

OutputComplete.R

2025-02-21

Setup

```
knitr::opts_chunk$set(echo = TRUE)
source('IHA/IHAPackages.r') # Inputs necessary packages
IHAPackages()

## Loading required package: tidyverse

## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
## v dplyr      1.1.4      v readr      2.1.5
## v forcats    1.0.0      v stringr   1.5.1
## v ggplot2    3.5.1      v tibble    3.2.1
## v lubridate  1.9.3      v tidyr     1.3.1
## v purrr      1.0.2
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
## Loading required package: IHA
##
## Loading required package: zoo
##
##
## Attaching package: 'zoo'
##
##
## The following objects are masked from 'package:base':
##
##      as.Date, as.Date.numeric
##
## Loading required package: dataRetrieval
## Warning: package 'dataRetrieval' was built under R version 4.3.3
## Loading required package: plyr
## -----
## You have loaded plyr after dplyr - this is likely to cause problems.
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:
## library(plyr); library(dplyr)
## -----
##
## Attaching package: 'plyr'
##
## The following objects are masked from 'package:dplyr':
##
```

```
##      arrange, count, desc, failwith, id, mutate, rename, summarise,
##      summarize
##
## The following object is masked from 'package:purrr':
##
##      compact
##
## Loading required package: exactRankTests
## Package 'exactRankTests' is no longer under development.
## Please consider using package 'coin' instead.
##
## Loading required package: eseis
## Loading required package: plot.matrix
## Loading required package: writexl
## Warning: package 'writexl' was built under R version 4.3.3
```

User Inputs

```
gauge_number <- '01042500' # USGS gauge code
gauge_name <- 'THE FORKS' # USGS gauge location name

# Dates for analysis
start_date <- as.Date('1903-10-01') # Date when gauge started recording
break1_date <- as.Date('1952-06-01') # Date of end of time 1 and start of time 2
break2_date <- as.Date('1954-06-01') # Date of end of time 2 and start of time 3
end_date <- as.Date('2025-03-07') # Current date or end of gauge record
                                   # Can be set as current date is streamgauge is
                                   # ongoing

#Stats output location
stats_export_path =
  '~/Library/CloudStorage/OneDrive-BowdoinCollege/Bowdoin/Honors/DAMS Outputs'

# Years split into two different time periods
inc_yr1 <- format(start_date, '%Y')
inc_yr2 <- format(break1_date, '%Y')
inc_yr3 <- format(break2_date, '%Y')
inc_yr4 <- format(end_date, '%Y')
```

download streamgaugue data and subset by date

Daily flow to zoo for IHA

IHA package functions

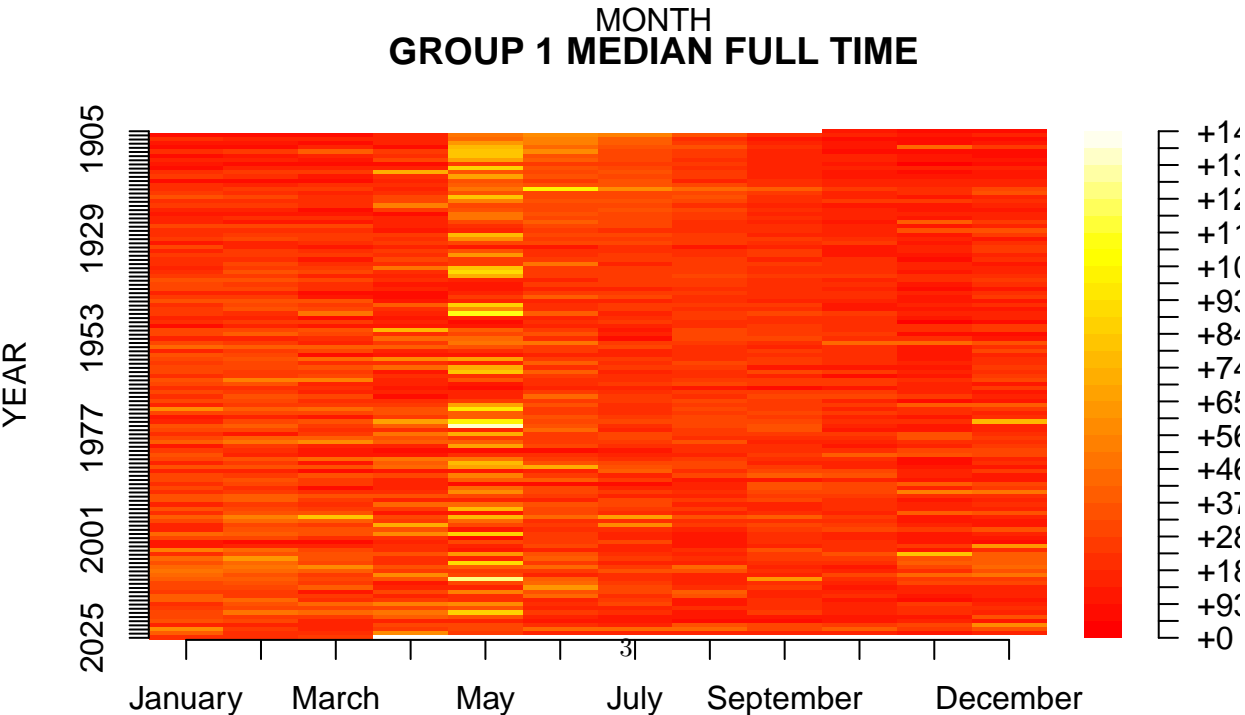
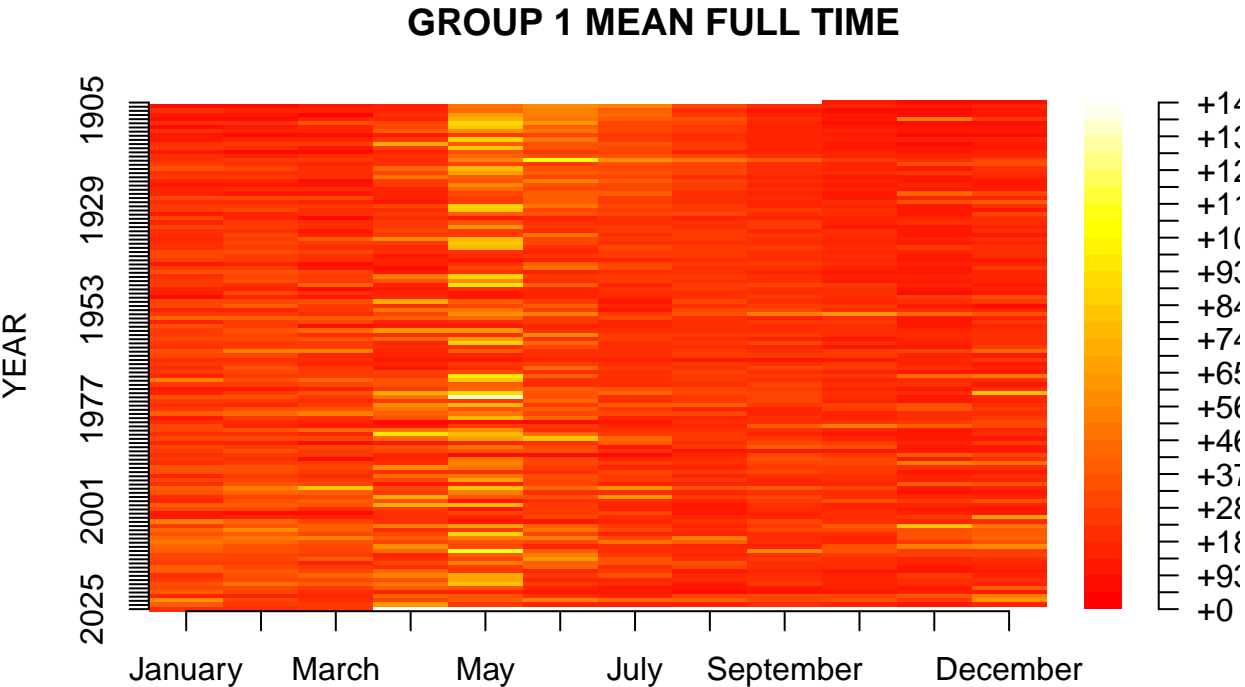
Field Measurements Functions

Annual Measure of Center Functions

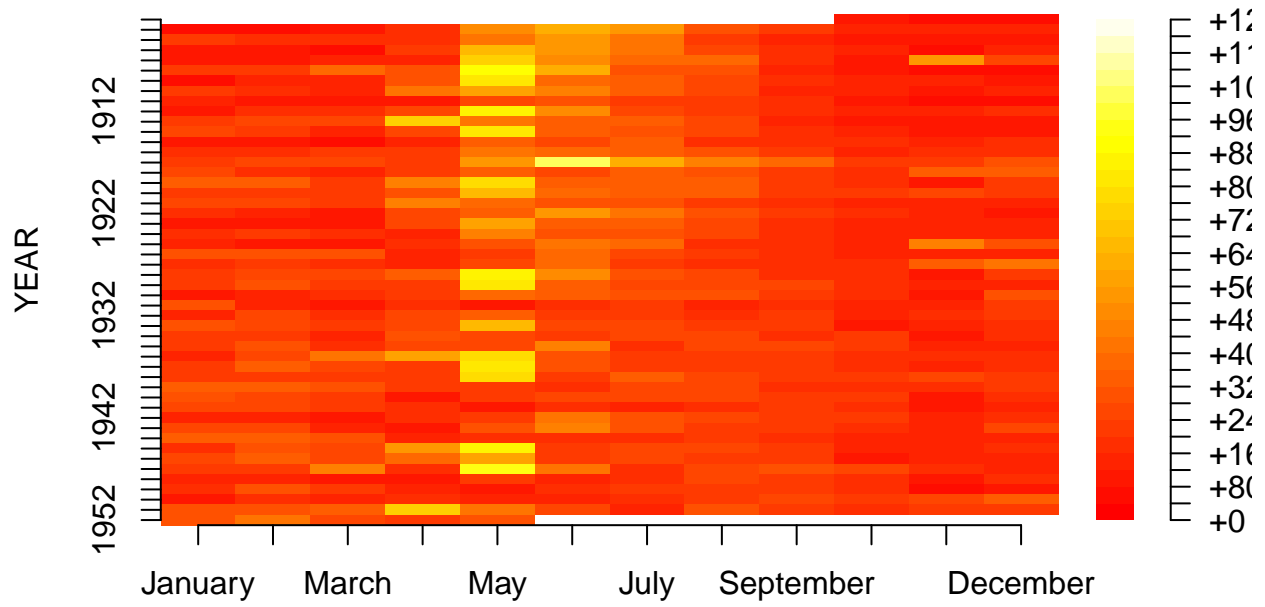
Annual Peak Flow Functions

Annual High Flow Functions

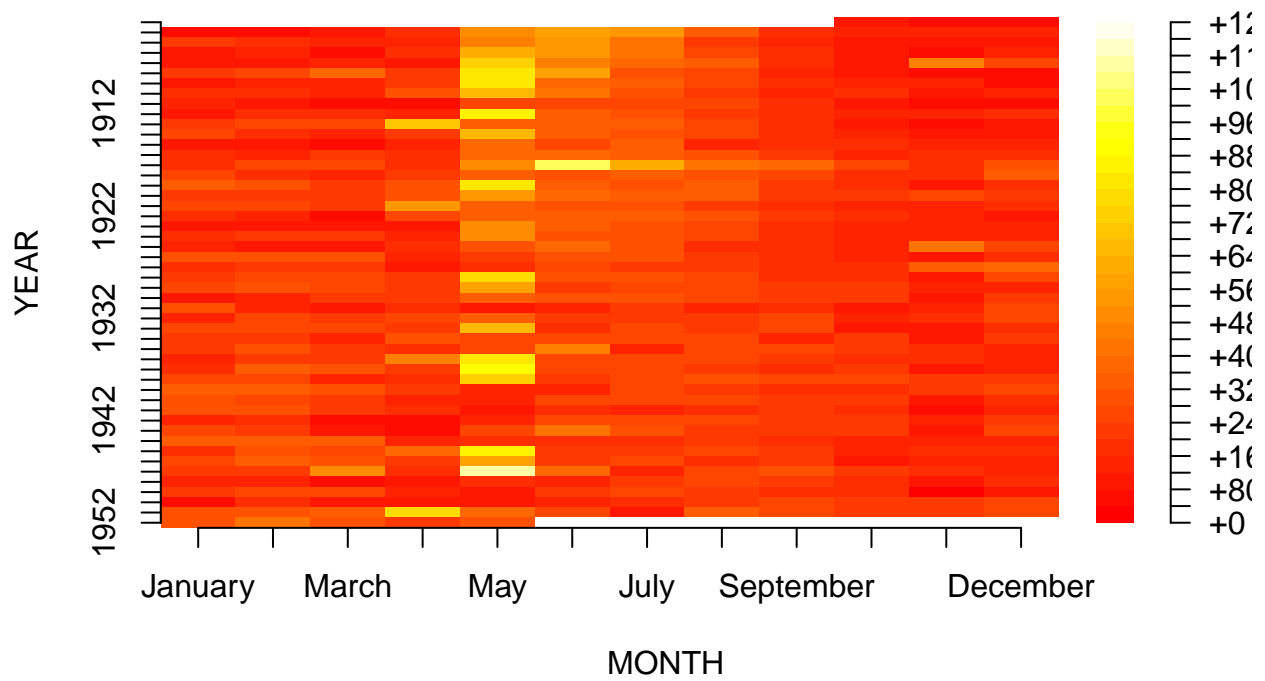
IHA Manipulation and Visualization



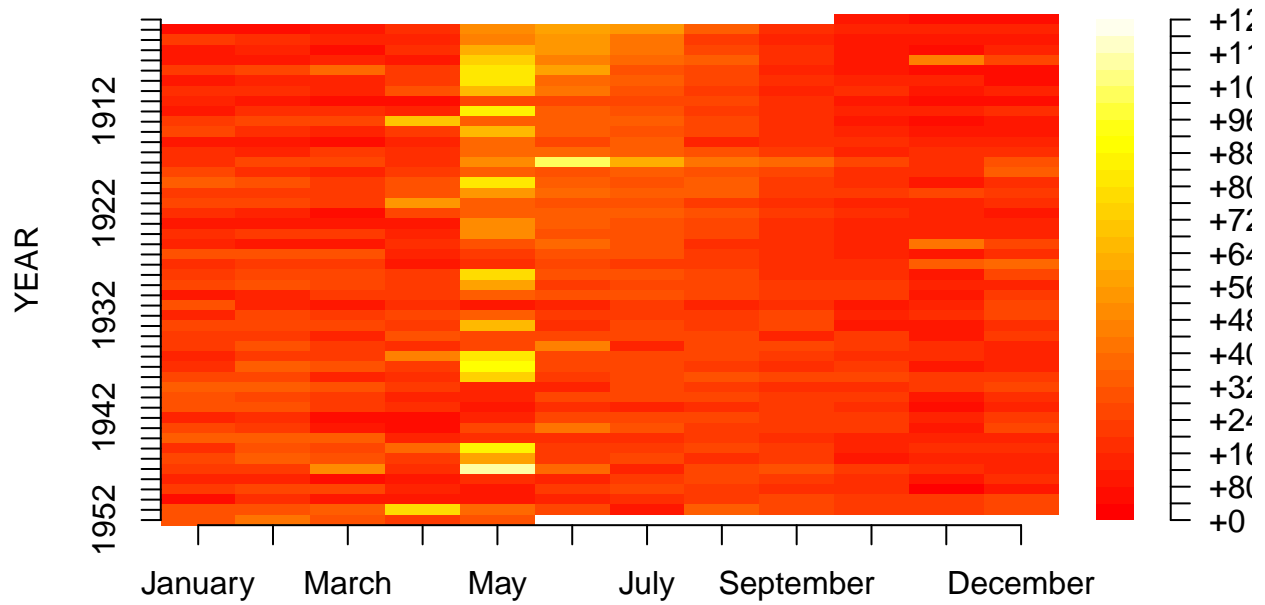
GROUP 1 MEAN TIME 1



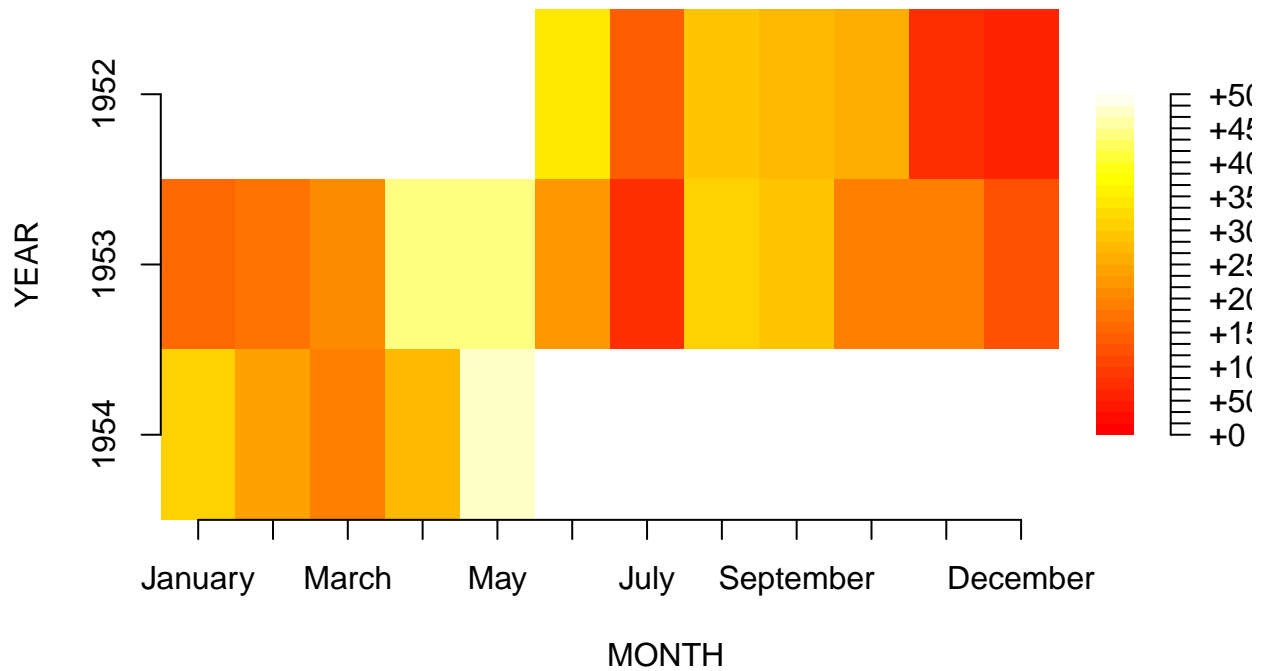
GROUP 1 MEDIAN TIME 1

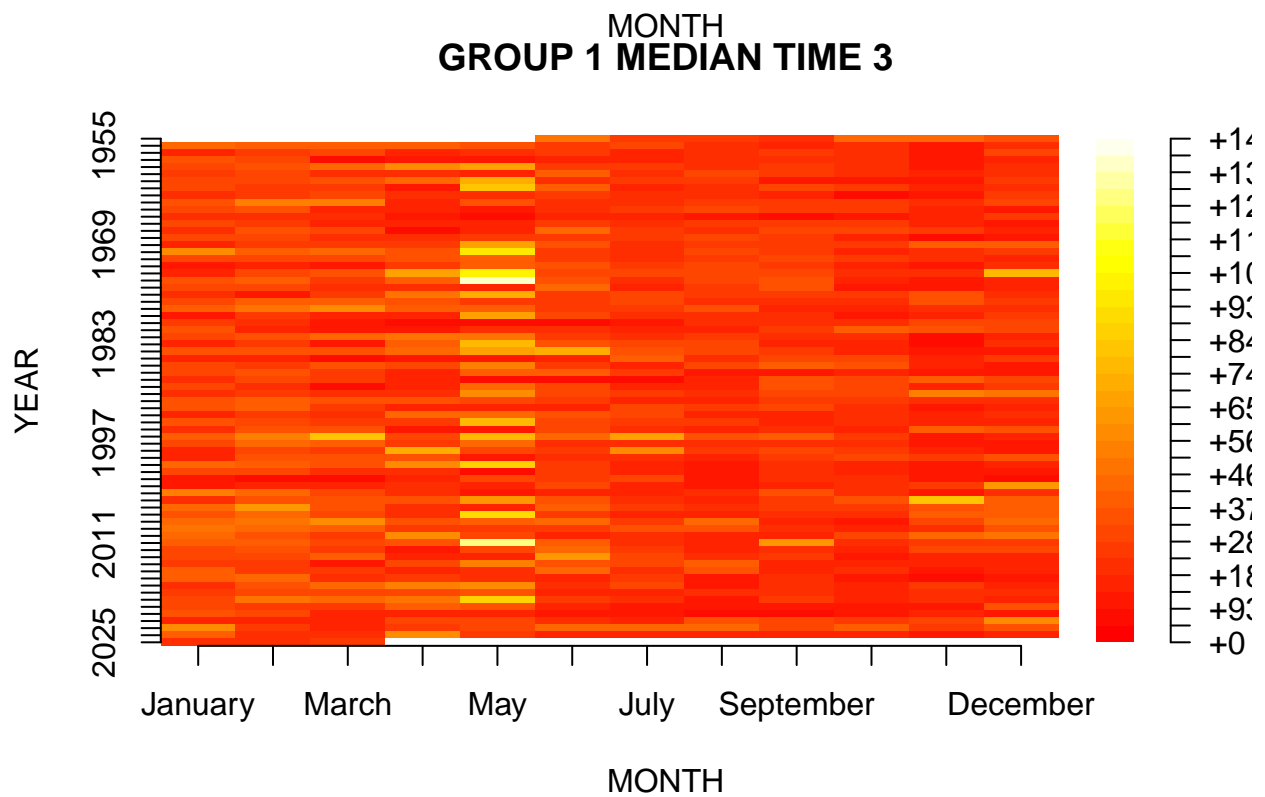
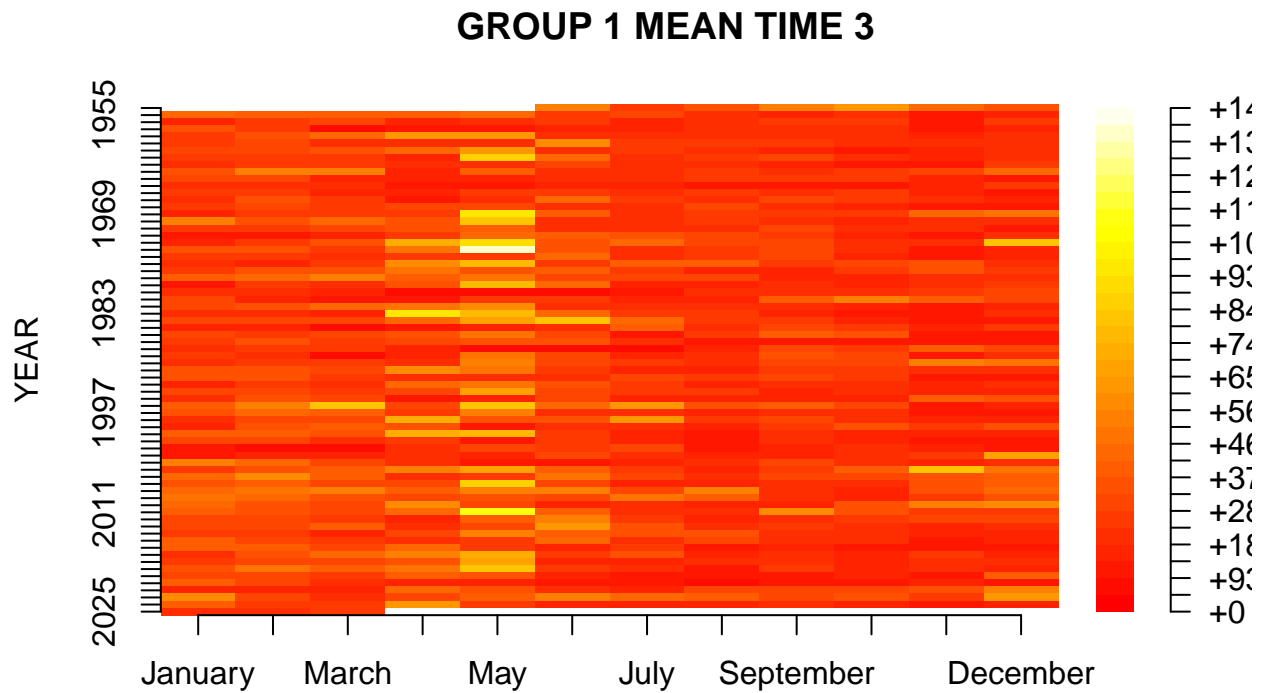


GROUP 1 MEAN TIME 2

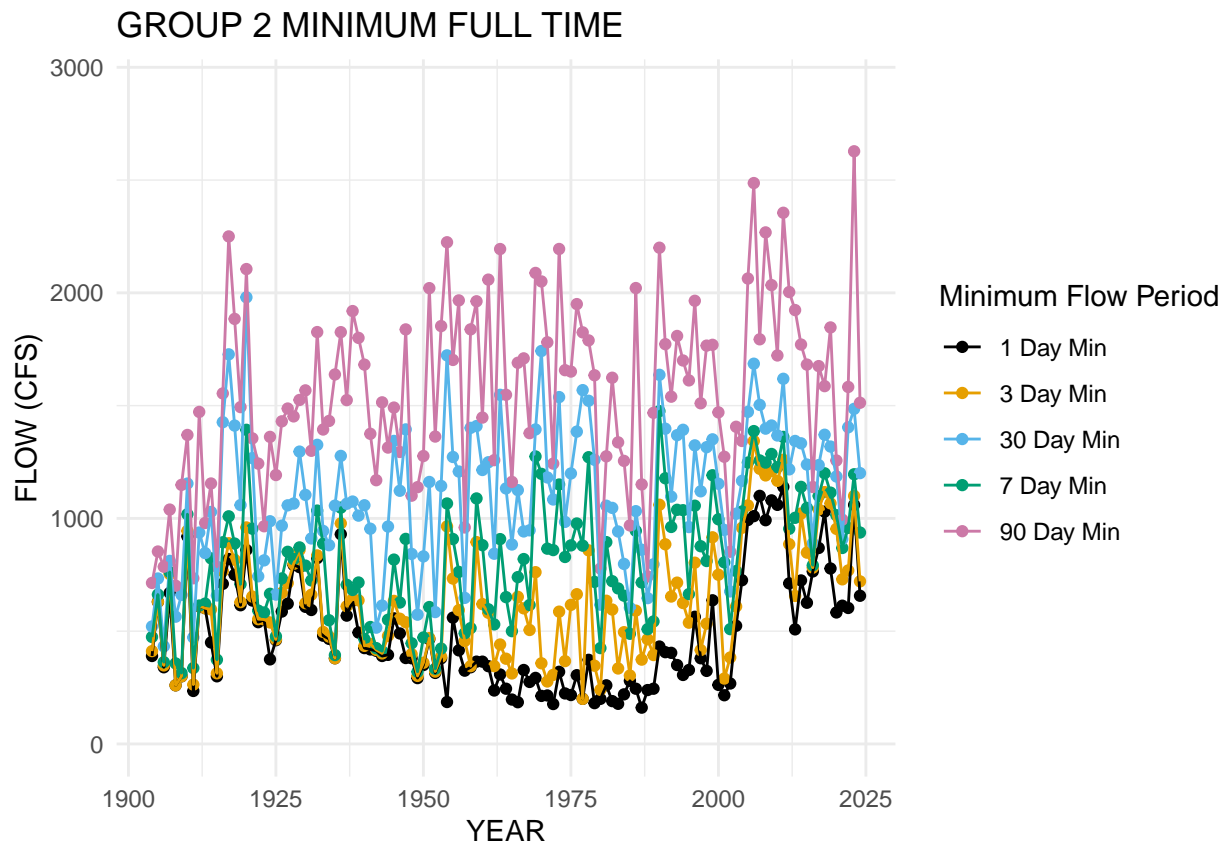


GROUP 1 MEDIAN TIME 2



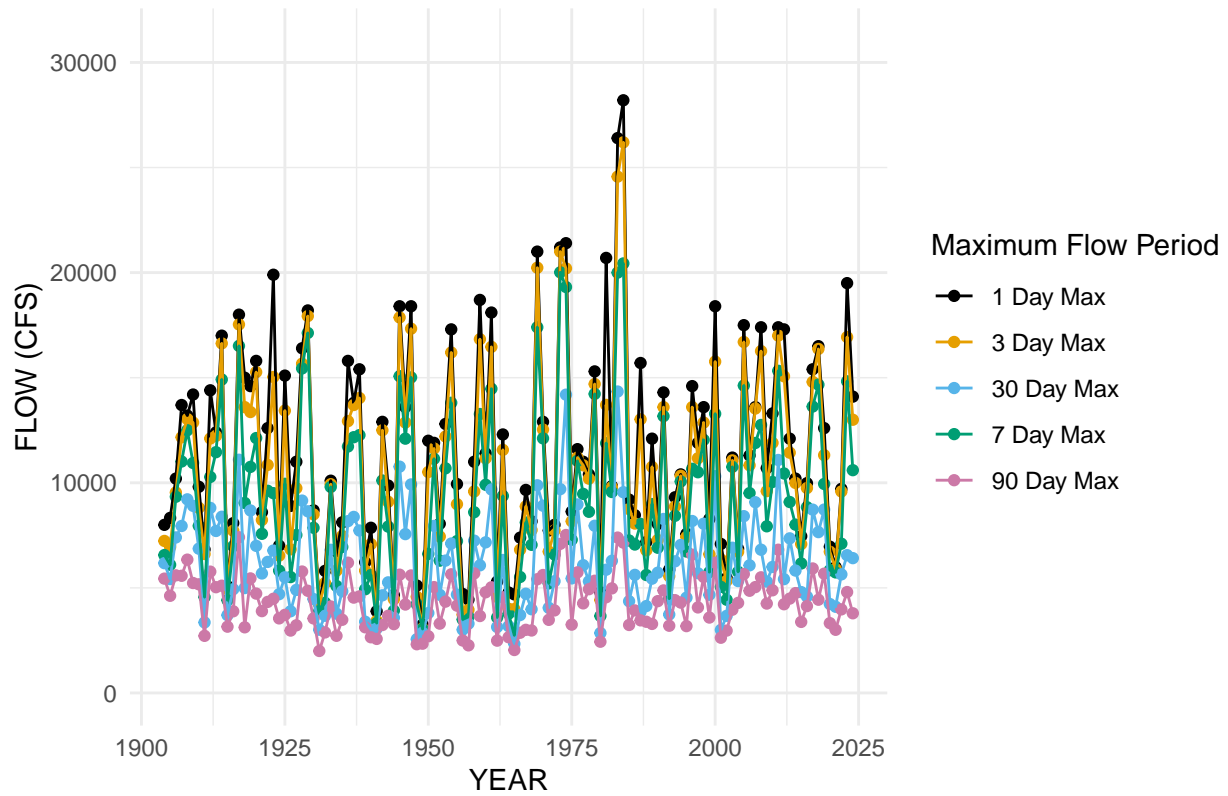


\$group2_min_full_plot

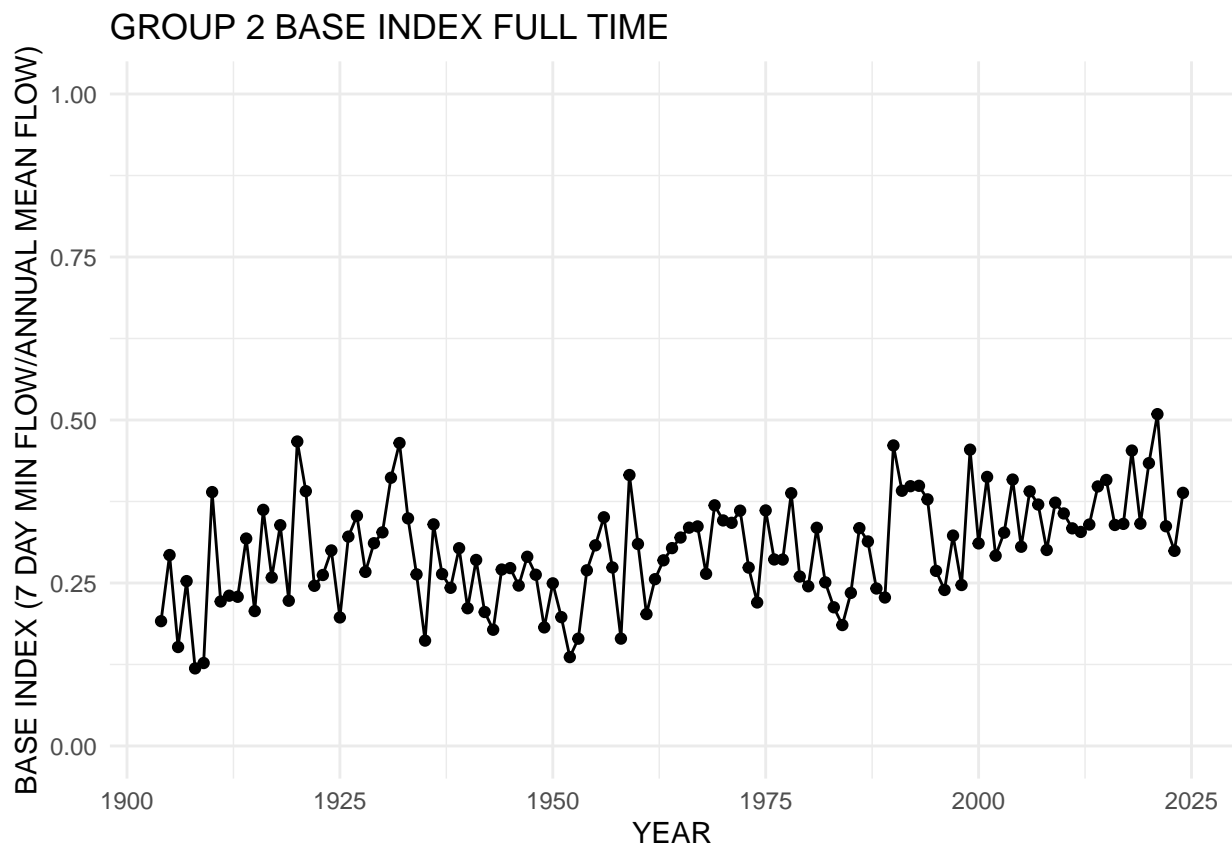


```
##  
## $group2_max_full_plot
```

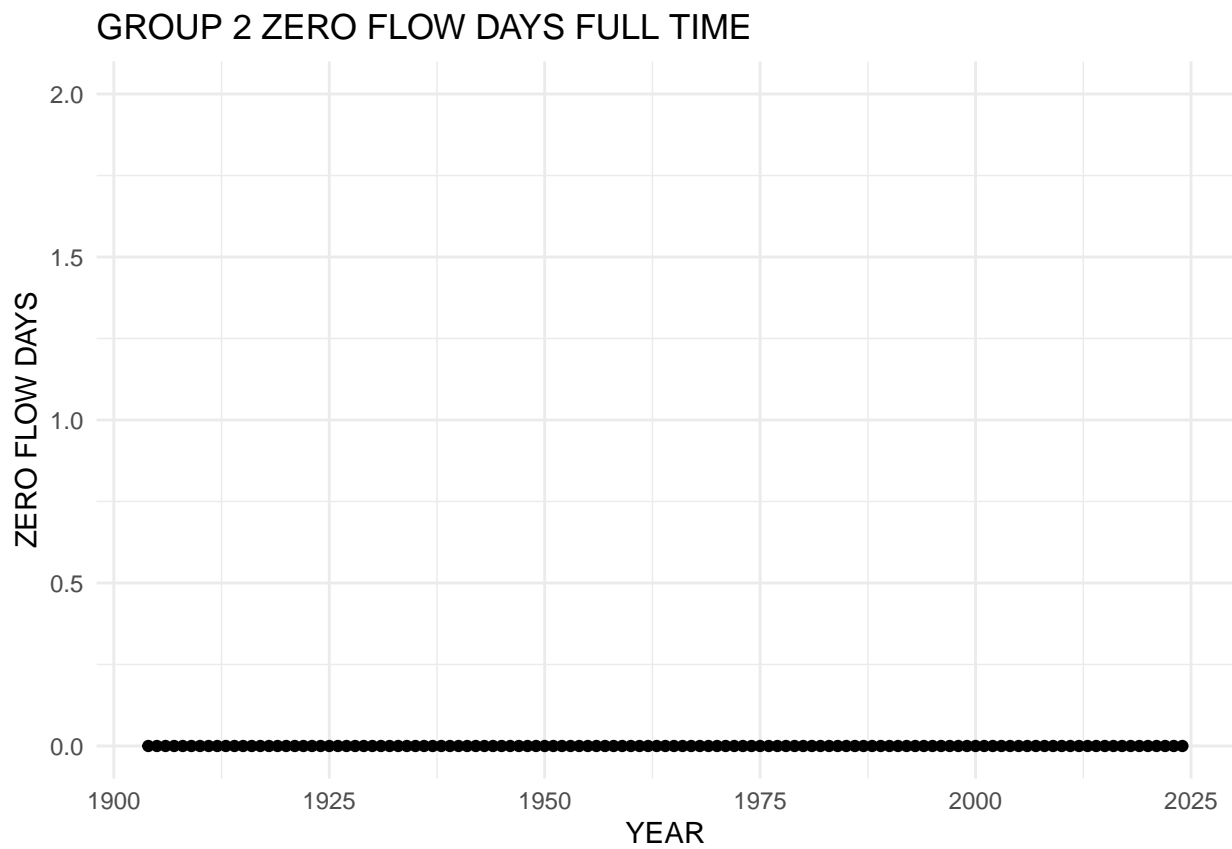
GROUP 2 MAXIMUM FULL TIME



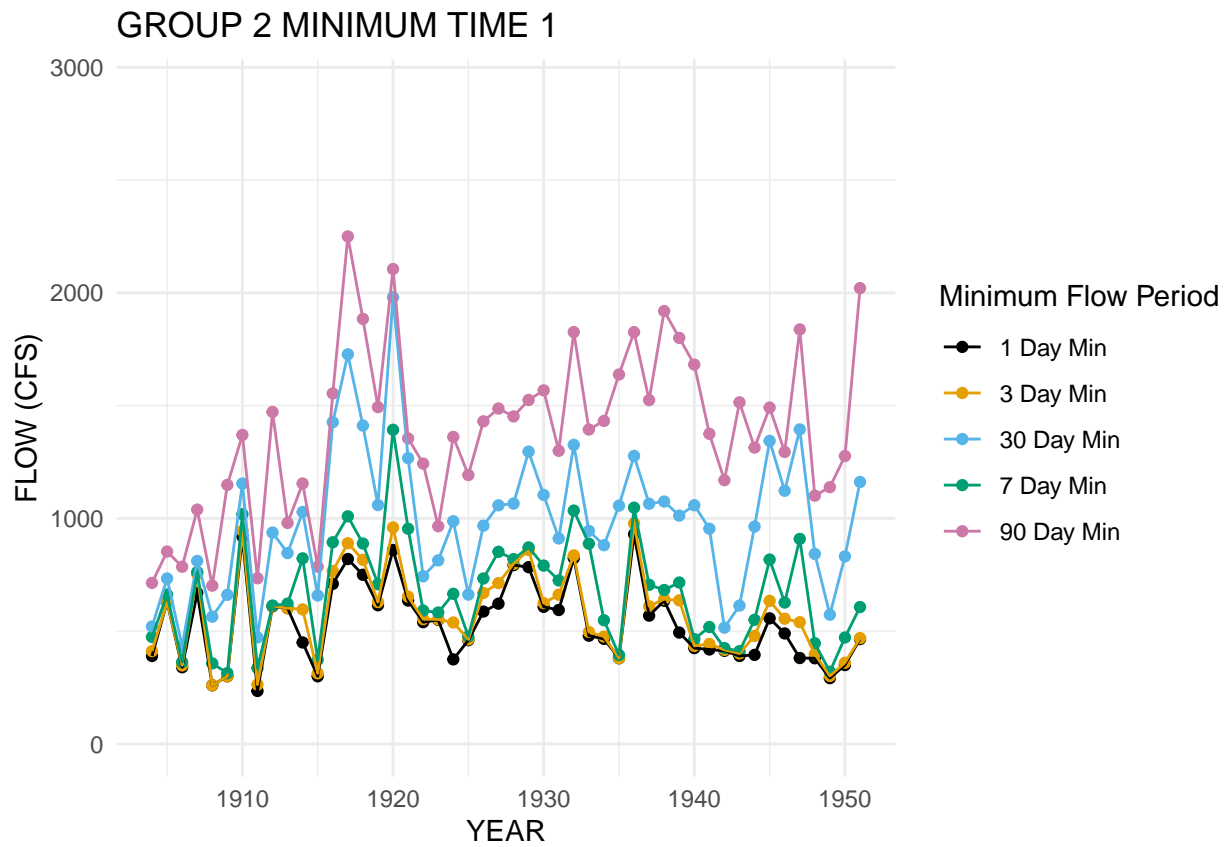
```
##  
## $group2_BI_full_plot
```

```
##  
## $group2_zero_full_plot
```

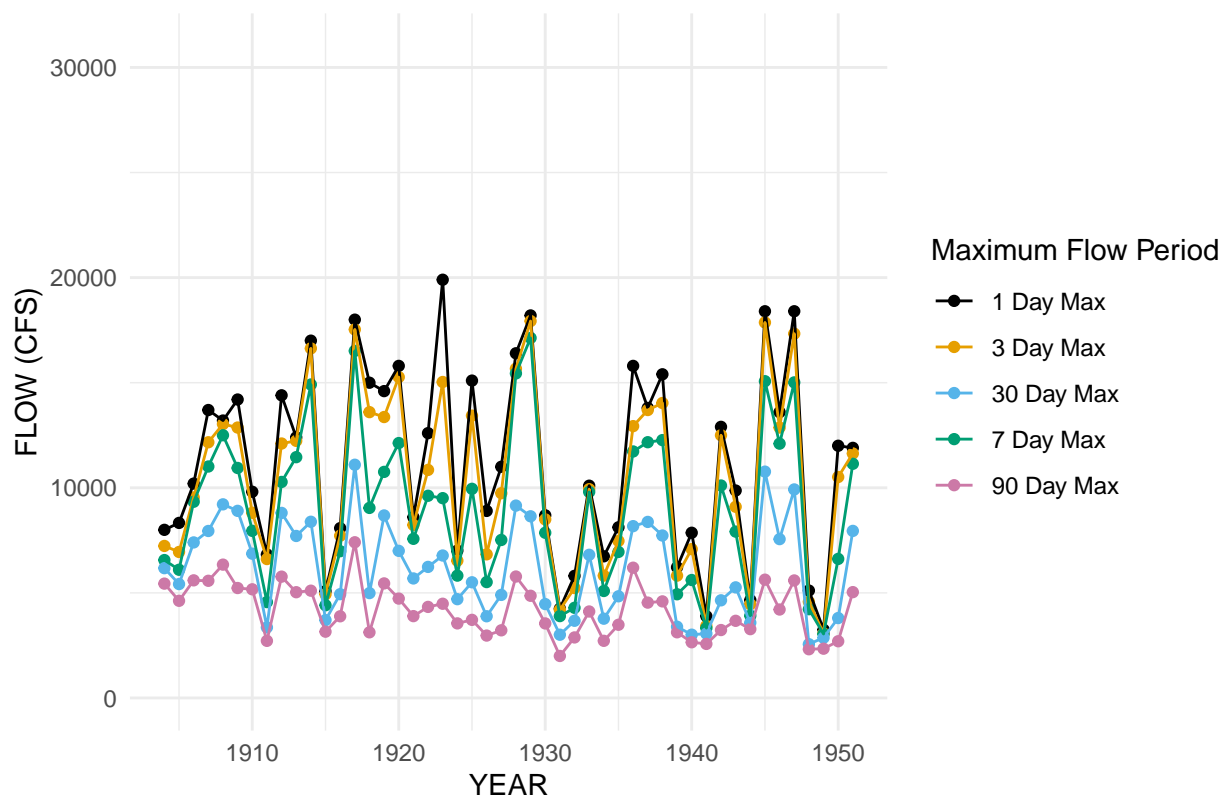


```
##  
## $group2_min_time1_plot
```



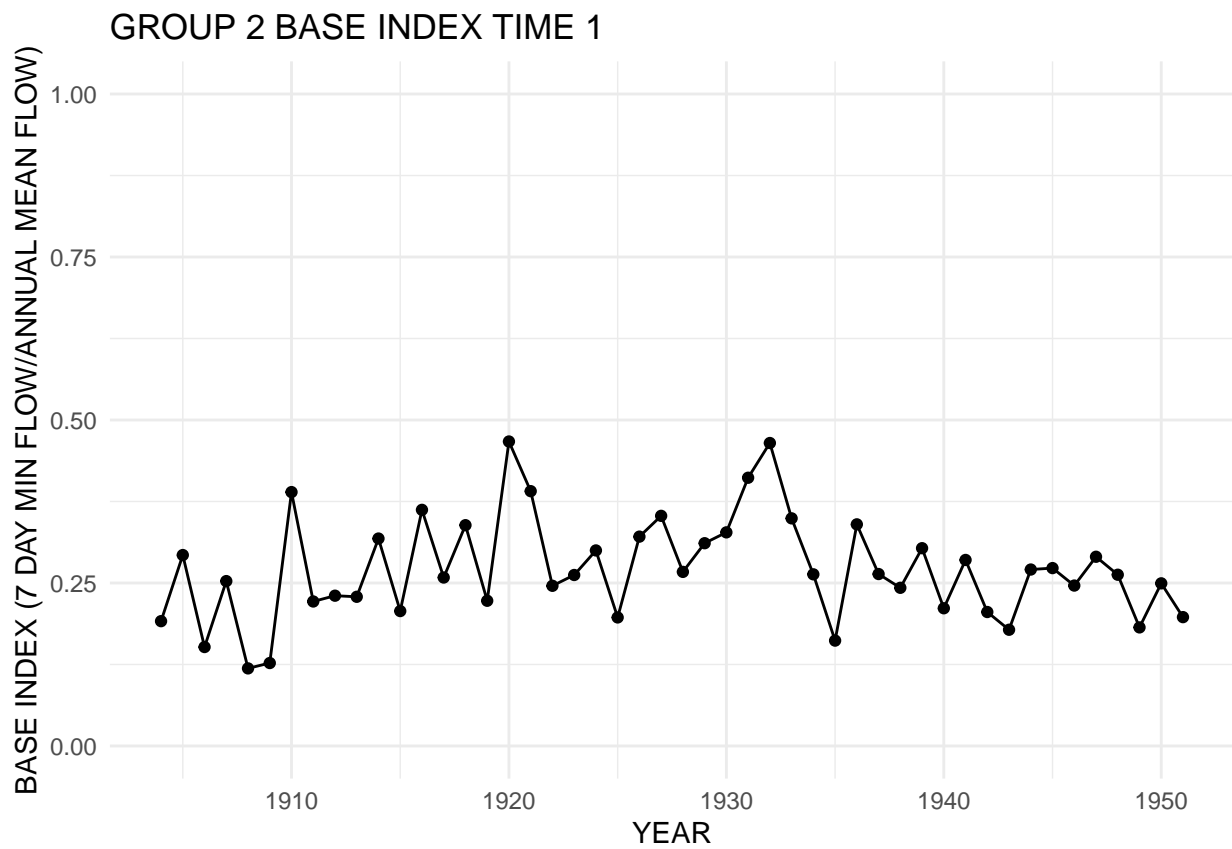
```
##  
## $group2_max_time1_plot
```

GROUP 2 MAXIMUM TIME 1

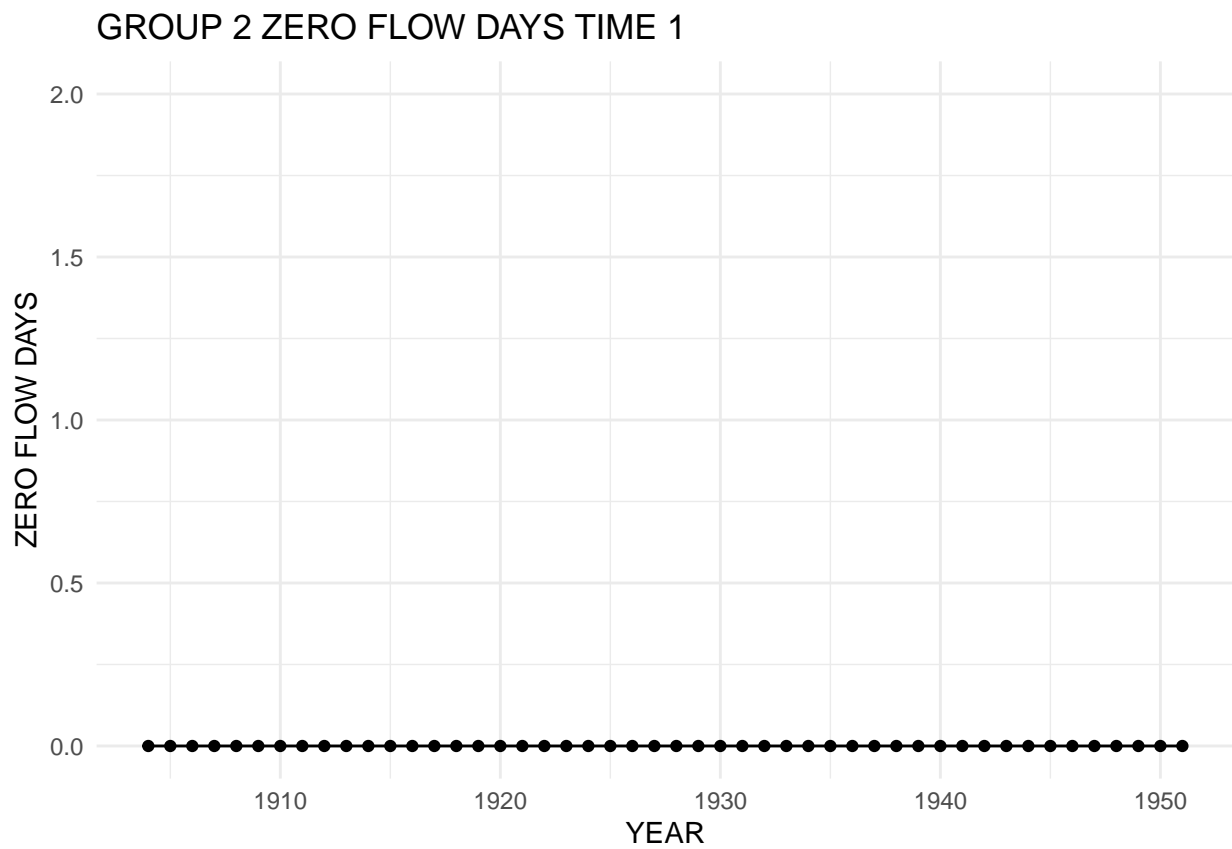


##

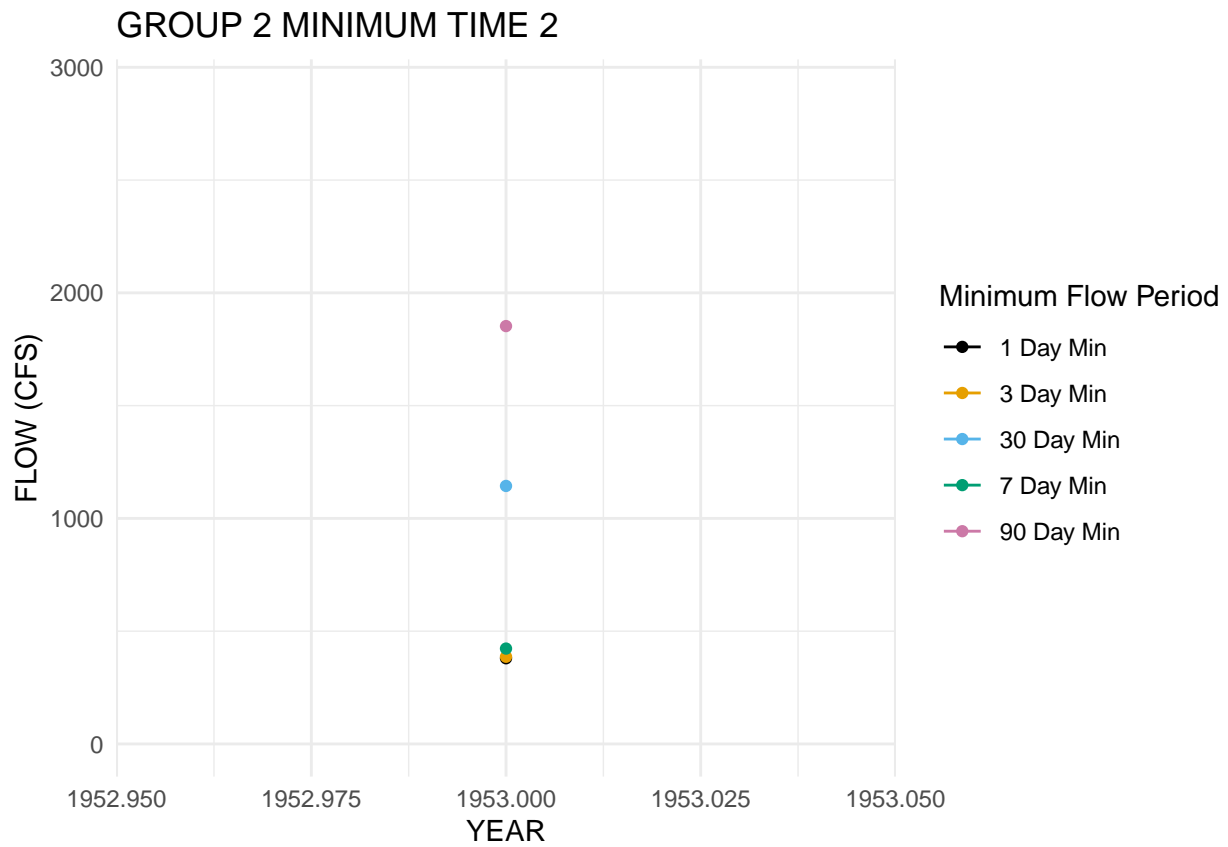
\$group2_BI_time1_plot



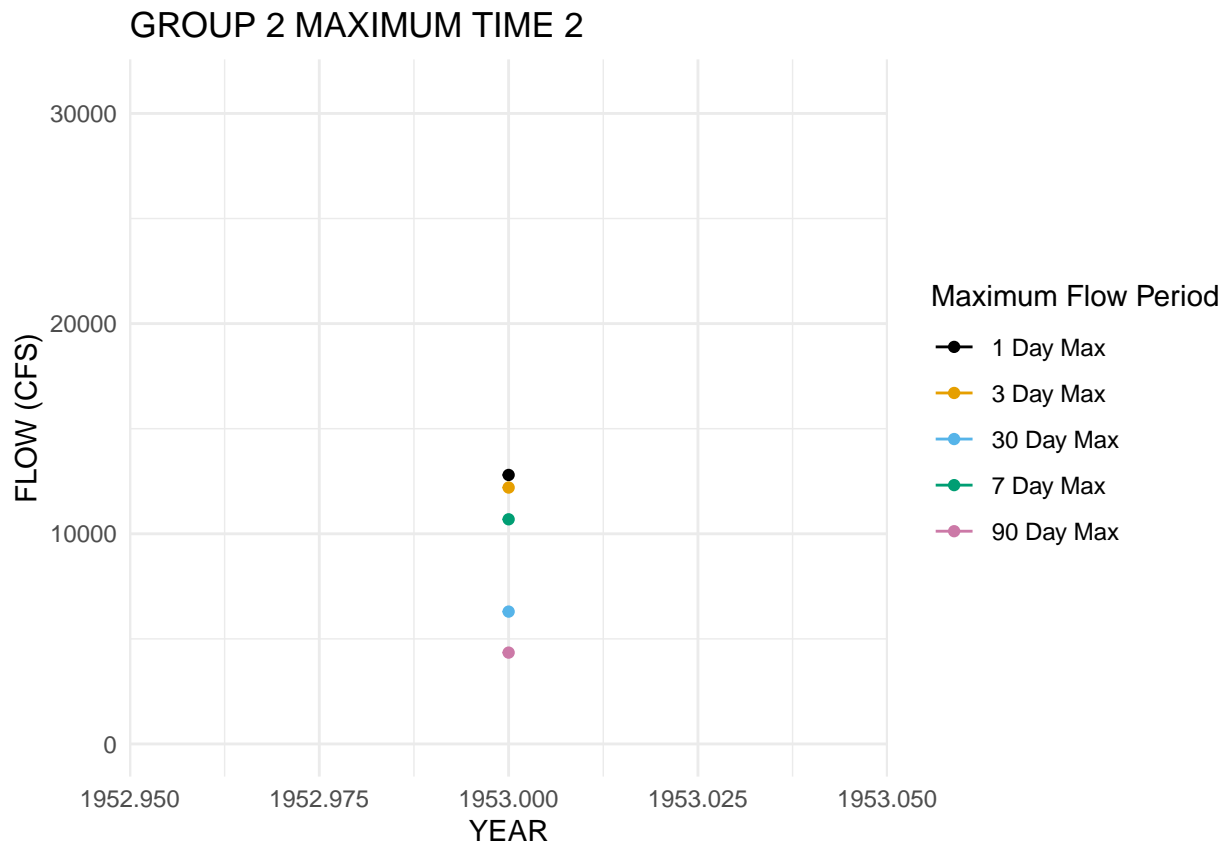
```
##  
## $group2_zero_time1_plot
```



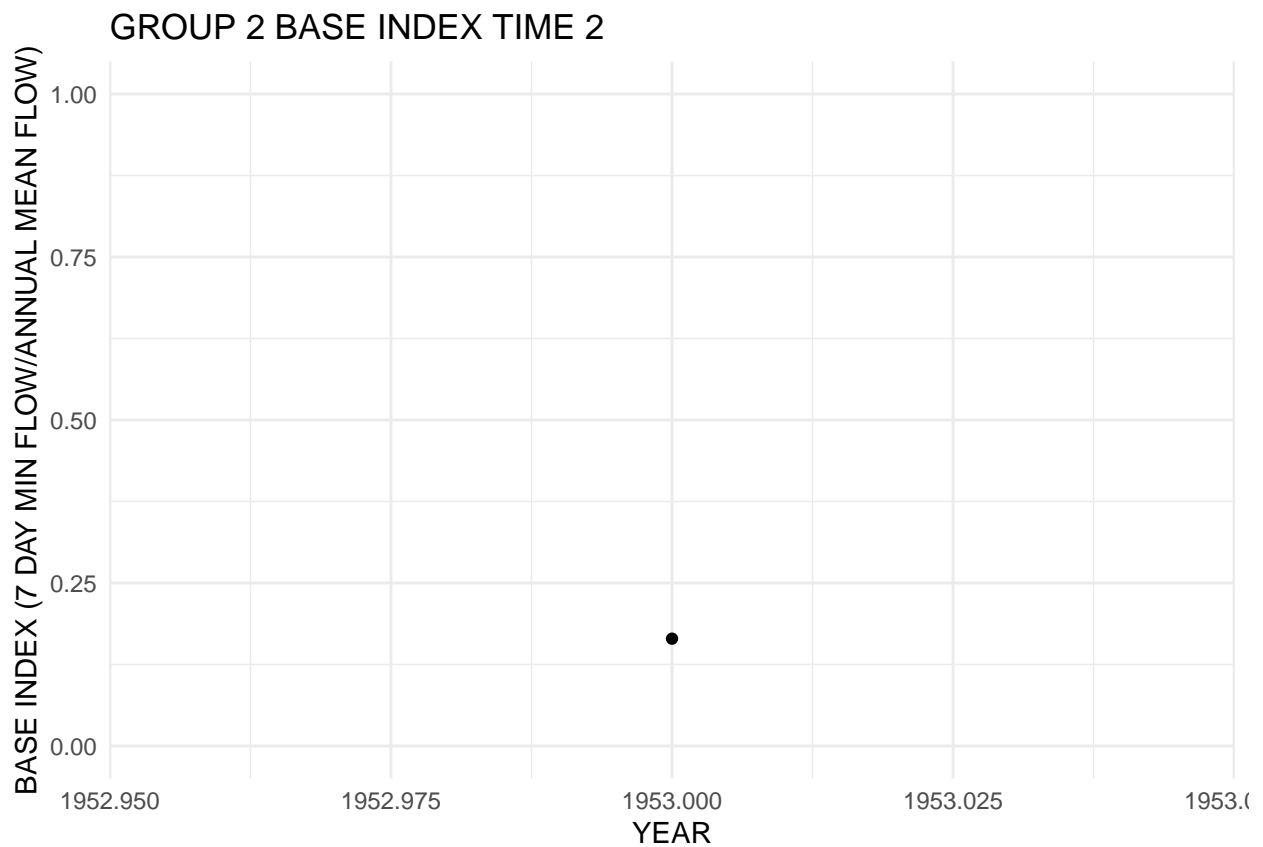
```
##  
## $group2_min_time2_plot  
## `geom_line()`: Each group consists of only one observation.  
## i Do you need to adjust the group aesthetic?
```



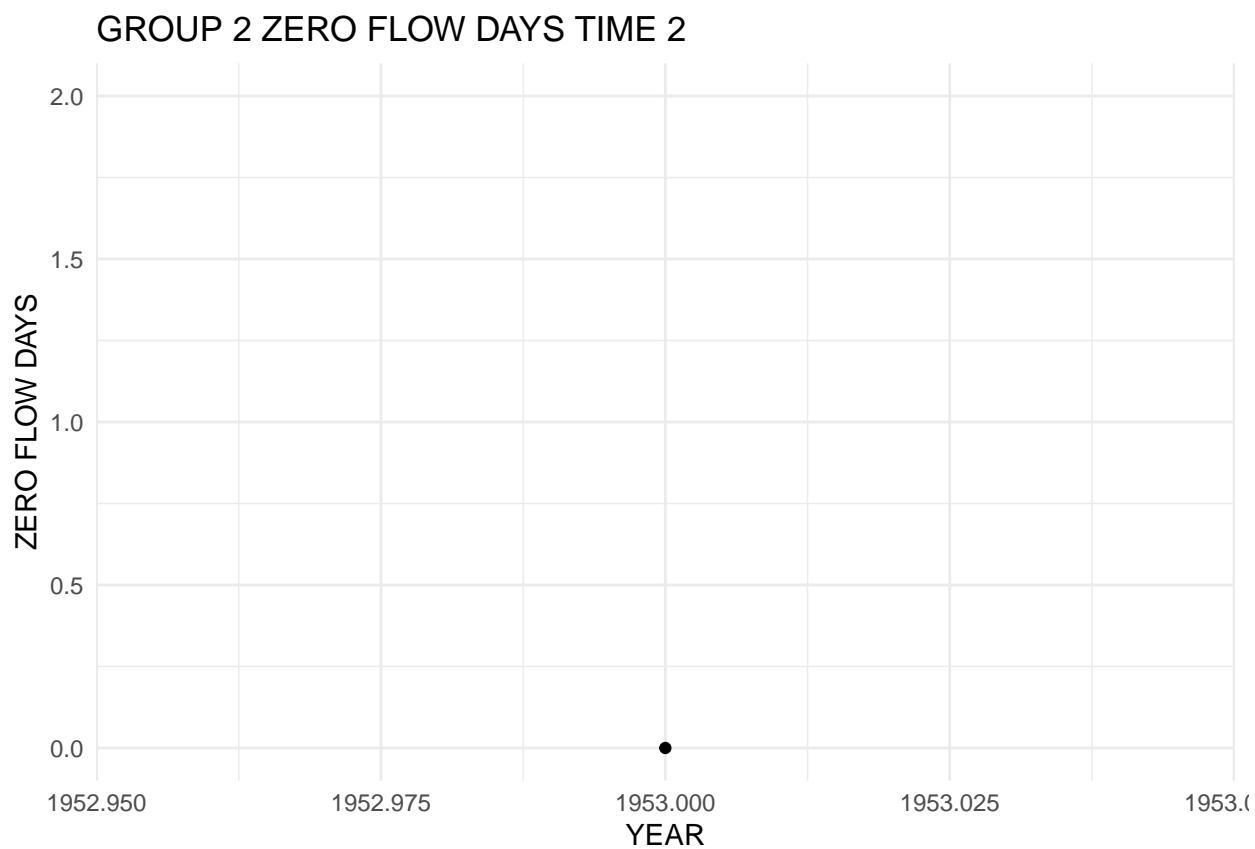
```
##  
## $group2_max_time2_plot  
  
## `geom_line()`: Each group consists of only one observation.  
## i Do you need to adjust the group aesthetic?
```



```
##  
## $group2_BI_time2_plot  
  
## `geom_line()`: Each group consists of only one observation.  
## i Do you need to adjust the group aesthetic?
```

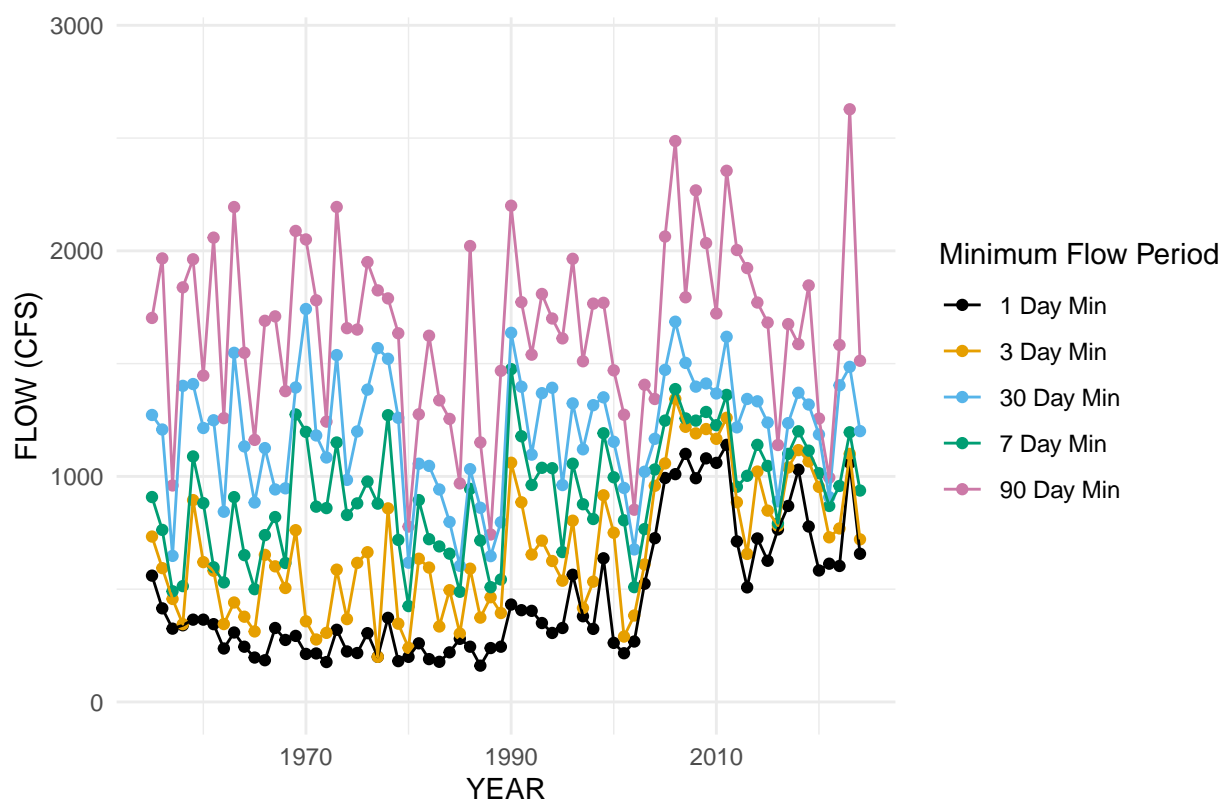



```
##  
## $group2_zero_time2_plot  
  
## `geom_line()`: Each group consists of only one observation.  
## i Do you need to adjust the group aesthetic?
```



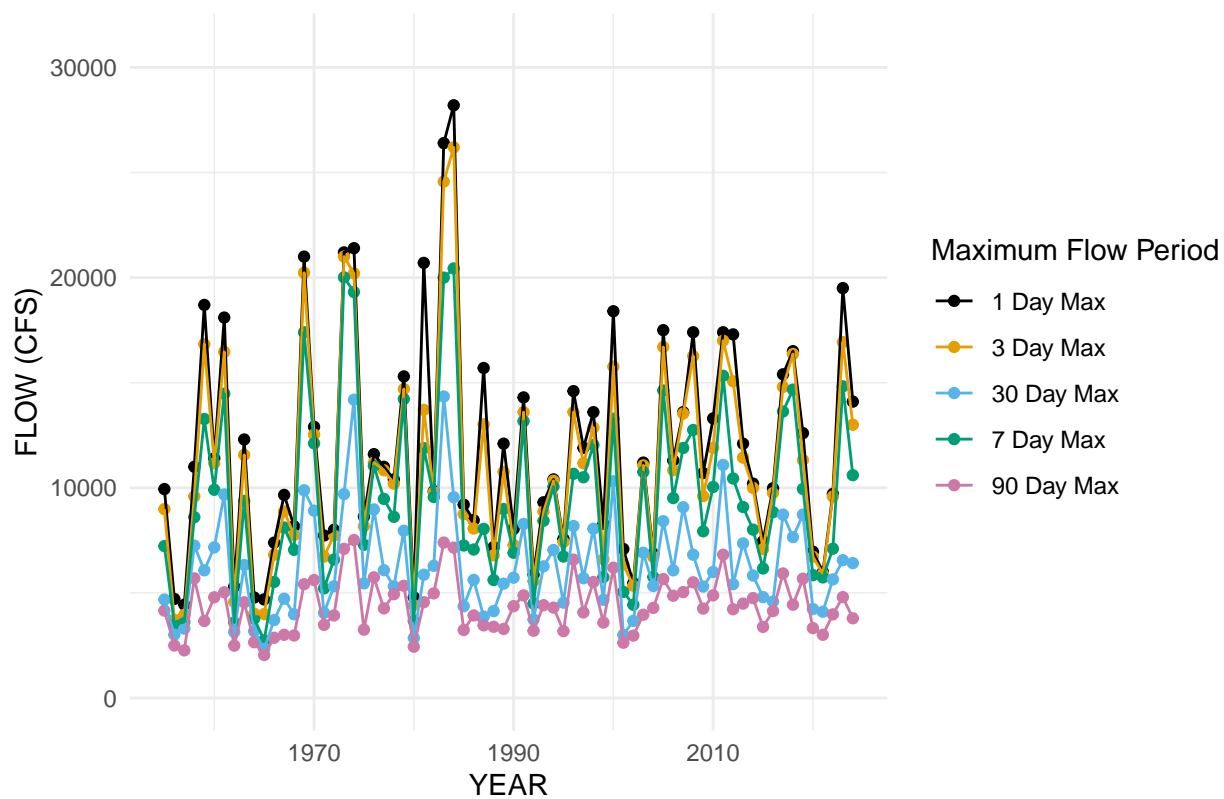
```
##  
## $group2_min_time3_plot
```

GROUP 2 MINIMUM TIME 3

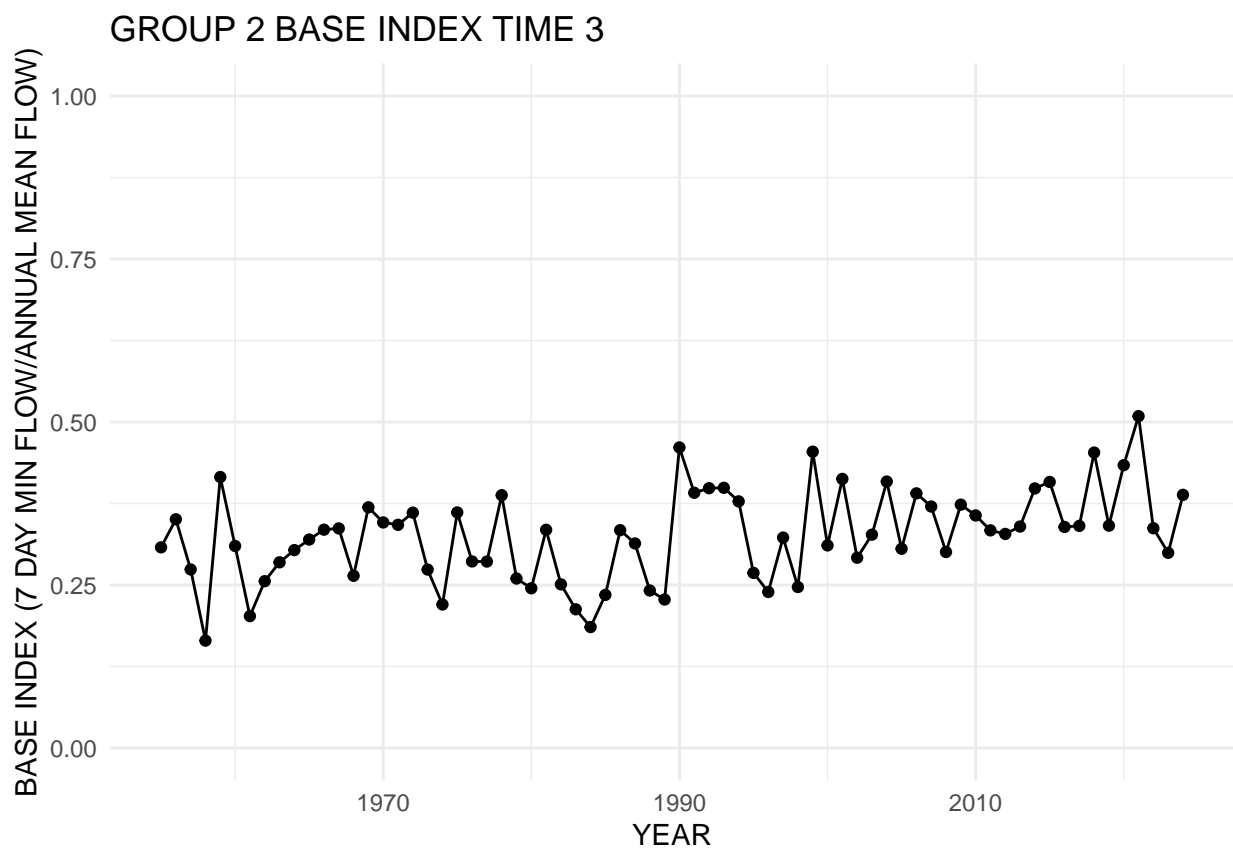


```
##
## $group2_max_time3_plot
```

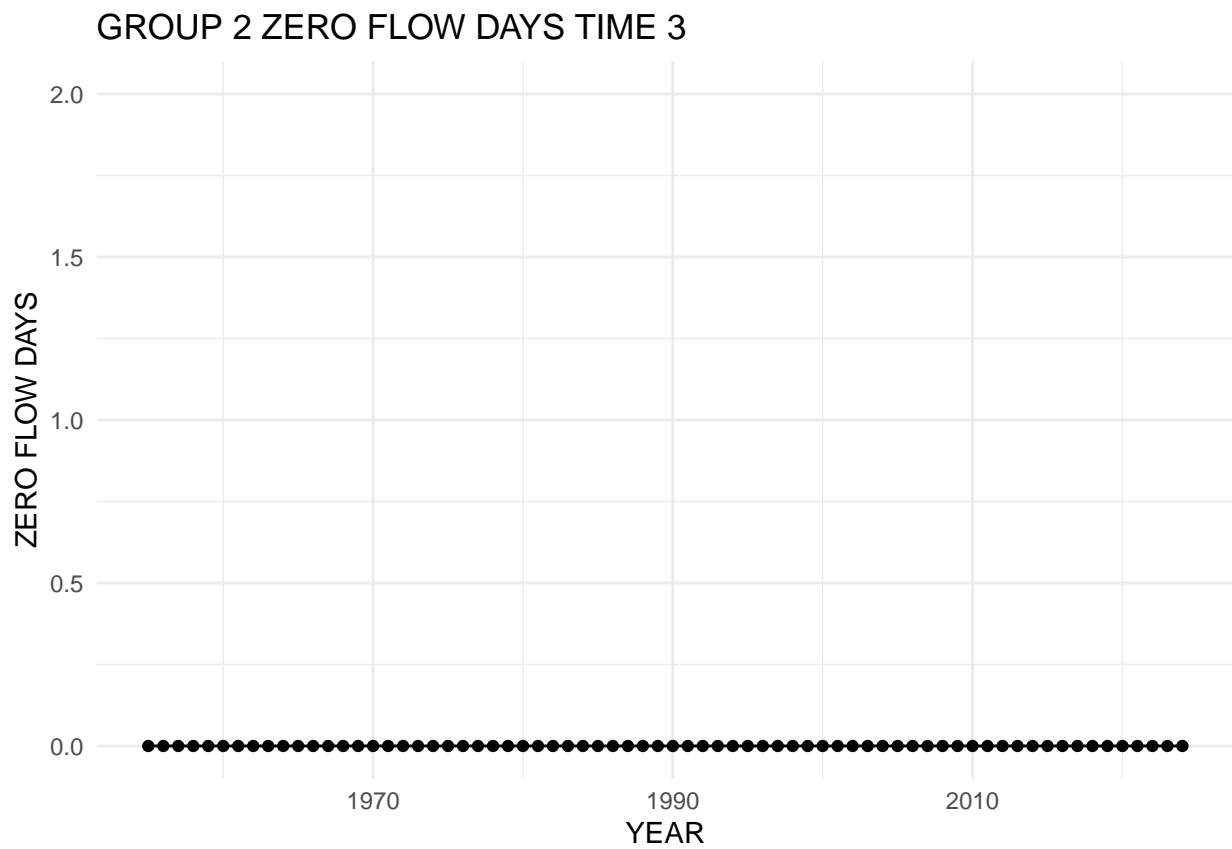
GROUP 2 MAXIMUM TIME 3



```
##  
## $group2_BI_time3_plot
```

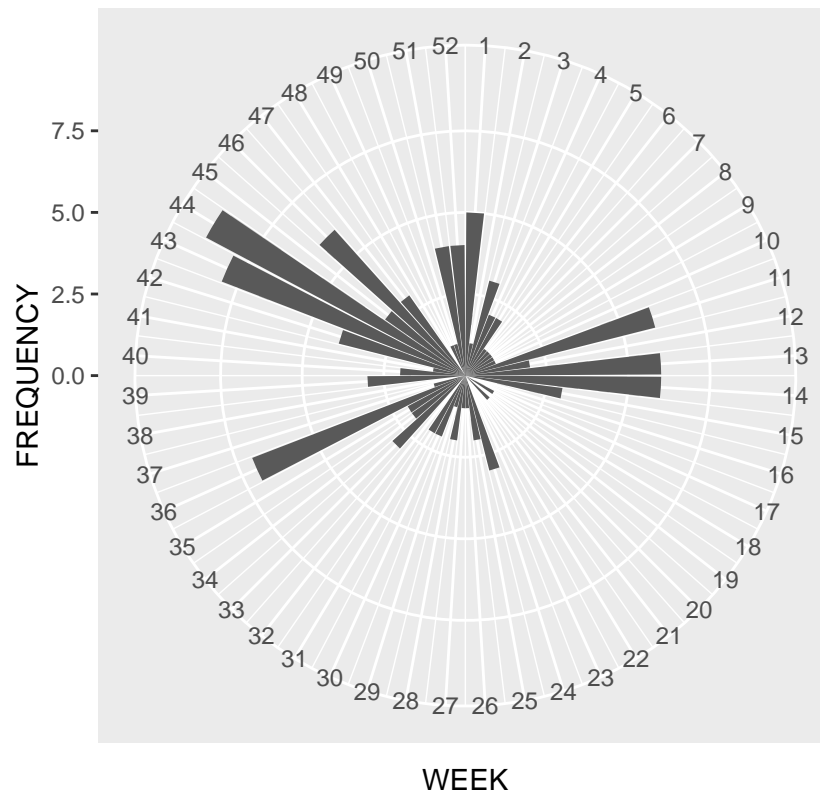


```
##  
## $group2_zero_time3_plot
```



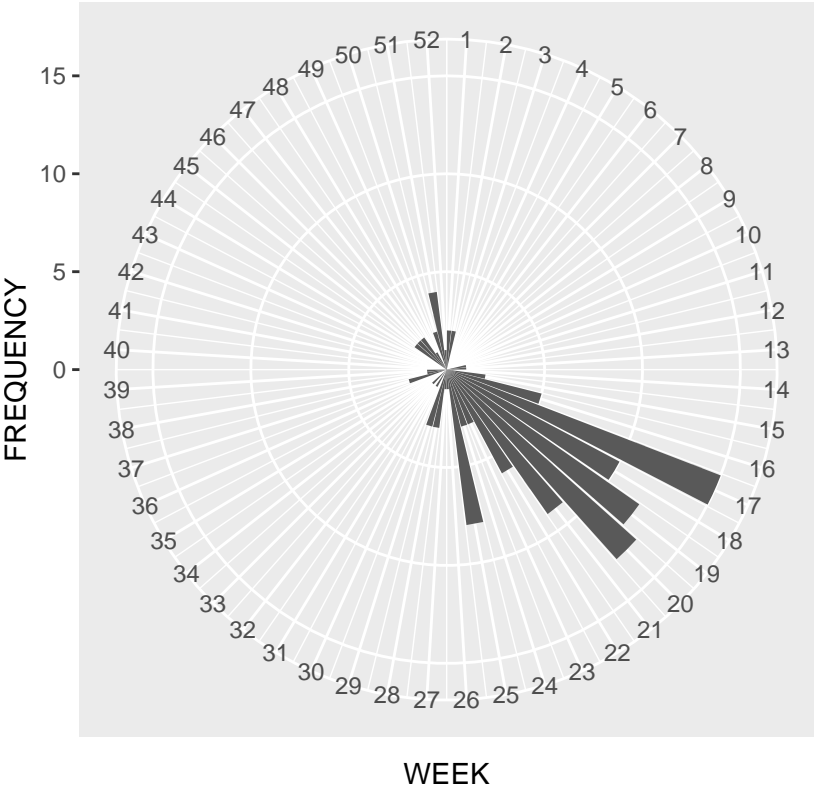
```
## $group3_min_full_week_plot
```

GROUP 3 MIN FULL TIME WEEK FREQUENCY

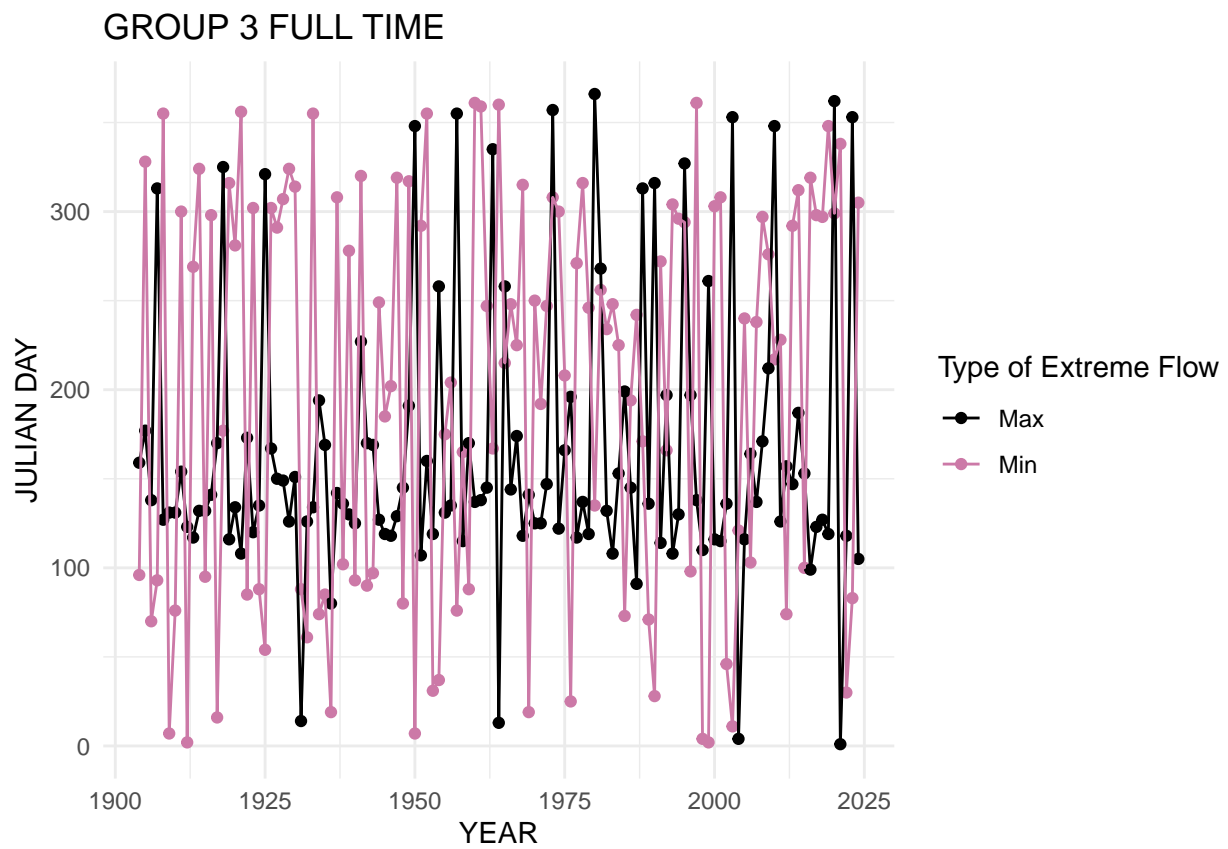


```
##
## $group3_max_full_week_plot
## Warning: Removed 1 row containing missing values or values outside the scale range
## (`geom_bar()`).
```

GROUP 3 MAX FULL TIME WEEK FREQUENCY

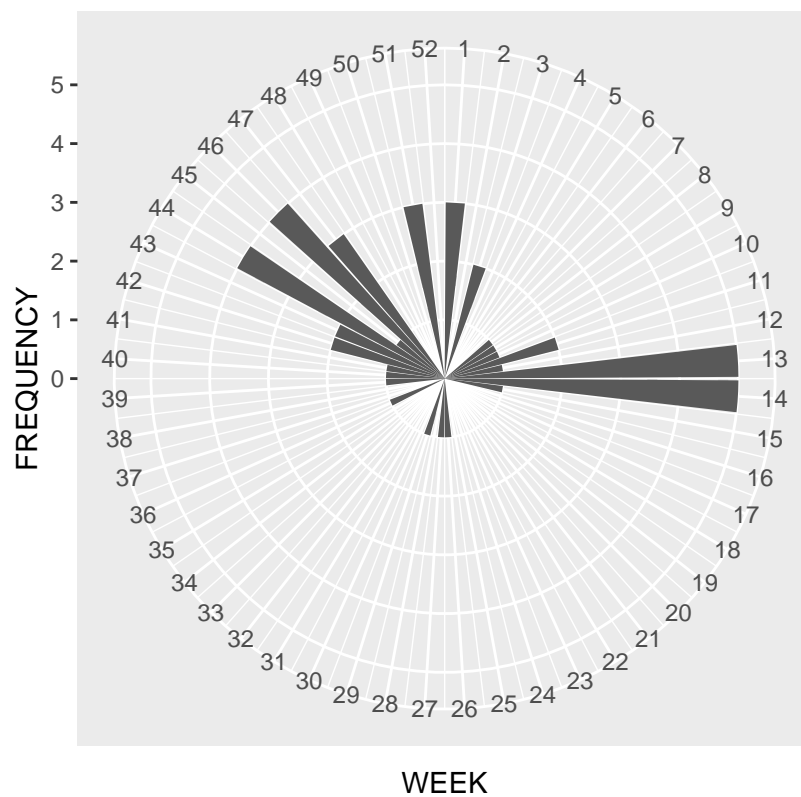


```
##  
## $group3_full_julian_day_plot
```

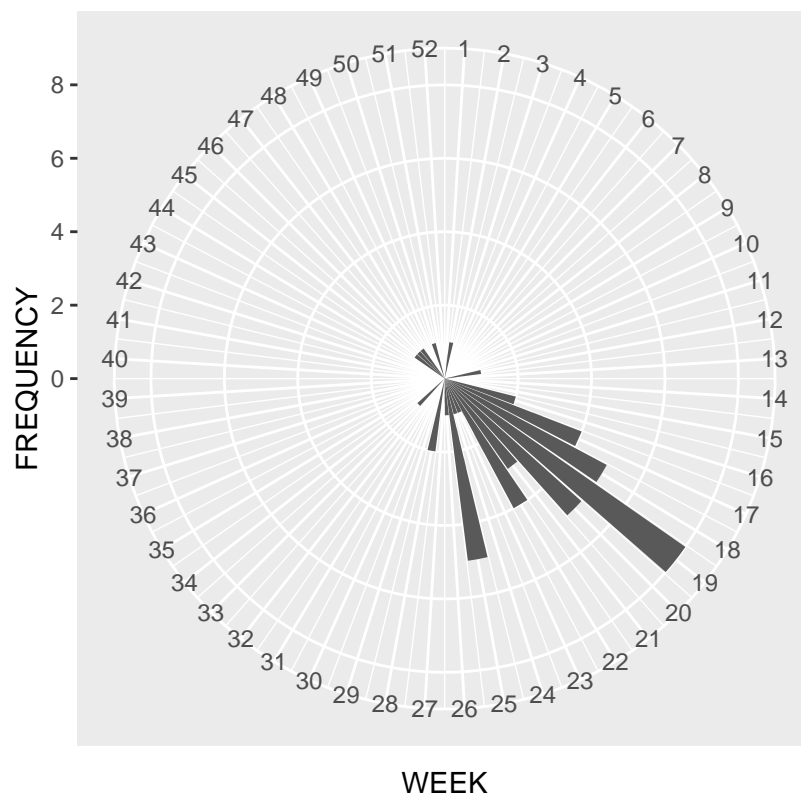
```
##  
## $group3_min_time1_week_plot
```

GROUP 3 MIN TIME 1 WEEK FREQUENCY

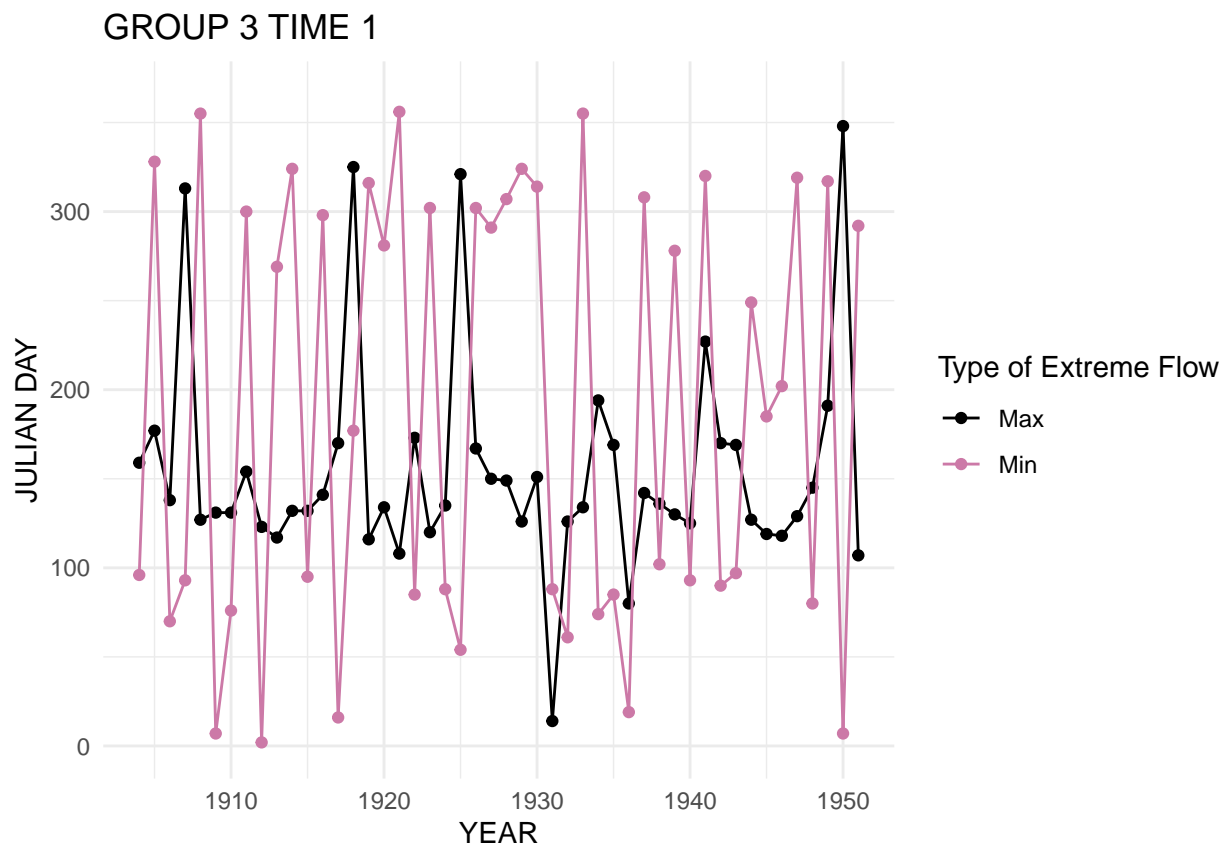


```
##
## $group3_max_time1_week_plot
```

GROUP 3 MAX TIME 1 WEEK FREQUENCY

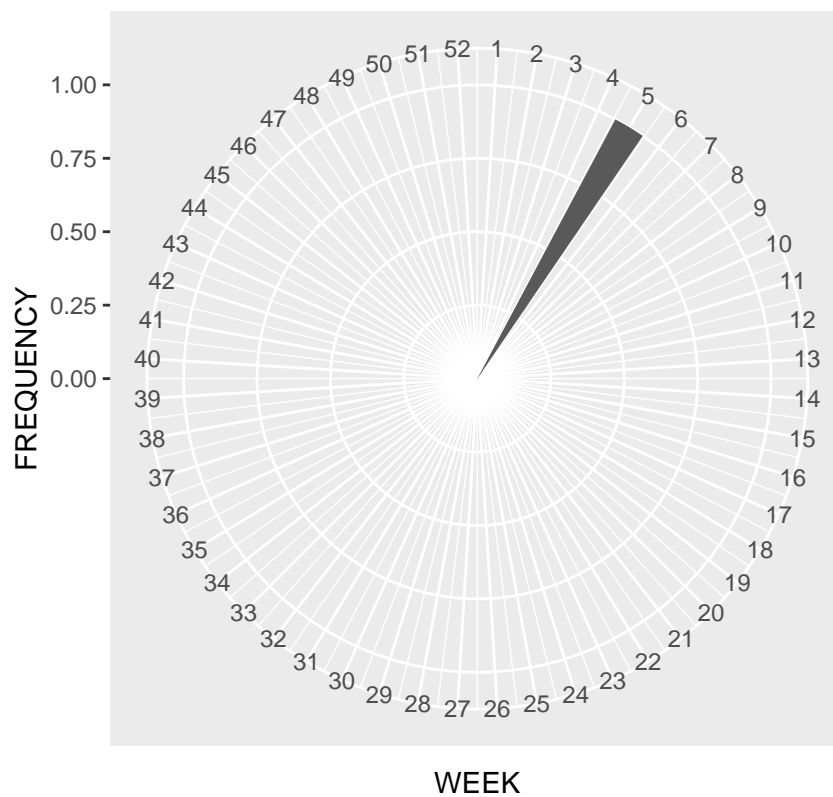


```
##
## $group3_time1_julian_day_plot
```



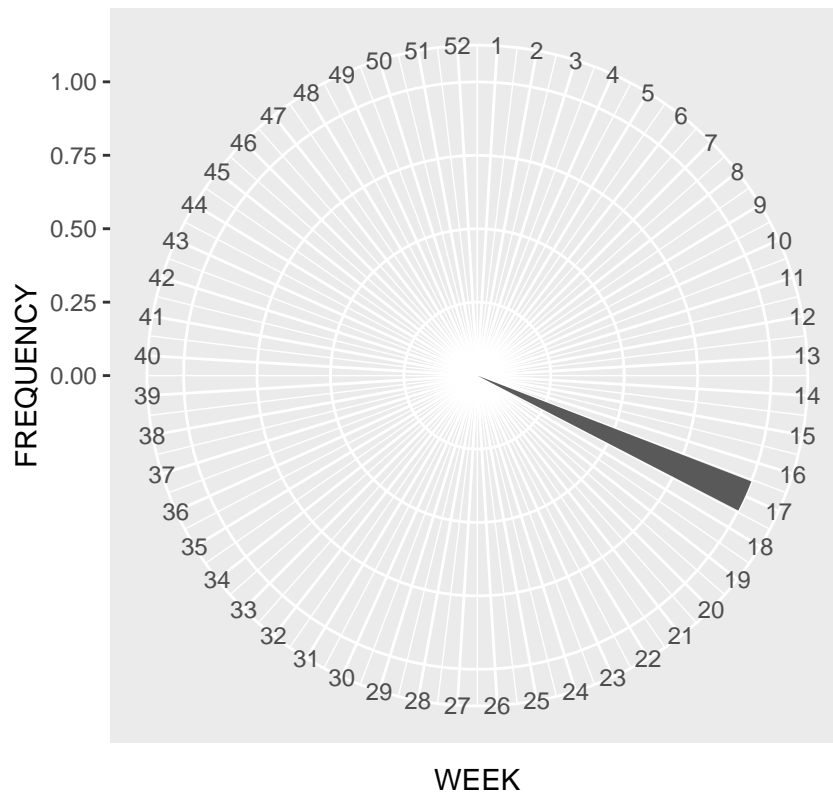
```
##
## $group3_min_time2_week_plot
```

GROUP 3 MIN TIME 2 WEEK FREQUENCY

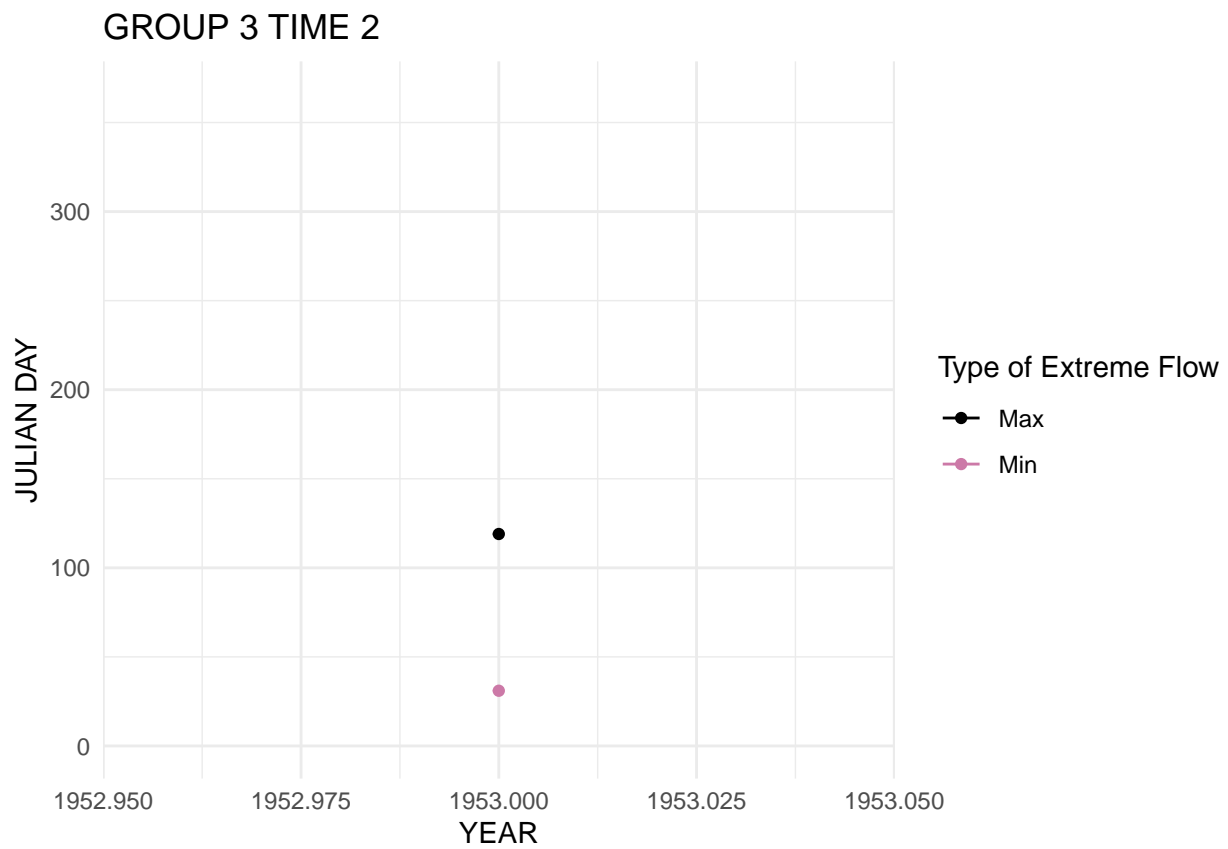


```
##
## $group3_max_time2_week_plot
```

GROUP 3 MAX TIME 2 WEEK FREQUENCY

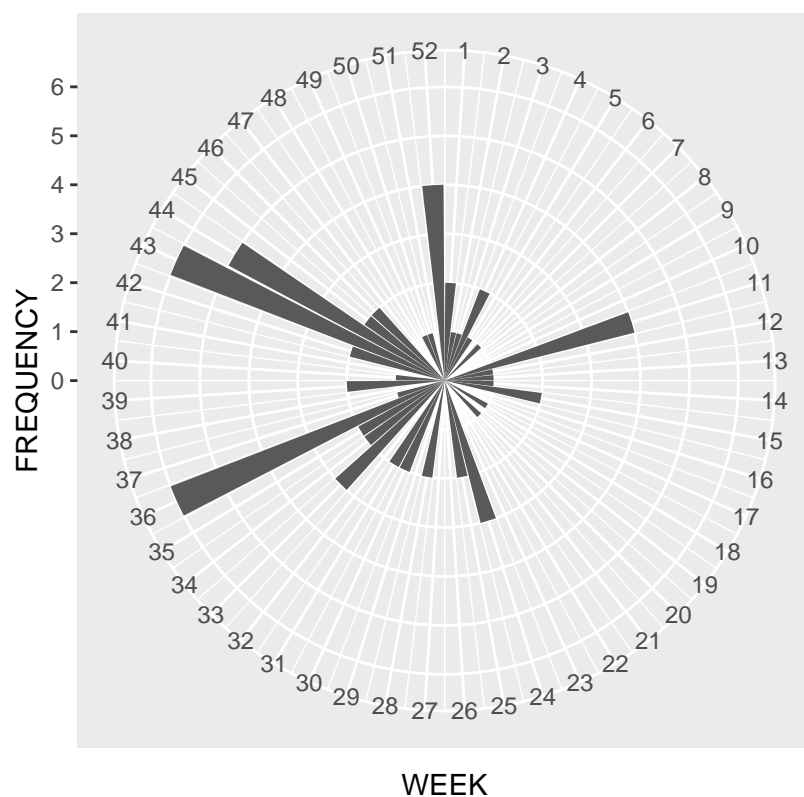


```
##
## $group3_time2_julian_day_plot
## `geom_line()`: Each group consists of only one observation.
## i Do you need to adjust the group aesthetic?
```



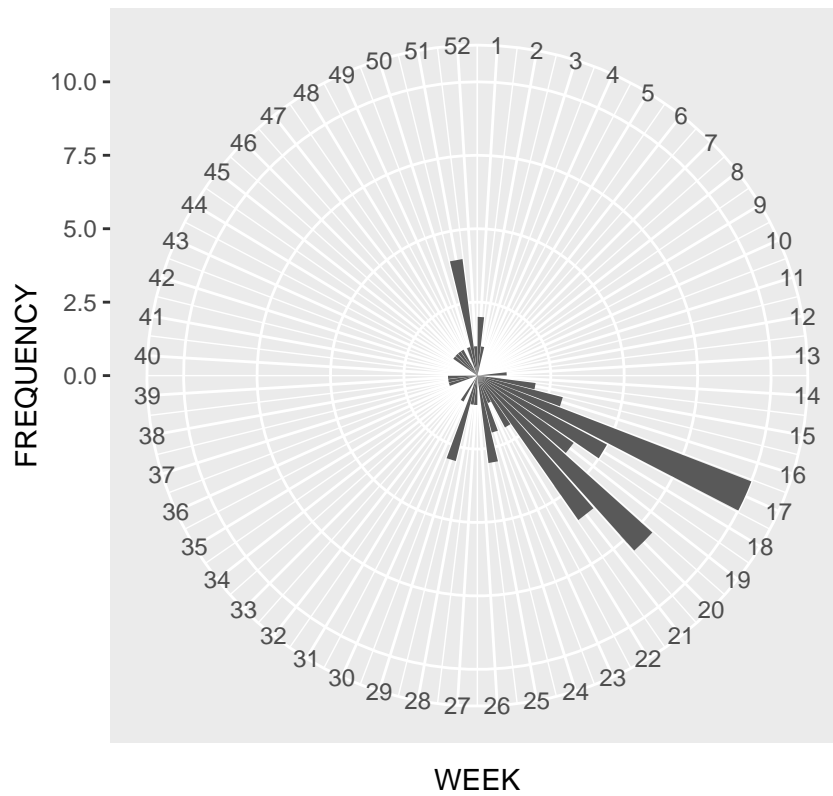
```
##  
## $group3_min_time3_week_plot
```

GROUP 3 MIN TIME 3 WEEK FREQUENCY

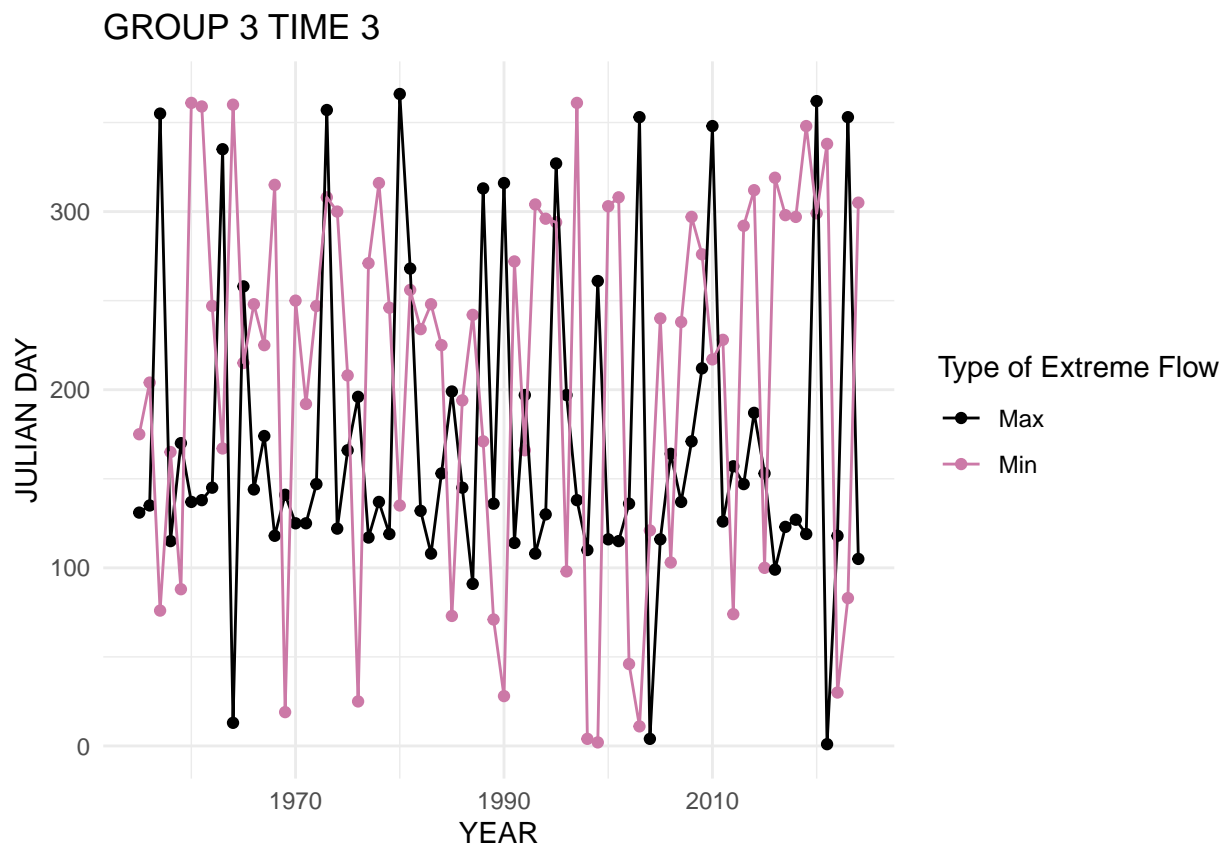


```
##
## $group3_max_time3_week_plot
## Warning: Removed 1 row containing missing values or values outside the scale range
## (`geom_bar()`).
```

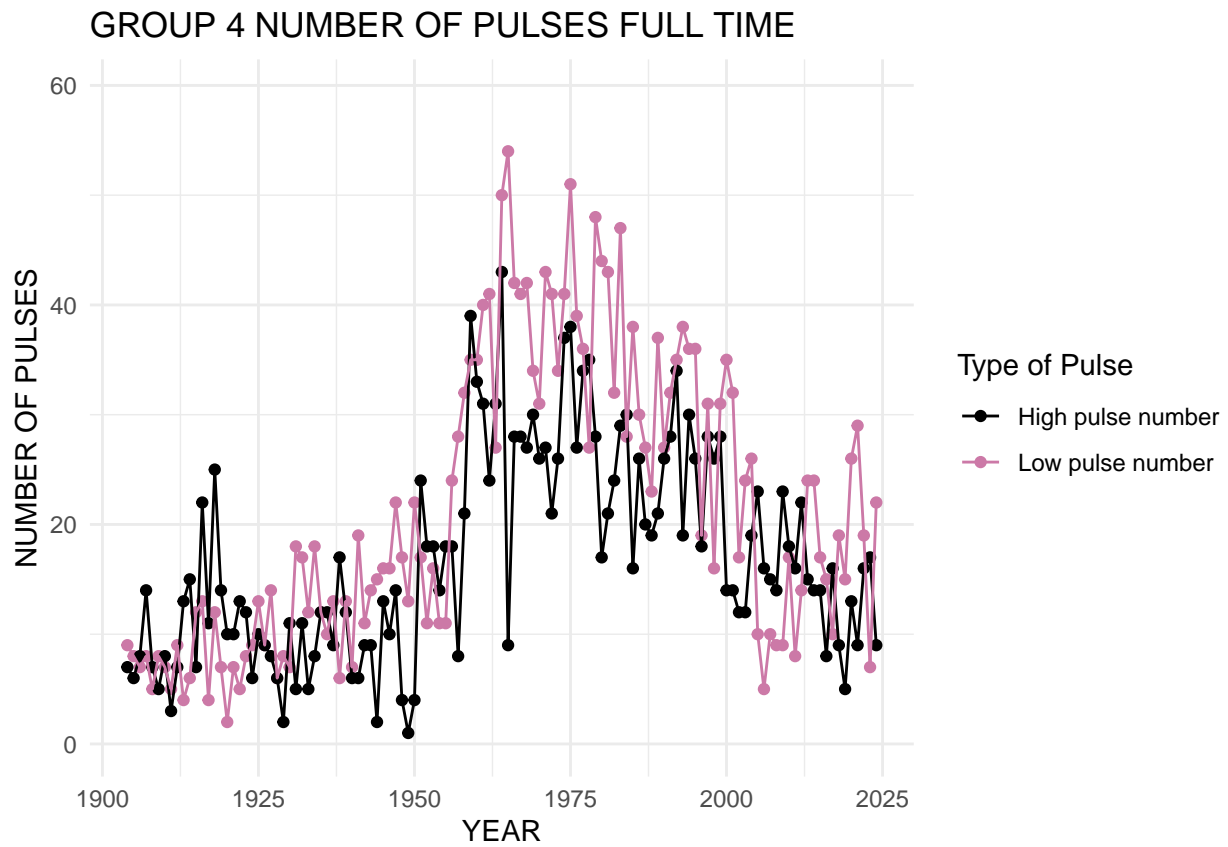

GROUP 3 MAX TIME 3 WEEK FREQUENCY



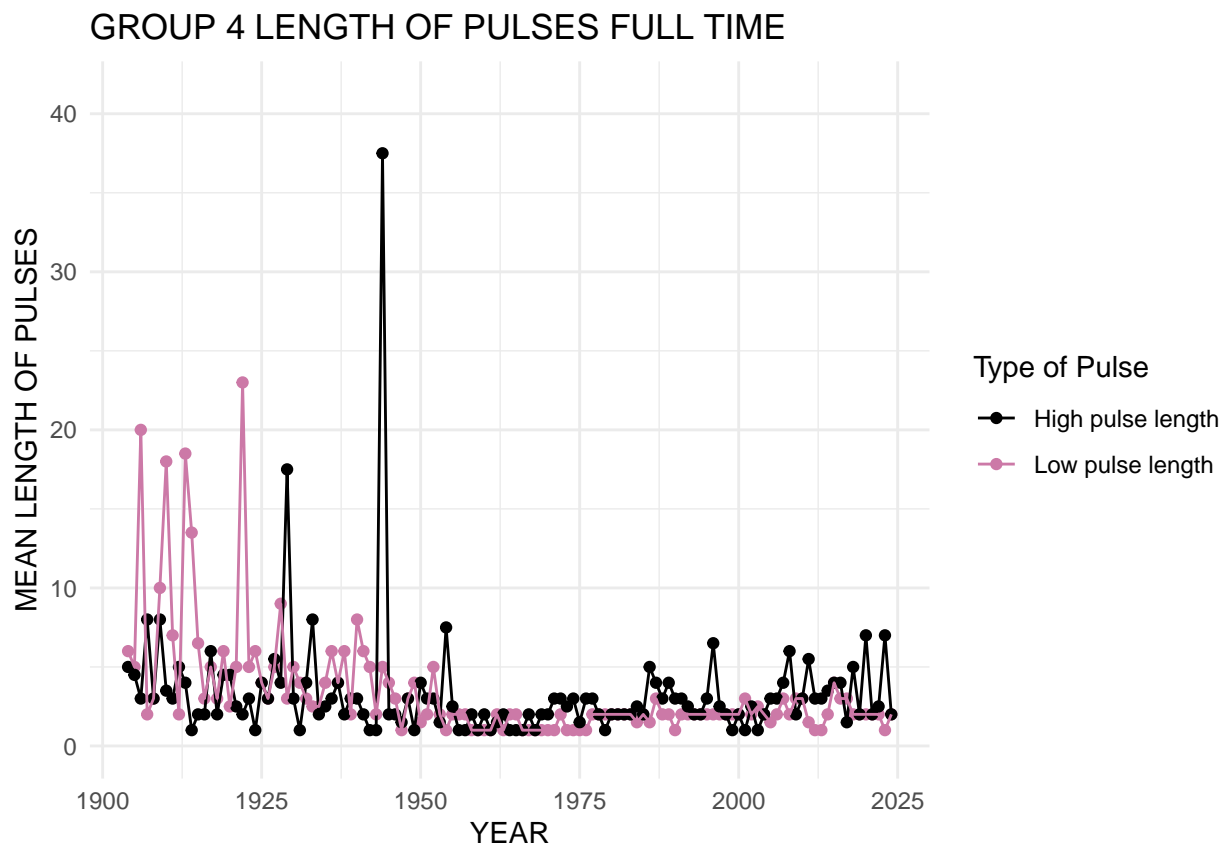
```
##
## $group3_time3_julian_day_plot
```



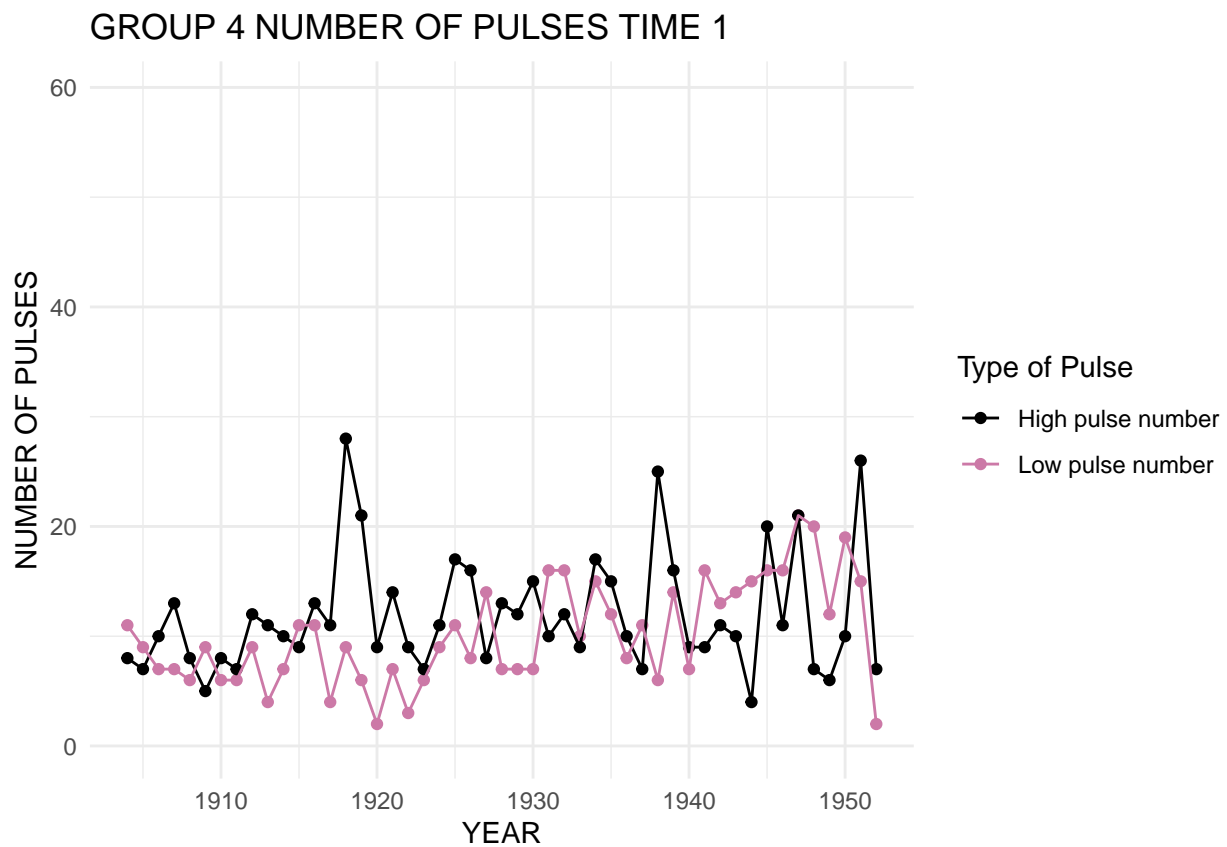
```
## $group4_number_full_plot
```



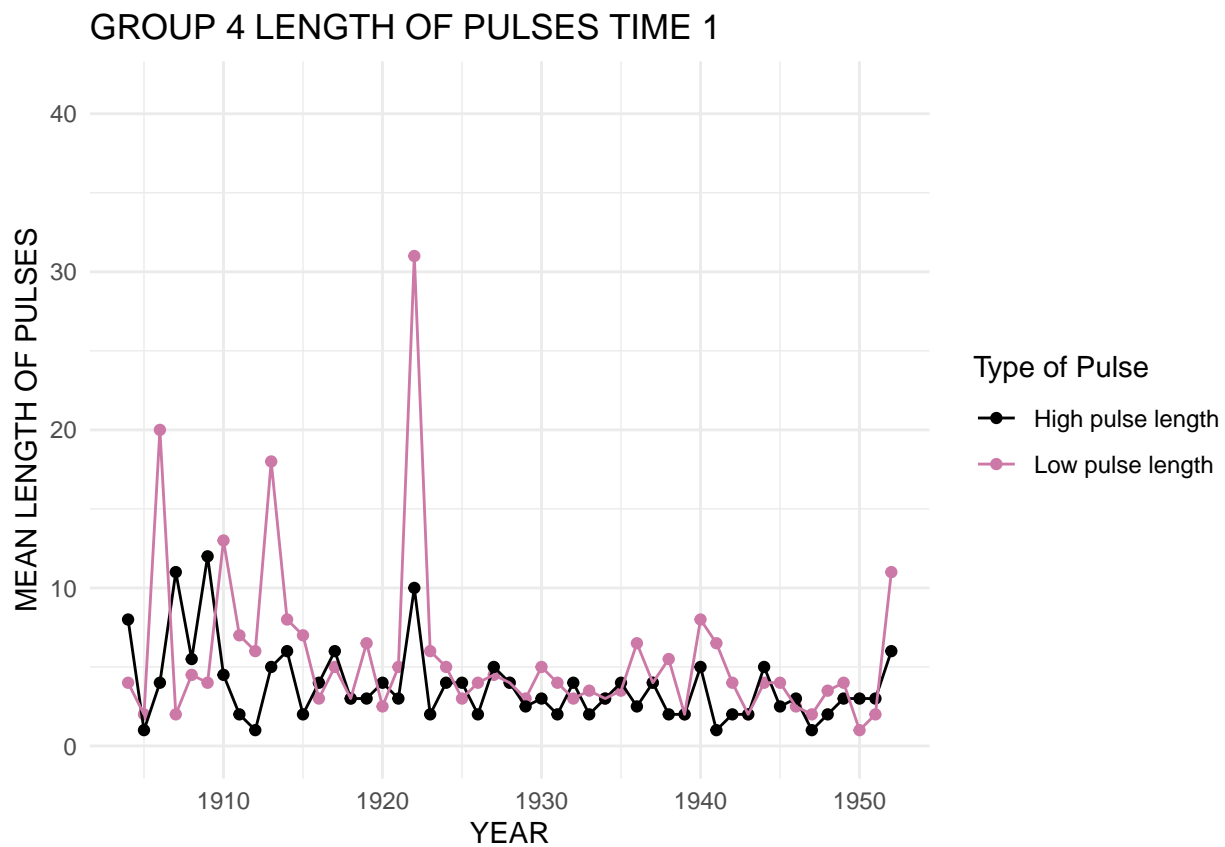
```
##  
## $group4_length_full_plot
```



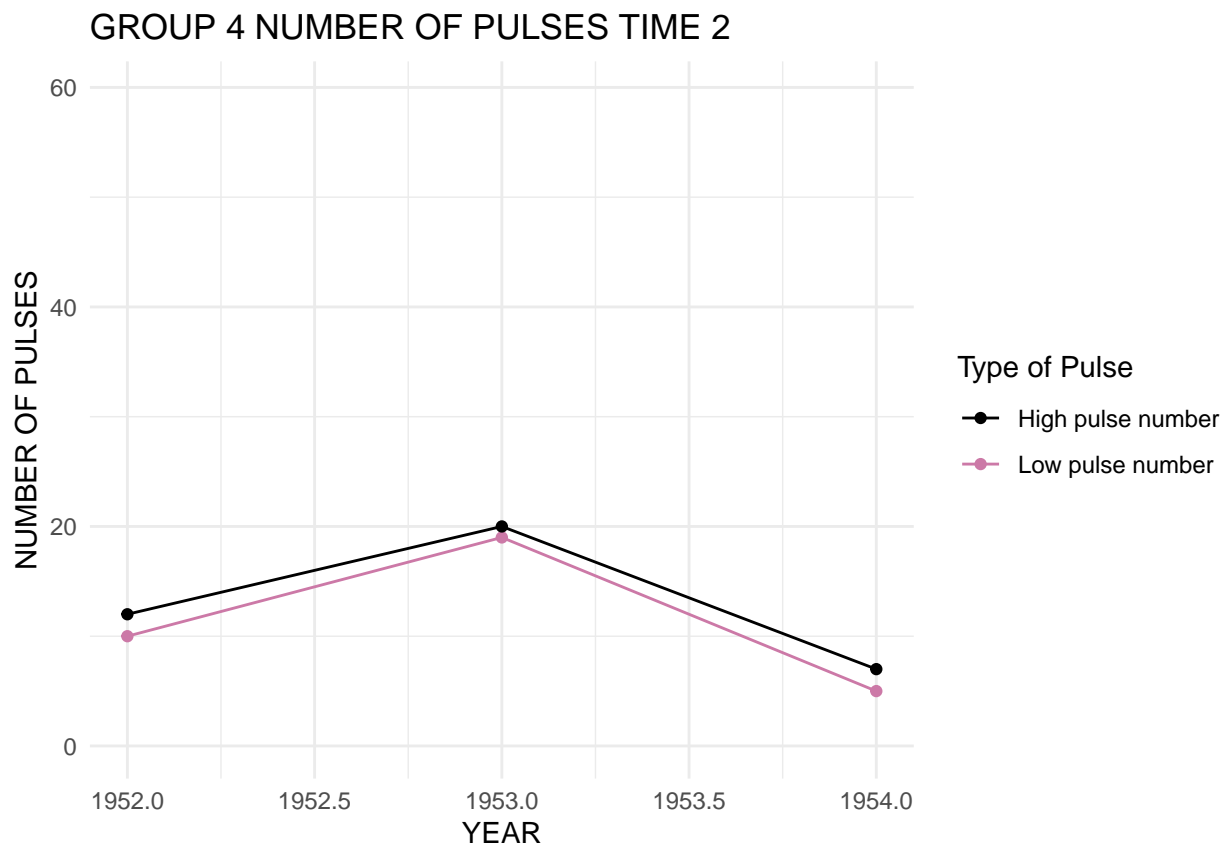
```
##  
## $group4_number_time1_plot
```



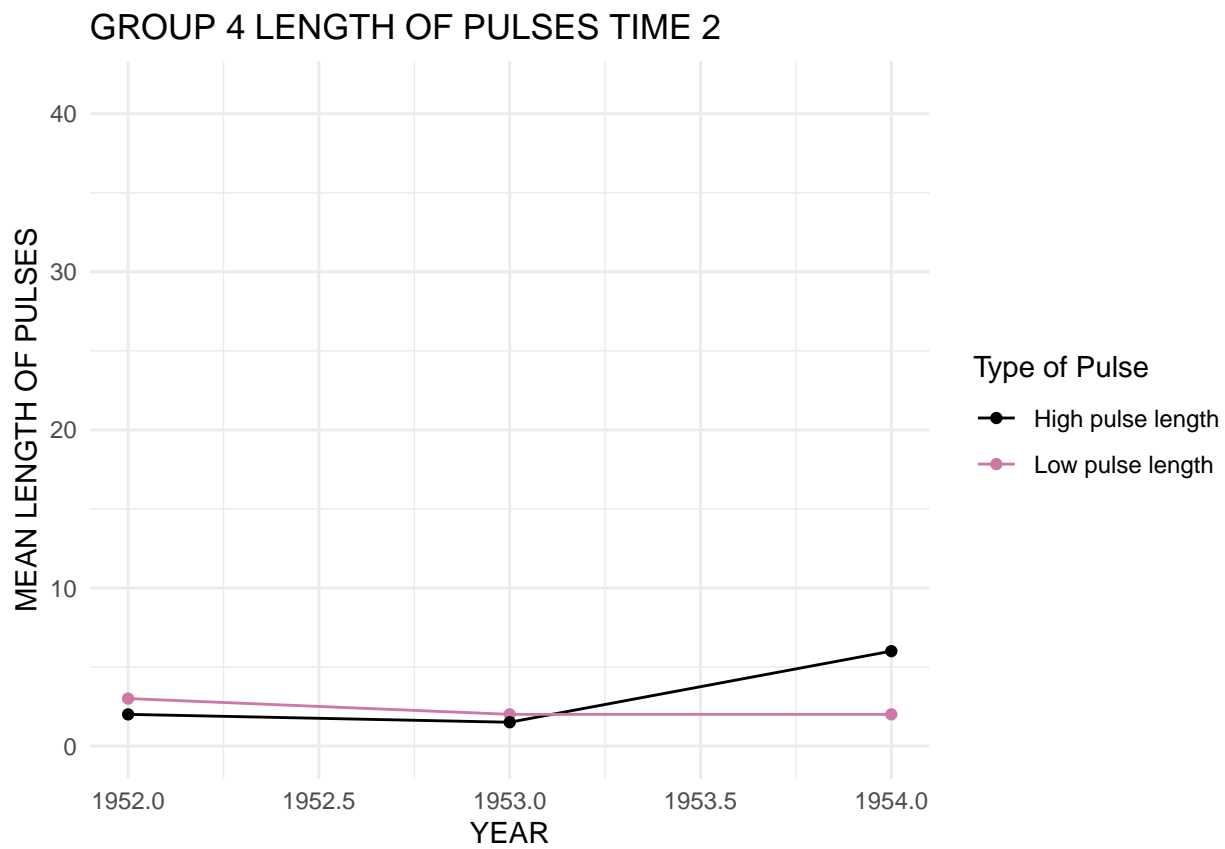
```
##  
## $group4_length_time1_plot
```



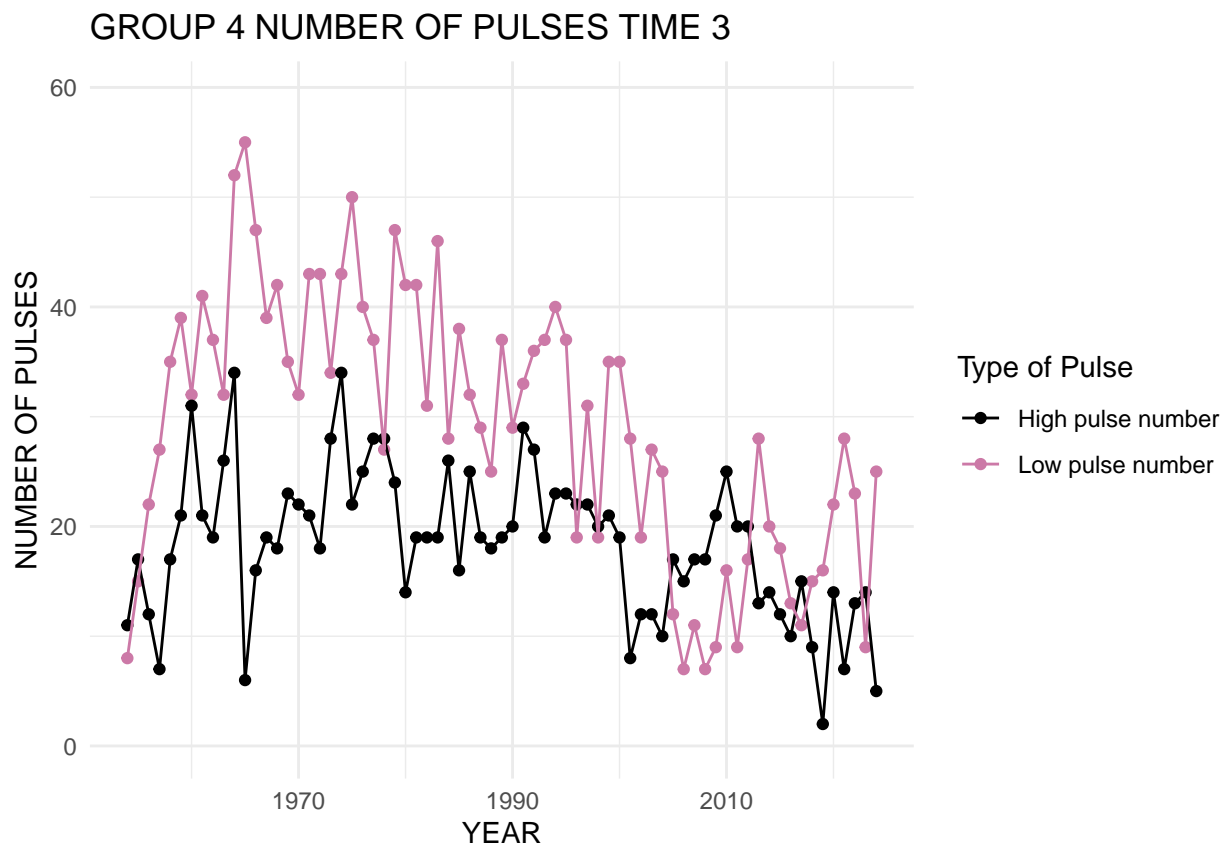
```
##  
## $group4_number_time2_plot
```



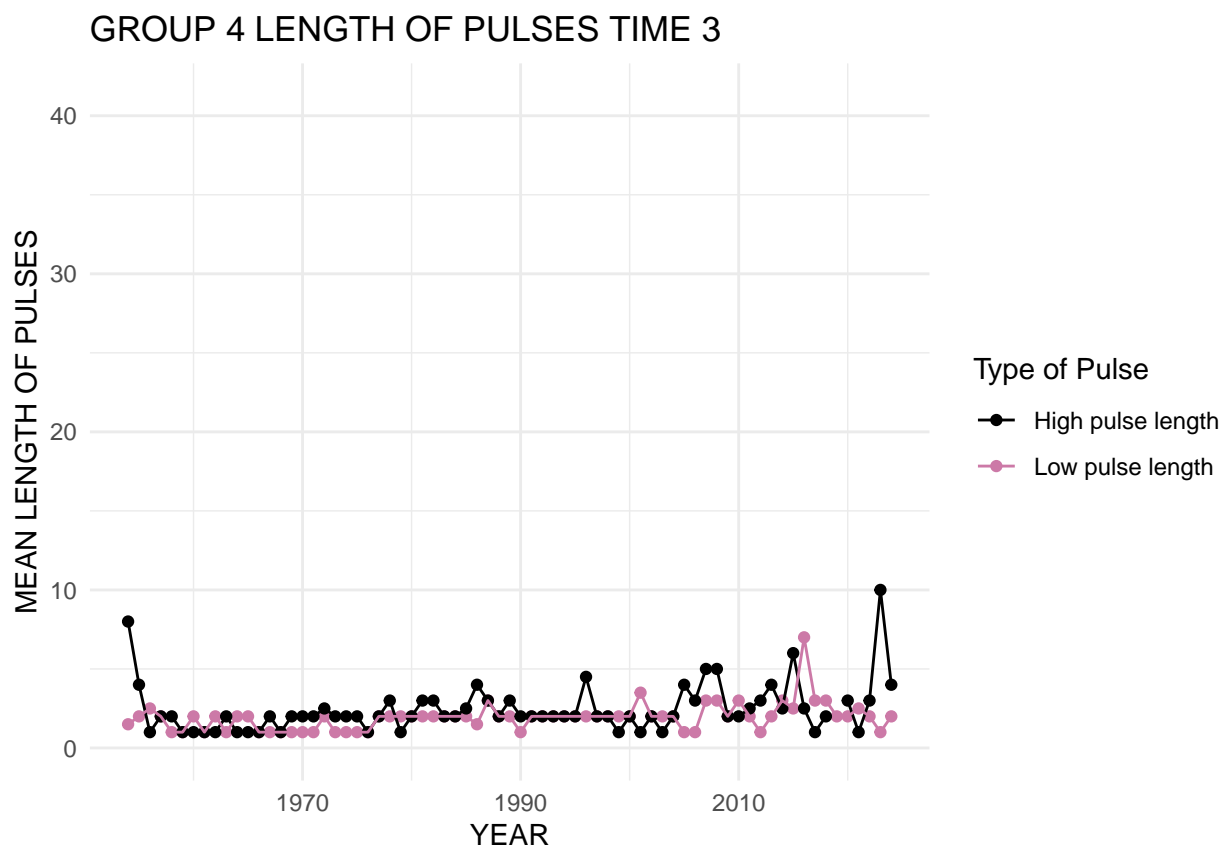
```
##  
## $group4_length_time2_plot
```



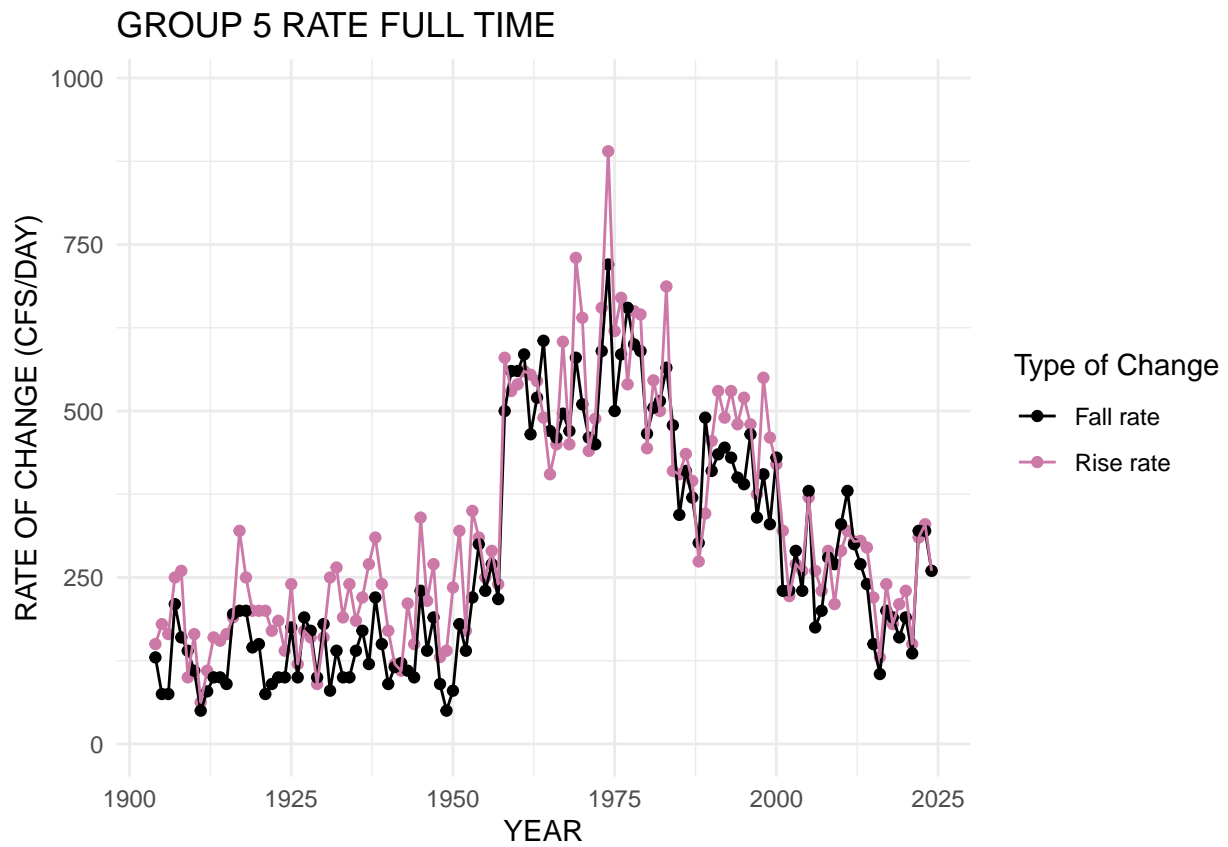
```
##  
## $group4_number_time3_plot
```

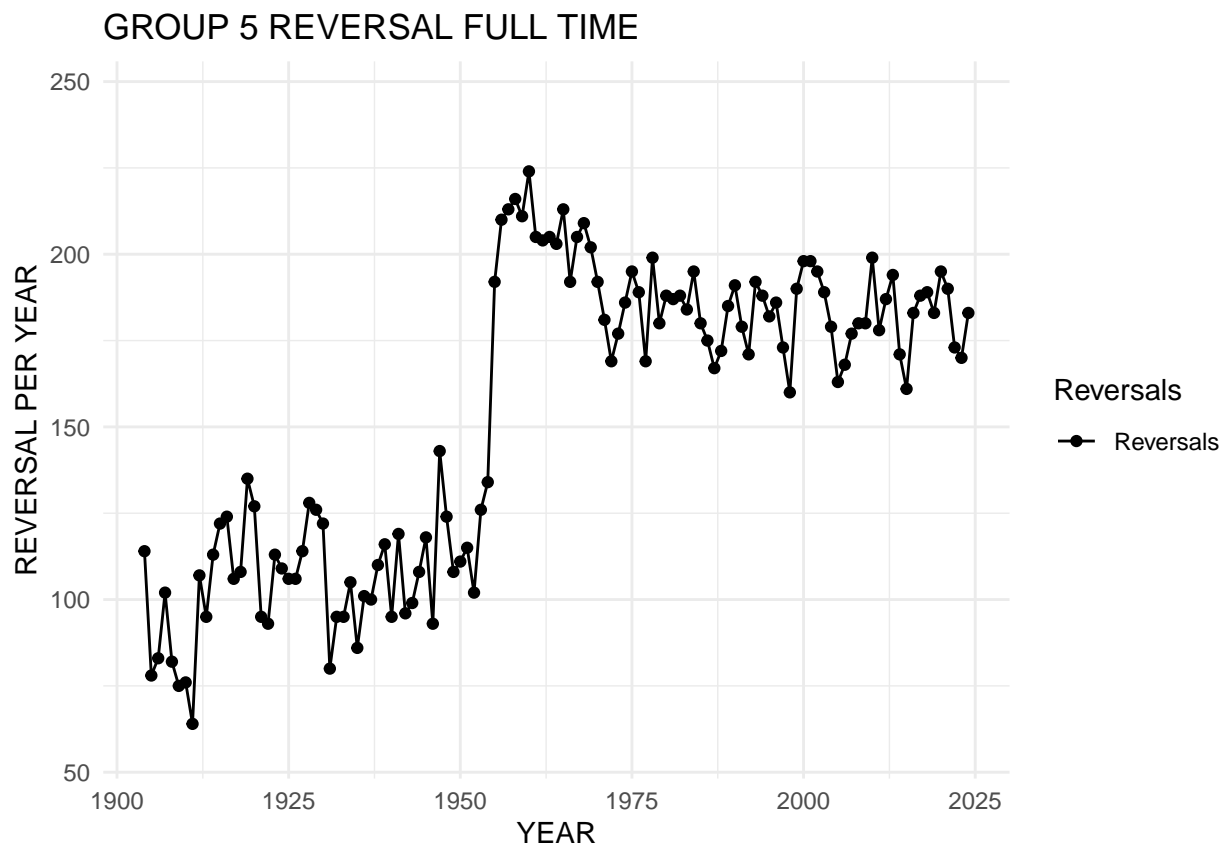
```
##  
## $group4_length_time3_plot
```



\$group5_rate_full_plot

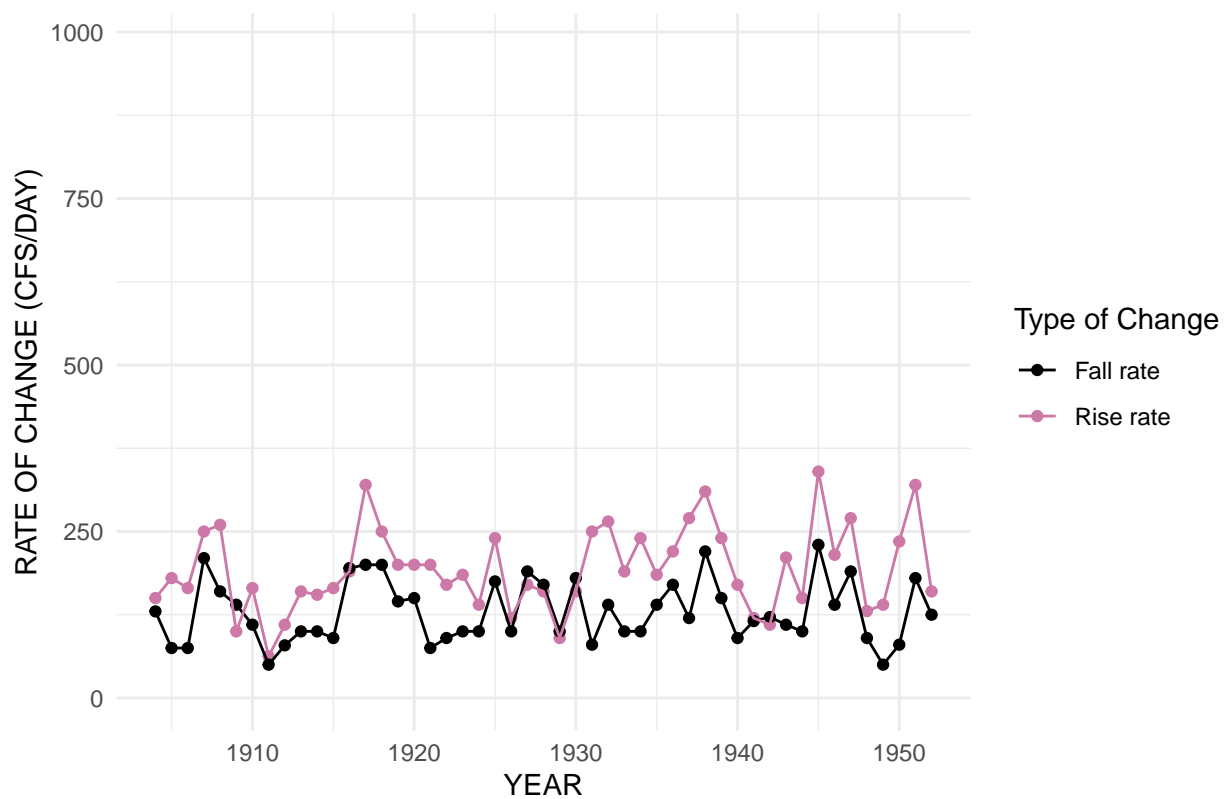


```
##  
## $group5_reversal_full_plot
```

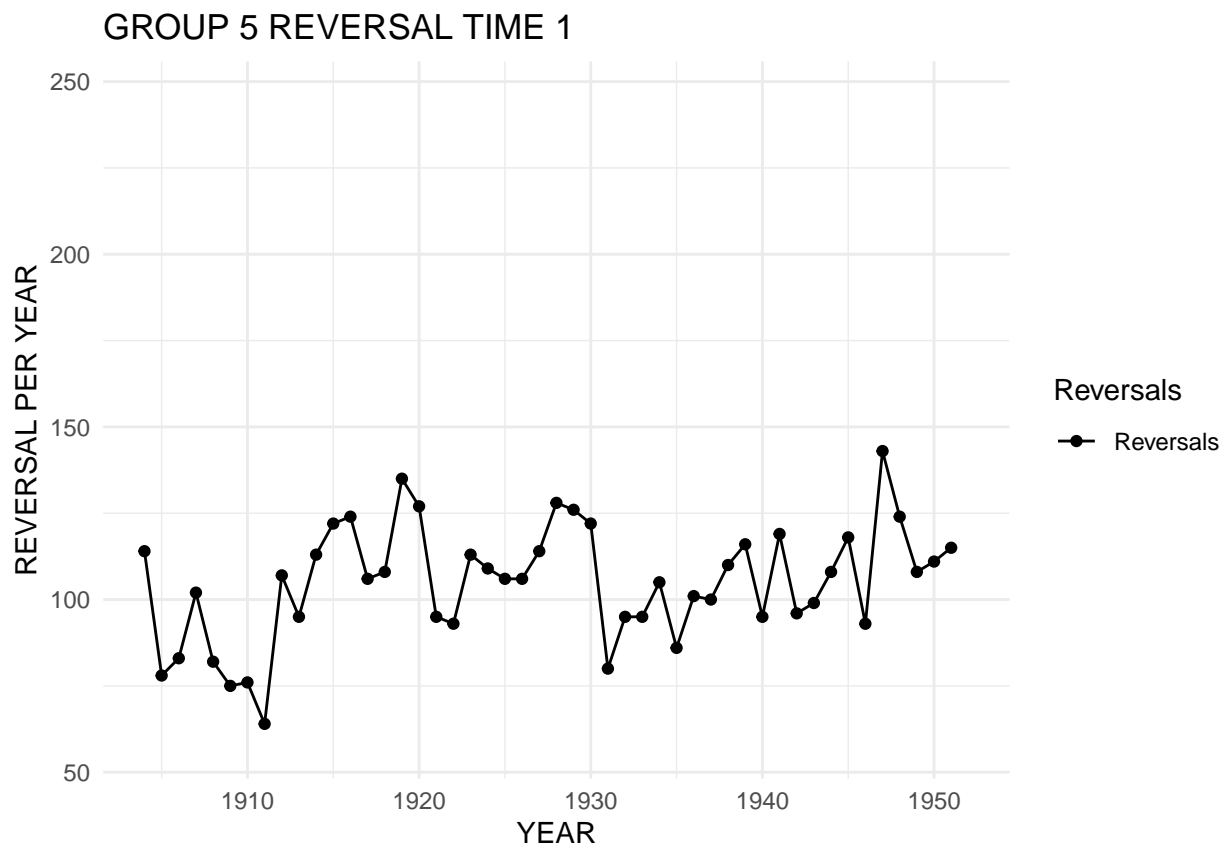


```
##  
## $group5_rate_time1_plot
```

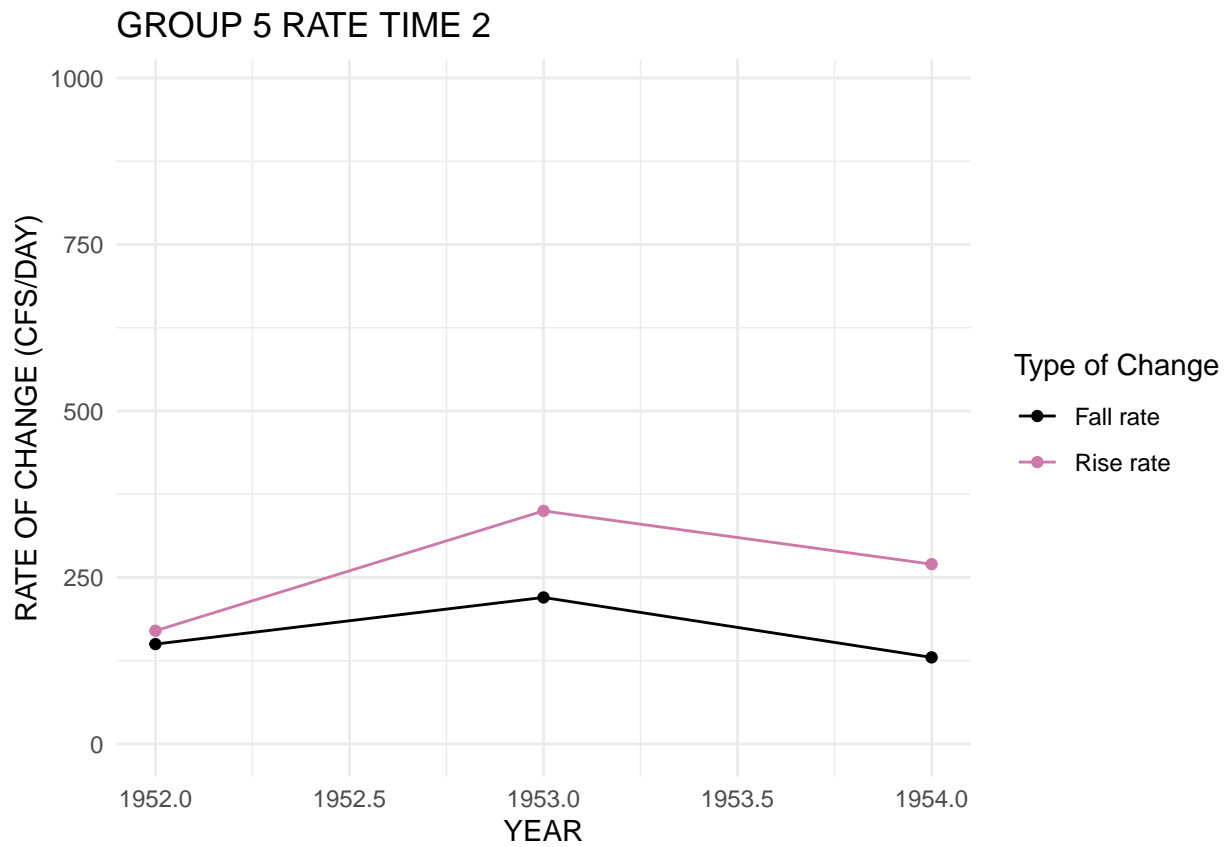
GROUP 5 RATE TIME 1



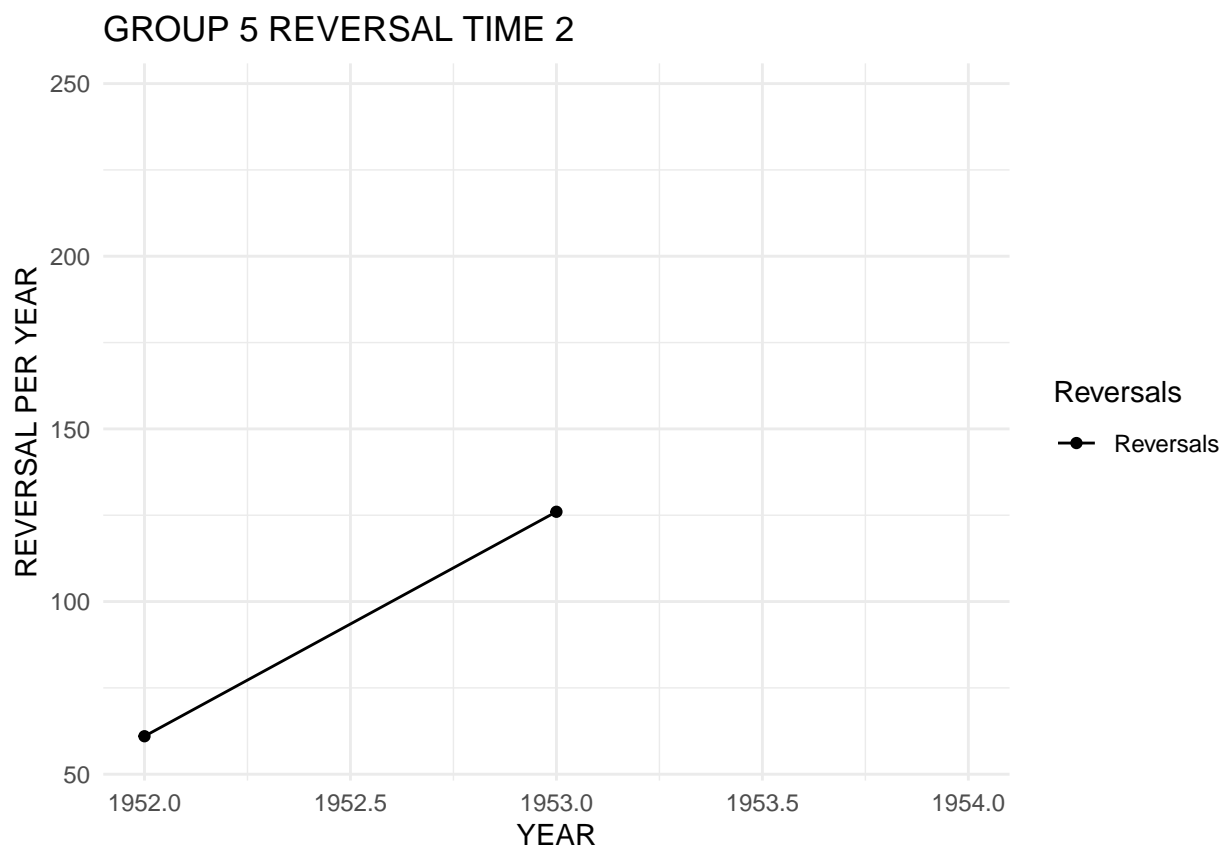
```
##  
## $group5_reversal_time1_plot
```



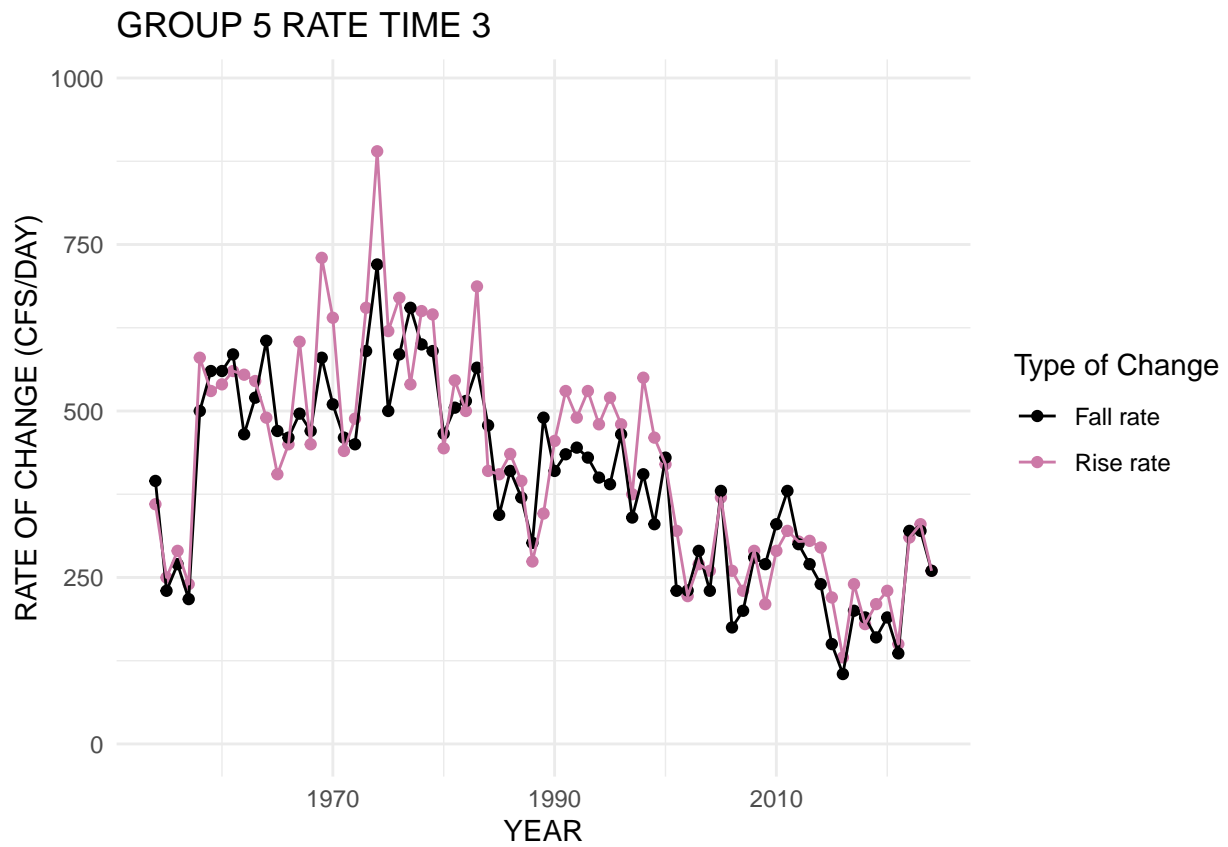
```
##  
## $group5_rate_time2_plot
```



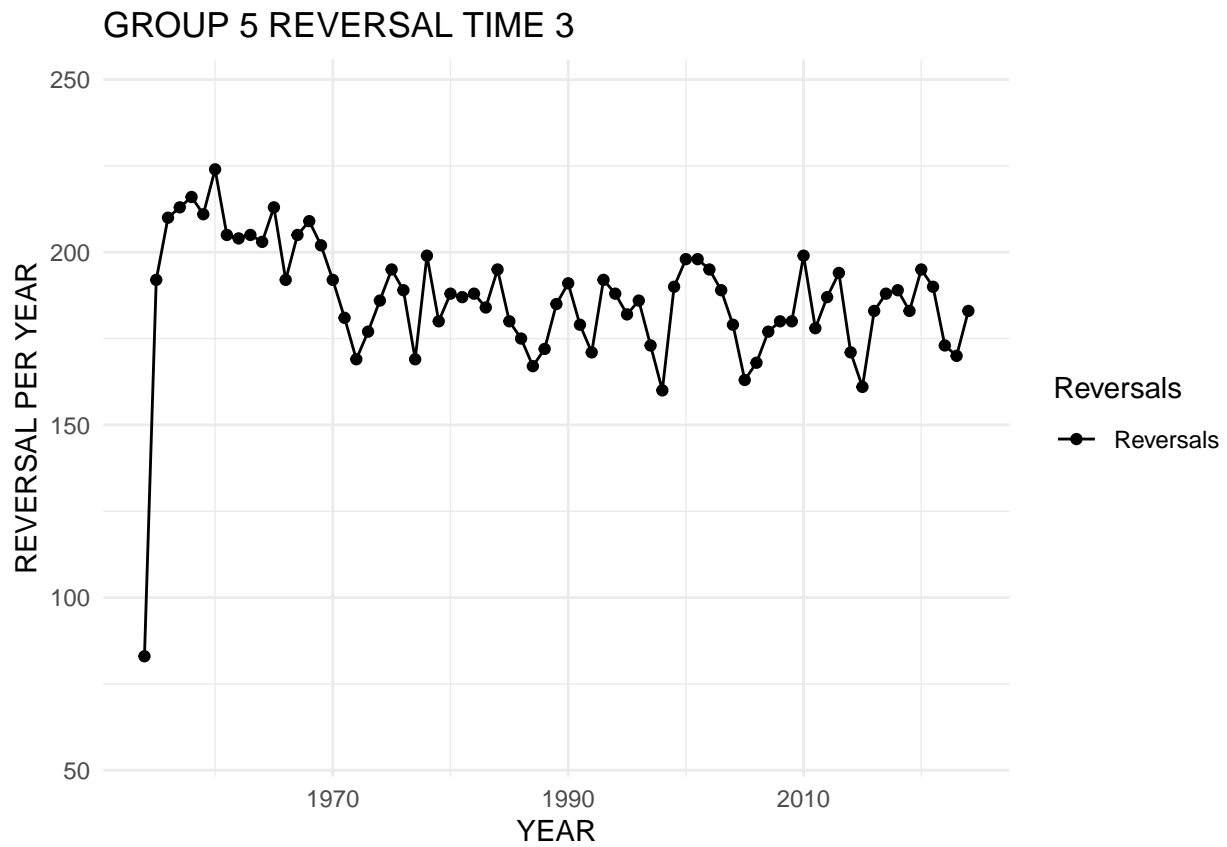
```
##  
## $group5_reversal_time2_plot
```



```
##  
## $group5_rate_time3_plot
```

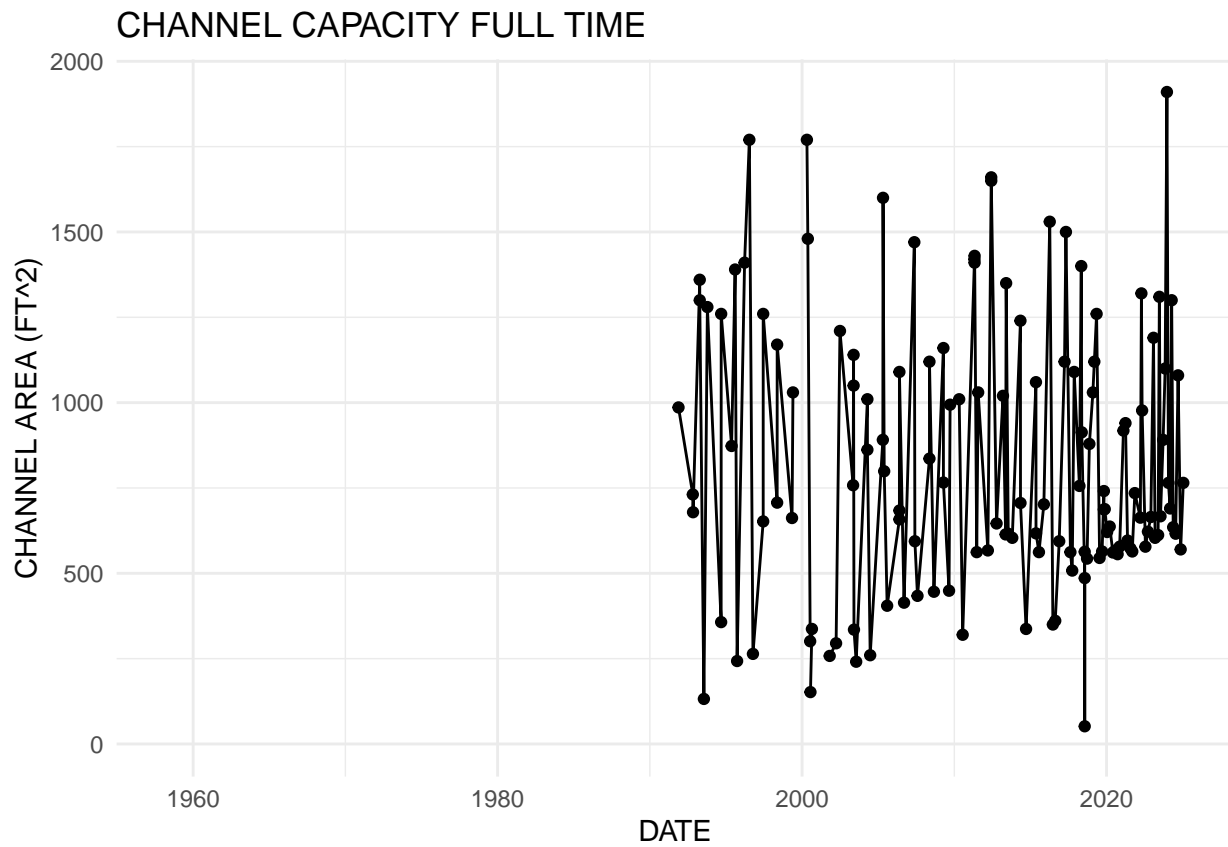



```
##  
## $group5_reversal_time3_plot
```

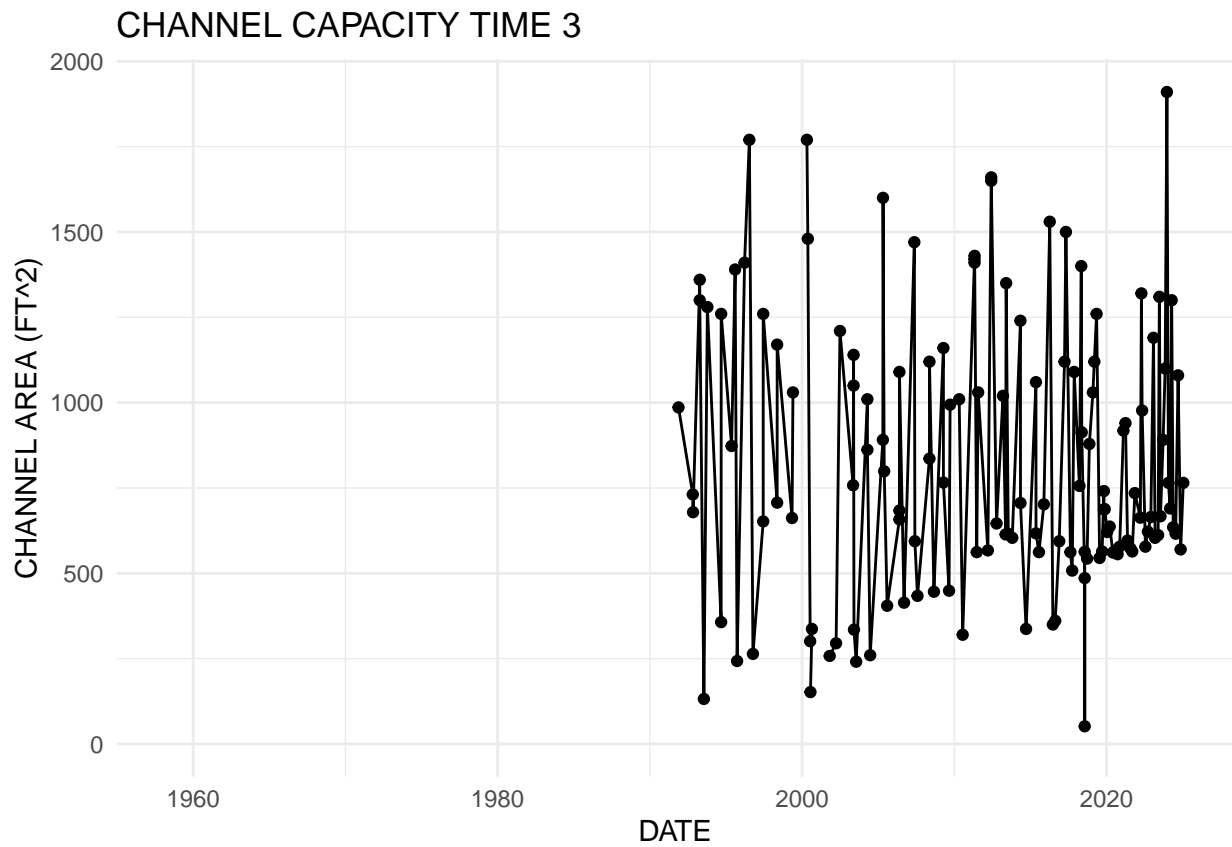


Field Measurements Manipulation and Visualization

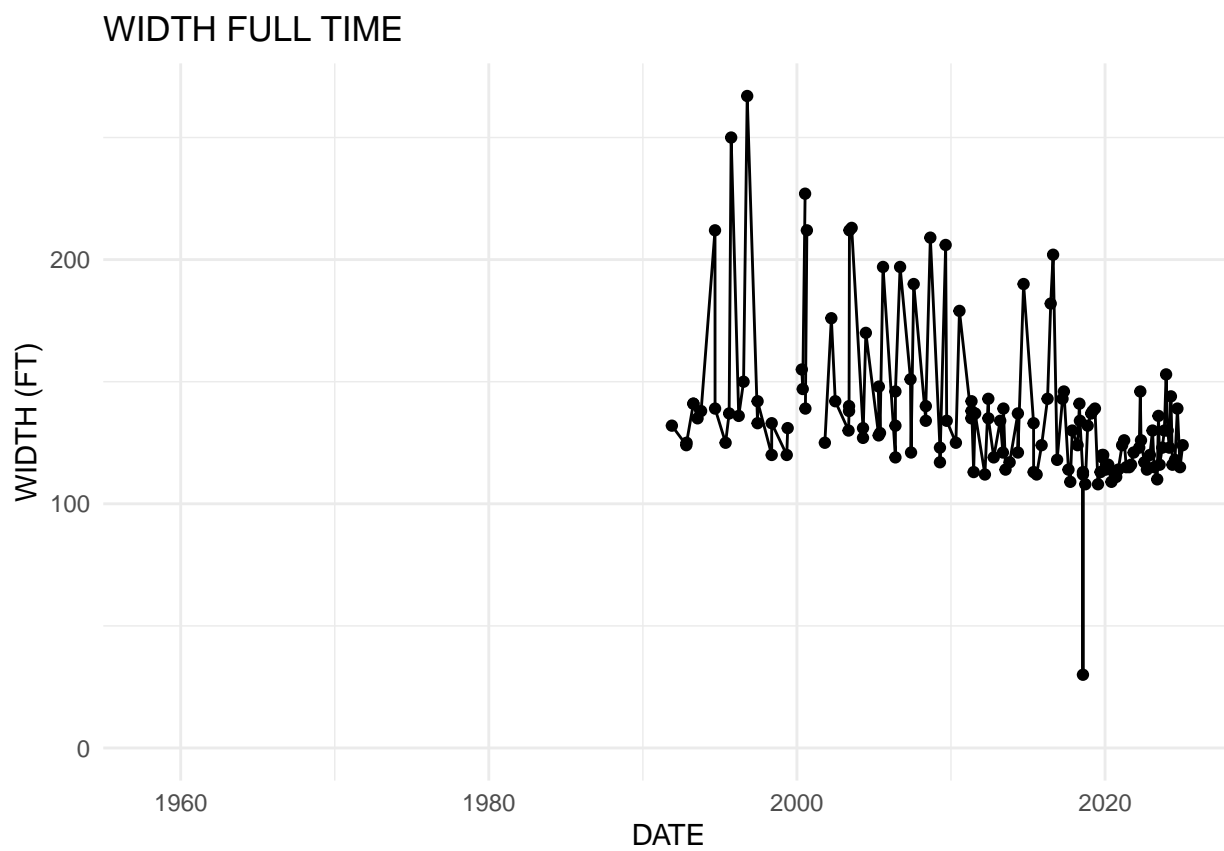
```
## $chan_capacity_full_plot
```



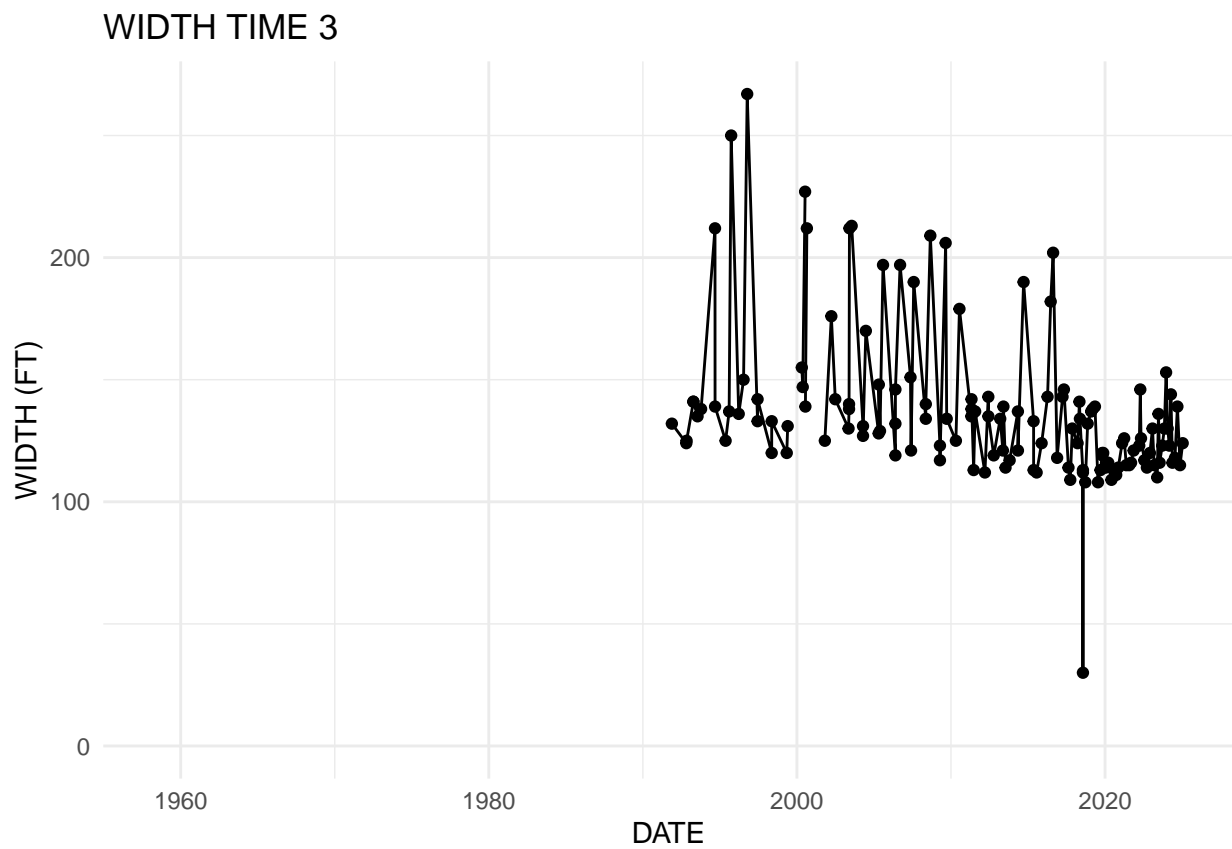
```
##  
## $chan_capacity_time3_plot
```



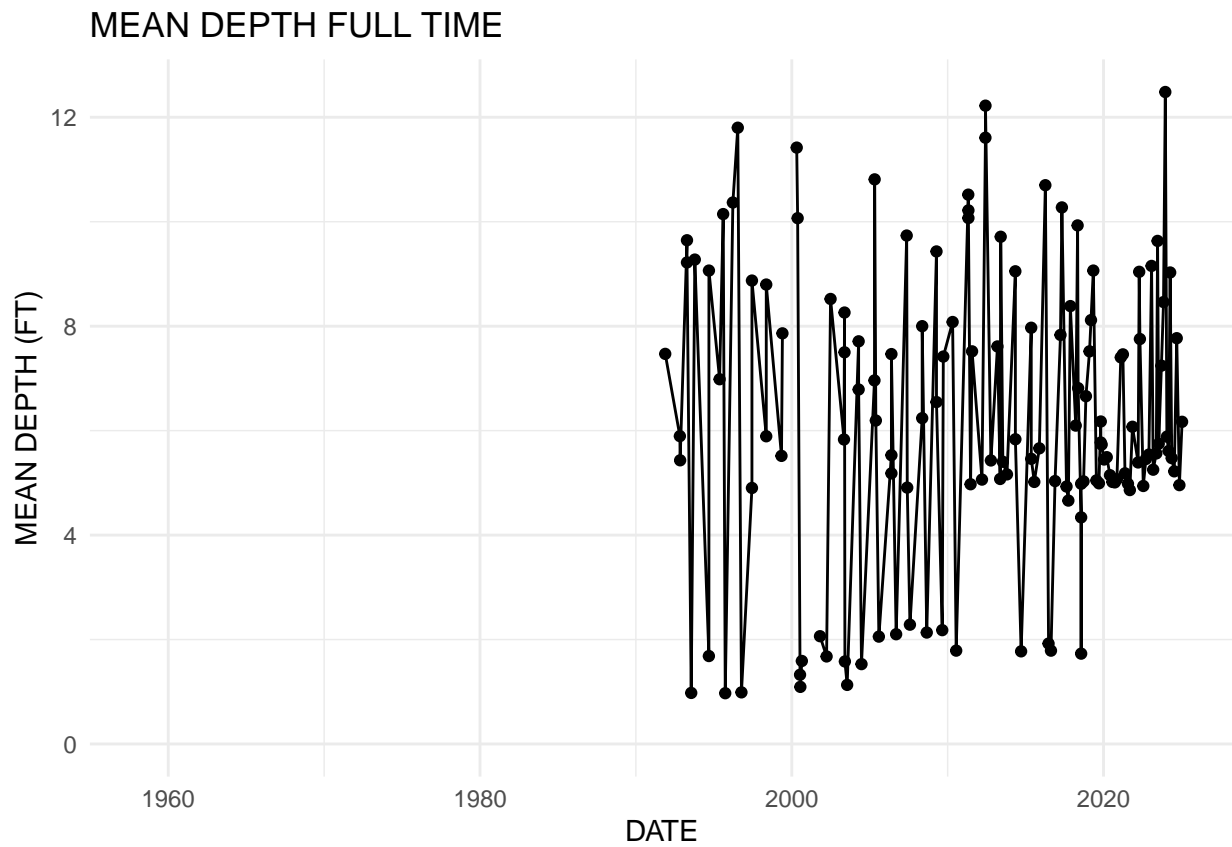
```
##  
## $width_full_plot
```



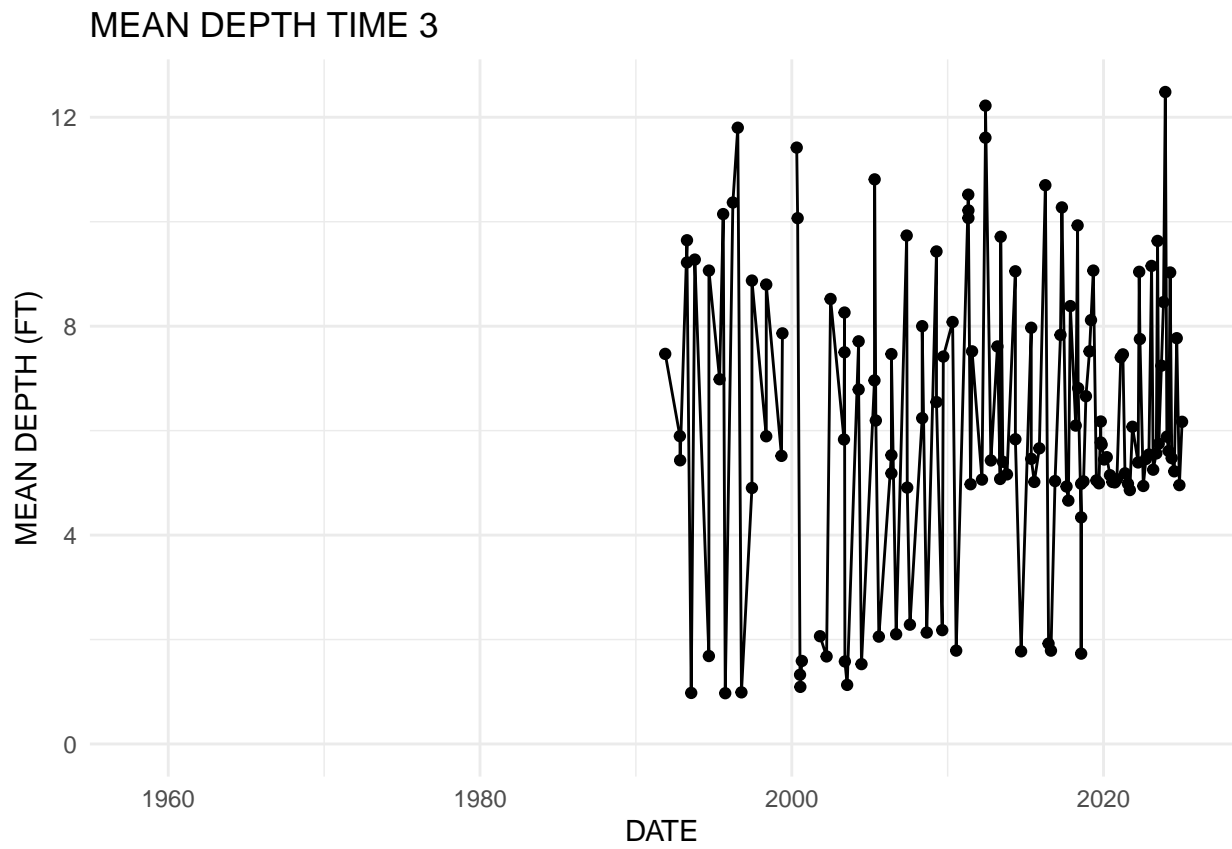
```
##  
## $width_time3_plot
```



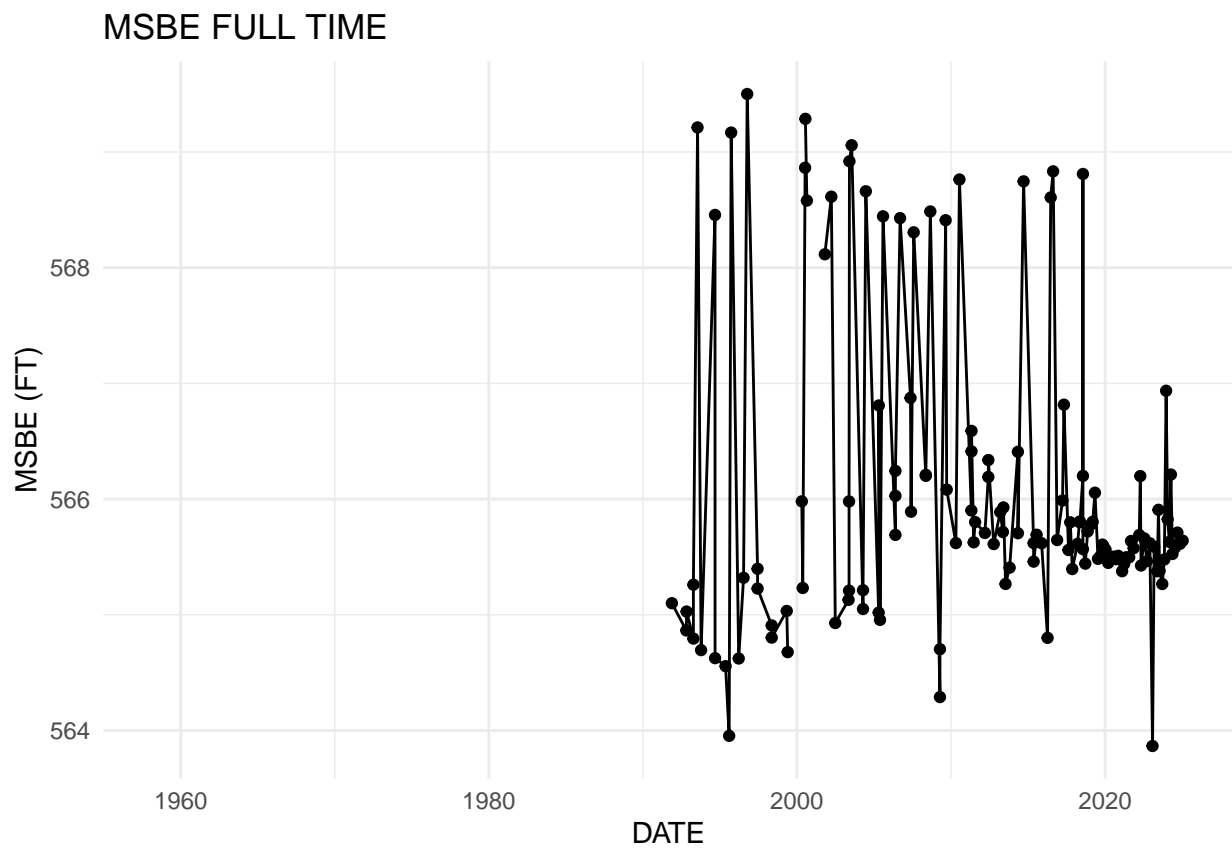
```
##  
## $mean_depth_full_plot
```



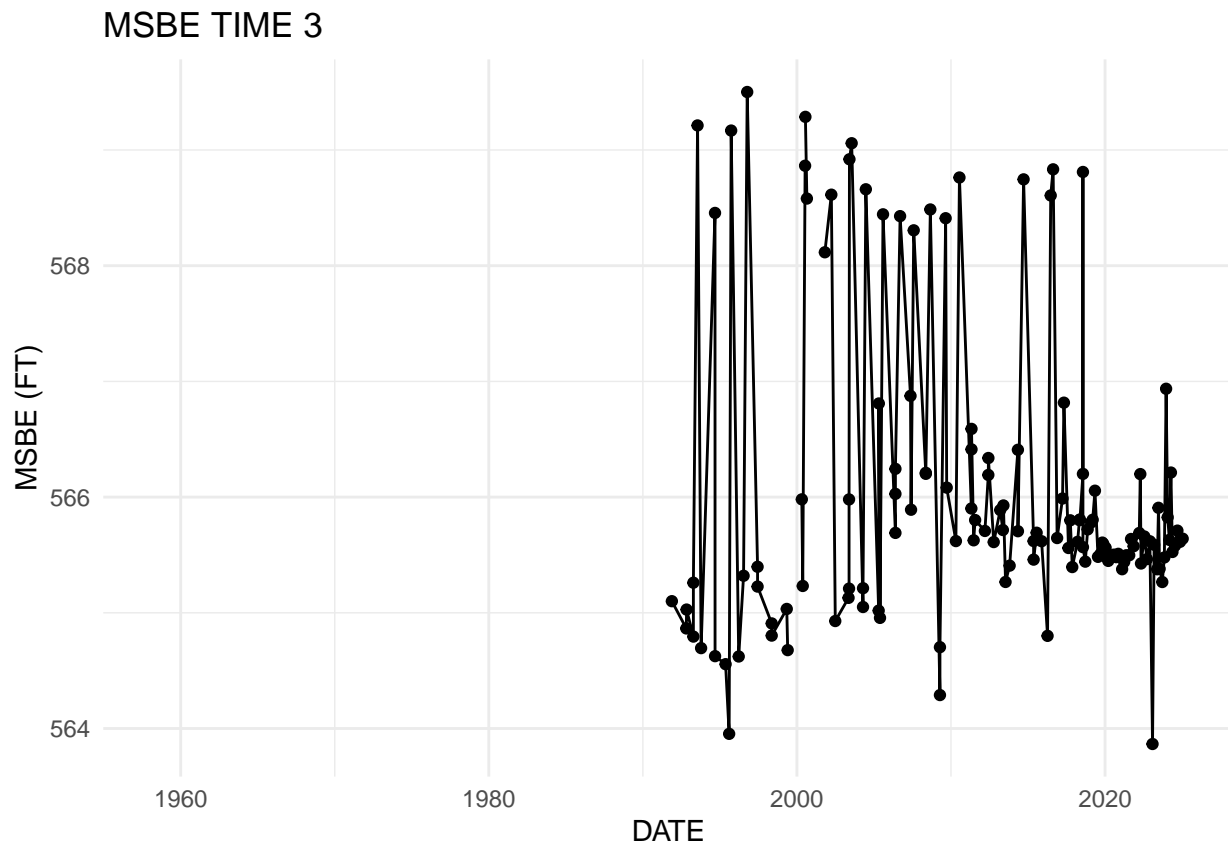
```
##  
## $mean_depth_time3_plot
```



```
##  
## $MSBE_full_plot
```

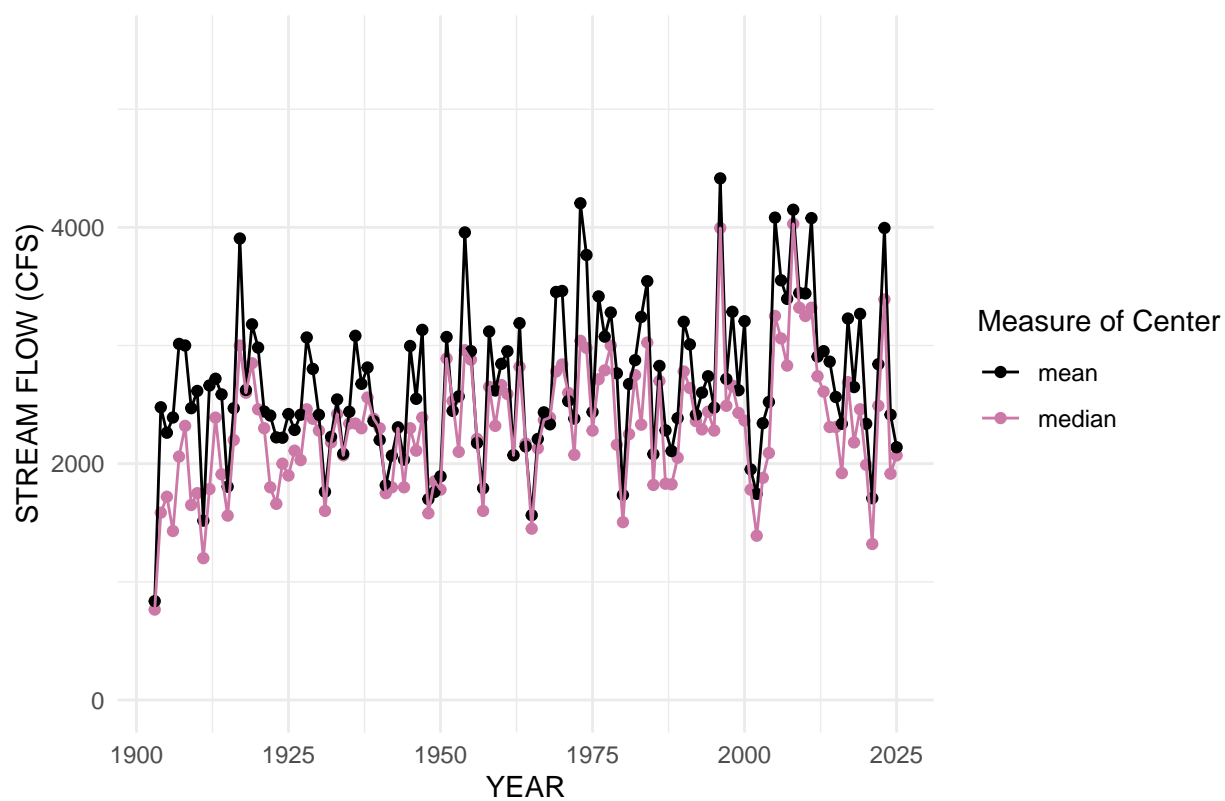
```
##  
## $MSBE_time3_plot
```



Measure of Center Manipulation and Visualization

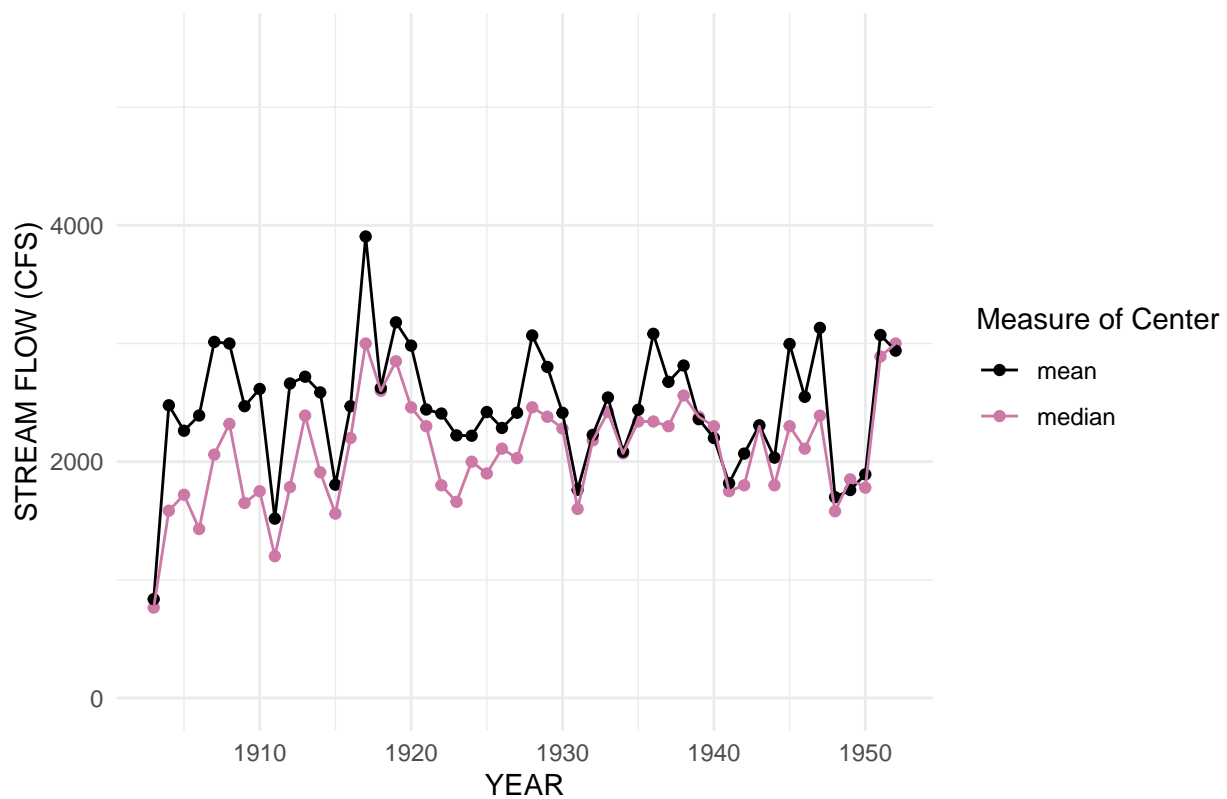
```
## $measure_of_center_full_plot
```

MEASURE OF CENTER FULL



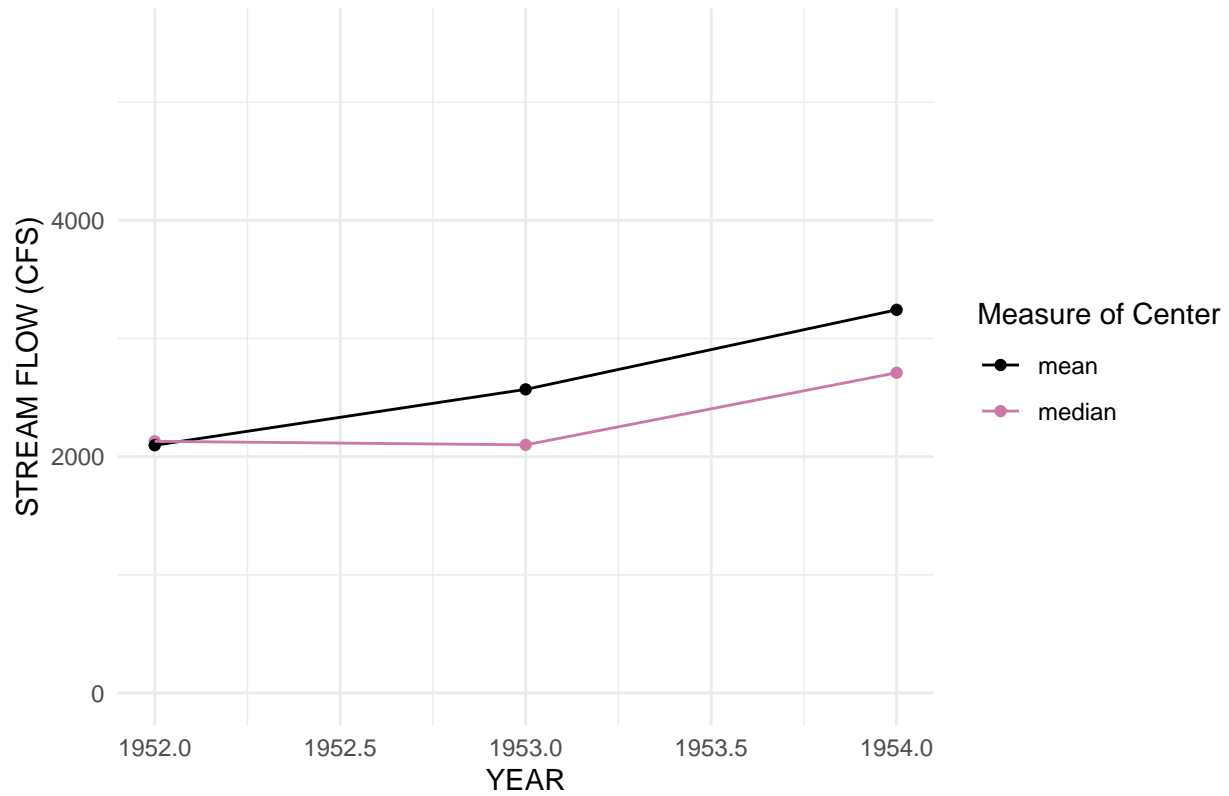
```
##  
## $measure_of_center_time1_plot
```

MEASURE OF CENTER TIME 1



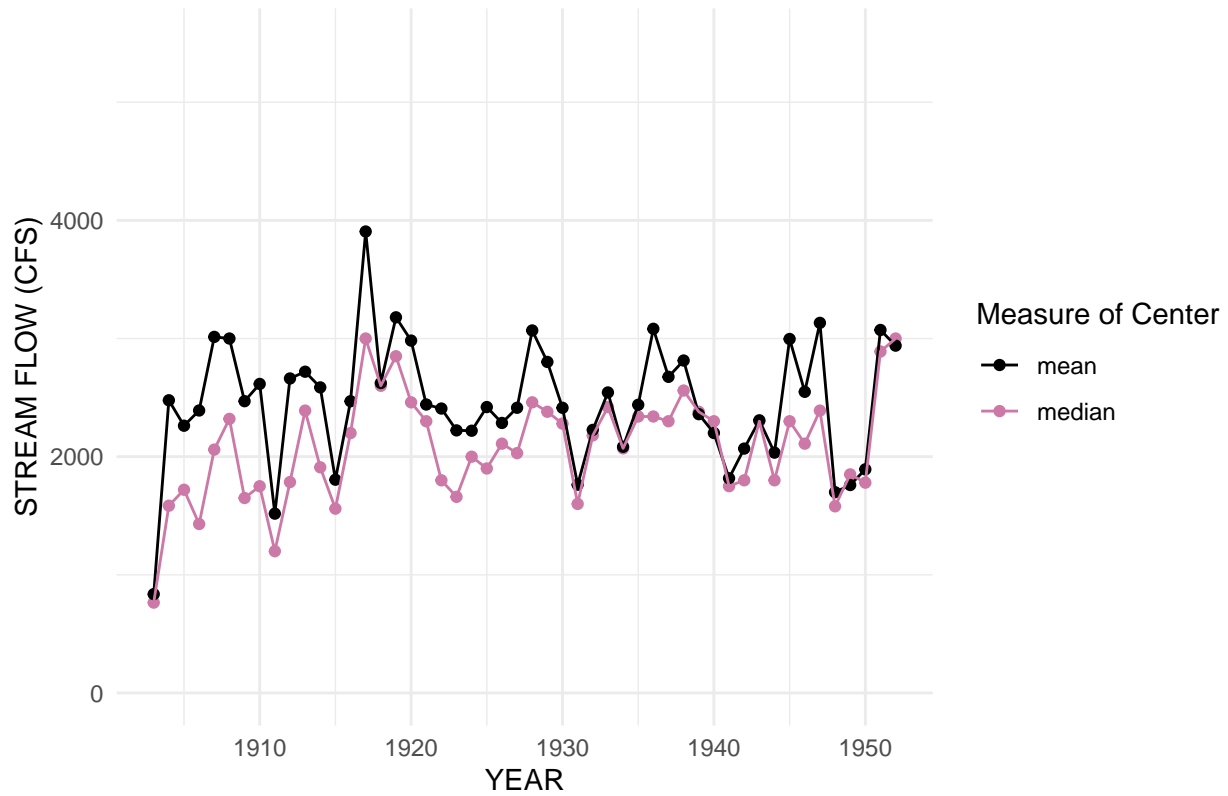
```
##  
## $measure_of_center_time2_plot
```

MEASURE OF CENTER TIME 2



```
##  
## $measure_of_center_time3_plot
```

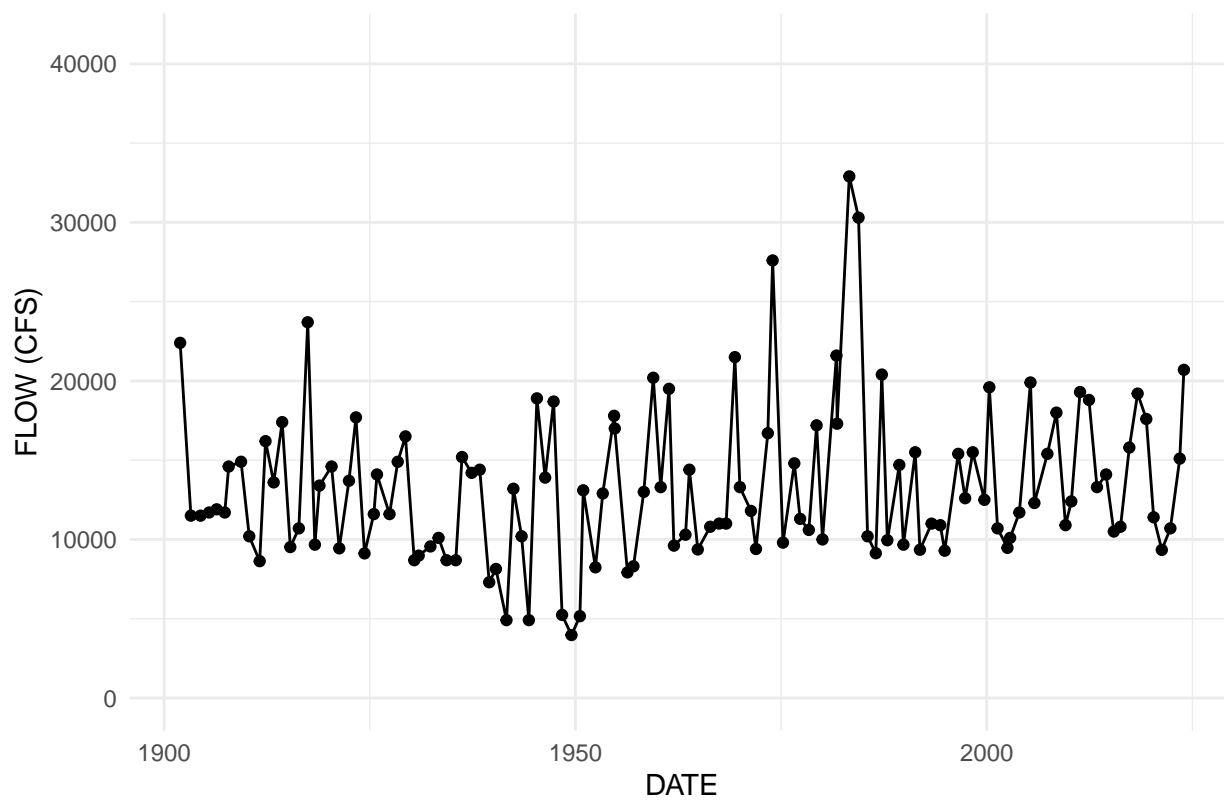
MEASURE OF CENTER TIME 3



Peak Flow Manipulation and Visualization

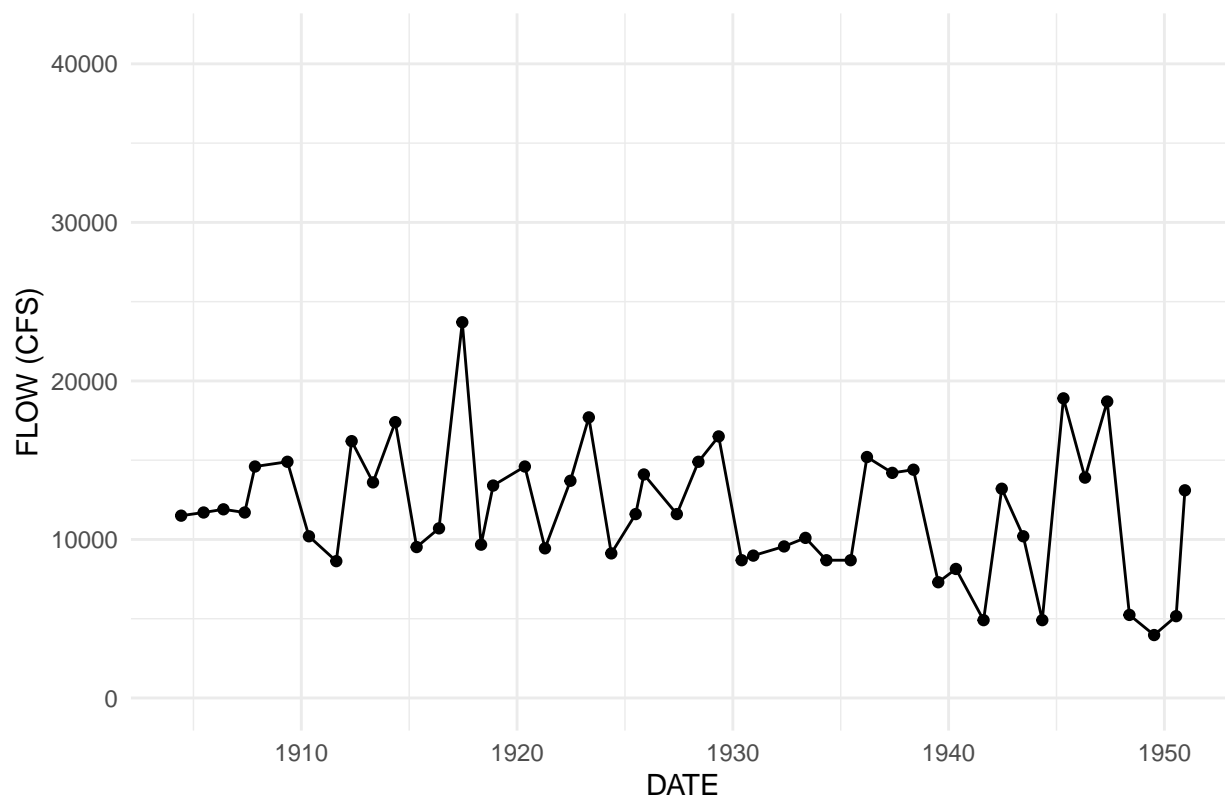
```
## $peak_flow_full_plot
```

ANNUAL PEAK FLOW FULL TIME



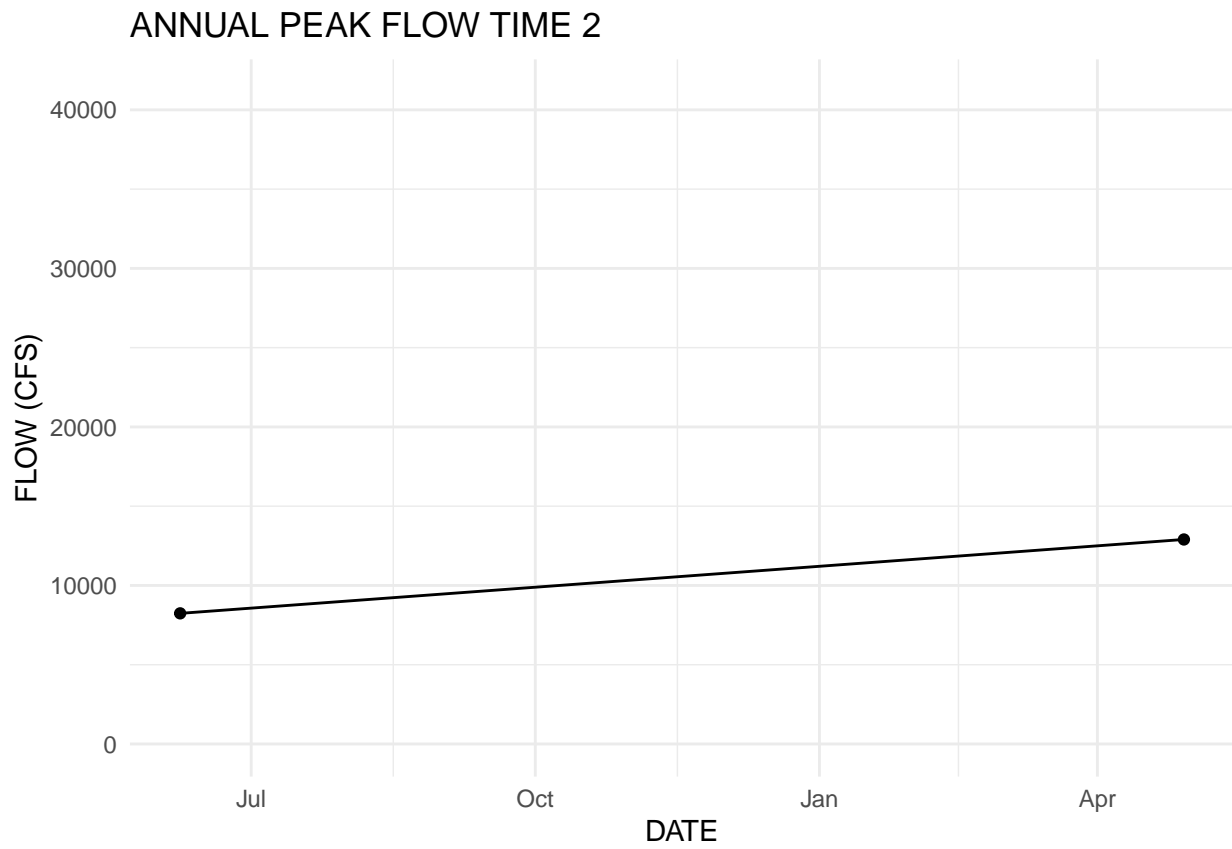
```
##  
## $peak_flow_time1_plot
```

ANNUAL PEAK FLOW TIME 1

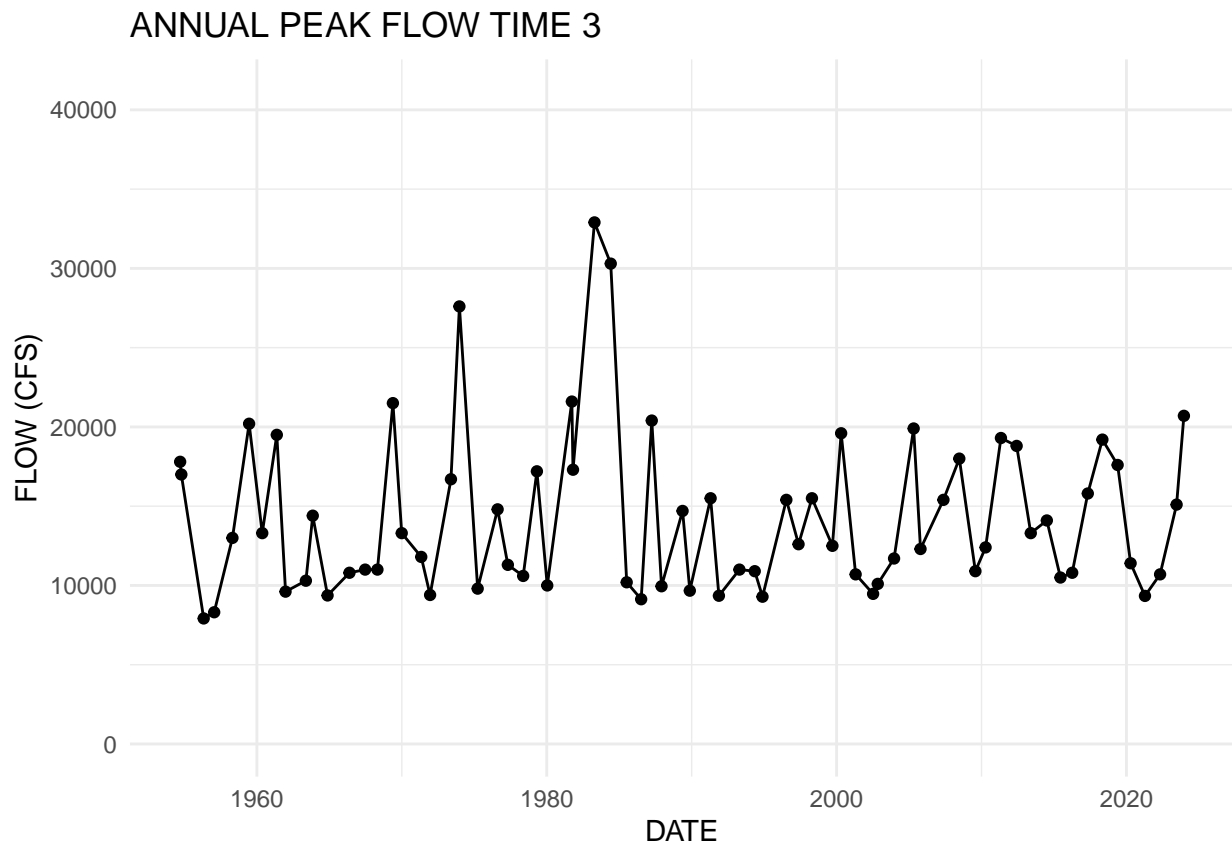


##

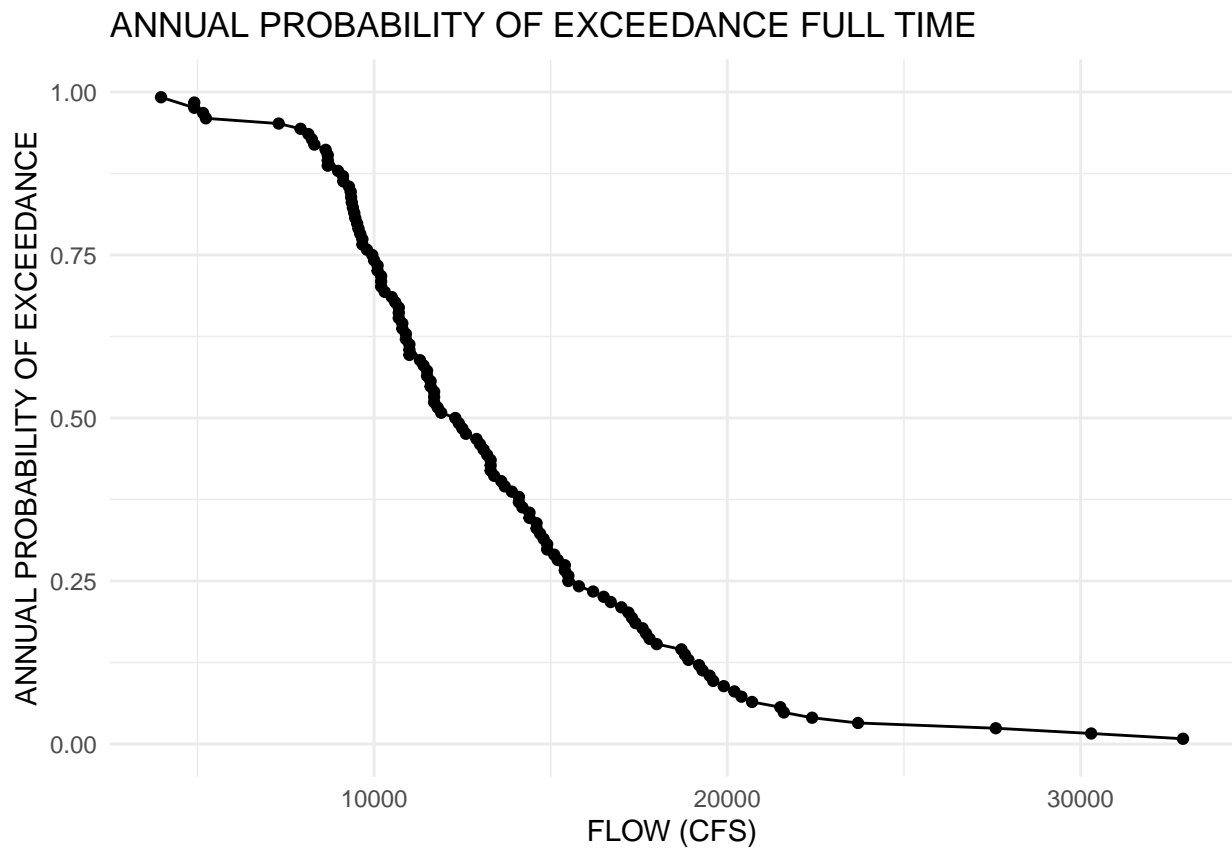
\$peak_flow_time2_plot



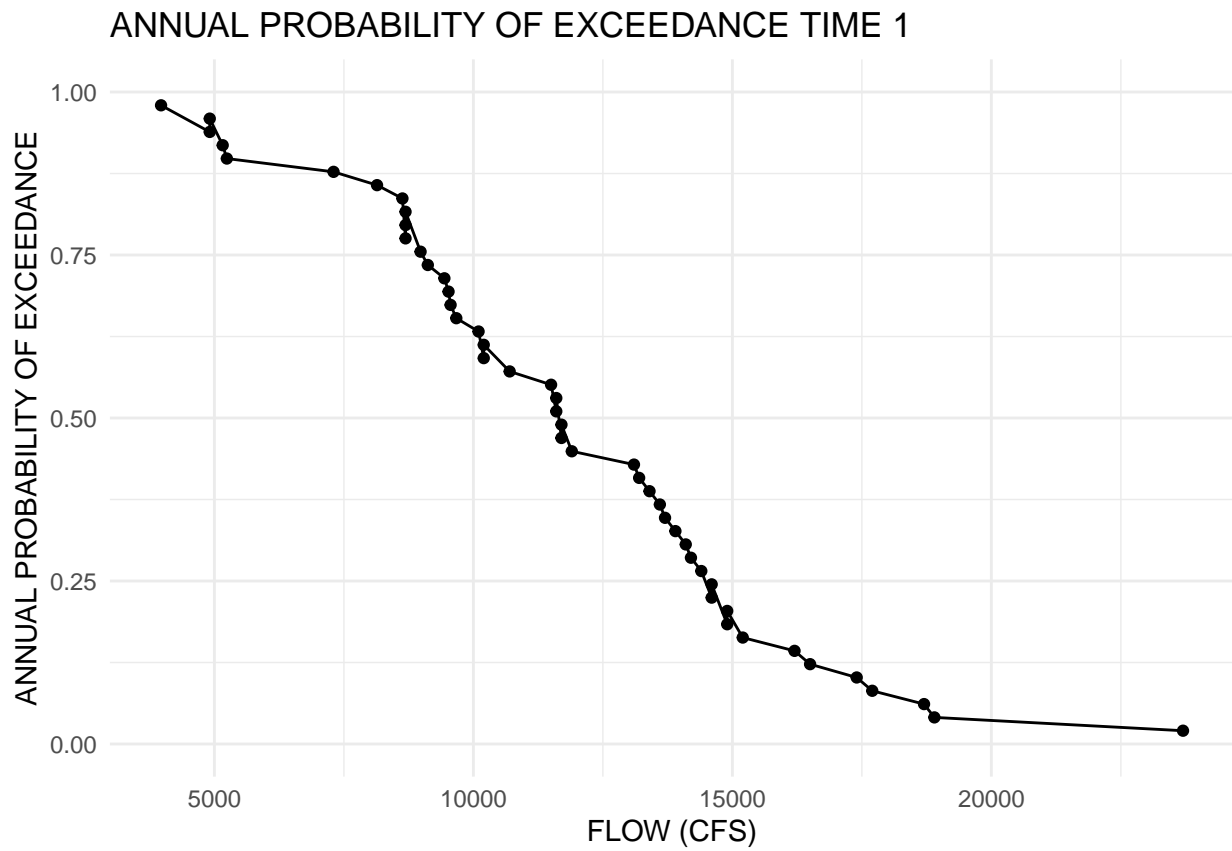
```
##  
## $peak_flow_time3_plot
```



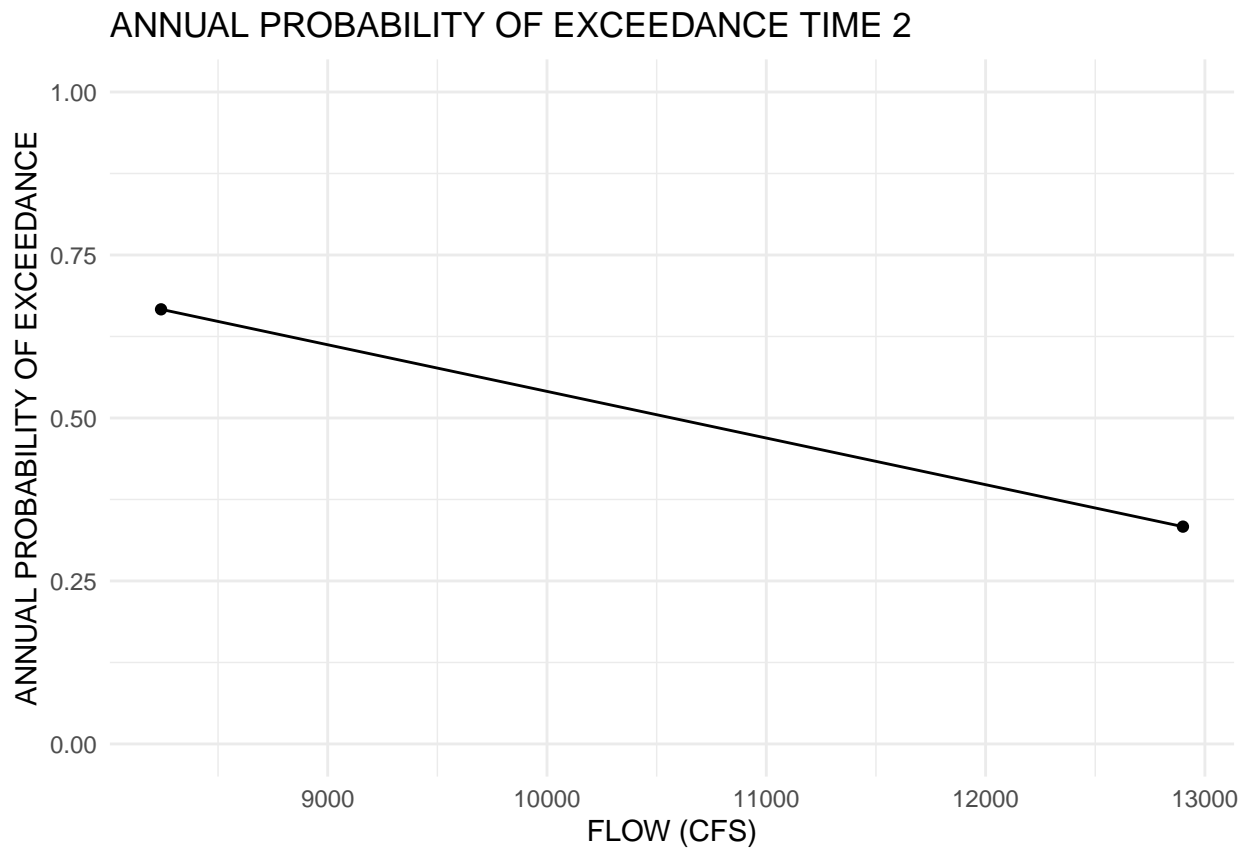
```
##  
## $APE_full_plot
```



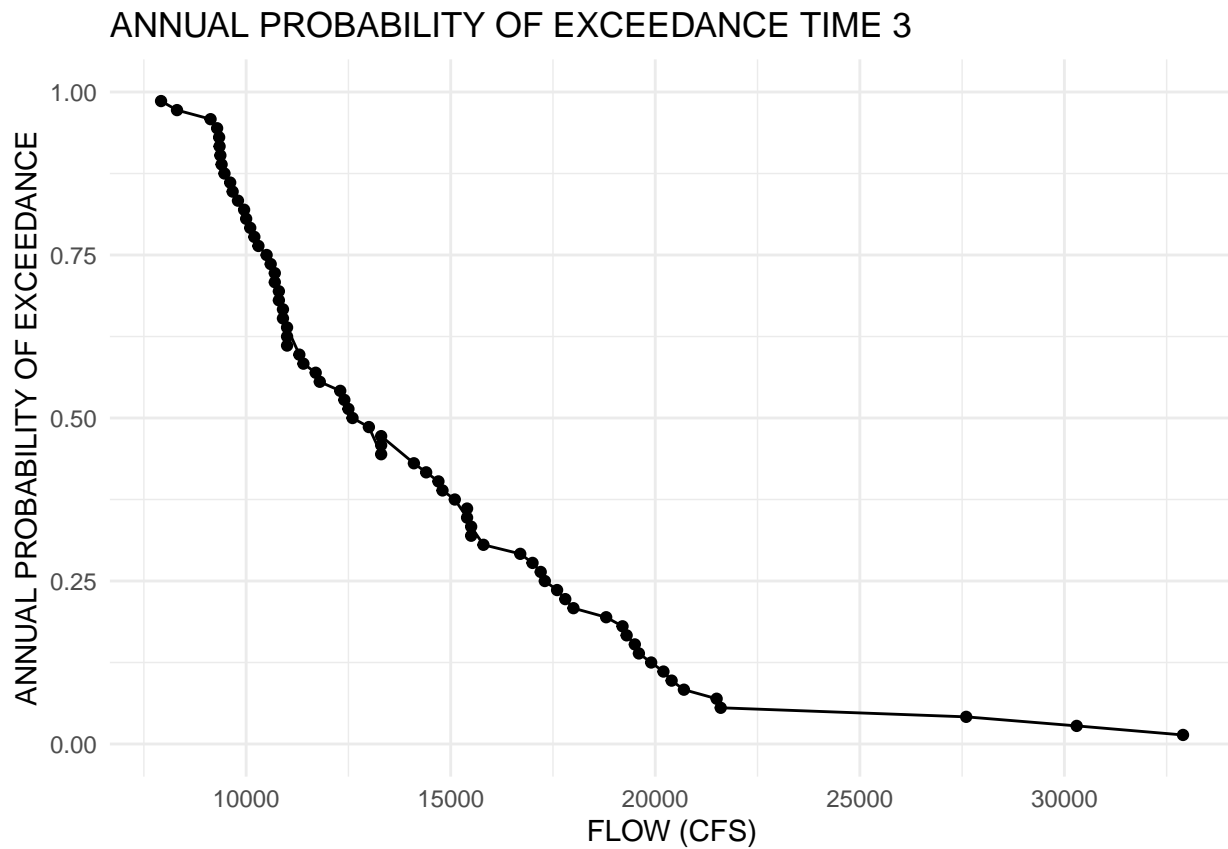
```
##  
## $APE_time1_plot
```



```
##  
## $APE_time2_plot
```



```
##  
## $APE_time3_plot
```

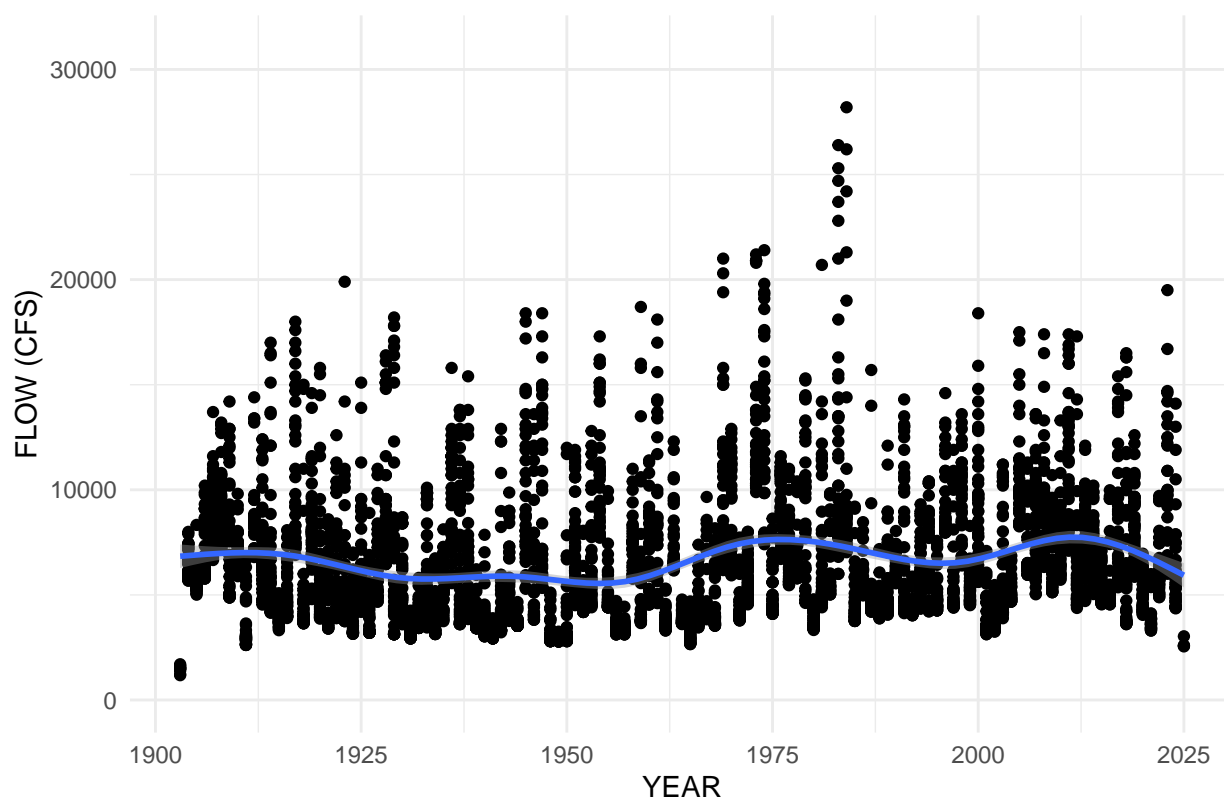


Annual High Flow Manipulation and Visualization

```
## $high_flow_full_plot
```

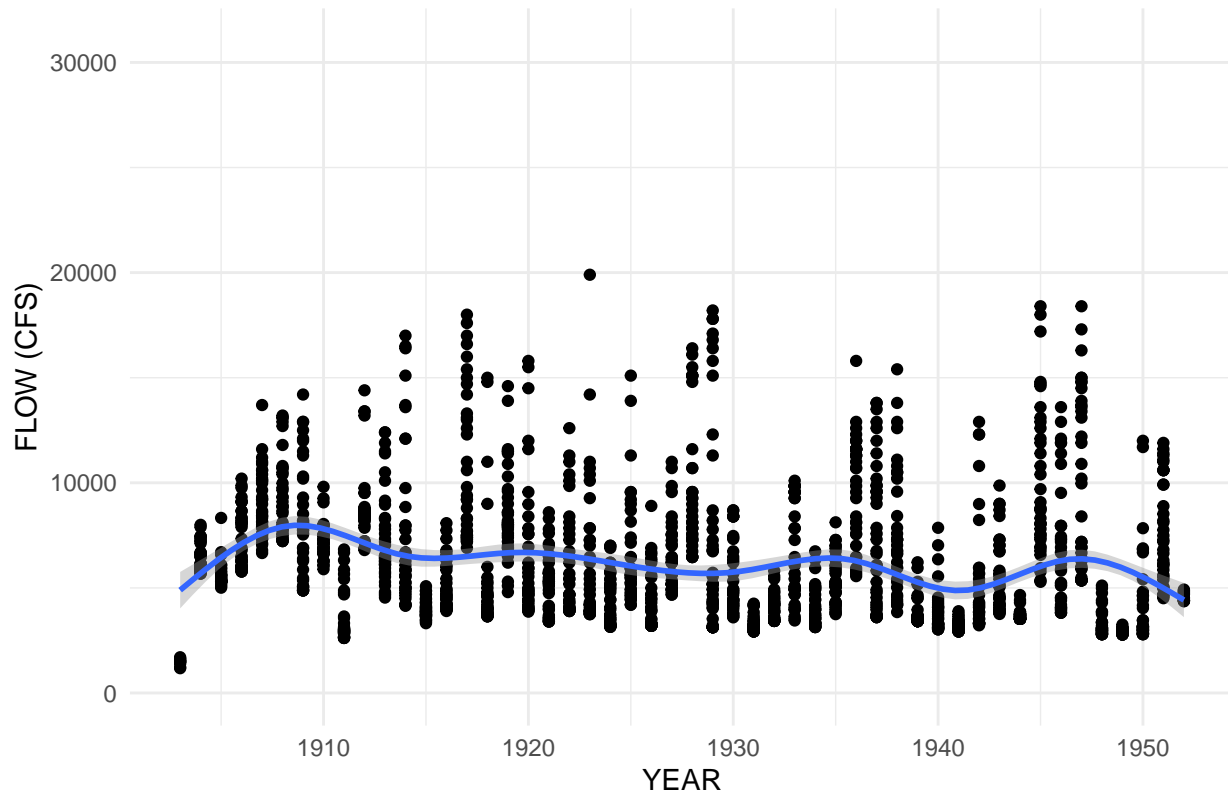
```
## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'
```

TOP 10% DAILY FLOW PER YEAR FULL TIME



```
##  
## $high_flow_time1_plot  
## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'
```

TOP 10% DAILY FLOW PER YEAR TIME 1

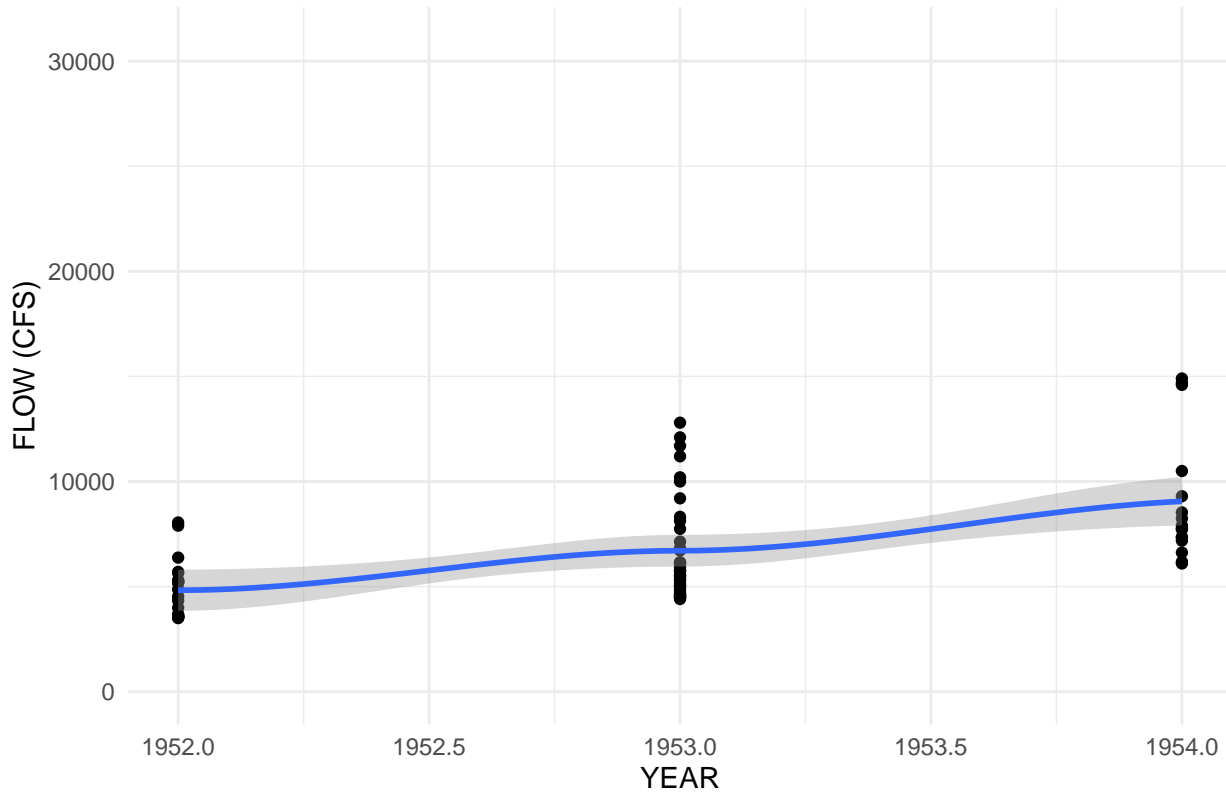


```
##
## $high_flow_time2_plot
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : pseudoinverse used at 1952
## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : neighborhood radius 1.01
## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : reciprocal condition number 0
## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : There are other near singularities as well. 4.0401
## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : pseudoinverse used at
## 1952
## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : neighborhood radius
## 1.01
## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : reciprocal condition
## number 0
```



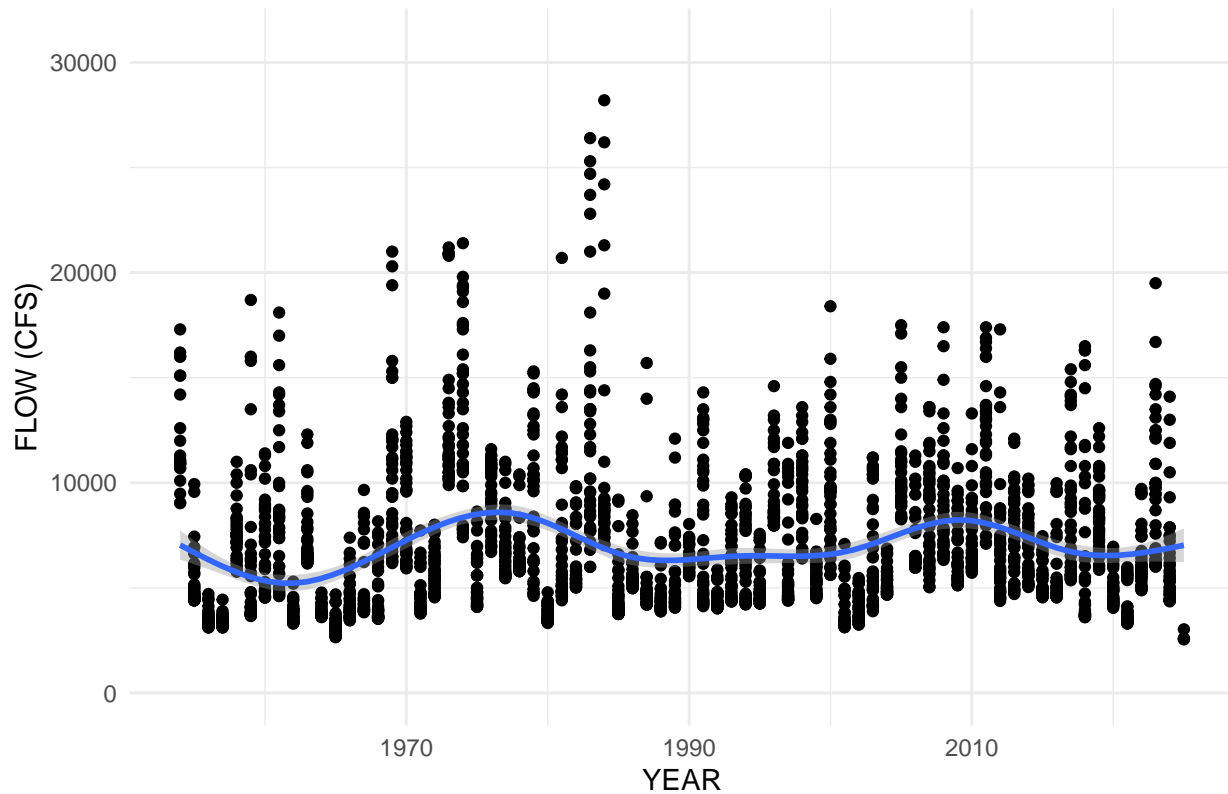
```
## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : There are other near
## singularities as well. 4.0401
```

TOP 10% DAILY FLOW PER YEAR TIME 2



```
##
## $high_flow_time3_plot
## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'
```

TOP 10% DAILY FLOW PER YEAR TIME 3



IHA Prestatistical Analysis Manipulation

DF Stats Create

IHA Stats

IHA Group 3 Stats

Field Measurements Manipulation and Stats

Measure of Center Stats

Peak Flow Manipulation and Stats

High Flow Stats

Schmidt and Wilcock 2008 Metrics

Stats DF Export