Hourly Weather Data Analysis in Newark, New Jersey

Akhilesh Nimse, Fateenah Farid, Jake Cohen

Introduction

Climate change is increasingly recognized as one of the defining challenges of the 21st century, affecting communities on local, national, and global scales. In this report, we analyze key climate patterns in Newark, New Jersey, in 2019 and 2023, focusing on changes in temperature, humidity, and precipitation. This analysis aims to determine whether Newark is experiencing noticeable shifts that align with broader trends in climate change.

Research Questions

Our investigation is guided by three research questions:

- **Have daily temperatures been increasing in the last 4 years?**
  Rising temperatures are a hallmark of climate change. By examining daily temperature trends, we aim to identify if Newark has experienced a steady increase, periodic anomalies, or extreme temperature events over the analyzed period.

- **How does humidity fluctuate with the seasons, and has it been moving in a pattern in the last 4 years?**
  Humidity levels are closely tied to temperature and seasonal weather patterns. Understanding how humidity fluctuates across seasons, and whether these fluctuations show systematic changes over the years, provides insight into the broader impacts of a warming climate.

- **How have precipitation levels changed in the last 4 years?**
  Changes in precipitation patterns can lead to severe consequences, such as flooding or droughts. By assessing variations in Newark's precipitation levels, we seek to uncover potential shifts in the frequency, intensity, or timing of rainfall.

Dataset Description

Methodology

Data Exploration

Results

References

Code Appendix

```
# 1. Load Packages
library(tidyverse)
library(rvest)
library(knitr)

# 2a. Get newark2019 data
path1 <- "/Users/fateenahs./Desktop/STAT184/newark_hourly_weather_2019.csv"
newark2019 <- read.csv(path1)

# 2b. Get newark2023 data
path2 <- "/Users/fateenahs./Desktop/STAT184/hourly_weather_newark_2023.csv"
newark2023 <- read.csv(path2)

# 3a. Normalizing Columns for Newark in 2019
newark2019$Time <- as.POSIXct(newark2019$Time, format = "%Y-%m-%d %H:%M") # Set format for T:
newark2019$Temperature <- as.integer(gsub("F", "", newark2019$Temperature)) # Remove F from '
newark2019$Humidity <- as.integer(gsub("%", "", newark2019$Humidity))
newark2019$Precipitation <- as.numeric(gsub("in", "", newark2019$Precipitation))

# 3b. Normalizing Columns for Newark in 2023
newark2023$Time <- as.POSIXct(newark2023$Time, format = "%Y-%m-%d %H:%M") # Set format for T:
newark2023$Temperature <- as.integer(gsub("F", "", newark2023$Temperature)) # Remove F from '
newark2023$Humidity <- as.integer(gsub("%", "", newark2023$Humidity))
newark2023$Precipitation <- as.numeric(gsub("in", "", newark2023$Precipitation))

# 4a. Calculate Monthly Average Temperature in Newark for year 2019
monthlyAvg2019 <- newark2019 %>%
  drop_na() %>%
  group_by(month = format(Time, "%m")) %>%
```

```r
  summarize(avg_temperature = mean(Temperature, na.rm = TRUE))

# 4b. Calculate Monthly Average Temperature in Newark for year 2023
monthlyAvg2023 <- newark2023 %>%
  drop_na() %>%
  group_by(month = format(Time, "%m")) %>%
  summarize(avg_temperature = mean(Temperature, na.rm = TRUE))

# 4c. Generate bar graphs for Average Monthly Temperatures in Newark, 2019
ggplot(monthlyAvg2019, aes(x = as.numeric(month), y = avg_temperature)) +
  geom_bar(stat = "identity", fill = "skyblue") +
  scale_x_continuous(breaks = 1:12, labels = month.abb) +
  labs(title = "Average Monthly Temperatures in 2019, Newark",
       x = "Month",
       y = "Average Temperature (°F)") +
  theme_minimal()

# 4d. Generate bar graphs for Average Monthly Temperatures in Newark, 2023
ggplot(monthlyAvg2023, aes(x = as.numeric(month), y = avg_temperature)) +
  geom_bar(stat = "identity", fill = "skyblue") +
  scale_x_continuous(breaks = 1:12, labels = month.abb) +
  labs(title = "Average Monthly Temperatures in 2023, Newark",
       x = "Month",
       y = "Average Temperature (°F)") +
  theme_minimal()

# 5a. Add a Season Column
assign_season <- function(month) {
  ifelse(month %in% c(12, 1, 2), "Winter",
         ifelse(month %in% c(3, 4, 5), "Spring",
                ifelse(month %in% c(6, 7, 8), "Summer", "Fall")
         )
  )
}

# 5b. Apply the function to create a "Season" column
newark2019$Season <- assign_season(as.numeric(format(newark2019$Time, "%m")))
newark2023$Season <- assign_season(as.numeric(format(newark2023$Time, "%m")))

# 5c. Calculate Average Humidity by Season
humidity2019 <- newark2019 %>%
  drop_na() %>%
```

```r
  group_by(Season) %>%
  summarize(avg_humidity = mean(Humidity, na.rm = TRUE))

humidity2023 <- newark2023 %>%
  drop_na() %>%
  group_by(Season) %>%
  summarize(avg_humidity = mean(Humidity, na.rm = TRUE))

# 5d. Add a year column to differentiate the datasets
humidity2019$Year <- 2019
humidity2023$Year <- 2023

# 5e. Combine both years into one data frame
combined_humidity <- rbind(humidity2019, humidity2023)

# 5f. Plot Double Bar Graph
ggplot(combined_humidity, aes(x = Season, y = avg_humidity, fill = as.factor(Year))) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "Seasonal Humidity Comparison in Newark, New Jersey: 2019 vs 2023",
       x = "Season",
       y = "Average Humidity (%)",
       fill = "Year") +
  theme_minimal()

# 6a. Add Month Column
newark2019$Month <- format(as.Date(newark2019$Time), "%B")  # Full month names
newark2023$Month <- format(as.Date(newark2023$Time), "%B")

# 6b. Calculate Average Monthly Precipitation
precipitation2019 <- newark2019 %>%
  drop_na() %>%
  group_by(Month) %>%
  summarize(avg_precipitation = mean(Precipitation, na.rm = TRUE)) %>%
  mutate(Year = 2019)

precipitation2023 <- newark2023 %>%
  drop_na() %>%
  group_by(Month) %>%
  summarize(avg_precipitation = mean(Precipitation, na.rm = TRUE)) %>%
  mutate(Year = 2023)

# 6c. Combine Both Years and Format the Table
```

```
summary_table <- merge(precipitation2019, precipitation2023, by = "Month", suffixes = c("_20
  select(Month, avg_precipitation_2019, avg_precipitation_2023) %>%
  arrange(factor(Month, levels = month.name))  # Order by month names

# 6d. Generate a Clean Table with knitr
kable(summary_table,
      caption = "Monthly Average Precipitation in Newark: 2019 vs 2023",
      col.names = c("Month", "2019 Avg Precipitation (inches)", "2023 Avg Precipitation (inch
      align = "c")
```