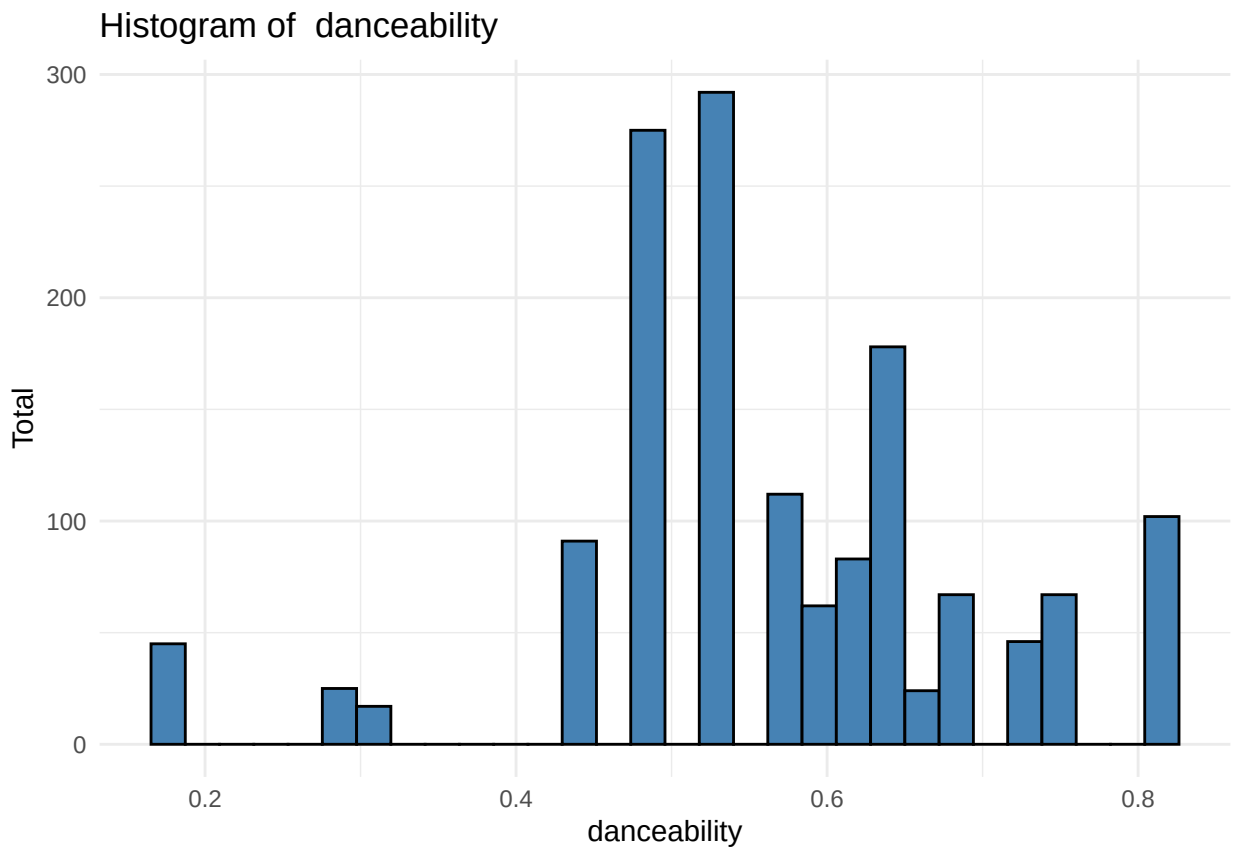

```
## Warning: Removed 831 rows containing non-finite values (stat_bin).

## Saving 6.5 x 4.5 in image

## Warning: Removed 831 rows containing non-finite values (stat_bin).
```



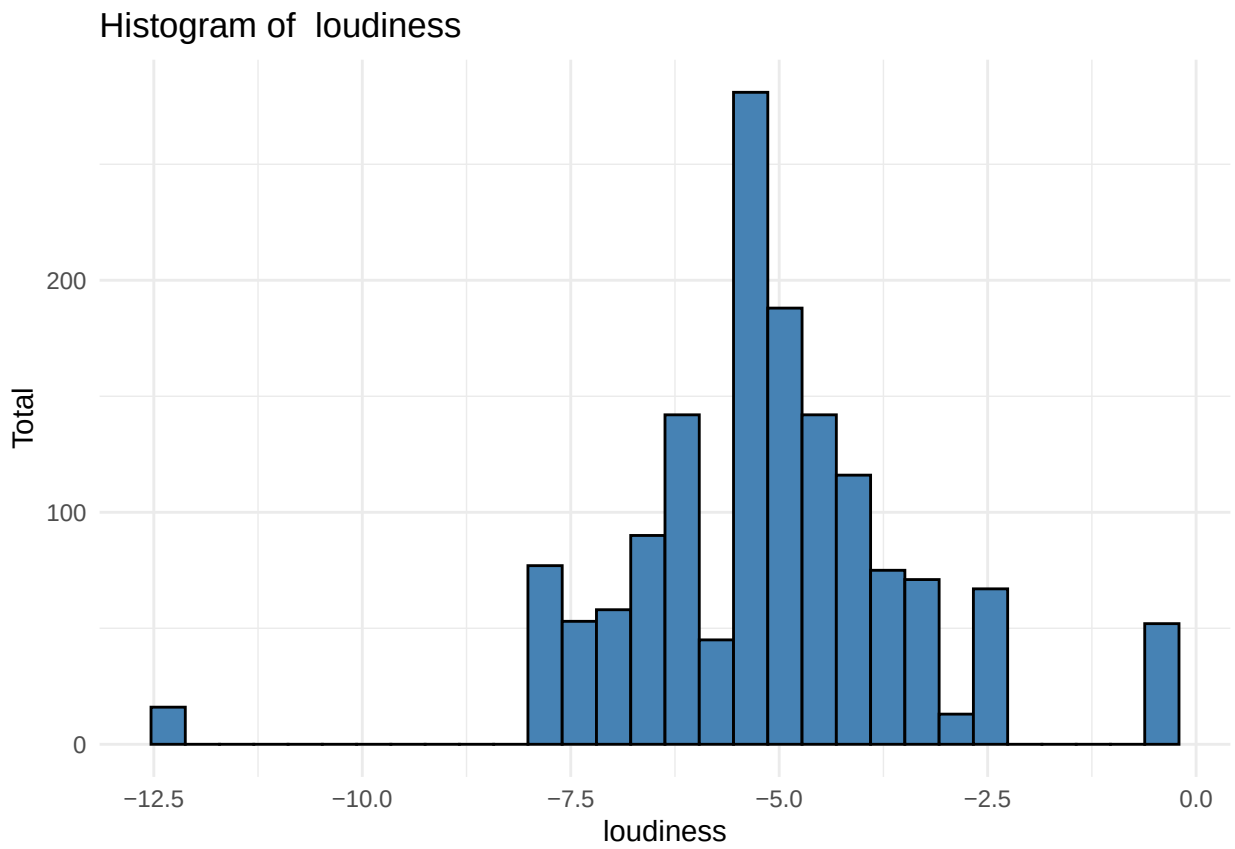

```
## Warning: Removed 947 rows containing non-finite values (stat_bin).

## Saving 6.5 x 4.5 in image

## Warning: Removed 947 rows containing non-finite values (stat_bin).
```



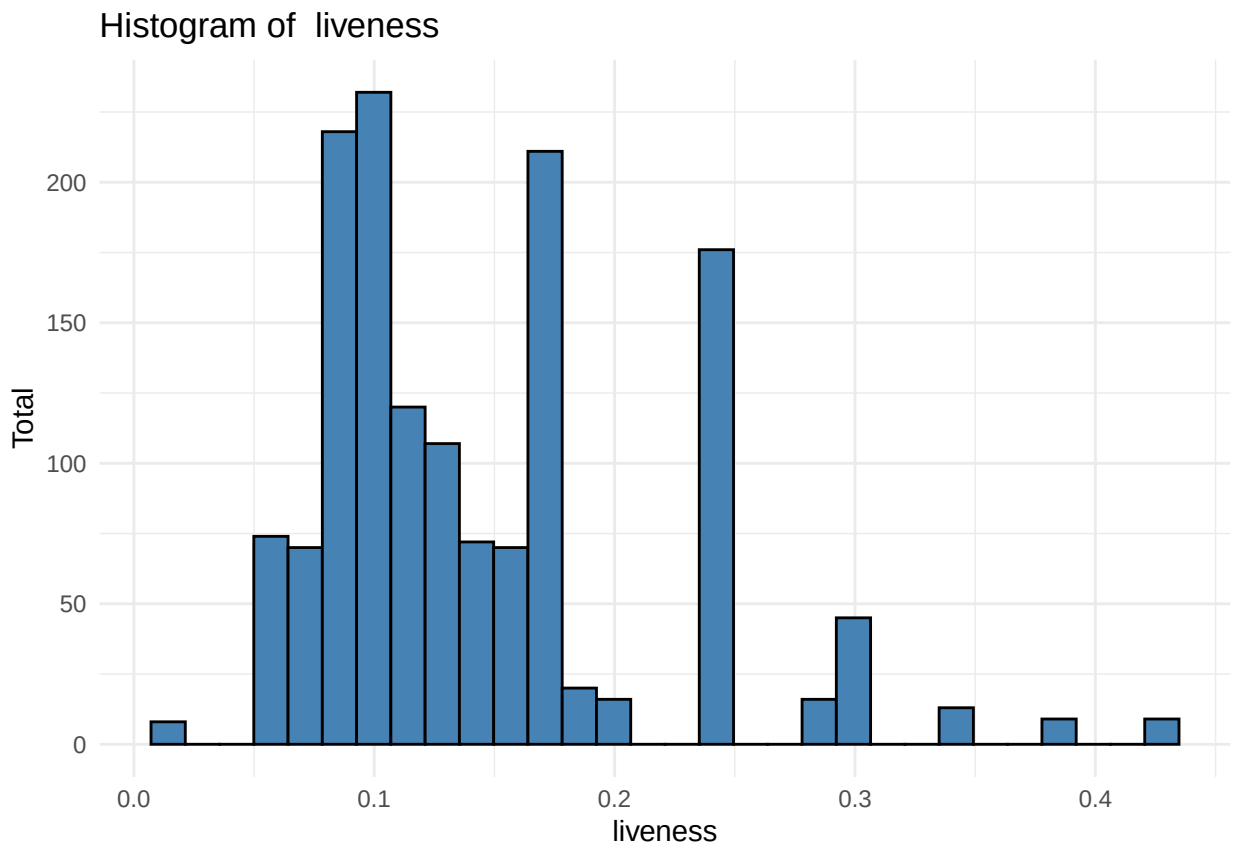

```
## Warning: Removed 194 rows containing non-finite values (stat_bin).

## Saving 6.5 x 4.5 in image

## Warning: Removed 194 rows containing non-finite values (stat_bin).
```



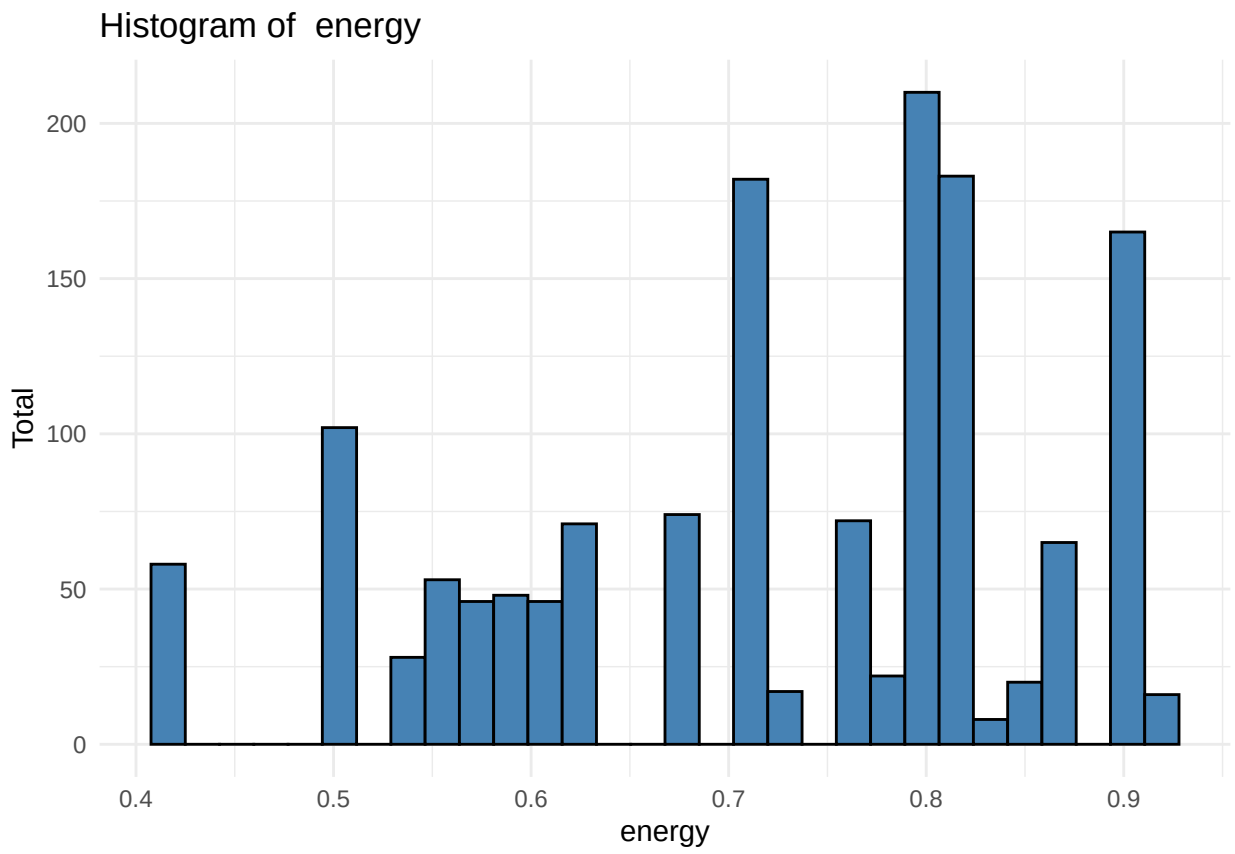

```
## Warning: Removed 194 rows containing non-finite values (stat_bin).

## Saving 6.5 x 4.5 in image

## Warning: Removed 194 rows containing non-finite values (stat_bin).
```



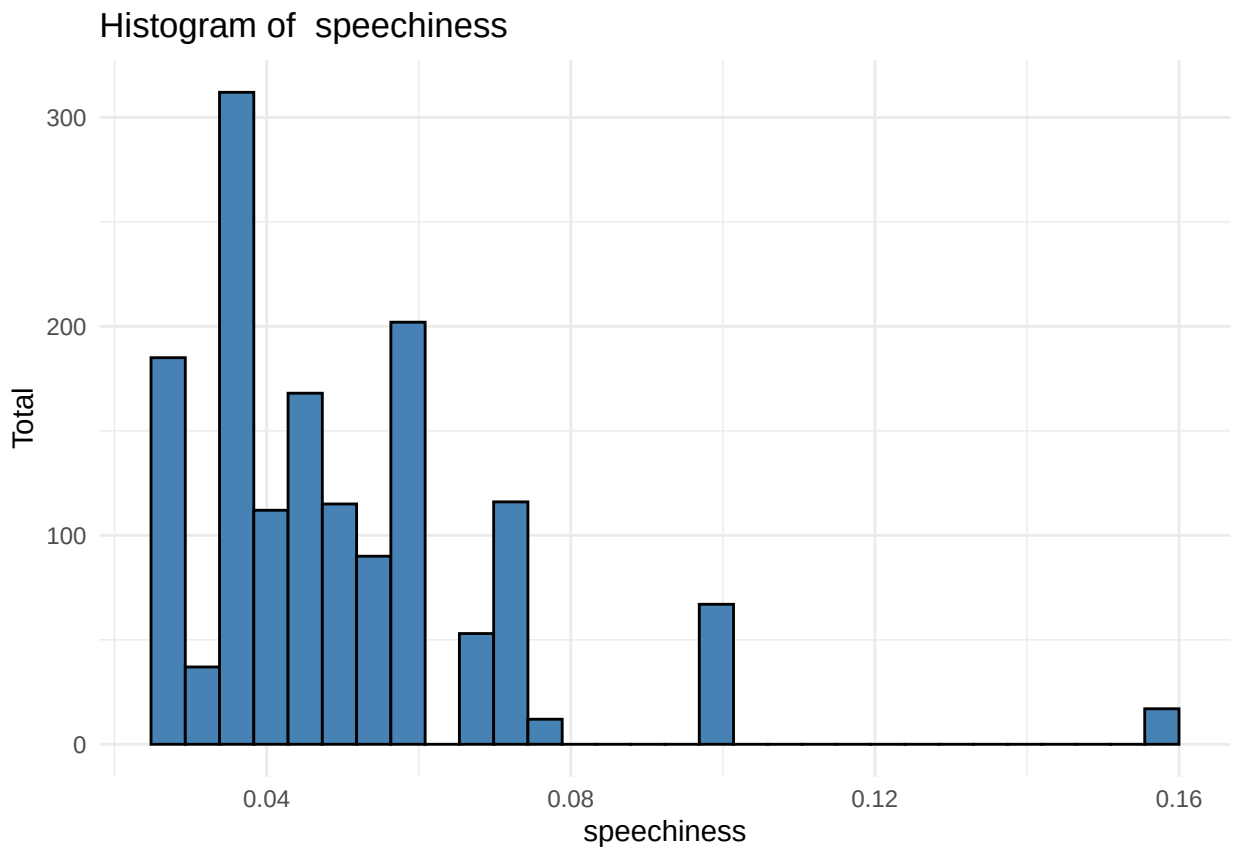

```
## Warning: Removed 194 rows containing non-finite values (stat_bin).

## Saving 6.5 x 4.5 in image

## Warning: Removed 194 rows containing non-finite values (stat_bin).
```



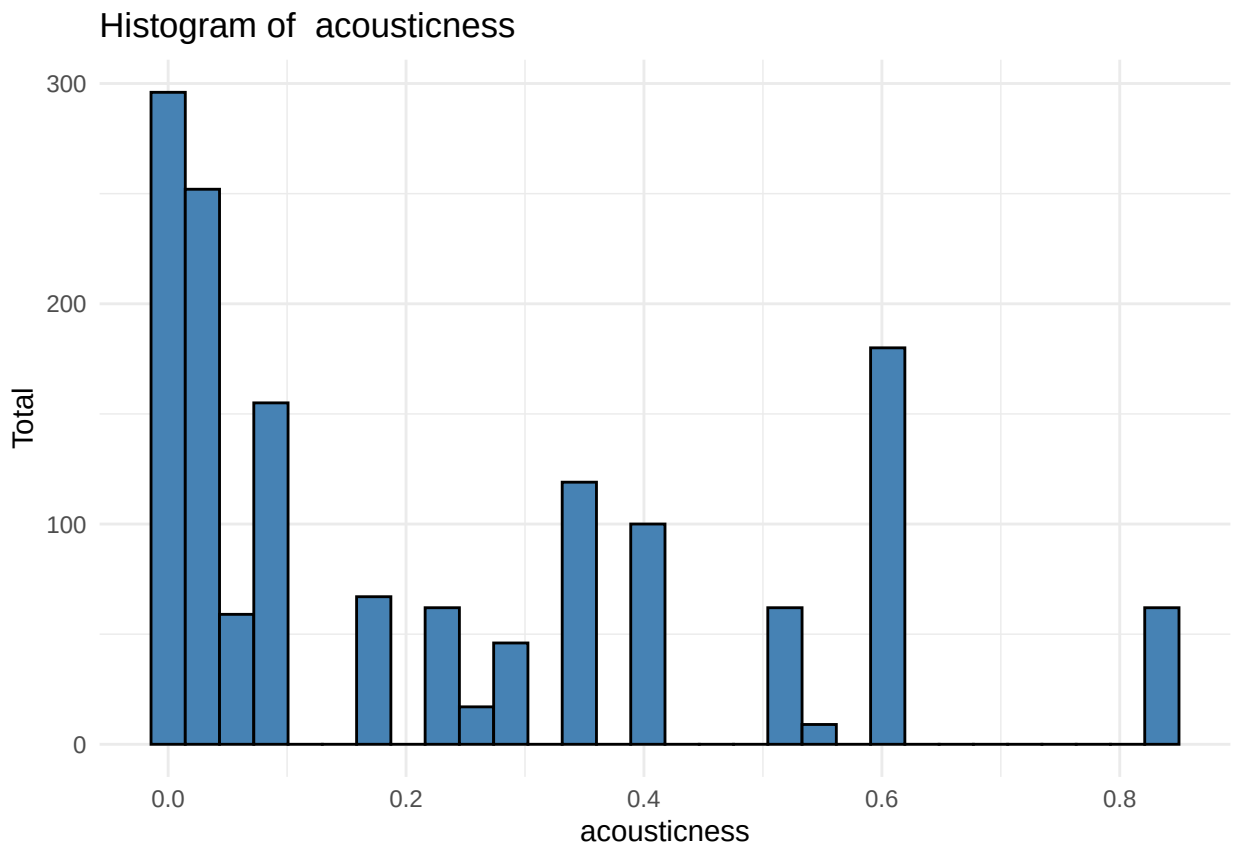

```
## Warning: Removed 194 rows containing non-finite values (stat_bin).

## Saving 6.5 x 4.5 in image

## Warning: Removed 194 rows containing non-finite values (stat_bin).
```



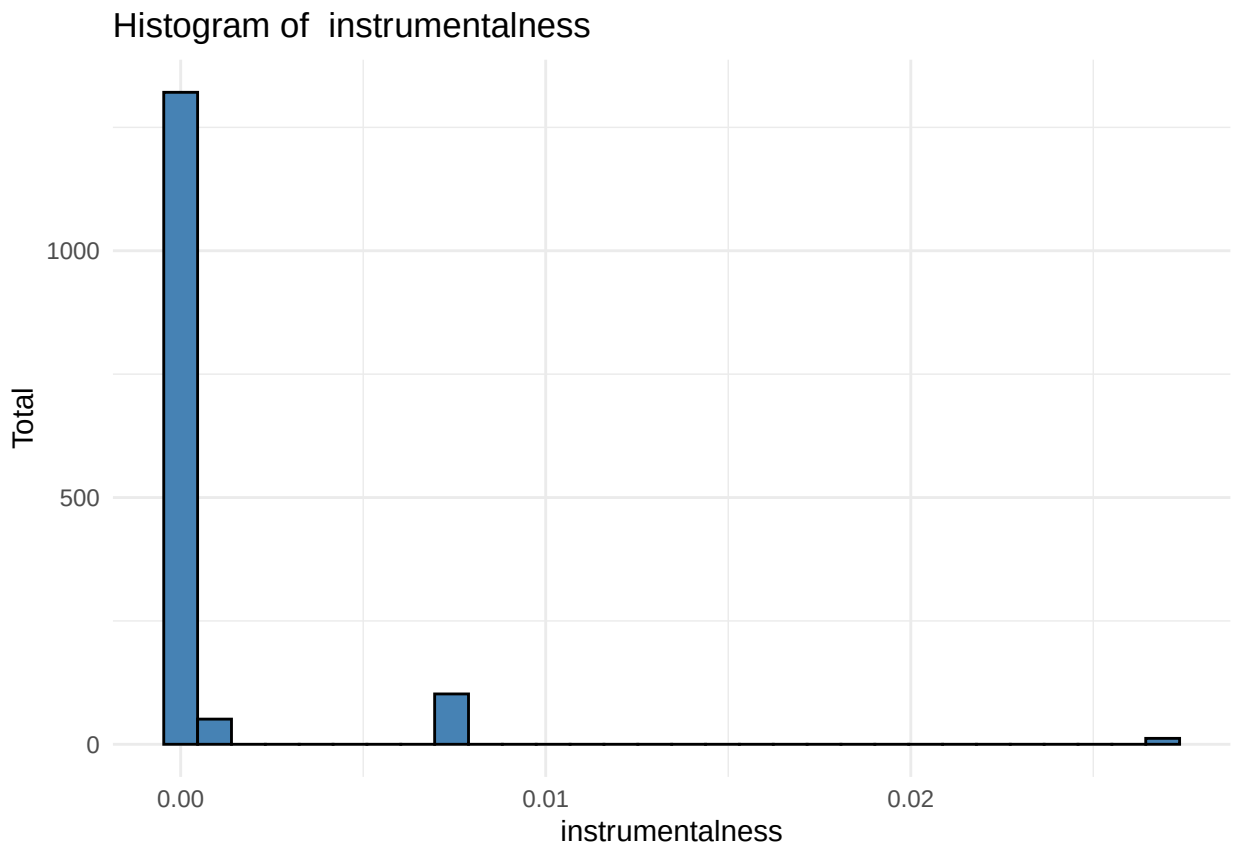

```
## Warning: Removed 194 rows containing non-finite values (stat_bin).

## Saving 6.5 x 4.5 in image

## Warning: Removed 194 rows containing non-finite values (stat_bin).
```



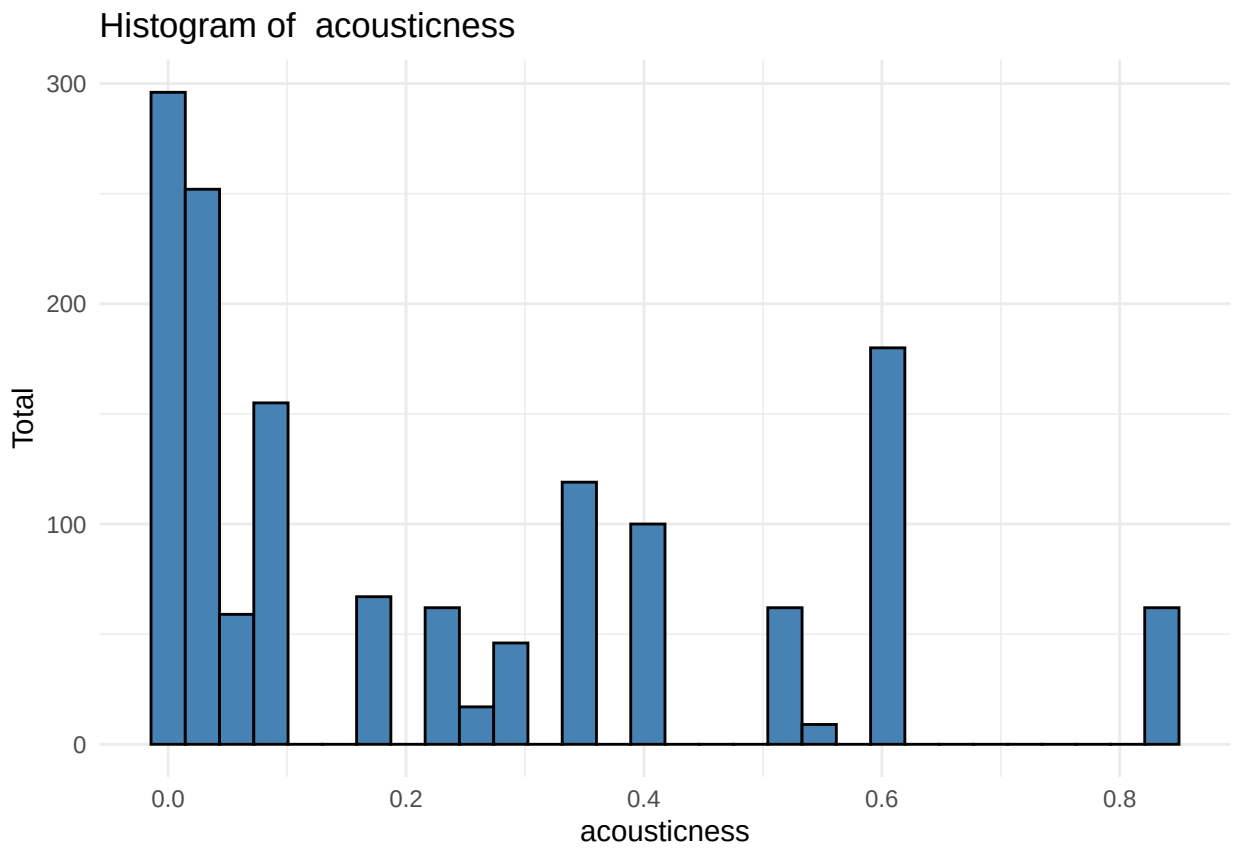

```
## Warning: Removed 194 rows containing non-finite values (stat_bin).

## Saving 6.5 x 4.5 in image

## Warning: Removed 194 rows containing non-finite values (stat_bin).
```



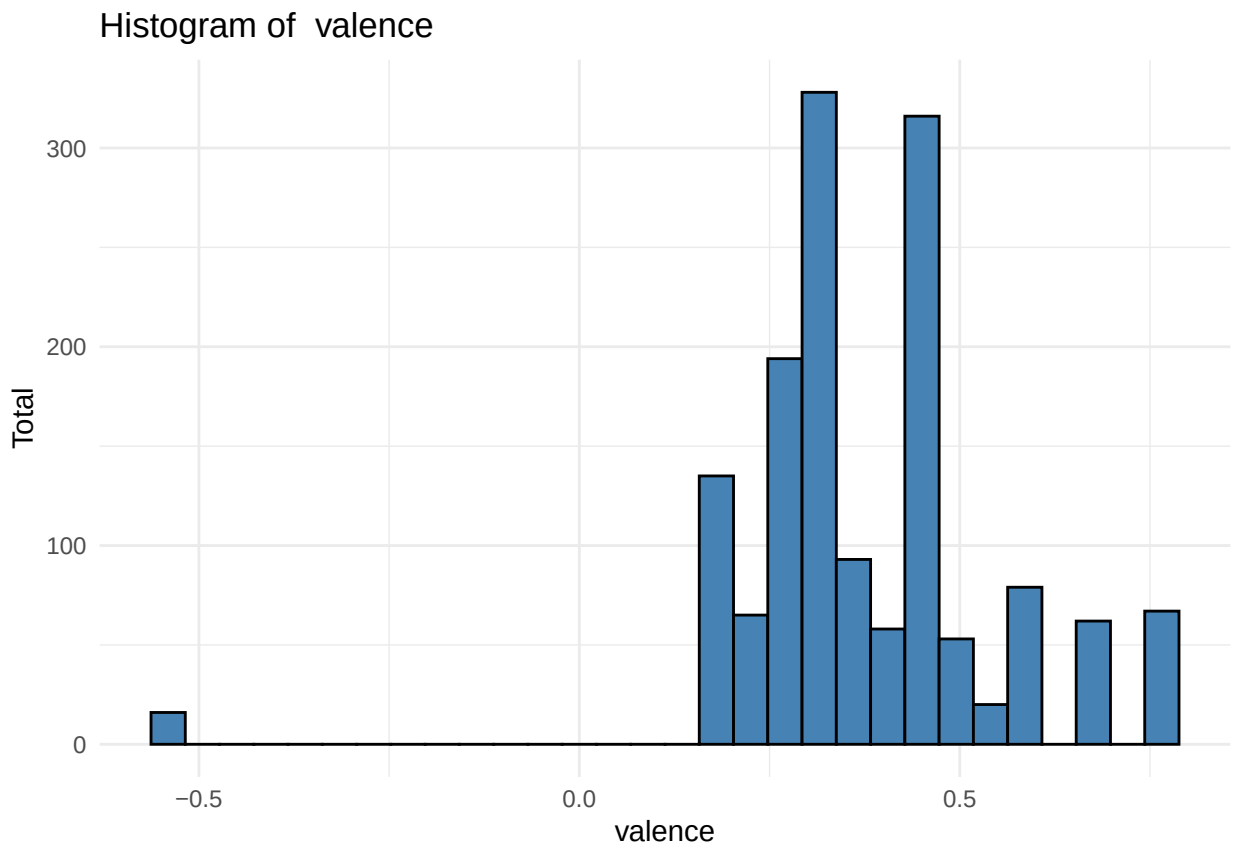

```
## Warning: Removed 194 rows containing non-finite values (stat_bin).

## Saving 6.5 x 4.5 in image

## Warning: Removed 194 rows containing non-finite values (stat_bin).
```



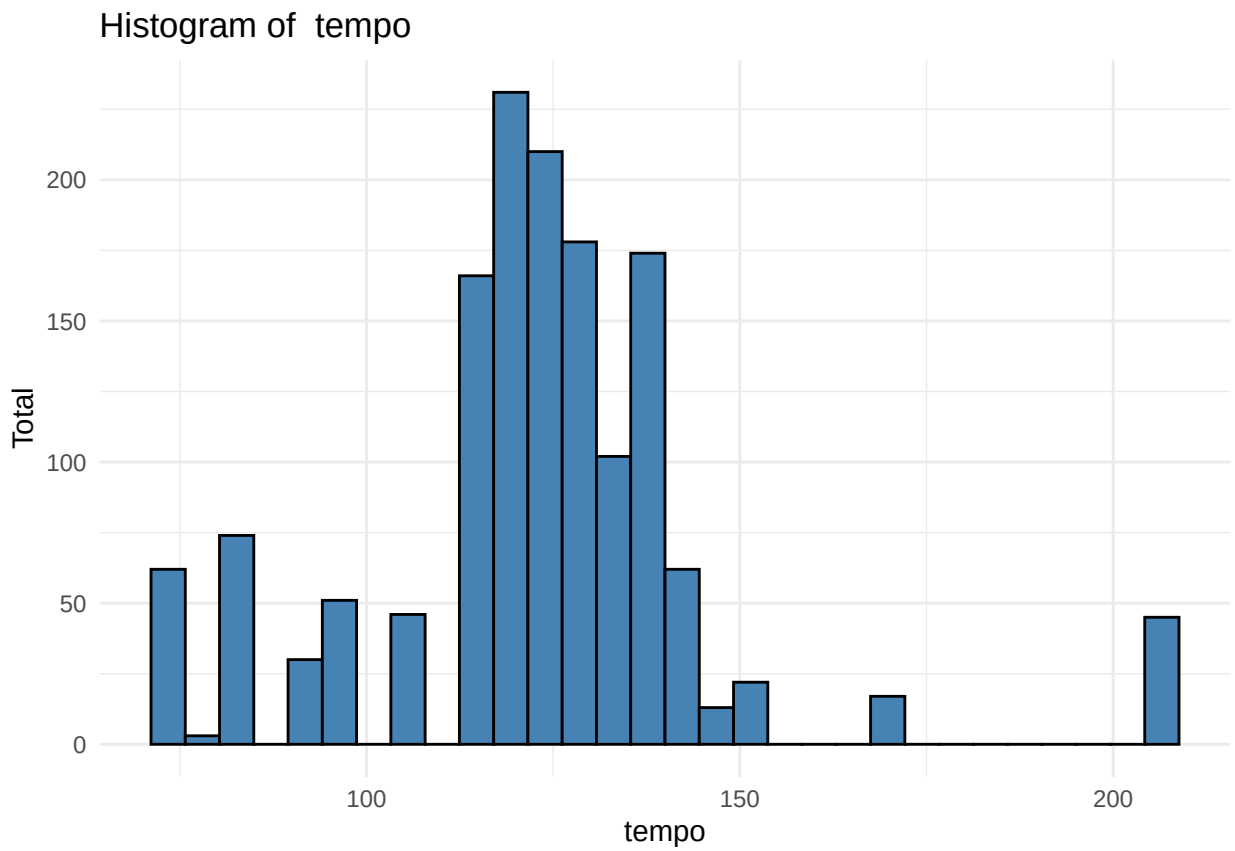

```
## Warning: Removed 194 rows containing non-finite values (stat_bin).

## Saving 6.5 x 4.5 in image

## Warning: Removed 194 rows containing non-finite values (stat_bin).
```



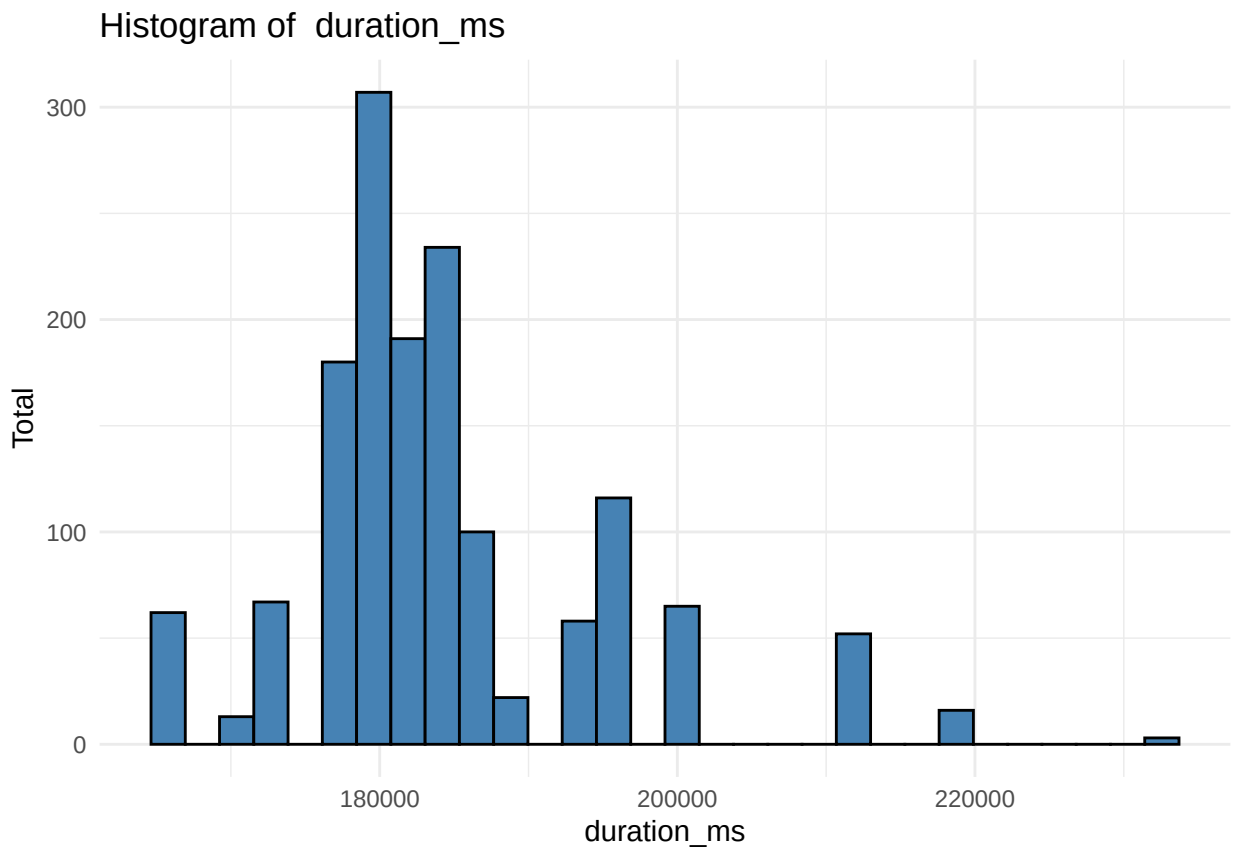

```
## Warning: Removed 194 rows containing non-finite values (stat_bin).

## Saving 6.5 x 4.5 in image

## Warning: Removed 194 rows containing non-finite values (stat_bin).
```



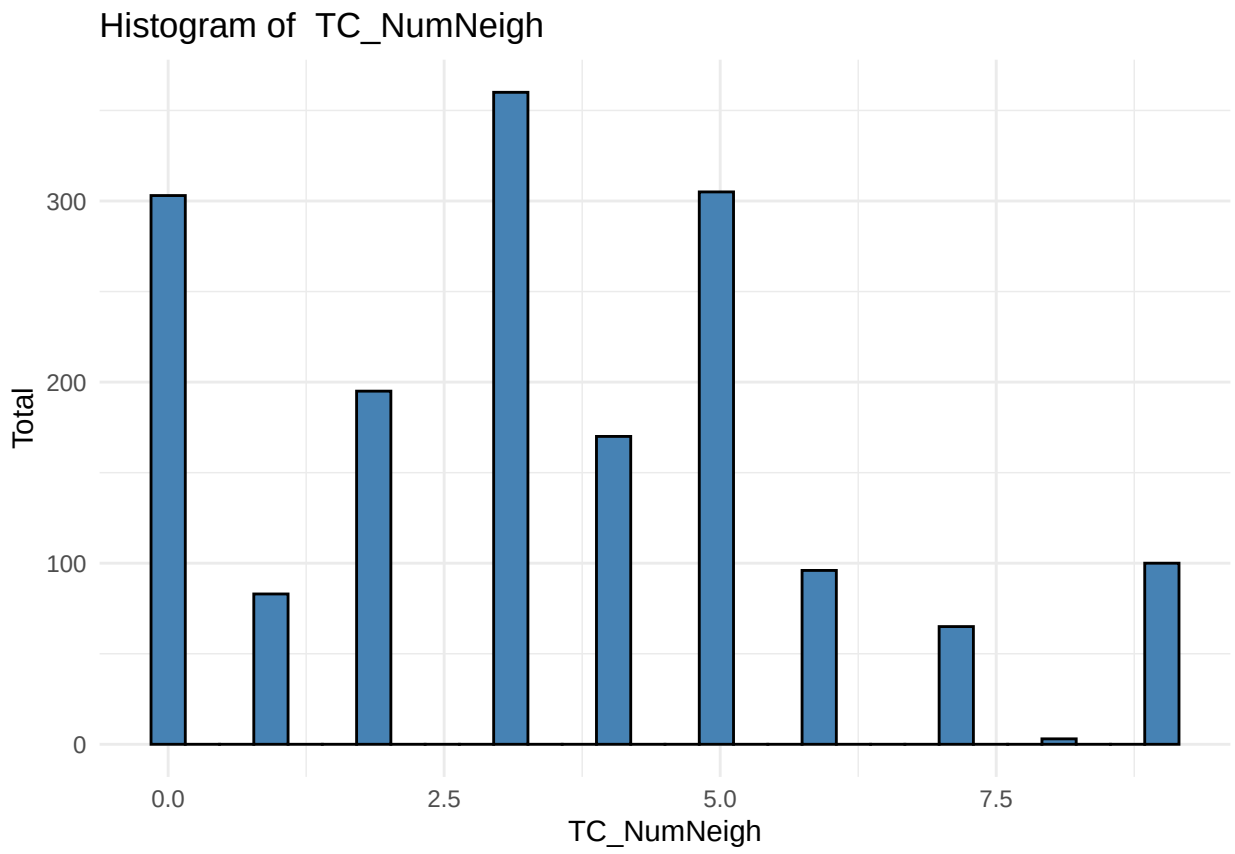

```
## Warning: Removed 194 rows containing non-finite values (stat_bin).

## Saving 6.5 x 4.5 in image

## Warning: Removed 194 rows containing non-finite values (stat_bin).
```



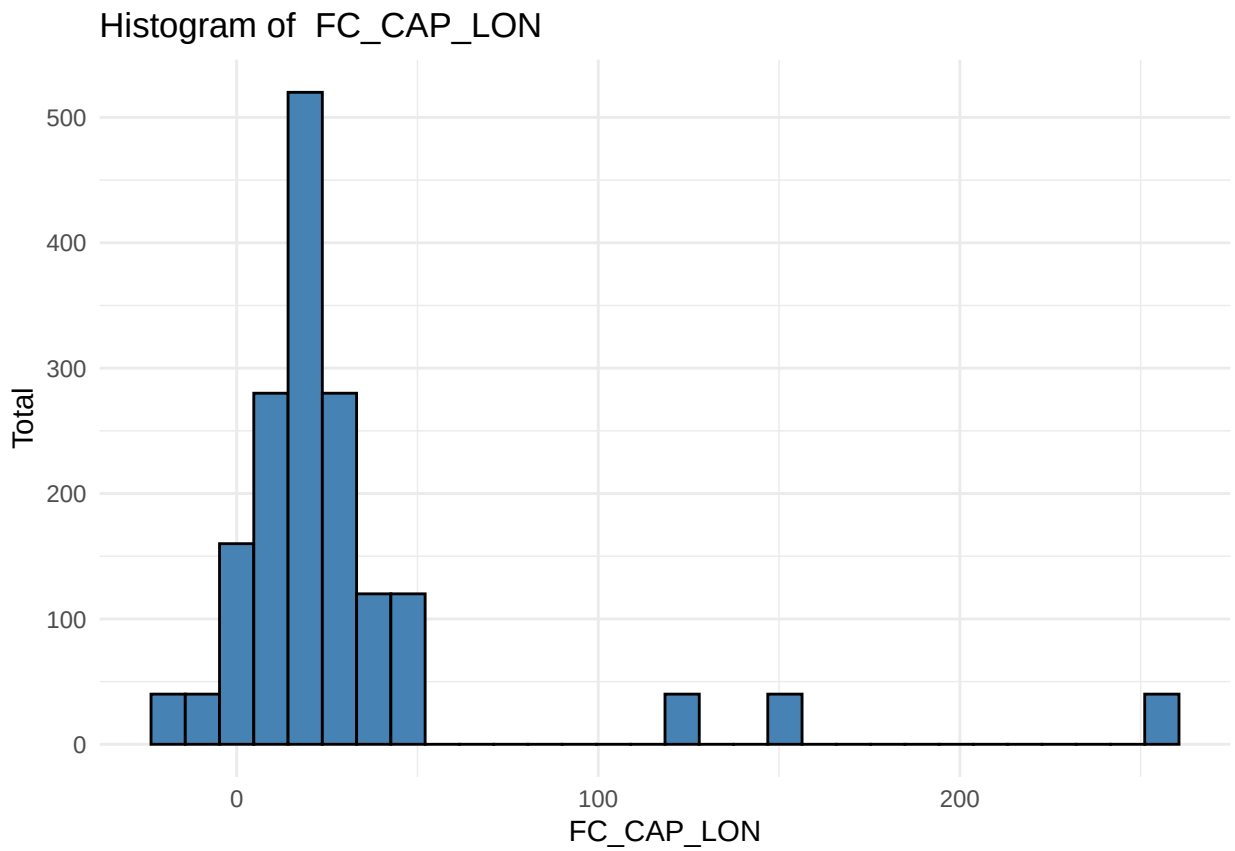

```
## Warning: Removed 194 rows containing non-finite values (stat_bin).

## Saving 6.5 x 4.5 in image

## Warning: Removed 194 rows containing non-finite values (stat_bin).
```



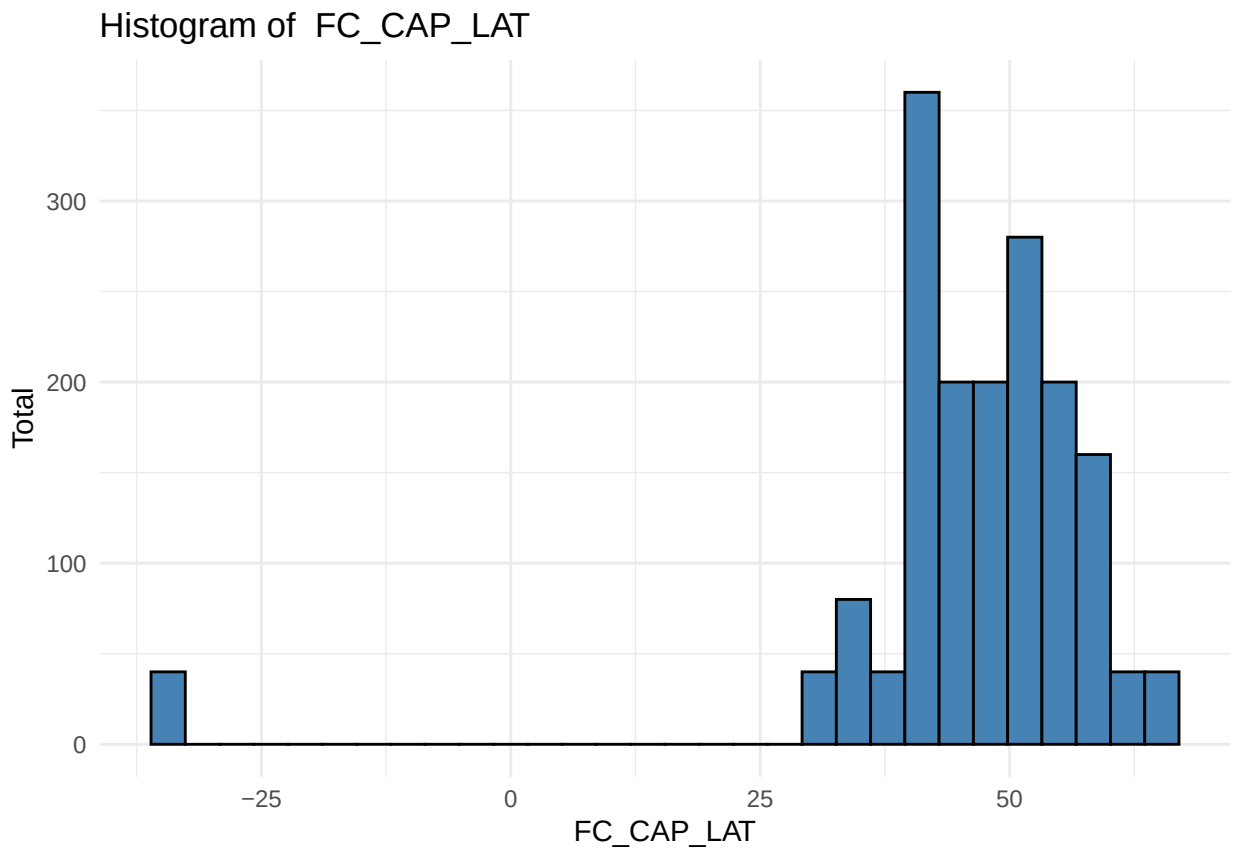

```
## Saving 6.5 x 4.5 in image
```



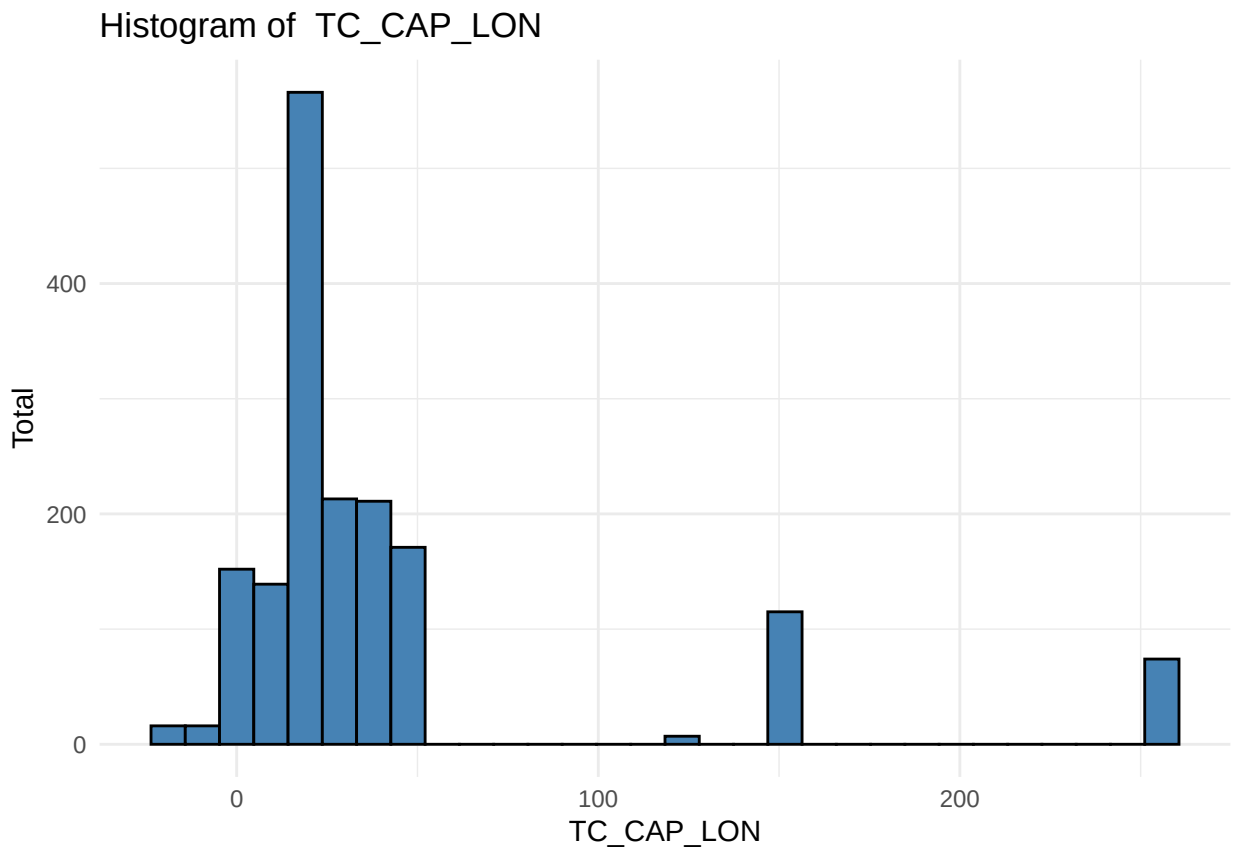

```
## Saving 6.5 x 4.5 in image
```



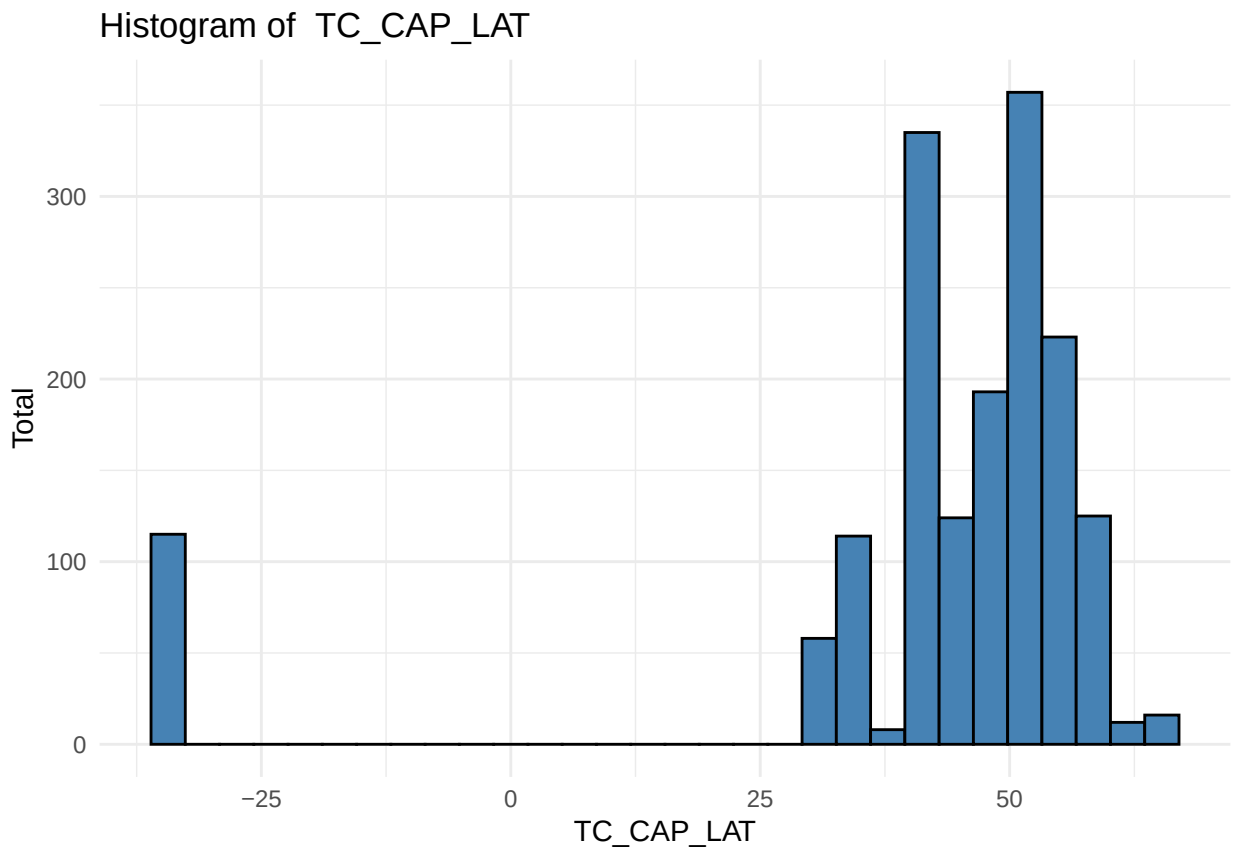

```
## Saving 6.5 x 4.5 in image
```



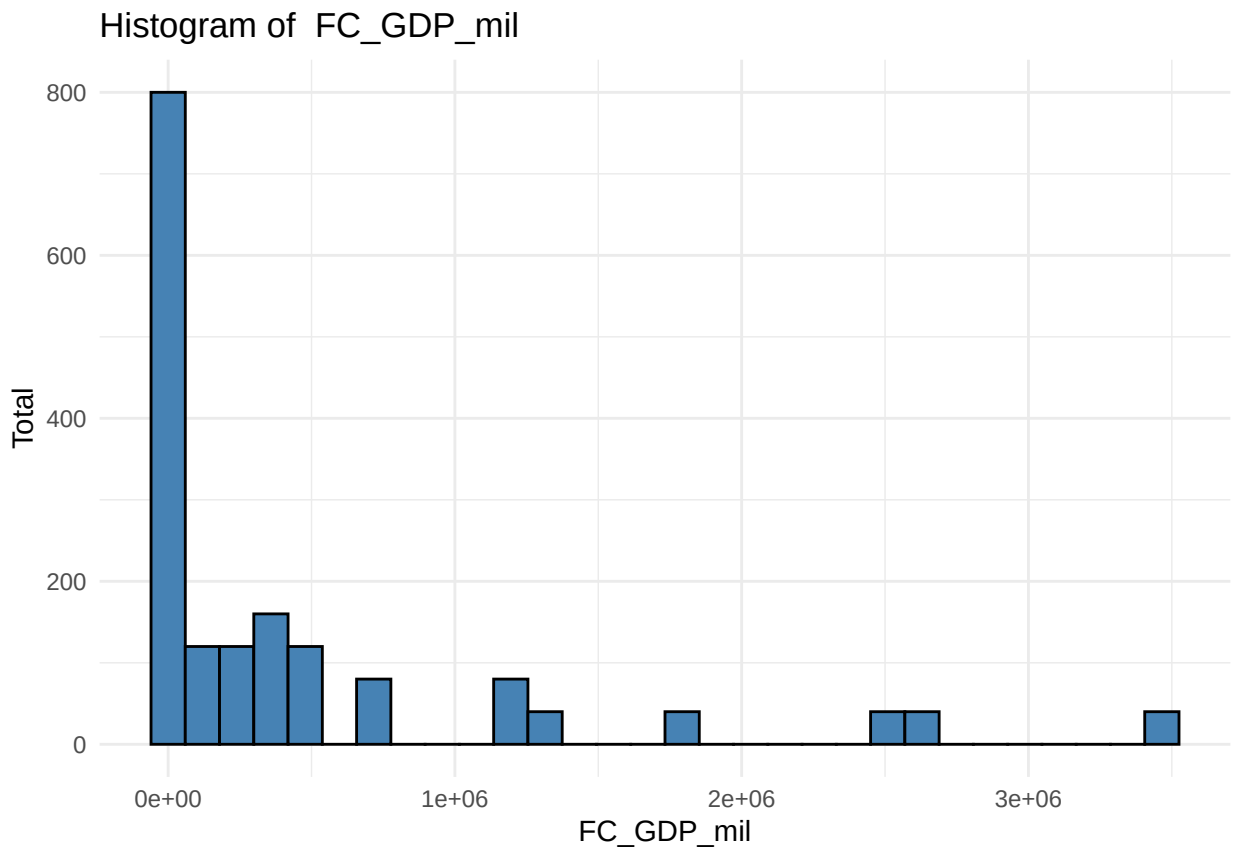

```
## Saving 6.5 x 4.5 in image
```



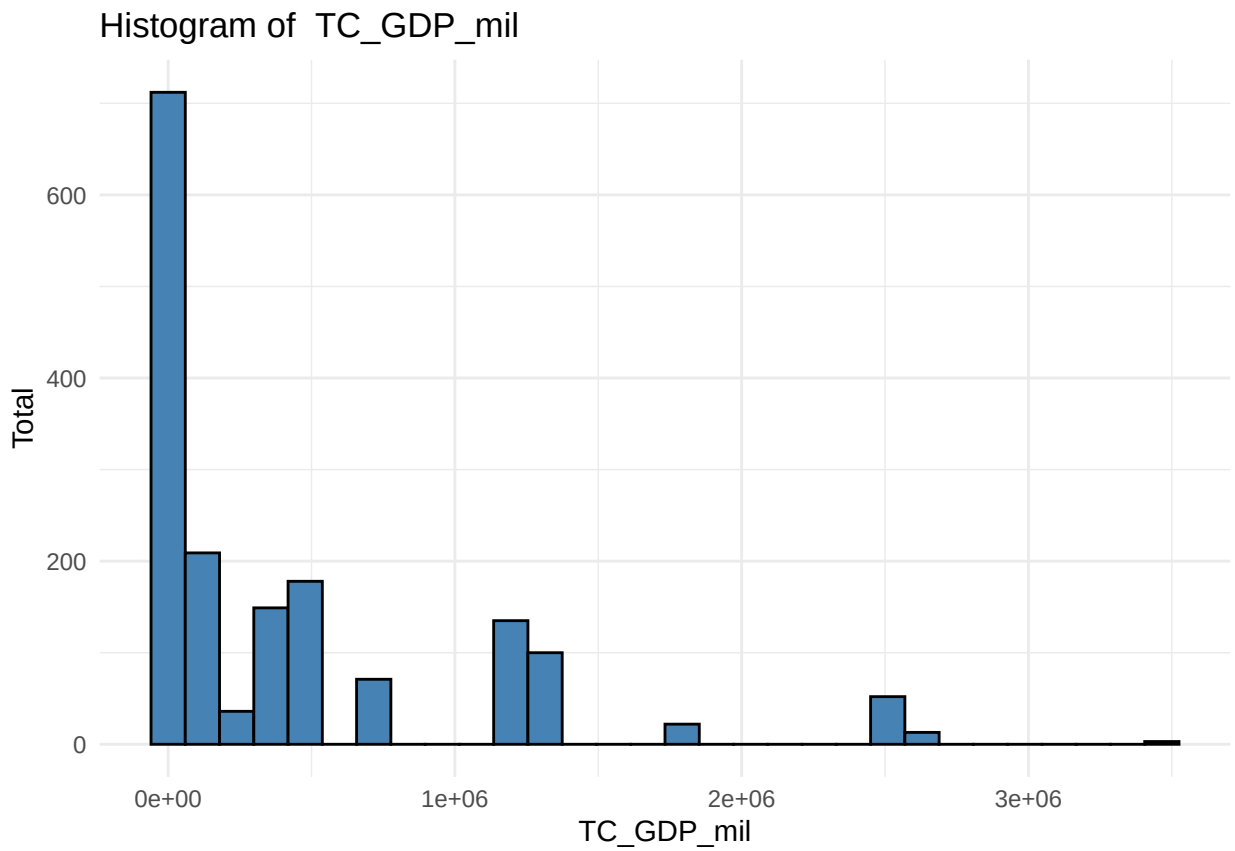

```
## Saving 6.5 x 4.5 in image
```



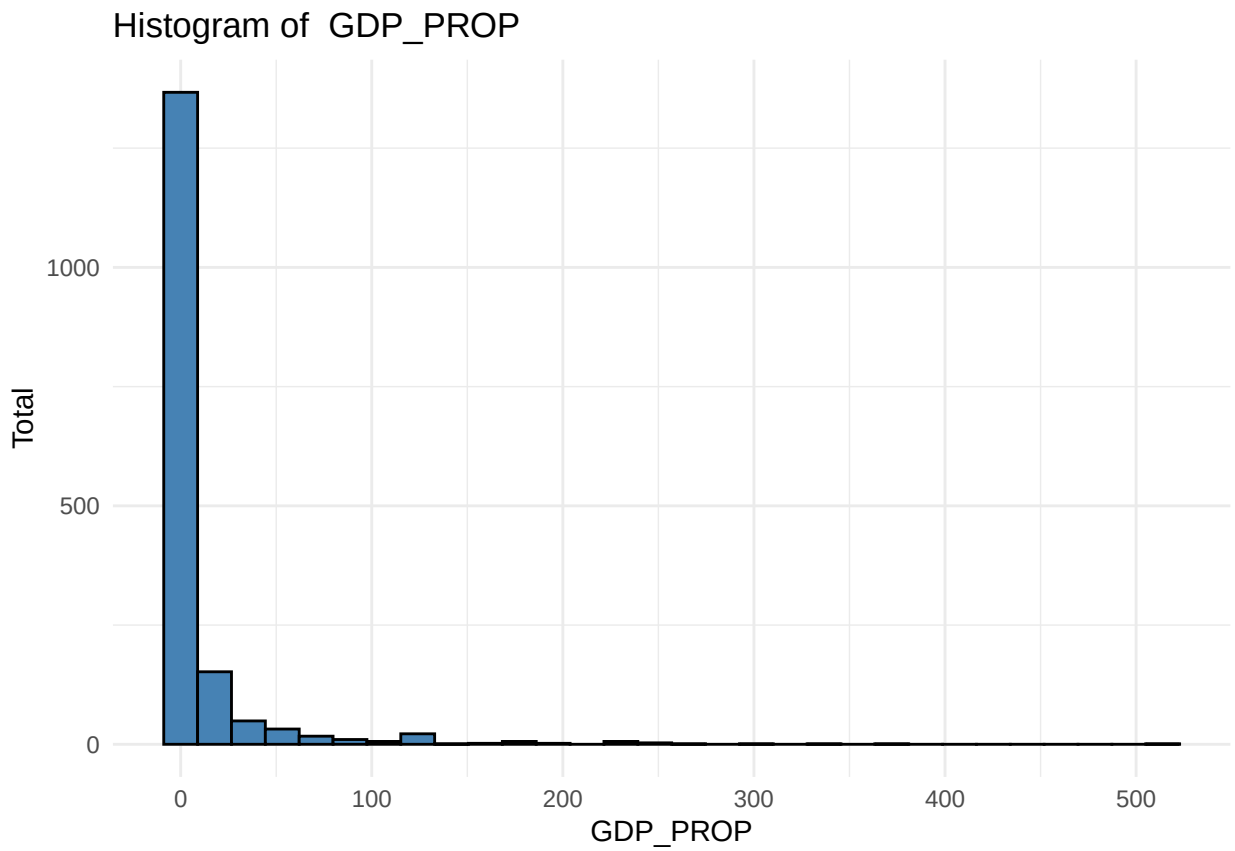

```
## Saving 6.5 x 4.5 in image
```



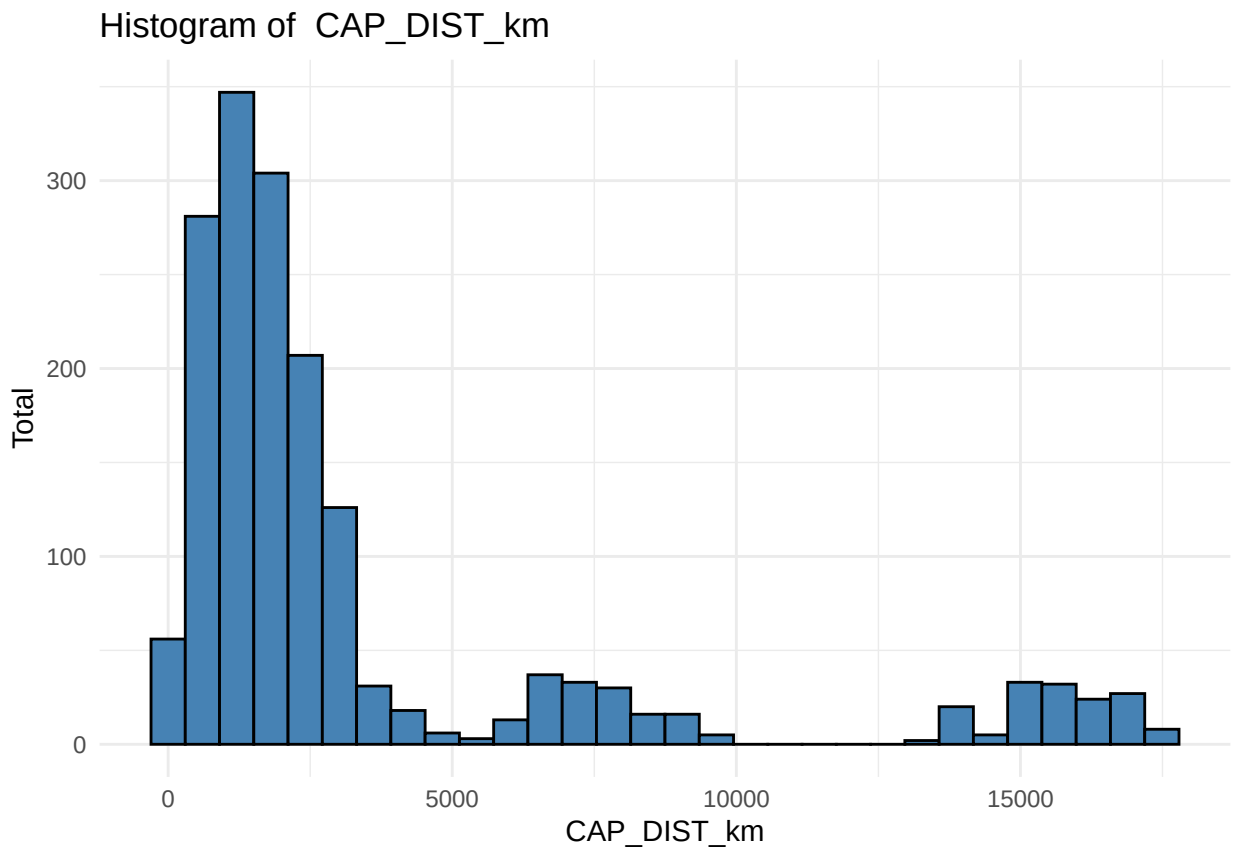

```
## Saving 6.5 x 4.5 in image
```



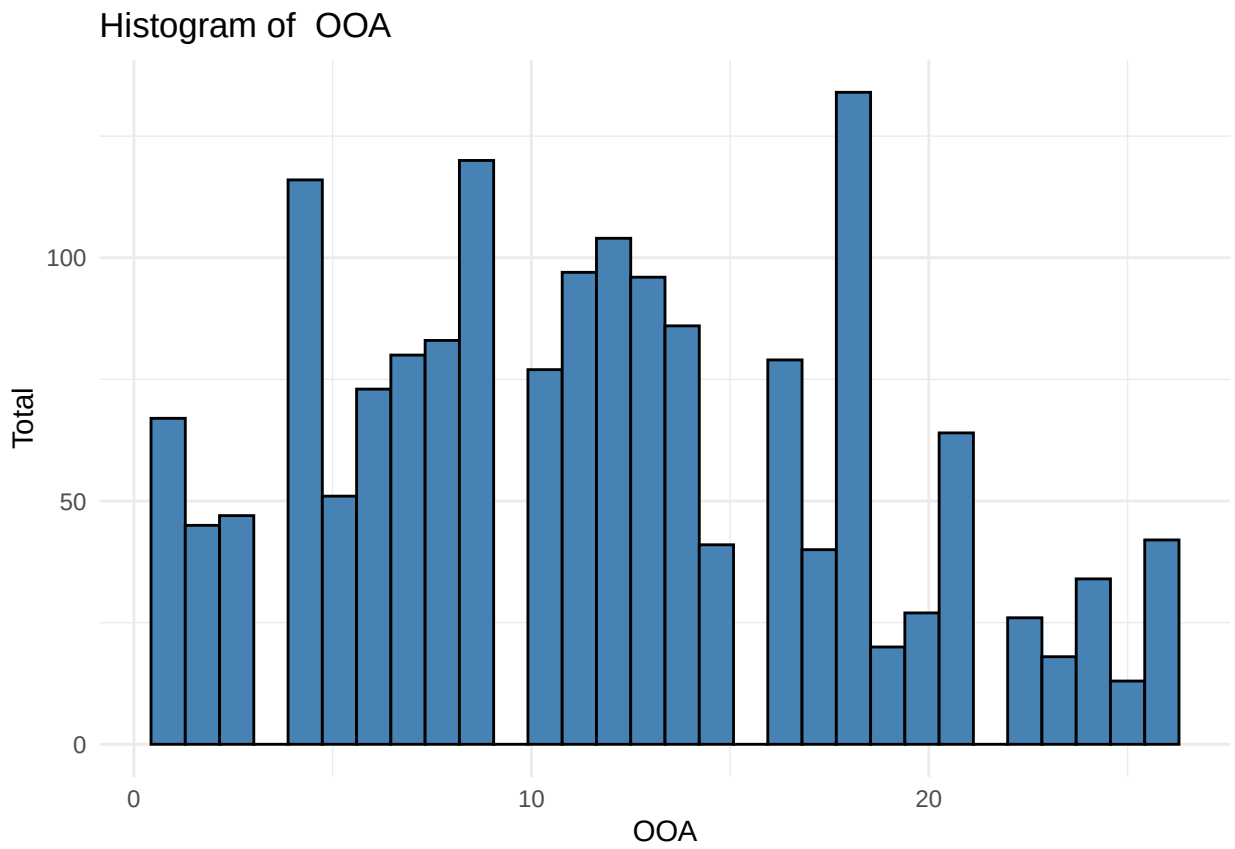

```
## Saving 6.5 x 4.5 in image
```




```
## Saving 6.5 x 4.5 in image
```




```
## Saving 6.5 x 4.5 in image
```




```
## [1] 0
```

Chi-squared Tests of Association

This section conducts chi-squared tests of association for each categorical predictor variable and the response variable

Hypothesis: Ho: x is independent of y Ha: x is associated with y

```
# perform chi-sq tests of association with Points
chisqtestdf <- chisq_assoc_test(data = ESCdata, col_names = all_factors)
# write the data frame to a csv file
write.csv(x = chisqtestdf, file = "./report/stats/chi_sq_tests_response.csv", row.names = F)
```

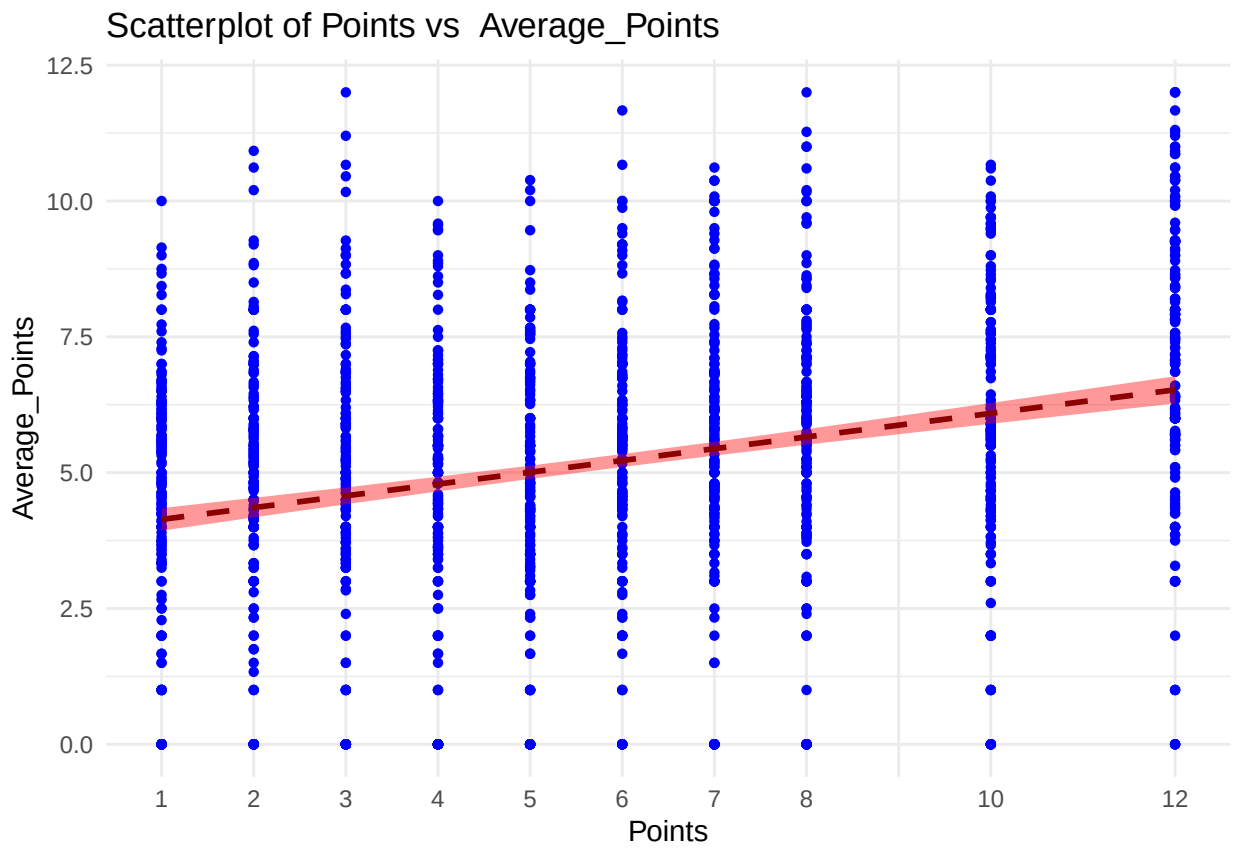
Correlation Scatter Plots and Correlation Tests

This section generates some scatter plots and performs correlation tests.

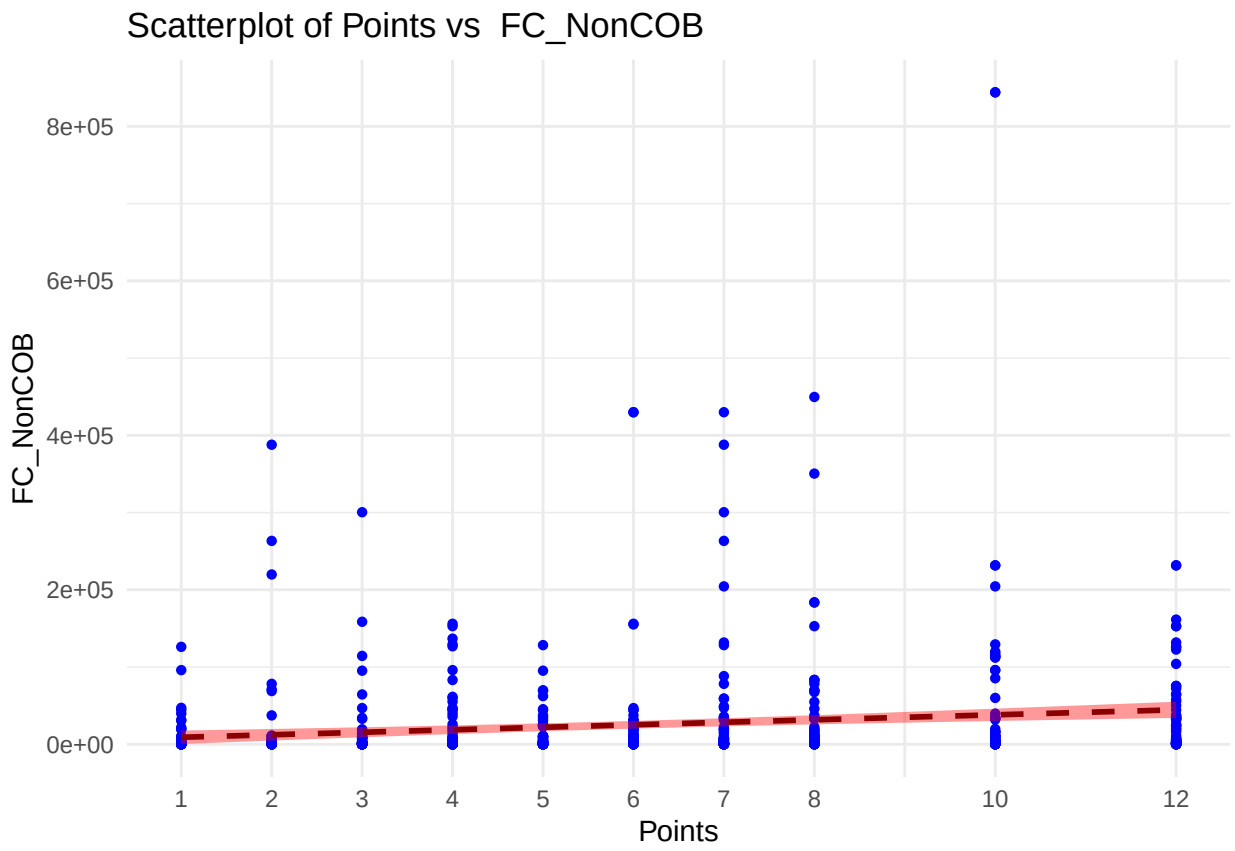
Correlation Test Hypothesis Ho: x is not correlated with y Ha: x is correlated with y

```
# plot scatter plot
plot_scatter(data = ESCdata, col_names = all_num, output_dir = './report/plots/scatterplots')
```

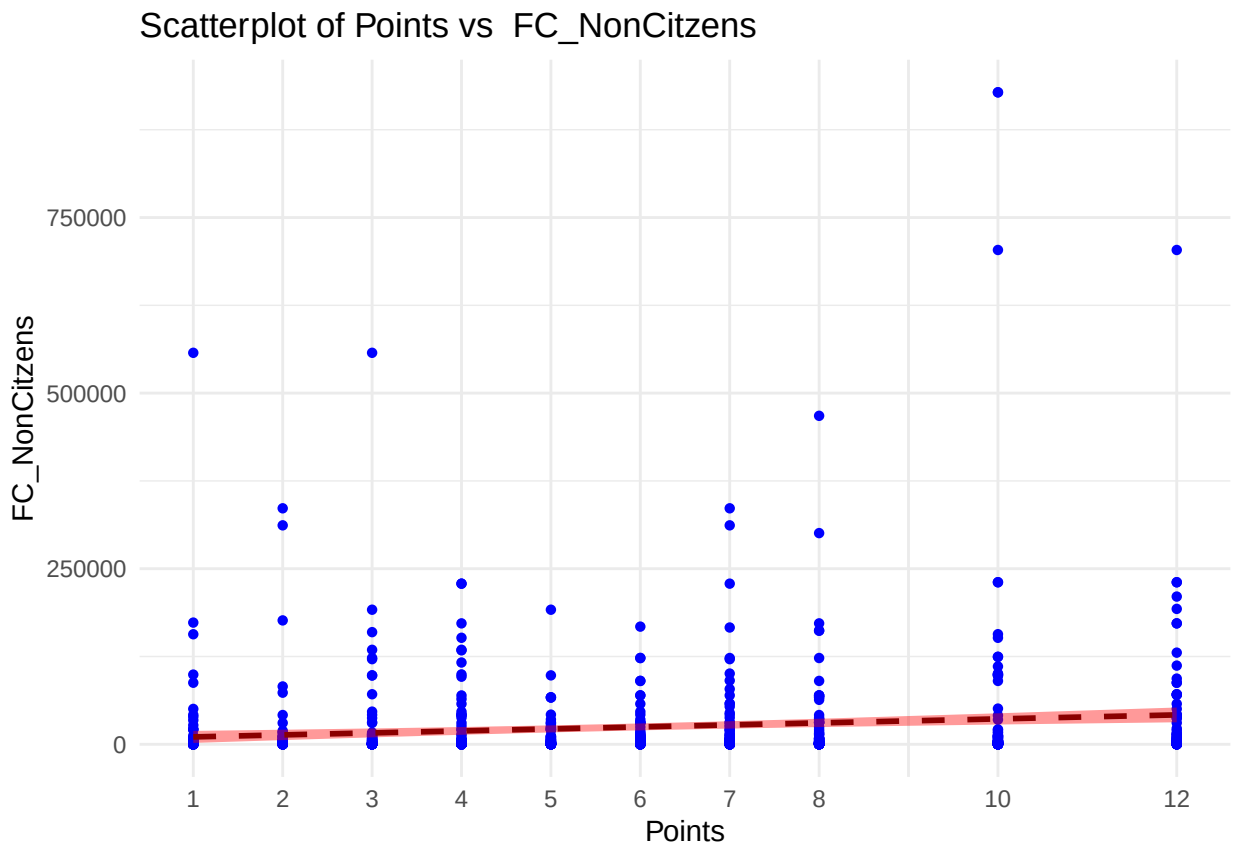
```
## Saving 6.5 x 4.5 in image
```



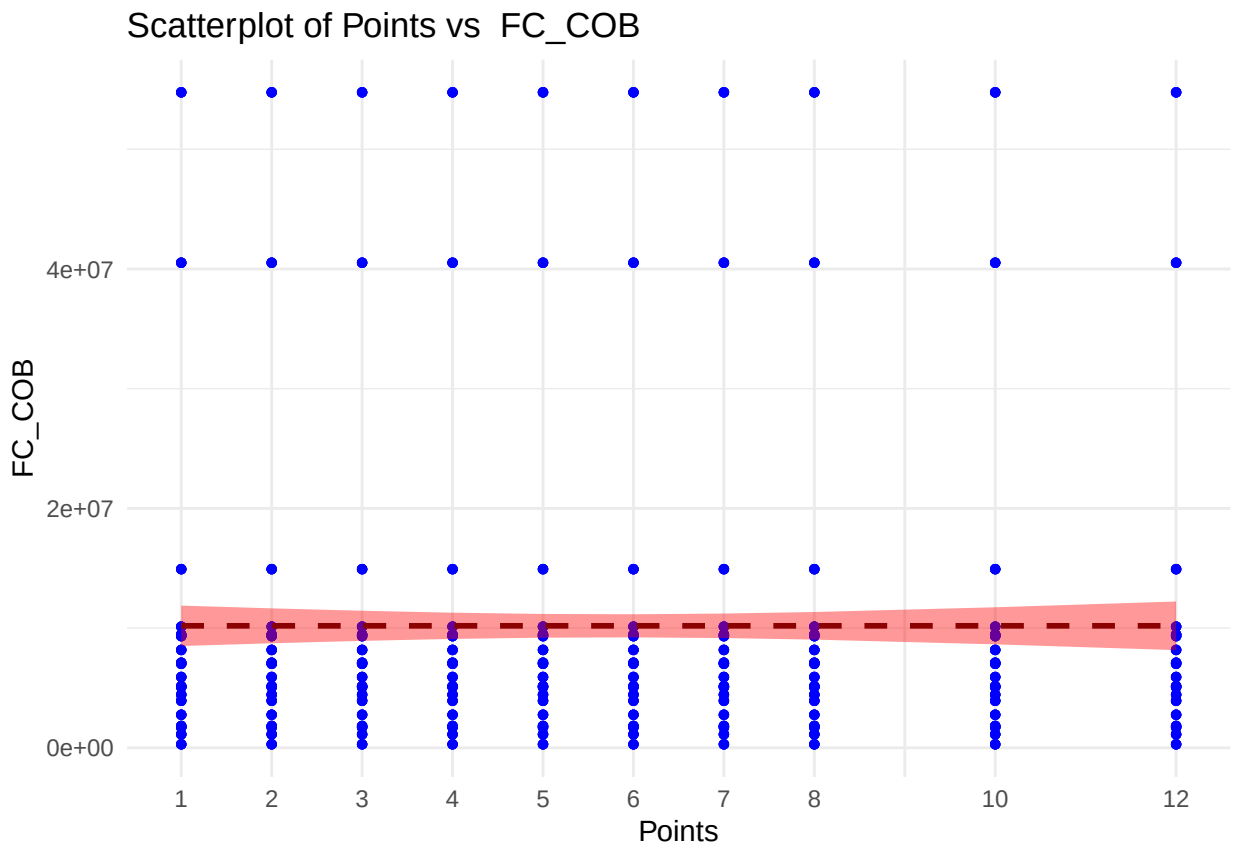

```
## Warning: Removed 924 rows containing non-finite values (stat_smooth).

## Warning: Removed 924 rows containing missing values (geom_point).

## Saving 6.5 x 4.5 in image

## Warning: Removed 924 rows containing non-finite values (stat_smooth).
## Removed 924 rows containing missing values (geom_point).
```



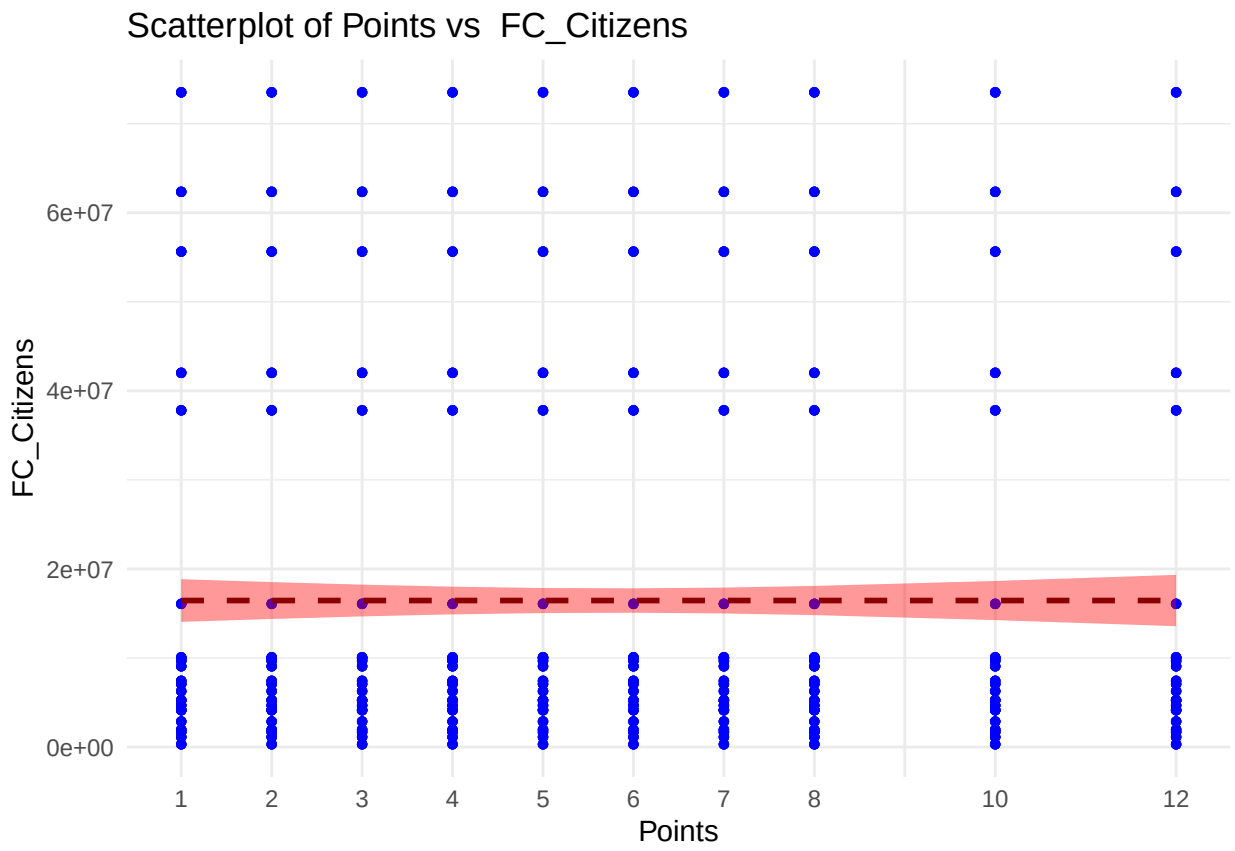

```
## Warning: Removed 775 rows containing non-finite values (stat_smooth).

## Warning: Removed 775 rows containing missing values (geom_point).

## Saving 6.5 x 4.5 in image

## Warning: Removed 775 rows containing non-finite values (stat_smooth).
## Removed 775 rows containing missing values (geom_point).
```



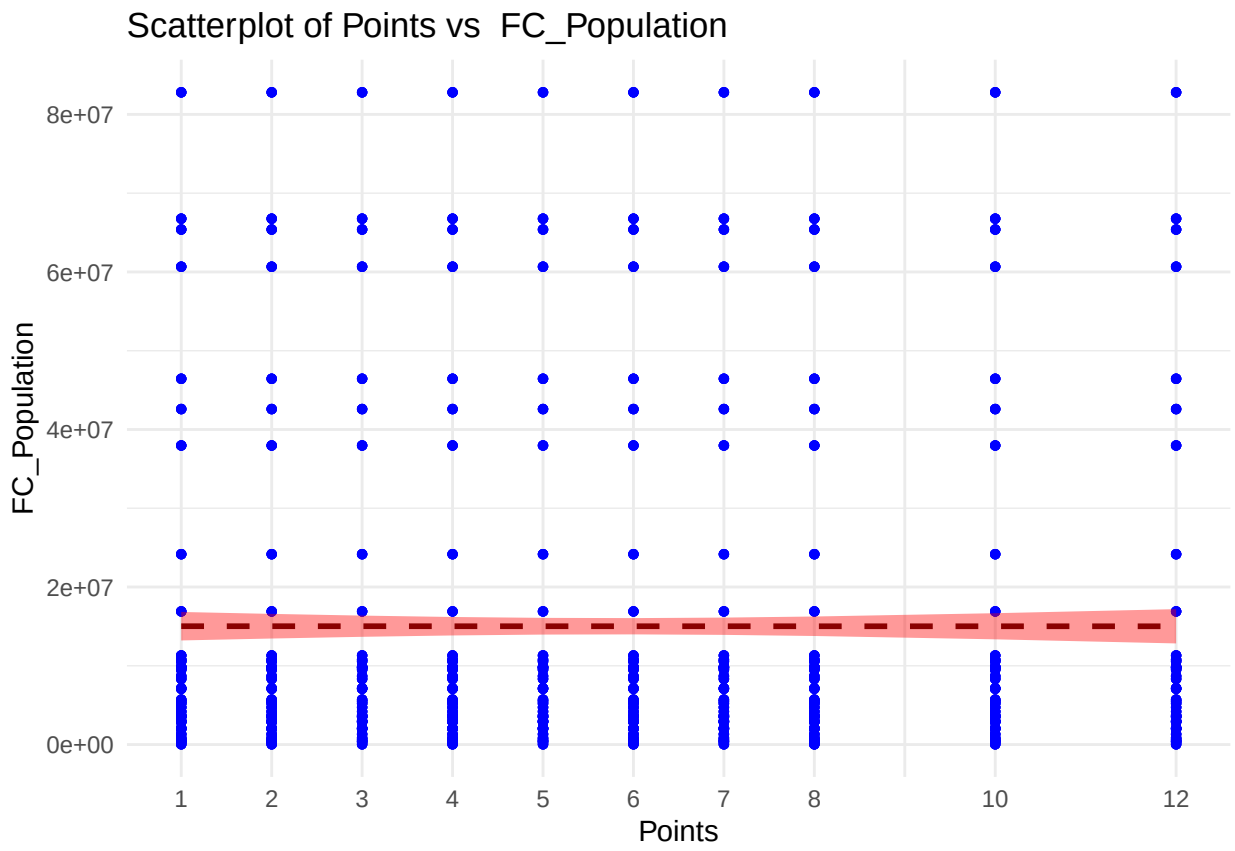

```
## Warning: Removed 920 rows containing non-finite values (stat_smooth).

## Warning: Removed 920 rows containing missing values (geom_point).

## Saving 6.5 x 4.5 in image

## Warning: Removed 920 rows containing non-finite values (stat_smooth).
## Removed 920 rows containing missing values (geom_point).
```



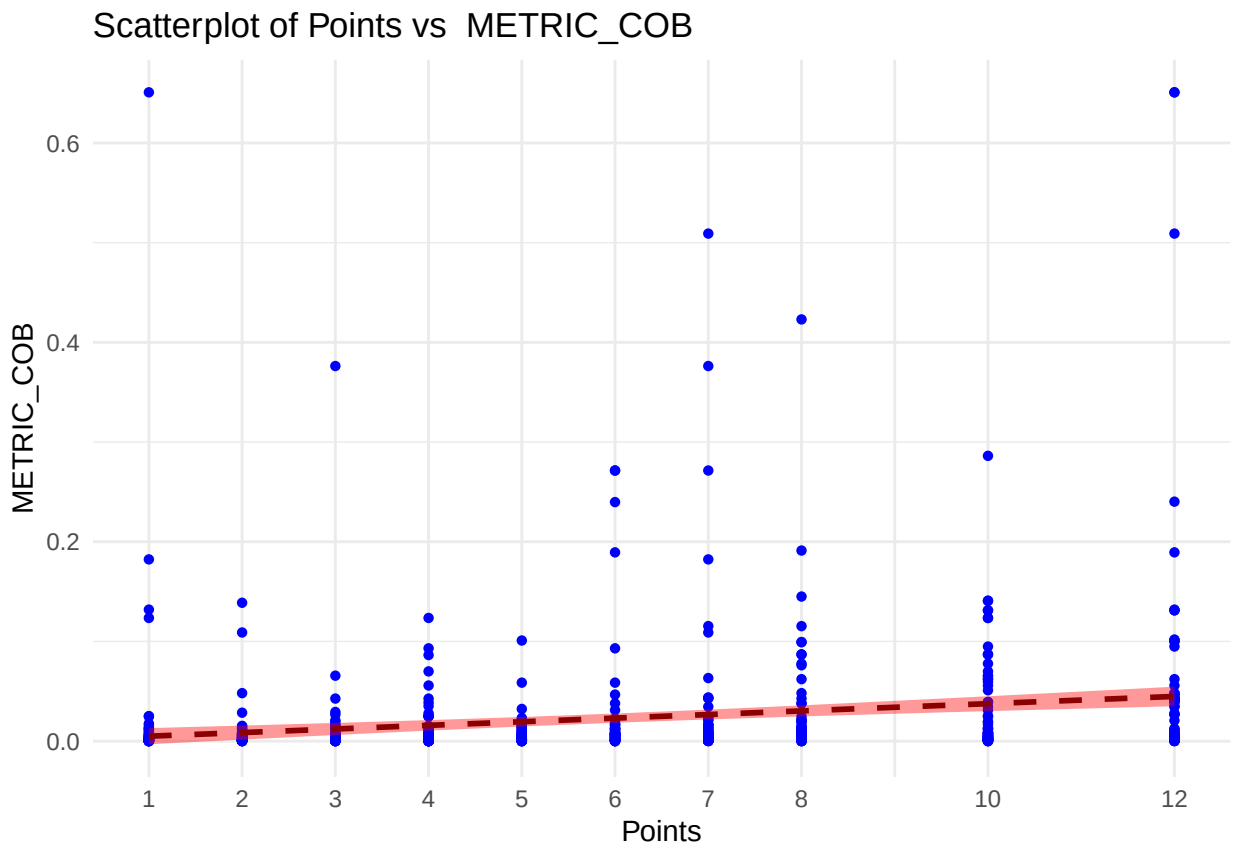

```
## Warning: Removed 760 rows containing non-finite values (stat_smooth).

## Warning: Removed 760 rows containing missing values (geom_point).

## Saving 6.5 x 4.5 in image

## Warning: Removed 760 rows containing non-finite values (stat_smooth).
## Removed 760 rows containing missing values (geom_point).
```



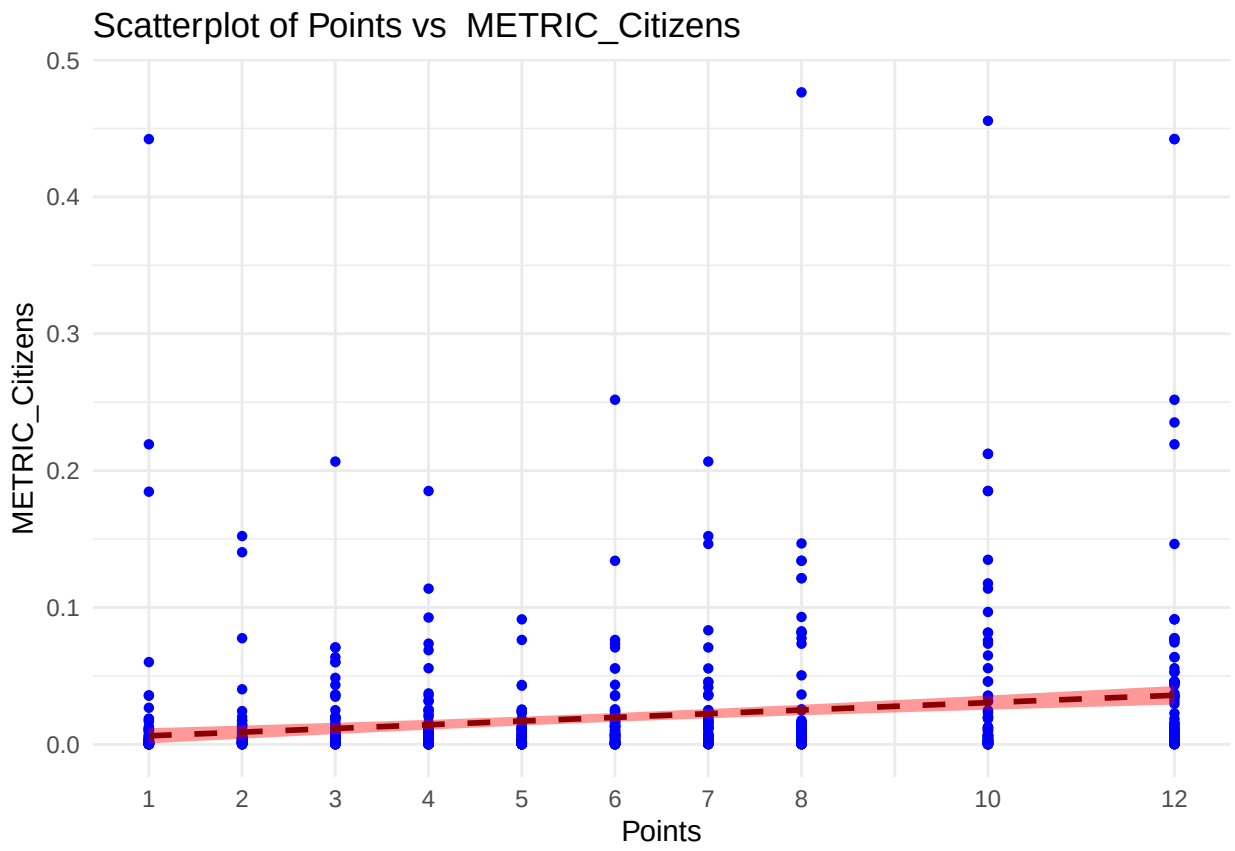

```
## Warning: Removed 80 rows containing non-finite values (stat_smooth).

## Warning: Removed 80 rows containing missing values (geom_point).

## Saving 6.5 x 4.5 in image

## Warning: Removed 80 rows containing non-finite values (stat_smooth).
## Removed 80 rows containing missing values (geom_point).
```



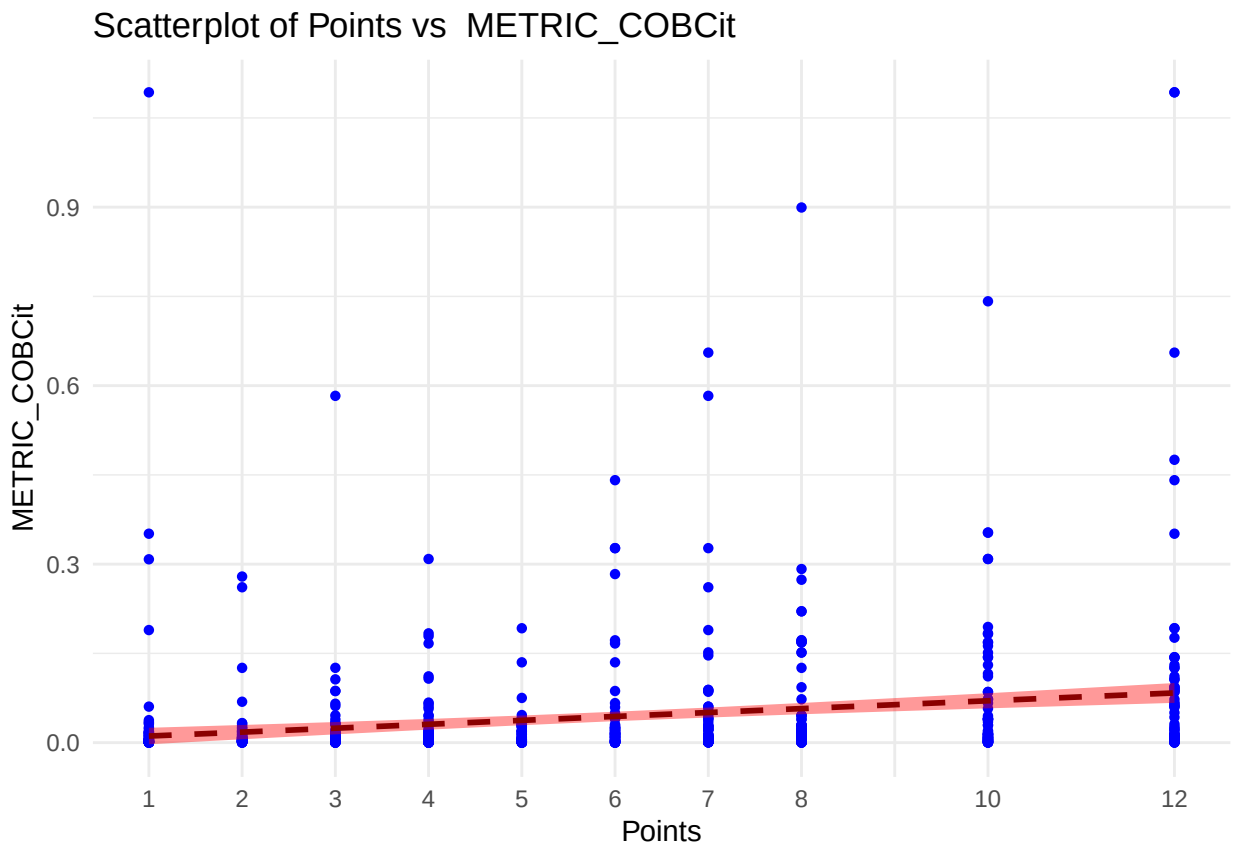

```
## Warning: Removed 946 rows containing non-finite values (stat_smooth).

## Warning: Removed 946 rows containing missing values (geom_point).

## Saving 6.5 x 4.5 in image

## Warning: Removed 946 rows containing non-finite values (stat_smooth).
## Removed 946 rows containing missing values (geom_point).
```



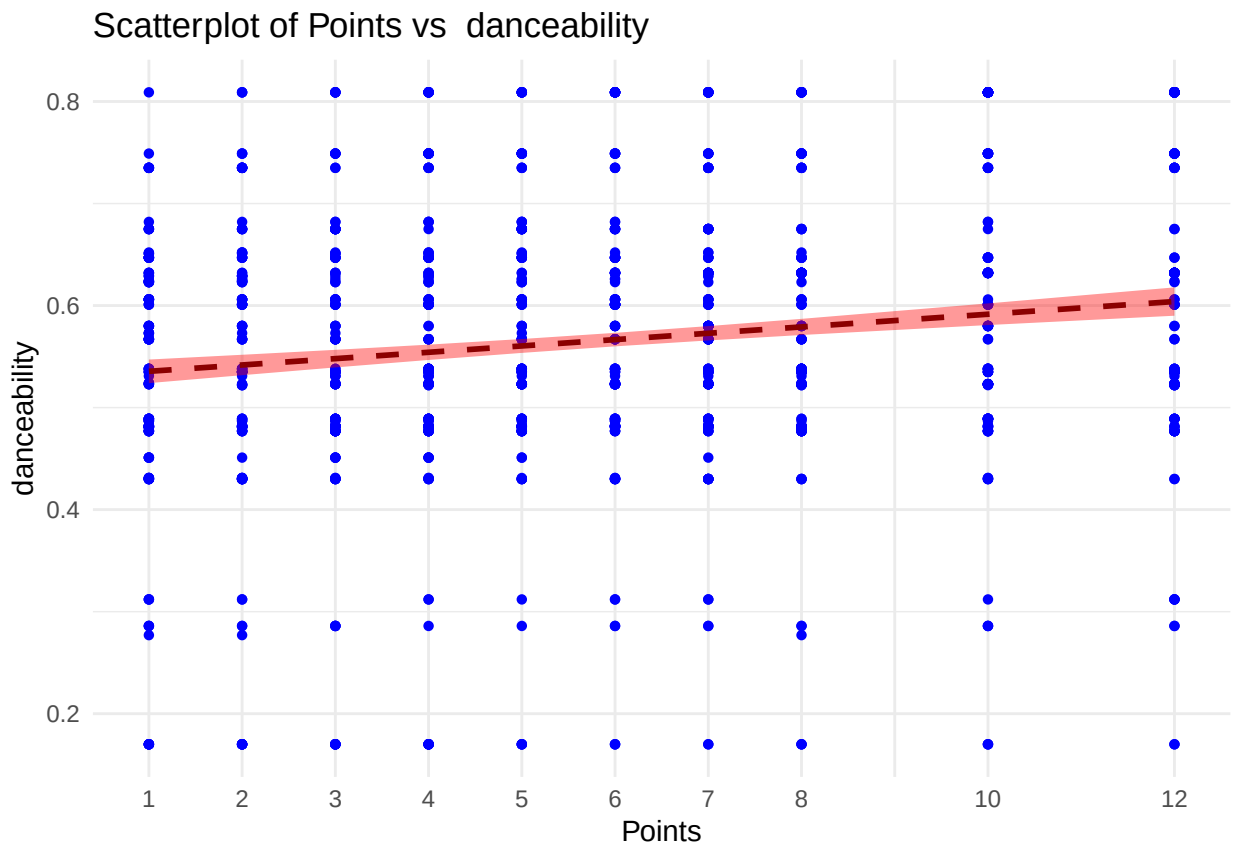

```
## Warning: Removed 831 rows containing non-finite values (stat_smooth).

## Warning: Removed 831 rows containing missing values (geom_point).

## Saving 6.5 x 4.5 in image

## Warning: Removed 831 rows containing non-finite values (stat_smooth).
## Removed 831 rows containing missing values (geom_point).
```



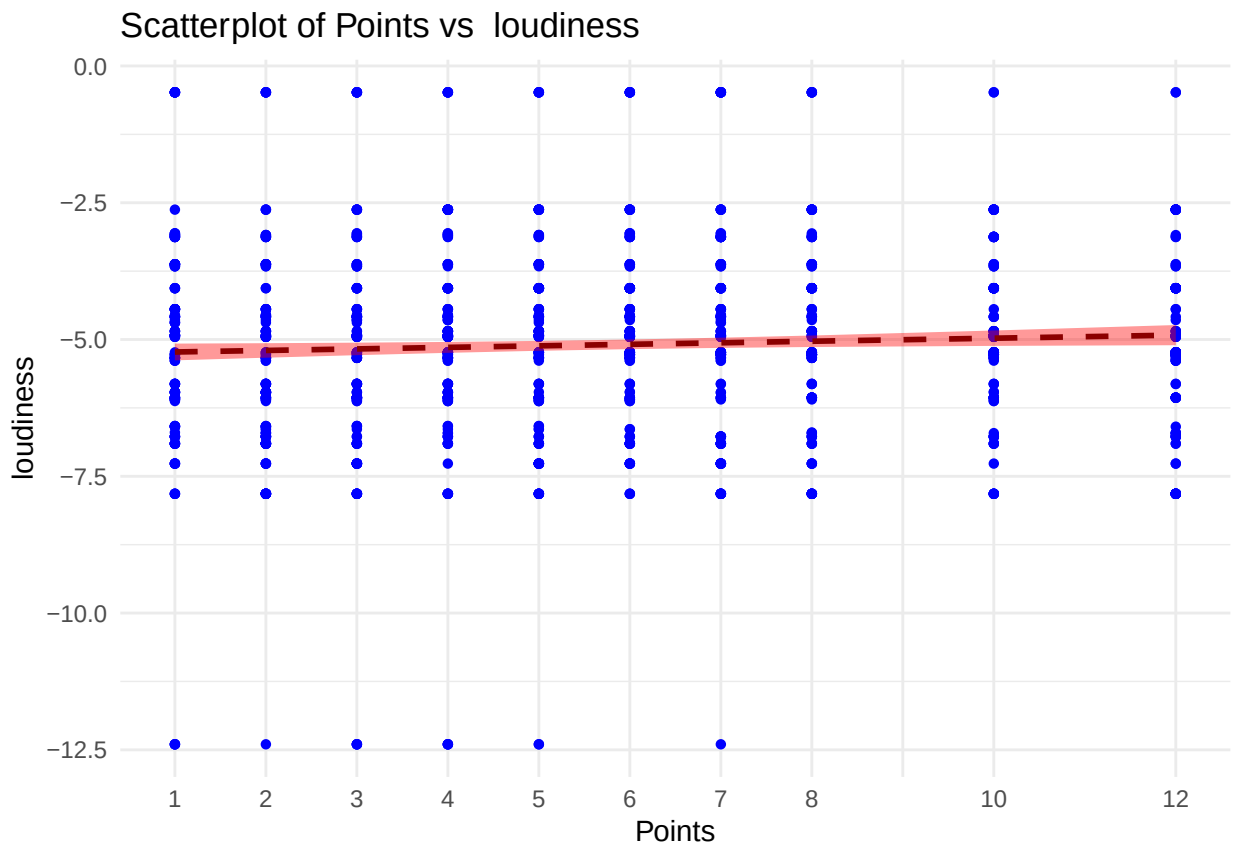

```
## Warning: Removed 947 rows containing non-finite values (stat_smooth).

## Warning: Removed 947 rows containing missing values (geom_point).

## Saving 6.5 x 4.5 in image

## Warning: Removed 947 rows containing non-finite values (stat_smooth).
## Removed 947 rows containing missing values (geom_point).
```




```
## Warning: Removed 194 rows containing non-finite values (stat_smooth).

## Warning: Removed 194 rows containing missing values (geom_point).

## Saving 6.5 x 4.5 in image

## Warning: Removed 194 rows containing non-finite values (stat_smooth).
## Removed 194 rows containing missing values (geom_point).
```



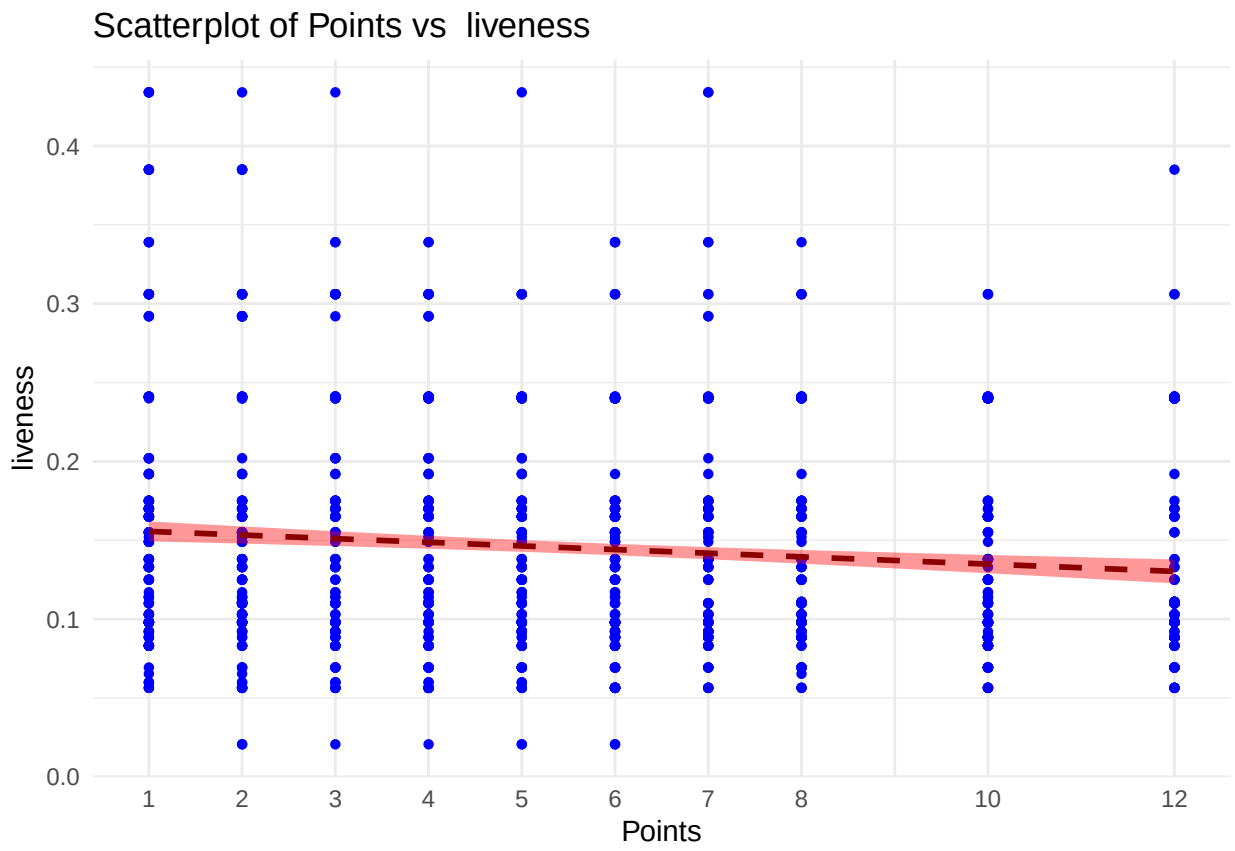

```
## Warning: Removed 194 rows containing non-finite values (stat_smooth).
## Removed 194 rows containing missing values (geom_point).

## Saving 6.5 x 4.5 in image

## Warning: Removed 194 rows containing non-finite values (stat_smooth).
## Removed 194 rows containing missing values (geom_point).
```




```
## Warning: Removed 194 rows containing non-finite values (stat_smooth).
## Removed 194 rows containing missing values (geom_point).
```

```
## Saving 6.5 x 4.5 in image
```

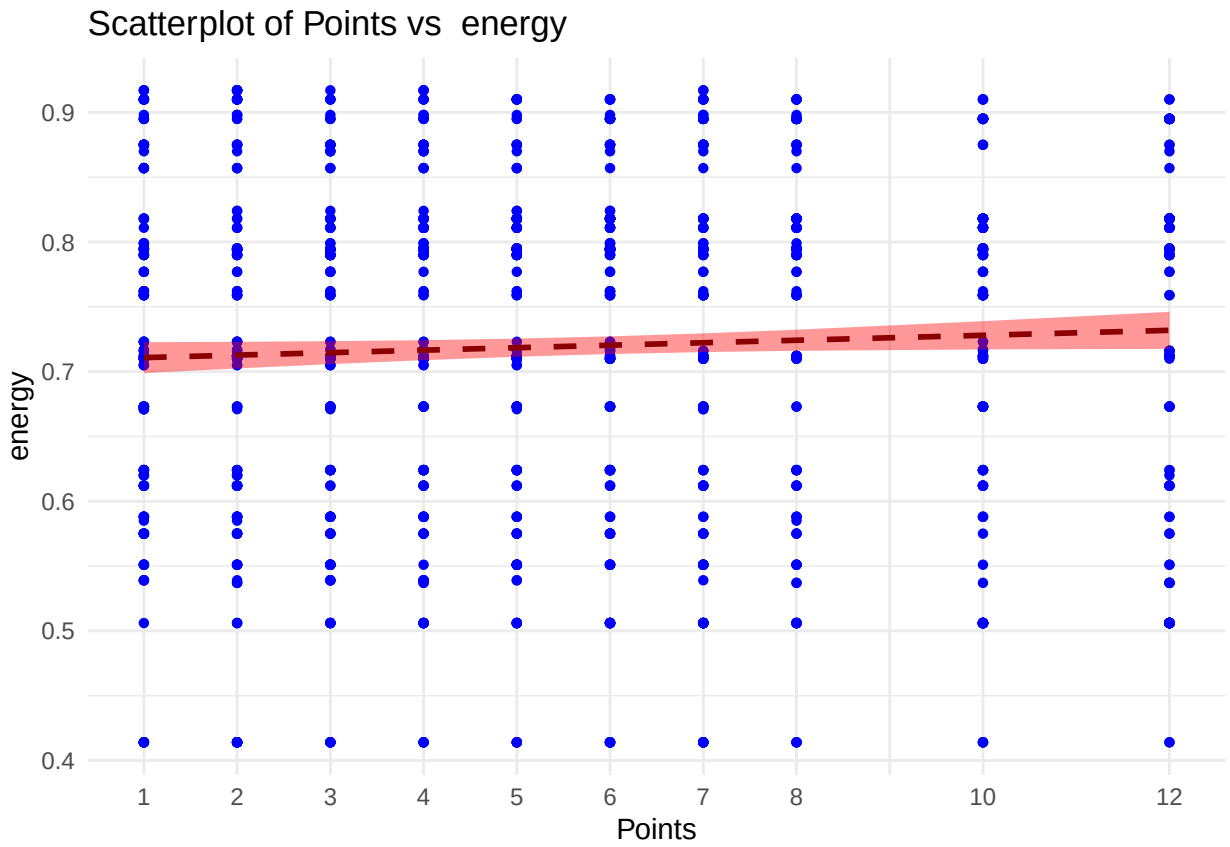
```
## Warning: Removed 194 rows containing non-finite values (stat_smooth).
## Removed 194 rows containing missing values (geom_point).
```



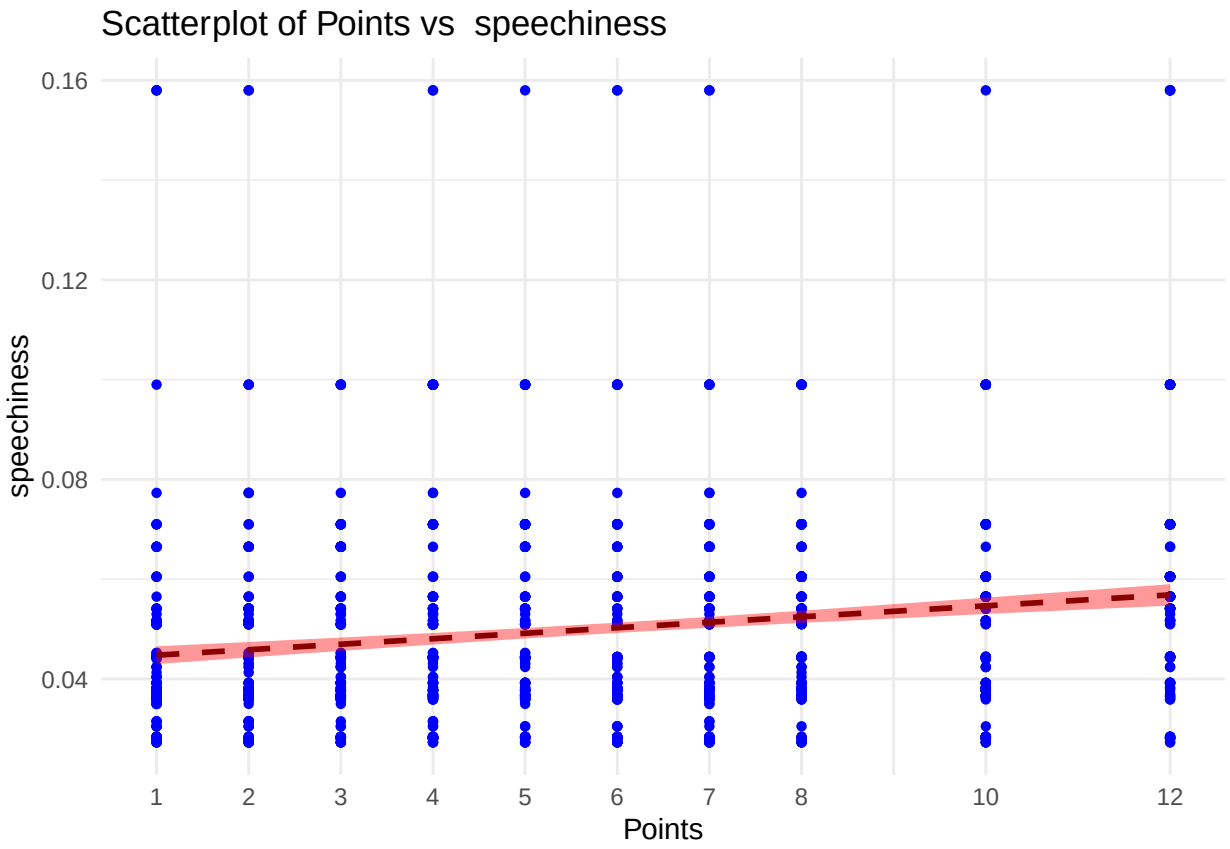

```
## Warning: Removed 194 rows containing non-finite values (stat_smooth).
## Removed 194 rows containing missing values (geom_point).
```

```
## Saving 6.5 x 4.5 in image
```

```
## Warning: Removed 194 rows containing non-finite values (stat_smooth).
## Removed 194 rows containing missing values (geom_point).
```



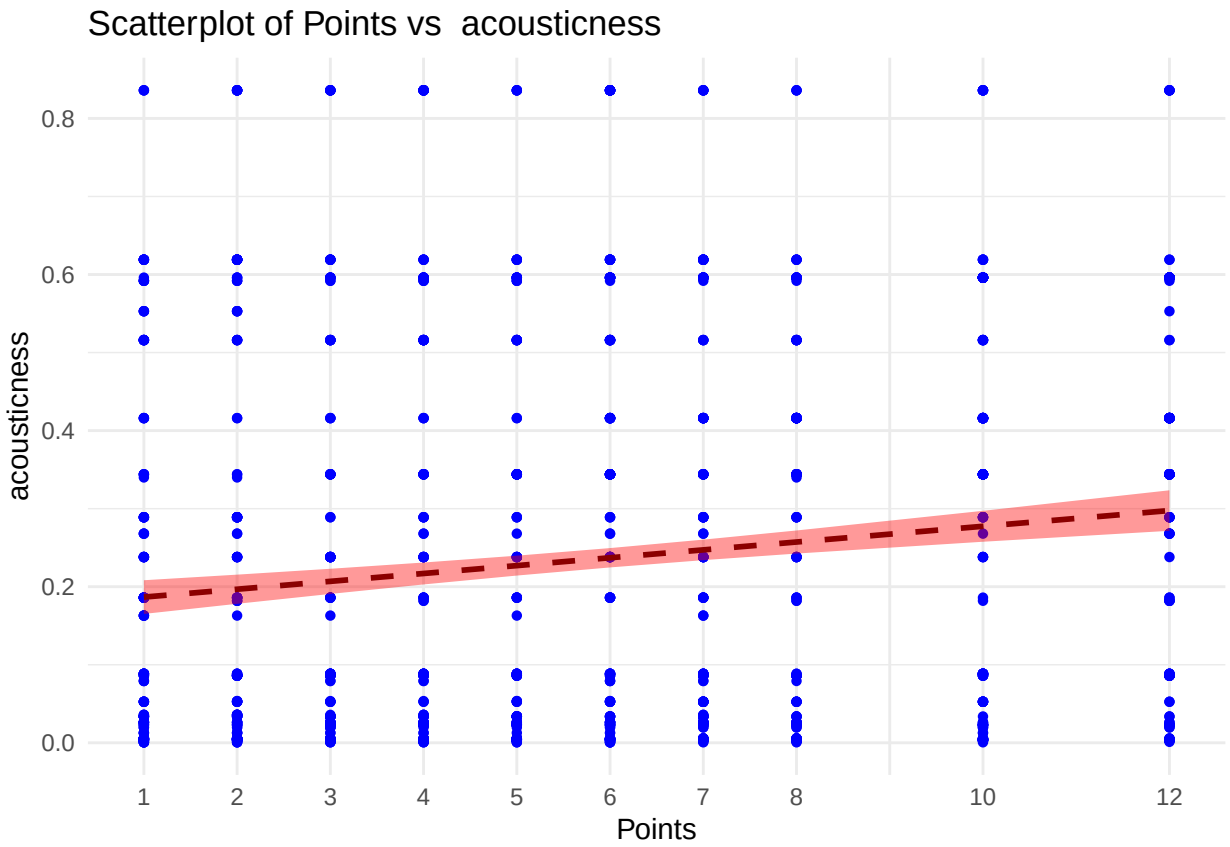

```
## Warning: Removed 194 rows containing non-finite values (stat_smooth).
## Removed 194 rows containing missing values (geom_point).

## Saving 6.5 x 4.5 in image

## Warning: Removed 194 rows containing non-finite values (stat_smooth).
## Removed 194 rows containing missing values (geom_point).
```




```
## Warning: Removed 194 rows containing non-finite values (stat_smooth).
## Removed 194 rows containing missing values (geom_point).
```

```
## Saving 6.5 x 4.5 in image
```

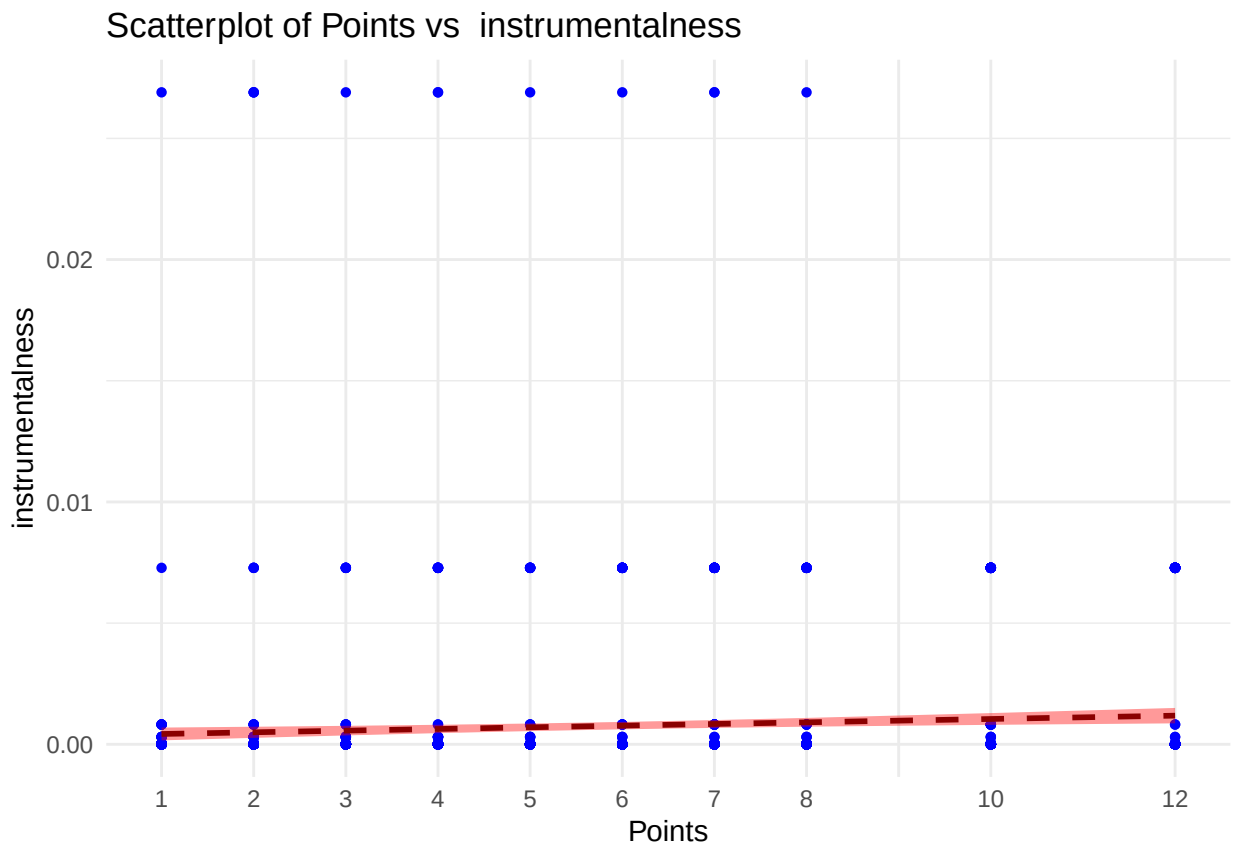
```
## Warning: Removed 194 rows containing non-finite values (stat_smooth).
## Removed 194 rows containing missing values (geom_point).
```



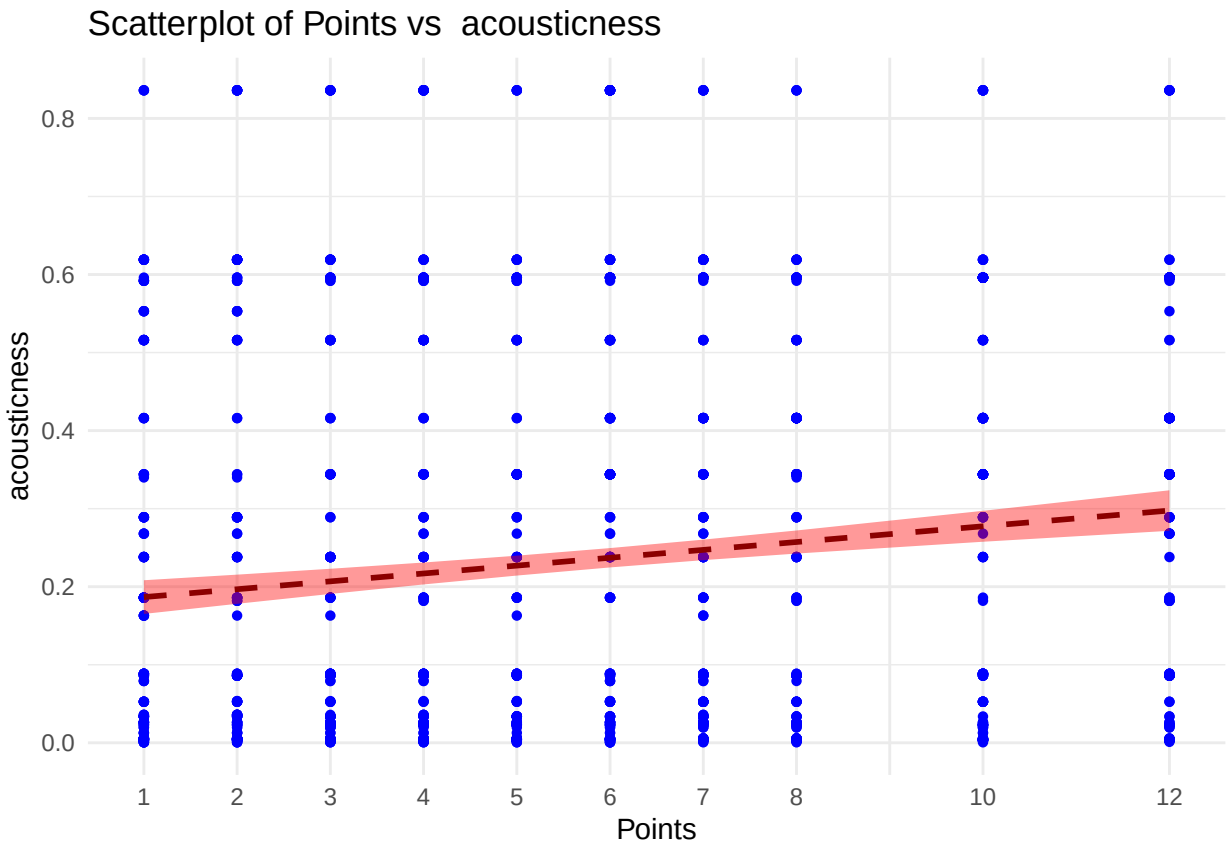

```
## Warning: Removed 194 rows containing non-finite values (stat_smooth).
## Removed 194 rows containing missing values (geom_point).
```

```
## Saving 6.5 x 4.5 in image
```

```
## Warning: Removed 194 rows containing non-finite values (stat_smooth).
## Removed 194 rows containing missing values (geom_point).
```



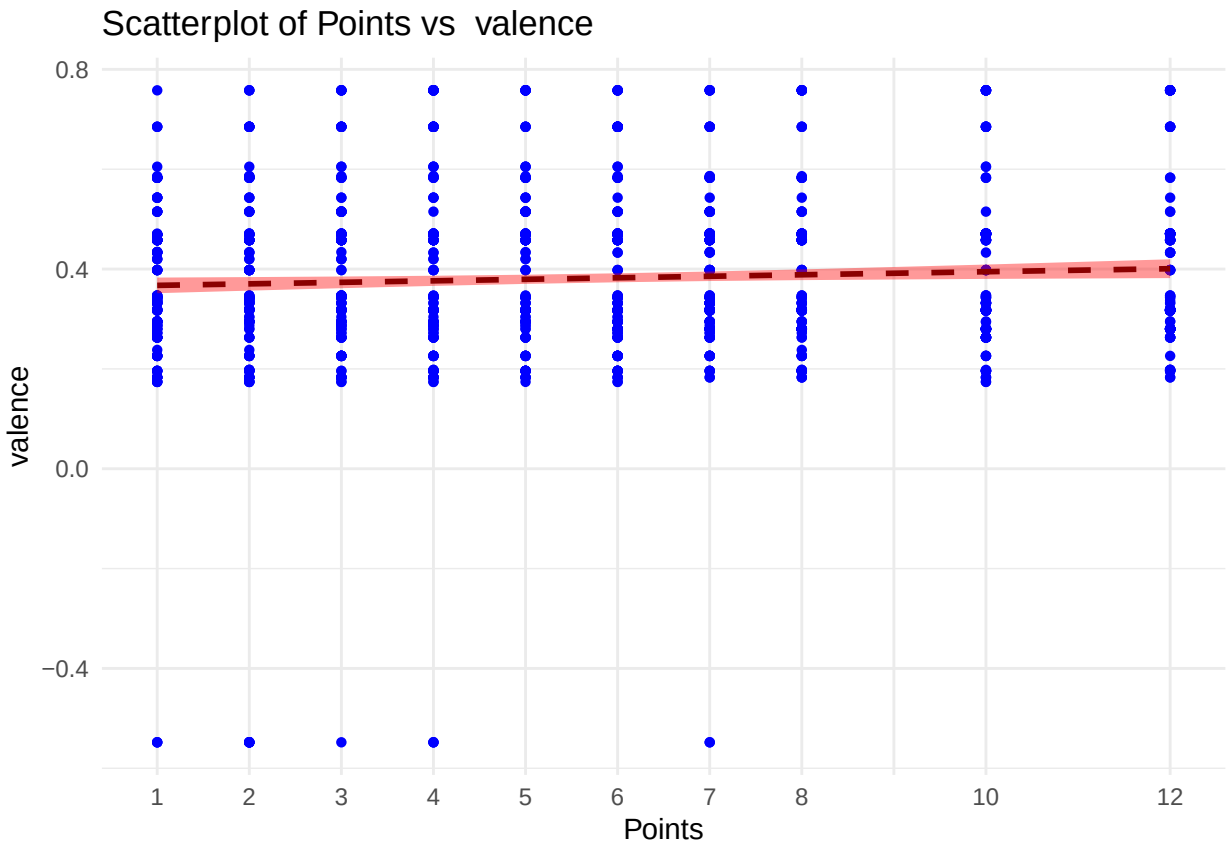

```
## Warning: Removed 194 rows containing non-finite values (stat_smooth).
## Removed 194 rows containing missing values (geom_point).

## Saving 6.5 x 4.5 in image

## Warning: Removed 194 rows containing non-finite values (stat_smooth).
## Removed 194 rows containing missing values (geom_point).
```



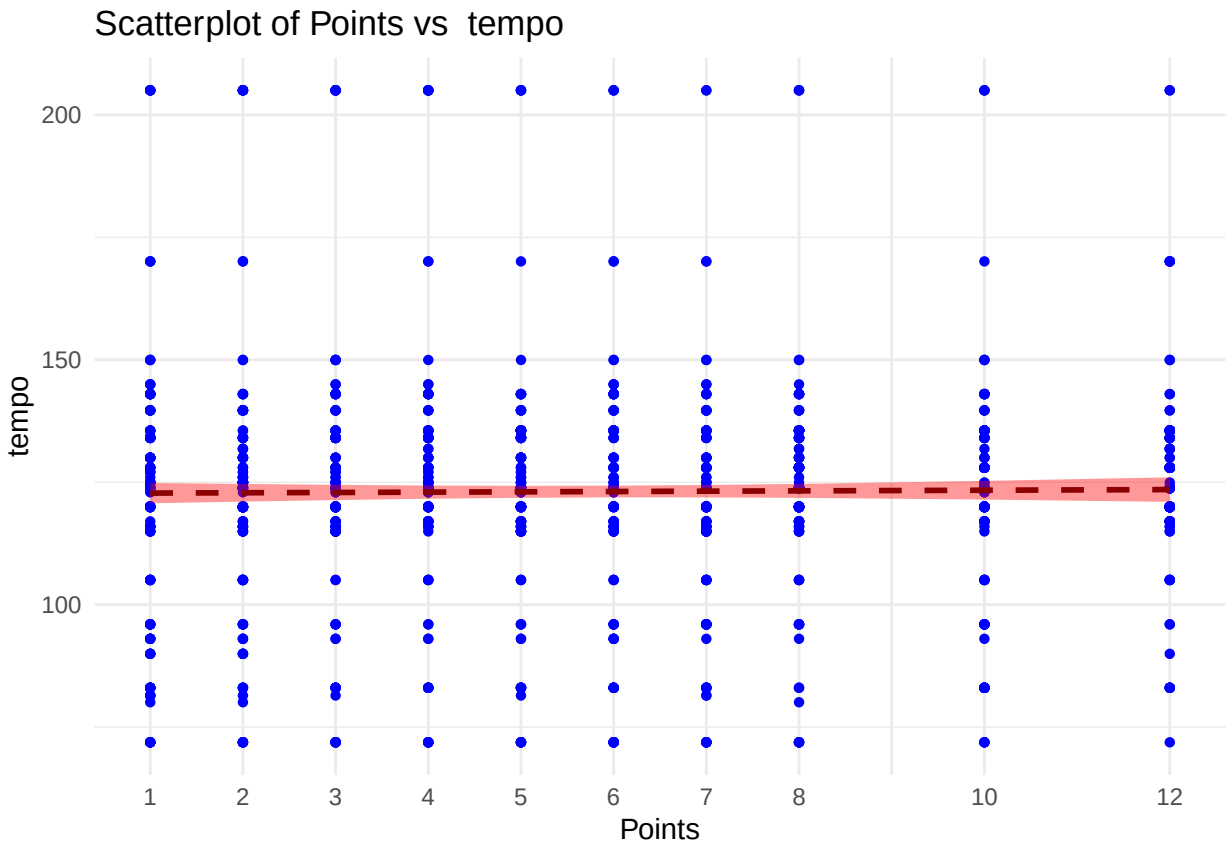

```
## Warning: Removed 194 rows containing non-finite values (stat_smooth).
## Removed 194 rows containing missing values (geom_point).
```

```
## Saving 6.5 x 4.5 in image
```

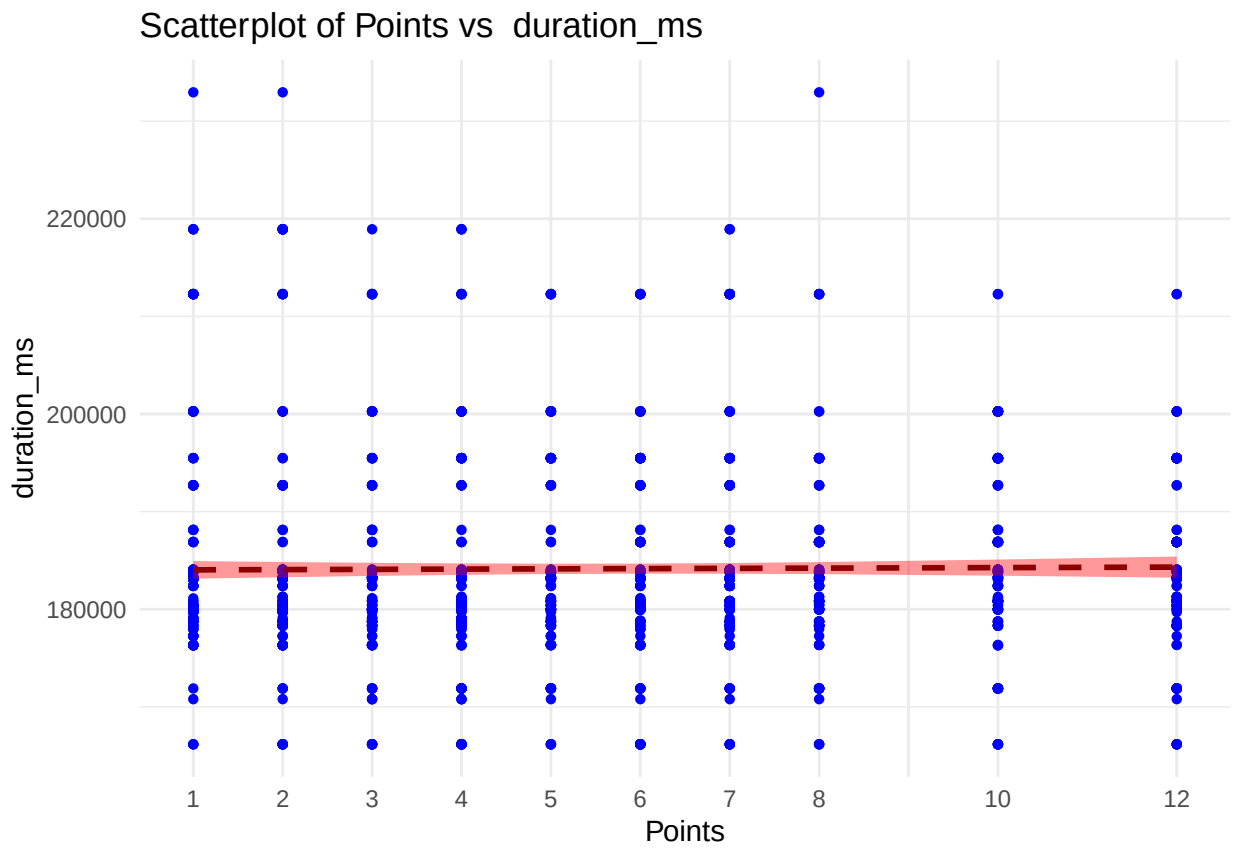
```
## Warning: Removed 194 rows containing non-finite values (stat_smooth).
## Removed 194 rows containing missing values (geom_point).
```



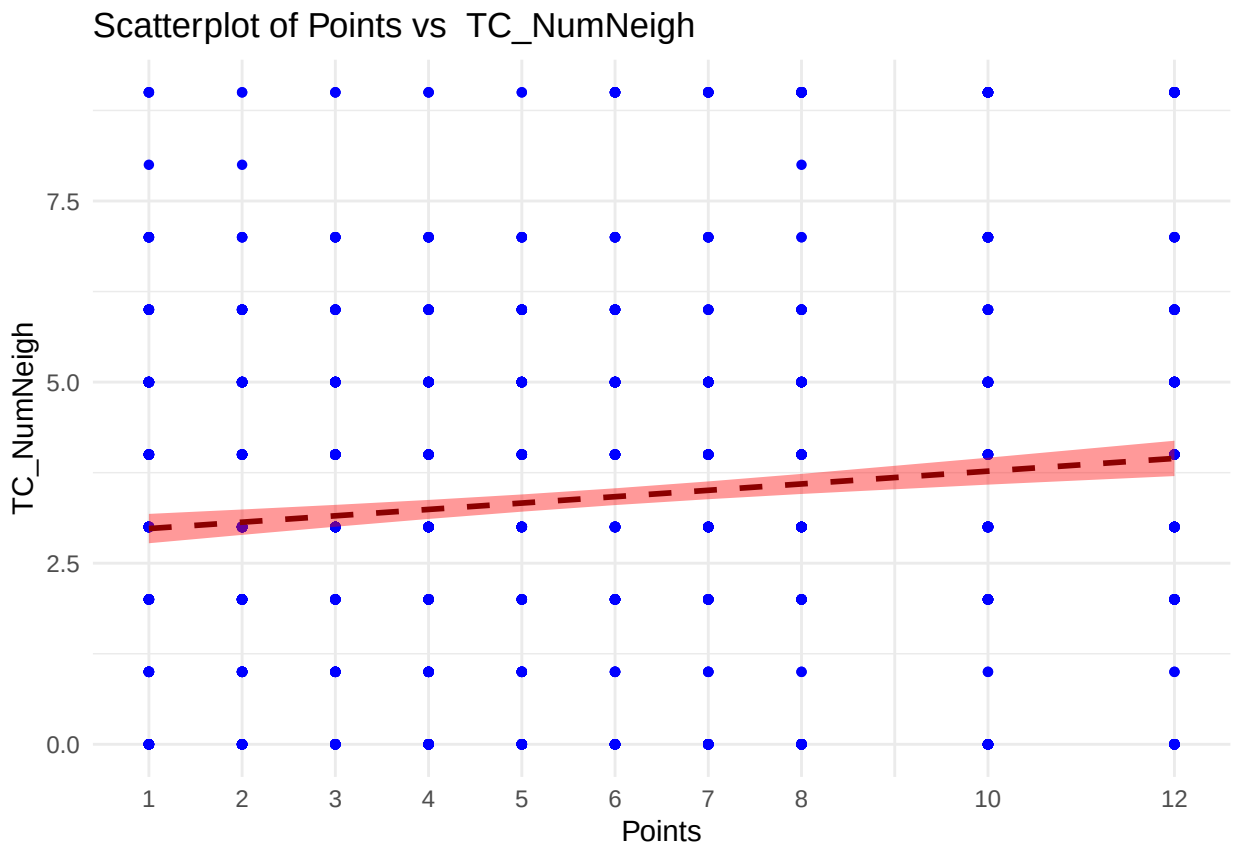

```
## Warning: Removed 194 rows containing non-finite values (stat_smooth).
## Removed 194 rows containing missing values (geom_point).

## Saving 6.5 x 4.5 in image

## Warning: Removed 194 rows containing non-finite values (stat_smooth).
## Removed 194 rows containing missing values (geom_point).
```



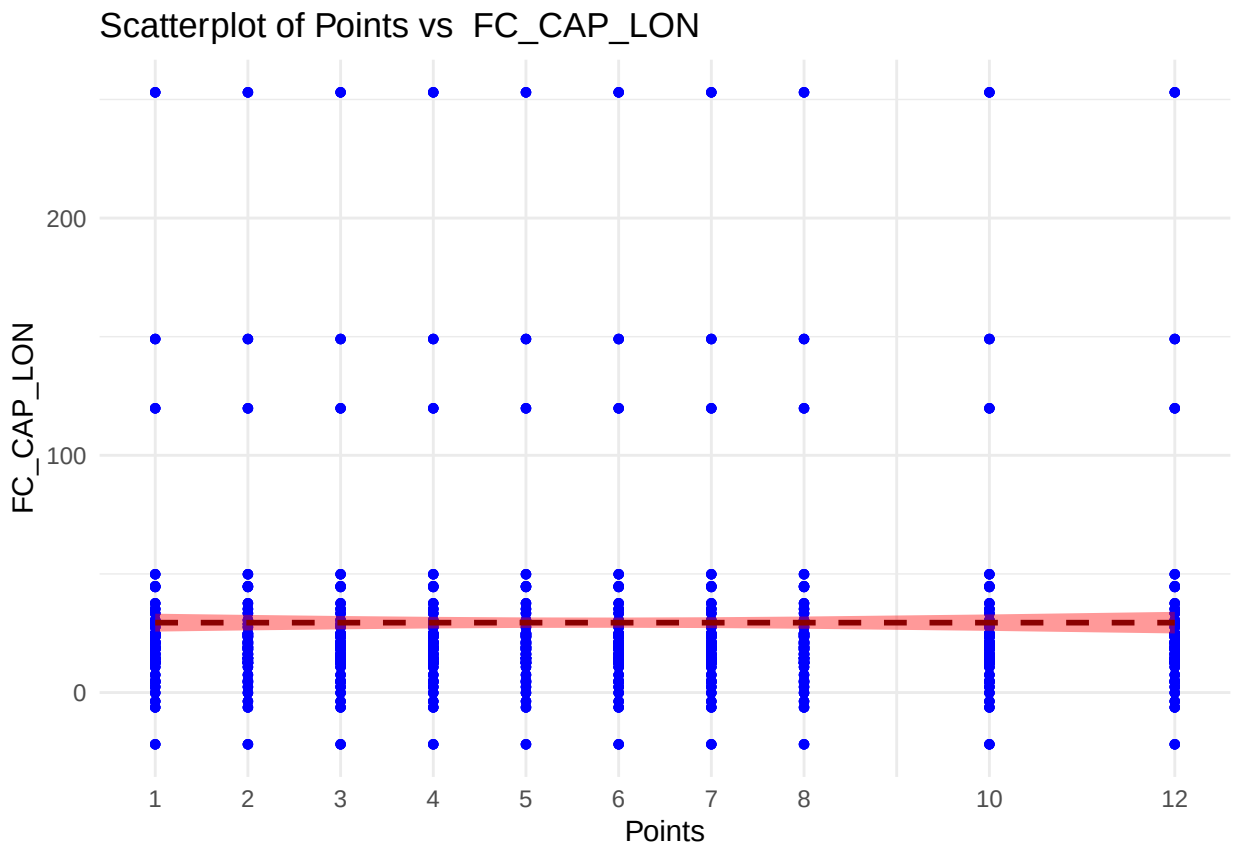

```
## Warning: Removed 194 rows containing non-finite values (stat_smooth).
## Removed 194 rows containing missing values (geom_point).

## Saving 6.5 x 4.5 in image

## Warning: Removed 194 rows containing non-finite values (stat_smooth).
## Removed 194 rows containing missing values (geom_point).
```



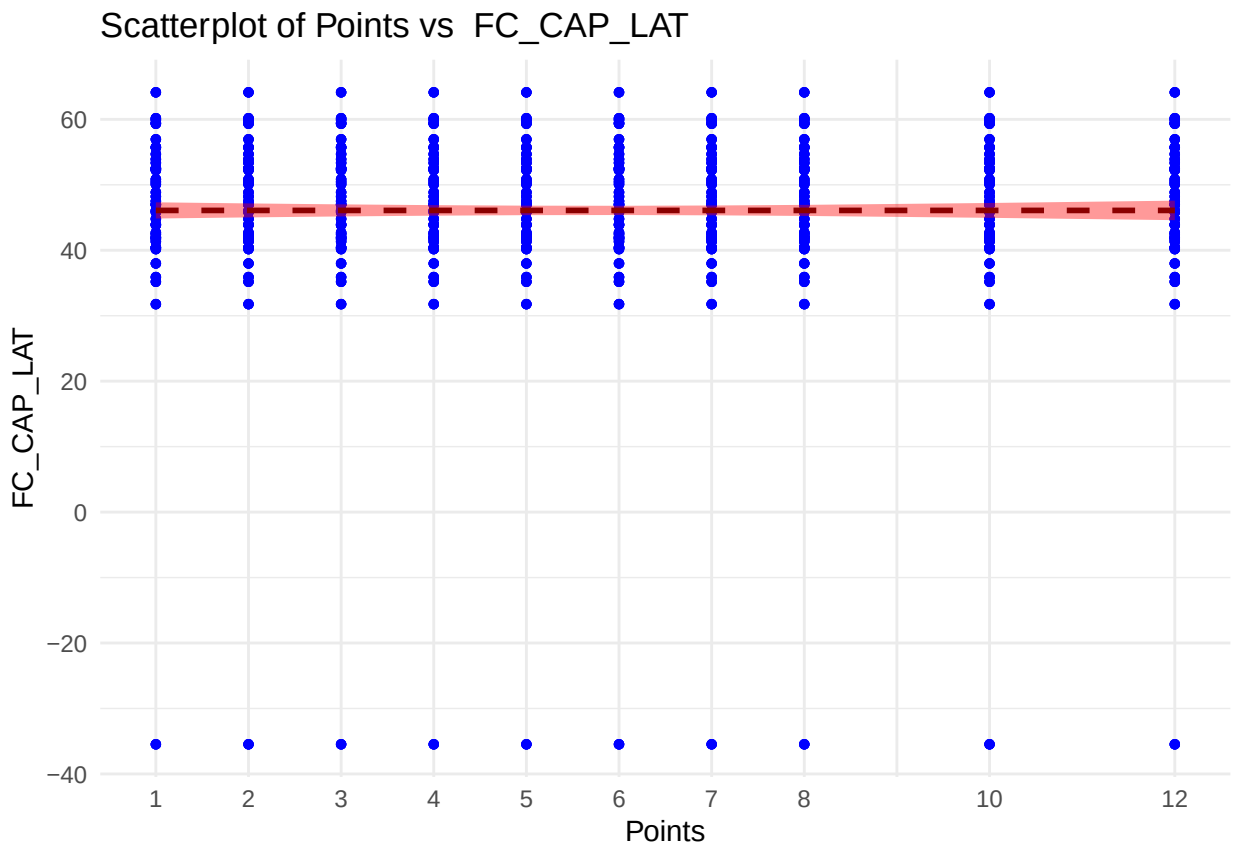

```
## Saving 6.5 x 4.5 in image
```



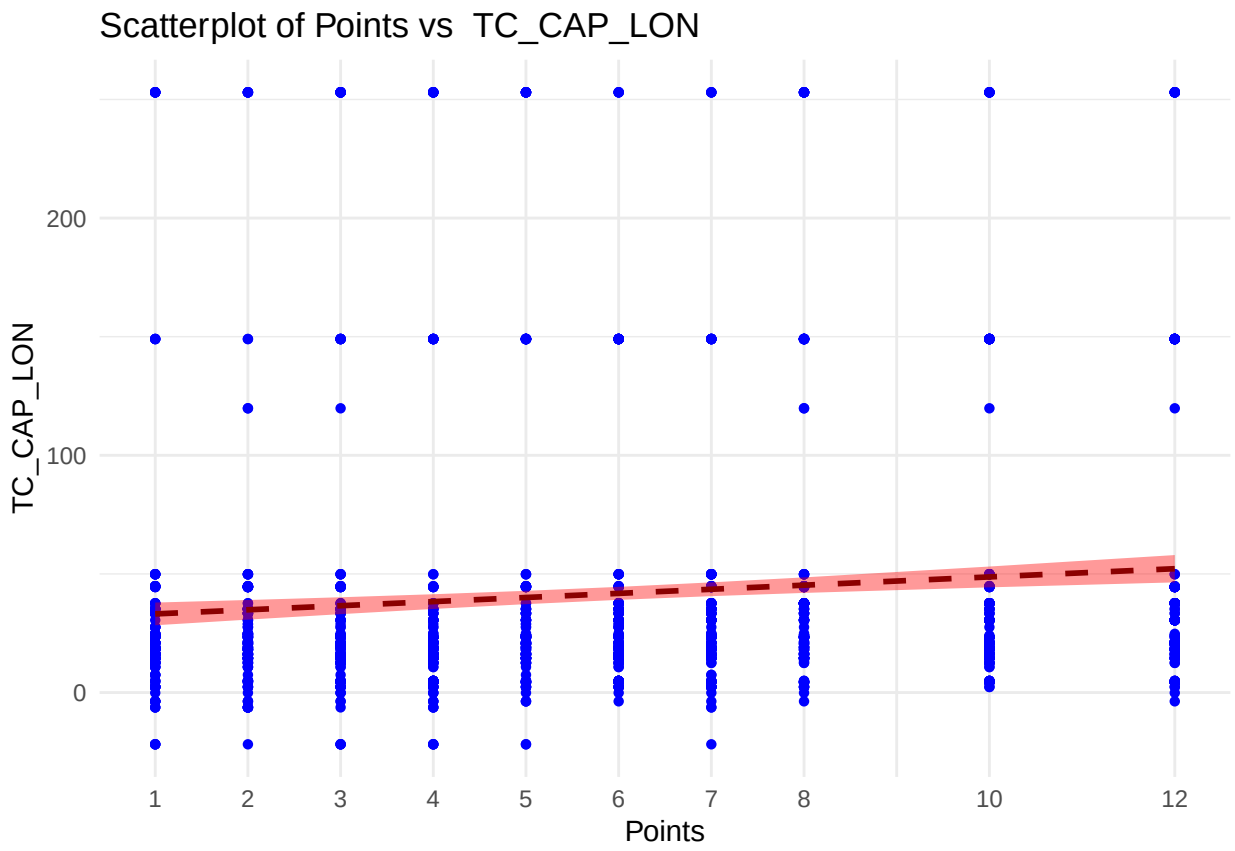

```
## Saving 6.5 x 4.5 in image
```



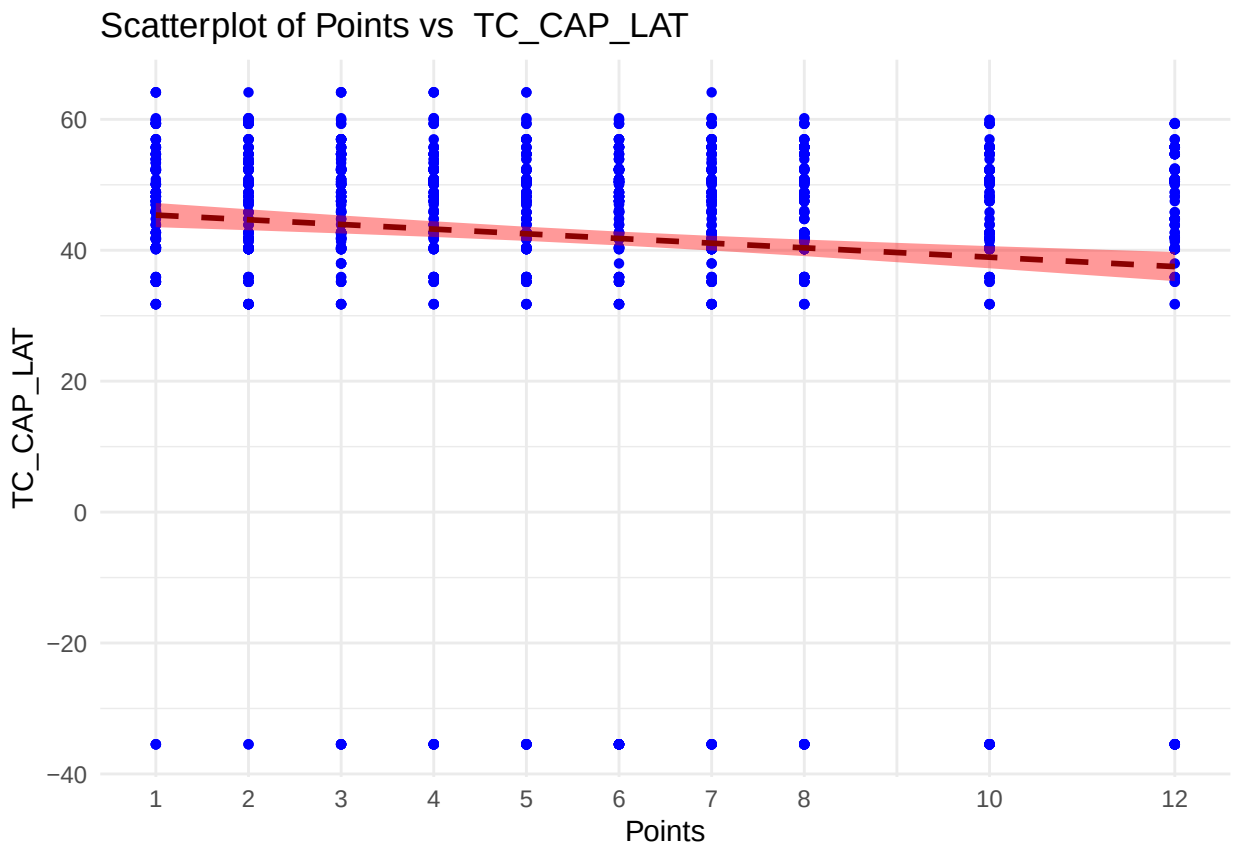

```
## Saving 6.5 x 4.5 in image
```



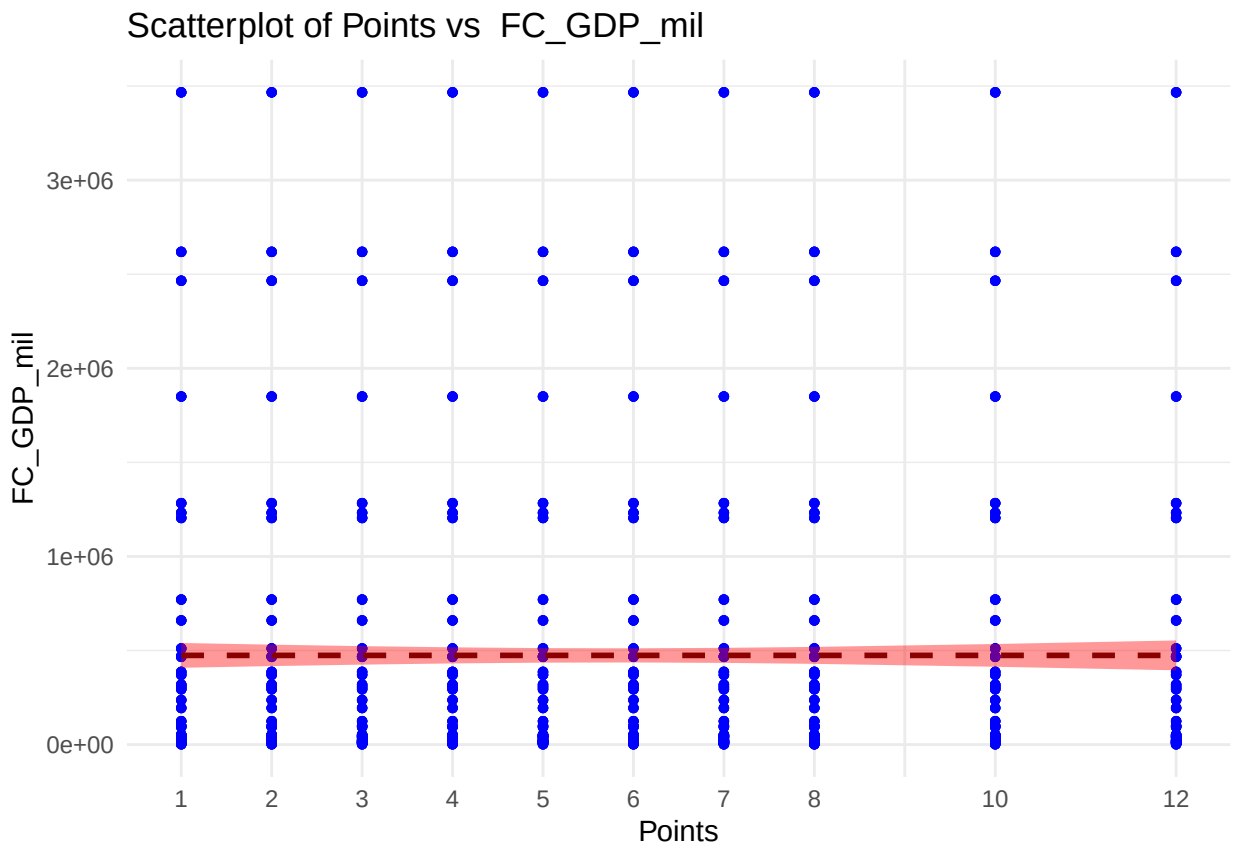

```
## Saving 6.5 x 4.5 in image
```



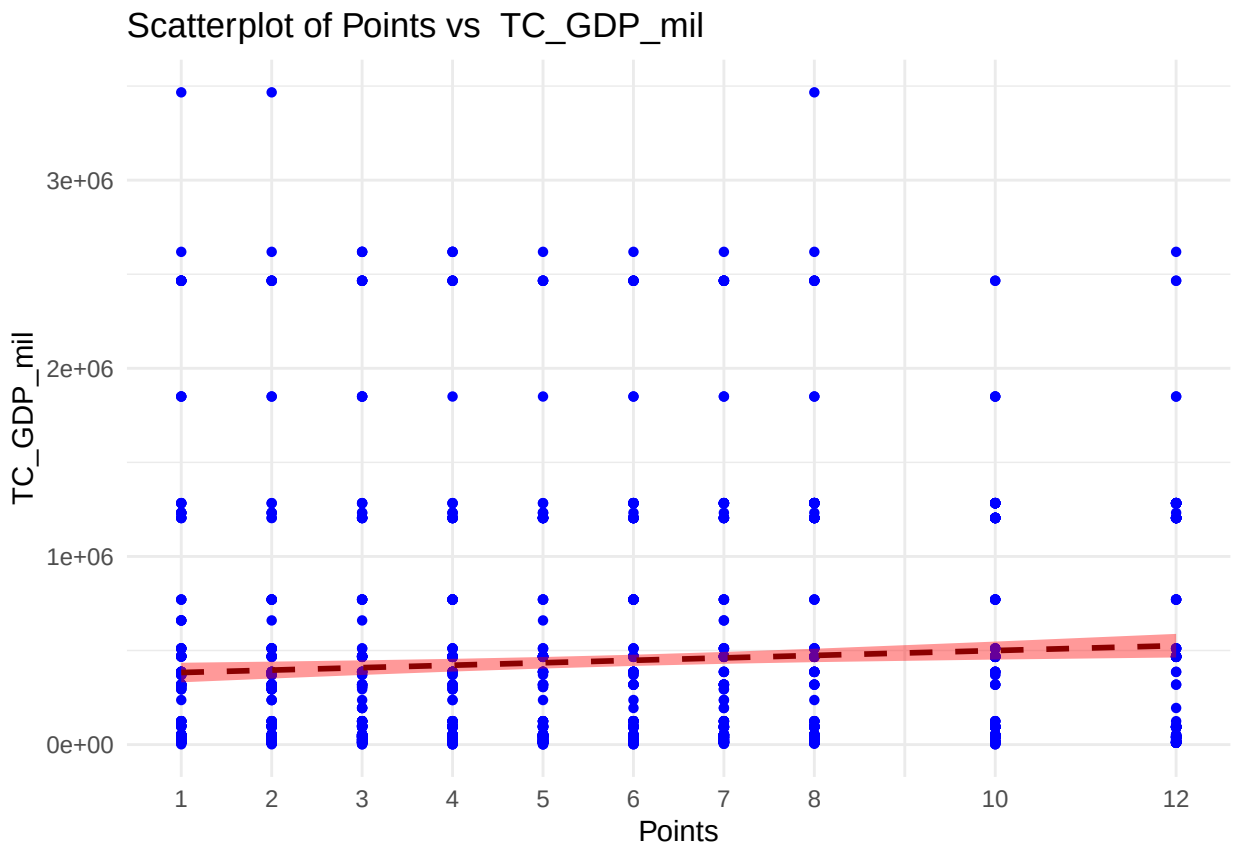

```
## Saving 6.5 x 4.5 in image
```



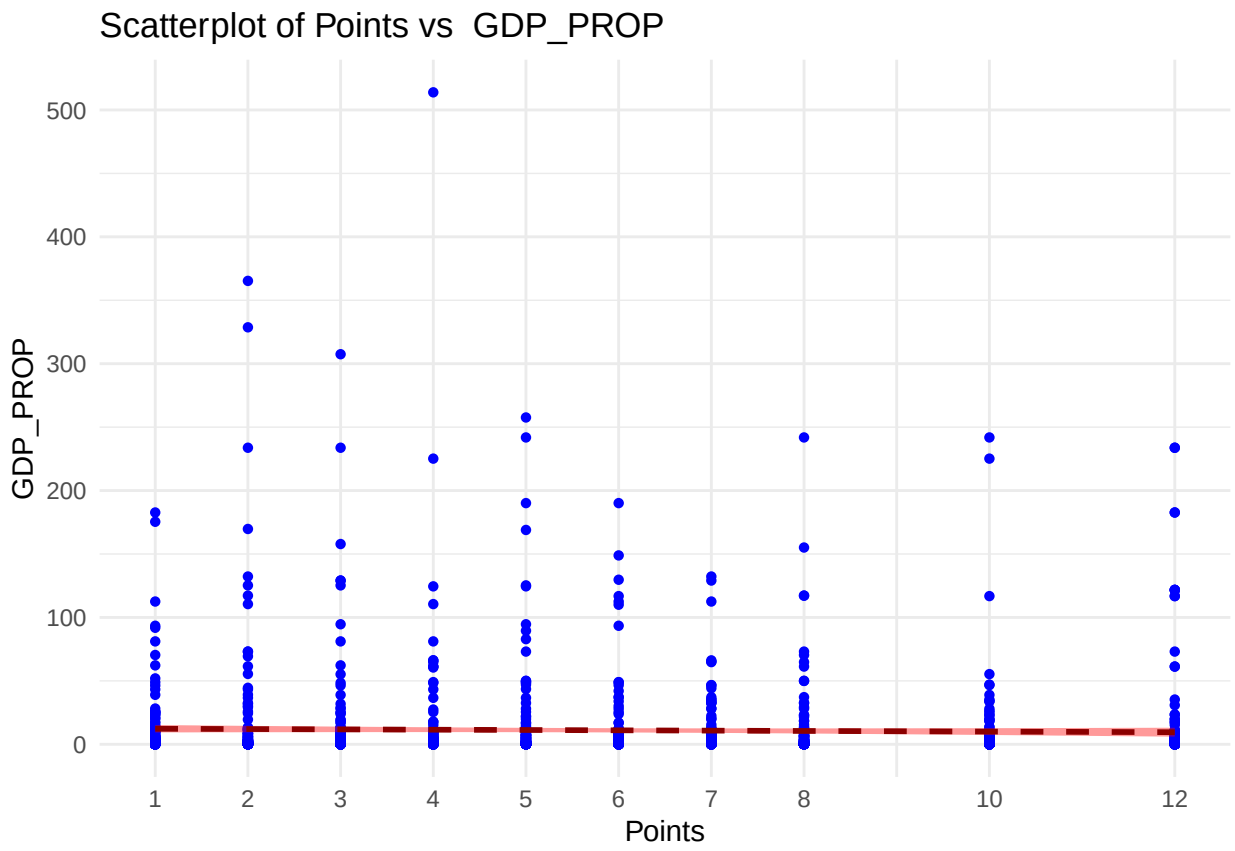

```
## Saving 6.5 x 4.5 in image
```



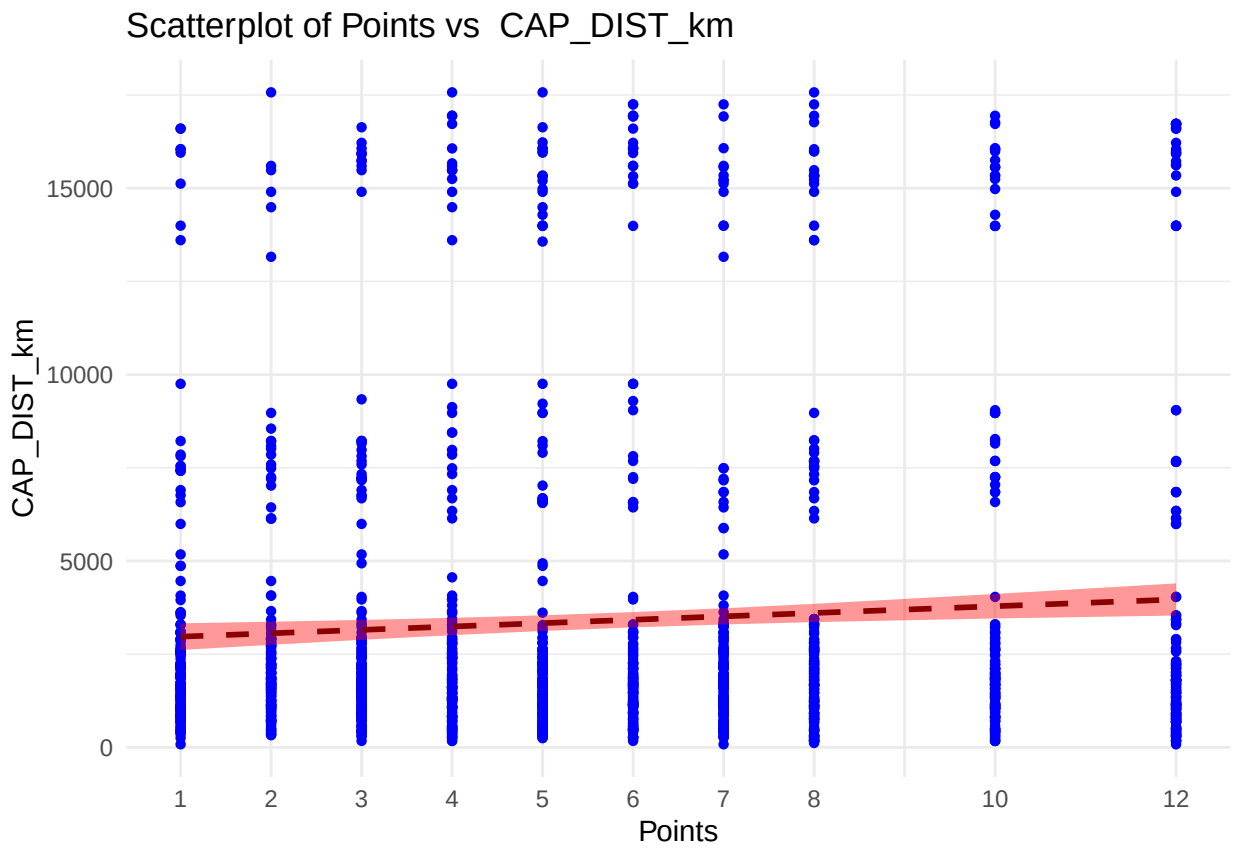

```
## Saving 6.5 x 4.5 in image
```



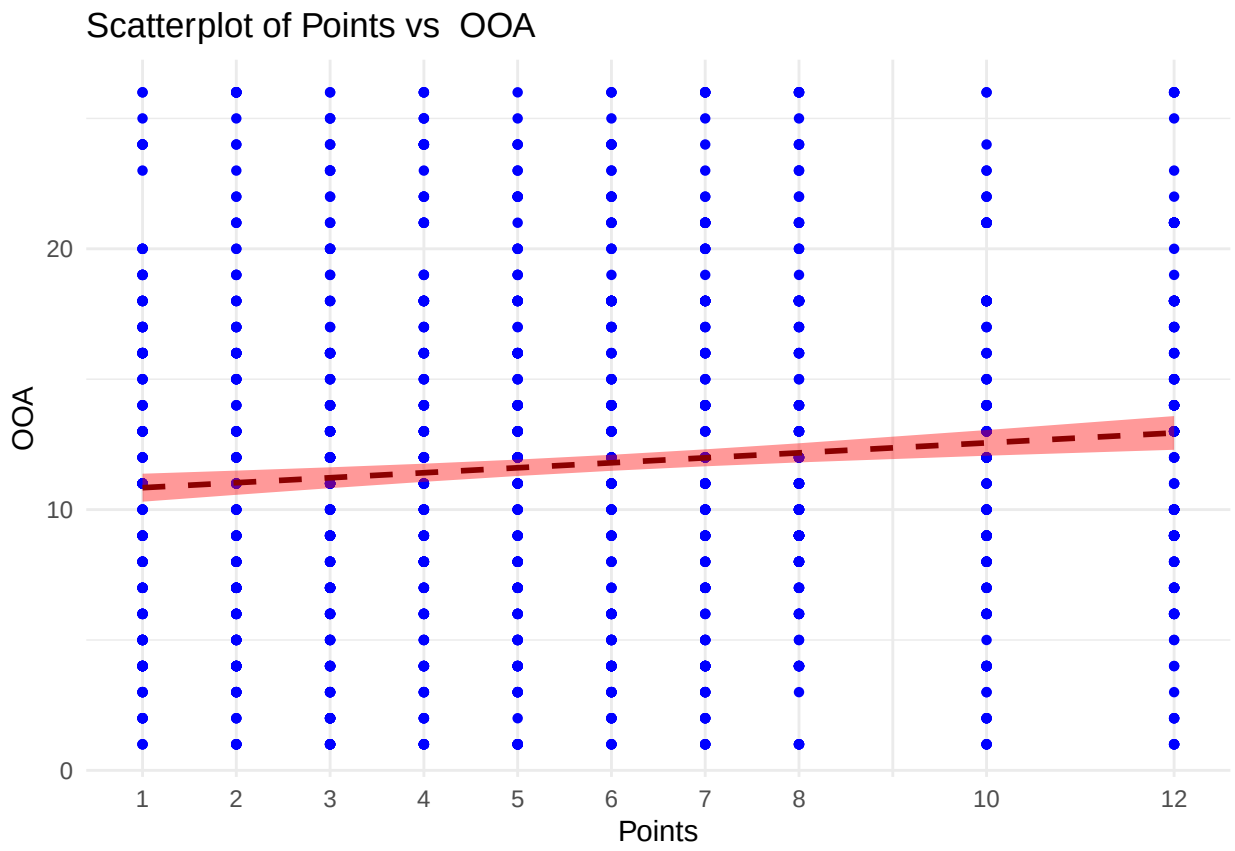

```
## Saving 6.5 x 4.5 in image
```




```
## Saving 6.5 x 4.5 in image
```




```
## Saving 6.5 x 4.5 in image
```




```
## [1] 0
```

```
# execute the correlation tests against the response
resp_cor_tests_df <- corr_tests(data = ESCdata, col_names = all_num)
# Write the correlation results to  a .csv file
write.csv(x = resp_cor_tests_df, file = "./report/stats/cor_tests_response.csv", row.names = F)

# run correlation tests for all predictors
pred_cor_tests_df <- pred_corr_tests(ESCdata[all_num])
# output results as a .csv file
write.csv(x = pred_cor_tests_df, file = "./report/Stats/cor_tests_predictors.csv", row.names = F)

# Determine which variables are highly correlated
pred_cor_tests_df[pred_cor_tests_df$Correlation > 0.8,]
```

	X	Y	Correlation	P-Value
30	FC_NonCOB	FC_NonCitzens	0.91531	0
85	FC_COB	FC_Citizens	0.99978	0
86	FC_COB	FC_Population	0.99914	0
106	FC_COB	FC_GDP_mil	0.92752	0
111	FC_Citizens	FC_Population	0.99904	0
131	FC_Citizens	FC_GDP_mil	0.93506	0
155	FC_Population	FC_GDP_mil	0.91609	0
160	METRIC_COB	METRIC_Citizens	0.83307	0

	X	Y	Correlation	P-Value
161	METRIC_COB	METRIC_COBCit	0.96724	0
183	METRIC_Citizens	METRIC_COBCit	0.94621	0
317	acousticness	acousticness.1	1.00000	0

Social Networks

A variety of Social Networks based on different closeness metrics given a specified Voting_Method. Here I shall explore the effect of the televote and the Jury Vote. For example see below a high diaspora graph

```
# NOTE: that there is more data missing for METIC_Citizens than METRIC_COB
sum(!is.na(ESCdata$METRIC_COB))
```

```
## [1] 734
```

```
sum(!is.na(ESCdata$METRIC_Citizens))
```

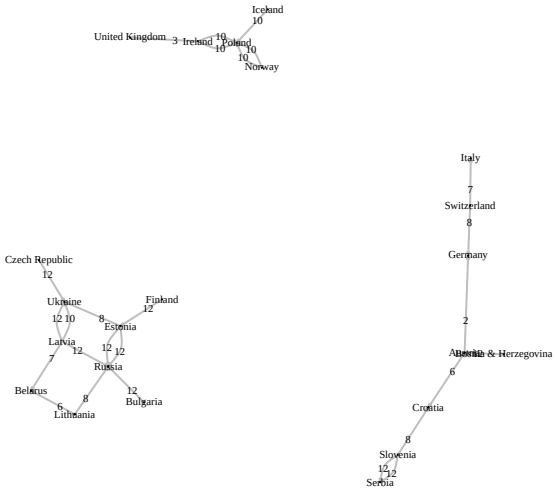
```
## [1] 849
```

```
networkdata <- ESCdata %>% filter(!is.na(METRIC_Citizens))
```

```
# generate a summary of metric cob
summary(networkdata$METRIC_COB)
```

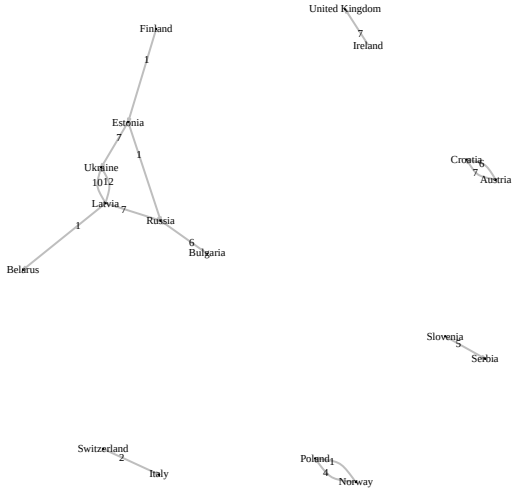
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
## 0.00000 0.00107 0.00383 0.02240 0.01274 0.65079     116
```

```
# (1) Voting_Method = Televote
graph_network(dataset = filter(networkdata, Voting_Method == "T" & METRIC_COB > 0.1), weights = 'Points
```




```
## [1] 0
```

```
# (2) Voting_Method = Jury
graph_network(dataset = filter(networkdata, Voting_Method == "J" & METRIC_COB > 0.1), weights = 'Points
```

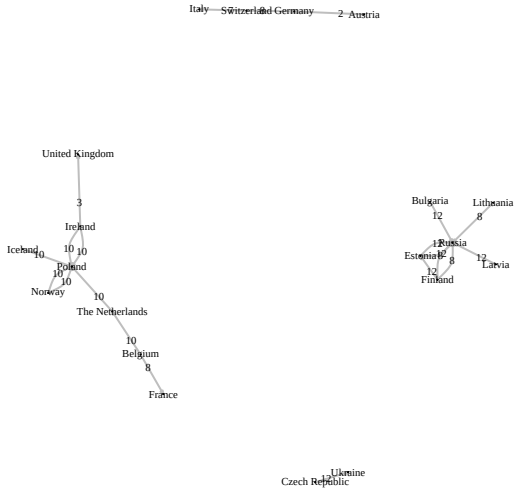



```
## [1] 0
```

```
# generate a summary of metric citizens
summary(networkdata$METRIC_Citizens)
```

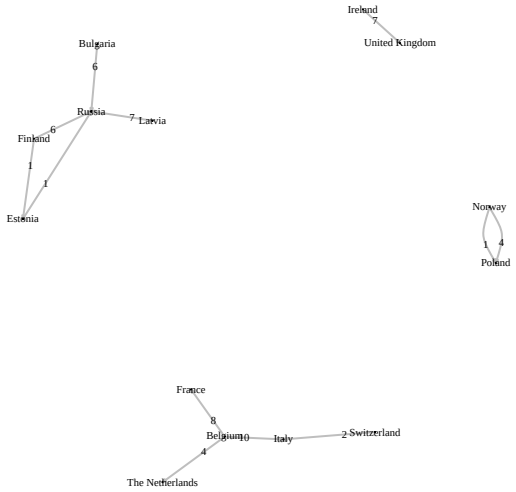
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.00000 0.00123 0.00421 0.01898 0.01325 0.47639
```

```
# (1) Voting_Method = Televote
graph_network(dataset = filter(networkdata, Voting_Method == "T" & METRIC_Citizens > 0.1), weights = 'P
```




```
## [1] 0
```

```
# (2) Voting_Method = Jury
# Construct Social Network
graph_network(dataset = filter(networkdata, Voting_Method == "J" & METRIC_Citizens > 0.1), weights = 'P
```




```
## [1] 0
```

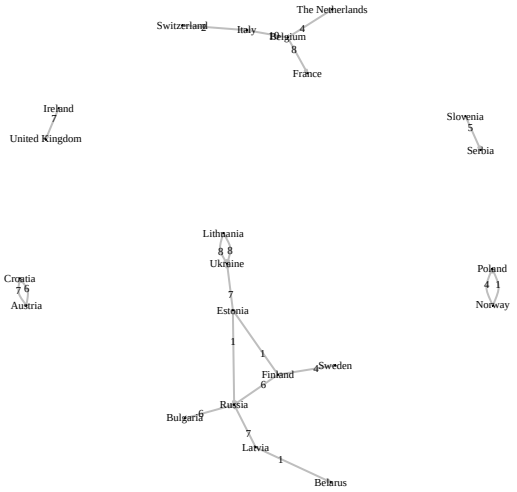
```
# generate a summary of metric cobcit
summary(networkdata$METRIC_COBCit)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
## 0.00000 0.00252 0.00815 0.04275 0.02925 1.09300     116
```

```
# (1) Voting_Method = Televote
graph_network(dataset = filter(networkdata, Voting_Method == "T" & METRIC_COBCit > 0.15), weights = 'Po
```

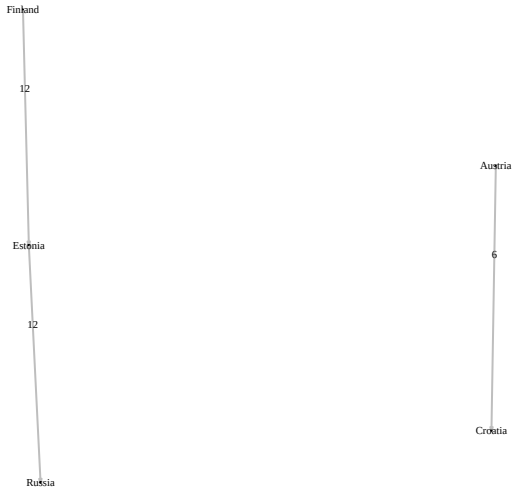
```
## [1] 0
```

```
# (2) Voting_Method = Jury
graph_network(dataset = filter(networkdata, Voting_Method == "J" & METRIC_COBCit > 0.15), weights = 'Po
```



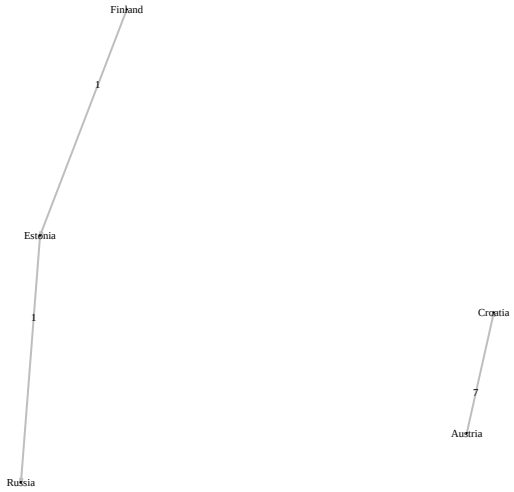

```
## [1] 0
```

```
# Further subdivide the data into semi-finals and final
# (1) Voting_Method = Televote & Round = sf1
graph_network(dataset = filter(networkdata, Voting_Method == "T" & METRIC_COBCit > 0.25 & Round == "sf1
```




```
## [1] 0
```

```
# (2) Voting_Method = Jury
graph_network(dataset = filter(networkdata, Voting_Method == "J" & METRIC_COBCit > 0.25 & Round == "sf1
```

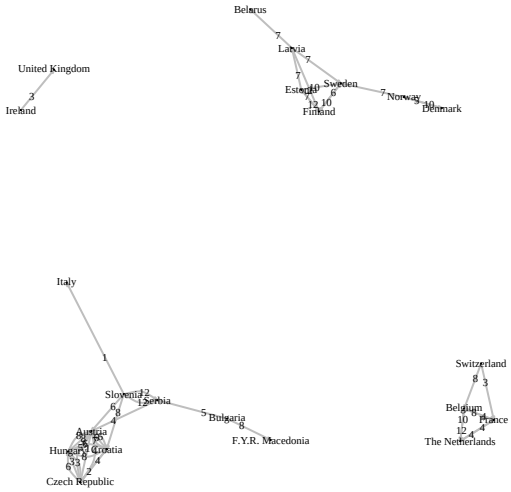



```
## [1] 0
```

```
# generate a summary of capital distance in km
summary(networkdata$CAP_DIST_km)
```

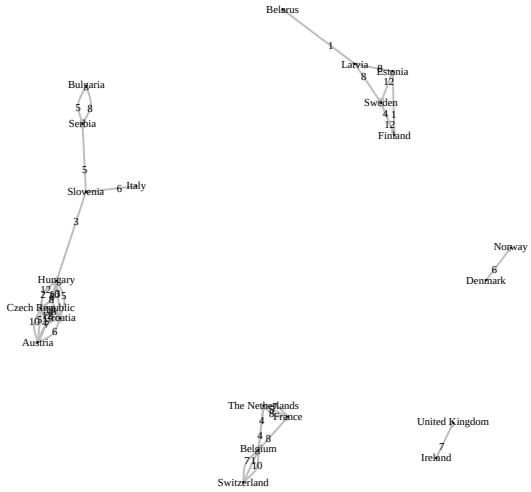
```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##    82.15   936.12  1585.71  3049.09  2862.54 17247.90
```

```
# (1) Voting_Method = Televote
graph_network(dataset = filter(networkdata, Voting_Method == "T" & CAP_DIST_km < 500), weights = 'Point
```




```
## [1] 0
```

```
# (2) Voting_Method = Jury
graph_network(dataset = filter(networkdata, Voting_Method == "J" & CAP_DIST_km < 500), weights = 'Points
```

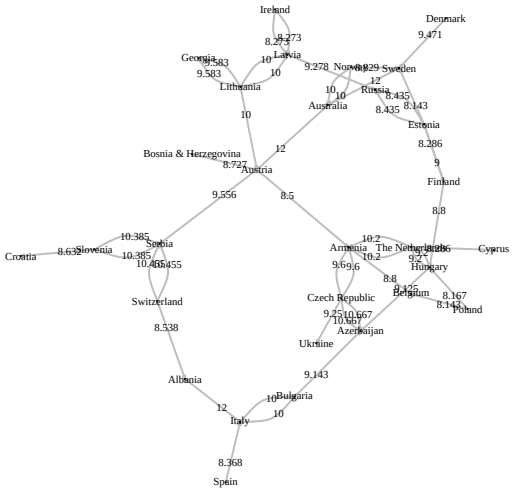



```
## [1] 0
```

```
# generate a summary of average point value
summary(networkdata$Average_Points)
```

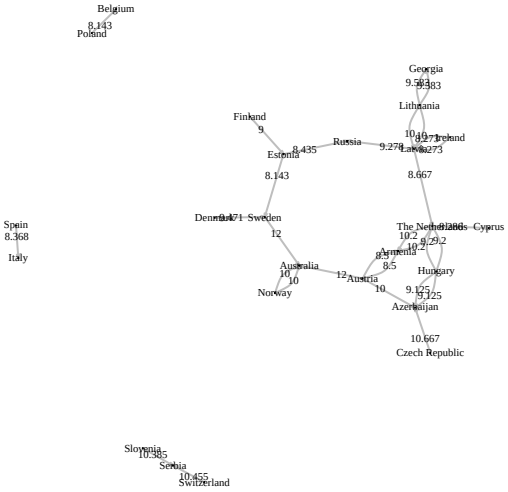
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   4.000   5.444   5.296   6.800  12.000
```

```
# (1) Voting_Method = Televote
graph_network(dataset = filter(networkdata, Voting_Method == "T" & Average_Points > 8), weights = 'Avera
```




```
## [1] 0
```

```
# (2) Voting_Method = Jury
graph_network(dataset = filter(networkdata, Voting_Method == "J" & Average_Points > 8), weights = 'Avera
```


Belgium
8.143
Poland
Georgia
9.583
9.583
Lithuania
Finland
9
Russia
9.278
Ireland
8.435
Estonia
8.143
8.667
Spain
8.368
Italy
Sweden
12
Cyprus
10.2
9.2
9.2
Australia
10
10
12
Austria
10
9.125
9.125
Hungary
Norway
Azerbaijan
10.667
Czech Republic
Slovenia
10.385
Serbia
10.455
Switzerland

```
## [1] 0
```