

Time warping

W. Grossmann

```
#=====
#Libraries
#=====
library(dtw) # for time warping

## Loading required package: proxy

##
## Attaching package: 'proxy'

## The following objects are masked from 'package:stats':
##
##   as.dist, dist

## The following object is masked from 'package:base':
##
##   as.matrix

## Loaded dtw v1.18-1. See ?dtw for help, citation("dtw") for use in publicat
ion.

library(lattice) # for graphics
library(ggplot2) # for graphics
library(cluster) # for clustering
library(gridExtra) # for position of graphics
#=====
#Data
#=====
data1<- read.csv("Logistic_stacked1.csv", sep=";", dec=",")
head(data1)

##   Group ID Time Index Temperature
## 1 normal 1   1    1    28.33253
## 2 normal 1   2   101    29.67492
## 3 normal 1   3   201    24.92381
## 4 normal 1   4   301    29.61756
## 5 normal 1   5   401    32.37686
## 6 normal 1   6   501    33.22257

group<-data1$Group[match(1:100,data1$ID)]

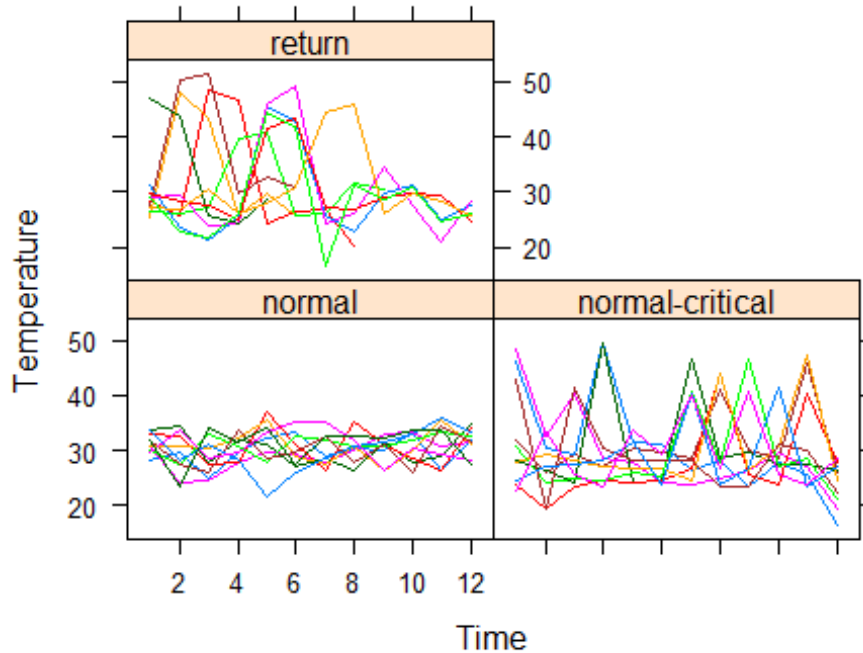
table(group)

## group
##      normal normal-critical      return
##      50      20      30
```

```

#=====
#Visualization
#=====
index<-c(1:10, 54:63,81:90)
data2<-subset(data1, ID %in% index)
#data2
xyplot(Temperature~Time|Group,data2,type="l",groups=ID)

```



```

#=====
# Compute warping distance for selected elements
#=====
normal_test<-subset(data1, select=Temperature, ID==50)
return_test<-subset(data1, select=Temperature, ID==80)
critical_test<-subset(data1, select=Temperature, ID==100)
#-----
#normal cases
#-----
distance_normal<-numeric(100)
for(i in 1:100){
ref<-subset(data1, select=Temperature, ID== i)
normal_test<-subset(data1, select=Temperature, ID==50)#Test sequence
align<-dtw(normal_test,ref)
distance_normal[i]<-align$distance
}
#-----
# Finding best matches in the classes
#-----

```

```

best_normal<-min(distance_normal[1:49])
index_best_normal<-which.min(distance_normal[1:49])
best_return<-min(distance_normal[51:80])
index_best_return<-which.min(distance_normal[51:80])
best_critical<-min(distance_normal[81:100])
index_best_critical<-which.min(distance_normal[81:100])
c("index",index_best_normal,index_best_critical+80,index_best_return+50)

## [1] "index" "9"      "81"      "68"

c("distance",best_normal,best_critical,best_return)

## [1] "distance"      "23.15909352" "53.84922236" "51.0411706"

#-----
#normal-critical cases
#-----
distance_critical<-numeric(100)
for(i in 1:100){
  ref<-subset(data1, select=Temperature, ID== i)
  critical_test<-subset(data1, select=Temperature, ID==100)#Test sequence
  align<-dtw(critical_test,ref)
  distance_critical[i]<-align$distance
}
#-----
#Finding best matches in the classes
#-----
best_normal<-min(distance_critical[1:50])
index_best_normal<-which.min(distance_critical[1:50])
best_return<-min(distance_critical[51:80])
index_bestreturn<-which.min(distance_critical[51:80])
best_critical<-min(distance_critical[81:99])
index_best_critical<-which.min(distance_critical[81:99])
c("index",index_best_normal,index_best_critical+80,index_best_return+50)

## [1] "index" "31"      "97"      "68"

c("distance",best_normal,best_critical,best_return)

## [1] "distance"      "56.3425404" "39.10064922" "43.17697031"

#-----
#return cases
#-----
distance_return<-numeric(100)
for(i in 1:100){
  ref<-subset(data1, select=Temperature, ID== i)
  return_test<-subset(data1, select=Temperature, ID==80)#Test sequence
  align<-dtw(return_test,ref)
  distance_return[i]<-align$distance
}
#-----

```

```

#Finding best matches in the classes
#-----
best_normal<-min(distance_return[1:50])
index_best_normal<-which.min(distance_return[1:50])
best_return<-min(distance_return[51:79])
index_best_return<-which.min(distance_return[51:79])
best_critical<-min(distance_return[81:100])
index_best_critical<-which.min(distance_return[81:100])
c("index",index_best_normal,index_best_critical+80,index_best_return+50)

## [1] "index" "47"      "96"      "77"

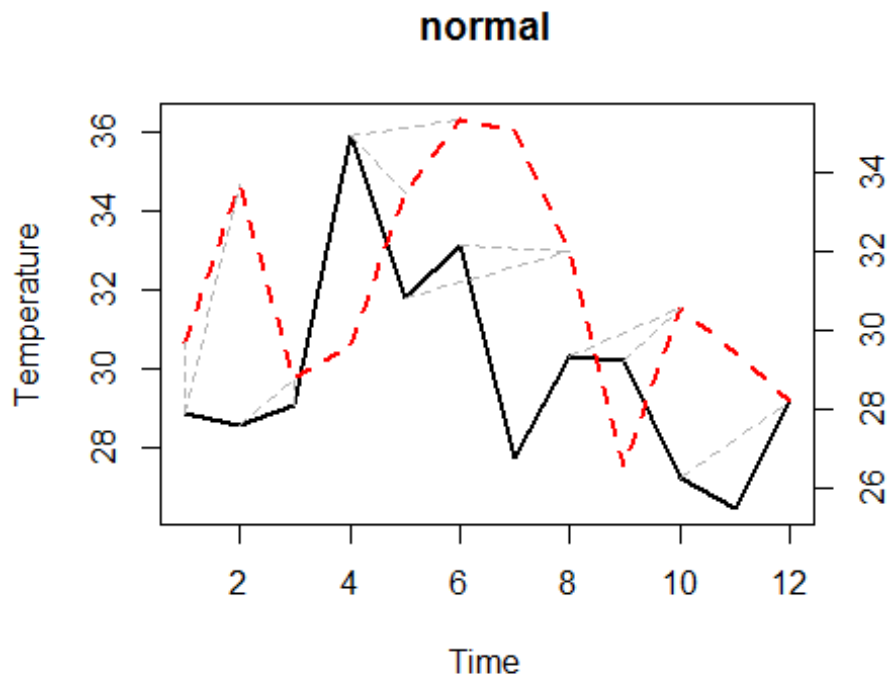
c("distance",best_normal,best_critical,best_return)

## [1] "distance"      "57.40261337" "42.56197617" "20.88310975"

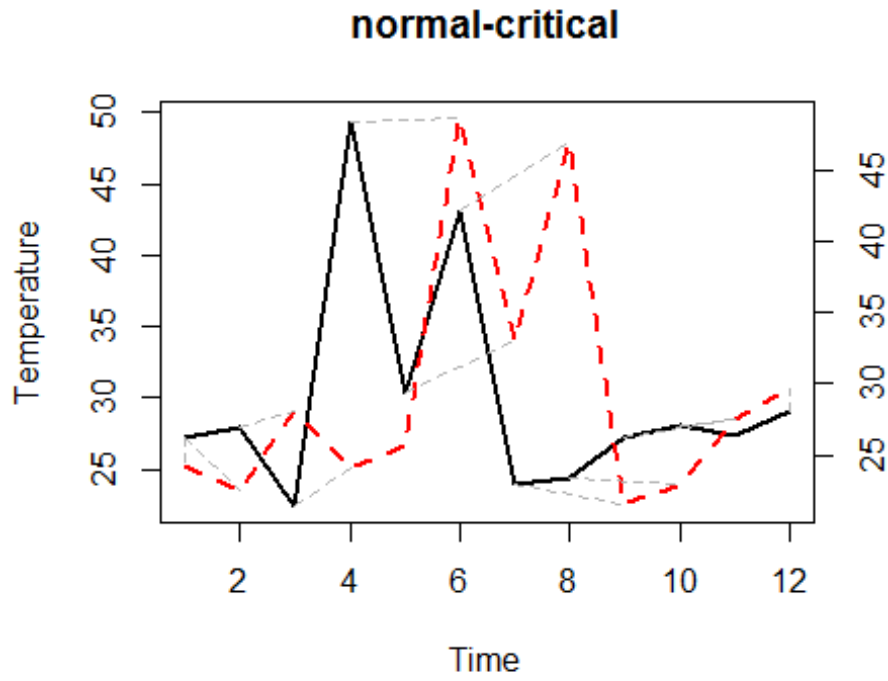
#=====
#Visualization of alignments
#=====
normal_ref<-subset(data1, select=Temperature, ID==9)
return_ref<-subset(data1, select=Temperature, ID==77)
critical_ref<-subset(data1, select=Temperature, ID==97)

plot(dtw(normal_test, normal_ref, k = TRUE), type = "two", off = 1, match.lty
= 2, lwd=2,match.indices = 12, xlab= "Time", ylab="Temperature", main = "norm
al")

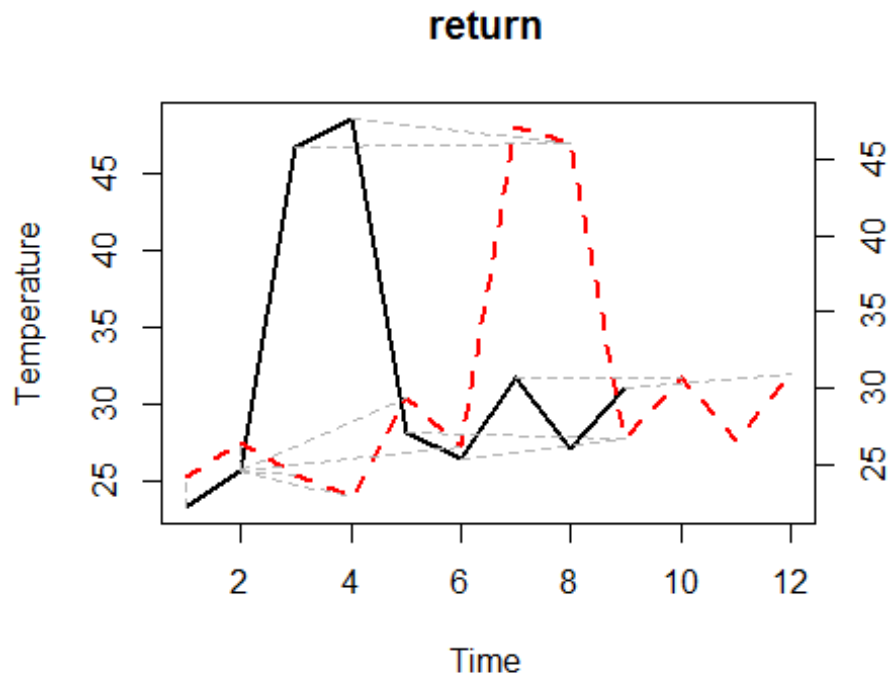
```



```
plot(dtw(critical_test, critical_ref, k = TRUE), type = "two", off = 1, match
.lty = 2,lwd = 2,match.indices = 12, xlab= "Time", ylab="Temperature", main="
normal-critical")
```



```
plot(dtw(return_test, return_ref, k = TRUE), type = "two", off = 1, match.lty
= 2,lwd = 2,match.indices = 12, xlab= "Time", ylab="Temperature", main = "ret
urn")
```



```

#=====
#Distance Matrix
#=====
100*99/2

## [1] 4950

simi<-diag(100)
for(i in 1:100){
  ref<-subset(data1, select=Temperature, ID== i)
  for(j in 1:100){
    comp<-subset(data1, select=Temperature, ID==j)
    align<-dtw(comp,ref)
    simi[i,j]<-align$distance
  }
}
simi[50,9]

## [1] 23.15909

simi[50,50]

## [1] 0

d1<-as.dist(simi)

#=====
# 1-NN Classification

```

```

#=====

sim1<-simi
diag(sim1)<-999

index<-numeric(100)
dist<-numeric(100)
class<-numeric(100)
table(group)

## group
##          normal normal-critical          return
##          50          20          30

for(i in 1:100){
  dist[i]<-min(sim1[i,1:100])
  index[i]<-which.min(sim1[i,1:100])
  group<-data1$Group[match(1:100,data1$ID)]
  class[i]<-group[index[i]]
}
table(class,group)

##      group
## class normal normal-critical return
##    1     50          1         0
##    2      0          8         0
##    3      0         11        30

index

## [1] 42 31 22 43 33 48 25 6 50 34 46 31 25 48 2 40 21
## [18] 26 43 49 44 44 44 37 7 18 48 44 24 40 12 1 42 6
## [35] 30 6 24 34 50 30 12 33 36 22 6 11 17 49 48 9 64
## [52] 56 69 55 71 52 80 55 54 64 55 55 58 60 51 61 54 61
## [69] 55 55 55 57 71 71 71 60 80 62 54 77 62 65 65 96 71
## [86] 61 54 23 52 52 85 95 95 76 92 84 100 69 51 97

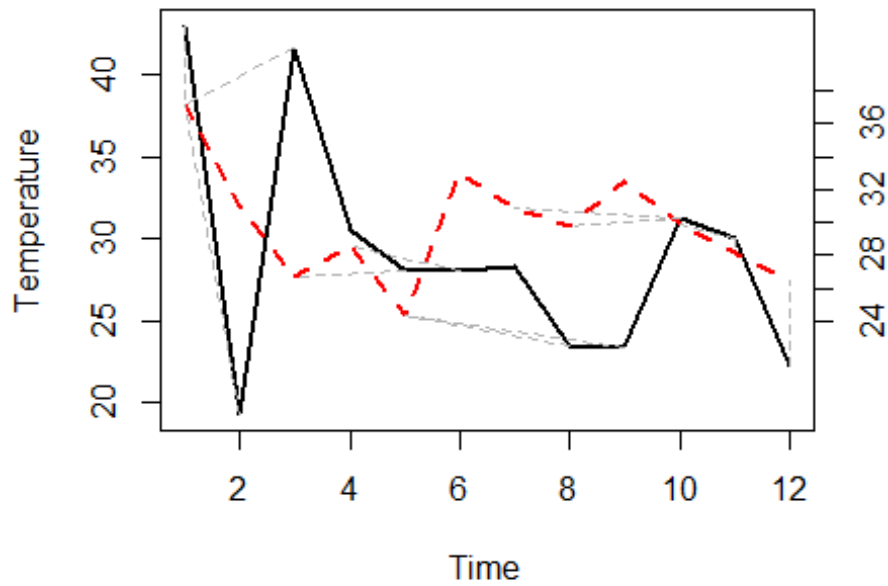
#-----
#Visualization of Misclassified cases
#-----

normal_test<-subset(data1, select=Temperature, ID==88)
return_test<-subset(data1, select=Temperature, ID==81)
critical_test<-subset(data1, select=Temperature, ID==100)
normal_ref<-subset(data1, select=Temperature, ID==23)
return_ref<-subset(data1, select=Temperature, ID==62)
critical_ref<-subset(data1, select=Temperature, ID==97)

fig1<-plot(dtw(normal_test, normal_ref, k = TRUE), type = "two", off = 1, mat
ch.lty = 2, lwd=2, match.indices = 12, xlab= "Time", ylab="Temperature", main
= "wrong normal")

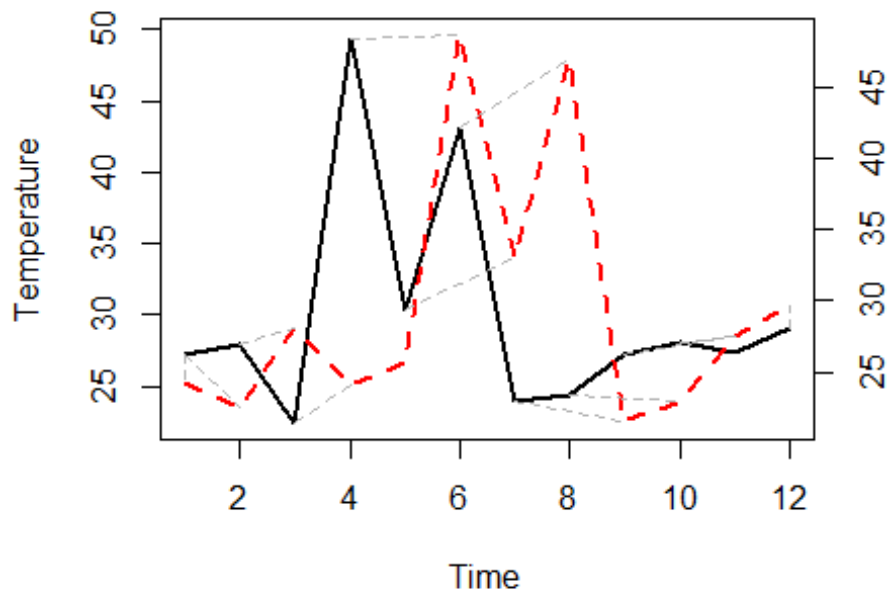
```

wrong normal



```
fig2<-plot(dtw(critical_test, critical_ref, k = TRUE), type = "two", off = 1,
match.lty = 2, lwd=2,match.indices = 12, xlab= "Time", ylab="Temperature", ma
in = "correct critical")
```

correct critical




```
fig3<-plot(dtw(return_test, return_ref, k = TRUE), type = "two", off = 1, mat  
ch.lty = 2, lwd=2, match.indices = 12, xlab= "Time", ylab="Temperature", main  
= "wrong return")
```

