Annual Report - Current Year's Values vs Historical CI

Jess S. Glanz | SMS-BEL Program Coordinator | jess.glanz@gmail.com

8 Mar 2023

Load the packages

```
library(readxl)
library(dplyr)
library(tidyverse)
library(tidyr)
library(ggplot2)
```

Load the data

```
env<-read_excel("~/Desktop/SMS-BEL/SMS-EnvlAnalysis/Data/20221102_UpToOct22Env (1).xlsx")
```

The `env` data contains all environmental data collected during field sampling since monitoring began in Feb 2005. You may receive a warning regarding non-numeric variables, but you can proceed to the next step without issue.

Tidy the data

Select only the needed columns. In this case, Site through Firmness.

```
env<-env%>%select(7:33)
```

Rename the columns for ease when calling them.

```
env<-env%>%rename(turb=`Turbidity (NTU)`,
                  stemp=`Surface temperature (°C)`,
                  sdo=`Surface oxygen`,
                  ssal=`Surface salinity (ppt)`,
                  sph=`Surface pH`,
                  btemp=`Bottom temperature (°C)`,
                  bdo=`Bottom oxygen`,
                  bsal=`Bottom salinity (ppt)`,
                  msal=`MidSalinity`,
                  bph=`Bottom pH`)
```

Add a column for each quarter when monitoring occurs based on the number in the month (M) column. .after="M" places the quarter column after the month (M) column.

```
env1<-env %>% mutate(q =
                     case_when(M == 1 ~ 1,
                               M == 2 ~ 1,
                               M == 4 ~ 2,
                               M == 5 ~ 2,
                               M == 6 ~ 3,
                               M == 7 ~ 3,
                               M == 9 ~ 4,
                               M == 10 ~ 4),
                   .after="M")
```

Separate the previous values from the current year's values.

```
env_pre<-env1%>%filter(YYYY<2022)

env_cur<-env1%>%filter(YYYY==2022)
```

Example Parameter: Create data for water temperature, bottom and surface

Bottom temp

Step 1: Calculate the mean, standard deviation, and sample size for each site and quarter based on historical values.

```
btempsq <- env_pre %>%
  group_by(Site, q) %>%
  summarize(mean=mean(btemp, na.rm=TRUE), s=sd(btemp, na.rm=TRUE), n=length(na.omit(btemp)))
```

Step 2: Calculate the margin of error per site and quarter.

```
bte<-btempsq%>%group_by(Site,q)%>%
  summarize(marg=qt(0.975,df=n-1)*s/sqrt(n))
```

Step 3: Merge the margin of error with the previously calculated mean and rename the "mean" column.

```
bte1<-cbind(bte, btempsq$mean)
bte1<-bte1%>%rename(mean=`...4`)
```

Step 4: Subtract and add the margin of error from and to the mean to obtain 95% CI's around the mean for each site and quarter.

```
bte2<-bte1%>%group_by(Site,q)%>%
  summarize(telo=mean-marg,tehi=mean+marg)
```

Surface temp

Repeat the same procedure for surface temperature.

```
stempsq <- env_pre %>%
  group_by(Site, q) %>%
  summarize(mean=mean(stemp, na.rm=TRUE), s=sd(stemp, na.rm=TRUE), n=length(na.omit(stemp)))

ste<-stempsq%>%group_by(Site,q)%>%
  summarize(marg=qt(0.975,df=n-1)*s/sqrt(n))

ste1<-cbind(ste, stempsq$mean)
ste1<-ste1%>%rename(mean=`...4`
)

ste2<-ste1%>%group_by(Site,q)%>%
  summarize(stelo=mean-marg,stehi=mean+marg)
```

Bind the bottom and surface CI's together and rename the respective columns.

```
envCI <- cbind(bte2, ste2$stelo)

envCI  <- cbind(envCI, ste2$stehi)

envCI <- envCI%>%
  rename(stelo = `...5`,
         stehi = `...6`)
```

Merge the CI data with the current year's values by Site and quarter.

```
env_CIcur<-inner_join(envCI,env_cur, by=c("Site","q"))
```

Select the columns for Site, q, temp CI's, and current temp values.

```
teCIcur<-env_CIcur%>%select(1:6,17,21)
```

Select the columns for Site, q, and the current values for both the bottom and surface.

```
te_cur<-teCIcur%>%select(1:2,7,8)
```

Select the columns for Site, q, and the low CI values for both the bottom and surface.

```
te_lo<-teCIcur%>%select(1:2,3,5)
```

Select the columns for Site, q, and the high CI values for both the bottom and surface.

```
te_hi<-teCIcur%>%select(1:2,4,6)
```

Convert each of the three previous datasets from wide to long for the plot.

```
tecurl<-te_cur %>% pivot_longer(cols=btemp:stemp, names_to="SB",values_to="temp")

telol<-te_lo %>% pivot_longer(cols=telo:stelo, names_to="sb1",values_to="lo")

tehil<-te_hi %>% pivot_longer(cols=tehi:stehi, names_to="sb1",values_to="hi")
```

Add a column so the name for the surface and bottom factor are consistent for each dataset.

```
telol<-telol %>% mutate(SB =
                          case_when(sb1 == "telo" ~ "btemp",
                                    sb1=="stelo"~"stemp"))

tehil<-tehil %>% mutate(SB =
                          case_when(sb1 == "tehi" ~ "btemp",
                                    sb1=="stehi"~"stemp"))
```

Merge the low and high CI datasets with the current year's dataset.

```
tel <- left_join(tecurl, telol, by = c("Site", "SB","q"))
tel <- left_join(tel, tehil, by = c("Site", "SB","q"))
```

Plot data for M6

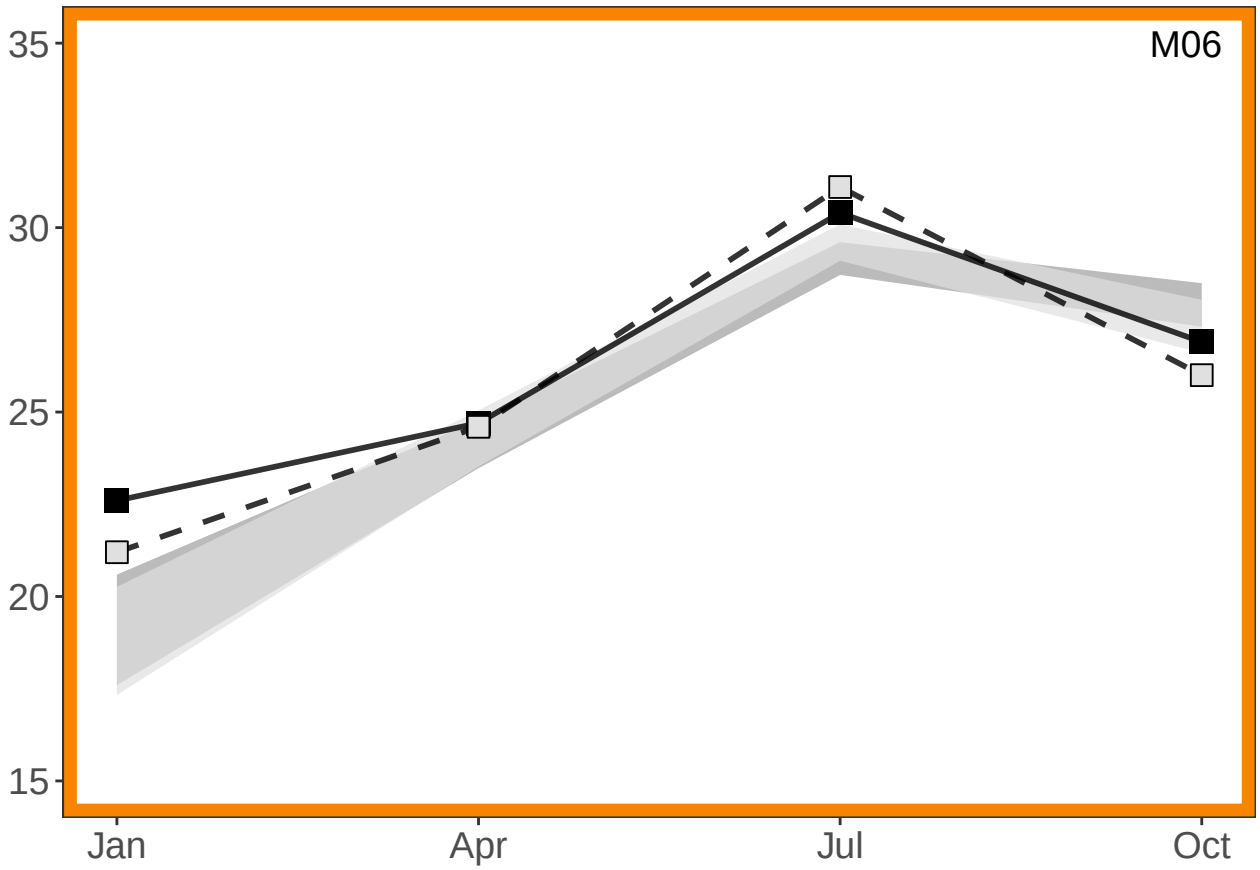
Filter the data for site M6.

```
te6<-tel%>%filter(Site=="M06")
```

Plot the data.

```
te6f<-te6 %>%
  ggplot( aes(x=q, y=temp,shape=SB,linetype=SB)) +
  geom_ribbon(aes(ymin=lo, ymax=hi,fill=SB),alpha=0.7)+
  scale_fill_manual(values=c("gray88", "gray62","white"), breaks=c("stemp", "btemp"),
                    labels=c( "Surface", "Bottom"))+
  scale_shape_manual(values=c(22, 15,16), breaks=c("stemp", "btemp"),
                     labels=c( "Surface", "Bottom"))+
  scale_linetype_manual(values=c("dashed", "solid","solid"), breaks=c("stemp", "btemp"),
                        labels=c( "Surface", "Bottom"))+
  geom_line(size=1, alpha=0.8) +
  geom_point( size=4, color="black",fill="gray88") +
  theme(legend.position = "none")+
  theme_bw() +
  theme(panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_rect(colour = "#F98400", size=5),
        plot.margin = margin(-.6, 0, 0, 0, "cm"))+
  labs(x=NULL, y=NULL)+ # New code
  ggtitle("M06") +
  scale_x_continuous(breaks=c(1,2,3,4),labels=c("Jan", "Apr","Jul","Oct"))+
  theme(legend.title=element_blank())+
  theme(axis.text=element_text(size=14),
        plot.title = element_text(hjust = 0.97, vjust=-8,size=14)) +
  ylim(15,35)+
  theme(legend.position = "none")
```

Check out the plot. The current year's bottom and surface temp values at M6 are in black and the 95% CI's are in grey ribbons for each quarter.



Save the plot to the working directory.

```
ggsave(
  "tempM6.jpg",#file name
  plot = te6f,#plot to save, defaults to last plot displayed
  device = "jpg",#file format (jpg, pdf, png, eps, etc)
  path = NULL, #defaults to working directory
  scale = 1,
  width =12,
  height = 8,
  units = "cm", #c("in", "cm", "mm", "px")
  dpi = 300,
  limitsize = TRUE)
```