Day12 exercise solutions

Ali Movasati

Dec. 2nd, 2024

```
# Set global code chunk options
knitr::opts_chunk$set(warning = FALSE)
```

```
# load required libraries
library("extremefit")
library("extRemes")
library("ismev")
library("skimr")
library("dplyr")
library("tidyr")
library("magrittr")
library("ggplot2")
library("lubridate")



# define functions
`%notin%` <- Negate(`%in%`)
```

Problem 1

```
# Load the data

data(dataWind)
dataWind$Date <- make_date(year = dataWind$Year, month = dataWind$Month, day = dataWind$Day)
```

1.A)

```
# Stat summary EDA

head(dataWind)
```

```
##   Year Month Day Speed       Date
## 1 1976     1   2  18.0 1976-01-02
## 2 1976     1   3   8.1 1976-01-03
## 3 1976     1   4   5.9 1976-01-04
## 4 1976     1   5   5.3 1976-01-05
## 5 1976     1   6   5.5 1976-01-06
## 6 1976     1   7   9.8 1976-01-07
```

```
skim(dataWind)
```

Table 1: Data summary

Name	dataWind
Number of rows	10903
Number of columns	5
Column type frequency:	
Date	1
numeric	4
Group variables	None

Variable type: Date

skim_variable	n_missing	complete_rate	min	max	median	n_unique
Date	0	1	1976-01-02	2005-12-31	1991-01-27	10903

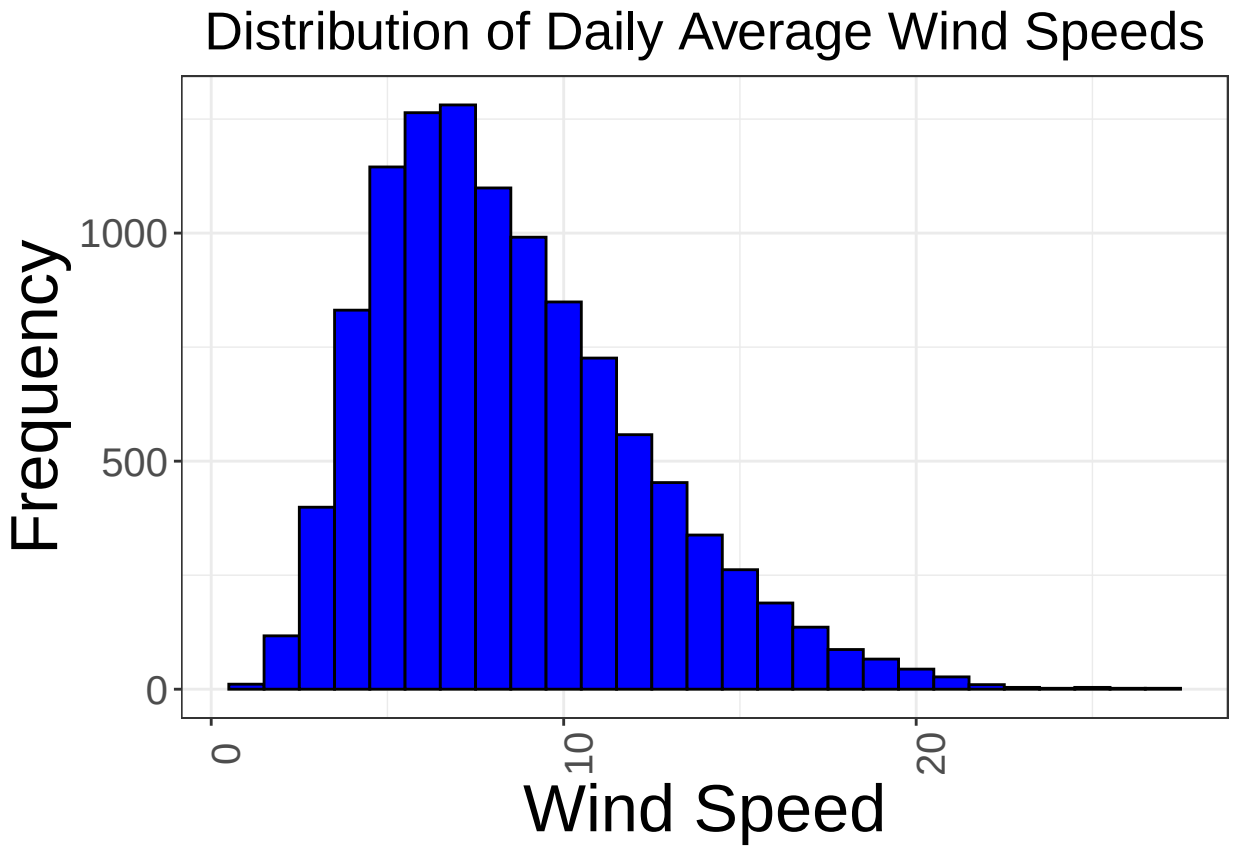
Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
Year	0	1	1990.51	8.67	1976.0	1983.0	1991.0	1998.0	2005.0	
Month	0	1	6.53	3.45	1.0	4.0	7.0	10.0	12.0	
Day	0	1	15.73	8.80	1.0	8.0	16.0	23.0	31.0	
Speed	6	1	8.55	3.75	0.7	5.7	7.9	10.8	27.4	

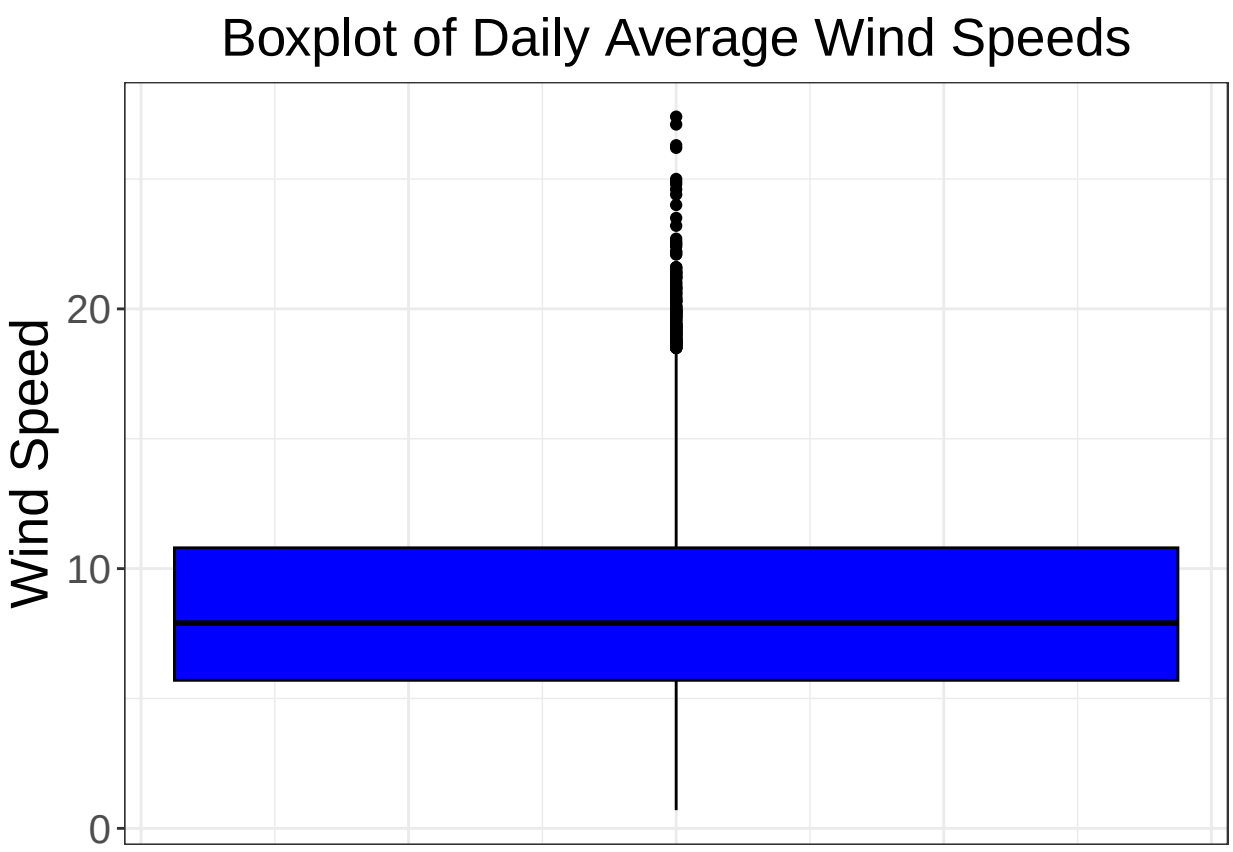
```
summary(dataWind)
```

```
##       Year          Month             Day            Speed             Date
##  Min.   :1976   Min.   : 1.000   Min.   : 1.00   Min.   : 0.700   Min.   :1976-01-02
##  1st Qu.:1983   1st Qu.: 4.000   1st Qu.: 8.00   1st Qu.: 5.700   1st Qu.:1983-06-27
##  Median :1991   Median : 7.000   Median :16.00   Median : 7.900   Median :1991-01-27
##  Mean   :1991   Mean   : 6.528   Mean   :15.73   Mean   : 8.553   Mean   :1991-01-06
##  3rd Qu.:1998   3rd Qu.:10.000   3rd Qu.:23.00   3rd Qu.:10.800   3rd Qu.:1998-07-14
##  Max.   :2005   Max.   :12.000   Max.   :31.00   Max.   :27.400   Max.   :2005-12-31
##                                                  NA's   :6
```

```r
# Visual EDA

## histogram of wind speed

hist_fig <-
    dataWind %>%
        ggplot(aes(x = Speed)) +
        geom_histogram(binwidth = 1, fill = "blue", color = "black") +
        labs(title = "Distribution of Daily Average Wind Speeds",
            x = "Wind Speed",
            y = "Frequency") +
        theme_bw() +
        theme(plot.title = element_text(size = 20, hjust = 0.5),
            axis.title = element_text(size = 25),
            axis.text.x = element_text(size = 15, angle = 90),
            axis.text.y = element_text(size = 15))

hist_fig
```

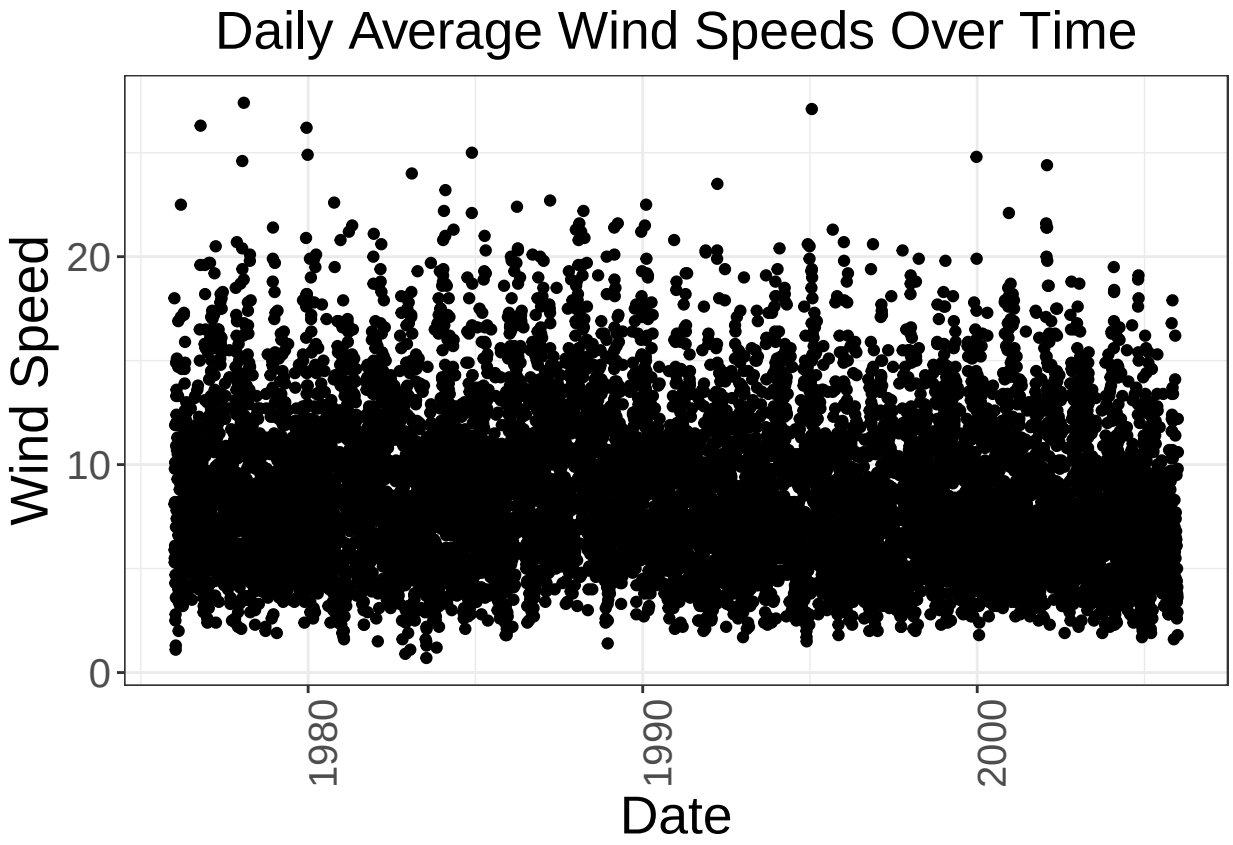


```r
box_fig <-
    dataWind %>%
        ggplot(aes(y = Speed)) +
        geom_boxplot(fill = "blue", color = "black") +
        labs(title = "Boxplot of Daily Average Wind Speeds",
            y = "Wind Speed") +
        theme_bw() +
        theme(plot.title = element_text(size = 20, hjust = 0.5),
            axis.title = element_text(size = 20),
            axis.text.x = element_blank(),
            axis.ticks.x = element_blank(),
            axis.text.y = element_text(size = 15))

box_fig
```

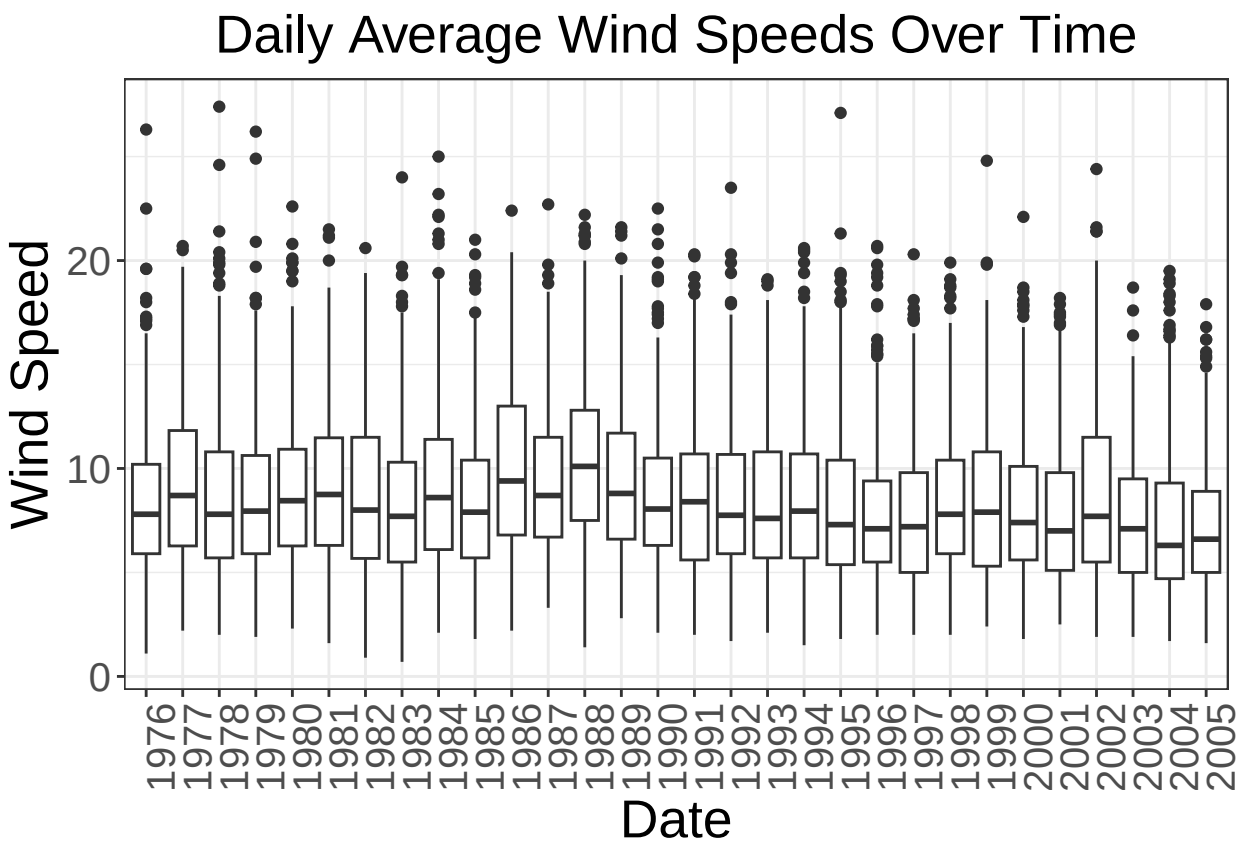



```r
point_fig <-
    dataWind %>%
    ggplot(aes(x = Date, y = Speed)) +
    geom_point() +
    labs(title = "Daily Average Wind Speeds Over Time",
        x = "Date",
        y = "Wind Speed") +
        theme_bw() +
        theme(plot.title = element_text(size = 20, hjust = 0.5),
            axis.title = element_text(size = 20),
            axis.text.x = element_text(size = 15, angle = 90),
            axis.text.y = element_text(size = 15))
```

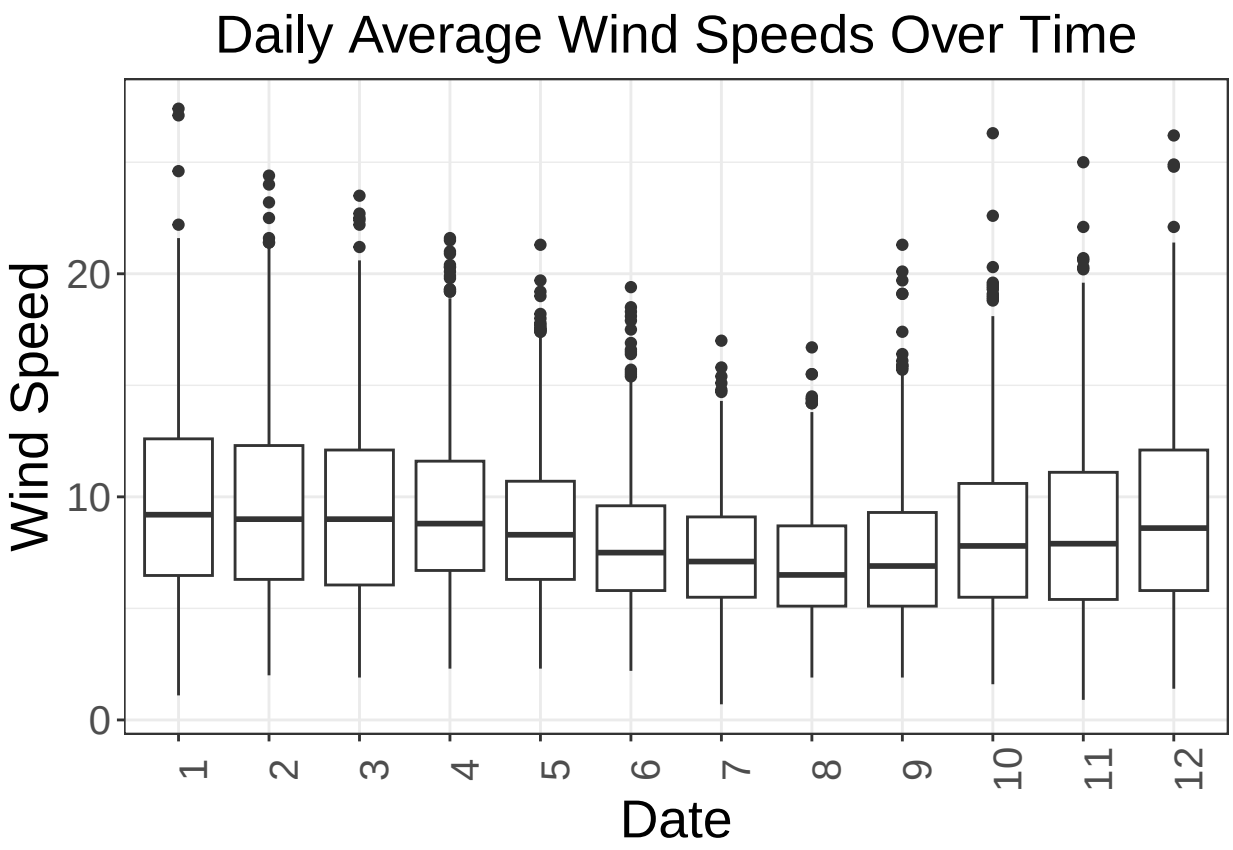
```
point_fig
```




```
yearly_fig <-
    dataWind %>%
    ggplot(aes(x = factor(Year), y = Speed)) +
    geom_boxplot() +
    labs(title = "Daily Average Wind Speeds Over Time",
        x = "Date",
        y = "Wind Speed") +
        theme_bw() +
        theme(plot.title = element_text(size = 20, hjust = 0.5),
            axis.title = element_text(size = 20),
            axis.text.x = element_text(size = 15, angle = 90),
            axis.text.y = element_text(size = 15))

yearly_fig
```




```
seasonality_fig <-
    dataWind %>%
    ggplot(aes(x = factor(Month), y = Speed)) +
    geom_boxplot() +
    labs(title = "Daily Average Wind Speeds Over Time",
        x = "Date",
        y = "Wind Speed") +
        theme_bw() +
        theme(plot.title = element_text(size = 20, hjust = 0.5),
            axis.title = element_text(size = 20),
            axis.text.x = element_text(size = 15, angle = 90),
            axis.text.y = element_text(size = 15))

seasonality_fig
```

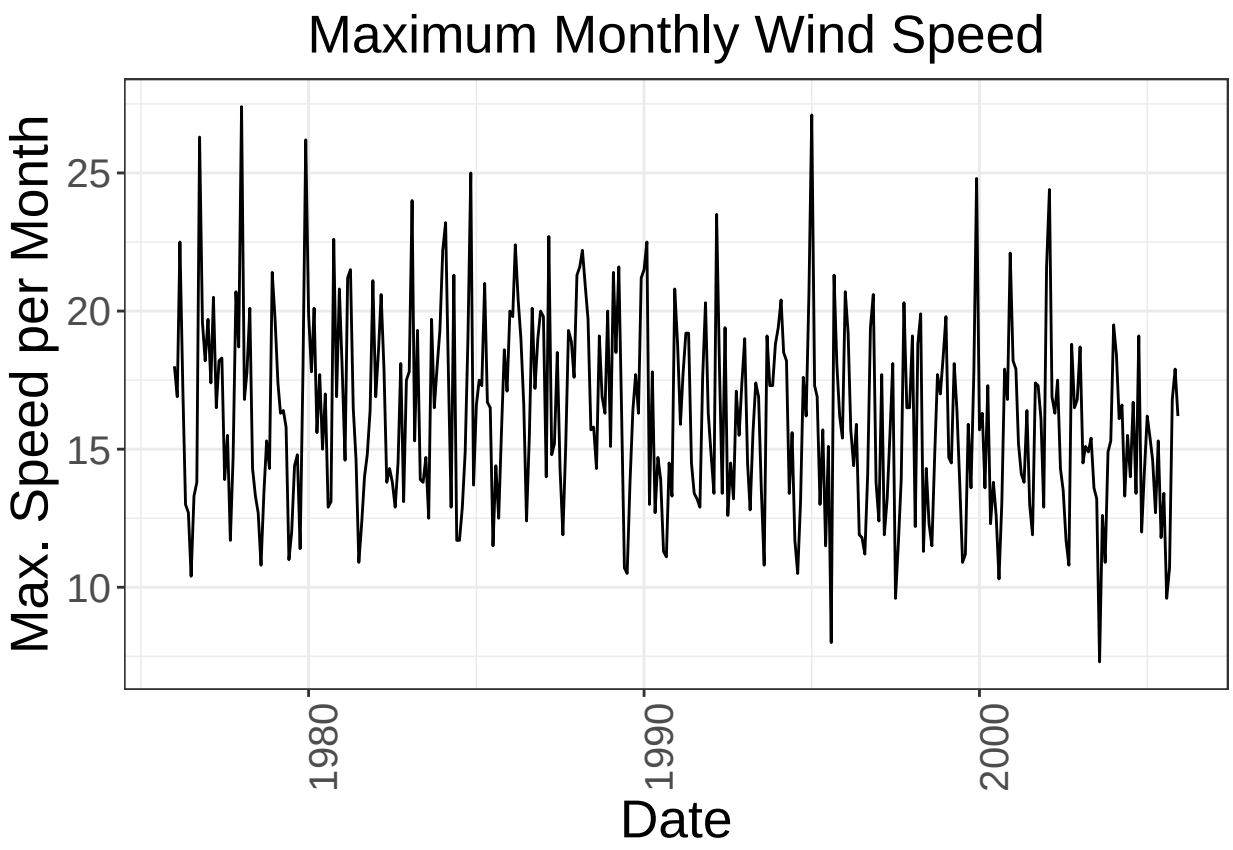


1.B)

```
# Prepare data
monthly_max <-
    dataWind %>%
        select(Year, Month, Speed) %>%
        group_by(Year, Month) %>%
        summarize(max_speed = max(Speed, na.rm = T)) %>%
        mutate(Date = make_date(year = Year, month = Month))
```

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

```
# plot
monthly_max_fig <-
    monthly_max %>%
    ggplot(aes(x= Date, y = max_speed)) +
    geom_line() +
    labs(title = "Maximum Monthly Wind Speed",
    x = "Date",
    y = "Max. Speed per Month") +
    theme_bw() +
        theme(plot.title = element_text(size = 20, hjust = 0.5),
            axis.title = element_text(size = 20),
            axis.text.x = element_text(size = 15, angle = 90),
            axis.text.y = element_text(size = 15))

monthly_max_fig
```



1.C)

```
?fevd

gev_fit <- fevd(monthly_max$max_speed)

summary(gev_fit)
```

```
##
## fevd(x = monthly_max$max_speed)
##
## [1] "Estimation Method used: MLE"
##
##
##  Negative Log-Likelihood Value:  947.4815
##
##
##  Estimated parameters:
##  location     scale     shape
## 14.866010  3.151157 -0.161416
##
##  Standard Error Estimates:
##   location      scale      shape
## 0.18331374 0.12820902 0.03231831
##
##  Estimated parameter covariance matrix.
##              location        scale        shape
```

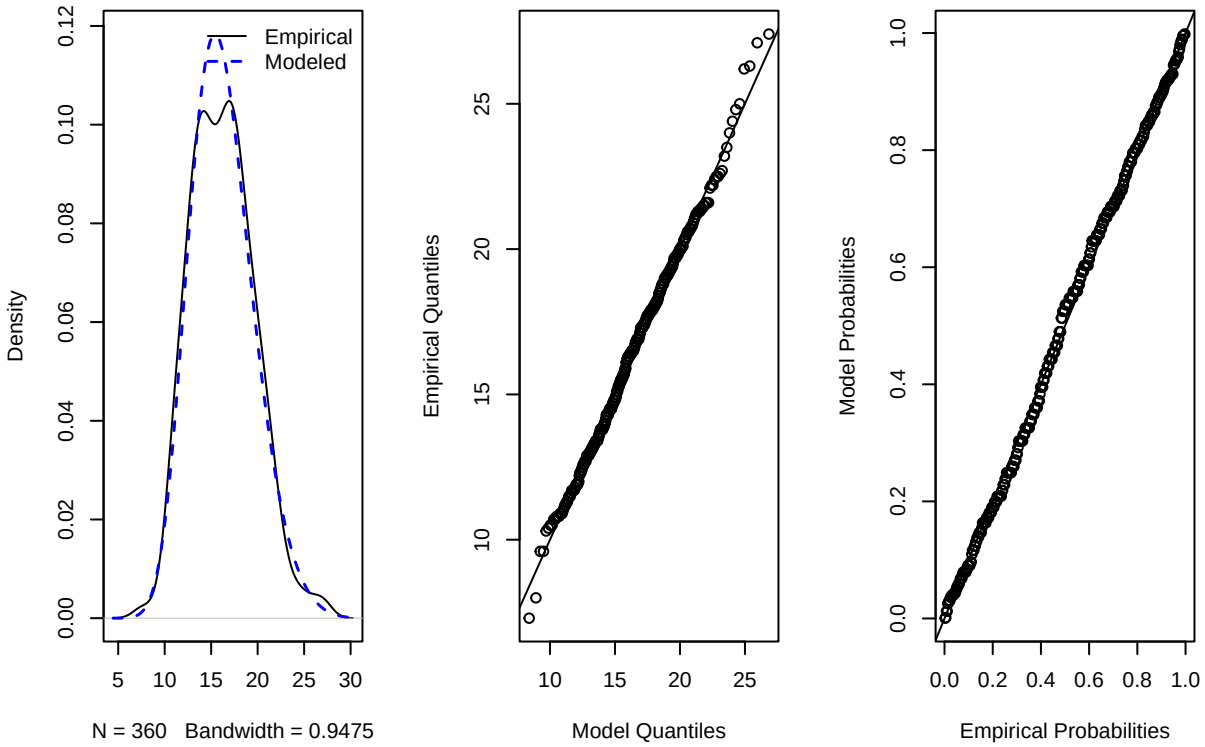
```
## location  0.033603927  0.004008646 -0.002101318
## scale     0.004008646  0.016437553 -0.001861240
## shape    -0.002101318 -0.001861240  0.001044473
##
##  AIC = 1900.963
##
##  BIC = 1912.621
```

```
ci(gev_fit, type = "parameter")
```

```
## fevd(x = monthly_max$max_speed)
##
## [1] "Normal Approx."
##
##          95% lower CI  Estimate 95% upper CI
## location   14.5067215 14.866010   15.2252981
## scale       2.8998716  3.151157    3.4024418
## shape      -0.2247587 -0.161416   -0.0980733
```

1.D)

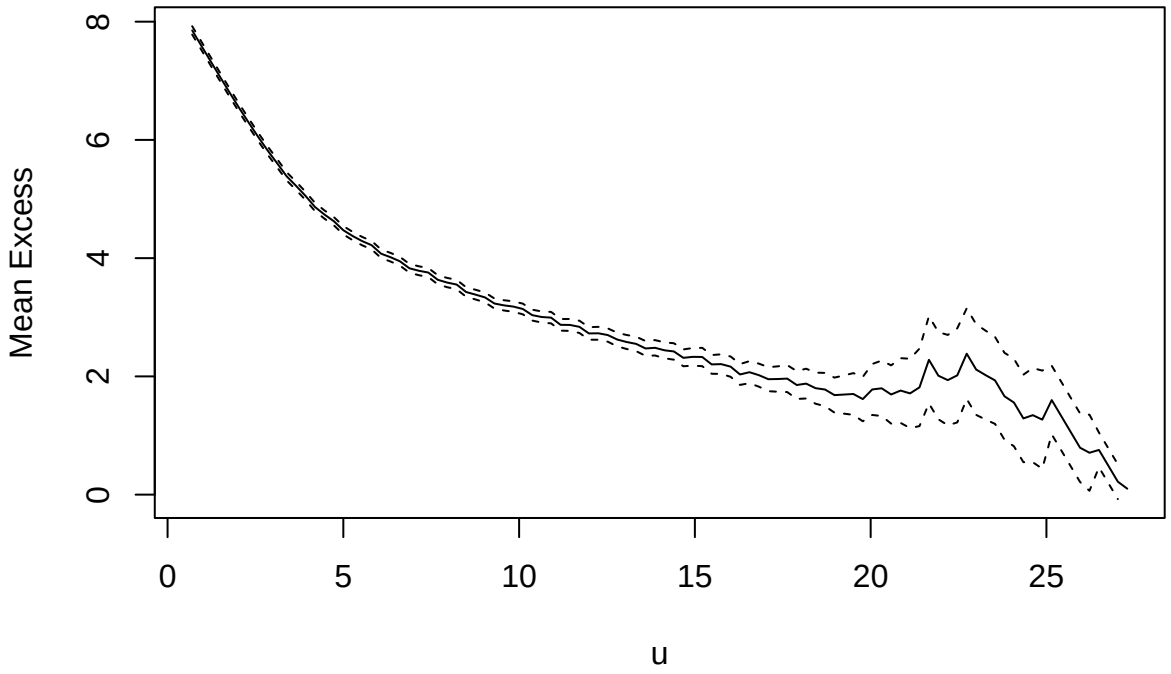
```
par(mfrow=c(1,3))
plot(gev_fit, type = "density", main = "")
plot(gev_fit, type = "qq", main = "")
plot(gev_fit, type = "prob", main = "")
```



1.E)

```
## choose a threshold
```

```r
mrl.plot(dataWind$Speed[!is.na(dataWind$Speed)])
```



« comments »

Based on the mean residual life, the range of thresholds that are somewhat stable (linear) is from 18 to 22. We take 20 as threshold.

```r
# Fit a GPD model

gpd_fit <- fevd(dataWind$Speed[!is.na(dataWind$Speed)], threshold = 20, type = "GP", time.units="days")

summary(gpd_fit)
```

```
##
## fevd(x = dataWind$Speed[!is.na(dataWind$Speed)], threshold = 20,
##     type = "GP", time.units = "days")
##
## [1] "Estimation Method used: MLE"
##
##
##  Negative Log-Likelihood Value:  105.6396
##
##
##  Estimated parameters:
##      scale      shape
##  1.8965652 -0.0394441
##
##  Standard Error Estimates:
##     scale     shape
## 0.3551409 0.1411950
##
##  Estimated parameter covariance matrix.
##              scale       shape
```

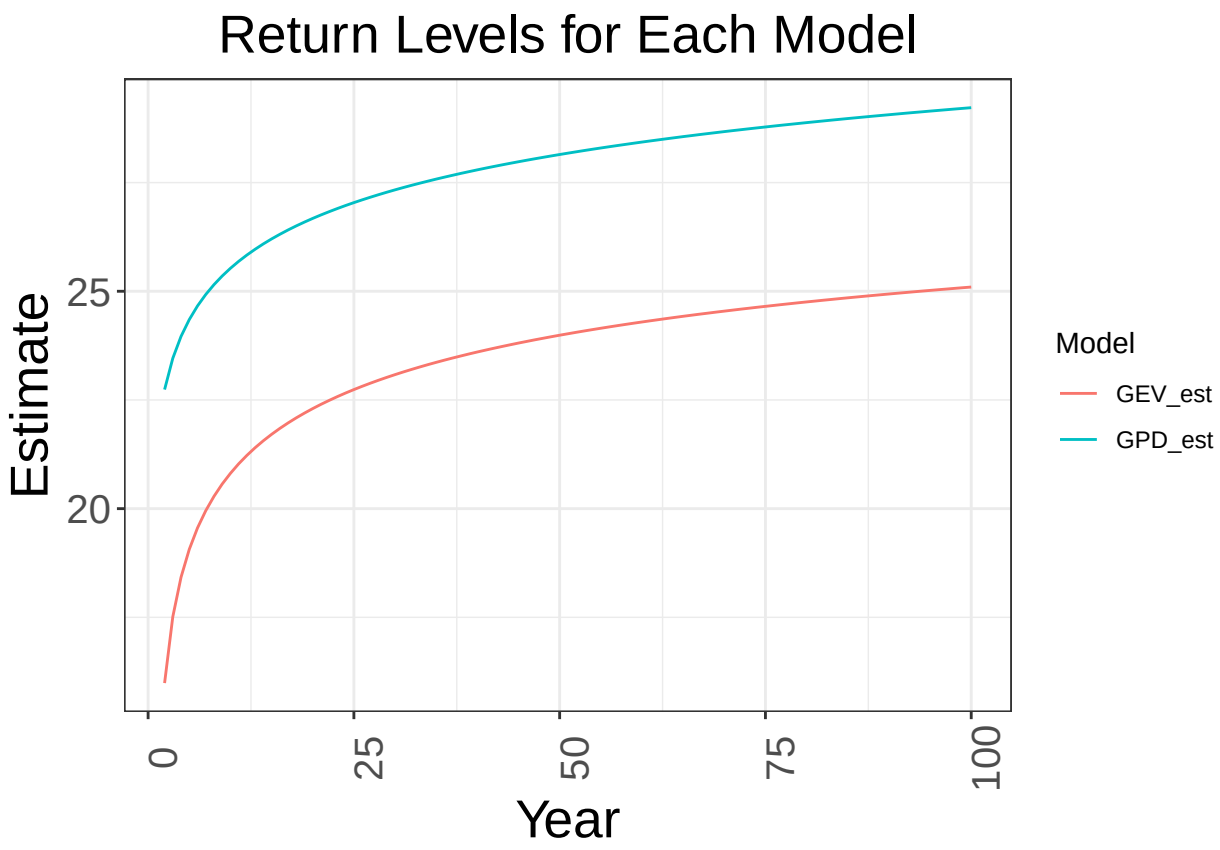
```
## scale  0.12612507 -0.03889909
## shape -0.03889909  0.01993603
##
##  AIC = 215.2792
##
##  BIC = 219.6585
```

```
ci(gpd_fit, type = "parameter")
```

```
## fevd(x = dataWind$Speed[!is.na(dataWind$Speed)], threshold = 20,
##     type = "GP", time.units = "days")
##
## [1] "Normal Approx."
##
##       95% lower CI   Estimate 95% upper CI
## scale    1.2005018  1.8965652     2.592629
## shape   -0.3161812 -0.0394441     0.237293
```

1.F)

```
df <- data.frame()

for (i in 2:100){
    df <- rbind(df, c(i, as.numeric(ci(gev_fit, type = "return.level", return.period = i)[2]), as.numeri
}

df %<>% setNames(c("Year", "GEV_est", "GPD_est")) %>% pivot_longer(cols = c(GEV_est, GPD_est), names_to


return_fig <-
    df %>%
        ggplot(aes(x = Year, y = Estimates, color = Method)) +
        geom_line() +
        labs(title = "Return Levels for Each Model",
        x = "Year",
        y = "Estimate",
        color = "Model") +
        theme_bw() +
        theme(plot.title = element_text(size = 20, hjust = 0.5),
            axis.title = element_text(size = 20),
            axis.text.x = element_text(size = 15, angle = 90),
            axis.text.y = element_text(size = 15))


return_fig
```


Return Levels for Each Model
Estimate
25
20
0
25
50
75
100
Year
Model
GEV_est
GPD_est