# Hydrology Analysis

### Configuration

```
inputFile <- './Data_Raw/MercedHI_Q_T_2022023.txt' # Set variable appropriate file
```

### Installation

Only necessary to run this cell once to set up local machine

```
install.packages('tidyverse', repos='http://cran.us.r-project.org')
```

```
##
## The downloaded binary packages are in
##  /var/folders/r_/4w9b5lnx7n542qjy74wfz0hr0000gn/T//RtmpAzXEDM/downloaded_packages
```

### Set up Environment

Loads the Tidyverse and imports necessary functions

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.3     v purrr   0.3.4
## v tibble  3.1.1     v dplyr   1.0.5
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
source('./Rcode/utils.R')
```

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

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

### Read Data

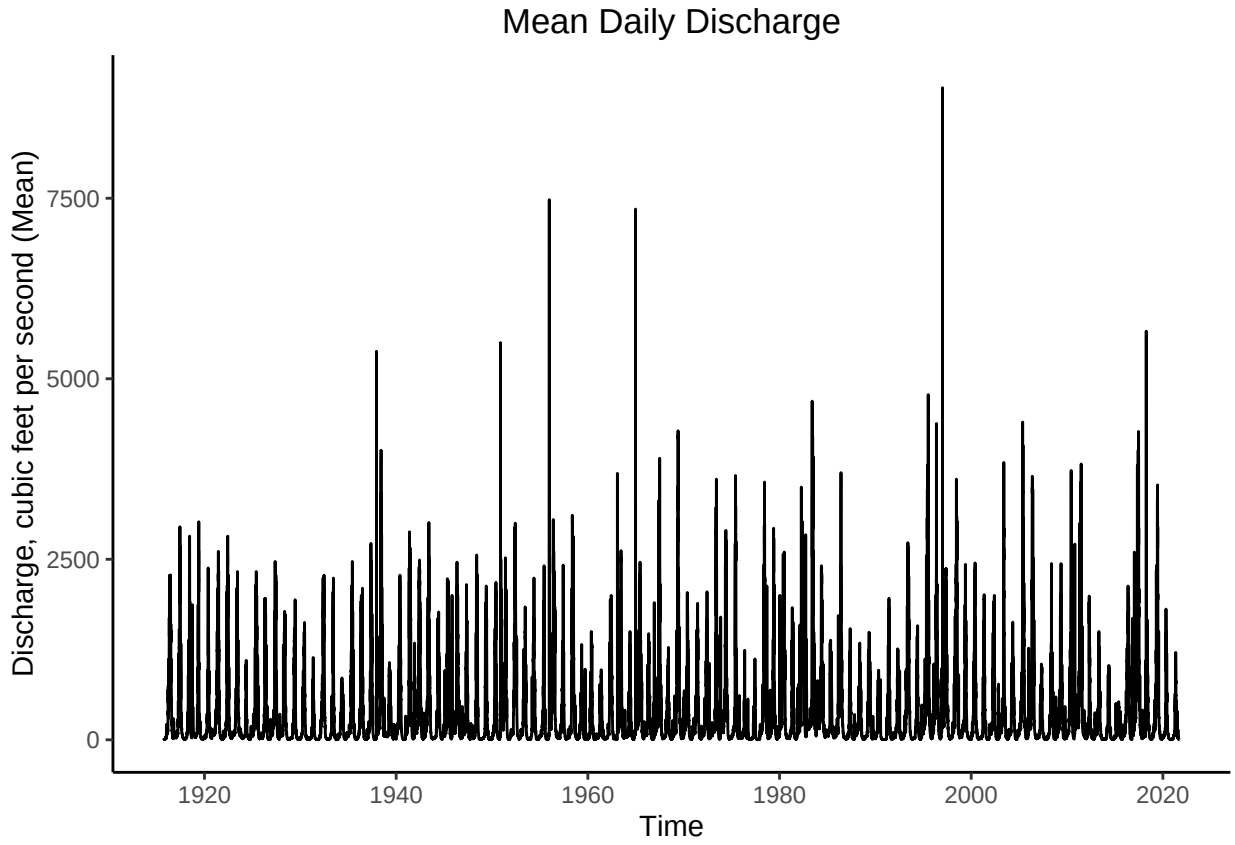
To run the programs on your desired file, set `inputFile` to the appropriate file path

```
data <- readUsgsData(inputFile, startDate = ymd('1915-10-01'))
```

```
## Rows: 38755 Columns: 2-- Column specification --------------------------------------------------------
## Delimiter: "\t"
## dbl  (1): X12
## date (1): X3
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

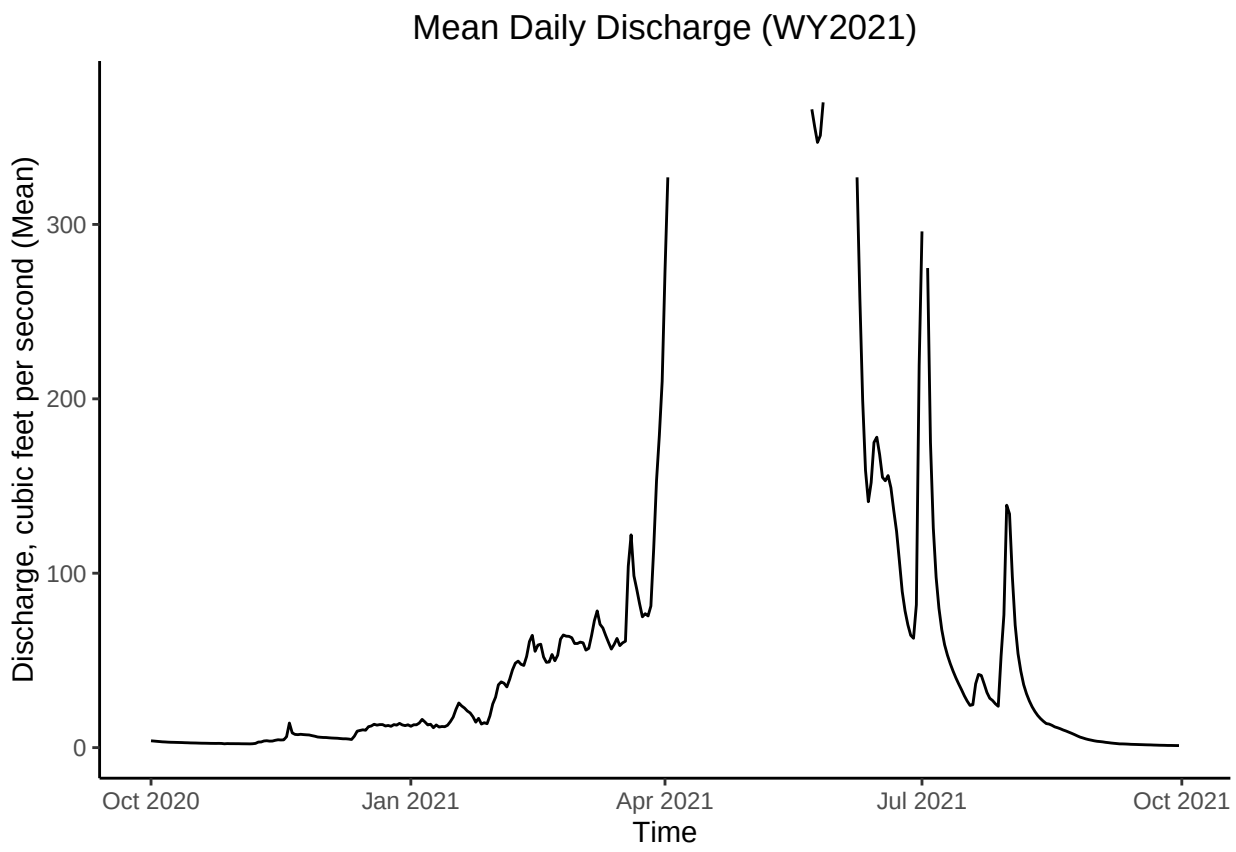
### Visualize Input Data

```
ggplot(data, aes(x=date, y=dmq, group=1)) +
  geom_line() +
  theme_classic() +
  ggtitle("Mean Daily Discharge") +
  theme(plot.title = element_text(hjust = 0.5)) +
  xlab("Time") +
  ylab("Discharge, cubic feet per second (Mean)")
```

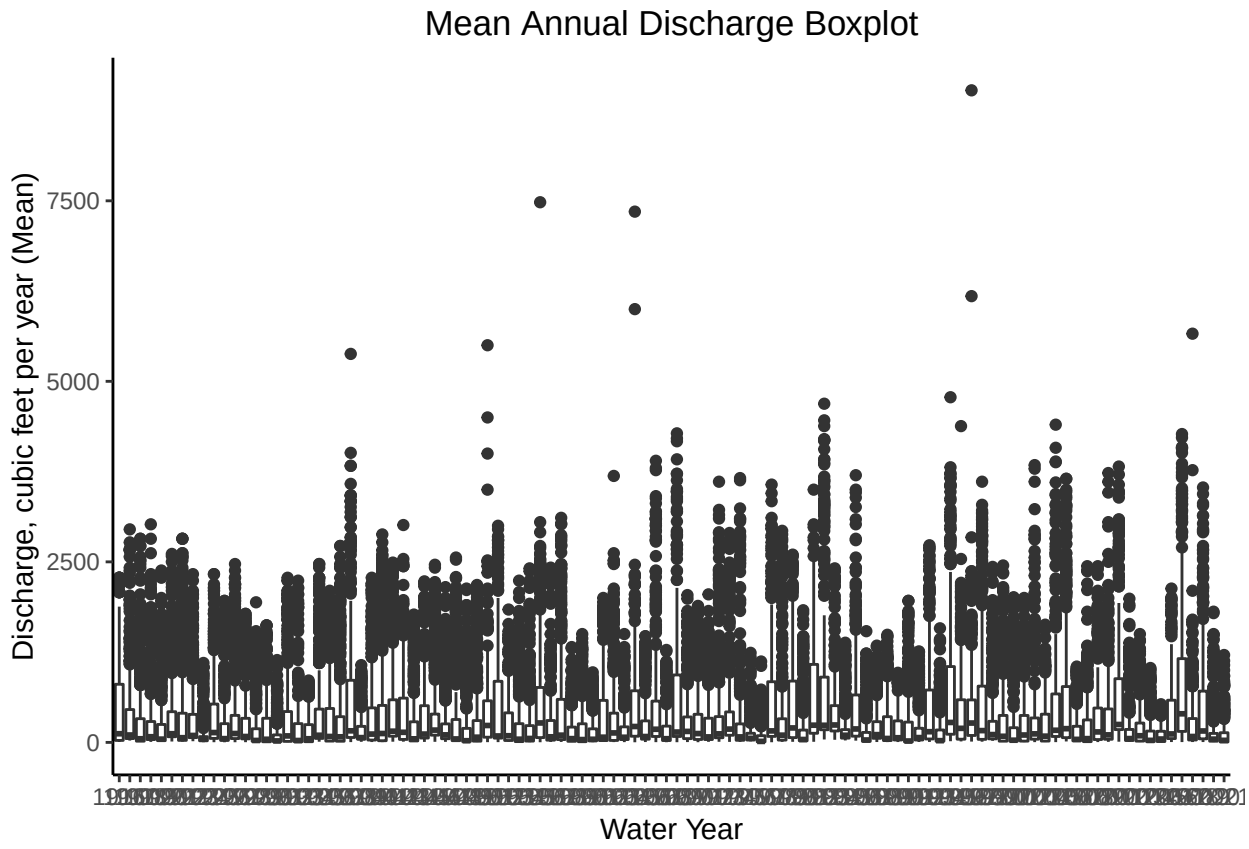



```
ggplot(data %>% filter(ifelse(mth >= 10, yr + 1, yr) == 2021), aes(x=date, y=dmq, group=1)) +
  geom_line() +
  ylim(NA, 375) +
```

```
  theme_classic() +
  ggtitle("Mean Daily Discharge (WY2021)") +
  theme(plot.title = element_text(hjust = 0.5)) +
  xlab("Time") +
  ylab("Discharge, cubic feet per second (Mean)")
```




```
ggplot(data %>% groupByWaterYear(), aes(x=as.character(waterYear), y=dmq)) +
  geom_boxplot() +
  theme_classic() +
  ggtitle("Mean Annual Discharge Boxplot") +
  theme(plot.title = element_text(hjust = 0.5)) +
  xlab("Water Year") +
  ylab("Discharge, cubic feet per year (Mean)")
```




**Generate and Analyze Output**

There are two cells each R program: the first will generate output and store it in the `./Output/` folder, and the second gives space to produce visualizations of the output

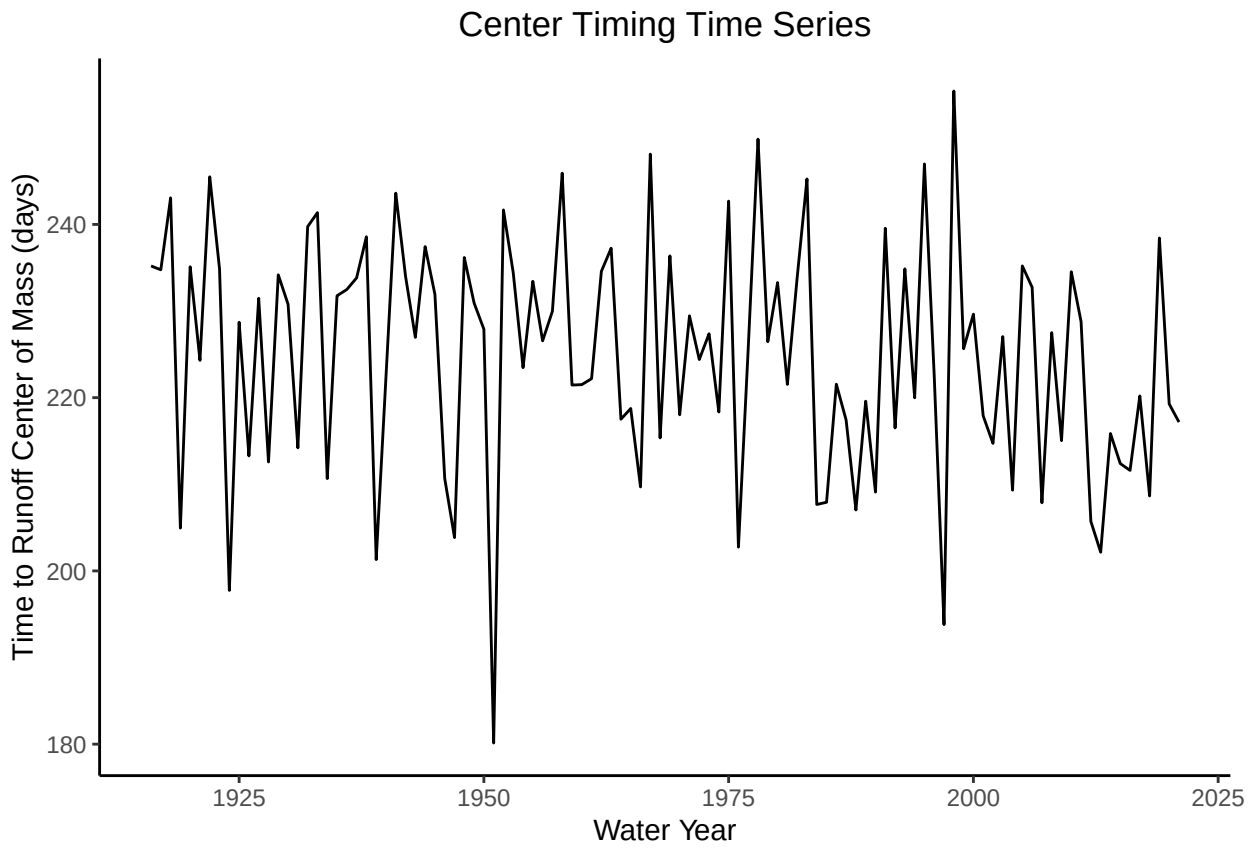
**surfwtr.R** Generate output

```
source('./Rcode/surfwtr.R')
surfwtrOutput <- surfwtr(data)
summarizeOutput(surfwtrOutput)
```

```
##                    mean      stdDev          min       median          max
## waterYear 1968.5000000  30.7435630 1916.0000000 1968.5000000 2021.0000000
## dur        365.2547170   0.4377719  365.0000000  365.0000000  366.0000000
## mdq        353.3120778 163.4815032   84.9175342  329.5944906  876.9132329
## cmt        224.8202139  13.5844564  180.1297975  226.5101810  255.3947903
## frsmq        0.8304872   0.0641086    0.5979763    0.8389947    0.9469093
## # A tibble: 6 x 5
##   waterYear   dur   mdq   cmt frsmq
##       <dbl> <int> <dbl> <dbl> <dbl>
## 1      1916   366  460.  235. 0.860
## 2      1917   365  405.  235. 0.853
## 3      1918   365  334.  243. 0.897
## 4      1919   365  308.  205. 0.824
```

```
## 5      1920   366  255.  235. 0.896
## 6      1921   365  396.  224. 0.813
```

```
# Space to visualize surfwtrOutput
ggplot(surfwtrOutput, aes(x=waterYear, y=cmt)) +
  geom_line() +
  theme_classic() +
  ggtitle("Center Timing Time Series") +
  theme(plot.title = element_text(hjust = 0.5)) +
  xlab("Water Year") +
  ylab("Time to Runoff Center of Mass (days)")
```



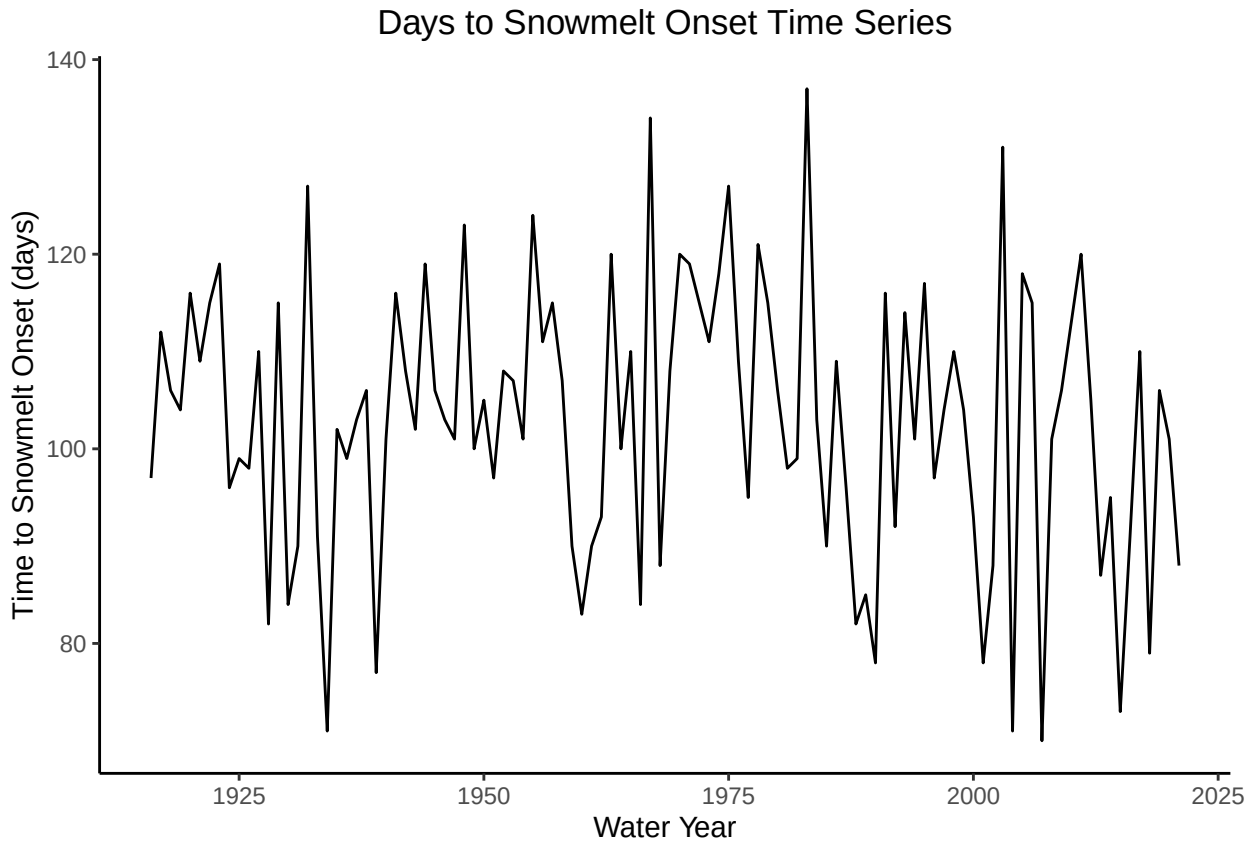

**snwpulse.R** Generate output

```
source('./Rcode/snwpulse.R')
snwpulseOutput <- snwpulse(data)
summarizeOutput(snwpulseOutput)
```

```
##                  mean      stdDev          min      median       max
## waterYear   1968.5000    30.74356    1916.0000   1968.5000  2021.000
## mdq          507.3037   232.28708     123.0429    478.3146  1235.983
## snwpulse  -34097.7462 19398.92110 -111131.2500 -30359.0792 -4456.488
## dypulse      102.9151    14.25423      70.0000    104.0000   137.000
## # A tibble: 6 x 4
```

```
##   waterYear   mdq snwpulse dypulse
##       <dbl> <dbl>    <dbl>   <dbl>
## 1      1916  691.  -42860.      97
## 2      1917  574.  -46141.     112
## 3      1918  483.  -37868.     106
## 4      1919  414.  -29451.     104
## 5      1920  377.  -29746.     116
## 6      1921  565.  -34475.     109
```

```
# Space to visualize snwpulseOutput
ggplot(snwpulseOutput, aes(x=waterYear, y=dypulse)) +
  geom_line() +
  theme_classic() +
  ggtitle("Days to Snowmelt Onset Time Series") +
  theme(plot.title = element_text(hjust = 0.5)) +
  xlab("Water Year") +
  ylab("Time to Snowmelt Onset (days)")
```




**lwflow.R** Generate output

```
source('./Rcode/lwflow.R')
lwflowOutputs <- lwflow(data) # Returns list of tables
invisible(lapply(lwflowOutputs, summarizeOutput))
```

```
##                mean     stdDev        min     median       max
```

```
## waterYear 1968.5000  30.74356 1916.00000 1968.5000 2021.0000
## amQ        353.3121 163.48150   84.91753  329.5945  876.9132
## # A tibble: 6 x 2
##   waterYear   amQ
##       <dbl> <dbl>
## 1      1916  460.
## 2      1917  405.
## 3      1918  334.
## 4      1919  308.
## 5      1920  255.
## 6      1921  396.
##                mean    stdDev         min      median       max
## waterYear 1968.50000 30.743563 1916.000000 1968.500000 2021.0000
## m3w         33.97025 32.825692    2.900000   21.166667  159.3333
## m7w         35.03976 33.962962    3.000000   21.500000  165.0000
## m14w        36.54613 36.081029    3.135714   21.378571  164.7857
## m3s         17.29513 20.972600    1.186667    9.426667  151.3333
## m7s         19.22662 23.904462    1.240000   10.290000  170.5714
## m14s        22.30961 26.810415    1.385000   12.249286  175.1429
## m7dw        83.28302 21.982571   61.000000   76.500000  152.0000
## m7ds       359.72642  7.234222  331.000000  364.000000  364.0000
## # A tibble: 6 x 9
##   waterYear   m3w   m7w  m14w   m3s   m7s  m14s  m7dw  m7ds
##       <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1      1916   9    8.74  8.87  18   20.6  28.4     61   356
## 2      1917  26   27    28.2   27.7 31.6  36.4    105   364
## 3      1918  10.7 11.7  12.2   26   27.6  36.4    103   347
## 4      1919  34   34.9  35.3    7    7.29  7.96   126   361
## 5      1920  13.3 11.2  10.8   23.3 23.4  24.5     61   357
## 6      1921  67.3 69.3  70.6   13   14.4  16       69   364
##          mean    stdDev         min    median         max
## exp   0.50000  0.287323 0.009345794  0.500000   0.9906542
## m3w  33.97025 32.825692 2.900000000 21.166667 159.3333333
## m7w  35.03976 33.962962 3.000000000 21.500000 165.0000000
## m14w 36.54613 36.081029 3.135714286 21.378571 164.7857143
## m3s  17.29513 20.972600 1.186666667  9.426667 151.3333333
## m7s  19.22662 23.904462 1.240000000 10.290000 170.5714286
## m14s 22.30961 26.810415 1.385000000 12.249286 175.1428571
## # A tibble: 6 x 7
##       exp   m3w   m7w  m14w   m3s   m7s  m14s
##     <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 0.00935  2.90  3     3.14  1.19  1.24  1.38
## 2 0.0187   4.13  4.40  4.86  1.57  1.90  2.58
## 3 0.0280   4.23  4.94  4.91  1.77  2.03  2.59
## 4 0.0374   4.88  5.10  5     2.33  2.63  2.98
## 5 0.0467   4.93  5.14  5.36  2.47  2.71  3.08
## 6 0.0561   5     5.33  5.50  2.60  2.72  3.22
##            mean     stdDev      min   median       max
## xxp   0.3062217  0.3393862 0.020000 0.121450   0.95000
## m3w  21.9347409 28.1079826 4.147333 6.780387 102.33333
## m7w  23.2319348 30.6565110 4.476000 6.982052 115.25714
## m14w 24.9434792 34.6168486 4.864143 7.345129 133.71071
## m3s  11.3906927 16.8285757 1.594667 3.456343  62.31167
## m7s  12.8140452 19.2882475 1.918000 3.740108  71.22143
```

```
## m14s 15.1748391 23.0126182 2.579571 4.235216  82.99786
## # A tibble: 6 x 7
##     xxp   m3w   m7w  m14w   m3s   m7s  m14s
##   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1  0.95 102.  115.  134.  62.3   71.2  83.0
## 2  0.9   71.4  74.6  80.4 44.5   51.9  65.1
## 3  0.8   52.9  54.9  57.4 25.3   26.7  30.7
## 4  0.7   42.2  41.7  40.1 17     19.5  22.4
## 5  0.6   30.7  32    34.6 12.3   13.1  15.1
## 6  0.5   21.2  21.5  21.4  9.43  10.3  12.2
```

```
# Space to visualize lwflowOutputs
```

**hiflow.R** Generate output

```
source('./Rcode/hiflow.R')
hiflowOutputs <- hiflow(data) # Returns list of tables
invisible(lapply(hiflowOutputs, summarizeOutput))
```

```
##                mean     stdDev       min   median      max
## waterYear 1968.5000   30.74356 1916.0000 1968.500 2021.000
## m3        2365.0031 1037.74536  500.6667 2253.333 6016.667
## m7        2099.5418  837.44642  469.8571 2086.429 4167.143
## m10       1979.1019  788.69941  466.5000 1907.000 3836.000
## m14       1870.8497  756.25014  436.6429 1784.286 3672.143
## m3d        231.8396   34.74345   52.0000  237.000  283.000
## m14d       244.5566   16.23170  199.0000  245.000  283.000
## # A tibble: 6 x 7
##   waterYear    m3    m7   m10   m14   m3d  m14d
##       <dbl> <dbl> <dbl> <dbl> <dbl> <int> <int>
## 1      1916 2247. 2180  2130  2088.   256   262
## 2      1917 2797. 2511. 2444  2446.   253   263
## 3      1918 2787. 2670  2586  2409.   256   261
## 4      1919 2660. 2440. 2361. 2133.   241   242
## 5      1920 2330  2039. 1817  1791.   234   243
## 6      1921 2563. 2409. 2271  2004.   255   258
##          mean      stdDev          min   median          max
## exp     0.500    0.287323 9.345794e-03    0.500    0.9906542
## m3   2365.003 1037.745358 5.006667e+02 2253.333 6016.6666667
## m7   2099.542  837.446425 4.698571e+02 2086.429 4167.1428571
## m10 1979.102  788.699406 4.665000e+02 1907.000 3836.0000000
## m14 1870.850  756.250135 4.366429e+02 1784.286 3672.1428571
## # A tibble: 6 x 5
##     exp    m3    m7   m10   m14
##   <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 0.991  501.  470.  466.  437.
## 2 0.981  835   697.  686.  644.
## 3 0.972  900.  788.  728.  726
## 4 0.963  917.  812.  760.  743.
## 5 0.953  947   821   763.  761.
## 6 0.944  967.  854.  807.  776.
##             mean       stdDev      min     median      max
```

```
## xxp    0.3062217    0.3393862   0.0200    0.12145    0.950
## m3  3326.5968944 1429.1417443 954.1167 3724.31217 5142.533
## m7  2814.4638810 1115.1541440 832.6000 3243.94243 3981.943
## m10 2664.9860772 1059.4310570 778.7350 3051.53490 3758.220
## m14 2542.6053381 1021.2499294 766.4679 2940.01882 3591.271
## # A tibble: 6 x 5
##     xxp    m3    m7   m10   m14
##   <dbl> <dbl> <dbl> <dbl> <dbl>
## 1  0.95  954.  833.  779.  766.
## 2  0.9  1123.  991.  951.  886.
## 3  0.8  1430  1307. 1212. 1163.
## 4  0.7  1780. 1561. 1510. 1409.
## 5  0.6  2007. 1839. 1766  1670.
## 6  0.5  2253. 2086. 1907  1784.
```

```
# Space to visualize hiflowOutputs
```

**fldur.R** Generate output

```
source('./Rcode/fldur.R')
fldurOutput <- fldur(data) %>% ungroup()
summarizeOutput(fldurOutput)
```

```
##             mean      stdDev         min     median        max
## pbs    50.000000   42.757843 0.010000000  50.000000   99.99000
## qfd  1166.215466 1739.821524 1.236154000 100.000000 6329.99400
## dqfd    3.301077    4.924721 0.003499045   0.283059   17.91761
## # A tibble: 6 x 3
##     pbs   qfd  dqfd
##   <dbl> <dbl> <dbl>
## 1  0.01 6330. 17.9
## 2  0.03 4475. 12.7
## 3  0.05 4206. 11.9
## 4  0.1  3873. 11.0
## 5  0.5  3120   8.83
## 6  1    2720   7.70
```

```
# Space to visualize fldurOutput
```