Spatiotemporal Stats

Spatiotemporal Stats Lab

Savita Upadhyay

```
knitr::opts_chunk$set(echo = TRUE,cache = TRUE)
library(STRbook)
library(ggplot2)
library(dplyr)
library(tidyr)
library(sp)
library(spacetime)
library(lubridate)
```

Questions

For the SST data loaded using the code below:

1. generate a data frame with the Empirical Spatial Means per decade (1970-1979, 1980-1989, 1990-2002) and plot them with one panel per decade
2. generate a spatial plot for the **yearly** SST 95th quantile for the years 1980, 1990, 2000 having one panel per year
3. Obtain a Hovmoller plot for these data
4. Calculate the EOFs for the *SST* dataset. How many EOFs would you retain?

```
data("SSTlandmask", package = "STRbook")
data("SSTlonlat", package = "STRbook")
data("SSTdata", package = "STRbook")


#combining the data frame
#SSTdata_lonlat <- cbind(SSTdata,
#                        SSTlonlat)

#remove years that are not complete
rm_rows <- which(SSTlandmask == 1)
SSTdata <- SSTdata[-rm_rows, 1:396]
SSTlonlat <- SSTlonlat[-rm_rows, 1:2]


SSTdata_n <- cbind(SSTlonlat, SSTdata)
SST_df <- gather(SSTdata_n, date, sst, -lon, -lat)

date_grid <- expand.grid(Month = c("Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug","Sep", "Oct", "Nov", "Dec"),
                         Year = 1970:2002,
stringsAsFactors = FALSE)

date_grid$date <- paste0("V", 1:396)
date_grid$t <- seq(1,396,1)

SST_df <- left_join(SST_df, date_grid)%>%
  mutate(decade=cut(Year, breaks=c(1969,1979,1989,1999,2009),
                    dig.lab=4,right = F,include.lowest = F))
```

```
## Joining, by = "date"
```

```
SST_df$date<-NULL
colnames(SST_df)<-c("lon","lat","values","Month","Year","decade","t")
```

1. generate a data frame with the Empirical Spatial Means per decade and plot them with one panel per decade

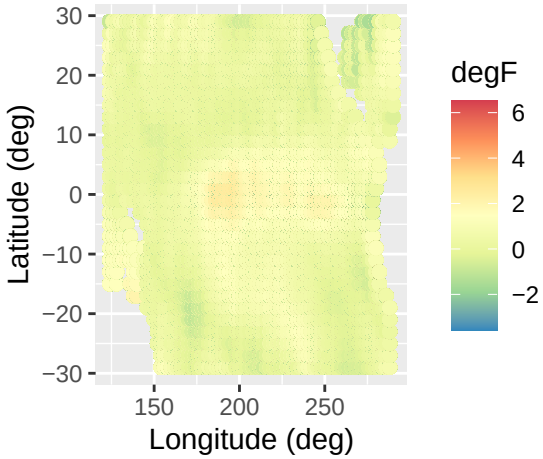
```
summ1 <- SST_df %>%
  group_by(lon,lat,Year) %>%
  summarise(proc_means=mean(values))
summ1
```

```
## # A tibble: 74,613 x 4
## # Groups:   lon, lat [2,261]
##       lon   lat  Year proc_means
##     <dbl> <dbl> <int>      <dbl>
##  1   124   -15  1970     0.445
##  2   124   -15  1971     0.145
##  3   124   -15  1972    -0.314
##  4   124   -15  1973     0.499
##  5   124   -15  1974    -0.384
##  6   124   -15  1975    -0.0677
##  7   124   -15  1976    -1.24
##  8   124   -15  1977    -0.639
##  9   124   -15  1978     0.103
## 10   124   -15  1979    -0.118
## # ... with 74,603 more rows
```

```
summ2 <- SST_df %>%
  group_by(lon,lat,decade) %>%
  summarise(proc_means=mean(values))
summ2
```

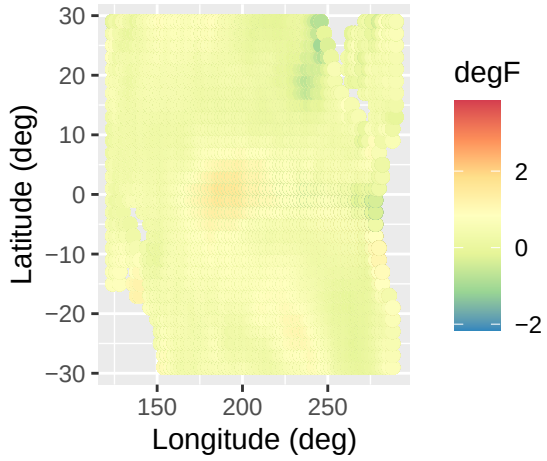
```
## # A tibble: 895,356 x 4
## # Groups:   lon, lat [2,261]
##       lon   lat decade proc_means
##     <dbl> <dbl>  <dbl>      <dbl>
##  1   124   -15      1     0.721
##  2   124   -15      2    -0.0286
##  3   124   -15      3     1.42
##  4   124   -15      4     1.06
##  5   124   -15      5     0.478
##  6   124   -15      6     0.682
##  7   124   -15      7     0.552
##  8   124   -15      8    -0.0973
##  9   124   -15      9     0.0779
## 10   124   -15     10    -0.0615
## # ... with 895,346 more rows
```

```
SST_df %>%
  group_by(lon,lat,decade) %>%
  summarise(z=mean(values)) %>%
  ggplot() +
  geom_point(aes(x=lon,y=lat,colour=z),size=2) +
  col_scale(name = "degF") +
  xlab("Longitude (deg)") +
  ylab("Latitude (deg)")
```




```
SST_df %>%
  group_by(lon,lat,Year) %>%
  summarise(z=mean(values)) %>%
  ggplot() +
  geom_point(aes(x=lon,y=lat,colour=z),size=2) +
  col_scale(name = "degF") +
  xlab("Longitude (deg)") +
```

```
  ylab("Latitude (deg)")
```

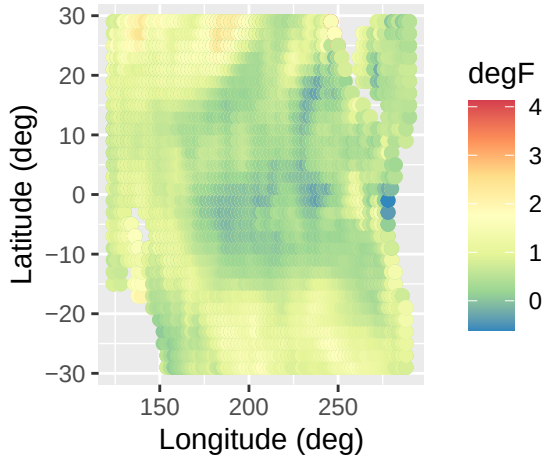


2. generate a spatial plot for the **yearly** SST 95th quantile for the years 1980, 1990, 2000 having one panel per year

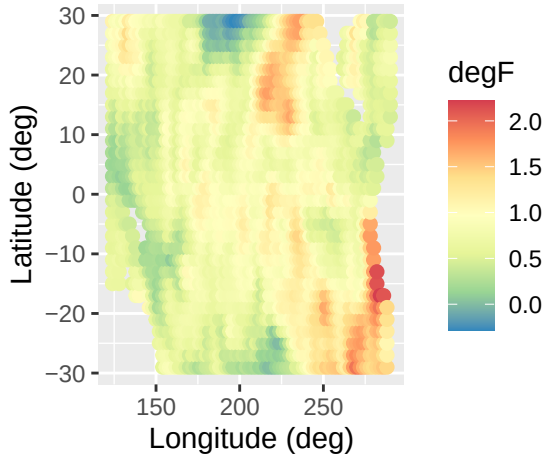
```
summ_quantile <- SST_df %>%
   filter(Year %in% c(1980,1990,2000))%>%
   group_by(lon,lat,Year) %>%
  summarise(proc_quantile=quantile(values,0.95))
summ_quantile
```

```
## # A tibble: 6,783 x 4
## # Groups:   lon, lat [2,261]
##      lon   lat  Year proc_quantile
##    <dbl> <dbl> <int>         <dbl>
##  1   124   -15  1980         0.646
##  2   124   -15  1990         0.606
##  3   124   -15  2000         0.839
##  4   124   -13  1980         0.673
##  5   124   -13  1990         0.720
##  6   124   -13  2000         0.697
##  7   124   -11  1980         0.676
##  8   124   -11  1990         0.796
##  9   124   -11  2000         0.560
## 10   124    -9  1980         0.588
## # ... with 6,773 more rows
```

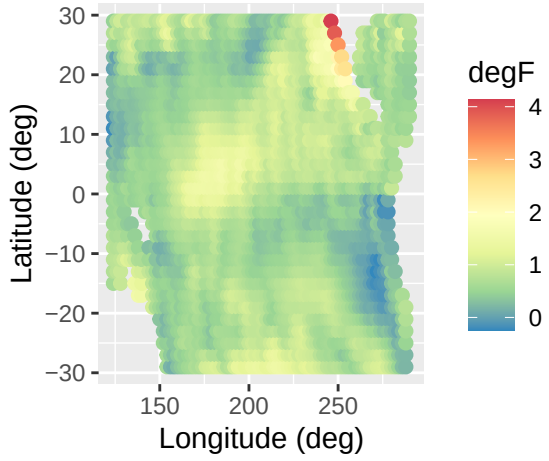
```
SST_df_1 <- subset(SST_df, Year %in% c(1980,1990,2000))

SST_df_1 %>%
  group_by(lon,lat,Year) %>%
  summarise(z = quantile(values,0.95)) %>%
  ggplot() +
  geom_point(aes(x=lon,y=lat,colour=z),size=2) +
  col_scale(name = "degF") +
  xlab("Longitude (deg)") +
  ylab("Latitude (deg)")
```



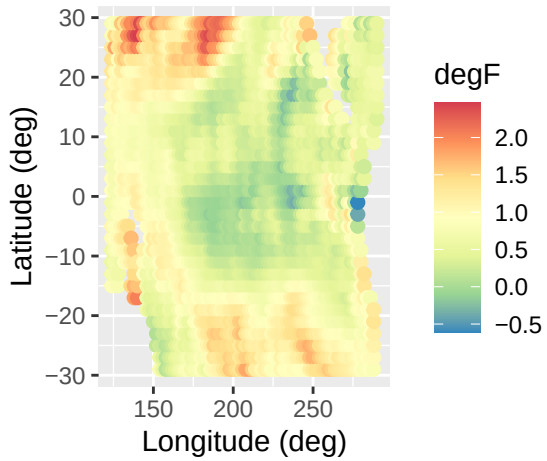
```
seq_along = c(1980,1990,2000)

plot_quantile_yearly <- function(i,SST_df_1) {
    # for each longitudinal strip
SST_df_1 %>%
   filter(Year ==seq_along[i])%>%
  group_by(lon,lat) %>%
  summarise(z = quantile(values,0.95)) %>%
  ggplot() +
  geom_point(aes(x=lon,y=lat,colour=z),size=2) +
  col_scale(name = "degF") +
  xlab("Longitude (deg)") +
  ylab("Latitude (deg)")
}
par(mfrow=c(1,3))
plot_quantile_yearly(1,SST_df_1)
```




```
plot_quantile_yearly(2,SST_df_1)
```




```
plot_quantile_yearly(3,SST_df_1)
```

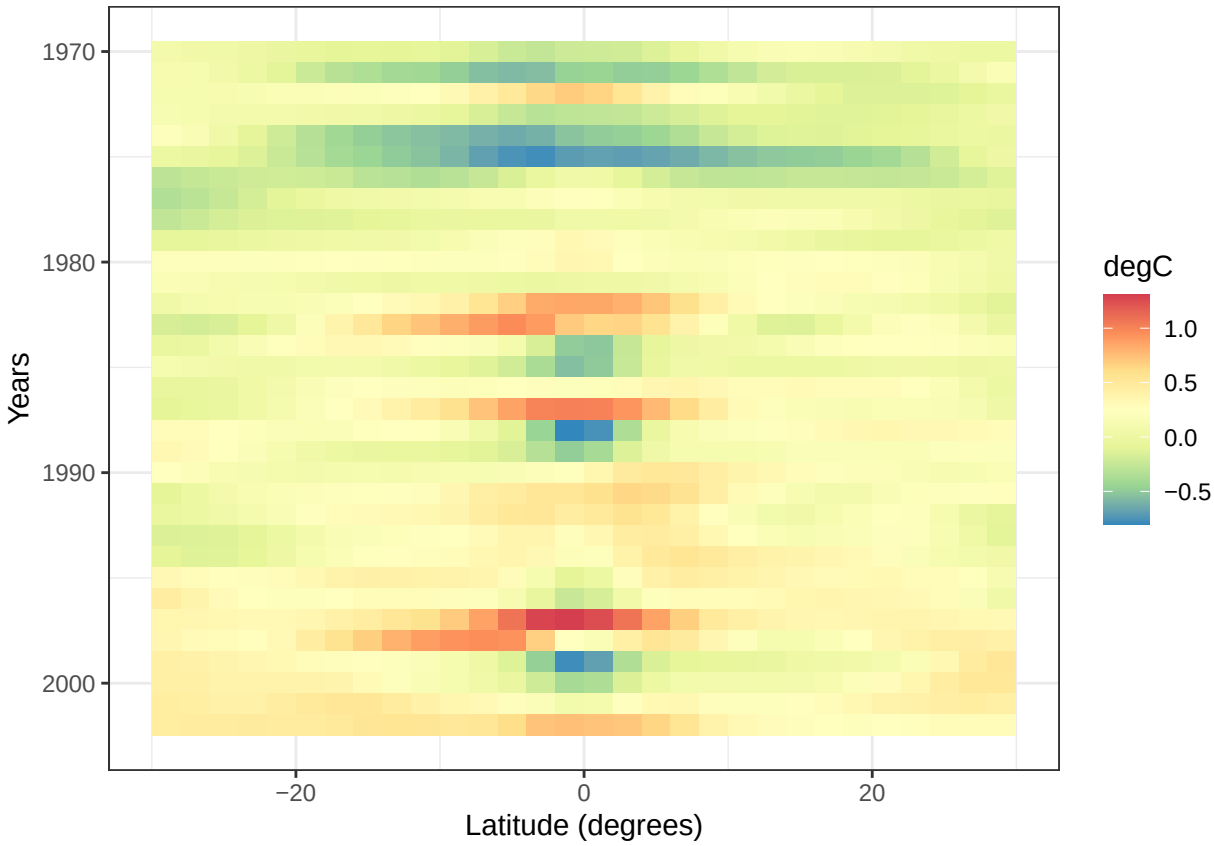


3. Obtain a Hovmoller plot for these data

```
lim_lat <- range(as.numeric(SST_df$lat))
lim_t <- range(as.numeric(SST_df$Year))
lim_t1 <- range(as.numeric(SST_df$decade))
lat_axis <- seq(lim_lat[1],
lim_lat[2],
length=25)
t_axis <- seq(lim_t[1], lim_t[2],
length=100)
lat_t_grid <- expand.grid(lat = lat_axis,
Year= t_axis)

SST_df_grid <- SST_df
dists <- abs(outer(SST_df$lat, lat_axis, "-"))
SST_df$lat <- lat_axis[apply(dists, 1, which.min)]

SST_df_lat_Hov <- group_by(SST_df_grid, lat, Year) %>% summarise(z = mean(values))
```

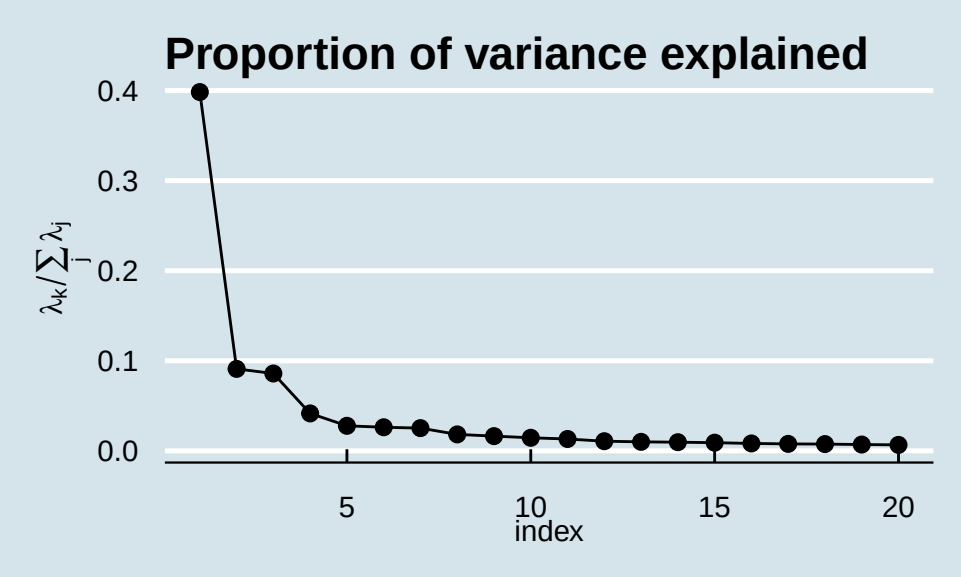
```
## `summarise()` has grouped output by 'lat'. You can override using the `.groups` argument.
```

```
Hovmoller_lat <- ggplot(SST_df_lat_Hov) +
geom_tile(aes(x = lat, y = Year, fill = z)) +
fill_scale(name = "degC") + scale_y_reverse() + ylab("Years") + xlab("Latitude (degrees)") + theme_bw()

Hovmoller_lat
```

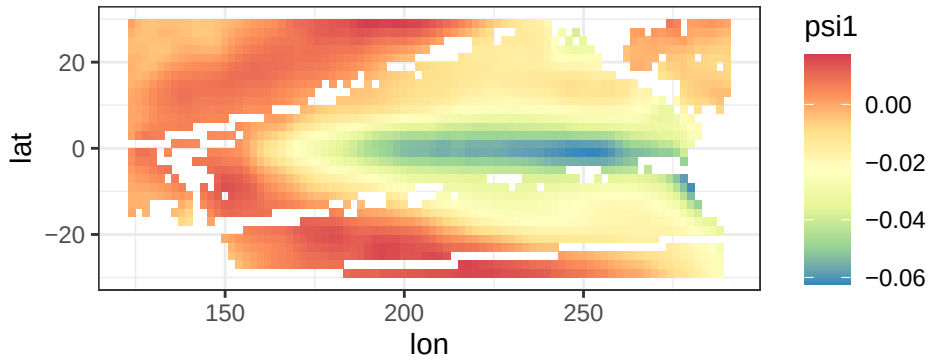


4. Calculate the EOFs for the *SST* dataset. How many EOFs would you retain?

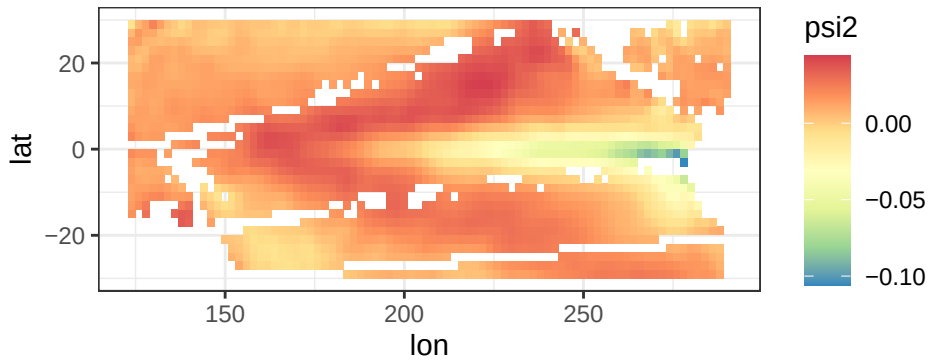
```
## Put data into space-wide form
nyr=(2002-1970)
colnames(SSTdata) <- paste(1, rep(1:12,nyr),
                           rep(1970:2002,each=12), sep="-")

# append coordinates
SSTdata <- cbind(SSTlonlat[-rm_rows,],SSTdata[-rm_rows,])
Z <- t(SSTdata[,-(1:2)])
Cz <- cov(Z)
Spec <- eigen(Cz)

# eigenvalues
lambda <- Spec$values
# eigenvectors
Psi <- Spec$vectors #EOF's

# variance explained
propvar <- lambda/sum(lambda)
tibble(index=1:length(lambda),cumprop=propvar) %>%
  filter(index<21) %>%
  ggplot() +
  geom_line(aes(x=index,y=cumprop)) +
  geom_point(aes(x=index,y=cumprop),size=2.5) +
  ggthemes::theme_economist() +
  ylab(expression(lambda[k]/sum(lambda[j],j,))) +
  labs(title="Proportion of variance explained")
```



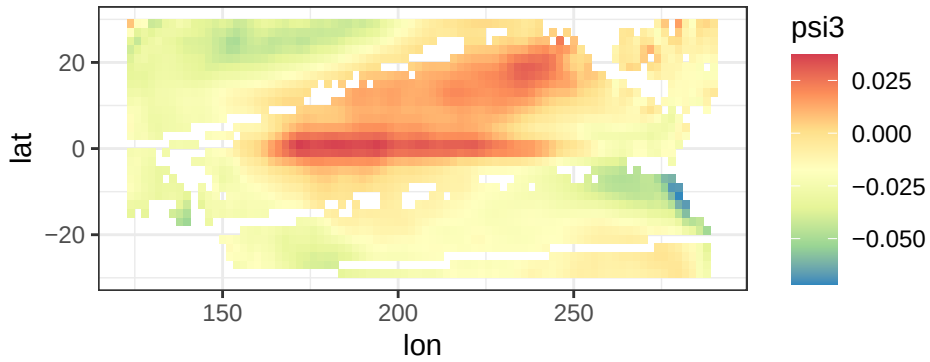

```
# Plot first 4 EOF's

EOF4 <- Psi[,1:4]
colnames(EOF4) <- paste0("psi",1:4)
EOF4 <- bind_cols(SSTdata[,1:2],as_tibble(EOF4[,1:4])) %>%
  pivot_longer(cols=starts_with("Psi"),
               names_to="EOF",
               values_to="value")

ggplot(subset(EOF4,EOF=="psi1"),aes(x=lon,y=lat))+
  geom_tile(aes(fill=-value)) +
  fill_scale(name = "psi1") +
  theme_bw()
```



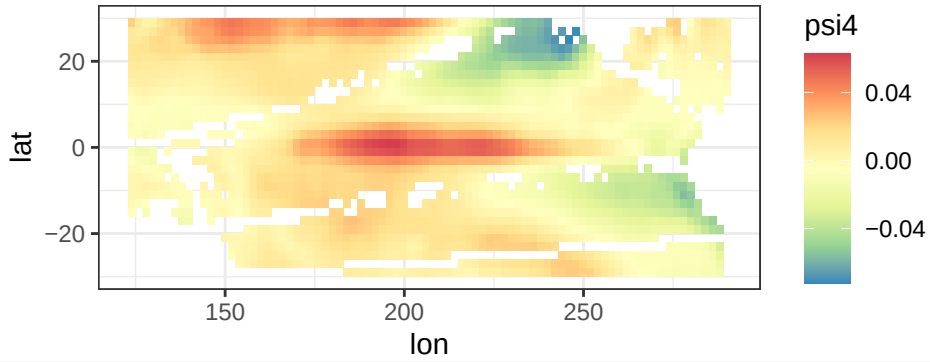

```
# Plot first 4 EOF's
ggplot(subset(EOF4,EOF=="psi2"),aes(x=lon,y=lat))+
  geom_tile(aes(fill=-value)) +
  fill_scale(name = "psi2") +
  theme_bw()
```



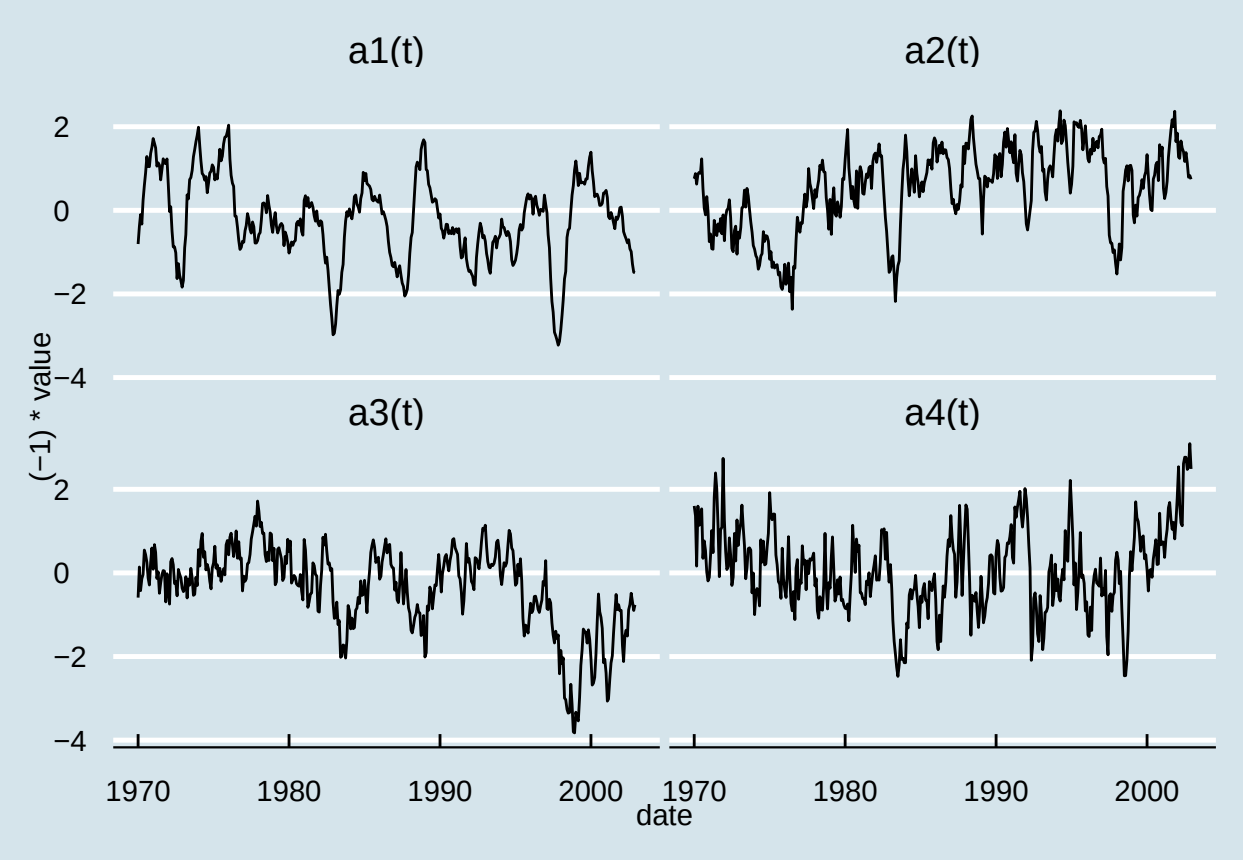
```
# Plot first 4 EOF's
ggplot(subset(EOF4,EOF=="psi3"),aes(x=lon,y=lat))+
  geom_tile(aes(fill=-value)) +
  fill_scale(name = "psi3") +
  theme_bw()
```




```
# Plot first 4 EOF's
ggplot(subset(EOF4,EOF=="psi4"),aes(x=lon,y=lat))+
  geom_tile(aes(fill=-value)) +
  fill_scale(name = "psi4") +
  theme_bw()
```




```
# PC time series (one per column)
A <- (Z%*%Psi)%*%diag(1/sqrt(round(lambda,8)))
colnames(A) <- paste0("a",1:ncol(A),"(t)")
A4 <- as_tibble(A[,1:4]) %>%
  mutate(date=dmy(rownames(A))) %>%
  pivot_longer(cols=starts_with("a"),
               names_to="pcts",
               values_to="value")

ggplot(A4) +
  geom_line(aes(x=date,y=(-1)*value)) +
  facet_wrap(~pcts) +
  ggthemes::theme_economist()
```


a1(t)
a2(t)
a3(t)
a4(t)
2
0
−2
−4
(−1) * value
1970
1980
1990
2000
date