Assignment 8: Time Series Analysis

Student Name

OVERVIEW

This exercise accompanies the lessons in Water Data Analytics on time series analysis

Directions

1. Change "Student Name" on line 3 (above) with your name.
2. Work through the steps, **creating code and output** that fulfill each instruction.
3. Be sure to **answer the questions** in this assignment document.
4. When you have completed the assignment, **Knit** the text and code into a single PDF file.
5. After Knitting, check your PDF against the key and then submit your assignment completion survey at https://forms.gle/dKEutwXiFewkSTwN9

Having trouble? See the assignment's answer key if you need a hint. Please try to complete the assignment without the key as much as possible - this is where the learning happens!

Target due date: 2022-03-29

Setup

1. Verify your working directory is set to the R project file. Load the tidyverse, lubridate, trend, forecast, and dataRetrieval packages. Set your ggplot theme (can be theme_classic or something else).

```
getwd()
```

```
## [1] "/Users/Jack/Documents/Duke/Spring 2022/Water Data Analytics/Water_Data_Analytics_2022"
```

```
library(tidyverse)
```

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

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.8
## v tidyr   1.2.0     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()
```

```
library(lubridate)
```

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

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

```r
library(trend)
library(forecast)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

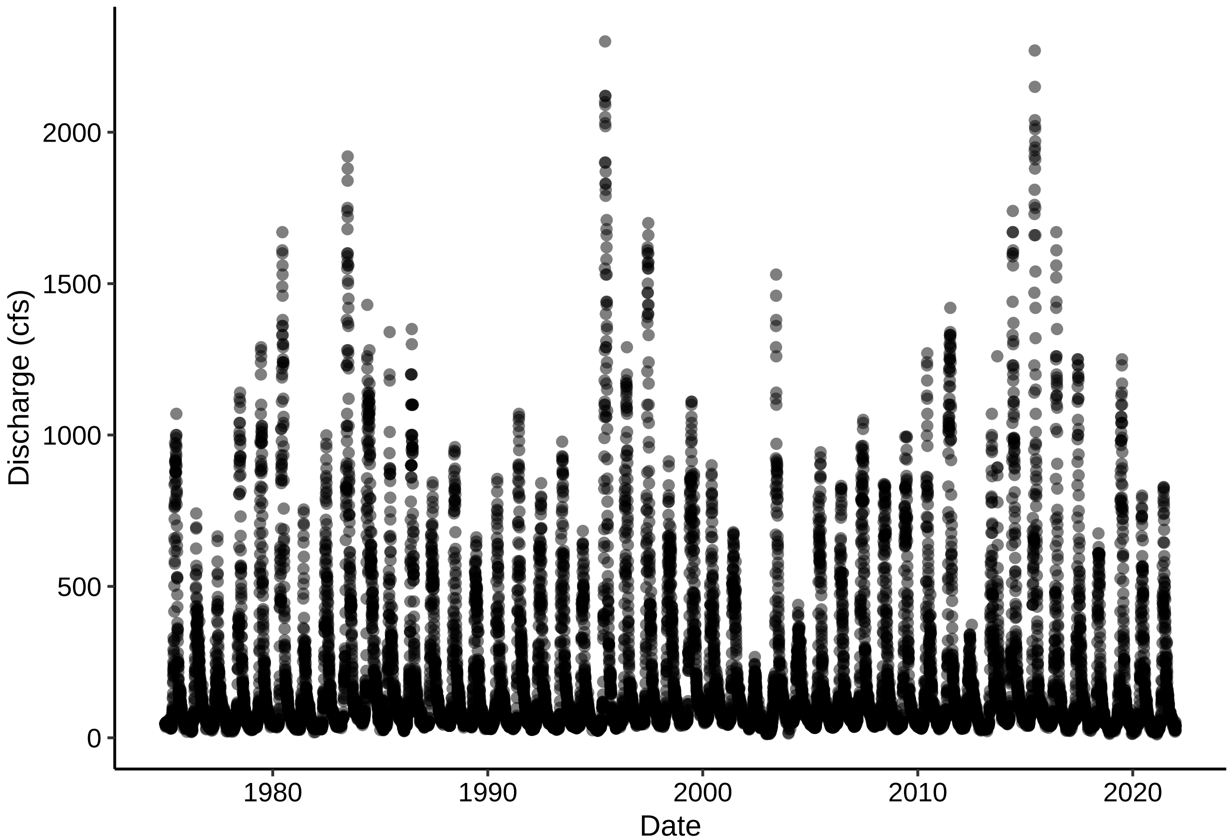
```r
library(dataRetrieval)

theme_set(theme_classic() +
  theme(axis.text = element_text(color = "black", size = 10),
        legend.position = "right"))
```

Data Import and Processing

2. Import discharge data (parameter 00060) from Clear Creek, Colorado (site 06719505) from the start of 1975 through the end of 2021.
3. Graph Clear Creek discharge over time.
4. Create a new data frame with the sum of monthly discharge in acre-feet per month.

```r
#import discharge data
ClearCreekDischarge <- readNWISdv(siteNumbers = "06719505",
                                  parameterCd = "00060", #cfs
                                  startDate = "1975-01-01",
                                  endDate = "2021-12-31")
#rename the columns that have useless names
names(ClearCreekDischarge)[4:5] <- c("Discharge", "Approval Code")
#plot discharge data over time
ClearCreekDischargePlot <- ggplot(ClearCreekDischarge,
                                  aes(x = Date, y = Discharge)) +
  geom_point(alpha = 0.5) +
  labs(x = "Date", y = "Discharge (cfs)")
ClearCreekDischargePlot
```




```r
#create a new df with monthly discharge in acre-feet
ClearCreekDischarge_monthly <- ClearCreekDischarge %>%
  mutate(Year = year(Date),
         Month = month(Date)) %>%
  group_by(Year, Month) %>%
  summarize(Discharge_acft.mo = sum(Discharge)*1.98347) #conversion cfs to acft
```

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

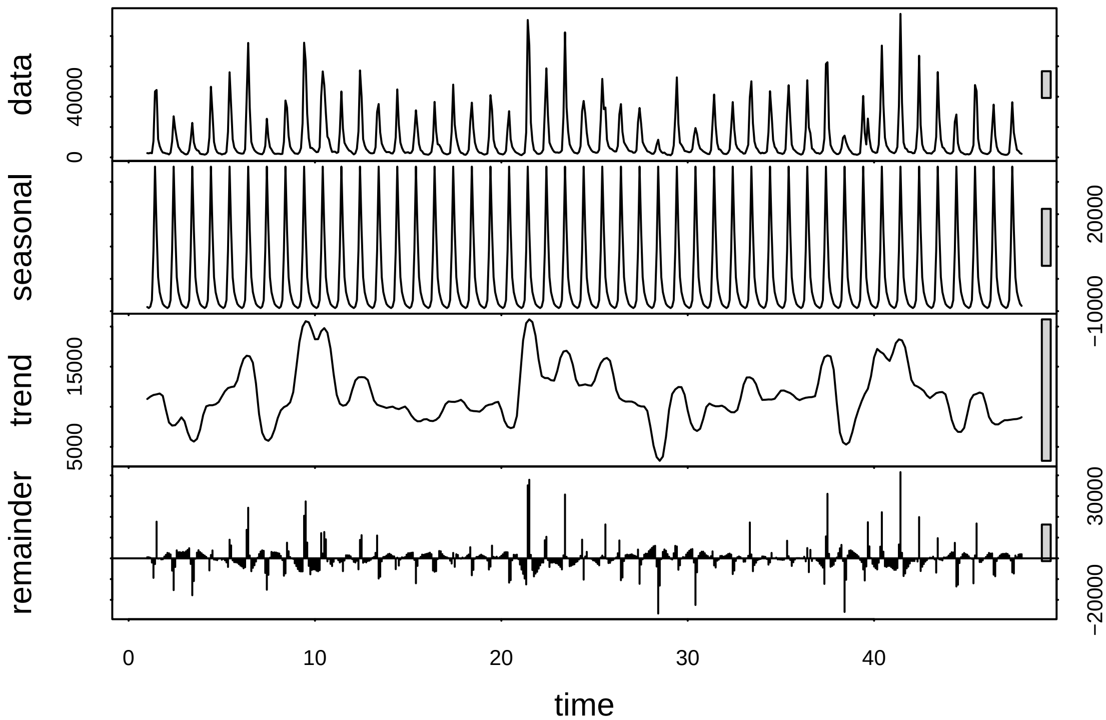
Time Series Decomposition

5. Create a time series of discharge from the monthly data frame. Make sure to add start and end dates like we did in class.
6. Decompose the time series using the `stl` function.
7. Visualize the decomposed time series.

```r
#create a time series
ClearCreek_ts <- ts(ClearCreekDischarge_monthly[[3]], frequency = 12)
#brackets are syntax, number inside is # of columns
#frequency of 12 bc monthly data

#decompose the ts with stl
ClearCreek_decomp <- stl(ClearCreek_ts, s.window = "periodic")

#visualize the decomp
plot(ClearCreek_decomp)
```



8. How do the seasonal and trend components of the decomposition compare to the Neuse River discharge dataset?

   Seasonal: The seasonal component of the Clear Creek decomposition is much much higher than that for the Neuse. Clear Creek is very highly seasonal (snowmelt runoff) and the Neuse is not very seasonal.

   Trend: Neither one has a particularly strong trend, but the trend for the Neuse is slightly stronger. Clear Creek does not have much of a trend at all, indicating a degree of stationarity.

Trend Analysis

Research question: Has there been a monotonic trend in discharge in Clear Creek over the period of study?

9. Run a Seasonal Mann-Kendall test on the monthly discharge data. Inspect the overall trend and the monthly trends.

```
#Run the seasonal mann-kendall test
ClearCreek_trend <- smk.test(ClearCreek_ts)

#now take a look at it
ClearCreek_trend
```

```
##
##  Seasonal Mann-Kendall trend test (Hirsch-Slack test)
##
## data:  ClearCreek_ts
## z = -0.23296, p-value = 0.8158
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##      S   varS
##    -89 142689
```

```r
summary(ClearCreek_trend)
```

```
##
##  Seasonal Mann-Kendall trend test (Hirsch-Slack test)
##
## data: ClearCreek_ts
## alternative hypothesis: two.sided
##
## Statistics for individual seasons
##
## H0
##                        S  varS    tau      z Pr(>|z|)
## Season 1:   S = 0   -26 11890 -0.024 -0.229 0.818658
## Season 2:   S = 0   -57 11891 -0.053 -0.514 0.607570
## Season 3:   S = 0   159 11891  0.147  1.449 0.147357
## Season 4:   S = 0   181 11891  0.167  1.651 0.098804 .
## Season 5:   S = 0   135 11891  0.125  1.229 0.219132
## Season 6:   S = 0    45 11891  0.042  0.403 0.686580
## Season 7:   S = 0  -117 11891 -0.108 -1.064 0.287432
## Season 8:   S = 0  -124 11890 -0.115 -1.128 0.259314
## Season 9:   S = 0  -169 11891 -0.156 -1.541 0.123405
## Season 10:   S = 0    4 11890  0.004  0.028 0.978051
## Season 11:   S = 0  -49 11891 -0.045 -0.440 0.659805
## Season 12:   S = 0  -71 11891 -0.066 -0.642 0.520918
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

10. Is there an overall monotonic trend in discharge over time? Are there monthly trends over time? If so, are they positive or negative?

    Whether or not there is a trend depends on the strictness of the tester - if the limit for a trend is a p-value of 0.05, then no there are no trends either overall or monthly. But if 0.1 is the cutoff, then yes there is one month that exhibits a trend: April. April is exhibiting a positive trend, which makes some sense as climate change has caused warmer spring weather earlier in the year. I can confirm anecdotally that April especially the past few years has been warmer with more melt than I remember as a kid and I'm sure the same has been occuring for longer than my lifetime.

Forecasting

Research question: can we predict discharge in Clear Creek moving into the future?

11. Run the auto.arima function on the Clear Creek time series to search for the best fit. Create an object that defines the best fit model.
12. Make a prediction into the future and plot the future predictions.

```r
#run arima function for best fit
auto.arima(ClearCreek_ts, trace = TRUE)
```

```
##
##  Fitting models using approximations to speed things up...
##
##  ARIMA(2,0,2)(1,1,1)[12] with drift         : Inf
##  ARIMA(0,0,0)(0,1,0)[12] with drift         : 11531.74
##  ARIMA(1,0,0)(1,1,0)[12] with drift         : 11207.99
##  ARIMA(0,0,1)(0,1,1)[12] with drift         : 11098.86
##  ARIMA(0,0,0)(0,1,0)[12]                    : 11529.75
```

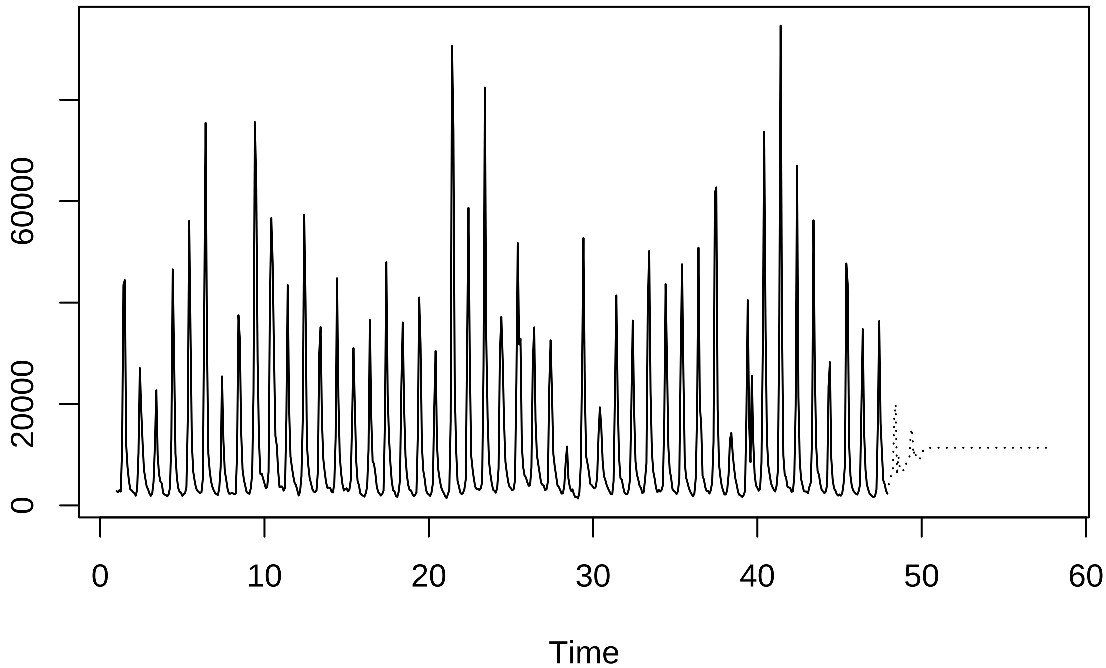
```
##  ARIMA(0,0,1)(0,1,0)[12] with drift         : 11360.07
##  ARIMA(0,0,1)(1,1,1)[12] with drift         : Inf
##  ARIMA(0,0,1)(0,1,2)[12] with drift         : 11097.52
##  ARIMA(0,0,1)(1,1,2)[12] with drift         : Inf
##  ARIMA(0,0,0)(0,1,2)[12] with drift         : Inf
##  ARIMA(1,0,1)(0,1,2)[12] with drift         : 11090.53
##  ARIMA(1,0,1)(0,1,1)[12] with drift         : 11091.09
##  ARIMA(1,0,1)(1,1,2)[12] with drift         : Inf
##  ARIMA(1,0,1)(1,1,1)[12] with drift         : Inf
##  ARIMA(1,0,0)(0,1,2)[12] with drift         : 11106.28
##  ARIMA(2,0,1)(0,1,2)[12] with drift         : 11093.22
##  ARIMA(1,0,2)(0,1,2)[12] with drift         : 11092.11
##  ARIMA(0,0,2)(0,1,2)[12] with drift         : 11090.91
##  ARIMA(2,0,0)(0,1,2)[12] with drift         : 11095.42
##  ARIMA(2,0,2)(0,1,2)[12] with drift         : Inf
##  ARIMA(1,0,1)(0,1,2)[12]                    : 11088.51
##  ARIMA(1,0,1)(0,1,1)[12]                    : 11089.08
##  ARIMA(1,0,1)(1,1,2)[12]                    : Inf
##  ARIMA(1,0,1)(1,1,1)[12]                    : Inf
##  ARIMA(0,0,1)(0,1,2)[12]                    : 11095.52
##  ARIMA(1,0,0)(0,1,2)[12]                    : 11104.26
##  ARIMA(2,0,1)(0,1,2)[12]                    : 11091.2
##  ARIMA(1,0,2)(0,1,2)[12]                    : 11090.08
##  ARIMA(0,0,0)(0,1,2)[12]                    : Inf
##  ARIMA(0,0,2)(0,1,2)[12]                    : 11088.9
##  ARIMA(2,0,0)(0,1,2)[12]                    : 11093.41
##  ARIMA(2,0,2)(0,1,2)[12]                    : Inf
##
##  Now re-fitting the best model(s) without approximations...
##
##  ARIMA(1,0,1)(0,1,2)[12]                    : Inf
##  ARIMA(0,0,2)(0,1,2)[12]                    : Inf
##  ARIMA(1,0,1)(0,1,1)[12]                    : Inf
##  ARIMA(1,0,2)(0,1,2)[12]                    : Inf
##  ARIMA(1,0,1)(0,1,2)[12] with drift         : Inf
##  ARIMA(0,0,2)(0,1,2)[12] with drift         : Inf
##  ARIMA(1,0,1)(0,1,1)[12] with drift         : Inf
##  ARIMA(2,0,1)(0,1,2)[12]                    : Inf
##  ARIMA(1,0,2)(0,1,2)[12] with drift         : Inf
##  ARIMA(2,0,1)(0,1,2)[12] with drift         : Inf
##  ARIMA(2,0,0)(0,1,2)[12]                    : Inf
##  ARIMA(2,0,0)(0,1,2)[12] with drift         : Inf
##  ARIMA(0,0,1)(0,1,2)[12]                    : Inf
##  ARIMA(0,0,1)(0,1,2)[12] with drift         : Inf
##  ARIMA(0,0,1)(0,1,1)[12] with drift         : Inf
##  ARIMA(1,0,0)(0,1,2)[12]                    : Inf
##  ARIMA(1,0,0)(0,1,2)[12] with drift         : Inf
##  ARIMA(1,0,0)(1,1,0)[12] with drift         : 11430.06
##
##  Best model: ARIMA(1,0,0)(1,1,0)[12] with drift
```

```
## Series: ClearCreek_ts
## ARIMA(1,0,0)(1,1,0)[12] with drift
##
```

```
## Coefficients:
##          ar1     sar1    drift
##       0.5288  -0.5061  -3.9277
## s.e.  0.0361   0.0363  37.7171
##
## sigma^2 = 56658295:  log likelihood = -5710.99
## AIC=11429.99   AICc=11430.06   BIC=11447.24
```

```
#best fit model
ClearCreek_arimafit <- arima(ClearCreek_ts, c(1,0,0),
                             seasonal = list(order = c(0,0,2),
                             period = 12))
#now let's predict something
ClearCreek_prediction <- predict(ClearCreek_arimafit, n.ahead = 10*12)

#plot future predictions
ts.plot(ClearCreek_ts, ClearCreek_prediction$pred, lty = c(1,3))
```



13. How did the forecasting for Clear Creek compare to the Neuse River?

    The Clear Creek prediction and the Neuse both are very small in comparison to historical flows, and both trickle down to the mean or median discharge in relatively short order - a few years into the future. Not a super useful prediction for either one, although Clear Creek's shows more typical seasonality for that stream, but not on the same scale as it should be.