# Reproducibility: Directional Spectra-based Clustering for Visualizing Patterns of Ocean Waves and Winds

*Carolina Euan*

*13/1/2019*

In this document we reproduce the plots obtained during the data analysis. First we load the source functions and libraries. (Non meaningful code is omited on pdf)

```
library(HMClust)
here()
```

```
## [1] "/Users/euancacd/Dropbox/Postdoc/Research/Waves/PaPer/JCGS/FinalSub/RCode"
```

```
source("Source.R")
```

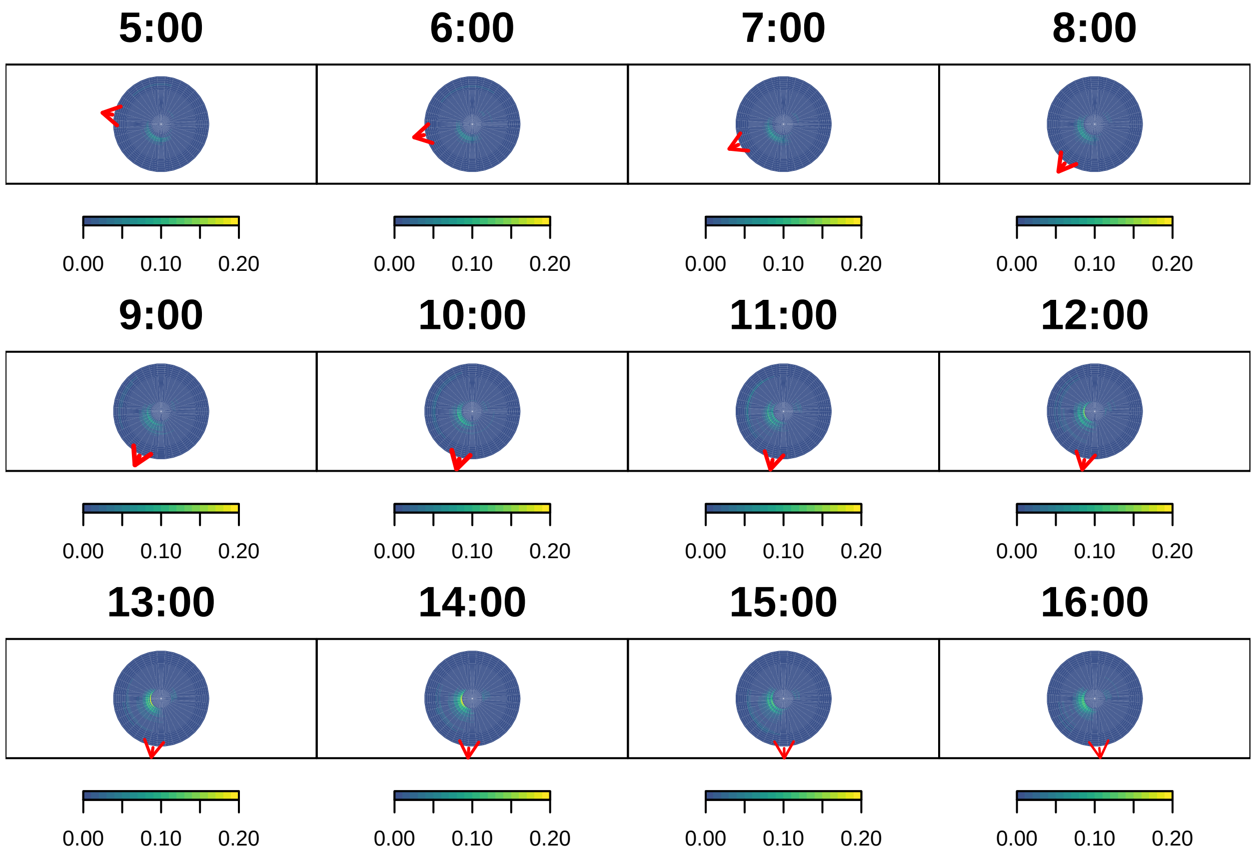
### Data Analysis

Now we load the Red Sea data set and plot the subsample of the data.

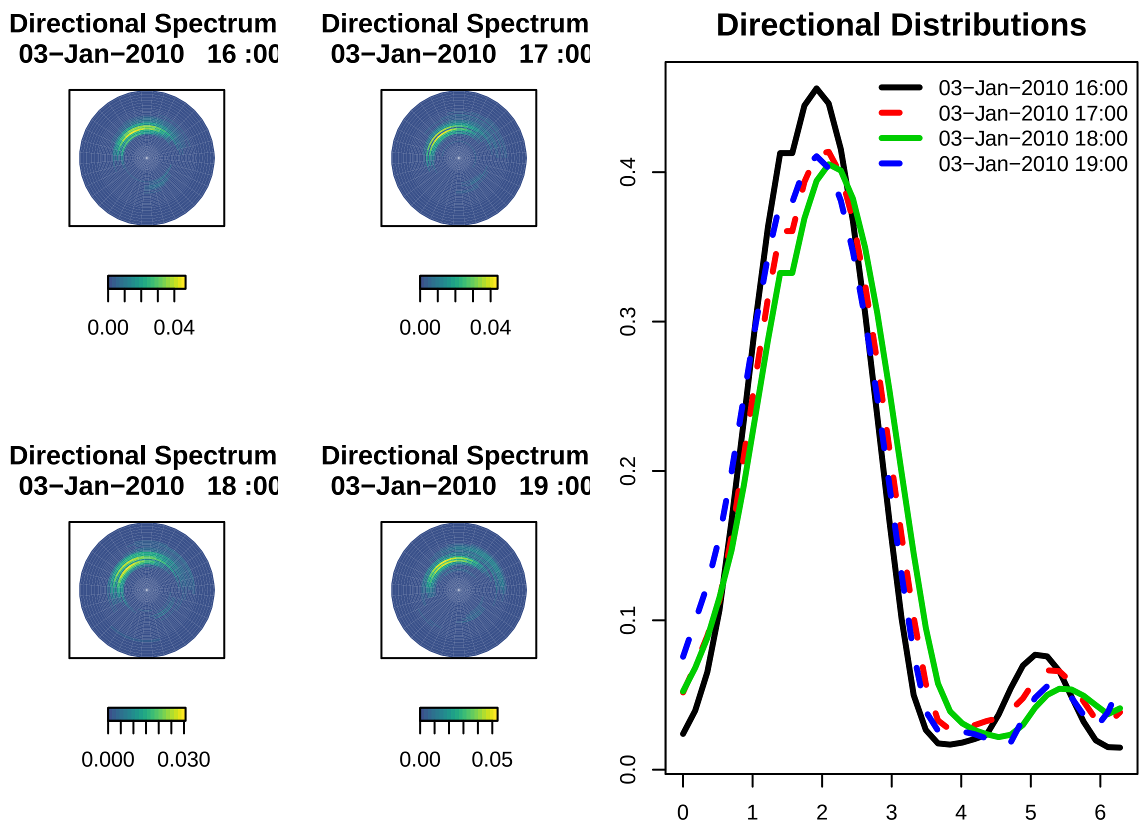
```
load("ClustRedSea.RData")
```

## Data Sub-sample

```
DirSpec<-BUOY.Wave.Data
par(mfrow=c(3,4),mar=c(2,0,3,0))
for(i in 1505:1516){
  image2D(x=Xmat,y=Ymat,z= DirSpec[[i]],colkey = list(side = 1, length = 0.5) ,theta = 0,xlim=c(-.6,.6)
          axes=FALSE,main=paste0(Hours[i],":00"),xlab=" ", ylab=" ",cex.main=2,zlim=c(0,.2),asp=1)
  DW<-as.numeric(BUOY.Wind.Data[i,2])
  arrows(x0=.5*cos(DW/360*2*pi),y0=.5*sin(DW/360*2*pi),
         x1=.6*cos(DW/360*2*pi),y1=.6*sin(DW/360*2*pi),
         length = .1,col="red",lwd = as.numeric(BUOY.Wind.Data[i,1])/2)
 }
```



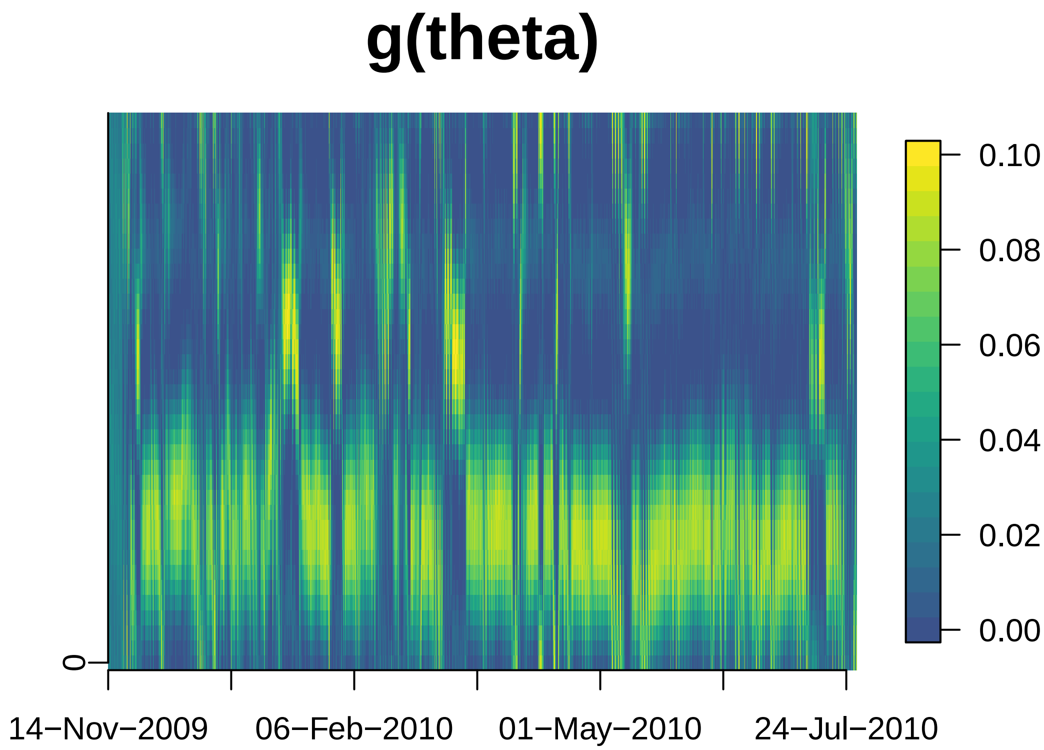

```r
#Figure 3
layout(matrix(c(1,2,5,5,3,4,5,5),ncol=4,byrow = TRUE))
for(i in 1204:1207){
  image2D(x=Xmat,y=Ymat,z= DirSpec[[i]],colkey = list(side = 1, length = 0.5) ,theta = 0, col=viridis_pa
          axes=FALSE,main=paste("Directional Spectrum \n",Date[i],
                                " ",Hours[i],":00"),xlab=" ", ylab=" ")}

par(mar=c(3,3,3,3))
matplot(theta,ftNExample,type="l",lwd=3,ylab="Energy",xlab="Theta",
        main="Directional Distributions",lty=c(1,2,1,2),cex.main=1.5)
legend("topright",paste0(Date[1204:1207],
      " ",Hours[1204:1207],":00"),lty=c(1,2,1,2),bty="n",col=1:4,lwd = 3)
```




```
c(TVD(w = theta,f1 =ftNExample[,1],f2 = ftNExample[,2]),
  TVD(w = theta,f1 =ftNExample[,2],f2 = ftNExample[,3]),
  TVD(w = theta,f1 =ftNExample[,3],f2 = ftNExample[,4]))
```

```
## [1] 0.07589596 0.05261417 0.09051586
```

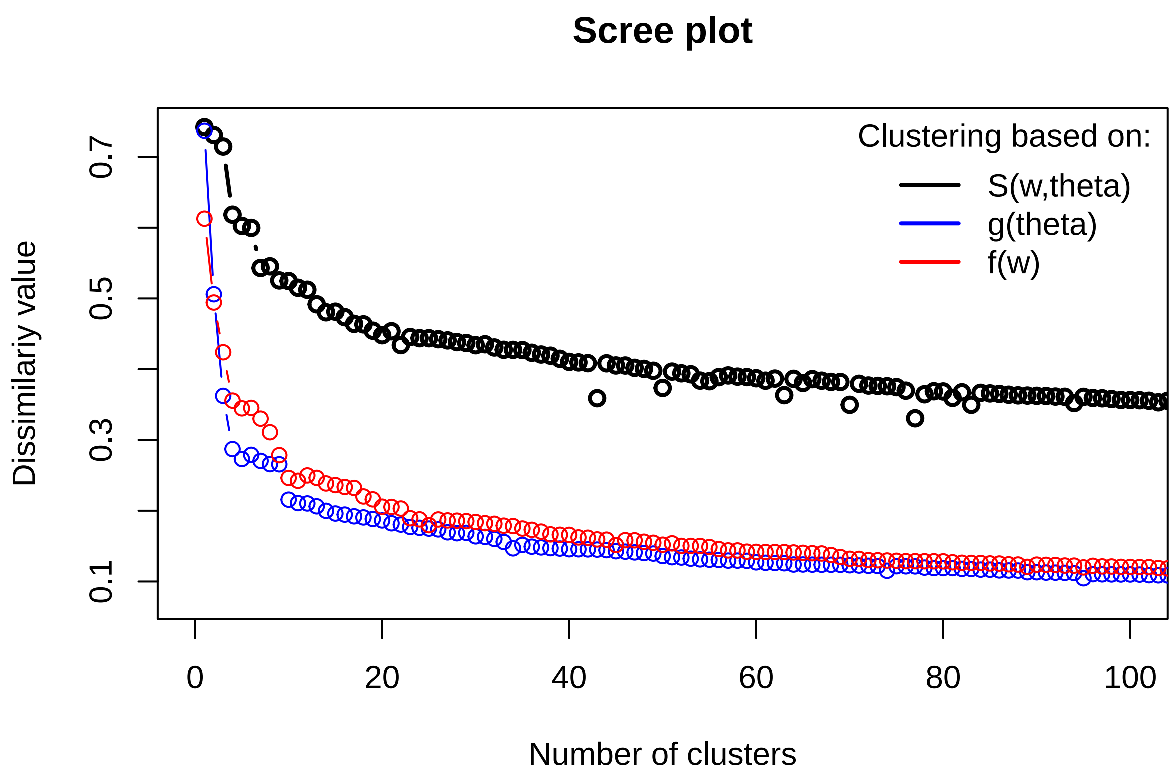
```
#Figure 6a
#BUOY
ft<-matrix(unlist(lapply(DirSpec[-dates.exclude],colMeans)),ncol=length(theta),byrow = TRUE)
ntheta<-length(theta)
#Figure 7
par(mar=c(4,7,4,6),mfrow=c(1,1))
image(y=theta,x=1:nrow(ft),z=ft/matrix(rep(rowSums(ft),ntheta),ncol=ntheta,byrow=F),
      axes=F, ylab="",main="g(theta)",xlab=" ",col=viridis_pal(begin = .25)(20),cex.main=2)
mtext(c("NE","NW","SW","SE"),side=2,at=c(45,135,225,315),las=1,line=1.5,cex=1.5)
axis(2,at=c(0,90,180,270,350),labels = c(0,90,180,270,350),cex.lab=1, padj= 1)
axis(1,at = seq(1,nrow(ft),1000),labels = (Date[-dates.exclude])[seq(1,nrow(ft),1000)],las=1,
     cex.lab=1.5)
image.plot(x=theta,y=1:nrow(ft),z=t(ft)/matrix(rep(rowSums(ft),ntheta),nrow=ntheta,byrow=T),
           col=viridis_pal(begin = .25)(20),legend.only = T)
```




## Clustering plots

Finally, we plot the clustering plots of 2D and 1D HDS clustering. First the scree plot.

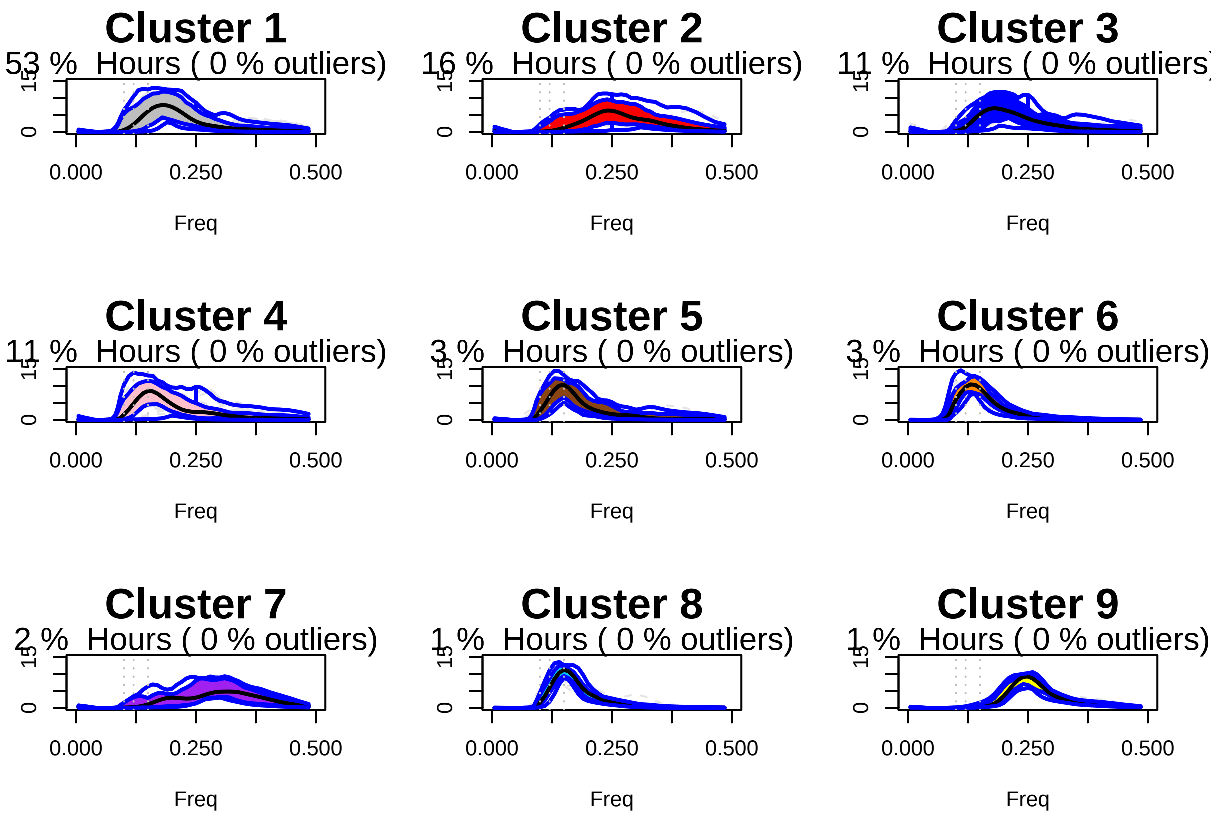
```
#Figure 6b
plot(rev(Clust2DBuoy$min.value),type="b",main="Scree plot",lwd=2,
     xlab="Number of clusters",xlim=c(0,100),ylab="Dissimilariy value")
lines(rev(ClustFreqBuoy$min.value),type="b",col="blue")
lines(rev(ClustDirBuoy$min.value),type="b",col="red")
legend("topright",c("S(w,theta)","g(theta)","f(w)"),col=c(1,4,2),
       bty = "n",lwd = 2,title = "Clustering based on: ")
```



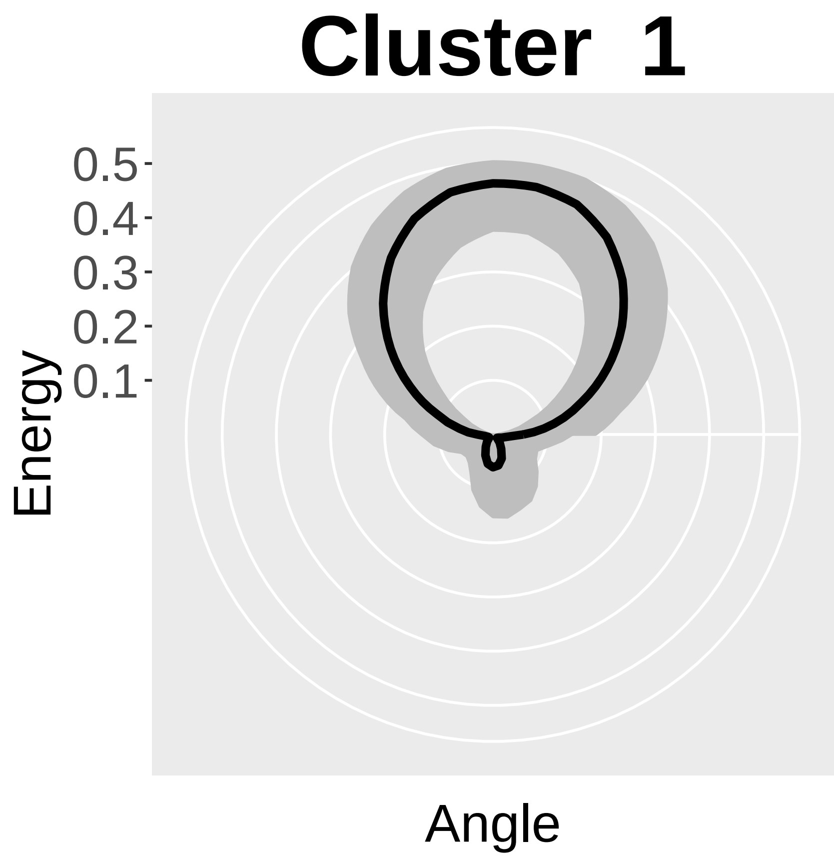

Then, the clustering visualization for the HDS clustering with the 2D-spectra.

```
################
#Compute calendar plot
################
DatesNew<-Date[-dates.exclude]
HoursNew<-Hours[-dates.exclude]
length.w<-length(w)
MatClust<-matrix(NA,nrow=24,ncol=length(dates))
colnames(MatClust)<-dates
rownames(MatClust)<-0:23
for(i in 1:nclust){
  AuxG<-((1:length(Date))[-dates.exclude])[G2[[i]]]+13
  for(j in 1:length(G2[[i]]))MatClust[AuxG[j]]<-i
}
```

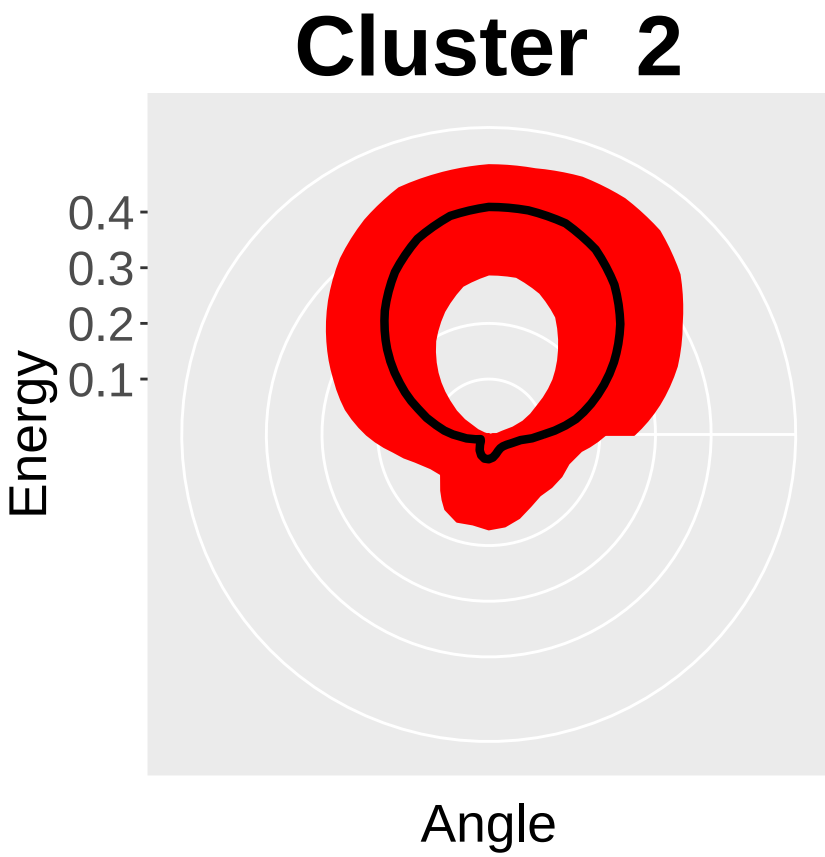
```
#fbplots of frequency spectra
par(mfrow=c(3,3))
for(indk in 1:nclust) {
  A<-fbplot(x=w,fit=fw.by.clust[[indk]],col=colors10[indk],xlim = c(0,.5),
          ylim=c(0,15),xaxp=c(0,.5,4),xlab="Freq",ylab="",outliercol="gray90",
          main=paste("Cluster",indk),cex.main=2)
  mtext(paste(round(length(G2[[indk]])/nrow(fw),2)*100,"%  Hours (",
            round(length(A$outpoint)/nrow(fw),2)*100,"% outliers)"))
  abline(v=c(.1,.12,.15),col="gray",lty=3)
}
```



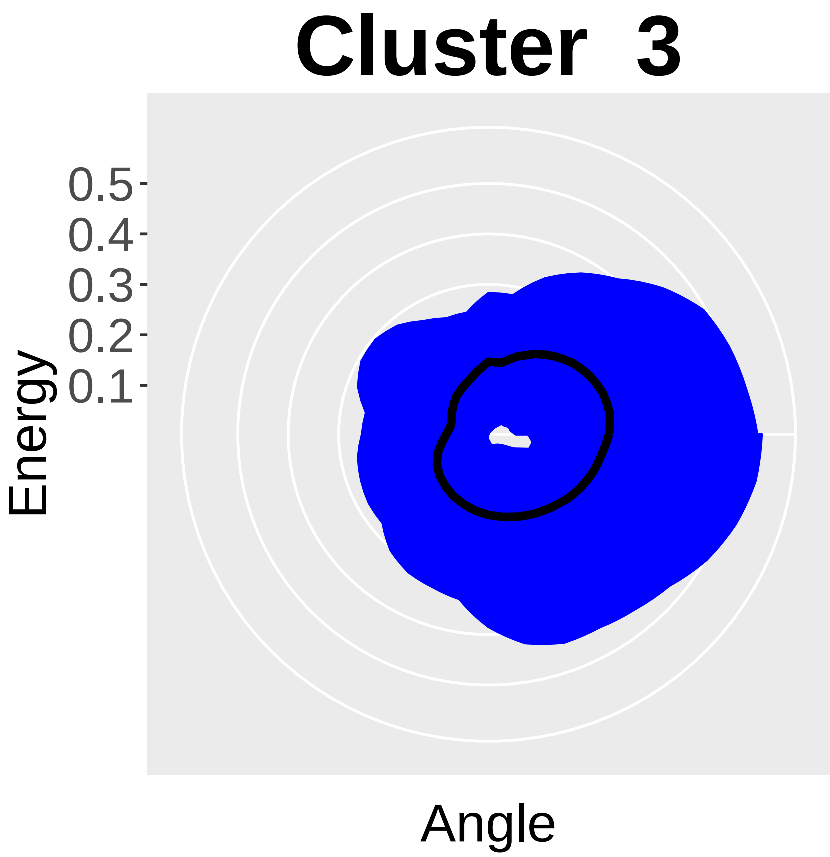

```r
#directiona fbplots
print(plotenergy[[1]])
```



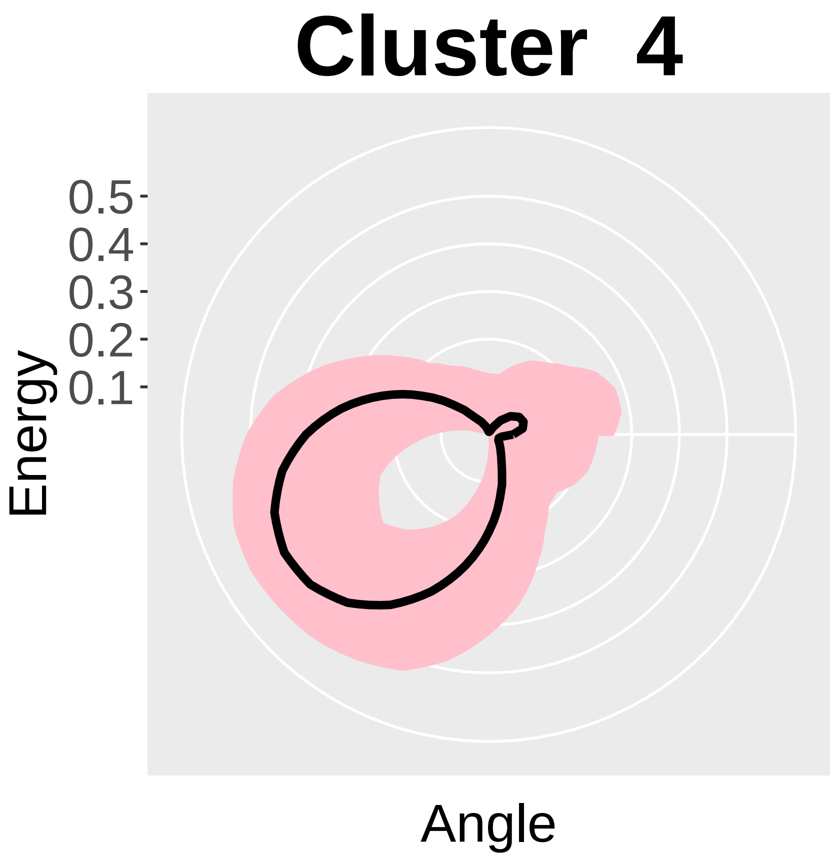

```
print(plotenergy[[2]])
```



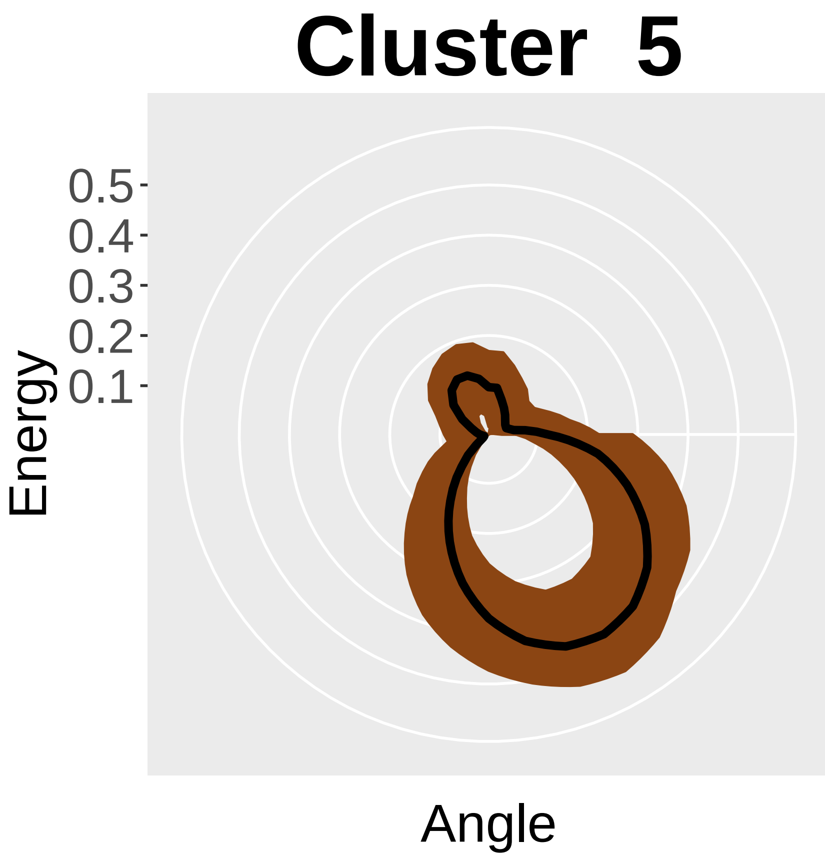

```
print(plotenergy[[3]])
```



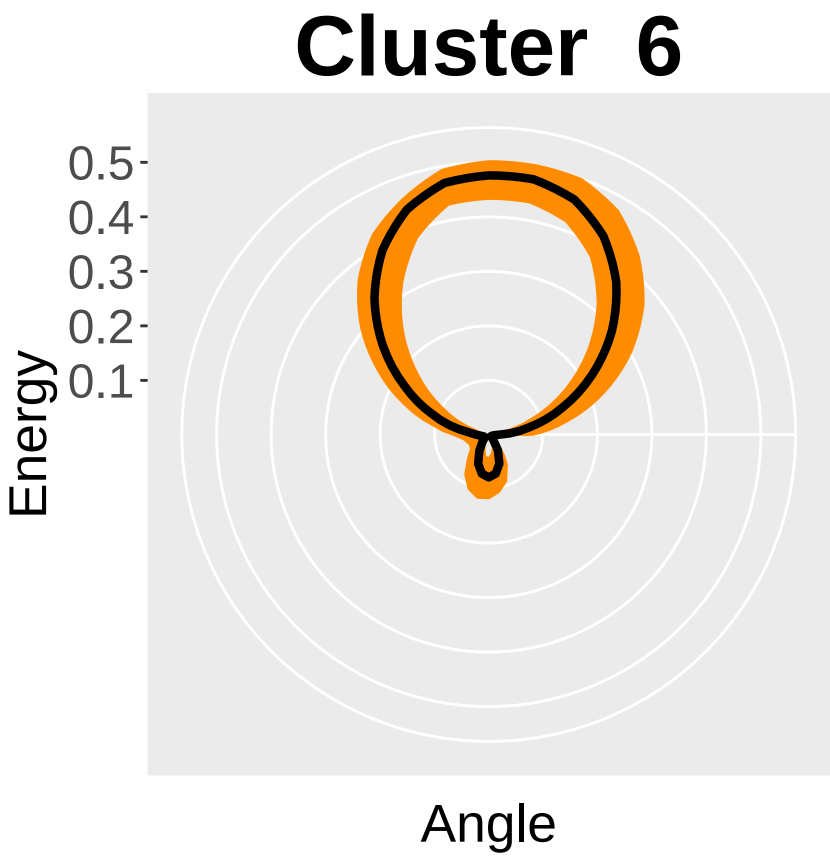

```
print(plotenergy[[4]])
```



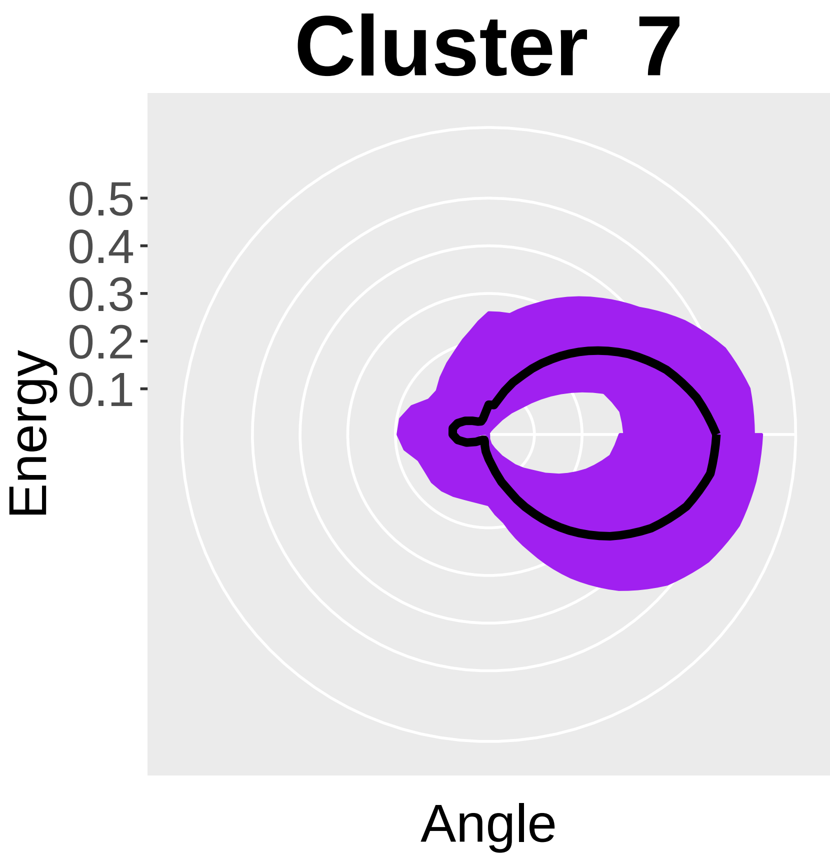

```
print(plotenergy[[5]])
```



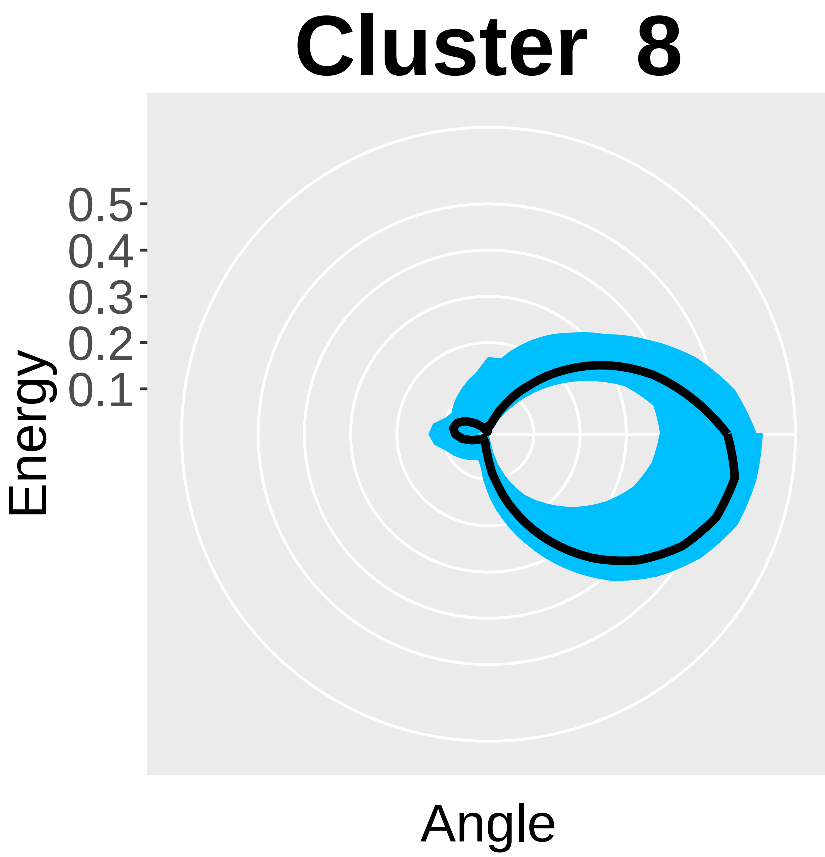

```
print(plotenergy[[6]])
```



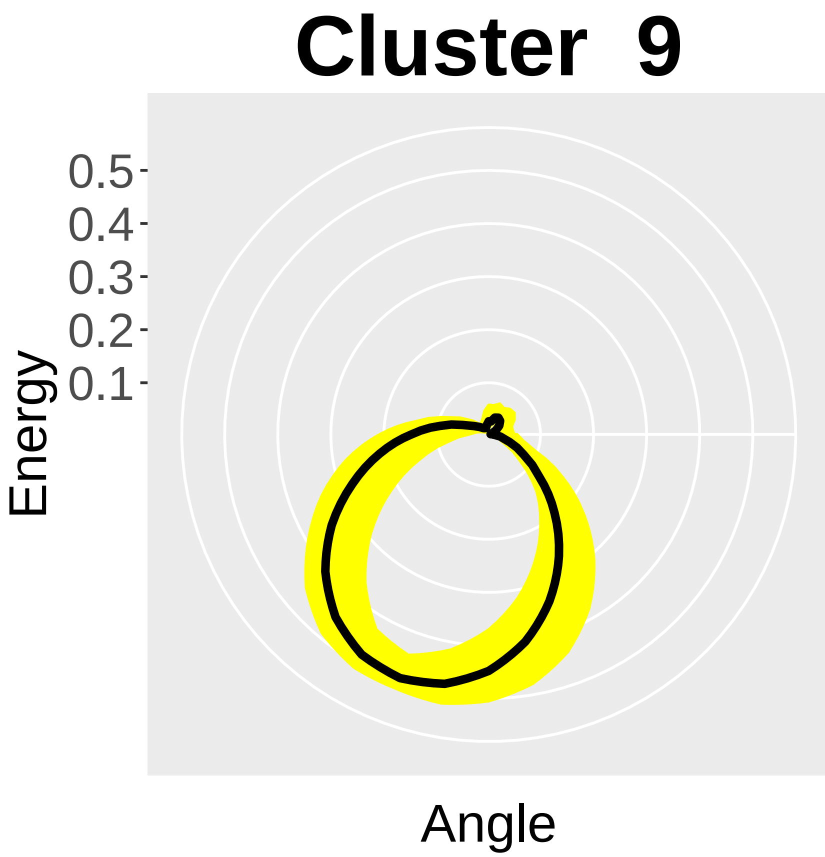

```r
print(plotenergy[[7]])
```



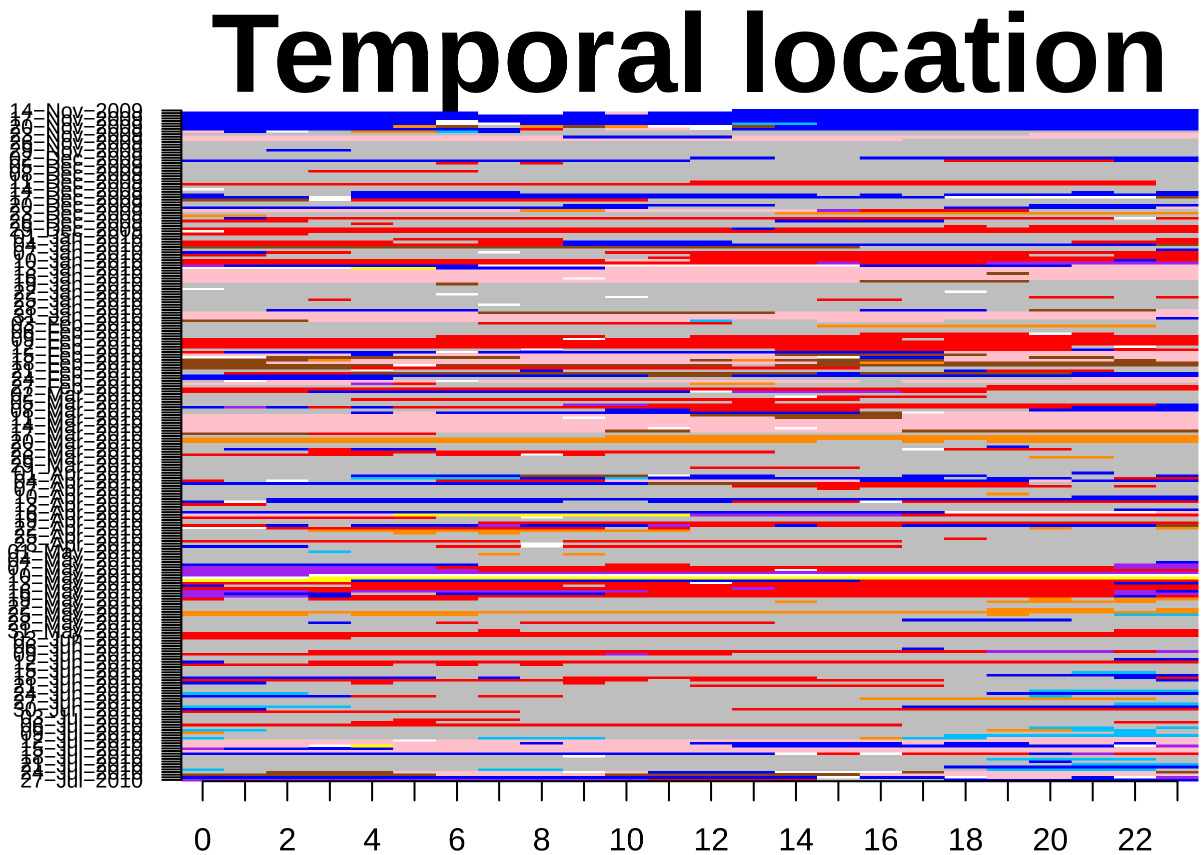

```
print(plotenergy[[8]])
```




```
print(plotenergy[[9]])
```

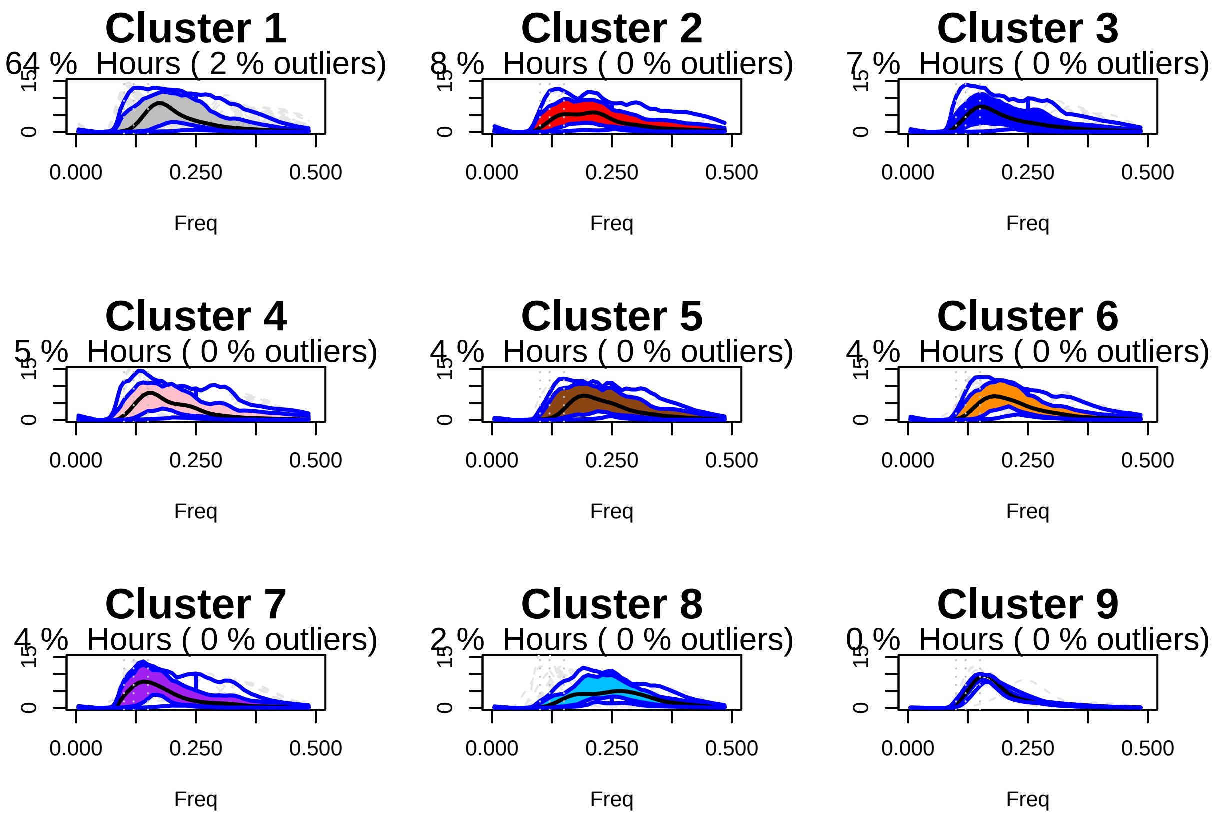



```r
#Calendar plot
par(mfrow=c(1,1),mar=c(2,5,3,1))
image(MatClust[,rev(1:256)],col=colors10[-10],axes=FALSE,main="Temporal location",
      cex.main=3.5)
axis(1,at=seq(0,1,length.out = 24),labels = 0:23)
auxlab<-rep(NA,256)
auxlab[seq(1,256,3)]<-rev(dates[seq(1,256,3)])
axis(2,at=seq(0,1,length.out = 256),labels = auxlab
     ,las=2, cex.axis=.7)
```

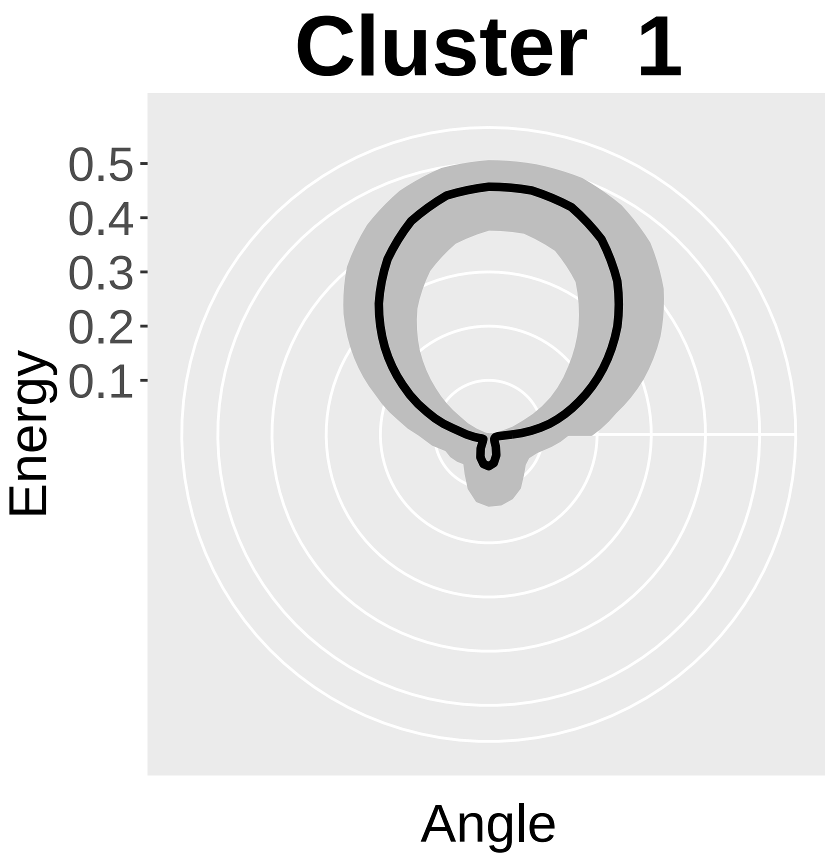



Then, the clustering visualization for the HDS direction only clustering with buoy data.

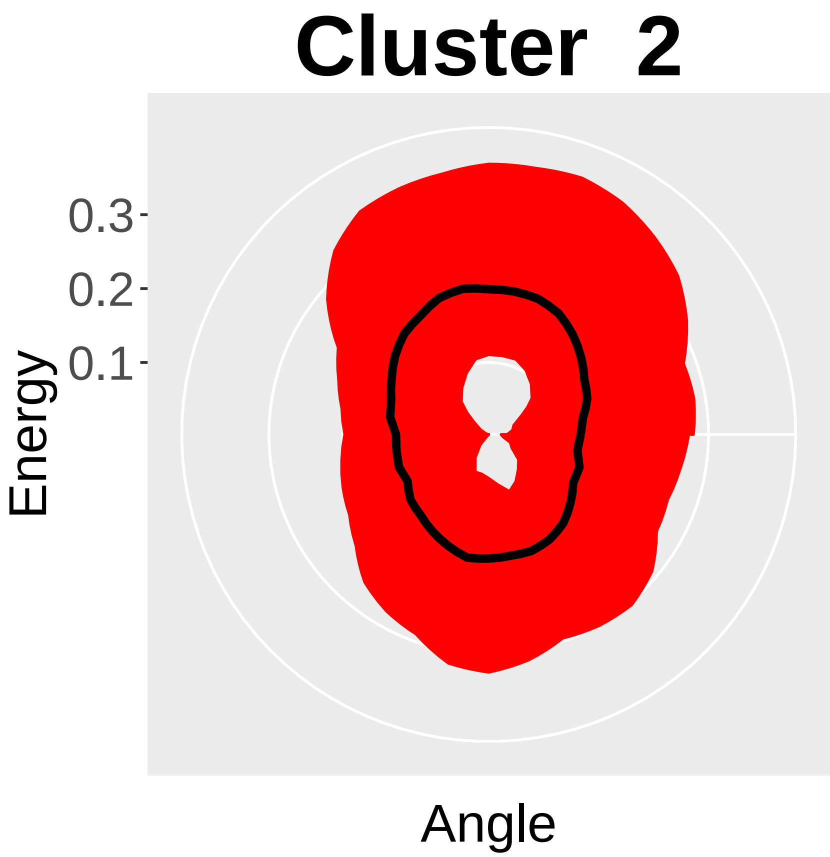
```
#fbplots of frequency spectra
par(mfrow=c(3,3))
for(indk in 1:nclust) {
  A<-fbplot(x=w,fit=fw.by.clust[[indk]],col=colors10[indk],xlim = c(0,.5),
            ylim=c(0,15),xaxp=c(0,.5,4),xlab="Freq",ylab="",outliercol="gray90",
            main=paste("Cluster",indk),cex.main=2)
  mtext(paste(round(length(G1[[indk]])/nrow(fw),2)*100,"%  Hours (",
              round(length(A$outpoint)/nrow(fw),2)*100,"% outliers)"))
  abline(v=c(.1,.12,.15),col="gray",lty=3)
}
```



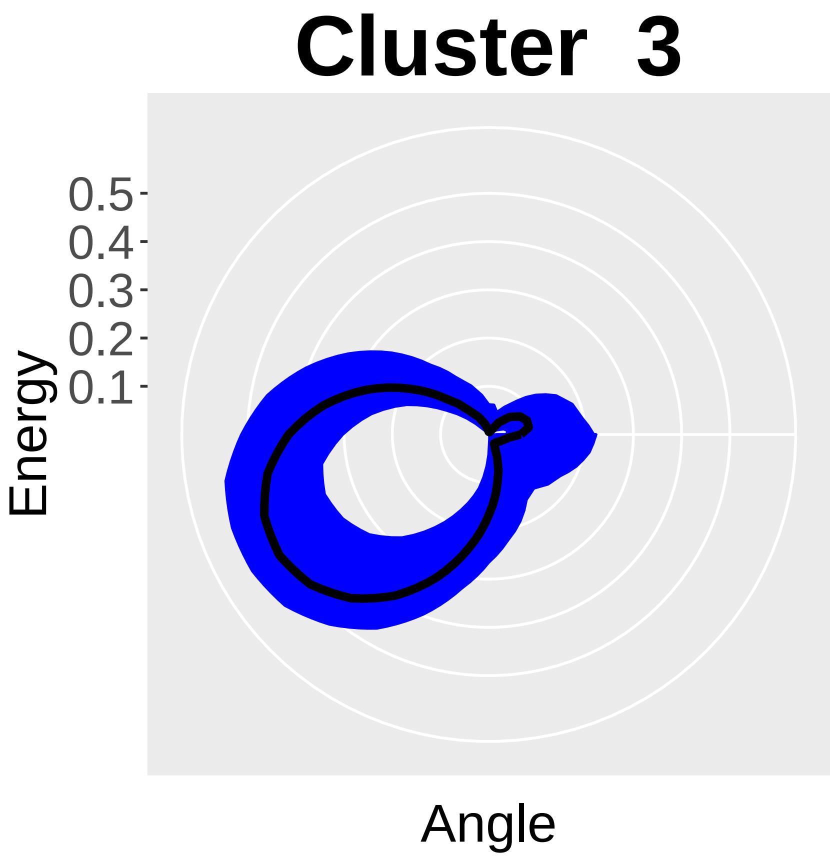

```
#directiona fbplots
print(plotenergy[[1]])
```



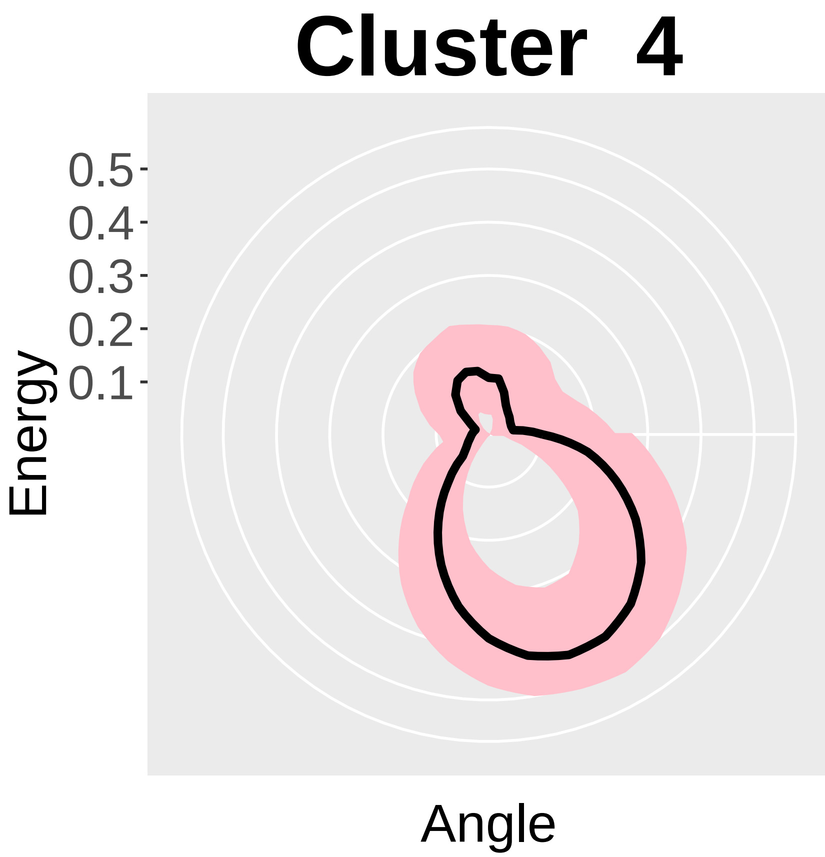

```
print(ploteenergy[[2]])
```



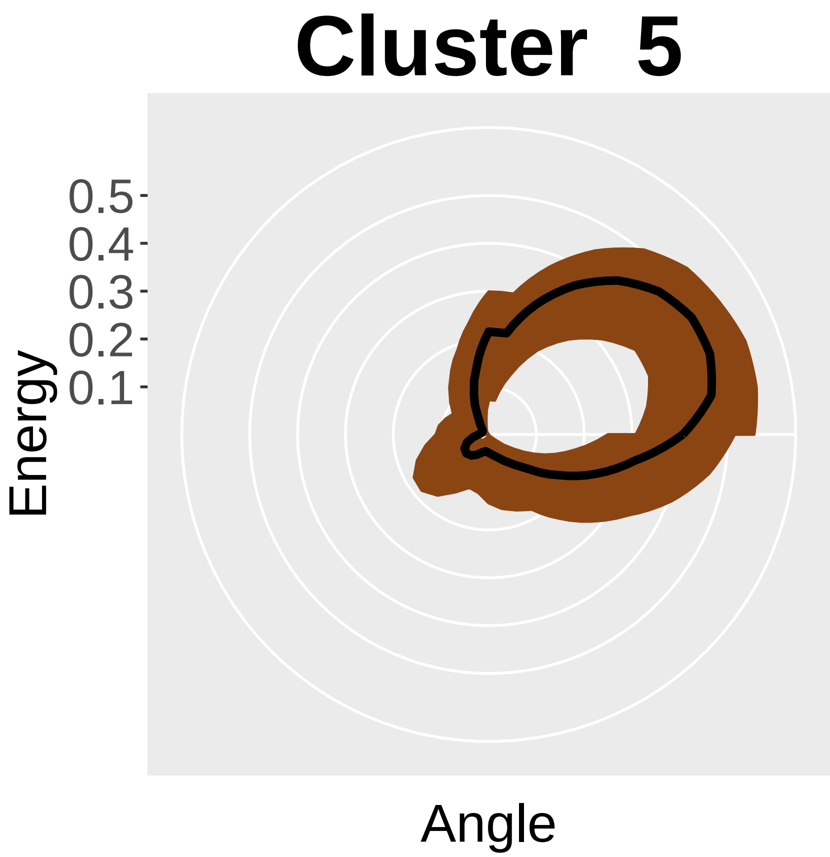

```
print(plotenergy[[3]])
```



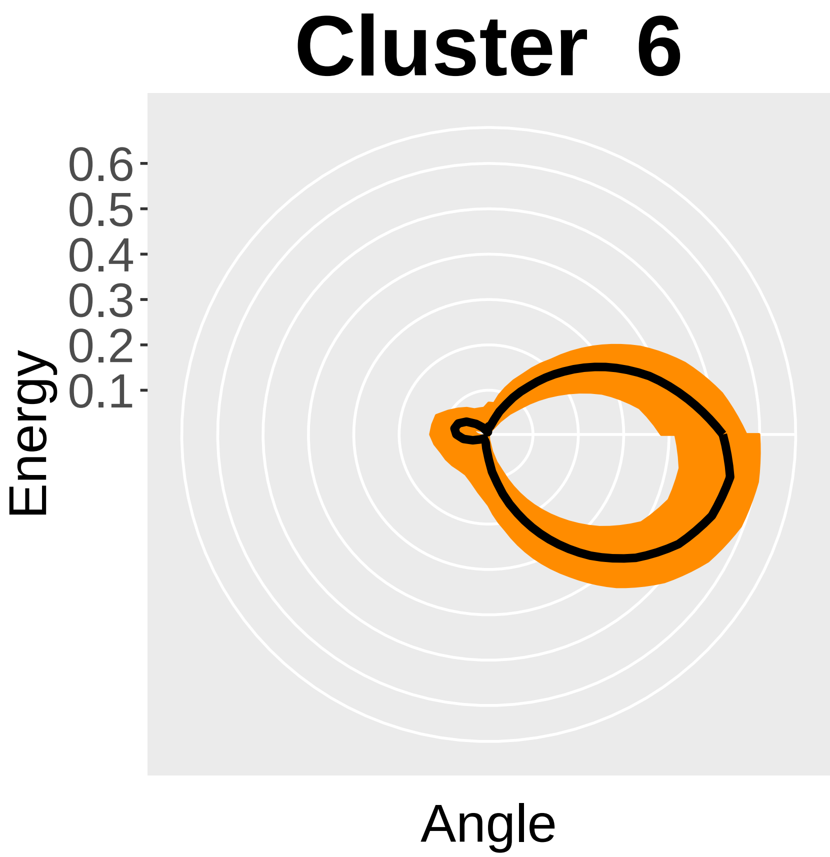

```
print(plotenergy[[4]])
```



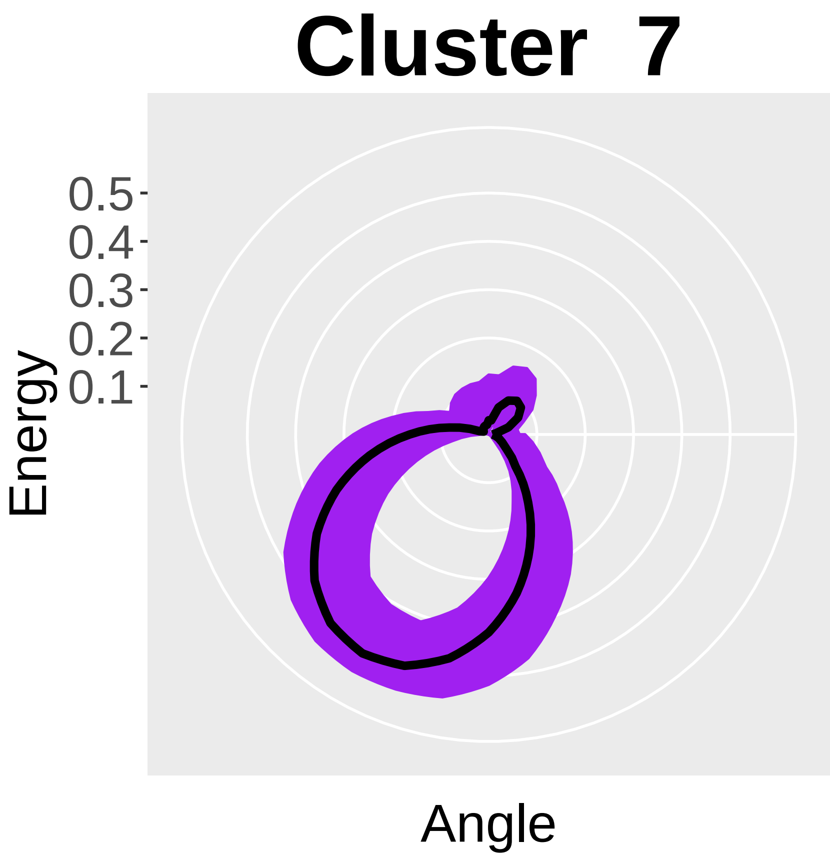

```
print(plotenergy[[5]])
```



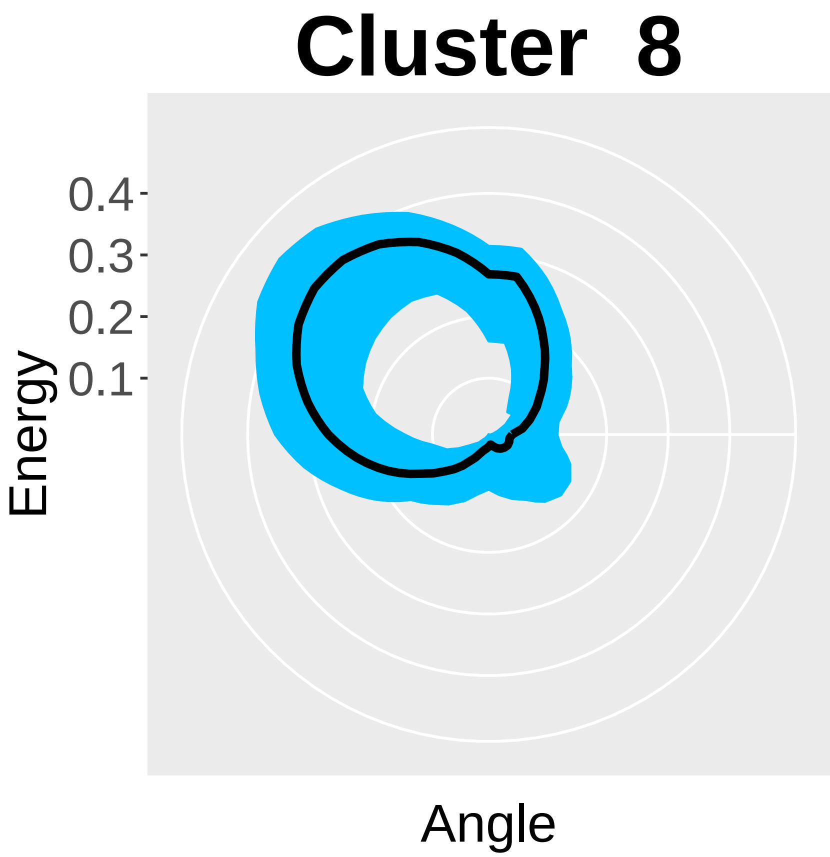

```
print(plotenergy[[6]])
```



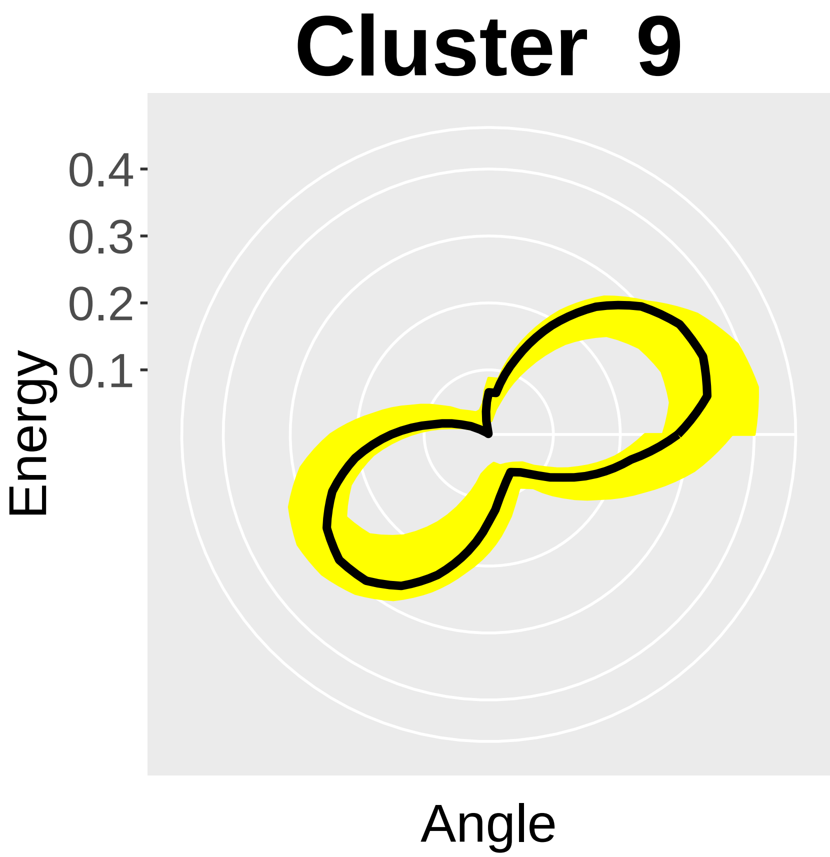

```r
print(plotenergy[[7]])
```



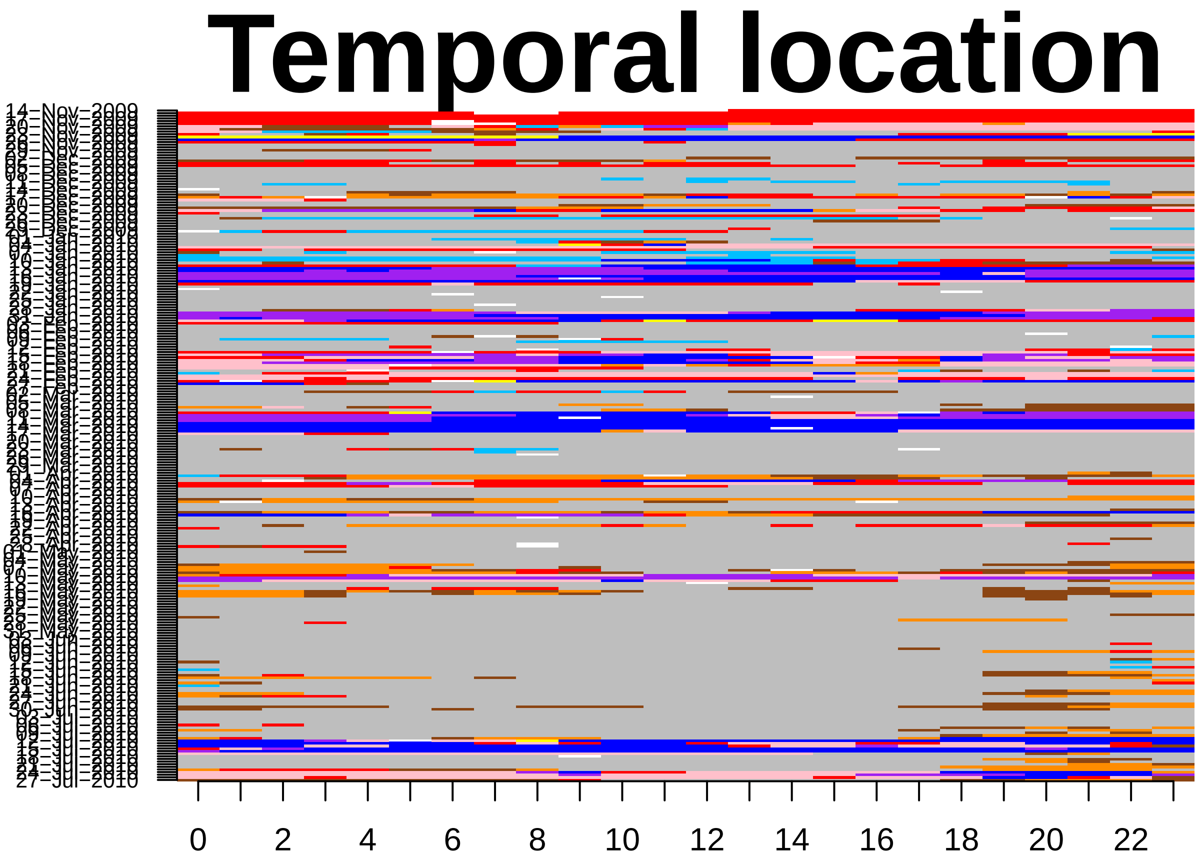

```
print(plotenergy[[8]])
```




```
print(plotenergy[[9]])
```

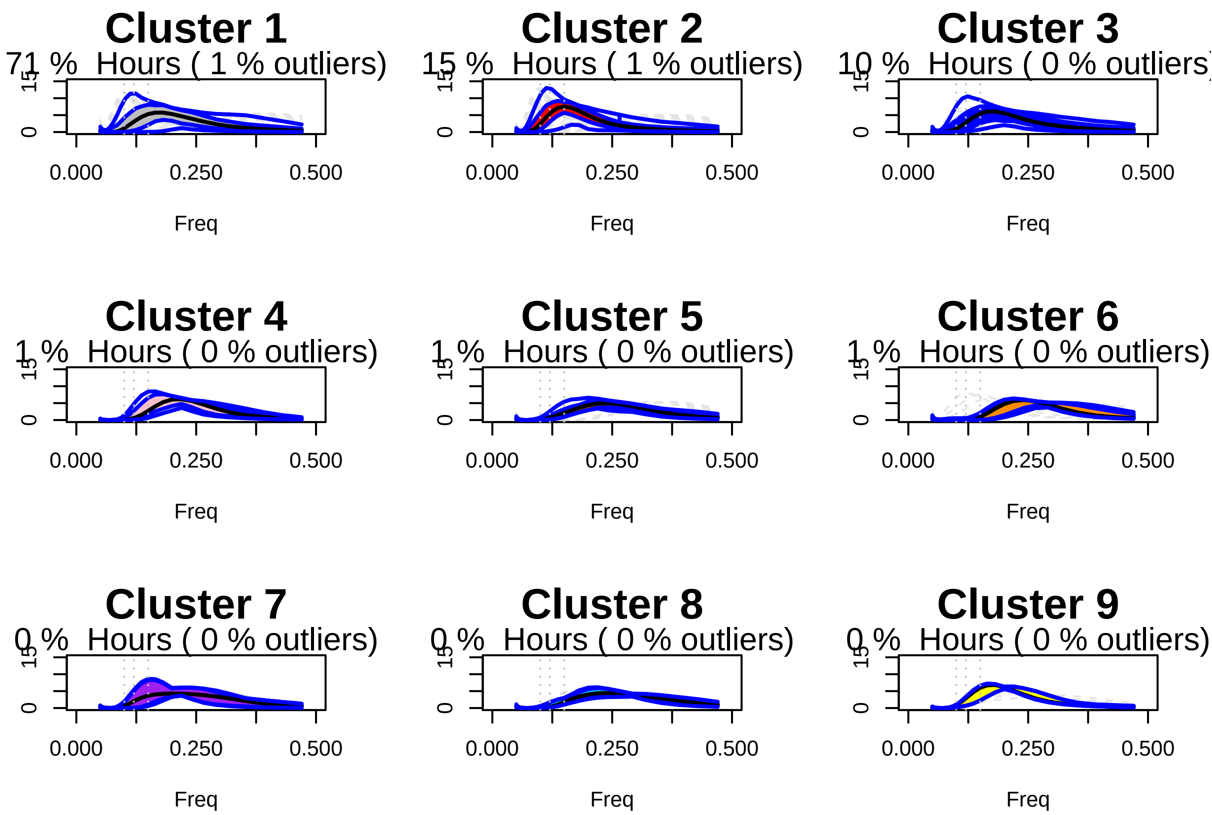



```
#Calendar plot
par(mfrow=c(1,1),mar=c(2,5,3,1))
image(MatClust[,rev(1:256)],col=colors10[-10],axes=FALSE,main="Temporal location",
      cex.main=3.5)
axis(1,at=seq(0,1,length.out = 24),labels = 0:23)
auxlab<-rep(NA,256)
auxlab[seq(1,256,3)]<-rev(dates[seq(1,256,3)])
axis(2,at=seq(0,1,length.out = 256),labels = auxlab
     ,las=2, cex.axis=.7)
```

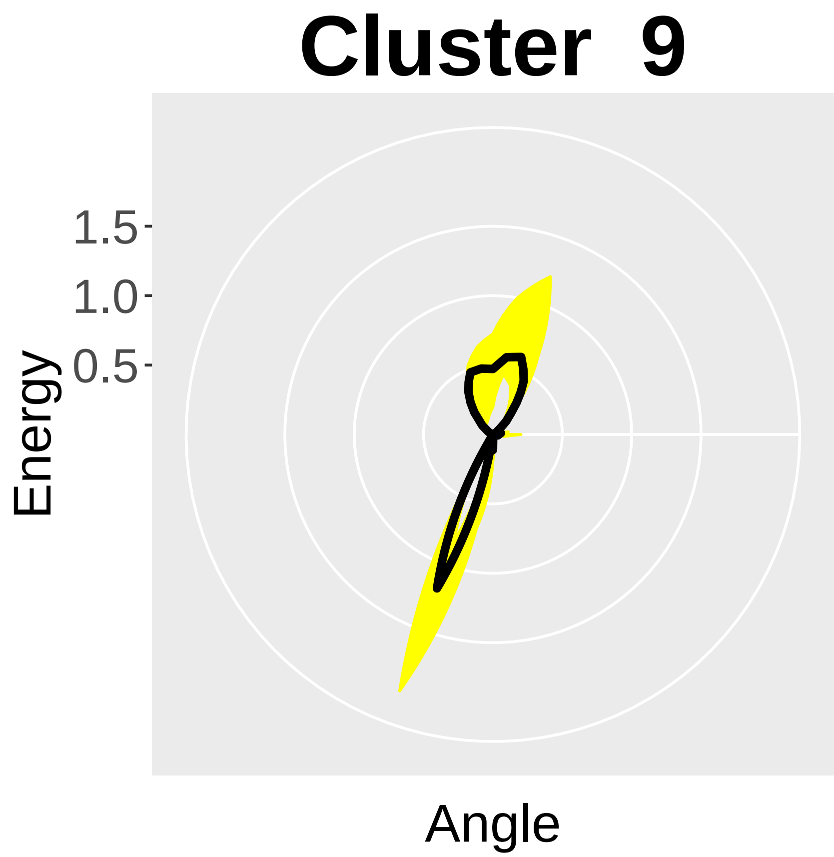



Finally, the clustering visualization for the HDS direction only clustering with simulated data.

```
#fbplots of frequency spectra
par(mfrow=c(3,3))
for(indk in 1:nclust) {
  A<-fbplot(x=w.sim,fit=fw.by.clust[[indk]],col=colors10[indk],xlim = c(0,.5),
            ylim=c(0,15),xaxp=c(0,.5,4),xlab="Freq",ylab="",outliercol="gray90",
            main=paste("Cluster",indk),cex.main=2)
  mtext(paste(round(length(G[[indk]])/nrow(fw),2)*100,"%  Hours (",
              round(length(A$outpoint)/nrow(fw),2)*100,"% outliers)"))
  abline(v=c(.1,.12,.15),col="gray",lty=3)
}
```




```r
#directiona fbplots
print(plotenergy[[1]])
print(plotenergy[[2]])
print(plotenergy[[3]])
print(plotenergy[[4]])
print(plotenergy[[5]])
print(plotenergy[[6]])
print(plotenergy[[7]])
print(plotenergy[[8]])
print(plotenergy[[9]])

#Calendar plot
par(mfrow=c(1,1),mar=c(2,5,3,1))
```




```
image(MatClust[,rev(1:256)],col=colors10[-10],axes=FALSE,main="Temporal location",
      cex.main=3.5)
axis(1,at=seq(0,1,length.out = 24),labels = 0:23)
auxlab<-rep(NA,256)
auxlab[seq(1,256,3)]<-rev(dates[seq(1,256,3)])
axis(2,at=seq(0,1,length.out = 256),labels = auxlab
     ,las=2, cex.axis=.7)
```

# Temporal location