

# Summary of data processing for Patuxent River Estuary

Marcus W. Beck, Rebecca Murphy

March 23, 2015

The following describes additional processing of raw station data for the Patuxent River Estuary. Note that the raw data file ('PAX\_TRIB...') was edited manually to remove station TF1.0 and to change the chlorophyll value for TF1.4 on 11/28/1988 from 2.99 to 5.98. Briefs descriptions of each step are provided.

The following raw data files were imported:

1. `PAX_TRIB_CHLAandSALINITY_85to14.csv`: chlorophyll and salinity data for all stations in the Patuxent River from 1985 to 2014 (from R. Murphy)
2. `PAX_station_info.csv`: metadata for each station including lat/lon, salinity zone, etc. (from R. Murphy)

The data were first imported into R.

```
# code for processing raw data, see email from R. Murphy on 3/13/15
# created March 2015, M. Beck

## packages to use
# this is just to load dplyr, ggplot2
devtools::load_all('M:/docs/tidal_comp/TidalComp')

## import

# meta
pax_meta <- system.file('PAX_station_info.csv', package = 'TidalComp')
pax_meta <- read.csv(pax_meta, header = TRUE,
  stringsAsFactors = FALSE)

# data
```

```

pax_data <- system.file('PAX_TRIB_CHLAandSALINITY_85T014.csv',
  package = 'TidalComp')
pax_data <- read.csv(pax_data, header = TRUE,
  stringsAsFactors = FALSE)

# reorder STATION variable along trib axis
stats <- c('TF1.3', 'TF1.4', 'TF1.5', 'TF1.6', 'TF1.7',
  'RET1.1', 'LE1.1', 'LE1.2', 'LE1.3', 'LE1.4')
pax_data$STATION <- factor(pax_data$STATION, level = stats)

```

Salinity data were vertically-integrated for each unique date. The integration function averaged all salinity values after interpolating from the surface to the maximum depth. Salinity values at the most shallow and deepest sampling depth were repeated for zero depth and maximum depths, respectively, to bound the interpolations within the range of the data. A similar process for vertically-integrating salinity across depth values was used for the remaining station data.

```

##
# get vertically integrated salinity

# vertical integration by date, station
int_fun <- function(TOTAL_DEPTH, DEPTH, AvgValue){

  if(length(na.omit(AvgValue)) < 2 ) return(na.omit(AvgValue))

  # setup for interpolation
  max_depths <- mean(unique(TOTAL_DEPTH), na.rm = TRUE)
  depths <- c(0, DEPTH, max_depths)
  vals <- c(AvgValue[1], AvgValue, AvgValue[length(AvgValue)])

  # only interpolate if > 1 salinity value
  out <- mean(approx(depths, vals)$y)

  return(out)

}

# process
# note that there are no 'PROBLEM' values, lab and method do not change
sal_tmp <- filter(pax_data, PARAMETER == 'SALINITY') %>%
  mutate(date = as.Date(date, format = '%m/%d/%Y')) %>%
  group_by(date, STATION) %>%
  summarize(sal = int_fun(TOTAL_DEPTH, DEPTH, AvgValue))

```

Chlorophyll values at each station were retained only for surface samples and no ‘problem’ codes. Chlorophyll were also transformed by the natural-log.

```
##  
# get only surface estimates for chlorophyll  
# remove those w/ problem codes  
chl_tmp <- filter(pax_data,  
  PARAMETER == 'CHLA' & LAYER == 'S' & PROBLEM == ''  
) %>%  
mutate(lnchl = log(AvgValue)) %>%  
mutate(date = as.Date(date, format = '%m/%d/%Y')) %>%  
select(date, STATION, lnchl)
```

Some plots of the raw data.

```
##  
# merge chl and salinity data, then plot  
  
pax_data <- full_join(chl_tmp, sal_tmp, by = c('date', 'STATION'))  
  
ggplot(pax_data, aes(x = date, y = sal, group = STATION)) +  
  geom_line() +  
  theme_classic() +  
  facet_wrap(~ STATION, ncol = 3)  
  
ggplot(pax_data, aes(x = date, y = lnchl, group = STATION)) +  
  geom_line() +  
  theme_classic() +  
  facet_wrap(~ STATION, ncol = 3)  
  
# save the data  
save_path <- gsub('text$', 'data', getwd())  
save(pax_data, file = paste0(save_path, '/pax_data.RData'))
```

