

# Boone

Sam Tolbert

2024-10-02

## Contents

R Markdown . . . . .	1
Including Plots . . . . .	3

## R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
Boone <- read.csv("Hydrology/Data/Raw/Boone_daily_precip_1980-present_HUC_050500010201_dayMet_split-dat
Boone_Data<-Boone

# 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
```

*# Assuming your dataset is called 'Boone', and 'Area Weighted Mean Precipitation (mm per day)' is the p*

*# Rename the precipitation column to 'Precipitation in mm'*

```
Boone_Processed <- Boone_Data %>%
  rename(Precipitation_mm = Area.Weighted.Mean.Precipitation..mm.per.day.)
```

*# Ensure the 'Date' column is in date format*

```
Boone_Processed <- Boone_Processed %>%
  mutate(Date = as.Date(Date))
```

*# 2. Calculate monthly averages from 1980-2016*

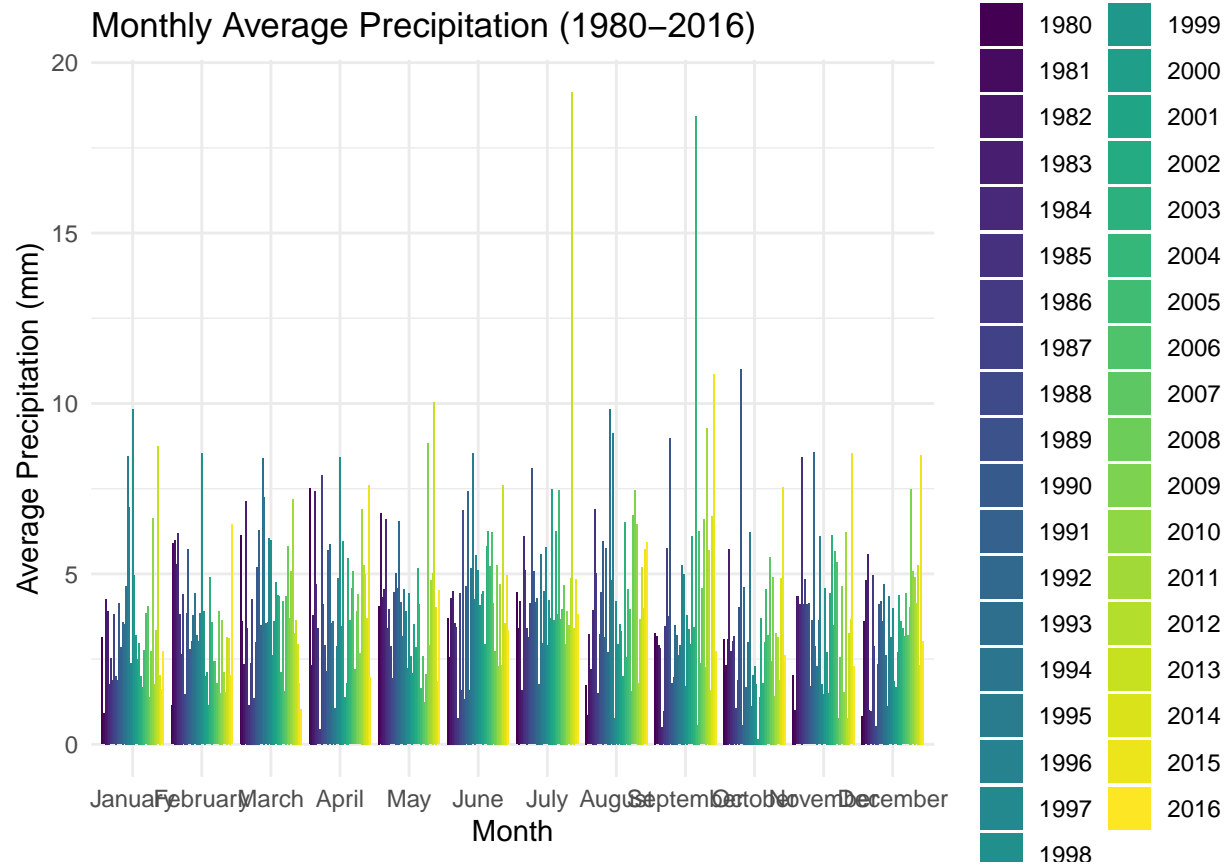
*# Group by year and month, and calculate the mean precipitation for each month*

```
Boone_Monthly_Averages <- Boone_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.
```

*# Plotting the monthly averages using a bar plot*

```
ggplot(Boone_Monthly_Averages, aes(x = factor(month), y = monthly_avg_precip, fill = factor(year))) +
  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
```

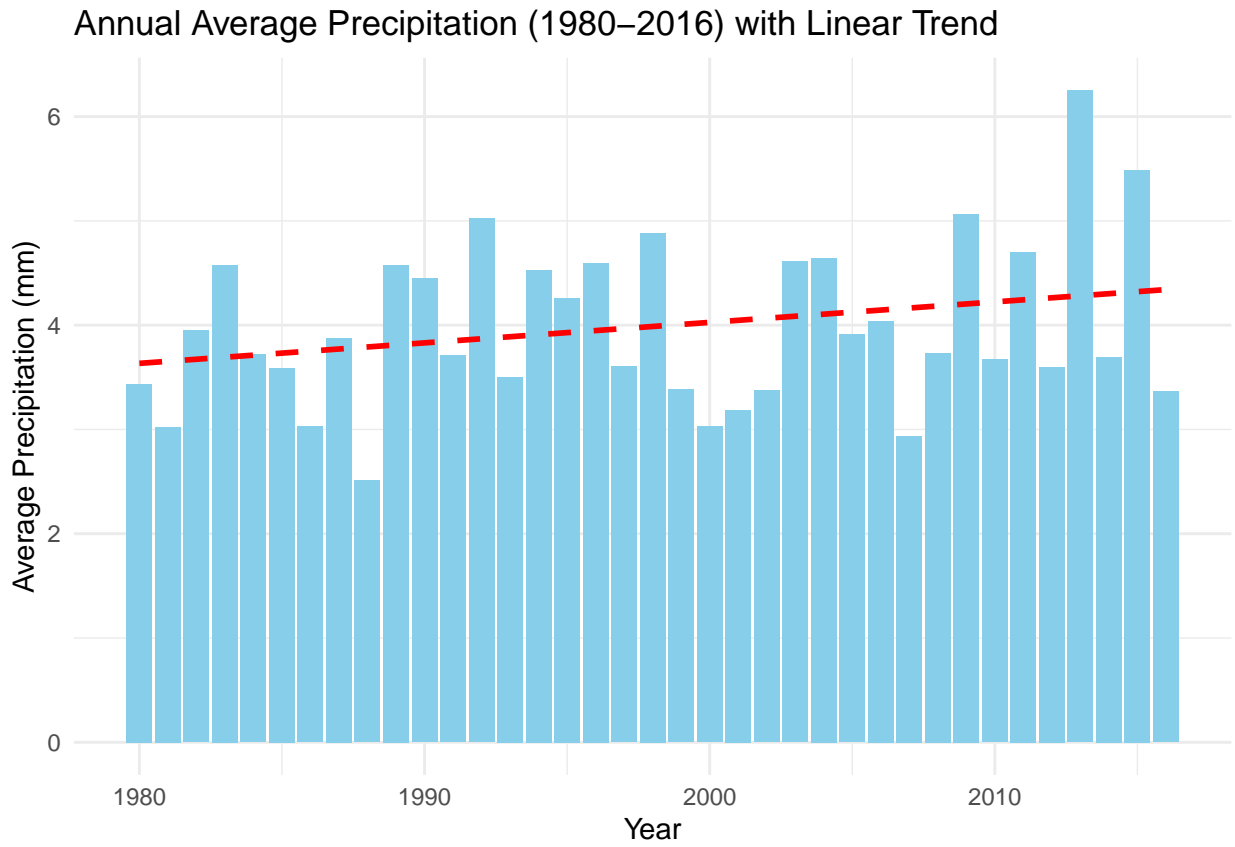


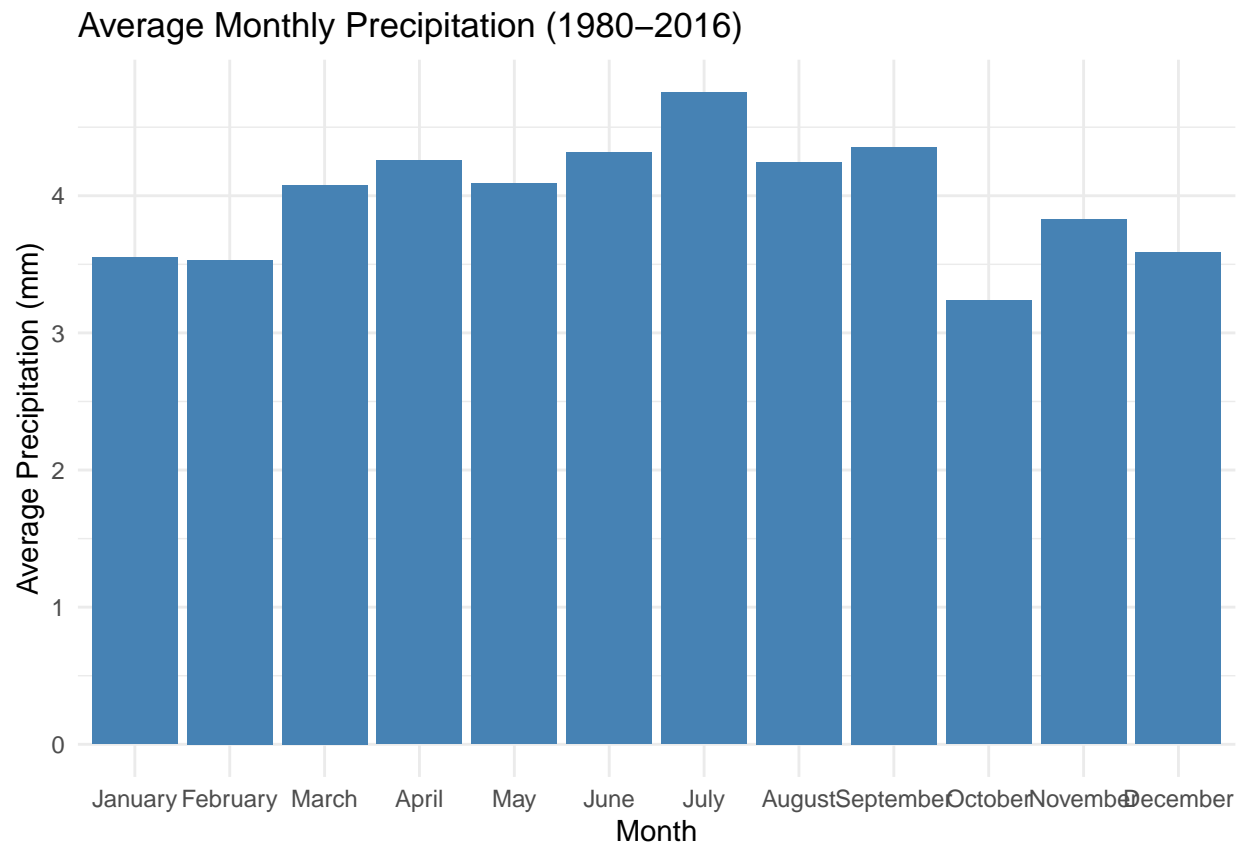
## Including Plots

You can also embed plots, for example:



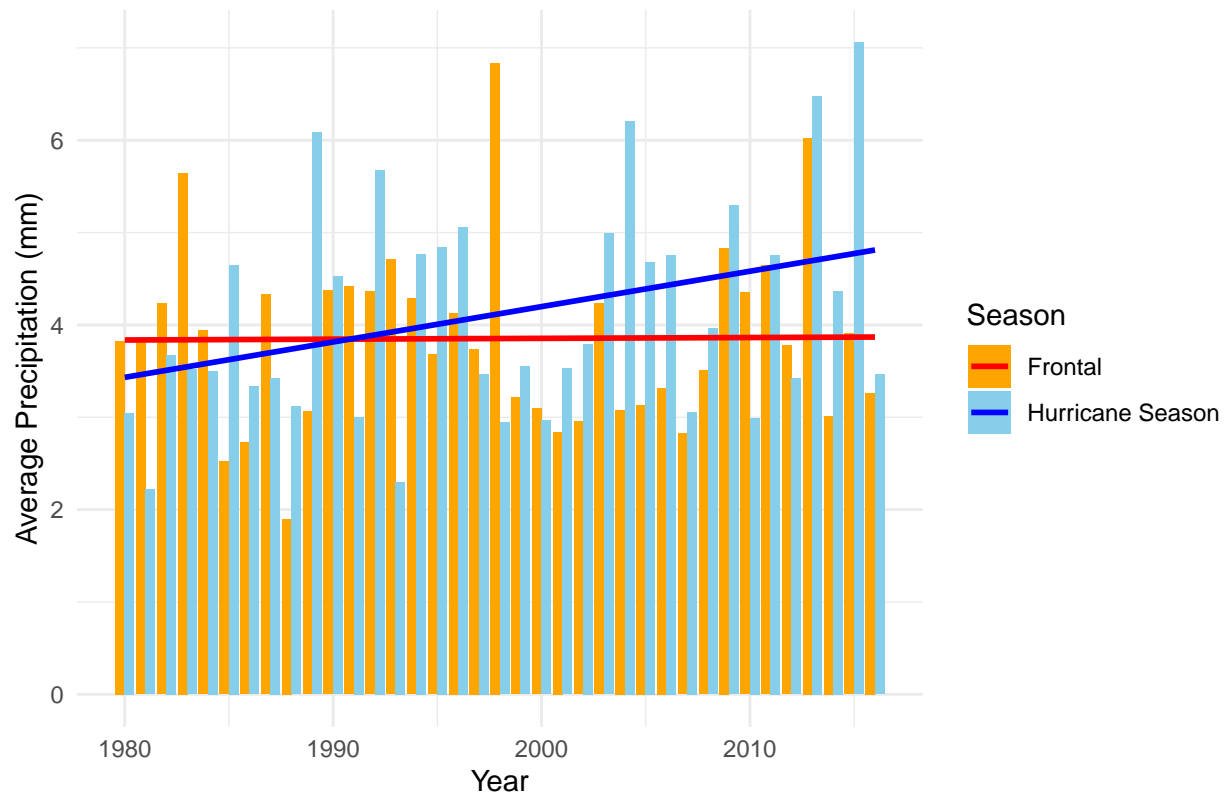
```
## 'geom_smooth()' using formula = 'y ~ x'
```



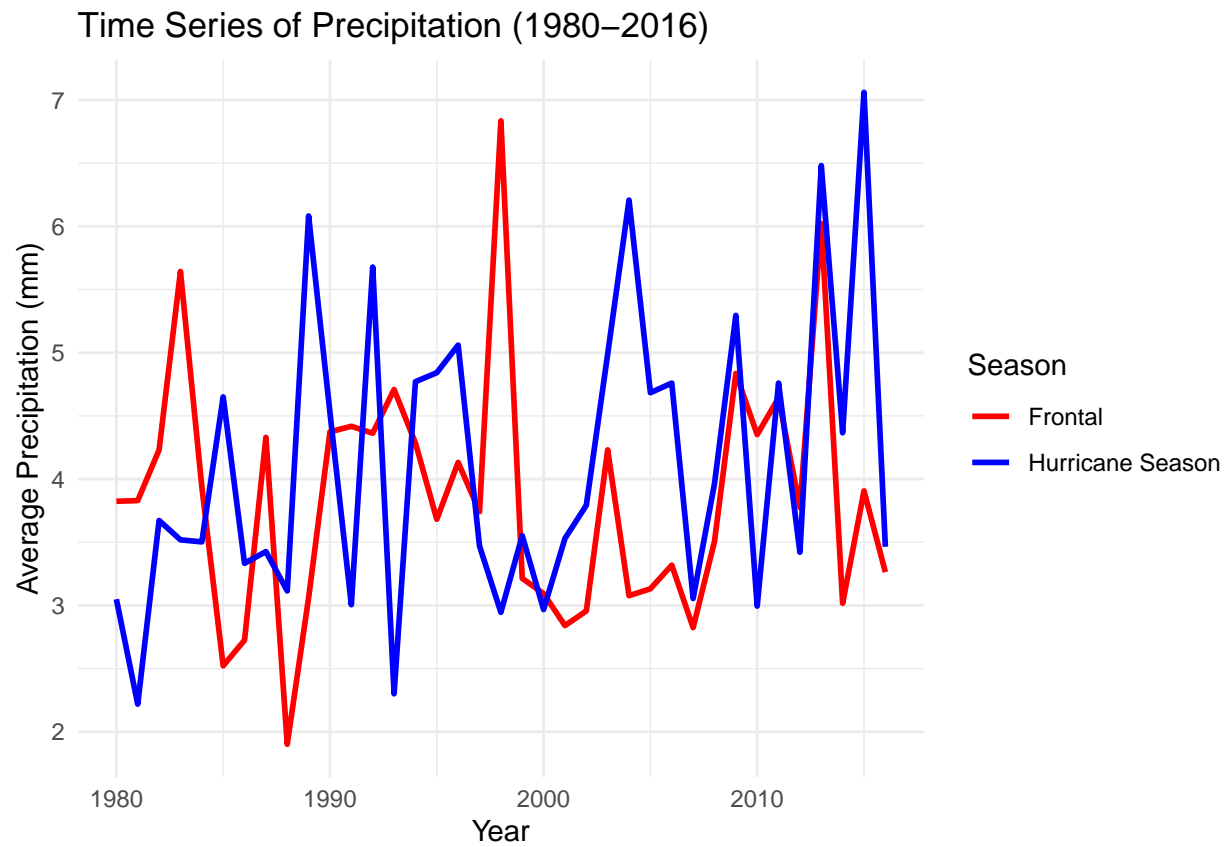


```
## 'summarise()' has grouped output by 'year'. You can override using the  
## '.groups' argument.  
## 'geom_smooth()' using formula = 'y ~ x'
```

Average Precipitation for Hurricane Season vs Frontal (1980–2016)



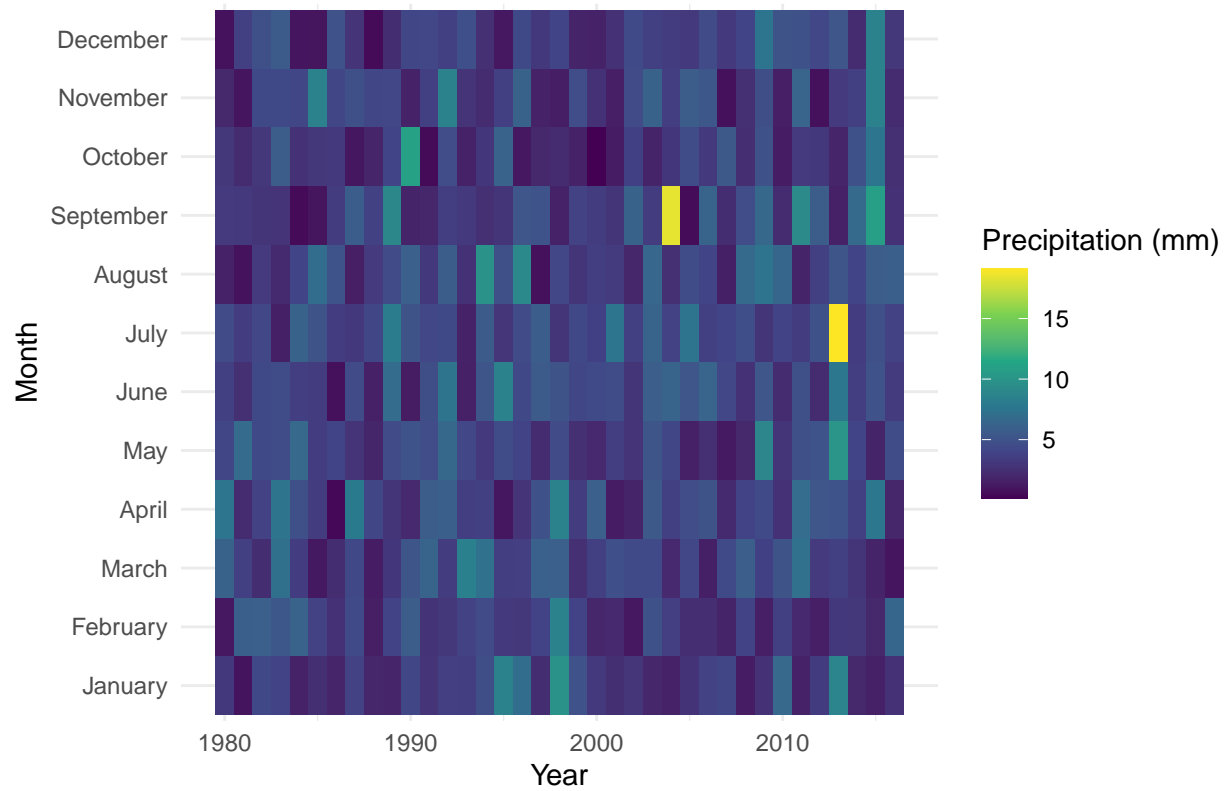
```
## Warning: Using 'size' aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use 'linewidth' instead.
## This warning is displayed once every 8 hours.
## Call 'lifecycle::last_lifecycle_warnings()' to see where this warning was
## generated.
```



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

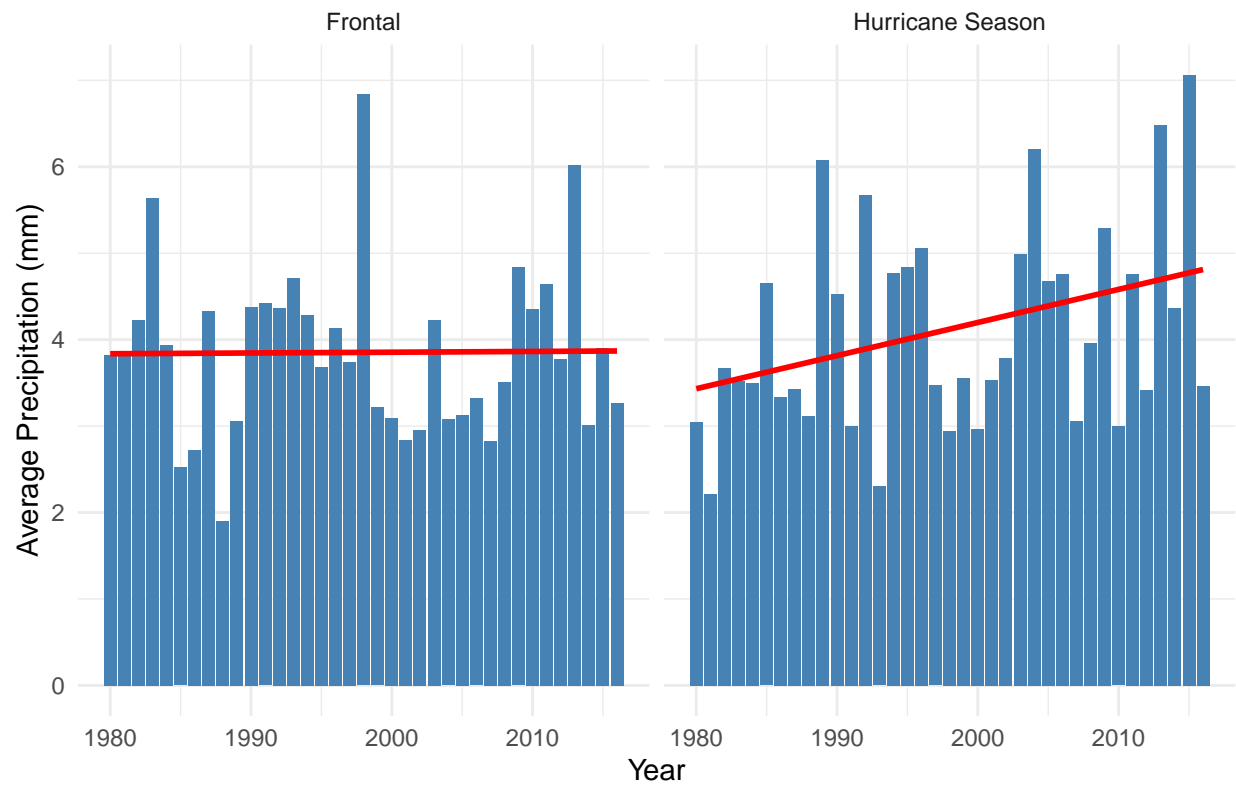


Heatmap of Monthly Average Precipitation (1980–2016)



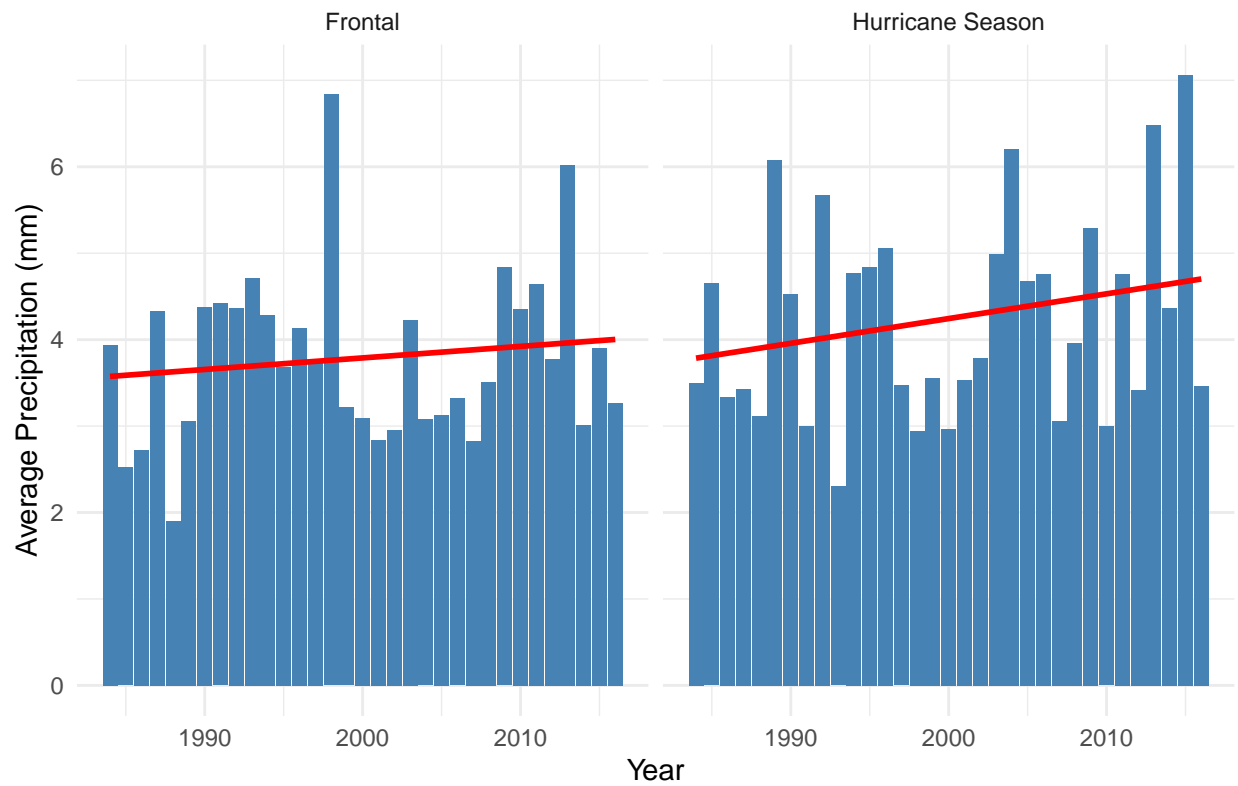
```
## 'geom_smooth()' using formula = 'y ~ x'
```

## Seasonal Precipitation Trends (1980–2016)



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

## Seasonal Precipitation Trends (1984–2016)



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