

# Lab 5

Sawyer Balint

Fall 2024; Marine Semester Block 3

## Contents

1	Introduction	1
2	Task 1	2
3	Task 2	5
4	Reflection Questions	7

## 1 Introduction

This document is available at <https://github.com/sjbalint/BI521/tree/main/scripts/labs>

```
#import packages
library(tidyverse) #for data wrangling
library(here) #for filepath management
library(ggsci) #for colors
library(scales) #for log axis breaks
library(rEDM) #for stats

#custom graphing theme

update_geom_defaults("point", list(shape = 21, fill="grey", stroke=0.8))

theme <- list(
  theme_classic(),
  scale_color_viridis_d(option="inferno", end=0.8),
  scale_fill_viridis_d(option="inferno", end=0.8),
  theme(legend.position="right",
        strip.placement="outside",
        strip.background=element_blank()),
  labs(x="Year", color=NULL)
)
```

## 2 Task 1

```
#import data
load(here("Labs/Lab 6 - EDM/DBdata[asmt][v4.495].RData"))

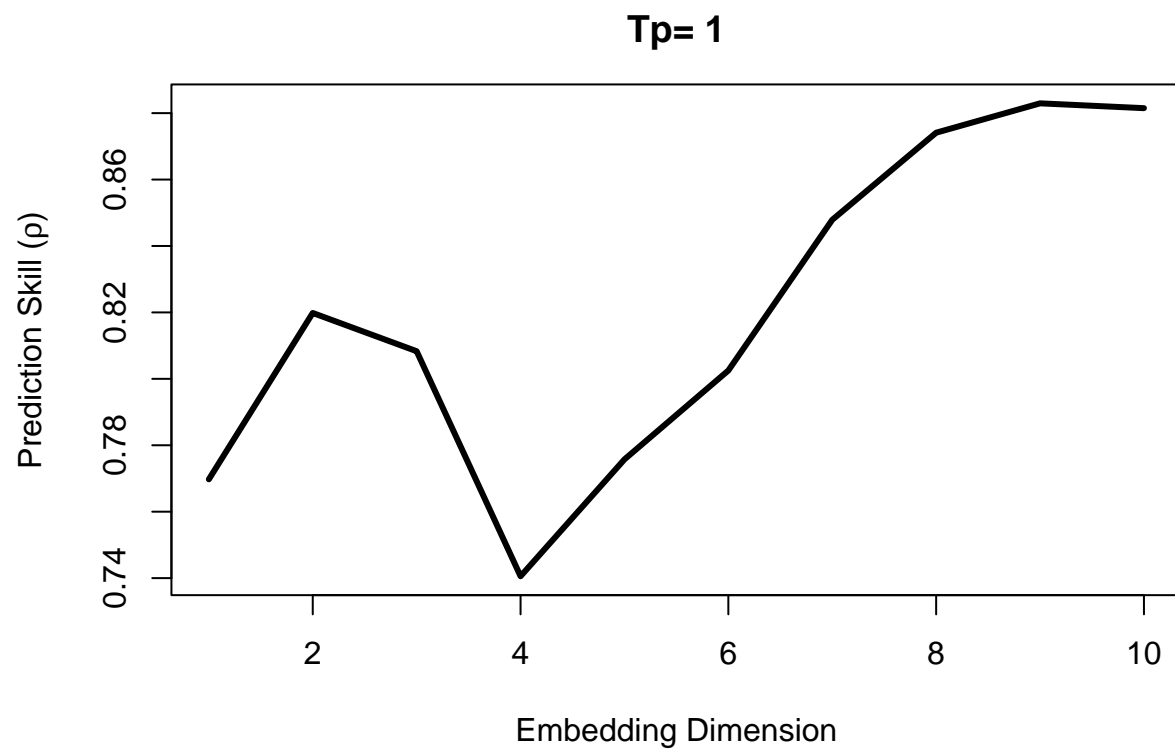
#identify stock IDs
stock_id.df <- stock %>%
  filter(region=="US East Coast") %>%
  filter(grepl("cod",commonname,ignore.case=TRUE) |
    grepl("herring",commonname,ignore.case=TRUE))

#subset timeseries for IDs of interest
lab6.df <- timeseries %>%
  filter(stockid %in% stock_id.df$stockid)

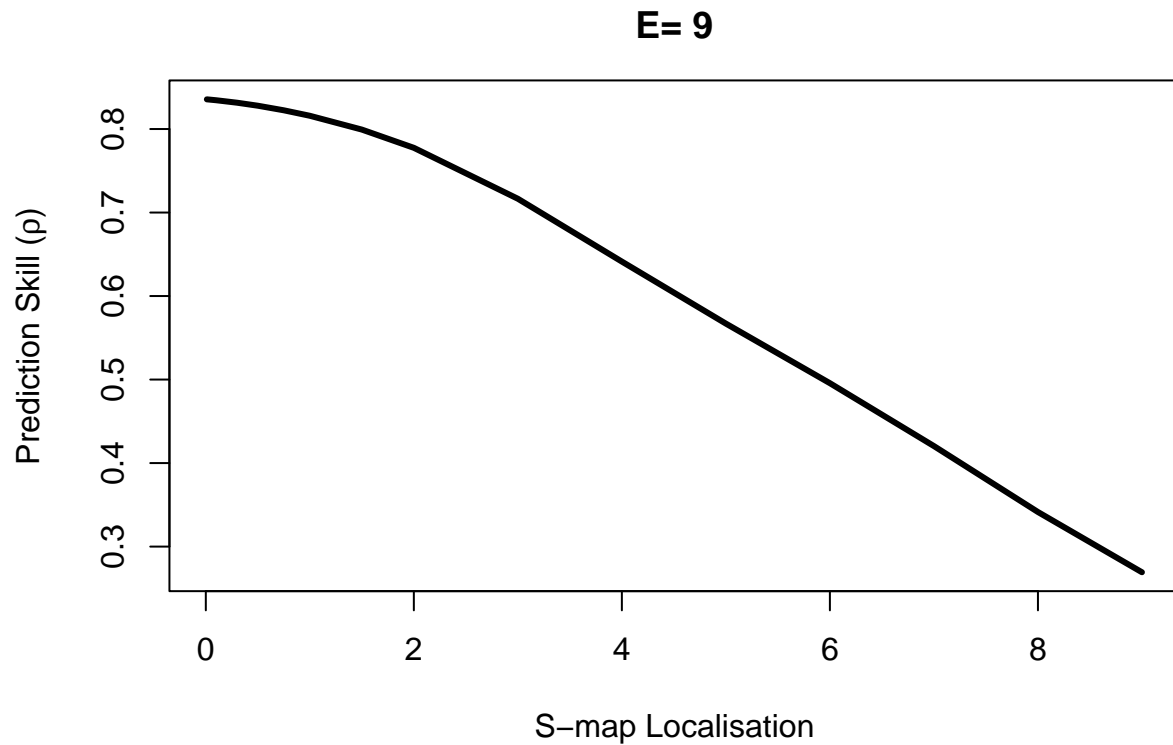
#more subsets
lab6_R.df <- lab6.df %>%
  filter(tsid=="R-EOO")
lab6_SSB.df <- lab6.df %>%
  filter(tsid=="SSB-MT")
lab6_TC.df <- lab6.df %>%
  filter(tsid=="TC-MT")

#first EDM
EDM_1.df <- lab6_SSB.df %>%
  filter(stockid=="HERRNWATLC" &
    assessid=="NEFSC-HERRNWATLC-1965-2014-SISIMP2016") %>%
  select(tsyear, tsvalue)

EDM_1_simplex.df <- EmbedDimension(dataFrame=EDM_1.df,
  lib="1 50", pred="1 50",
  columns="tsvalue", target="tsvalue")
```



```
EDM_1_smap.df <- PredictNonlinear(dataFrame=EDM_1.df,  
                                  lib="1 50", pred="1 50",  
                                  columns="tsvalue", target="tsvalue", E=9)
```



```
#use a for loop to do multiple iterations
result.list <- list()

for (i in c(1:3)){

  EDM_1.df$tssurr <- SurrogateData(EDM_1.df$tssvalue, method="ebisuzaki",1)

  plot.df <- EDM_1.df %>%
    pivot_longer(!tsyear) %>%
    mutate(name=factor(name, levels=c("tsvalue","tssurr"),
                                labels=c("Data","Surrogate")),
           iteration=as.character(i))

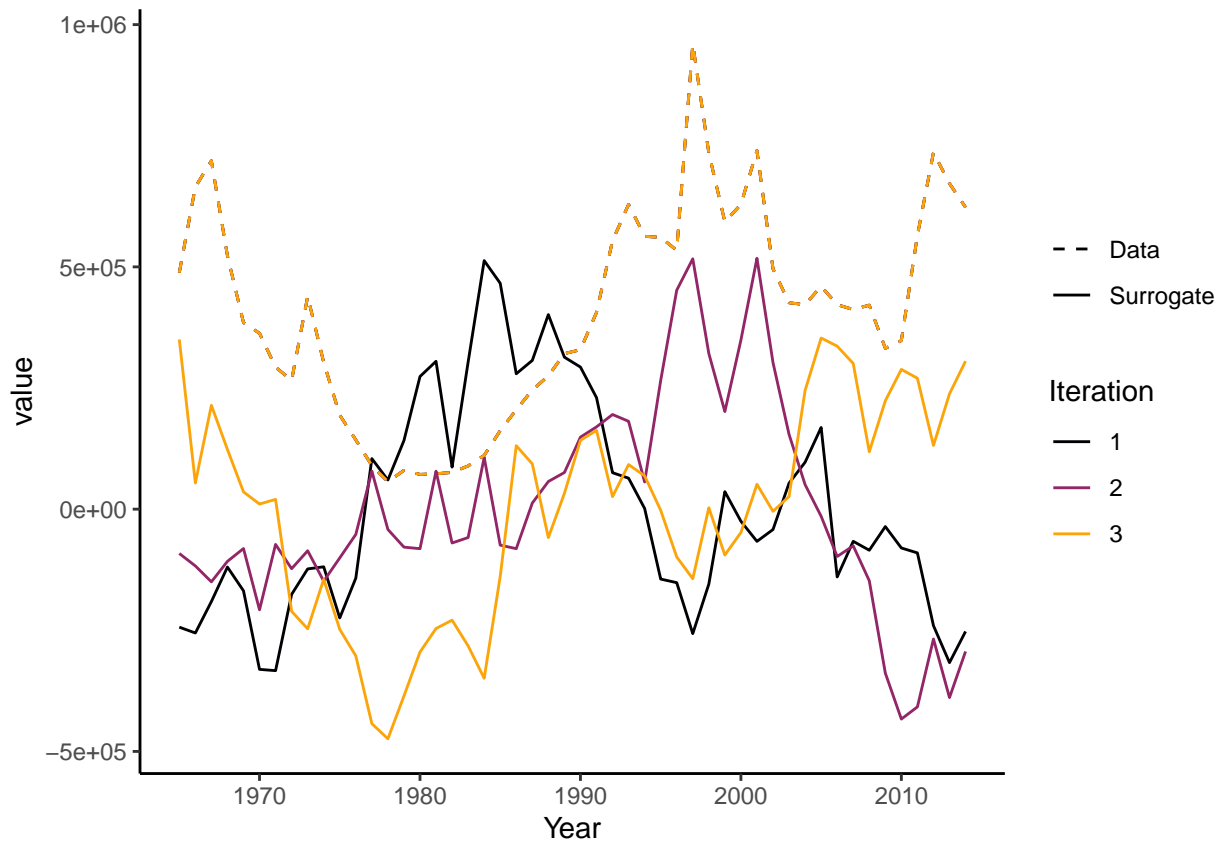
  result.list <- c(result.list, list(plot.df))
}

result.df <- bind_rows(result.list)

#make a plot
ggplot(result.df, aes(tsyear, value,
                      color=iteration,
                      linetype=name,
                      group=interaction(name, iteration)))+

  theme+
  geom_line()+
```

```
scale_linetype_manual(values=c("dashed","solid"))+
labs(linetype=NULL, color="Iteration")
```



**Fig. 1:** When we run the code multiple times without a set seed, we get different time series for the surrogate. The surrogate data is phase-randomized, and the random component of the method results in different results for every iteration. If we had instead used an AR(1) method, we would have consistent results.

### 3 Task 2

```
#calculate p value in a brute-forcy way
rho.v <- vector()

for (i in 1:500){
  EDM_1.df$tssurr <- SurrogateData(EDM_1.df$t$value, method="ebisuzaki",1)

  rho.df <- EmbedDimension(dataFrame=EDM_1.df,
                           lib="1 50",pred="1 50",
                           target="tssurr",columns="tssurr",
                           showPlot = FALSE)

  rho.v <- c(rho.v, mean(rho.df$rho))
}
```

```

}

mean_rho <- EmbedDimension(dataFrame=EDM_1.df,
                           lib="1 50",pred="1 50",
                           target="tsvalue",columns="tsvalue",
                           showPlot = FALSE) %>%

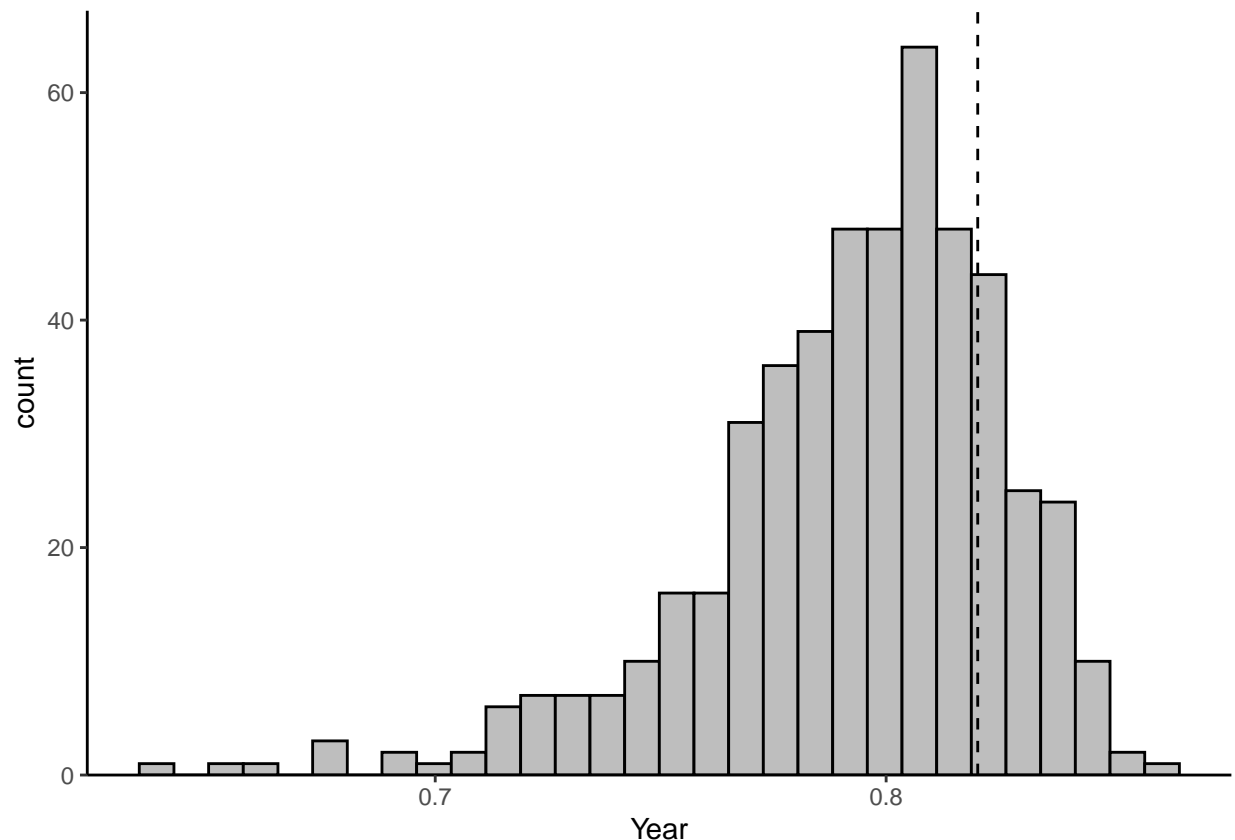
  pull(rho) %>%
  mean()

rho.df <- data.frame(rho=rho.v) %>%
  mutate(signif=ifelse(rho<mean_rho, TRUE, FALSE))

ggplot(rho.df, aes(rho))+
  theme+
  geom_histogram(color="black", fill="grey")+
  geom_vline(xintercept = mean_rho, linetype="dashed")+
  scale_y_continuous(expand=expansion(mult=c(0,0.05)))

```

## 'stat\_bin()' using 'bins = 30'. Pick better value with 'binwidth'.



```

summary.df <- rho.df %>%
  group_by(signif) %>%
  count() %>%
  ungroup() %>%

```

```
  pivot_wider(names_from="signif", values_from="n")

pval <- summary.df$`FALSE`/summary.df$`TRUE`
signif(pval, digits=3) #not significant!
```

```
## [1] 0.247
```

## 4 Reflection Questions