Name: Cheredla Jasvina **Reg. No.:** 192324026

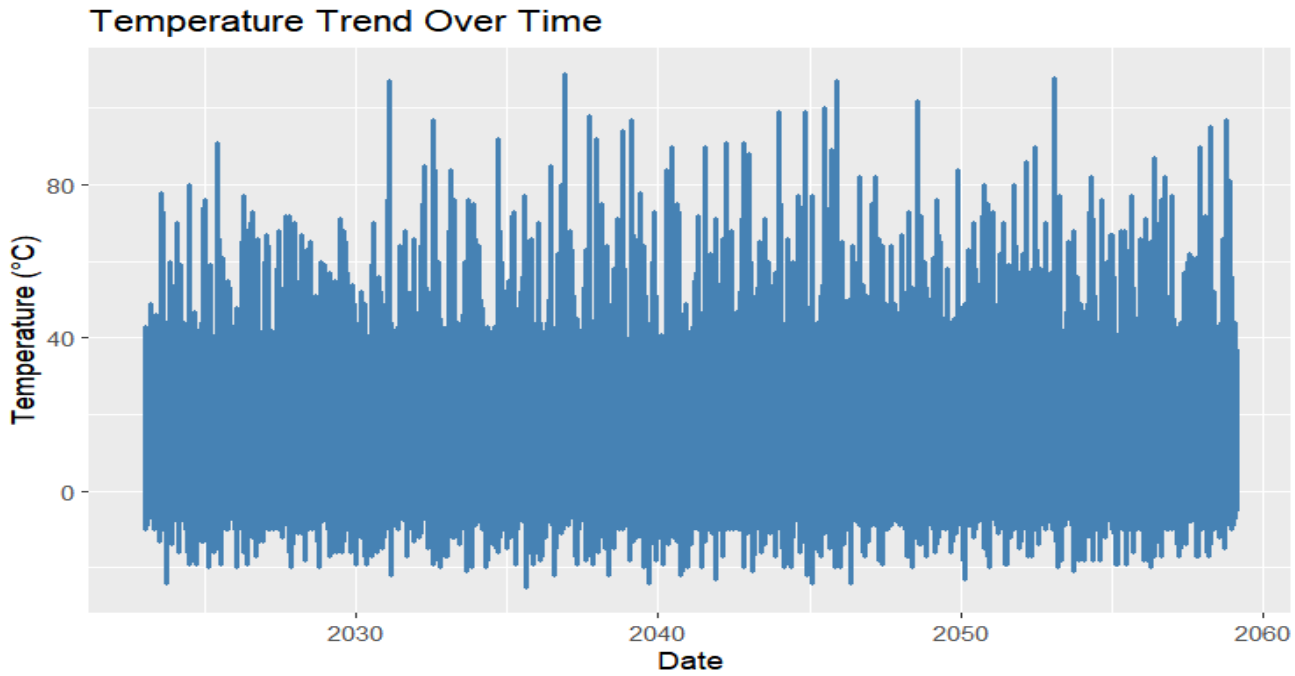
Course Code: DSA0613
Course Name: Data Handling and Visualization for Data Analytics

4. Visualizing Time Series and Trends in Weather Data Using R

1. Line Plot – Trend Over Time

```
ggplot(df, aes(x = Time, y = Temperature)) +
  geom_line(color = "steelblue") +
  labs(title = "Temperature Trend Over Time") +
  theme_minimal()
```

OUTPUT:

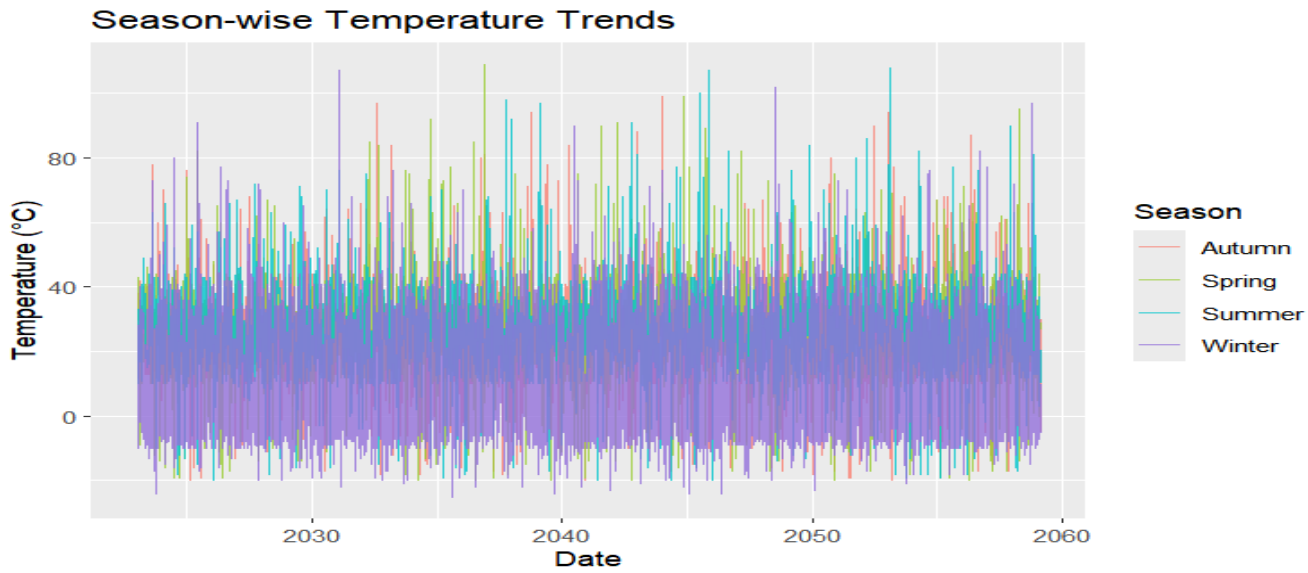


2. Season-wise Temperature Trend

```
ggplot(df, aes(x = Time)) +
  geom_line(aes(y = Temperature, color = Season)) +
  scale_color_manual(values = c(
```

```
  "Winter" = "purple",
  "Spring" = "lightgreen",
  "Summer" = "turquoise3",
  "Autumn" = "orange"
)) +
labs(
  title = "Season-wise Temperature Trend Over Time",
  x = "Time",
  y = "Temperature",
  color = "Season"
) +
theme_minimal()
```

OUTPUT:

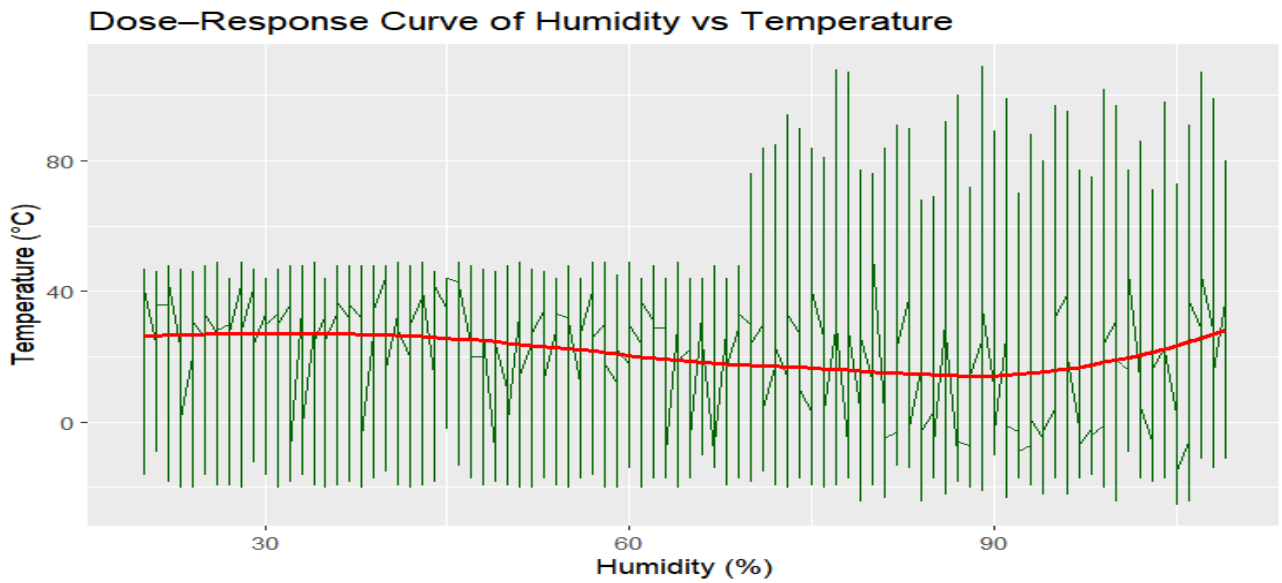


3. Dose–Response Curve

```
ggplot(df, aes(x = Humidity, y = Temperature)) +
  geom_line(color = "darkseagreen", alpha = 0.8) +
  geom_smooth(
```

```
  method = "loess",
  se = FALSE,
  color = "red",
  linewidth = 1
) +
labs(
  title = "Dose–Response Curve of Humidity vs Temperature",
  x = "Humidity (%)",
  y = "Temperature (°C)"
) +
theme_minimal()
```

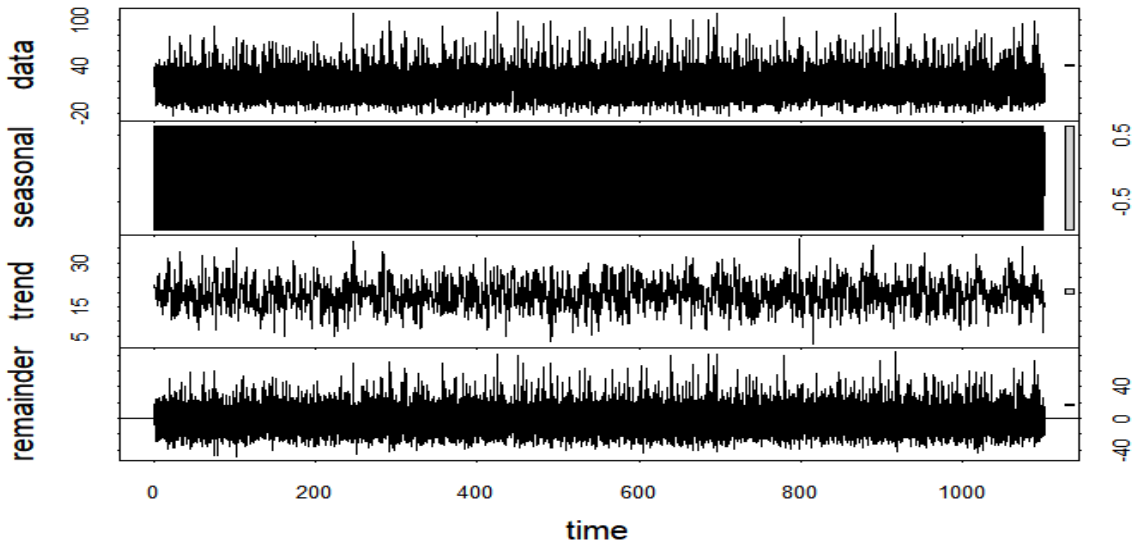
OUTPUT:



4.Seasonal Decomposition / STL

```
temp_ts <- ts(df$Temperature, frequency = 12)
stl_result <- stl(temp_ts, s.window = "periodic")
plot(stl_result)
```

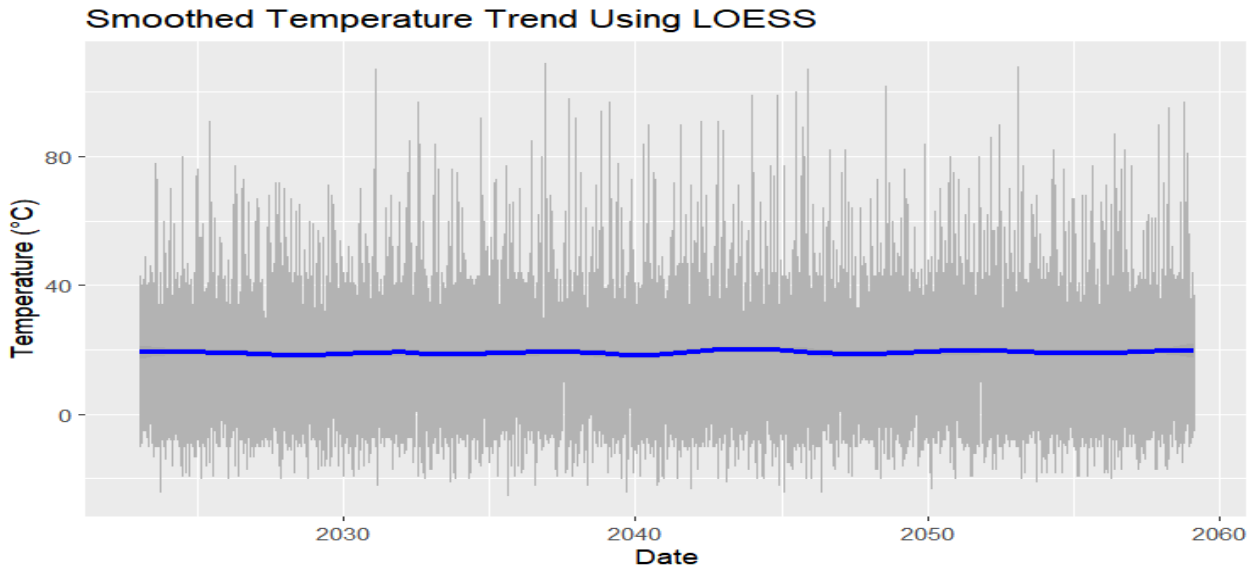
OUTPUT:



5. Smoothing / Moving Average

```
ggplot(df, aes(x = Time, y = Temperature)) +
  geom_line(color = "gray") +
  geom_smooth(method = "loess", color = "darkblue") +
  labs(title = "Temperature Trend with Smoothing") +
  theme_minimal()
```

OUTPUT:



5.Visualizing Geospatial Weather Data Using R

```
setwd("C:/Users/Jasvi/Documents/DSA0613")
df <- read.csv("weather_classification_data.csv")
install.packages("ggplot2")
install.packages("maps")
install.packages("dplyr")
install.packages("rgl")
install.packages("leaflet")
install.packages("rlang")
install.packages("vctrs")
install.packages("cli")
library(ggplot2)
library(maps)
library(dplyr)
library(rgl)
library(leaflet)
library(rlang)
library(vctrs)
library(cli)
```

 # Create Dummy Geospatial Data

```
set.seed(123)
df$Latitude  <- runif(nrow(df), 8, 37)
df$Longitude <- runif(nrow(df), 68, 97)
```

1. Choropleth Map (WORKING)

```
world_map <- map_data("world")
india_map <- subset(world_map, region == "India")
ggplot(india_map, aes(x = long, y = lat, group = group)) +
```

```
  geom_polygon(fill = "lightblue", color = "white") +
  coord_fixed(1.3) +
  labs(title = "Choropleth Map of India") +
  theme_void()
```

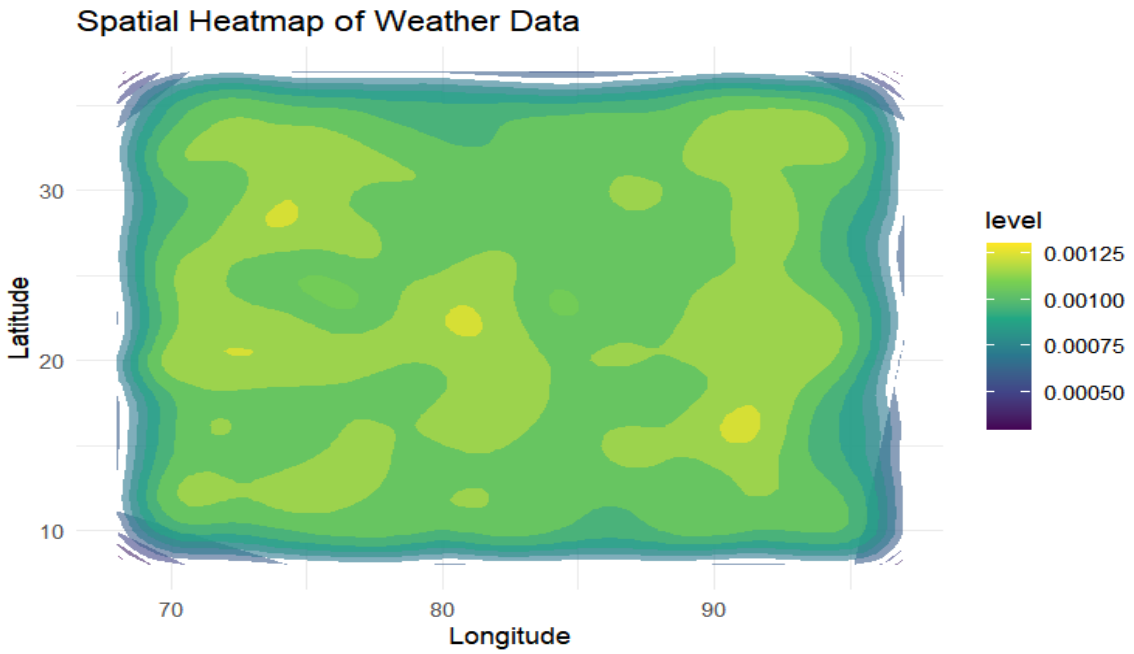
OUTPUT:

2. Spatial Heatmap

```
ggplot(df, aes(x = Longitude, y = Latitude)) +
  stat_density2d(
    aes(fill = after_stat(level)),
    geom = "polygon",
    alpha = 0.6
  ) +
  scale_fill_viridis_c() +
  labs(
    title = "Spatial Heatmap of Weather Data",
    x = "Longitude",
```

```
  y = "Latitude"
) +
theme_minimal()
```

OUTPUT:

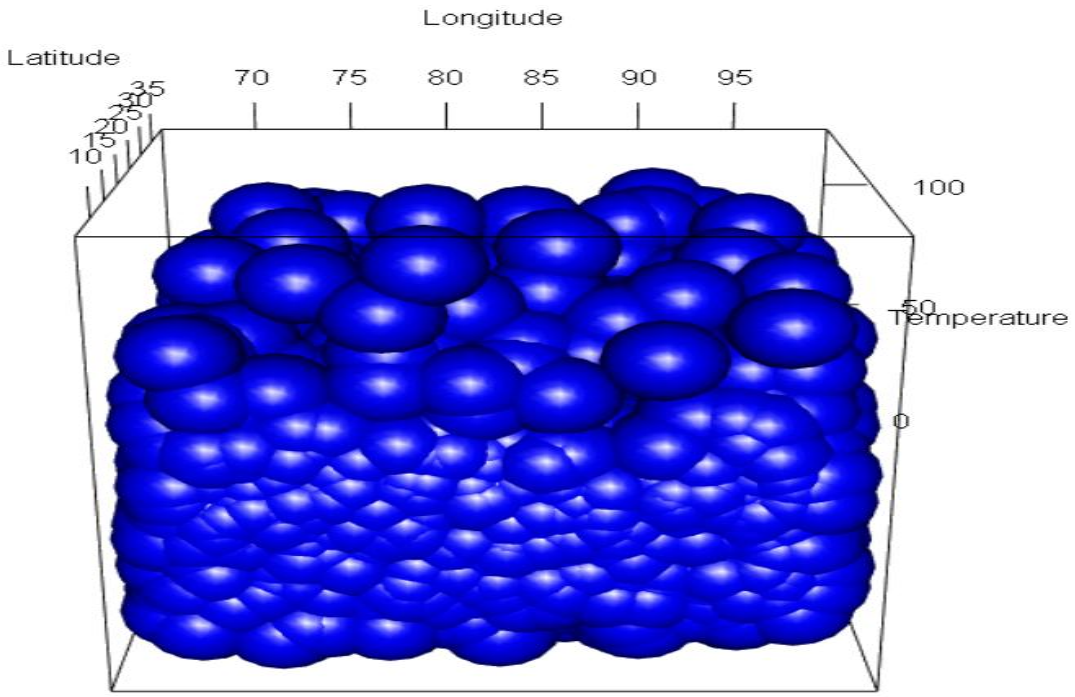


3. 3D Geospatial Plot

```
open3d()
plot3d(
  x = df$Longitude,
  y = df$Latitude,
  z = df$Temperature,
  col = "blue",
  size = 6,
  type = "s",
  xlab = "Longitude",
  ylab = "Latitude",
  zlab = "Temperature"
```

)

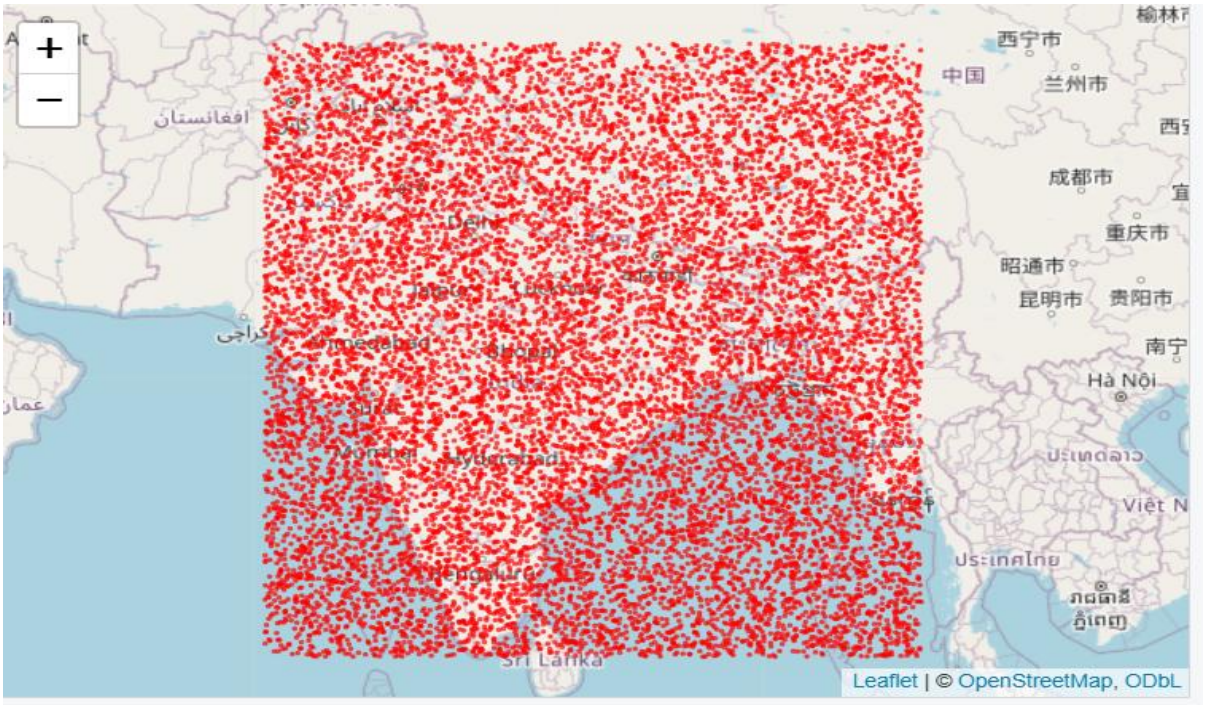
OUTPUT:



4. Interactive Map

```
leaflet(df) %>%
  addTiles() %>%
  addCircles(
    lng = ~Longitude,
    lat = ~Latitude,
    radius = 4,
    weight = 1,
    color = "red",
    fillOpacity = 0.6,
    popup = ~paste("Temperature:", Temperature)
  )
```

OUTPUT:



6.Visualizing Uncertainty in Weather Data Using R

```
library(dplyr)

install.packages("ggplot2")
```

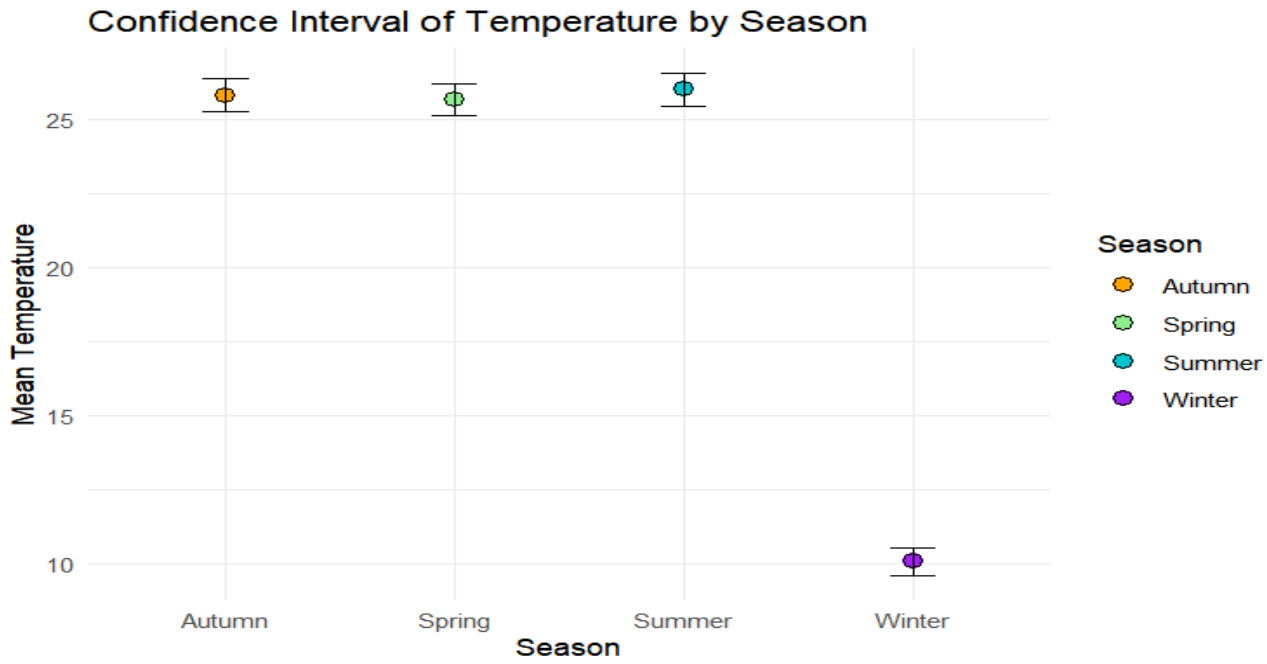
```
install.packages("dplyr")
setwd("C:/Users/Jasvi/Documents/DSA0613")
df <- read.csv("weather_classification_data.csv")
# Define SAME Season Colors
season_colors <- c(
  "Winter" = "purple",
  "Spring" = "lightgreen",
  "Summer" = "turquoise3",
  "Autumn" = "orange"
)
```

1. Confidence Interval

```
ci_df <- df %>%
  group_by(Season) %>%
  summarise(
    Mean = mean(Temperature),
    SD = sd(Temperature),
    N = n(),
    Lower = Mean - 1.96 * SD / sqrt(N),
    Upper = Mean + 1.96 * SD / sqrt(N)
  )
ggplot(ci_df, aes(x = Season, y = Mean, fill = Season)) +
  geom_point(size = 3, shape = 21) +
  geom_errorbar(aes(ymin = Lower, ymax = Upper), width = 0.2) +
  scale_fill_manual(values = season_colors) +
  labs(
    title = "Confidence Interval of Temperature by Season",
    y = "Mean Temperature"
```

```
  ) +
  theme_minimal()
```

OUTPUT:

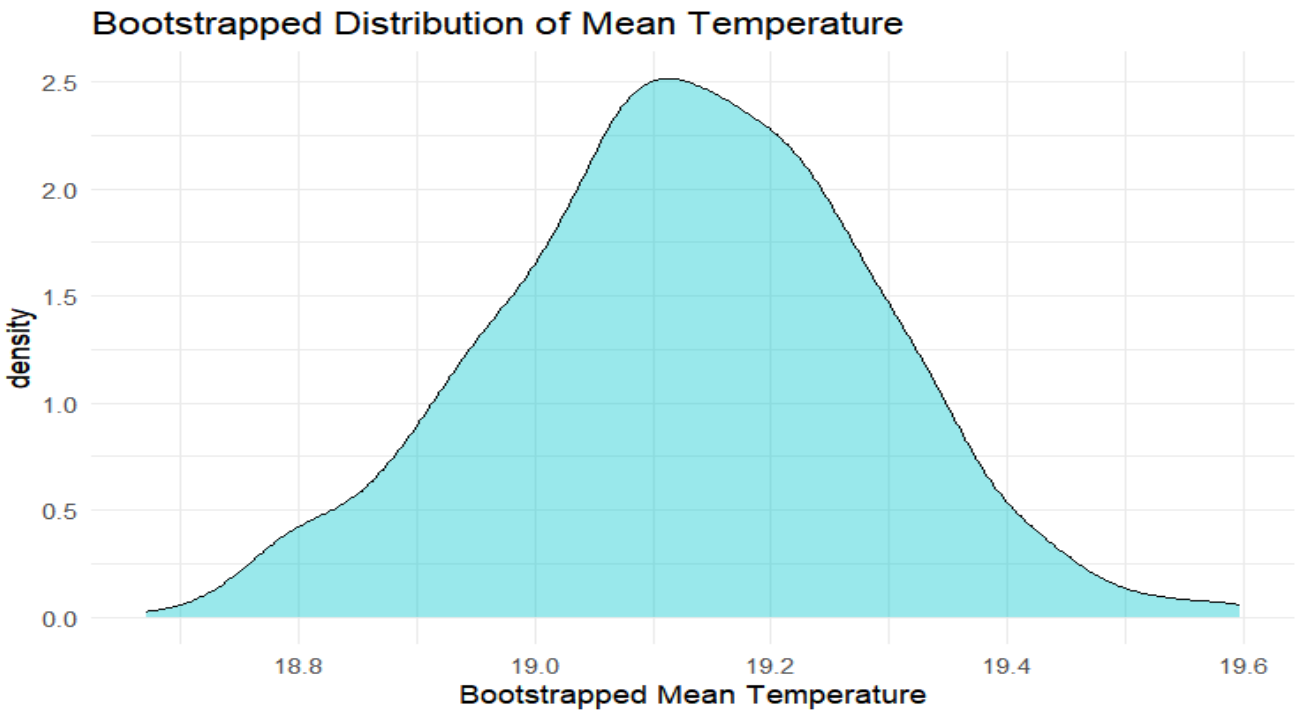


2. Bootstrapping

```
set.seed(123)
boot_mean <- replicate(
  1000,
  mean(sample(df$Temperature, replace = TRUE))
)
boot_df <- data.frame(Value = boot_mean)
ggplot(boot_df, aes(x = Value)) +
  geom_density(fill = "turquoise3", alpha = 0.4) +
  labs(
    title = "Bootstrapped Distribution of Mean Temperature",
    x = "Bootstrapped Mean Temperature"
  ) +
```

theme_minimal()

OUTPUT:

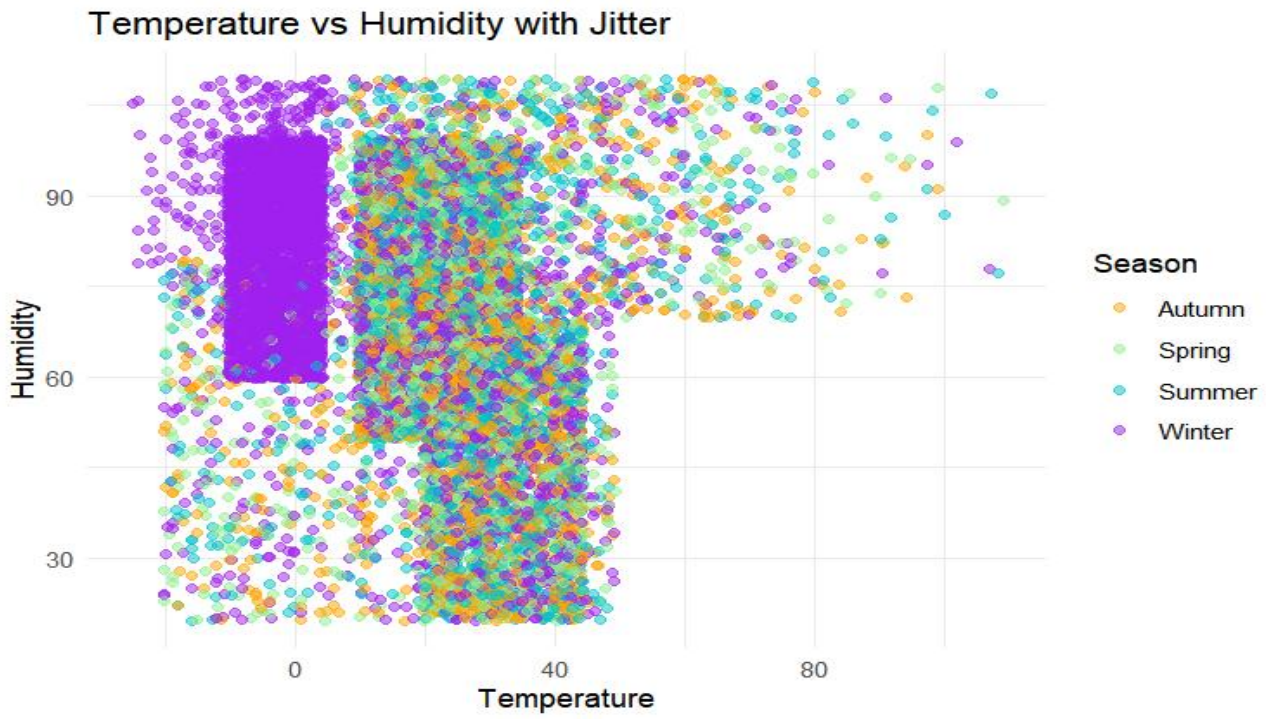


3. Transparency & Jitter

```
ggplot(df, aes(x = Temperature, y = Humidity, color = Season)) +
  geom_point(
    alpha = 0.5,
    position = position_jitter(width = 0.3, height = 0.3)
  ) +
  scale_color_manual(values = season_colors) +
  labs(
    title = "Temperature vs Humidity with Jitter",
    x = "Temperature",
    y = "Humidity"
  ) +
```

theme_minimal()

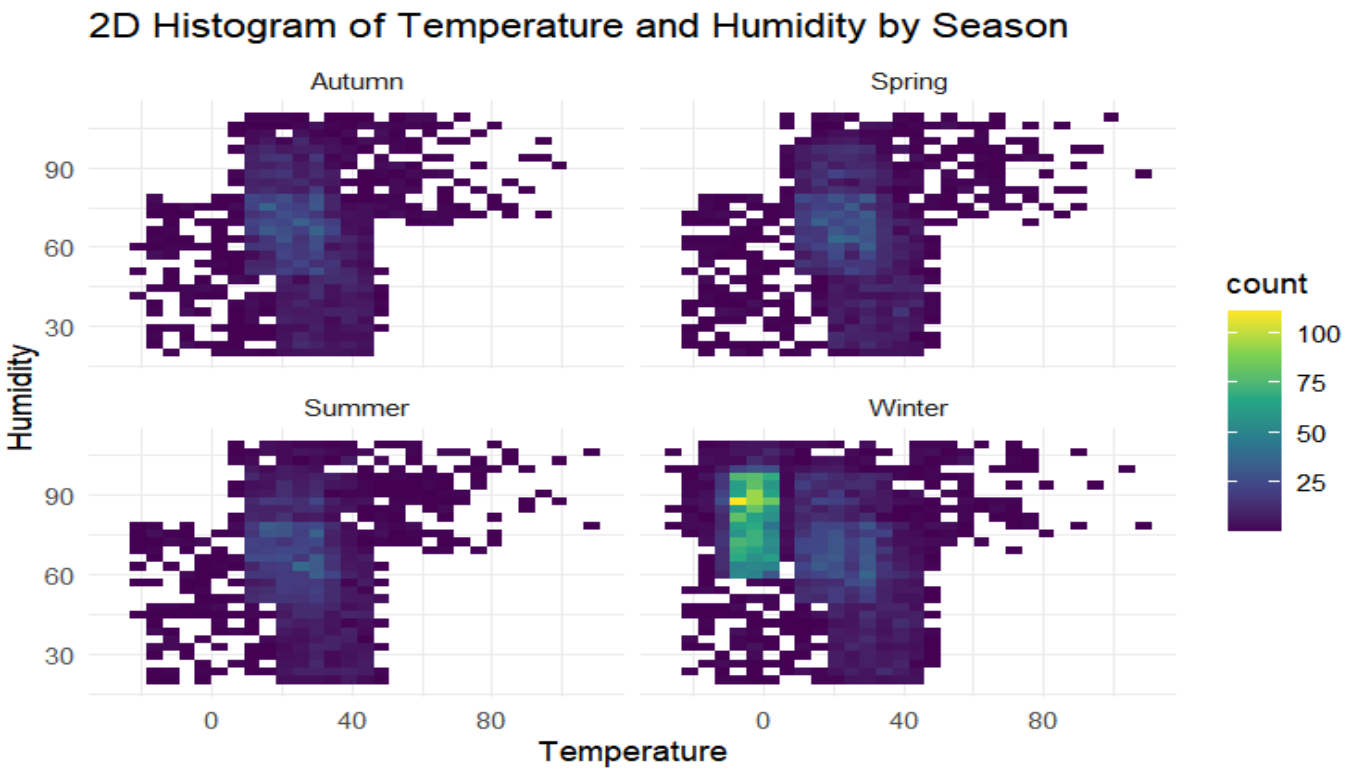
OUTPUT:



4. 2D Histogram (Faceted by Season)

```
ggplot(df, aes(x = Temperature, y = Humidity)) +
  geom_bin2d(bins = 30) +
  facet_wrap(~Season) +
  scale_fill_viridis_c() +
  labs(
    title = "2D Histogram of Temperature and Humidity by Season",
    x = "Temperature",
    y = "Humidity"
  ) +
  theme_minimal()
```

OUTPUT:


2D Histogram of Temperature and Humidity by Season
Autumn
Spring
Summer
Winter
Humidity
90
60
30
0
40
80
Temperature
count
100
75
50
25