1. Visualizing Amounts and Distributions

```
setwd("D:/DSA0613")
getwd()
tips <- read.csv("tips.csv")
head(tips)
#import libraries
library(ggplot2)
library(dplyr)
library(ggridges)
#Bar plot
ggplot(tips, aes(x = day, y = tip, fill = day)) +
  stat_summary(fun = mean, geom = "bar") +
  labs(title = "Average Tip by Day",
       x = "Day",
       y = "Average Tip") +
  theme_minimal()
#Grouped Bar Plot
ggplot(tips, aes(x = day, y = tip, fill = gender)) +
  stat_summary(fun = mean, geom = "bar", position = "dodge")
#Stacked bar plot
ggplot(tips, aes(x = day, y = tip, fill = gender)) +
  stat_summary(fun = sum, geom = "bar")
#Dot plot
ggplot(tips, aes(x = tip, y = day)) +
  geom_point()
```

#Heatmap

```
ggplot(tips, aes(x = day, y = time, fill = tip)) +
  stat_summary(fun = mean, geom = "tile")
```

#Violin Plot

```
ggplot(tips, aes(x = day, y = tip)) +
  geom_violin(fill = "lightgreen")
```

#Ridgeline Plot

```
ggplot(tips, aes(x = tip, y = day)) +
  geom_density_ridges()
```

#Histogram

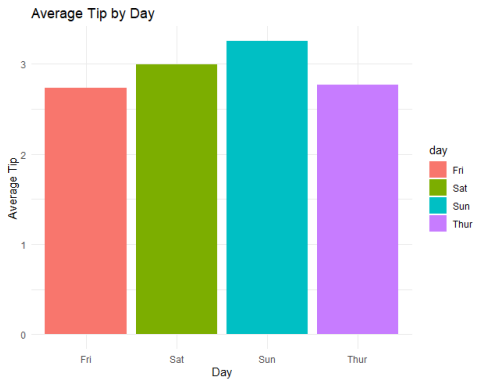
```
ggplot(tips, aes(x = tip)) +
  geom_histogram(binwidth = 1, fill = "red", color = "black")
```

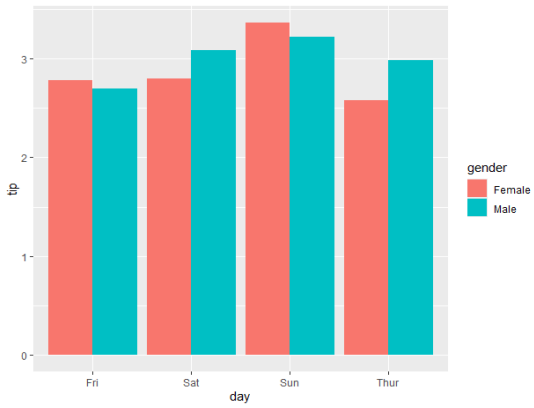
#Density Plot

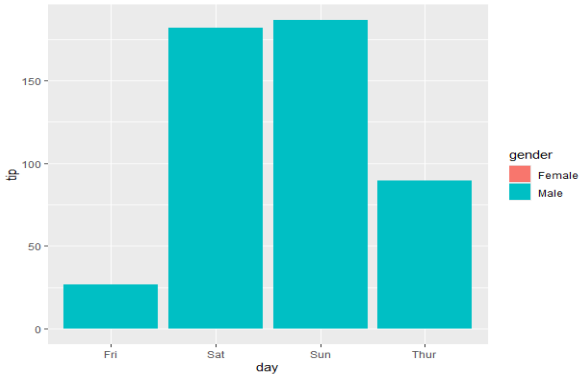
```
ggplot(tips, aes(x = tip)) +
  geom_density(fill = "skyblue")
```

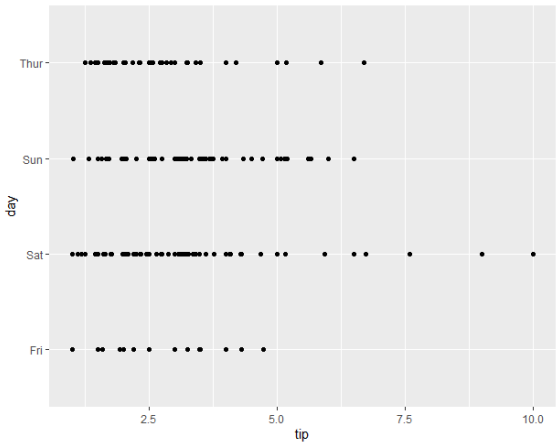
#Boxplot

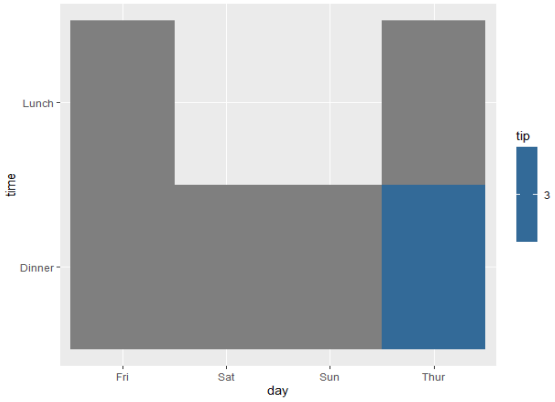
```
ggplot(tips, aes(x = day, y = tip)) +
  geom_boxplot()
```

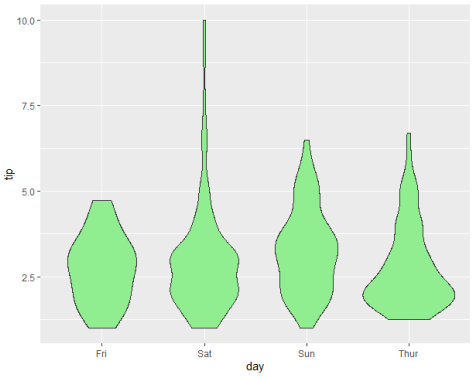


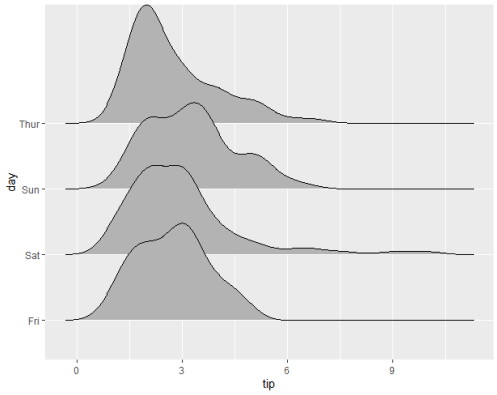


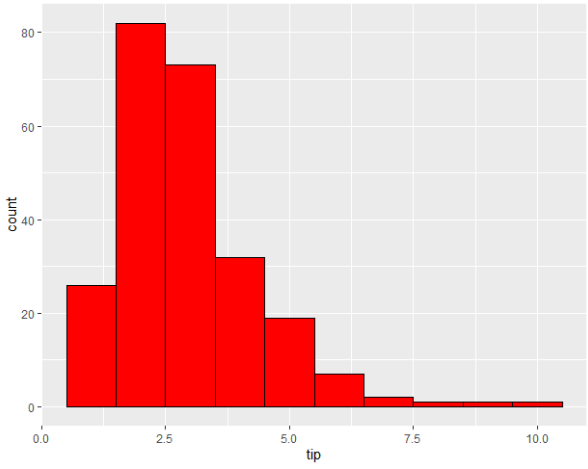

tip
150
100
50
0
Fri
Sat
Sun
Thur
day
gender
Female
Male

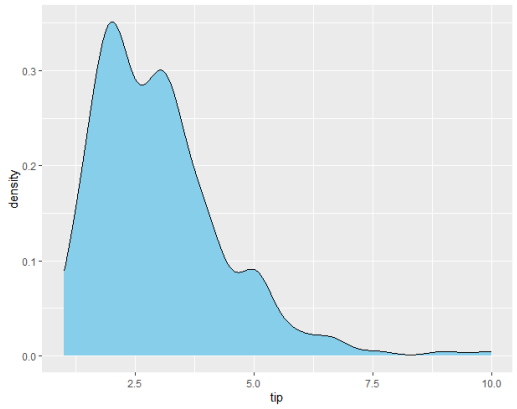

Thur
Sun
Sat
Fri
day
2.5
5.0
7.5
10.0
tip

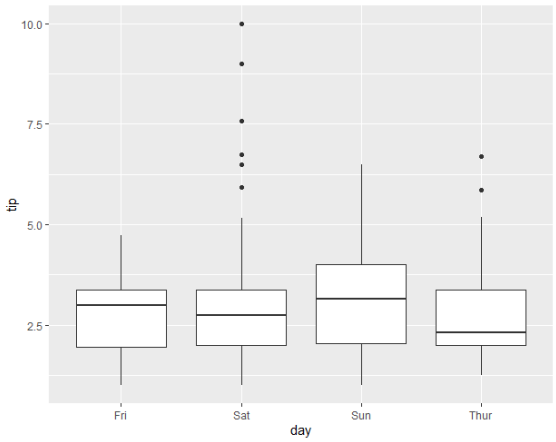

Lunch
Dinner
time
tip
3
Fri
Sat
Sun
Thur
day


10.0
7.5
5.0
2.5
tip
Fri
Sat
Sun
Thur
day


Thur
Sun
Sat
Fri
day
0
3
6
9
tip


80
60
40
20
0
count
0.0
2.5
5.0
7.5
10.0
tip


density
0.3
0.2
0.1
0.0
2.5
5.0
7.5
10.0
tip

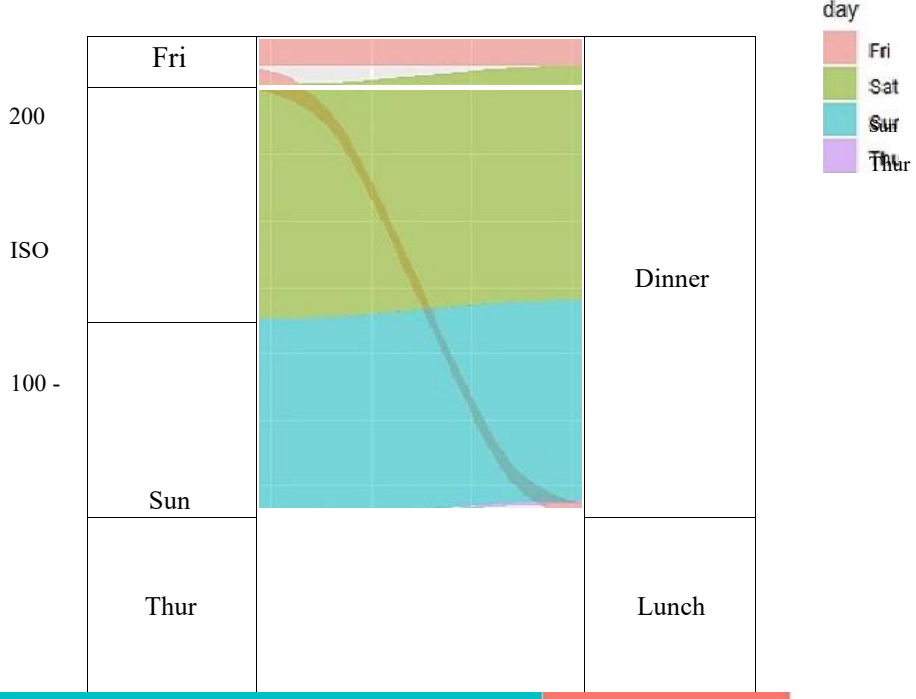

tip
10.0
7.5
5.0
2.5
Fri
Sat
Sun
Thur
day

2. Visualizing Proportions

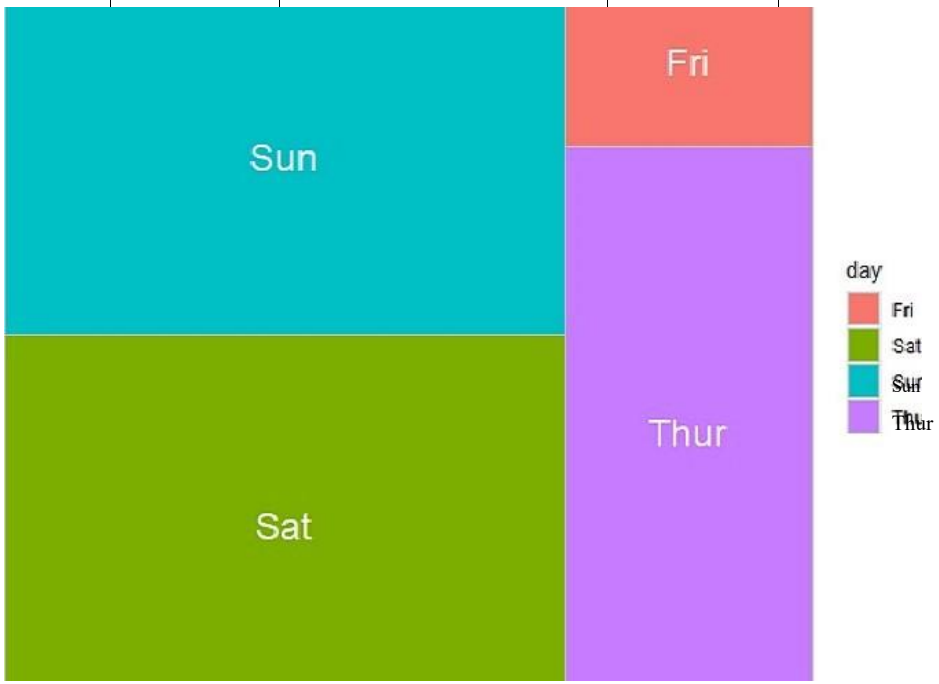
```
setwd("D:/DSA0613")
getwd()
tips <- read.csv("tips.csv")
head(tips)

#import libraries
library(ggplot2)
library(dplyr)
library(treemapify)
library(ggalluvial)
library(networkD3)

#pie chart
pie_data <- tips %>%
  group_by(day) %>%
  summarise(Value = sum(tip))

ggplot(pie_data, aes(x = "", y = Value, fill = day)) +
  geom_bar(stat = "identity", width = 1) +
  coord_polar("y") +
  labs(fill = "Day")

#tree map
ggplot(pie_data,
       aes(area = Value, fill = day, label = day)) +
  geom_treemap() +
  geom_treemap_text(colour = "white", place = "centre")
```

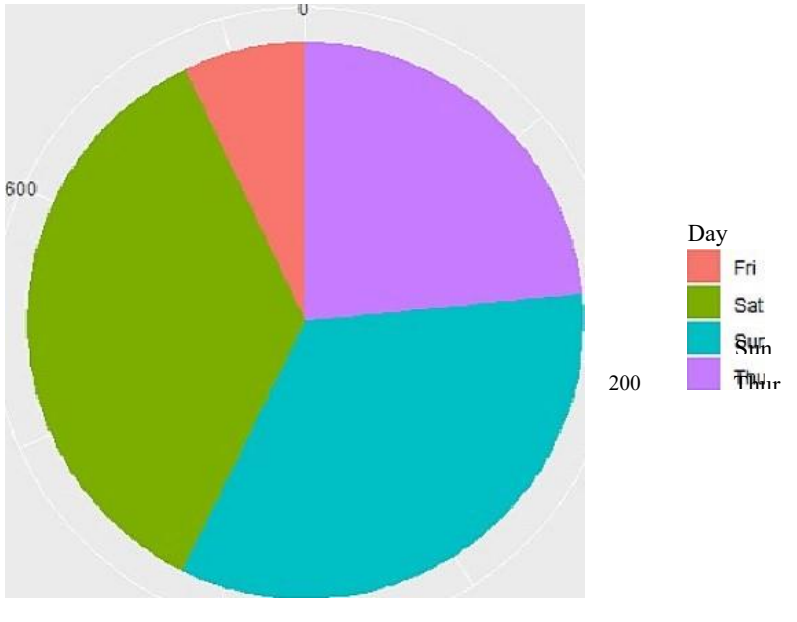
```
#sunburst chart
# Load libraries
library(dplyr)
library(sunburstR)

# Create hierarchical data (required for sunburst)
sunburst_data <- tips %>%
  group_by(day, gender) %>%
  summarise(Value = sum(tip), .groups = "drop") %>%
  mutate(path = paste(day, gender, sep = "-"))

# Draw sunburst chart
sunburst(sunburst_data[, c("path", "Value")])

#parallel sets
parallel_data <- tips %>%
  count(day, time)

ggplot(parallel_data,
       aes(axis1 = day, axis2 = time, y = n)) +
  geom_alluvium(aes(fill = day)) +
  geom_stratum() +
  geom_text(stat = "stratum", aes(label = after_stat(stratum)))

#Sankey diagram
links <- tips %>%
  count(day, time)
```

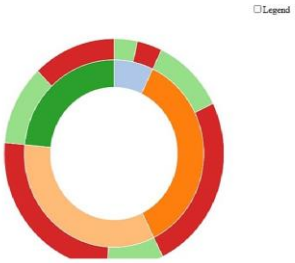
```
nodes <- data.frame(
  name = unique(c(links$day, links$time))
)

links$source <- match(links$day, nodes$name) - 1
links$target <- match(links$time, nodes$name) - 1

sankeyNetwork(
  Links = links,
  Nodes = nodes,
  Source = "source",
  Target = "target",
  Value = "n",
  NodeID = "name"
)
```


Fri
200
ISO
100 -
Sun
Thur
Dinner
Lunch
day
Fri
Sat
Sun
Thur


Sat
Sun
Fri
Thur
Dinner
Lunch


Fri
Sun
Thur
Sat
day
Fri
Sat
Sun
Thur

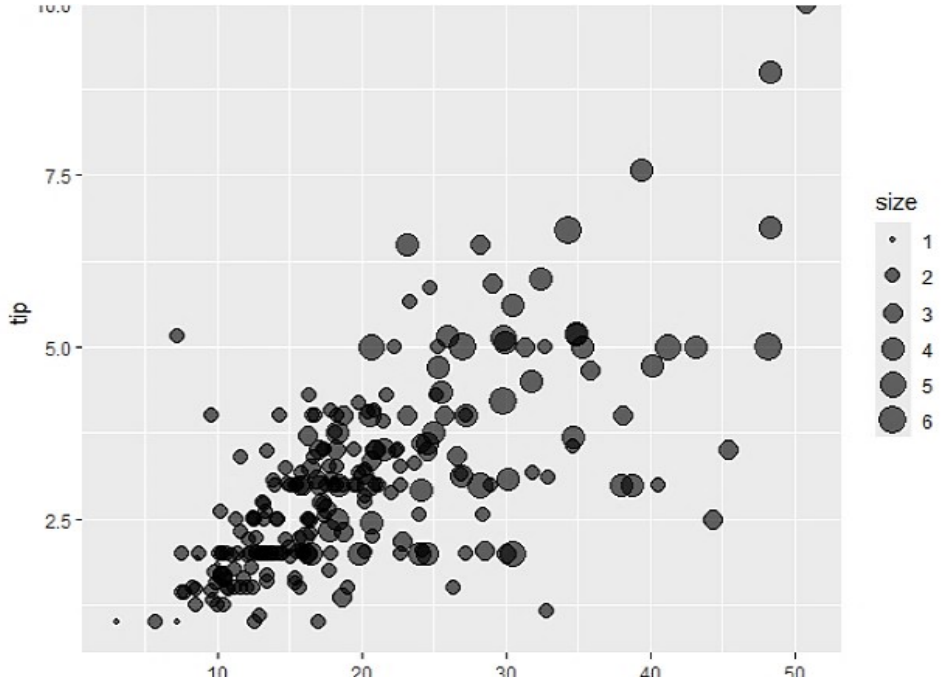

0
600
200
400
Day
Fri
Sat
Sun
Thur

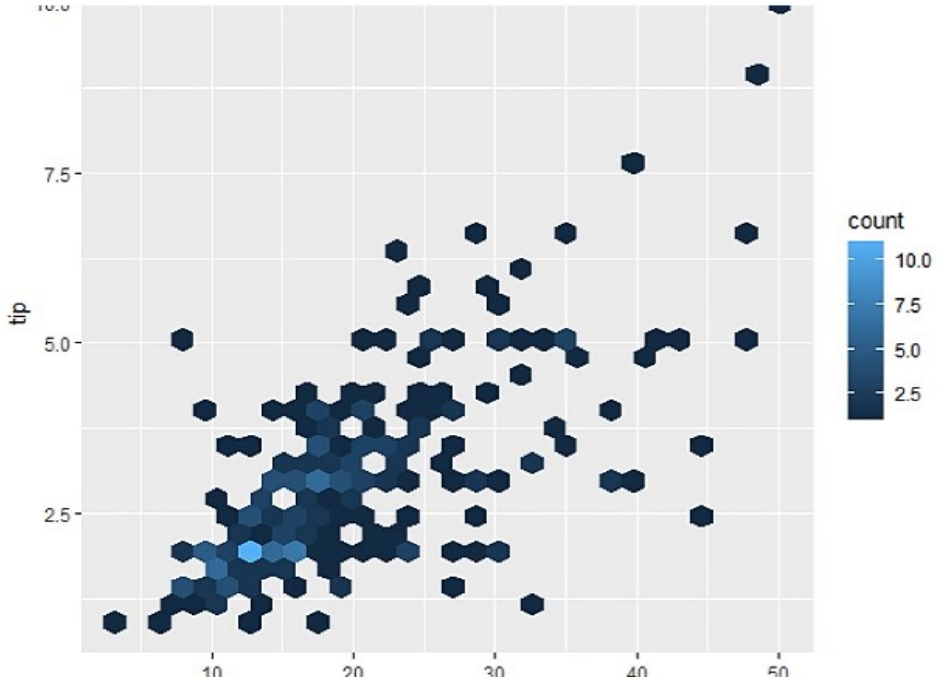

Legend

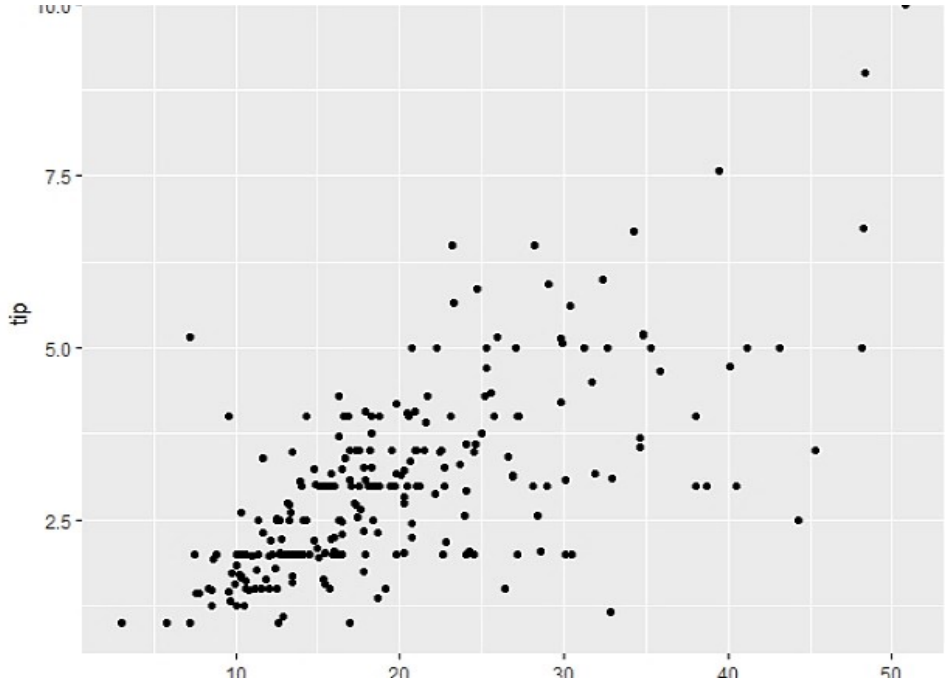
3. Visualizing Relationships and Associations

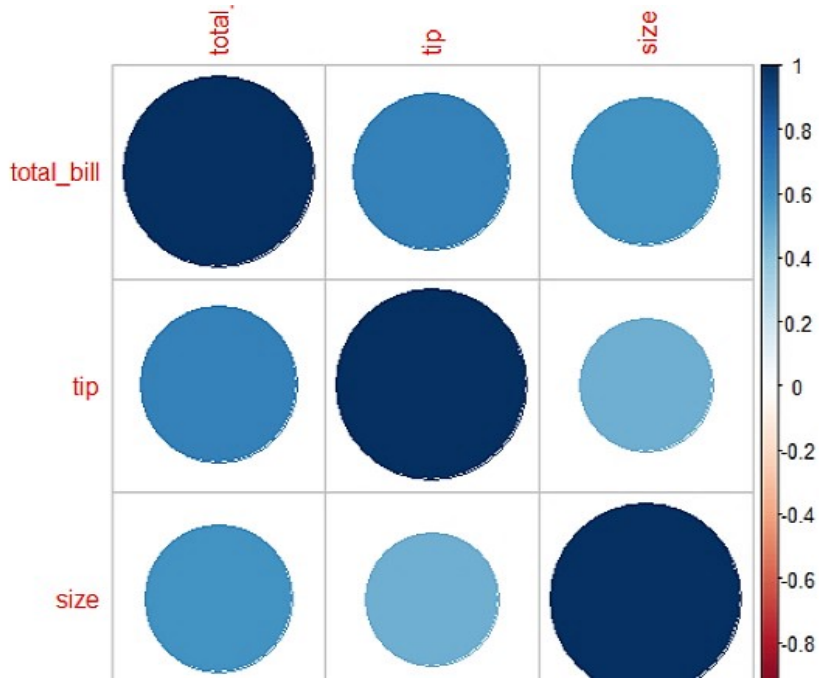
```
setwd("D:/DSA0613")
getwd()
tips <- read.csv("tips.csv")
head(tips)

library(ggplot2)
library(hexbin)
library(corrplot)
library(igraph)
library(ggfortify)
library(dplyr)

#Scatterplot
ggplot(tips, aes(x = total_bill, y = tip)) +
  geom_point()

#Bubble Chart
ggplot(tips, aes(x = total_bill, y = tip, size = size)) +
  geom_point(alpha = 0.6)

# Hexbin Plot
ggplot(tips, aes(x = total_bill, y = tip)) +
  geom_hex()

#Correlogram
numeric_data <- tips %>%
```

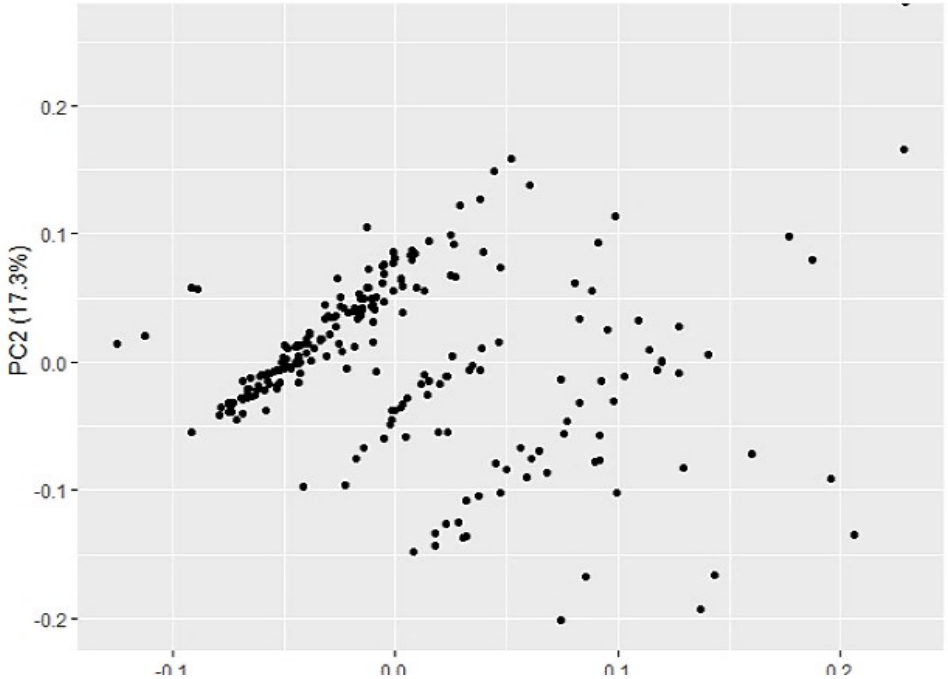
```
  select(total_bill, tip, size)

corrplot(cor(numeric_data))

#Network Graph
edges <- tips %>%
  count(day, time)

graph <- graph_from_data_frame(edges, directed = FALSE)

plot(graph)

#PCA
pca_data <- tips %>%
  select(total_bill, tip, size)

autoplot(prcomp(pca_data, scale. = TRUE))
```

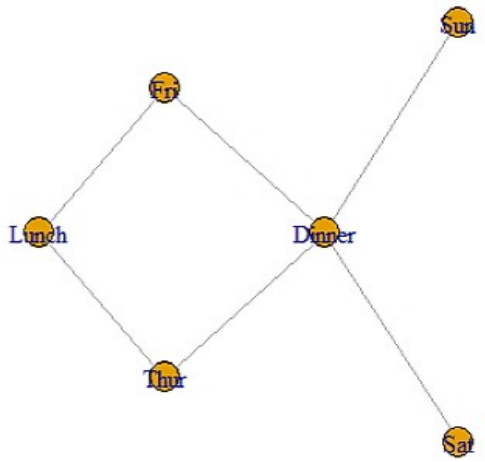

tip
size
1
2
3
4
5
6
10
20
30
40
50
2.5
5.0
7.5


tip
count
10.0
7.5
5.0
2.5
10
20
30
40
50


tip
10
20
30
40
50
2.5
5.0
7.5


total_
tip
size
total_bill
tip
size
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8

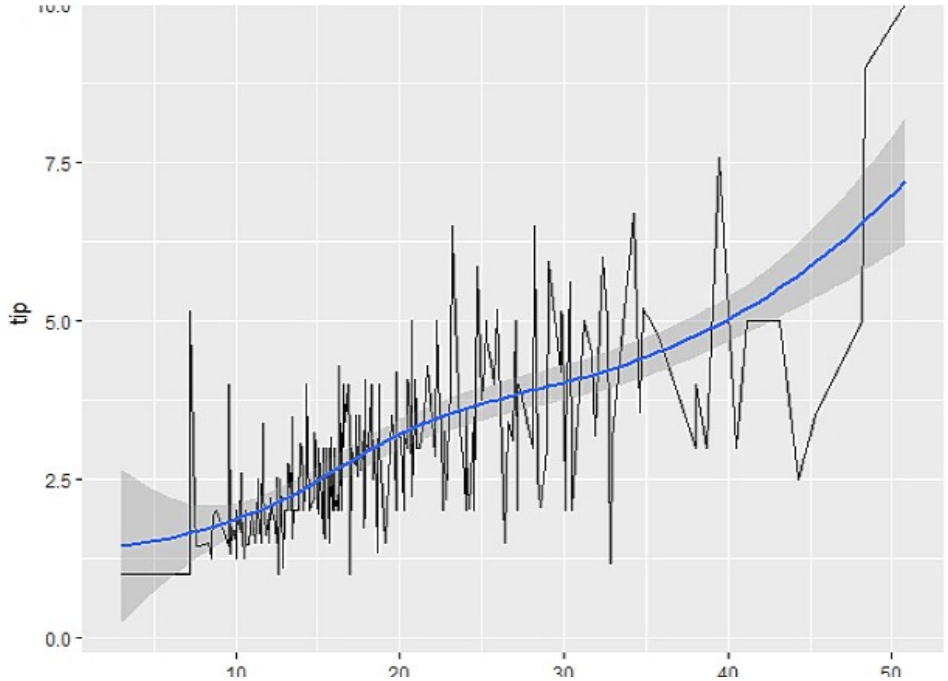

PC2 (17.3%)
0.2
0.1
0.0
-0.1
-0.2
-0.1
0.0
0.1
0.2

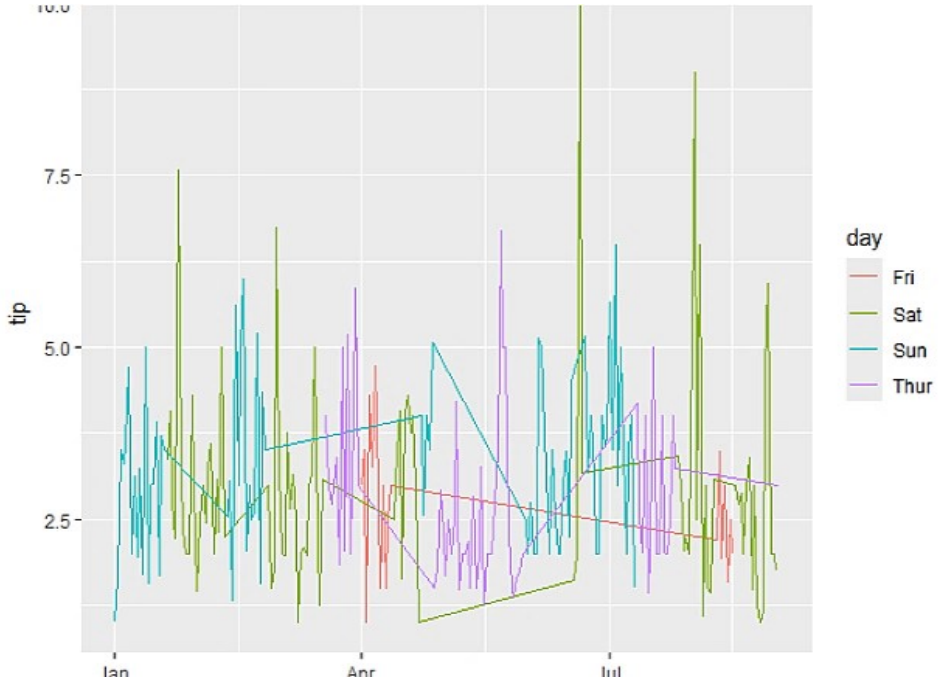

Sun
Fri
Lunch
Dinner
Thur
Sat

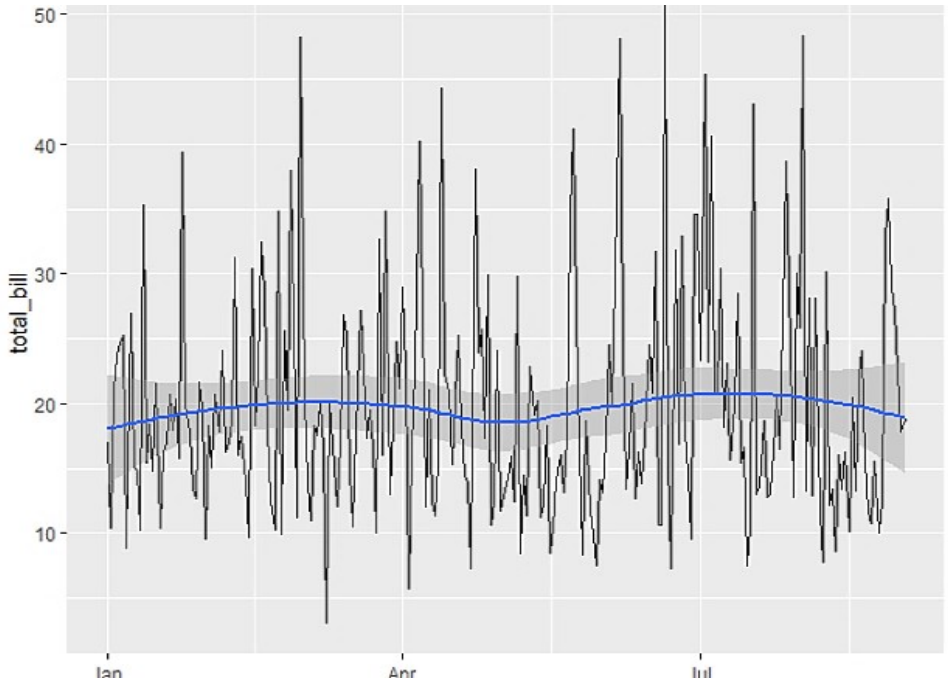
4. Visualizing Time Series / Trends

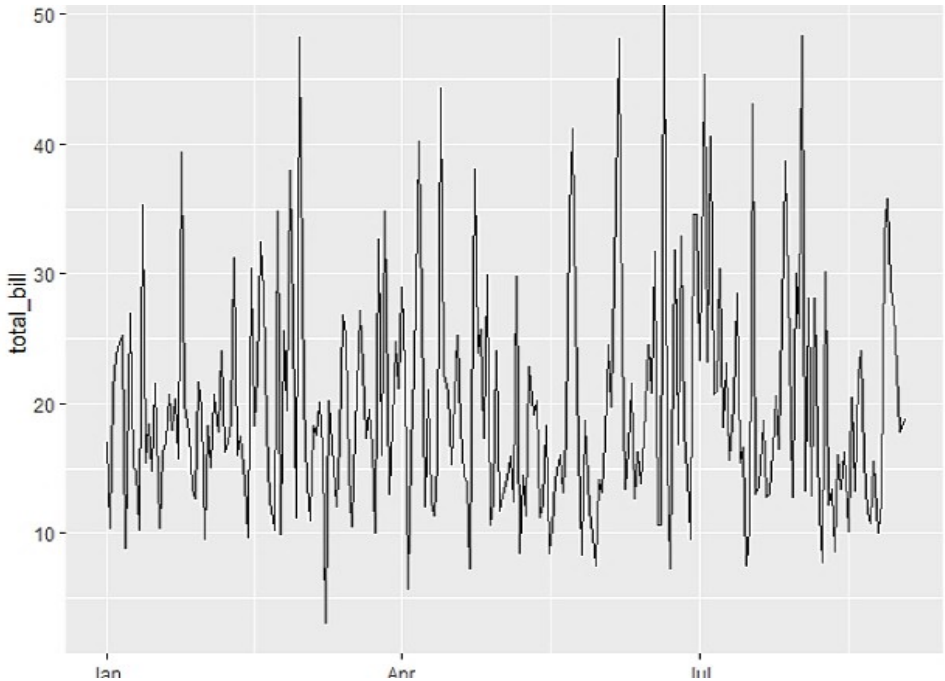
```
setwd("D:/DSA0613")
getwd()
tips <- read.csv("tips.csv")
head(tips)

library(ggplot2)
library(dplyr)
library(forecast)

#create date column
tips_ts <- tips %>%
  mutate(Date = seq.Date(from = as.Date("2023-01-01"),
                         by = "day",
                         length.out = n()))

#Line Plot
ggplot(tips_ts, aes(x = Date, y = total_bill)) +
  geom_line()

#Multiple Line Plot
ggplot(tips_ts, aes(x = Date, y = tip, color = day)) +
  geom_line()

#Dose–Response Curve
ggplot(tips, aes(x = total_bill, y = tip)) +
  geom_line() +
```

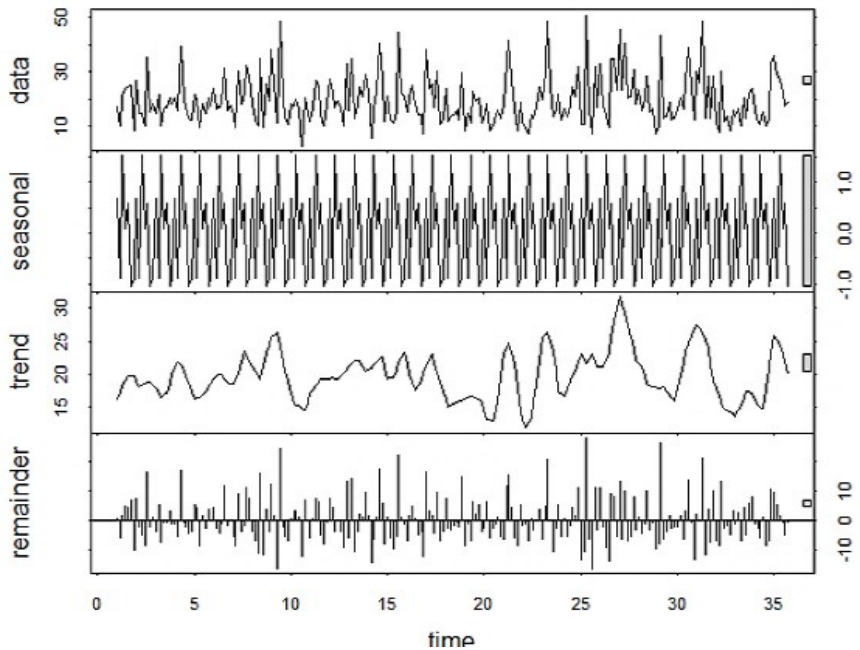
```
  geom_smooth()

#Seasonal Decomposition / STL
bill_ts <- ts(tips_ts$total_bill, frequency = 7)

stl_decomp <- stl(bill_ts, s.window = "periodic")

plot(stl_decomp)

#Smoothing / Moving Average
ggplot(tips_ts, aes(x = Date, y = total_bill)) +
  geom_line() +
  geom_smooth(method = "loess")
```

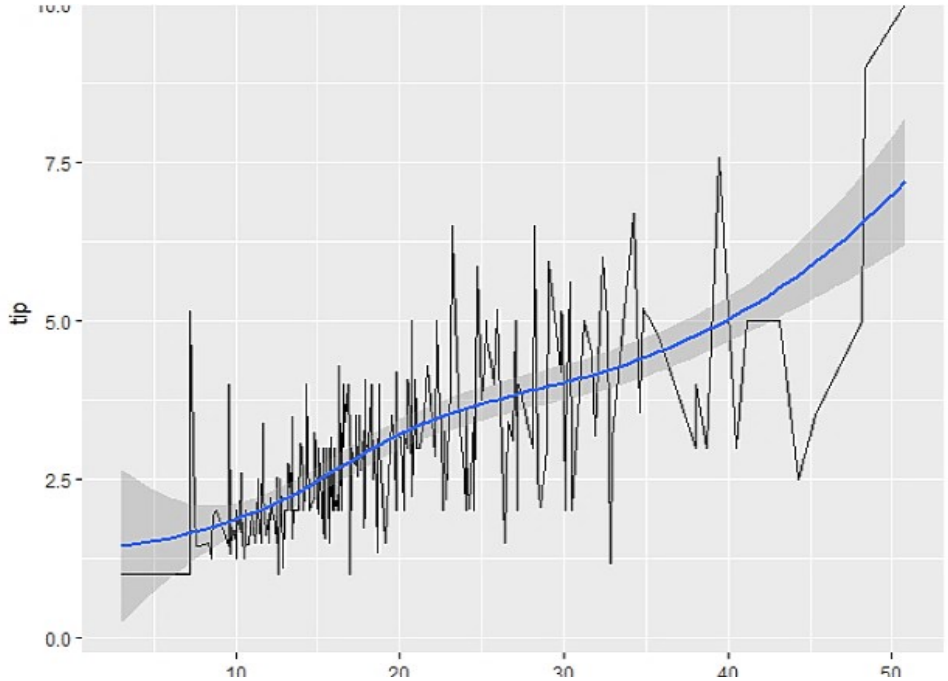

tip
7.5
5.0
2.5
0.0
10
20
30
40
50

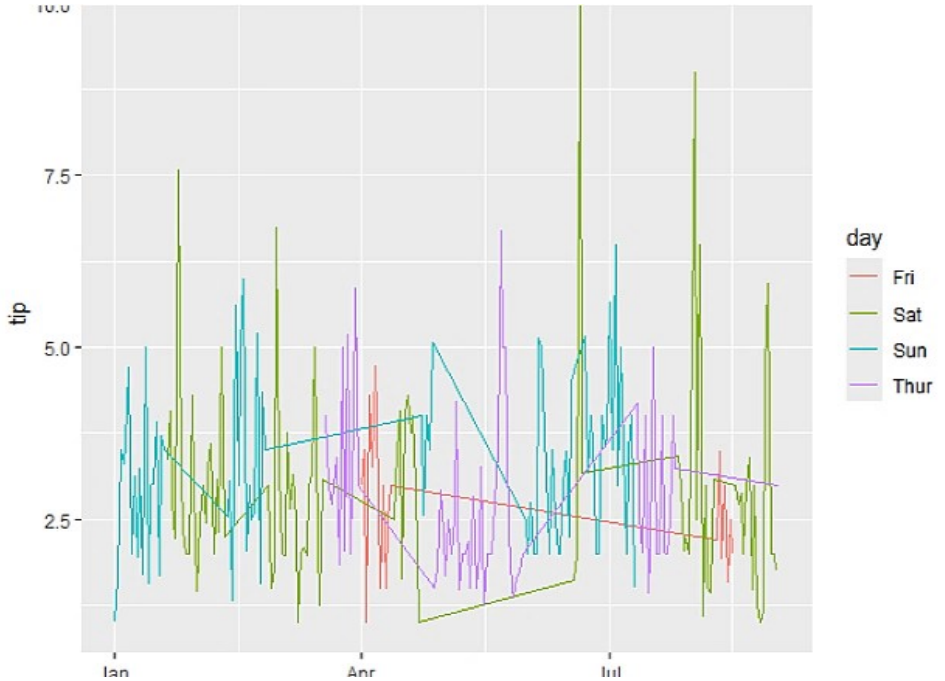

tip
7.5
5.0
2.5
day
Fri
Sat
Sun
Thur

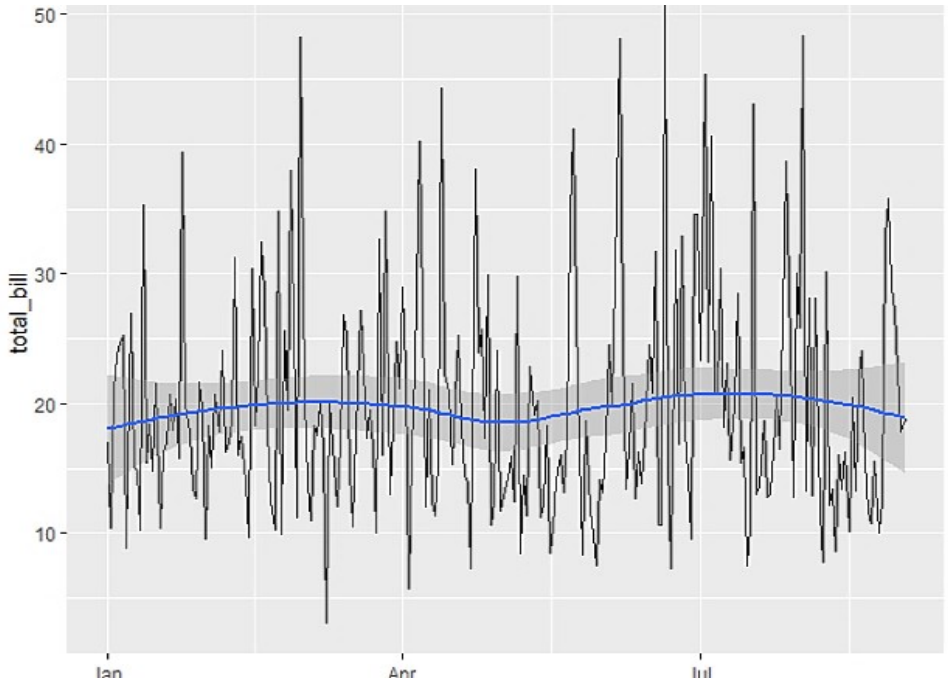

total_bill
50
40
30
20
10

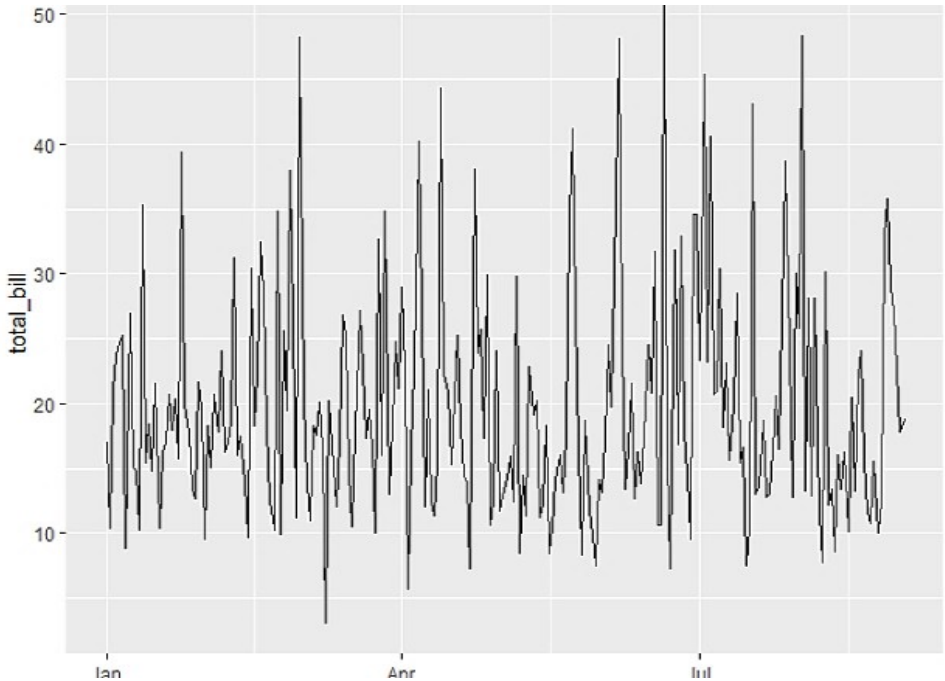

total_bill
50
40
30
20
10

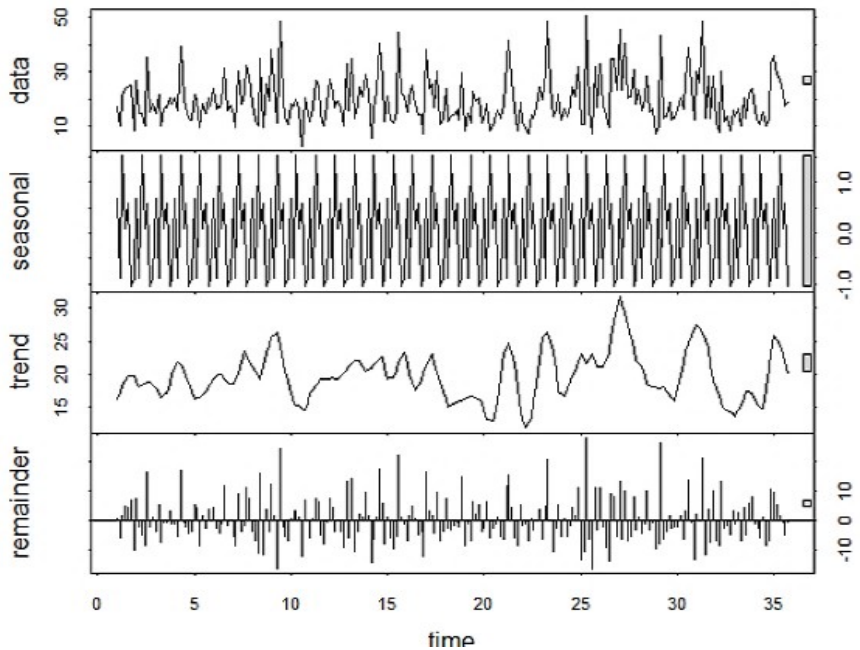

data
seasonal
trend
remainder
10
30
50
-1.0
0.0
1.0
15
20
25
30
-10
0
10
0
5
10
15
20
25
30
35
time


tip
7.5
5.0
2.5
0.0
10
20
30
40
50


tip
7.5
5.0
2.5
day
Fri
Sat
Sun
Thur


total_bill
50
40
30
20
10


total_bill
50
40
30
20
10

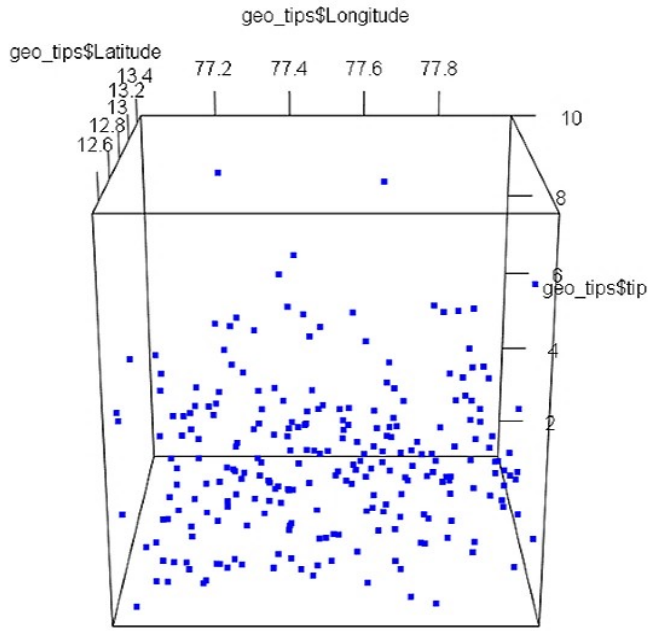

data
seasonal
trend
remainder
10
30
50
-1.0
0.0
1.0
15
20
25
30
-10
0
10
0
5
10
15
20
25
30
35
time

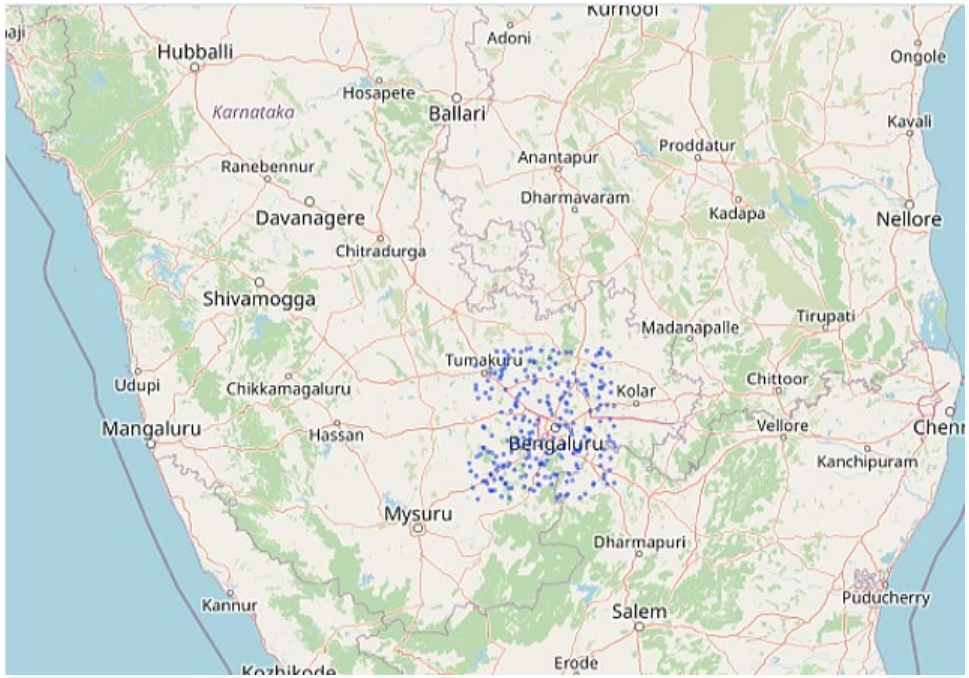
5. Visualizing Geospatial Data

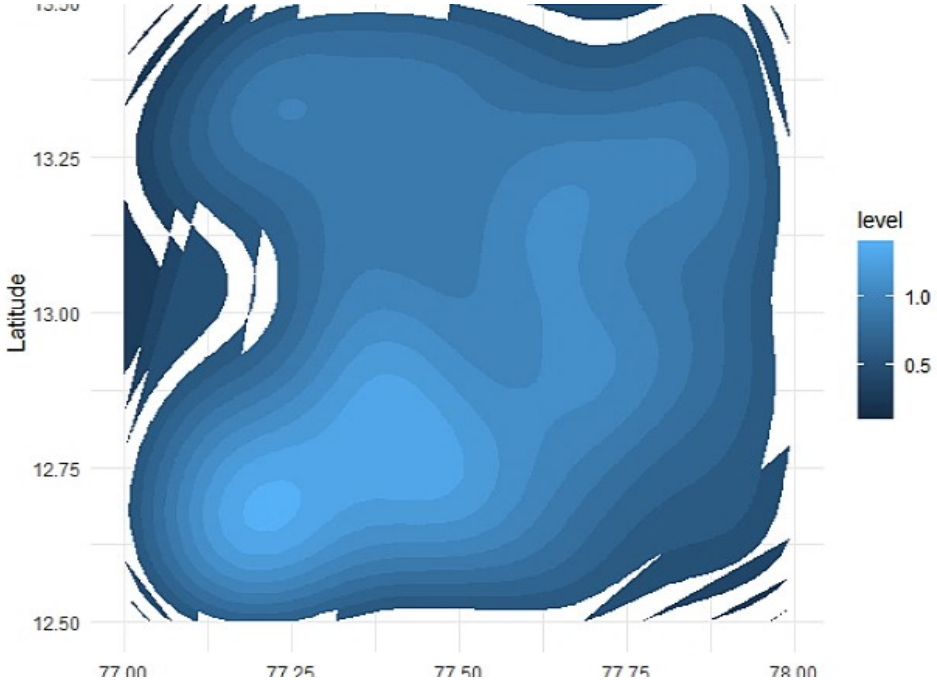
```
setwd("D:/DSA0613")
getwd()
tips <- read.csv("tips.csv")
head(tips)

library(ggplot2)
library(dplyr)
library(rgl)
library(leaflet)

#Assign Sample Geographic
Coordinates
set.seed(123)

geo_tips <- tips %>%
  mutate(
    Longitude = runif(n(), 77.0,
78.0),
    Latitude  = runif(n(), 12.5, 13.5)
  )

#Choropleth Map
library(ggplot2)
library(dplyr)
```

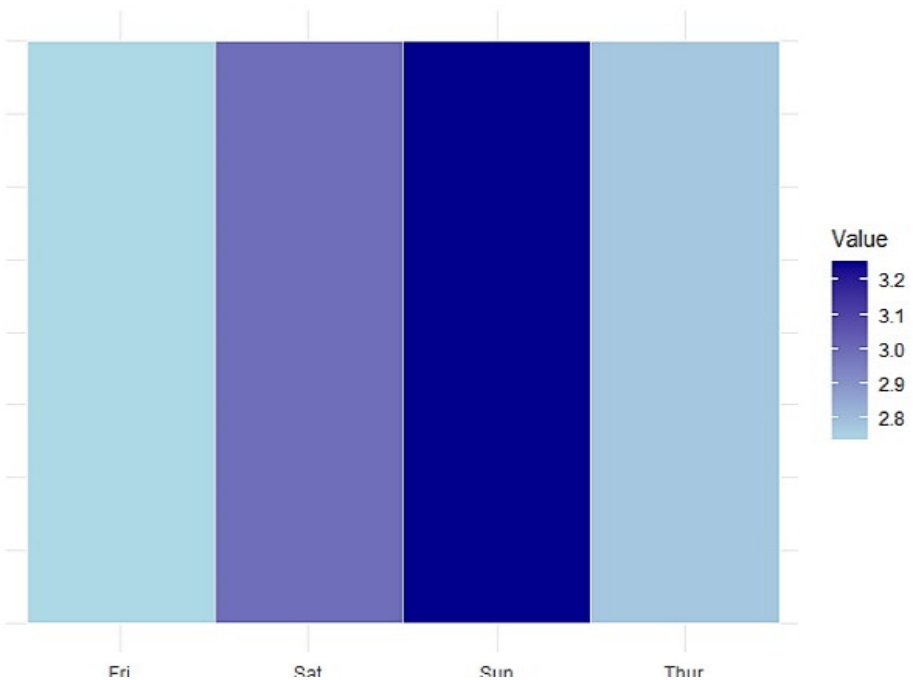
```
choropleth_data <- tips %>%
  group_by(day) %>%
  summarise(Value = mean(tip))

ggplot(choropleth_data, aes(x = day, y = 1, fill = Value)) +
  geom_tile(color = "white") +
  scale_fill_gradient(low = "lightblue", high = "darkblue") +
  labs(
    title = "Choropleth Representation of Average Tip by Day",
    x = "Day",
    y = ""
  ) +
  theme_minimal() +
  theme(
    axis.text.y = element_blank(),
    axis.ticks.y = element_blank()
  )

#Spatial Heatmap
ggplot(geo_tips, aes(x = Longitude, y = Latitude)) +
  stat_density2d(aes(fill = ..level..), geom = "polygon") +
```

```
  theme_minimal()

#3D Geospatial Plot
plot3d(
  x = geo_tips$Longitude,
  y = geo_tips$Latitude,
  z = geo_tips$tip,
  col = "blue",
  size = 5
)

#Interactive Map
leaflet(geo_tips) %>%
  addTiles() %>%
  addCircles(
    lng = ~Longitude,
    lat = ~Latitude,
    radius = ~tip * 50,
    weight = 1
  )
```


geo_tips$Longitude
geo_tips$Latitude
13.4
13.2
13
12.8
12.6
77.2
77.4
77.6
77.8
10
8
6
4
2
geo_tips$tip


Hubballi
Adoni
Ongole
Karnataka
Hosapete
Ballari
Kavali
Ranebennur
Anantapur
Proddatur
Dharmavaram
Davanagere
Kadapa
Nellore
Chitradurga
Shivamogga
Madanapalle
Tirupati
Tumakuru
Udupi
Chikkamagaluru
Kolar
Chittoor
Mangaluru
Hassan
Vellore
Bengaluru
Kanchipuram
Mysuru
Dharmapuri
Kannur
Salem
Puducherry
Erode

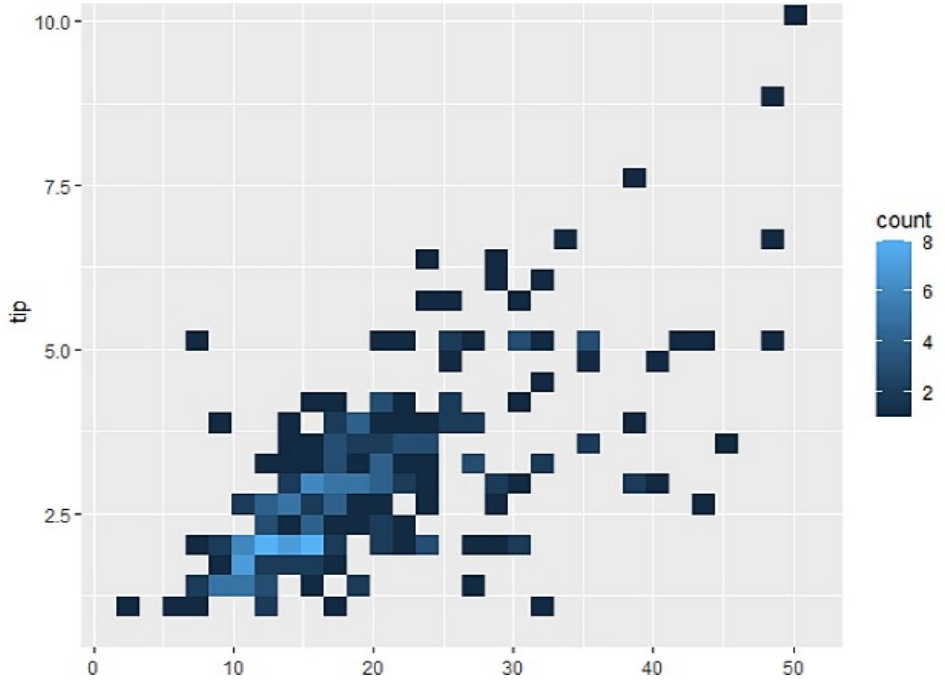

Latitude
13.25
13.00
12.75
12.50
77.00
77.25
77.50
77.75
78.00
level
1.0
0.5

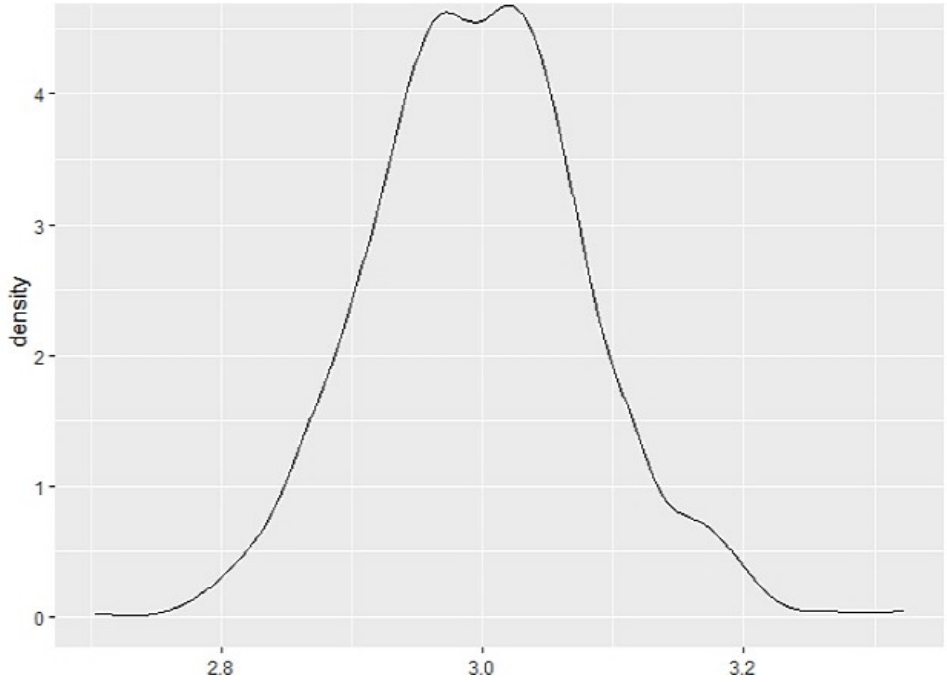

Value
3.2
3.1
3.0
2.9
2.8
Fri
Sat
Sun
Thur

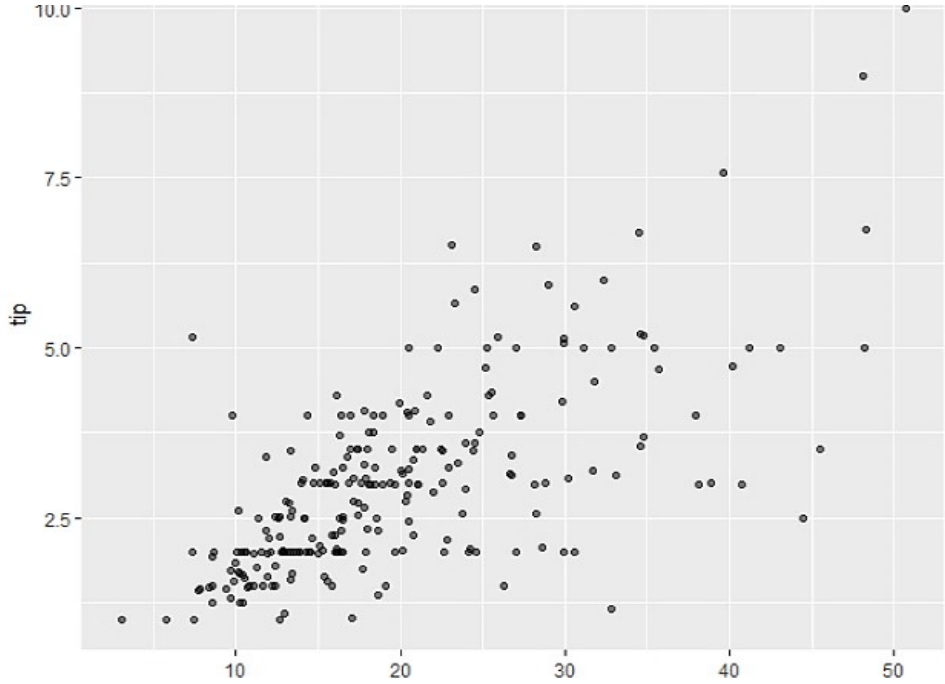
6. Visualizing Uncertainty

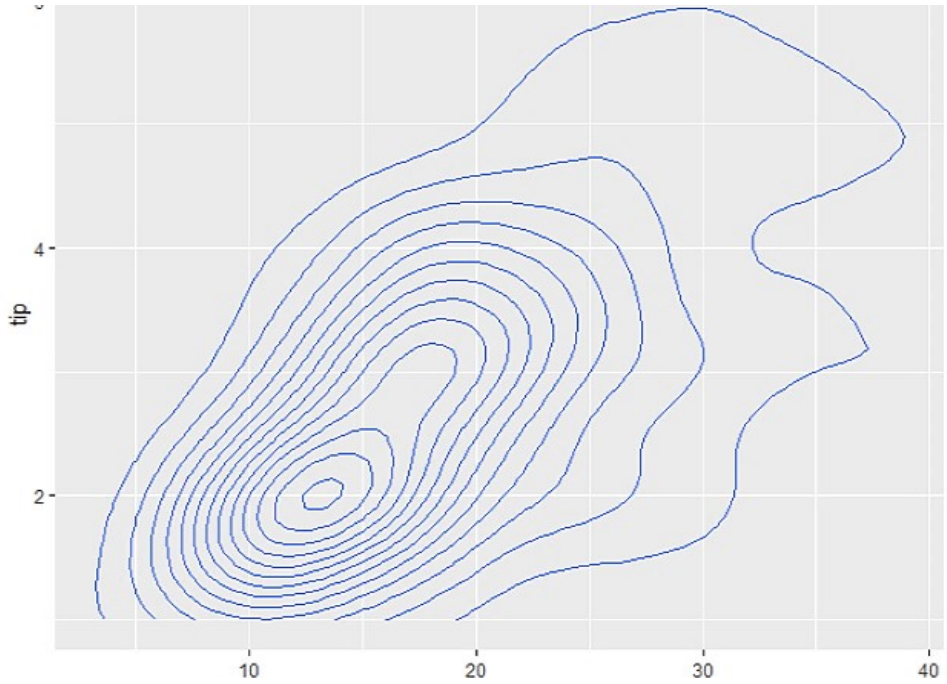
```
setwd("D:/DSA0613")
getwd()
tips <- read.csv("tips.csv")
head(tips)

library(ggplot2)
library(dplyr)

#Confidence Interval
ci_data <- tips %>%
  group_by(day) %>%
  summarise(
    mean_tip = mean(tip),
    sd_tip = sd(tip)
  )

ggplot(ci_data, aes(x = day, y = mean_tip)) +
  geom_point() +
  geom_errorbar(
    aes(ymin = mean_tip - sd_tip, ymax = mean_tip + sd_tip),
    width = 0.2
  )

#Bootstrapping / Hypothetical Outcomes
set.seed(123)
```

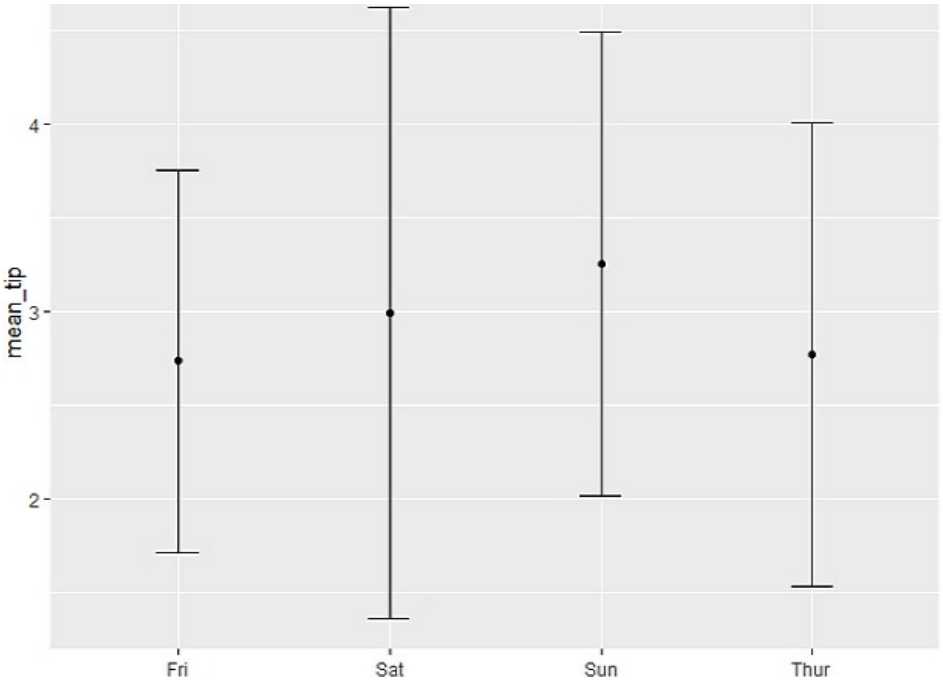
```
df_boot <- data.frame(
  Value = replicate(1000, mean(sample(tips$tip, replace = TRUE)))
)

ggplot(df_boot, aes(x = Value)) +
  geom_density(alpha = 0.3)

#Partial Transparency & Jitter
ggplot(tips, aes(x = total_bill, y = tip)) +
  geom_point(alpha = 0.5,
             position = position_jitter(width = 0.2))

#2D Histogram / Contour Lines
#(a) 2D Histogram
ggplot(tips, aes(x = total_bill, y = tip)) +
  geom_bin2d()

#(b) 2D Density Contours
ggplot(tips, aes(x = total_bill, y = tip)) +
  geom_density_2d()
```


tip
count
8
6
4
2
10.0
7.5
5.0
2.5
0
10
20
30
40
50


density
4
3
2
1
0
2.8
3.0
3.2


tip
10.0
7.5
5.0
2.5
10
20
30
40
50


tip
4
2
10
20
30
40


mean_tip
4
3
2
Fri
Sat
Sun
Thur