Greenville

Sam Tolbert

2024-10-02

Contents

R Markdown 1
Including Plots 4

R Markdown

```
Greenville <- read.csv("Hydrology/Data/Raw/Greenville_daily_precip_1980-present_HUC_030201030403_dayMet_

Greenville_Data <- Greenville

# Load necessary 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
```

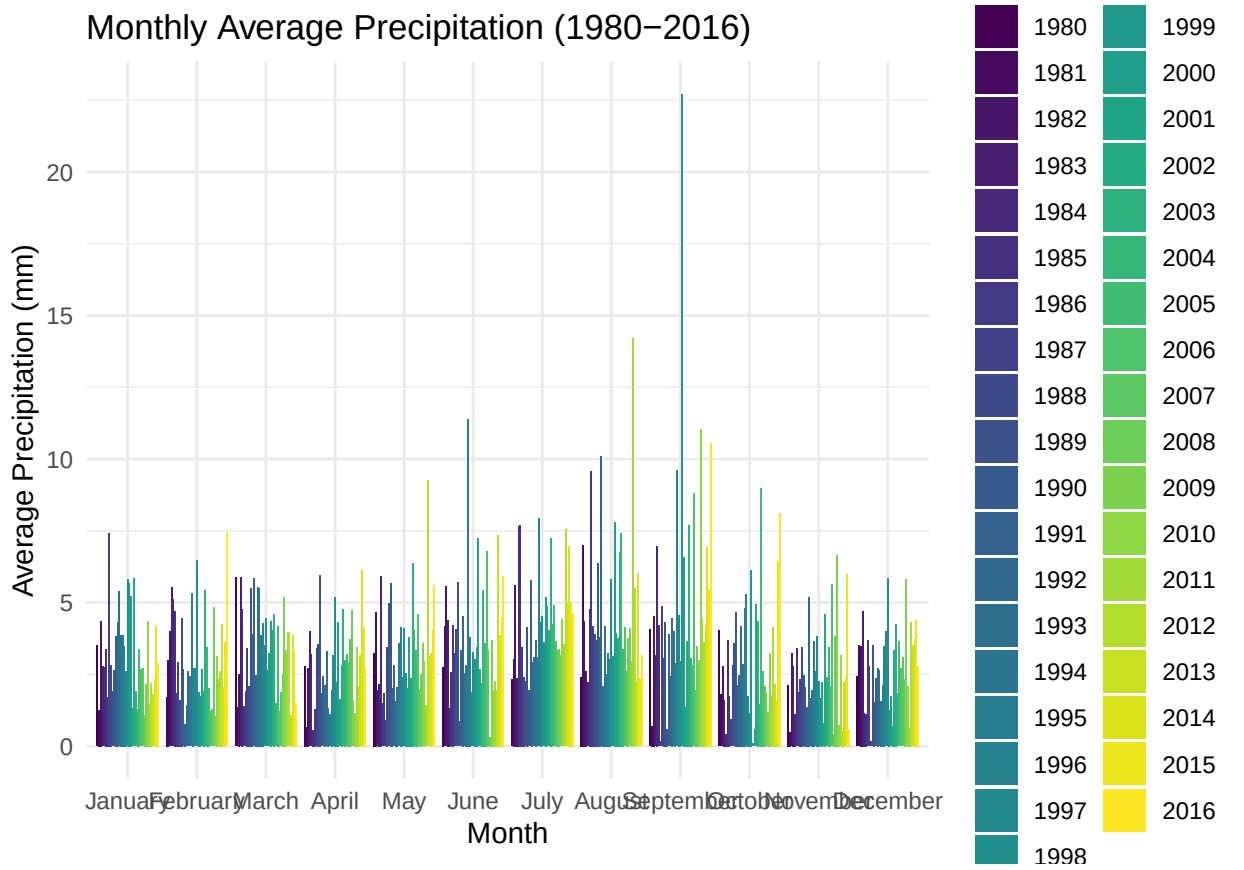
```
library(ggplot2)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

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

```r
# Rename the precipitation column to 'Precipitation in mm'
Greenville_Processed <- Greenville_Data %>%
  rename(Precipitation_mm = Area.Weighted.Mean.Precipitation..mm.per.day.)

# Ensure the 'Date' column is in date format
Greenville_Processed <- Greenville_Processed %>%
  mutate(Date = as.Date(Date))

# 2. Calculate monthly averages from 1980-2016
# Group by year and month, and calculate the mean precipitation for each month
Greenville_Monthly_Averages <- Greenville_Processed %>%
  filter(year >= 1980 & year <= 2016) %>%
  group_by(year, month) %>%
  summarize(monthly_avg_precip = mean(Precipitation_mm, na.rm = TRUE))
```

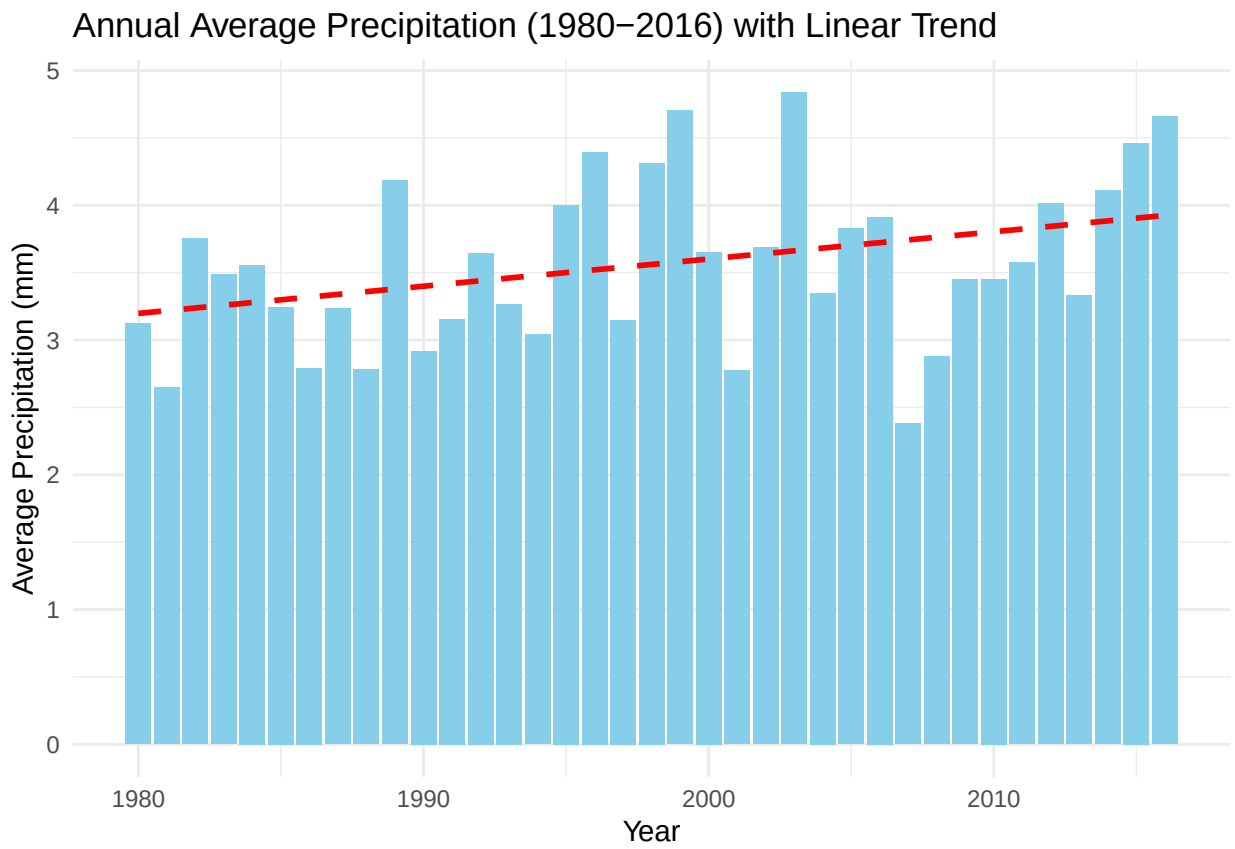
```
## `summarise()` has grouped output by 'year'. You can override using the
## `.groups` argument.
```

```r
# Plotting the monthly averages using a bar plot
Greenville_Annual_Average_Precip <- ggplot(Greenville_Monthly_Averages, aes(x = factor(month), y = montl
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "Monthly Average Precipitation (1980-2016)",
       x = "Month",
       y = "Average Precipitation (mm)") +
  theme_minimal() +
  scale_fill_viridis_d(name = "Year") +
  scale_x_discrete(labels = month.name)  # Adding month names to the x-axis
print(Greenville_Annual_Average_Precip)
```




```
Greenville_Annual_Averages <- Greenville_Processed %>%
  filter(year >= 1980 & year <= 2016) %>%
  group_by(year) %>%
  summarize(annual_avg_precip = mean(Precipitation_mm, na.rm = TRUE))

# Plotting the annual averages using a bar plot with a linear regression line
Greenville_Annual_Averages_Linear <- ggplot(Greenville_Annual_Averages, aes(x = year, y = annual_avg_pr
  geom_bar(stat = "identity", fill = "skyblue") +  # Bar plot
  geom_smooth(method = "lm", se = FALSE, color = "red", linetype = "dashed") +  # Linear regression lin
  labs(title = "Annual Average Precipitation (1980-2016) with Linear Trend",
       x = "Year",
       y = "Average Precipitation (mm)") +
  theme_minimal()
print(Greenville_Annual_Averages_Linear)
```

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



Including Plots

You can also embed plots, for example:

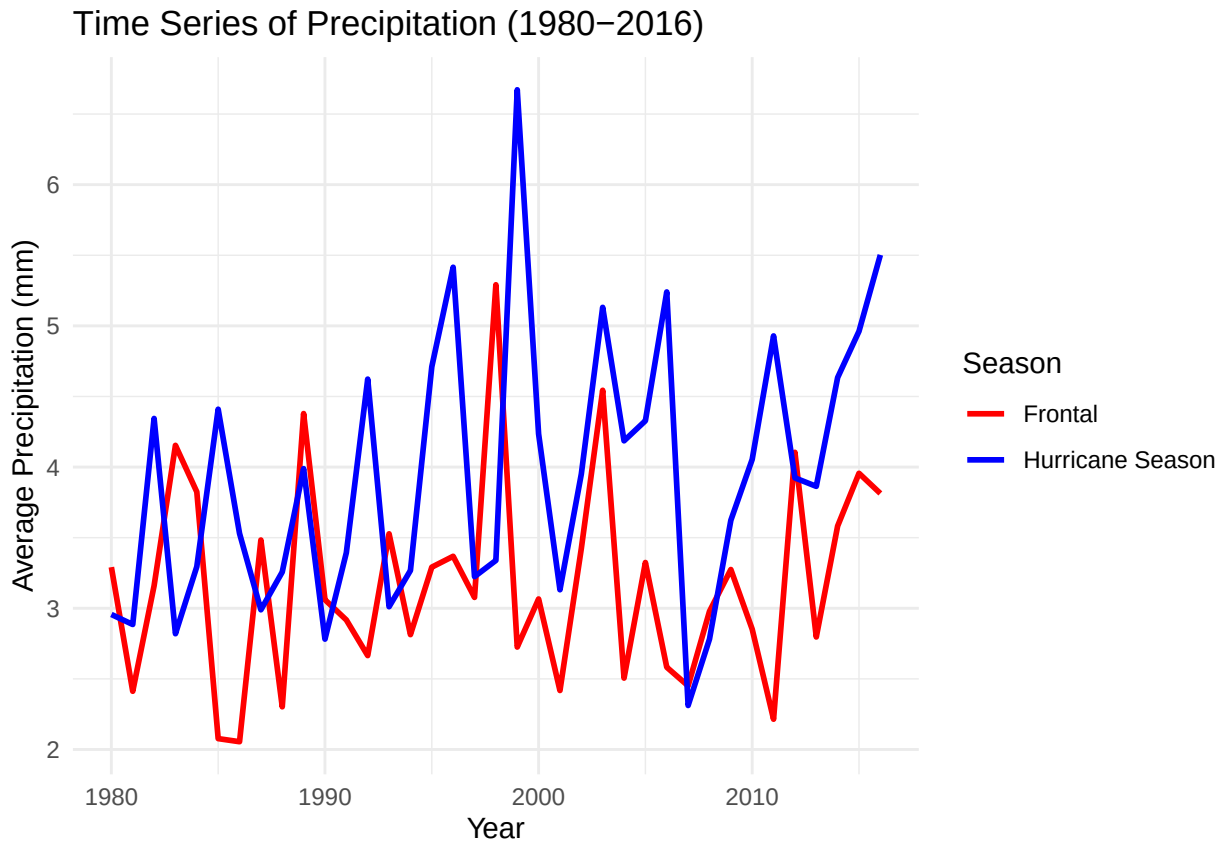
```
Greenville_Seasonal <- Greenville_Processed %>%
  mutate(Season = case_when(
    (month >= 6 & month <= 11) ~ "Hurricane Season",  # June to November
    TRUE ~ "Frontal"  # December to May
  ))

# 2. Filter data for the years 1980-2016
Greenville_Seasonal <- Greenville_Seasonal %>%
  filter(year >= 1980 & year <= 2016)

# 3. Group by year and season, and calculate the average precipitation for each year and season
Greenville_Seasonal_Averages <- Greenville_Seasonal %>%
  group_by(year, Season) %>%
  summarize(avg_precip = mean(Precipitation_mm, na.rm = TRUE))
```

```
## `summarise()` has grouped output by 'year'. You can override using the
## `.groups` argument.
```

```
# 4. Plot the results using a bar plot with separate linear regression lines for each season
Greenville_Seasonal_Averages_Plot <- ggplot(Greenville_Seasonal_Averages, aes(x = year, y = avg_precip,
```

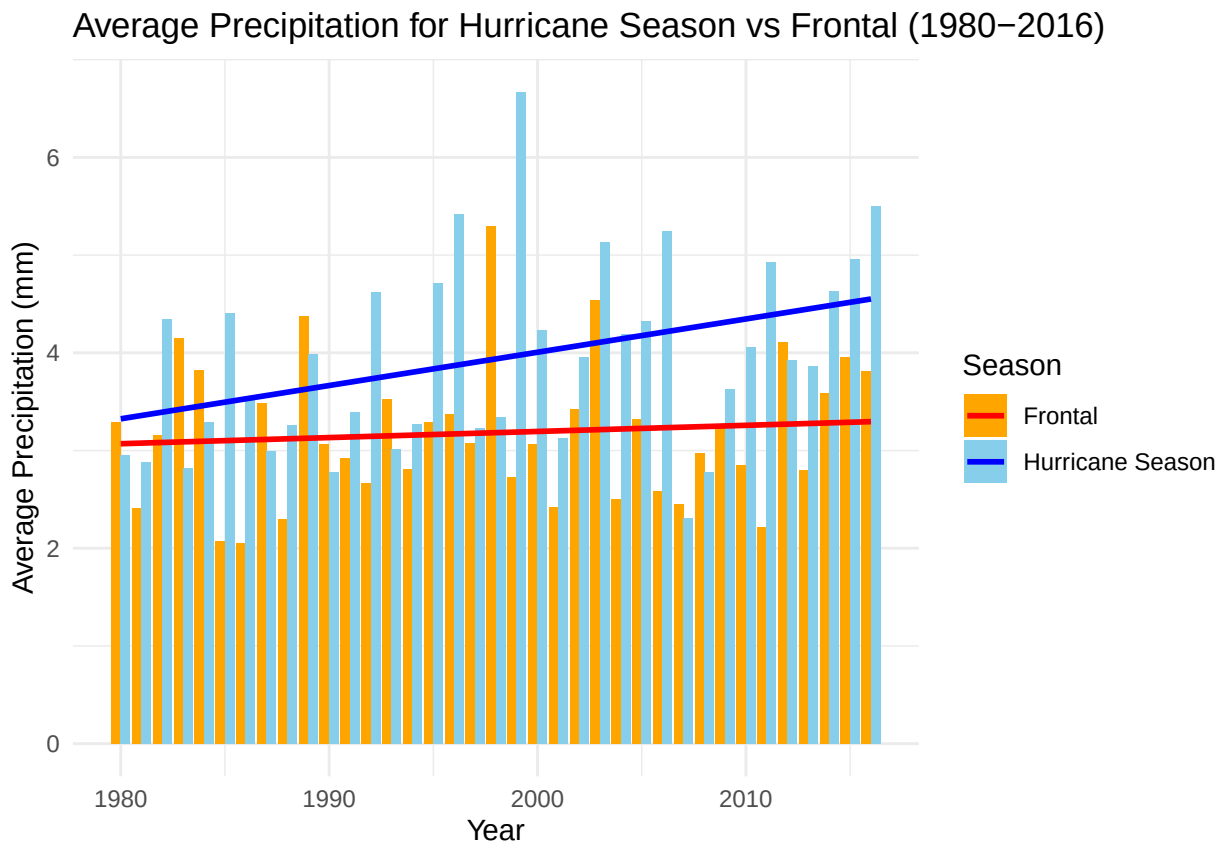
```r
  geom_bar(stat = "identity", position = "dodge") +  # Bar plot for both seasons side-by-side
  geom_smooth(method = "lm", aes(color = Season), se = FALSE) +  # Add separate linear regression lines
  labs(title = "Average Precipitation for Hurricane Season vs Frontal (1980-2016)",
       x = "Year",
       y = "Average Precipitation (mm)") +
  theme_minimal() +
  scale_fill_manual(values = c("Hurricane Season" = "skyblue", "Frontal" = "orange")) +
  scale_color_manual(values = c("Hurricane Season" = "blue", "Frontal" = "red"))

ggplot(Greenville_Seasonal_Averages, aes(x = year, y = avg_precip, color = Season, group = Season)) +
  geom_line(size = 1) +
  labs(title = "Time Series of Precipitation (1980-2016)",
       x = "Year",
       y = "Average Precipitation (mm)") +
  theme_minimal() +
  scale_color_manual(values = c("Hurricane Season" = "blue", "Frontal" = "red"))
```

```
## 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.
```

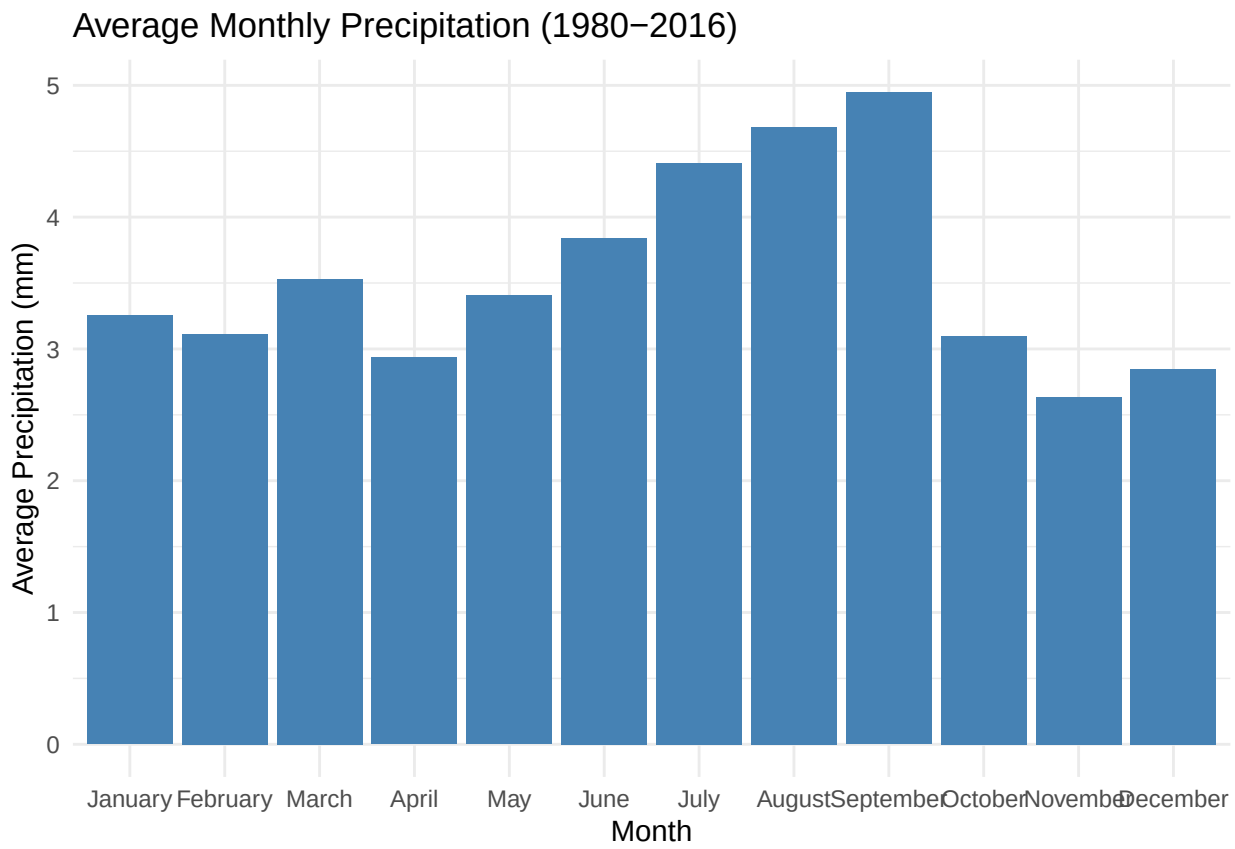


```
print(Greenville_Seasonal_Averages_Plot)
```

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




```
Greenville_Monthly_Averages_AllYears <- Greenville_Processed %>%
  filter(year >= 1980 & year <= 2016) %>%
  group_by(month) %>%
  summarize(monthly_avg_precip = mean(Precipitation_mm, na.rm = TRUE))

# Plotting the monthly averages (across all years) using a bar plot
Monthly_Averages_AllYears <- ggplot(Greenville_Monthly_Averages_AllYears, aes(x = factor(month), y = mo
  geom_bar(stat = "identity", fill = "steelblue") +
  labs(title = "Average Monthly Precipitation (1980-2016)",
       x = "Month",
       y = "Average Precipitation (mm)") +
  theme_minimal() +
  scale_x_discrete(labels = month.name)  # Adding month names to the x-axis
print(Monthly_Averages_AllYears)
```

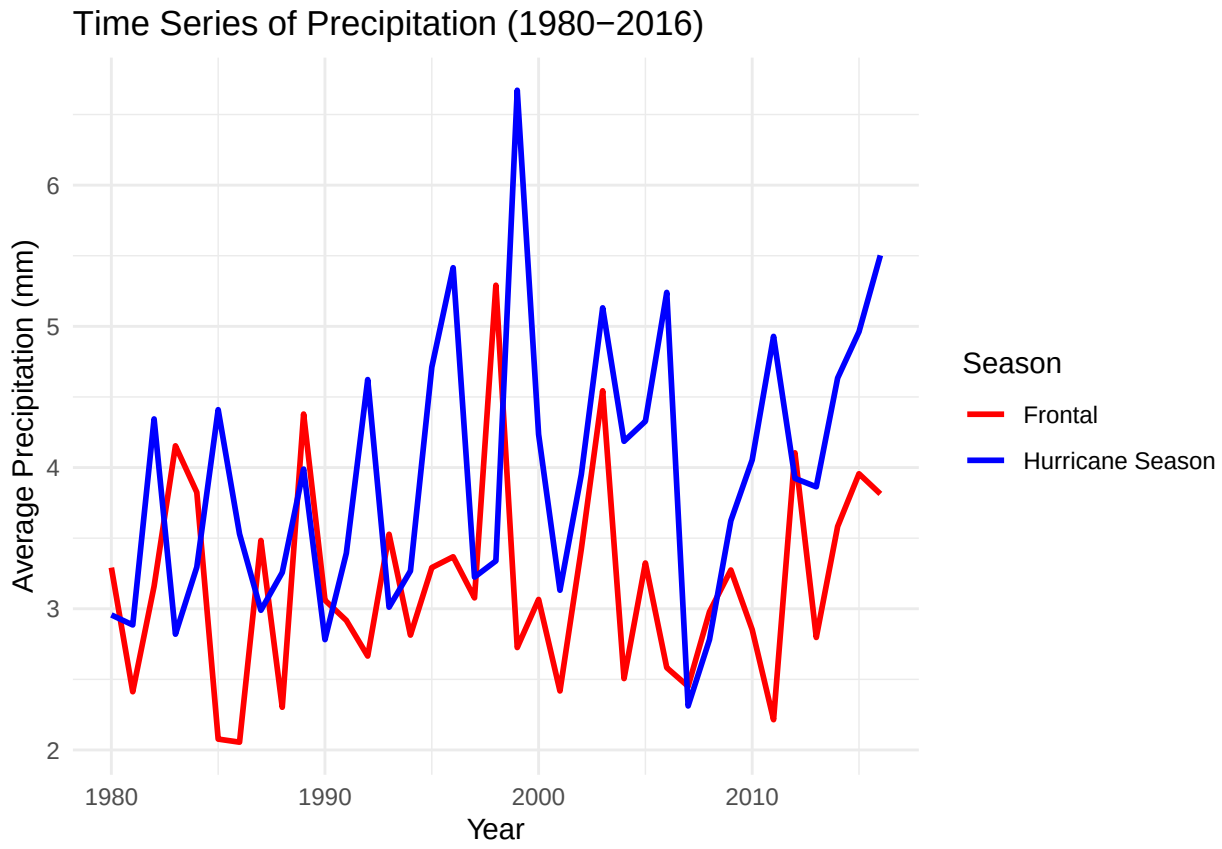



```
Greenville_Seasonal <- Greenville_Processed %>%
  mutate(Season = case_when(
    (month >= 6 & month <= 11) ~ "Hurricane Season",  # June to November
    TRUE ~ "Frontal"  # December to May
  ))

# 2. Filter data for the years 1980-2016
Greenville_Seasonal <- Greenville_Seasonal %>%
  filter(year >= 1980 & year <= 2016)

# 3. Group by year and season, and calculate the average precipitation for each year and season
Greenville_Seasonal_Averages <- Greenville_Seasonal %>%
  group_by(year, Season) %>%
  summarize(avg_precip = mean(Precipitation_mm, na.rm = TRUE))
```

```
## `summarise()` has grouped output by 'year'. You can override using the
## `.groups` argument.
```

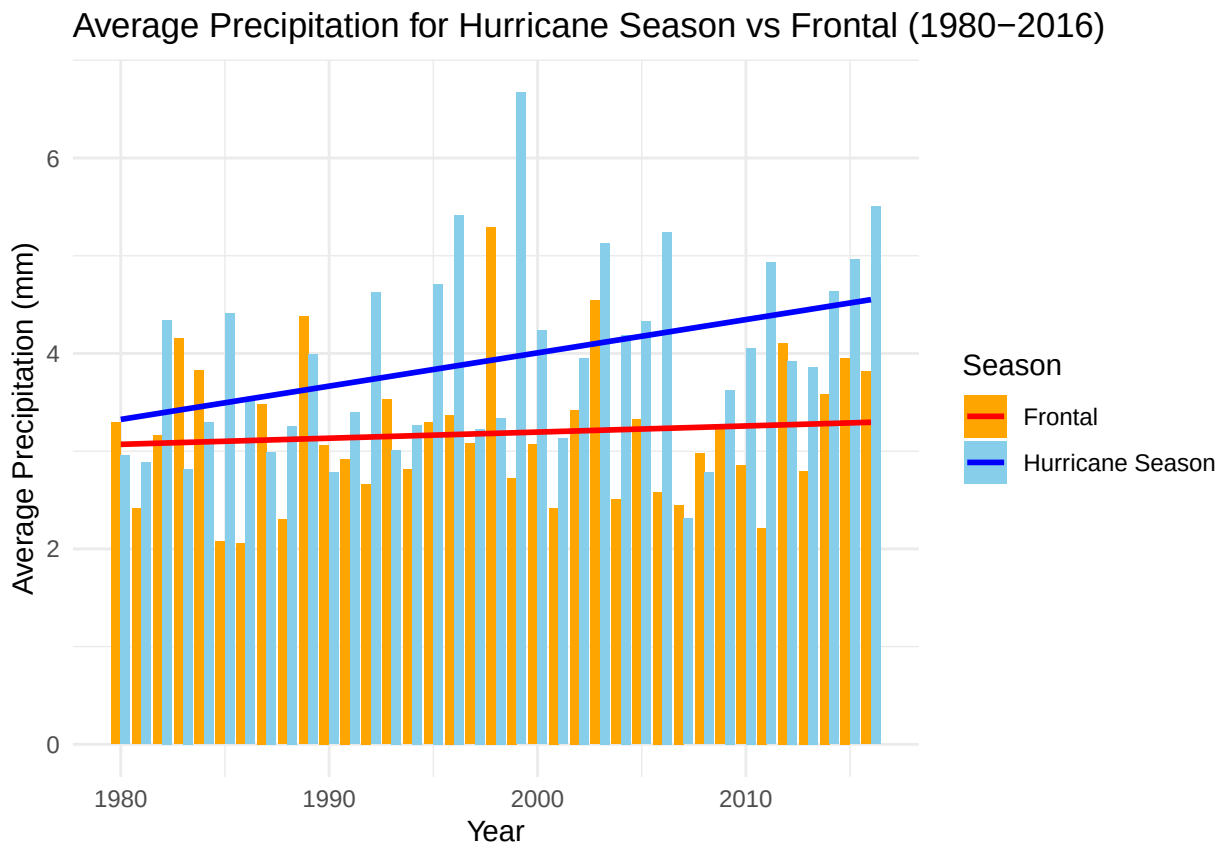
```
# 4. Plot the results using a bar plot with separate linear regression lines for each season
Greenville_Seasonal_Averages_Plot <- ggplot(Greenville_Seasonal_Averages, aes(x = year, y = avg_precip,
  geom_bar(stat = "identity", position = "dodge") +  # Bar plot for both seasons side-by-side
  geom_smooth(method = "lm", aes(color = Season), se = FALSE) +  # Add separate linear regression lines
  labs(title = "Average Precipitation for Hurricane Season vs Frontal (1980-2016)",
       x = "Year",
       y = "Average Precipitation (mm)") +
```

```
  theme_minimal() +
  scale_fill_manual(values = c("Hurricane Season" = "skyblue", "Frontal" = "orange")) +
  scale_color_manual(values = c("Hurricane Season" = "blue", "Frontal" = "red"))

ggplot(Greenville_Seasonal_Averages, aes(x = year, y = avg_precip, color = Season, group = Season)) +
  geom_line(size = 1) +
  labs(title = "Time Series of Precipitation (1980-2016)",
       x = "Year",
       y = "Average Precipitation (mm)") +
  theme_minimal() +
  scale_color_manual(values = c("Hurricane Season" = "blue", "Frontal" = "red"))
```



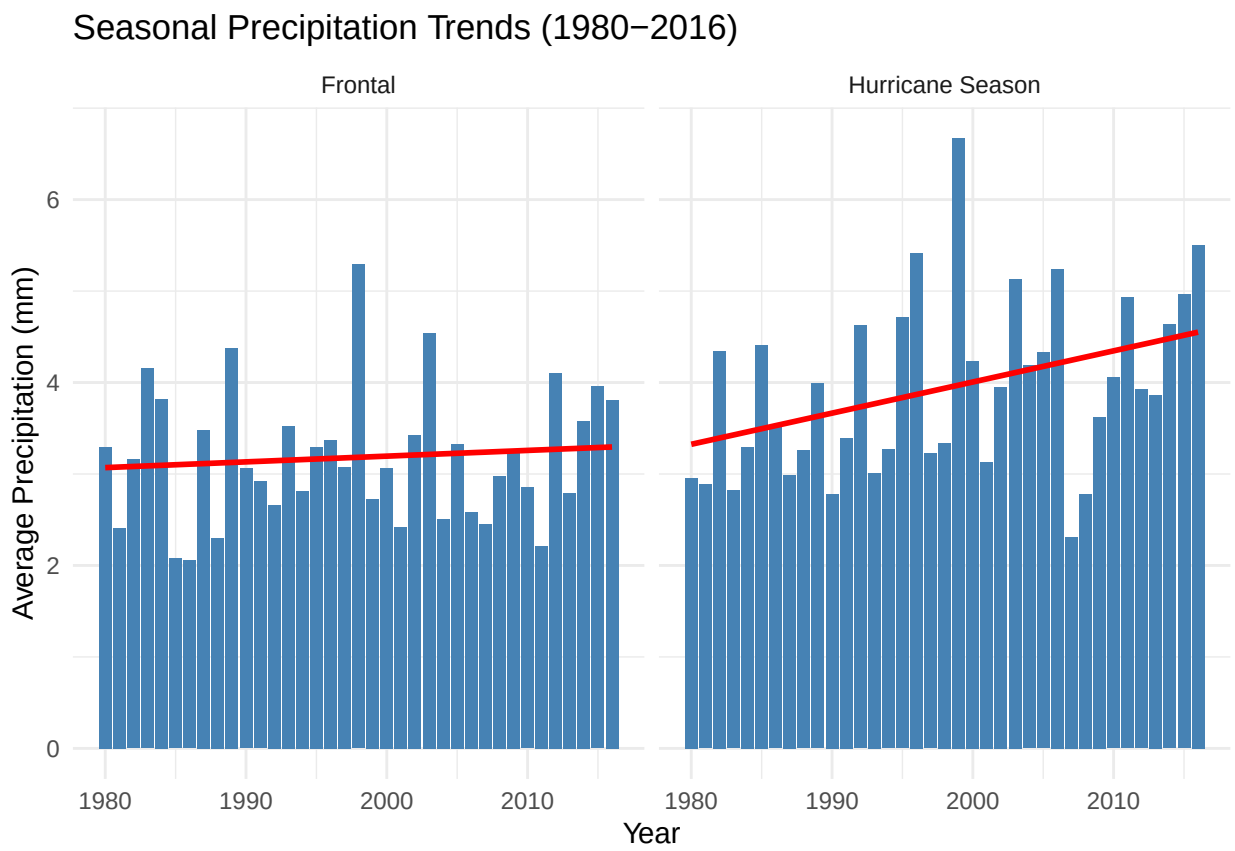

```
print(Greenville_Seasonal_Averages_Plot)
```

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



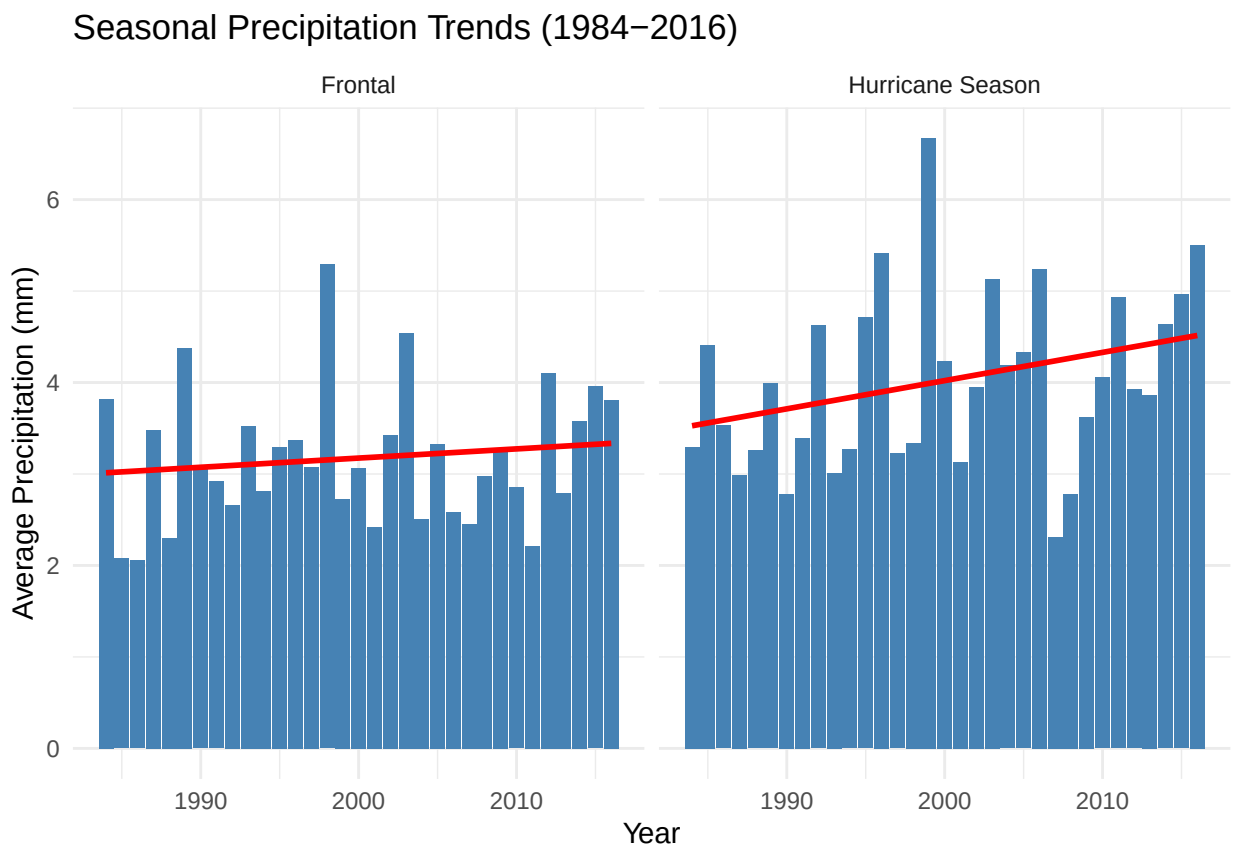

```
Greenville_Seasonal_Faceted_19802016 <- ggplot(Greenville_Seasonal_Averages, aes(x = year, y = avg_preci
  geom_bar(stat = "identity", fill = "steelblue") +  # Bar plot for average precipitation
  geom_smooth(method = "lm", se = FALSE, color = "red") +  # Add linear regression line
  facet_wrap(~ Season) +  # Facet by season ("Hurricane Season" and "Frontal")
  labs(title = "Seasonal Precipitation Trends (1980-2016)",
       x = "Year",
       y = "Average Precipitation (mm)") +
  theme_minimal()
print(Greenville_Seasonal_Faceted_19802016)
```

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




```
Greenville_Seasonal_Averages_Modified <- Greenville_Seasonal_Averages %>%
  filter(year >= 1984 & year <= 2016)

# Plot with separate regression lines for each season (1984-2016)
Greenville_Faceted_Seasonal_19842016 <- ggplot(Greenville_Seasonal_Averages_Modified, aes(x = year, y =
  geom_bar(stat = "identity", fill = "steelblue") +  # Bar plot for average precipitation
  geom_smooth(method = "lm", se = FALSE, color = "red") +  # Add linear regression line
  facet_wrap(~ Season) +  # Facet by season ("Hurricane Season" and "Frontal")
  labs(title = "Seasonal Precipitation Trends (1984-2016)",
       x = "Year",
       y = "Average Precipitation (mm)") +
  theme_minimal()
print(Greenville_Faceted_Seasonal_19842016)
```

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



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