Exploratory Data Analysis with SWMP

Marcus W. Beck[1] Todd D. O'Brien[2]

[1]ORISE, USEPA NHEERL Gulf Ecology Division
Email: beck.marcus@epa.gov

[2]NOAA/NMFS Copepod Project
Email: todd.obrien@noaa.gov

Objectives and agenda

- Objectives
  - What are some basic time series analysis techniques and when would you use them?
  - How are the data set up, what functions are used, and how are the results interpreted?

Objectives and agenda

- Objectives
  - What are some basic time series analysis techniques and when would you use them?
  - How are the data set up, what functions are used, and how are the results interpreted?
- Agenda
  - Common functions for exploratory data analysis
  - Analysis 1 - missing data and interpolation
  - Analysis 2 - smoothing and aggregation
  - Analysis 3 - basic trend analysis

Interactive portion

You can follow along in this module:

- dataset3
- script3

Interactive!

Common functions for EDA

What is exploratory data analysis (EDA)?

A general term that describes preliminary evaluation of a variable or multiple variables in a dataset to assess quantitative properties for further analysis

EDA can inform you of the types of variables (categorical, continuous), distribution of a variable (central tendency, spread), correlations between variables, and presence of outliers

You may decide to omit variables or specific observations, transform, standardize, etc.

Many of the same principles that apply to standard data analysis apply to time series analysis

Common functions for EDA

R has many functions available for EDA - see the R reference card for some ideas

We will cover a few basic techniques but keep in mind EDA is a general term and much of what we have already covered, and will cover, can be considered exploratory

Let's import some data:

```
# reload the SWMPr package if you started a new session
library(SWMPr)

# import data, qaqc, and subset
# change this path for the flash drive
path <- 'C:/data/dataset3'
nut_dat <- import_local(path, 'cbmmcnut')
nut_dat <- qaqc(nut_dat)
nut_dat <- subset(nut_dat, select = c('po4f', 'nh4f'))
```

Common functions for EDA

Perhaps the most useful function in R is 'summary'

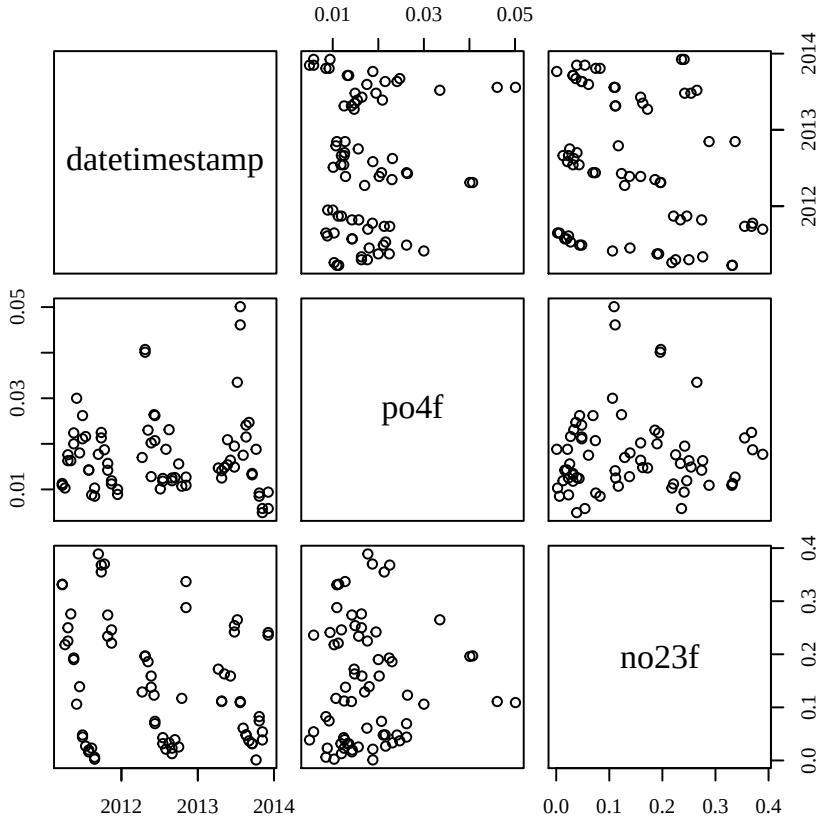
```
# get a summary of the data
summary(nut_dat)
```

```
##  datetimestamp                     po4f          no23f       chla_n
##  Min.   :2011-03-23 11:45:00   Min.   :0.00   Min.   :0   Min.   : 2
##  1st Qu.:2011-09-15 22:37:30   1st Qu.:0.01   1st Qu.:0   1st Qu.: 5
##  Median :2012-06-22 10:30:30   Median :0.02   Median :0   Median : 8
##  Mean   :2012-07-13 12:56:29   Mean   :0.02   Mean   :0   Mean   :17
##  3rd Qu.:2013-06-02 21:26:30   3rd Qu.:0.02   3rd Qu.:0   3rd Qu.:17
##  Max.   :2013-12-04 11:46:00   Max.   :0.05   Max.   :0   Max.   :98
##                                               NA's   :5   NA's   :4
```

Common functions for EDA

The pairs function is useful for evaluating simple bivariate correlations

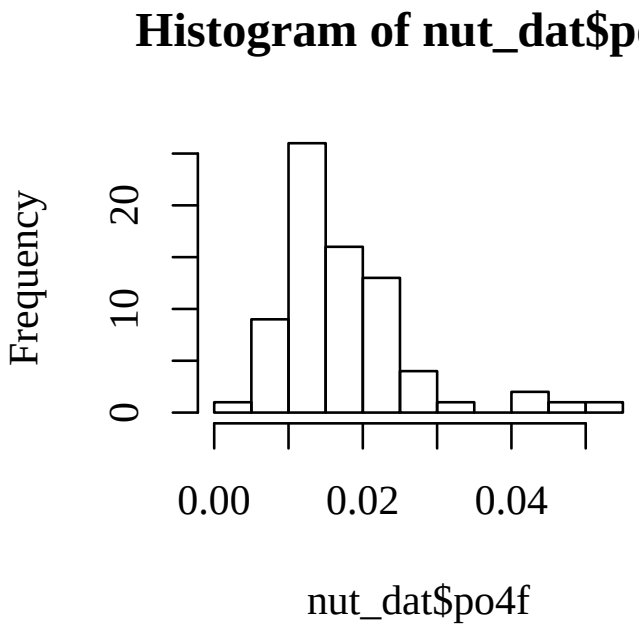
```
# bivariate scatterplots
pairs(nut_dat)
```

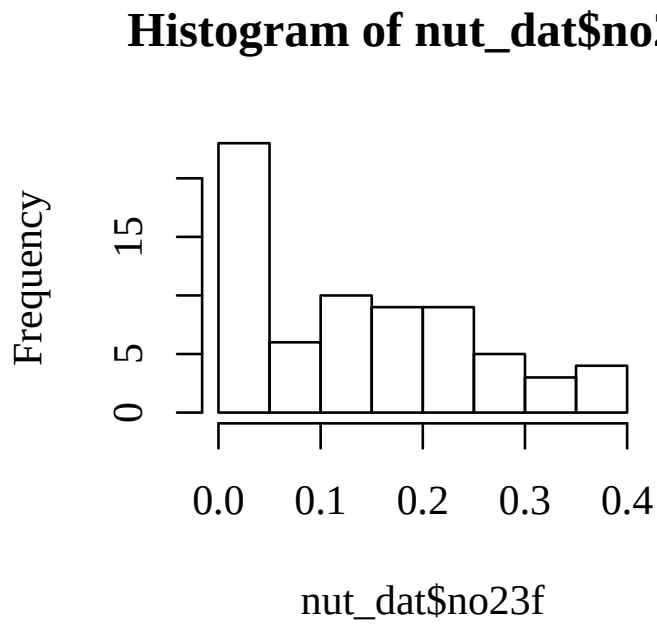


Common functions for EDA

Histograms are useful...

```
# some histograms
hist(nut_dat$po4f)
hist(nut_dat$no23f)
```

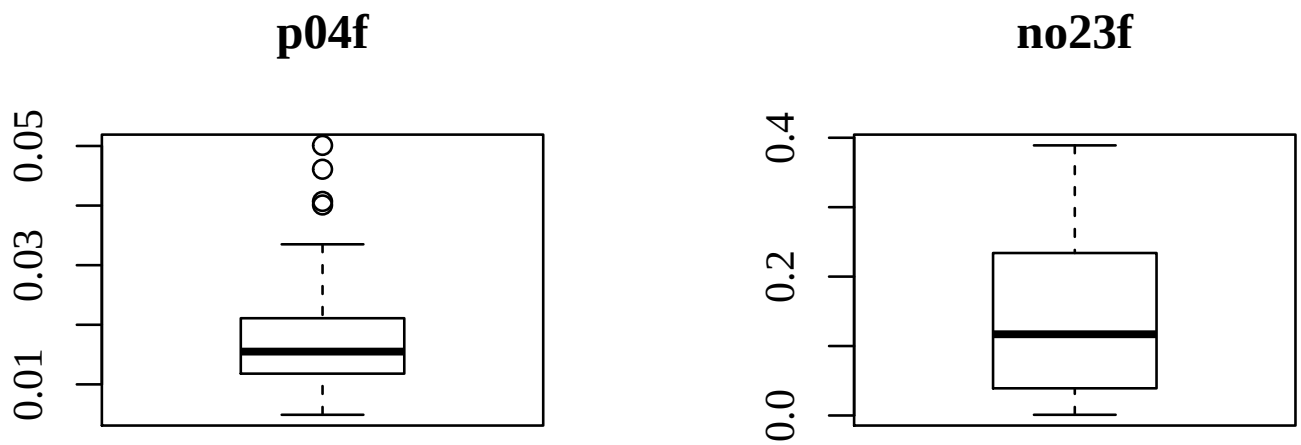




Common functions for EDA

Boxplots are useful...

```
# some boxplots
boxplot(nut_dat$po4f, main = 'p04f')
boxplot(nut_dat$no23f, main = 'no23f')
```



Common functions for EDA

Plotting individual variables or simple scatterplots between two variables will get you familiar with a dataset

Again, R has many functions for EDA and we don't want to focus on general approaches that can be learned at home

A quick google search of 'exploratory data analysis in r' will point you in the right direction

For now, we will focus on some tasks that have specific relevance to SWMP

Analysis 1 - Missing data and interpolation

Time series will usually include missing data - you will have to decide how to handle missing values

Let's import some wq data

```
# import data, qaqc, and subset
# change this path for the flash drive
path <- 'C:/data/dataset3'
wq_dat <- import_local(path, 'cbmmcwq2012')
```

```
# remove qaqc, and subset do_mgl
wq_dat <- qaqc(wq_dat)
wq_dat <- subset(wq_dat, select = 'do_mgl')

# how many missing values?
sum(is.na(wq_dat$do_mgl))

## [1] 419
```

Analysis 1 - Missing data and interpolation

Mising data can be removed with the subset function or replaced with the mean

```
# a temporary object so we don't overwrite wq_dat
wq_tmp <- wq_dat

# remove missing values with subset function
wq_tmp <- subset(wq_tmp, rem_row = T)

# or replace missing values with the mean
wq_tmp <- wq_dat
wq_tmp[is.na(wq_tmp$do_mgl), 'do_mgl'] <- mean(wq_tmp$do_mgl, na.rm = T)
```

What are some issues with these approaches?

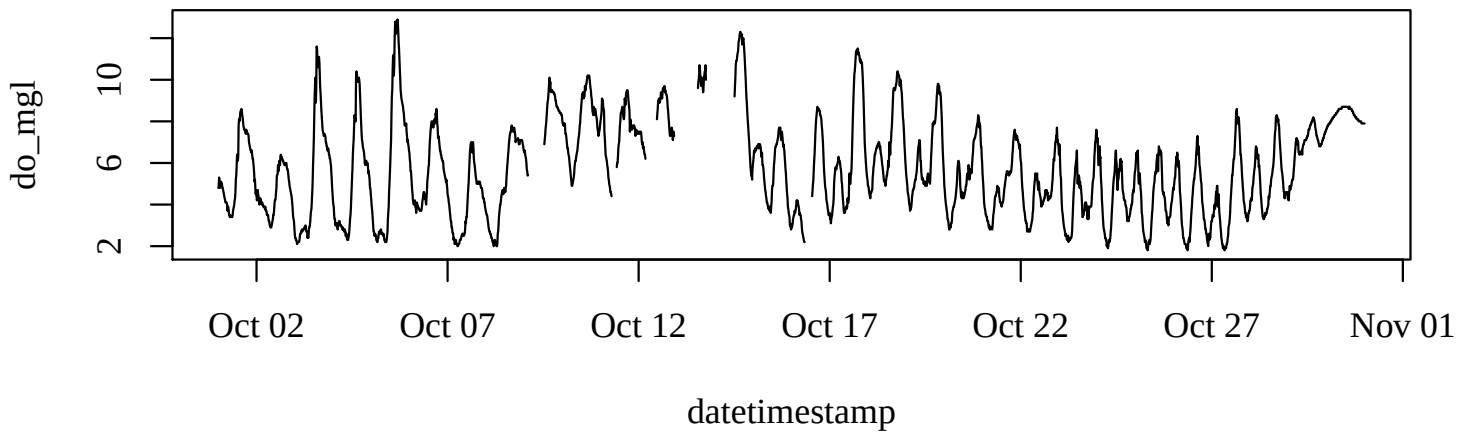
'subset' will change the time step

Neither approach is very true to the data...

Analysis 1 - Missing data and interpolation

Introducing the 'na.approx' function - this method can interpolate missing data

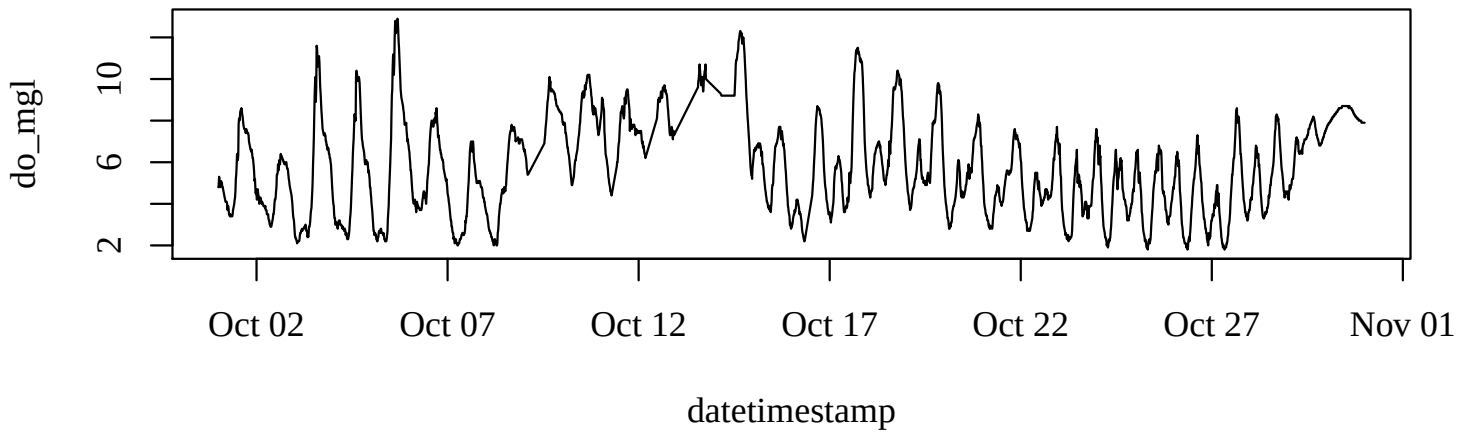
```
# subset the do time series for plotting
wq_dat <- subset(wq_dat, subset = c('2012-10-01 0:0', '2012-10-31 0:0'))
plot(do_mgl ~ datetimestamp, wq_dat, type = 'l')
```



Notice the missing values around October 12th

Analysis 1 - Missing data and interpolation

Here's what the time series looks like after using 'na.approx'



The missing values have been linearly interpolated - not a true representation but better than some other approaches

Analysis 1 - Missing data and interpolation

The ‘na.approx’ function has only a few arguments


NERRS / SWMP
Data Analysis Workshop: Time Series
November 17, 2014

Questions??