AERO: preliminary analysis of Niger measles data, 1995-2005

```
dim(d)
```

```
## [1]  40 573
```

```
rownames(d)<-d[,1]
d<-d[,-1]
rownames(d)
```

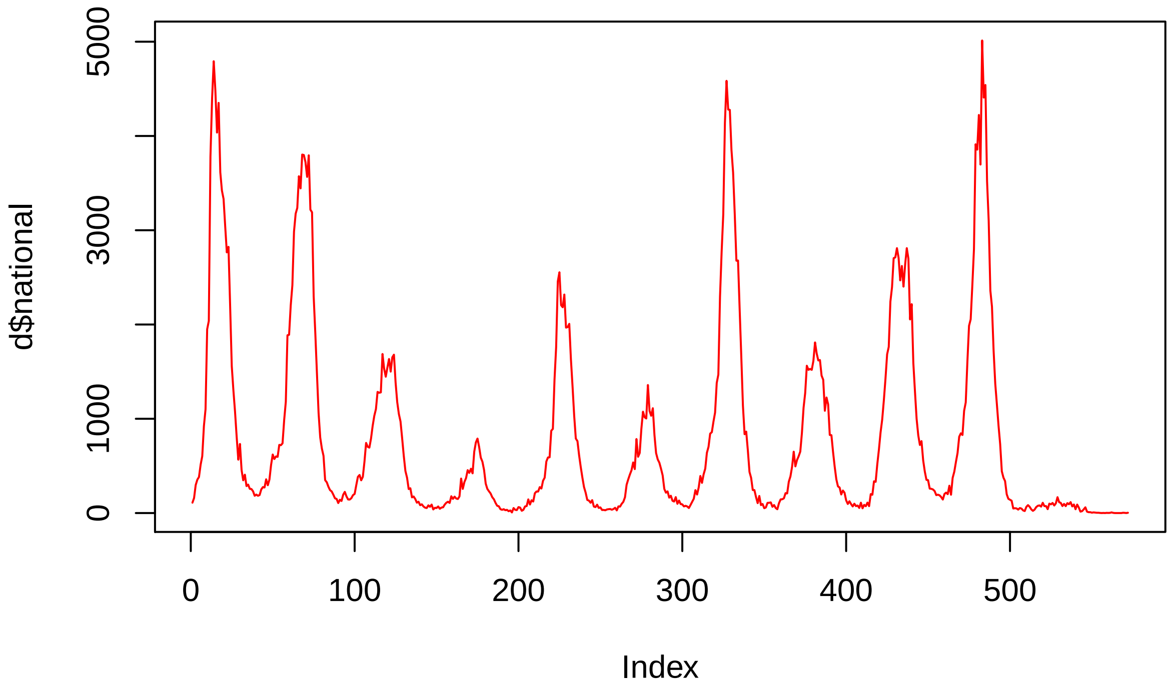
```
##  [1] "Agadez (City)"   "Tchirozerine"    "Arlit"
##  [4] "Bilma"           "Diffa"           "Maine-Sora"
##  [7] "N'Guimi"         "Birni N'Gaour"   "Dogon-Doutchi"
## [10] "Dosso"           "Gaya"            "Loga"
## [13] "Aguie"           "Dakoro"          "Guidan-Roumdji"
## [16] "Madarounfa"      "Maradi (City)"   "Mayahi"
## [19] "Tessaoua"        "Filingue"        "Kollo"
## [22] "Ouallam"         "Say"             "Tera"
## [25] "Tillabery"       "Abalak"          "Birni N'Konni"
## [28] "Bouza"           "Illela"          "Keita"
## [31] "Madaoua"         "Tahoua"          "Tchin-Tabaraden"
## [34] "Goure"           "Magaria"         "Matamey"
## [37] "Miria"           "Tannout"         "Zinder (City)"
## [40] "Niamey (City)"
```

The data frame contains 40 regions and 572 time points spanning 11 years (presumably weeks starting 1 Jan 1995, since 11*52=572). First, transpose the data frame and name the weeks 1 through 572

```
d<-as.data.frame(t(d))
rownames(d)<-1:572
```

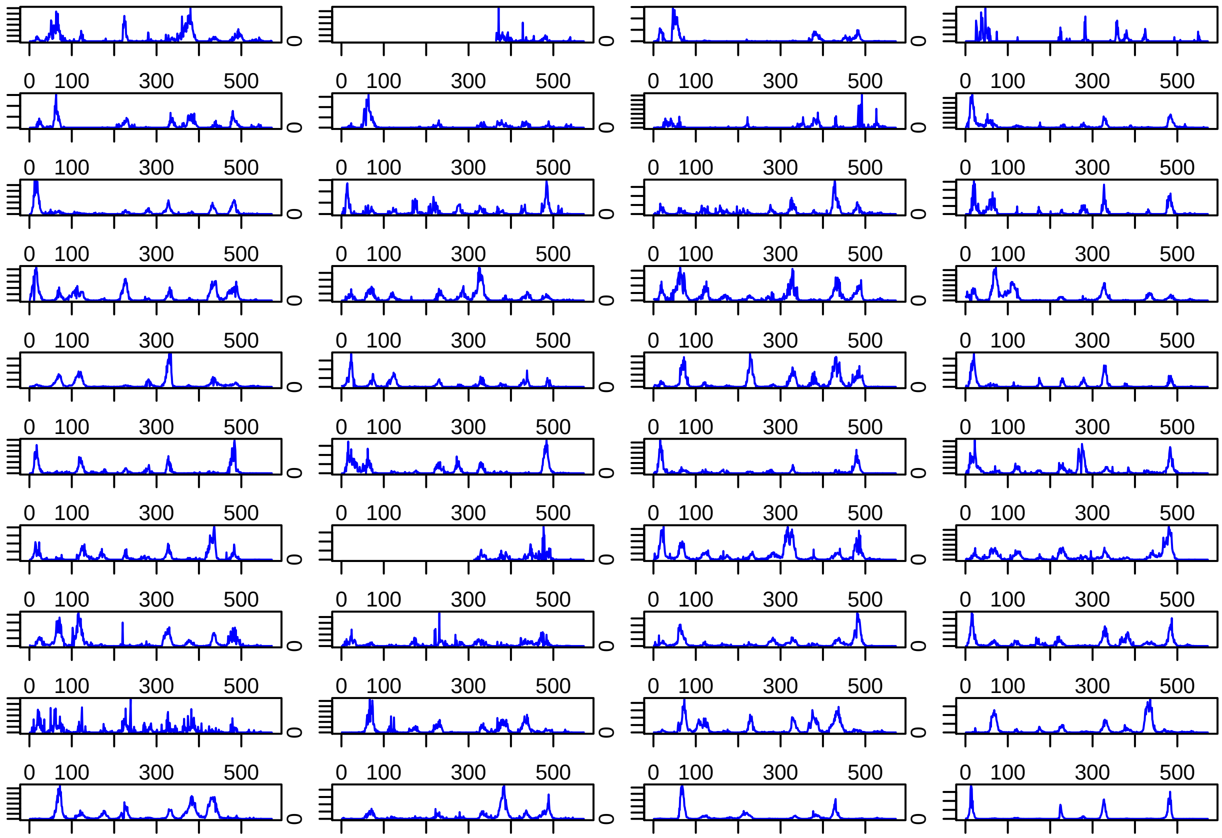
Each entry in the data frame is (presumably) the number of reported measles cases in that region in that week. Now we add a column for the country-level reported measles cases and inspect this

```
d$national<-rowSums(d,na.rm=T)
plot(d$national,type="l",col="red")
```



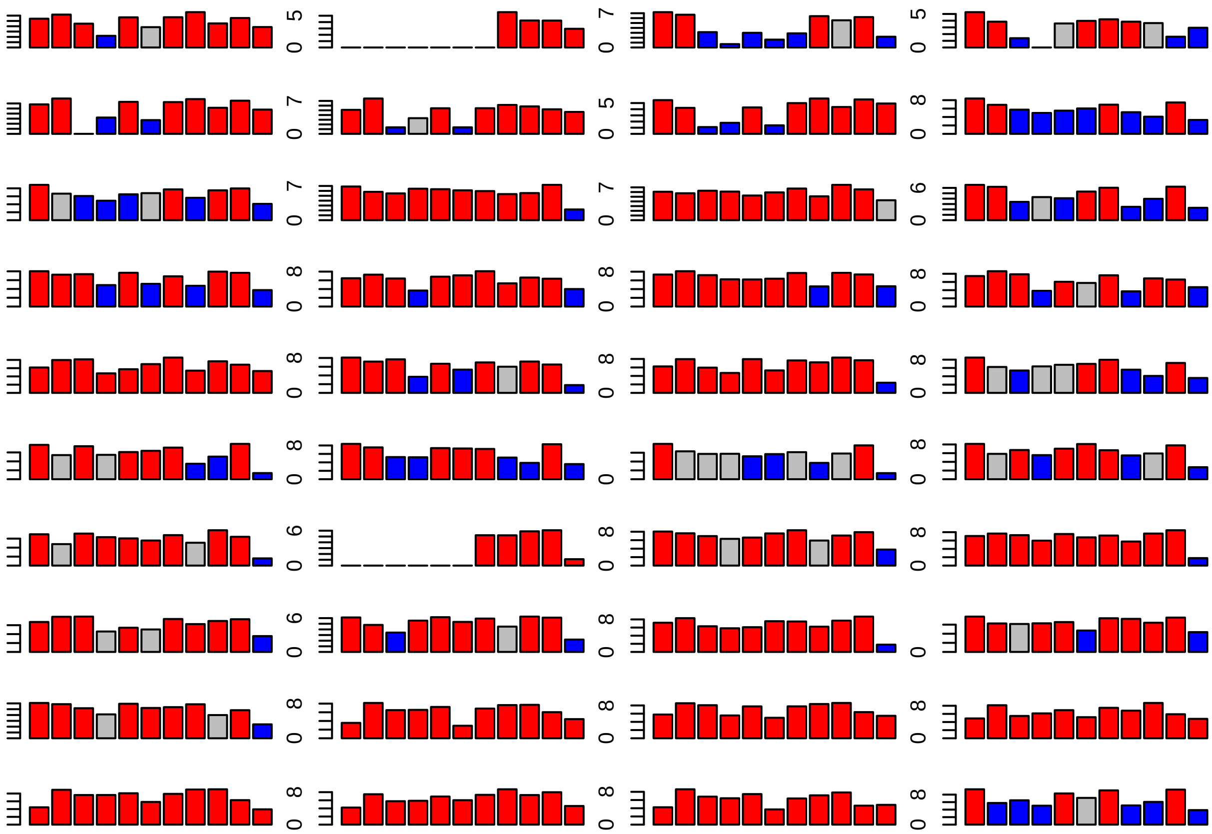
We observe a seaonal (annual) incidence pattern with considerable variation in total cases per year. Many regions have >1 year intervals between large outbreaks. This is posited to be due to weak coupling between regions, resulting in a distribution of import times >1 year. During this time, regions may be building up susceptible numbers (mainly through births) driving the system to criticality.

```
par(mfrow=c(10,4),mar=c(1,1,1,1))
for (i in 1:40){
  plot(d[,i],type="l",col="blue")
}
```



We need a method to establish interepidemic periods. We can aggregate cases by years and look for threshold number of cases.

```
cases.by.year<-NULL
for (i in 1:41){
  tmp<-rep(0,11)
  for (j in 1:572){
    k<-(j-1)%/%52+1
    if (is.na(d[j,i])==F){tmp[k]<-tmp[k]+d[j,i]}
  }
  cases.by.year<-rbind(cases.by.year,tmp)
}
ln.cases.by.year<-log(cases.by.year+1)
par(mfrow=c(10,4),mar=c(1,1,1,1))
for (i in 1:40){
  my.max<-max(cases.by.year[i])
  my.cols<-rep("tbd",11)
  for (j in 1:11){
    my.cols[j]<-ifelse(cases.by.year[i,j]>0.1*my.max,"red","blue") #visually distinguish years with <20
    if (my.cols[j]=="red"){my.cols[j]<-ifelse(cases.by.year[i,j]>0.2*my.max,"red","gray")}
  }
  barplot(ln.cases.by.year[i,],col=my.cols)
}
```



Experimental design: Based on criteria for interepidemic regions/times, we classify these as years with low activity (<10% maximum annual cases) immediately followed by an outbreak year (>20% maximum annual cases). We put the (weekly time series for the) first year (low activity year) in one list (interepidemic) and an equivalent number of time series' (non-interepidemic, currently outbreak - but should probably be changed to random) in another list

```
#define thresholds for low and high activity
lt4iepi<-0.1
gt4epi<-0.2
# append a list
lappend <- function(lst, obj) {
  lst[[length(lst)+1]] <- obj
  return(lst)
}
inter.epi<-list()
non.inter.epi<-list()
for (i in 1:40){
  for (j in 1:8){
    if ((cases.by.year[i,j]<lt4iepi*max(cases.by.year[i,]))&(cases.by.year[i,(j+1)]>gt4epi*max(cases.by
      inter.epi<-lappend(inter.epi,d[seq((j-1)*52+1,(j-1)*52+(2*52),1),i])
    }
    if ((cases.by.year[i,j]>gt4epi*max(cases.by.year[i,]))&(cases.by.year[i,(j+1)]>gt4epi*max(cases.by.
      non.inter.epi<-lappend(non.inter.epi,d[seq((j-1)*52+1,(j-1)*52+(2*52),1),i])
    }
  }
}
inter.epi.2<-list()
```

```r
non.inter.epi.2<-list()
for (i in 1:length(inter.epi)){
  if (max(is.na(inter.epi[[i]]))==0){inter.epi.2<-lappend(inter.epi.2,inter.epi[[i]])}
}
for (i in 1:length(non.inter.epi)){
  if (max(is.na(non.inter.epi[[i]]))==0){non.inter.epi.2<-lappend(non.inter.epi.2,non.inter.epi[[i]])}
}
```

Now we see if the Kendall's tau statistic for a given EWS statistic is different for time series of the two lists

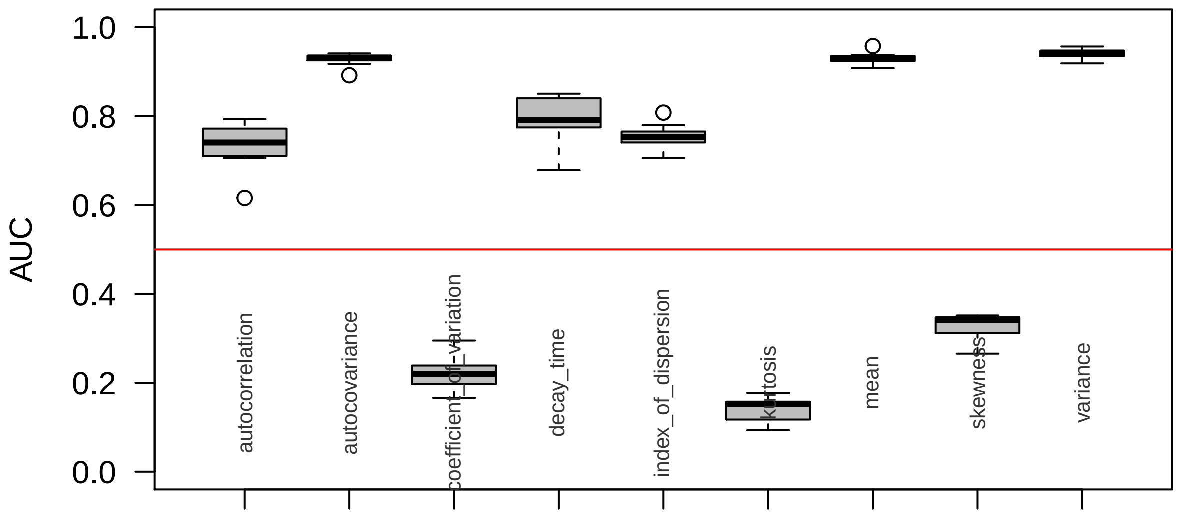
```r
library(VGAM)
```

```
## Loading required package: stats4
## Loading required package: splines
```

```r
library(spaero)
cf.tau<-NULL
for (i in 1:length(inter.epi.2)){
  st1<-get_stats(inter.epi.2[[i]],center_kernel="uniform",center_trend="local_constant",center_bandwidt
  cf.tau<-rbind(cf.tau,c(1,kendall.tau(seq(1,2*52,1),st1$stats$variance),kendall.tau(seq(1,2*52,1),st1$
}
for (i in 1:length(non.inter.epi.2)){
  st1<-get_stats(non.inter.epi.2[[i]],center_kernel="uniform",center_trend="local_constant",center_band
  cf.tau<-rbind(cf.tau,c(0,kendall.tau(seq(1,2*52,1),st1$stats$variance),kendall.tau(seq(1,2*52,1),st1$
}

cf.tau<-as.data.frame(cf.tau)
names(cf.tau)<-c("label",names(st1$stats))
cf.tau.orig<-cf.tau
my.aucs<-NULL
for (i in 1:9){
  tmp<-cf.tau[,c(1,(i+1))]
  real.name<-names(tmp)[2]
  names(tmp)[2]<-"stat.of.interest"
  tmp<-tmp[order(tmp$stat.of.interest),]
  pos1<-which(tmp$label==1)#position in ranked list of 1's
  n1<-length(pos1)#number of 1's in vector
  pos0<-which(tmp$label==0)
  n0<-length(pos0)
  auc<-0
  for (j in 1:n1){
    auc<-auc+(pos1[j]-j)
  }
  auc<-auc/(n0*n1)
  my.aucs<-rbind(my.aucs,c(real.name,auc))
}

my.aucs<-as.data.frame(my.aucs)
names(my.aucs)<-c("EWS","AUC")
my.aucs$AUC<-as.numeric(as.character(my.aucs$AUC))
```

```
bstrap.num<-10
for (i in 1:(bstrap.num-1)){# add to original to make n=10,100 bootstrap
my.subsample.nonepi<-sample(1:length(non.inter.epi.2),replace=T)
non.inter.epi.3<-non.inter.epi.2[my.subsample.nonepi]
my.subsample.epi<-sample(1:length(inter.epi.2),replace=T)
inter.epi.3<-inter.epi.2[my.subsample.epi]

cf.tau<-NULL
for (i in 1:length(inter.epi.3)){
  st1<-get_stats(inter.epi.3[[i]],center_kernel="uniform",center_trend="local_constant",center_bandwidt
  cf.tau<-rbind(cf.tau,c(1,kendall.tau(seq(1,2*52,1),st1$stats$variance),kendall.tau(seq(1,2*52,1),st1$
}
for (i in 1:length(non.inter.epi.3)){
  st1<-get_stats(non.inter.epi.2[[i]],center_kernel="uniform",center_trend="local_constant",center_band
  cf.tau<-rbind(cf.tau,c(0,kendall.tau(seq(1,2*52,1),st1$stats$variance),kendall.tau(seq(1,2*52,1),st1$
}

cf.tau<-as.data.frame(cf.tau)
names(cf.tau)<-c("label",names(st1$stats))

my.aucs.2<-NULL
for (i in 1:9){
  tmp<-cf.tau[,c(1,(i+1))]
  real.name<-names(tmp)[2]
  names(tmp)[2]<-"stat.of.interest"
  tmp<-tmp[order(tmp$stat.of.interest),]
  pos1<-which(tmp$label==1)#position in ranked list of 1's
  pos0<-which(tmp$label==0)
  n1<-length(pos1)#number of 1's in vector
  n0<-length(pos0)
  auc<-0
  for (j in 1:n1){
    auc<-auc+(pos1[j]-j)
  }
  auc<-auc/(n0*n1)
  my.aucs.2<-rbind(my.aucs.2,c(real.name,auc))
}
my.aucs.2<-as.data.frame(my.aucs.2)
names(my.aucs.2)<-c("EWS","AUC")
my.aucs.2$AUC<-as.numeric(as.character(my.aucs.2$AUC))
my.aucs<-merge(my.aucs,my.aucs.2,by="EWS")

}

par(mfrow=c(1,1),mar=c(5,5,5,1))

ews.list<-list()
my.aucs.vals<-t(my.aucs[,2:(bstrap.num+1)])
for (i in 1:9){
  ews.list<-lappend(ews.list,unname(my.aucs.vals[,i]))
}

boxplot(ews.list,names=NA,cex.axis=1.0,las=1,ylim=c(0,1),col="gray",ylab="AUC")
```

```r
lines(c(0.12,9.88),c(0.5,0.5),col="red")
text(1:9,rep(0.2,9),my.aucs$EWS,srt=90,col="gray20",cex=0.7)
```



Check degeneracy of EWS statistics under a scrambling protocol (label swap)

```r
epi.scramble<-list()
non.epi.scramble<-list()

for (j in 1:length(non.inter.epi.2)){
  coin<-rbinom(1,1,0.5)
  ifelse(coin==0,epi.scramble<-lappend(epi.scramble,non.inter.epi.2[[j]]),non.epi.scramble<-lappend(non
}
for (j in 1:length(inter.epi.2)){
  coin<-rbinom(1,1,0.5)
  ifelse(coin==0,epi.scramble<-lappend(epi.scramble,inter.epi.2[[j]]),non.epi.scramble<-lappend(non.epi
}

cf.tau<-NULL
for (i in 1:length(epi.scramble)){
  st1<-get_stats(epi.scramble[[i]],center_kernel="uniform",center_trend="local_constant",center_bandwid
  cf.tau<-rbind(cf.tau,c(1,kendall.tau(seq(1,2*52,1),st1$stats$variance),kendall.tau(seq(1,2*52,1),st1$
}
for (i in 1:length(non.epi.scramble)){
  st1<-get_stats(non.epi.scramble[[i]],center_kernel="uniform",center_trend="local_constant",center_banc
  cf.tau<-rbind(cf.tau,c(0,kendall.tau(seq(1,2*52,1),st1$stats$variance),kendall.tau(seq(1,2*52,1),st1$
}

cf.tau<-as.data.frame(cf.tau)
names(cf.tau)<-c("label",names(st1$stats))
cf.tau.orig<-cf.tau
my.aucs<-NULL
for (i in 1:9){
  tmp<-cf.tau[,c(1,(i+1))]
  real.name<-names(tmp)[2]
  names(tmp)[2]<-"stat.of.interest"
  tmp<-tmp[order(tmp$stat.of.interest),]
  pos1<-which(tmp$label==1)#position in ranked list of 1's
```

```
  n1<-length(pos1)#number of 1's in vector
  pos0<-which(tmp$label==0)
  n0<-length(pos0)
  auc<-0
  for (j in 1:n1){
    auc<-auc+(pos1[j]-j)
  }
  auc<-auc/(n0*n1)
  my.aucs<-rbind(my.aucs,c(real.name,auc))
}

my.aucs<-as.data.frame(my.aucs)
names(my.aucs)<-c("EWS","AUC")
my.aucs$AUC<-as.numeric(as.character(my.aucs$AUC))
```

```
bstrap.num<-10
for (i in 1:(bstrap.num-1)){# add to original to make n=10,100 bootstrap
my.subsample.nonepi<-sample(1:length(non.epi.scramble),replace=T)
non.inter.epi.scram<-non.epi.scramble[my.subsample.nonepi]
my.subsample.epi<-sample(1:length(epi.scramble),replace=T)
inter.epi.scram<-epi.scramble[my.subsample.epi]

cf.tau<-NULL
for (i in 1:length(inter.epi.scram)){
  st1<-get_stats(inter.epi.scram[[i]],center_kernel="uniform",center_trend="local_constant",center_bandw
  cf.tau<-rbind(cf.tau,c(1,kendall.tau(seq(1,2*52,1),st1$stats$variance),kendall.tau(seq(1,2*52,1),st1$s
}
for (i in 1:length(non.inter.epi.scram)){
  st1<-get_stats(non.inter.epi.scram[[i]],center_kernel="uniform",center_trend="local_constant",center_b
  cf.tau<-rbind(cf.tau,c(0,kendall.tau(seq(1,2*52,1),st1$stats$variance),kendall.tau(seq(1,2*52,1),st1$s
}

cf.tau<-as.data.frame(cf.tau)
names(cf.tau)<-c("label",names(st1$stats))

my.aucs.2<-NULL
for (i in 1:9){
  tmp<-cf.tau[,c(1,(i+1))]
  real.name<-names(tmp)[2]
  names(tmp)[2]<-"stat.of.interest"
  tmp<-tmp[order(tmp$stat.of.interest),]
  pos1<-which(tmp$label==1)#position in ranked list of 1's
  pos0<-which(tmp$label==0)
  n1<-length(pos1)#number of 1's in vector
  n0<-length(pos0)
  auc<-0
  for (j in 1:n1){
    auc<-auc+(pos1[j]-j)
    #auc<-auc+(pos1[j]-n1*(n1-1)/2)
  }
  auc<-auc/(n0*n1)
  my.aucs.2<-rbind(my.aucs.2,c(real.name,auc))
}
```

```
my.aucs.2<-as.data.frame(my.aucs.2)
names(my.aucs.2)<-c("EWS","AUC")
my.aucs.2$AUC<-as.numeric(as.character(my.aucs.2$AUC))
my.aucs<-merge(my.aucs,my.aucs.2,by="EWS")

}

par(mfrow=c(1,1),mar=c(5,5,5,1))

ews.list<-list()
my.aucs.vals<-t(my.aucs[,2:(bstrap.num+1)])
for (i in 1:9){
  ews.list<-lappend(ews.list,unname(my.aucs.vals[,i]))
}

boxplot(ews.list,names=NA,cex.axis=1.0,las=1,ylim=c(0,1),col="gray",ylab="AUC")
lines(c(0.12,9.88),c(0.5,0.5),col="red")
text(1:9,rep(0.2,9),my.aucs$EWS,srt=90,col="gray20",cex=0.7)
```