DTW-based discord discovery uses sequence windows (seq > 1) and flags sequences far from any centroid as discords. Steps:

* Load and visualize a dataset with repeating sequences
* Configure and run hanct\_dtw(seq > 1)
* Inspect detections, evaluate, and plot results

# Install Harbinger (if needed)  
#install.packages("harbinger")

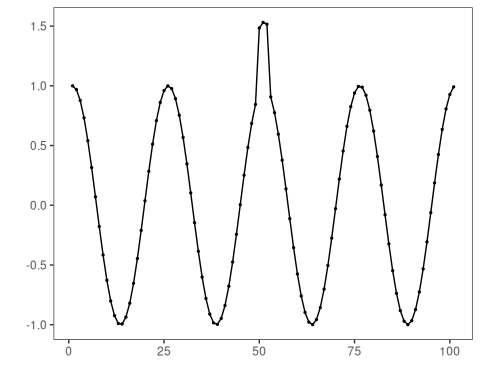
# Load required packages  
library(daltoolbox)  
library(harbinger)

# Load example anomaly datasets  
data(examples\_anomalies)

# Select the sequence dataset  
dataset <- examples\_anomalies$sequence  
head(dataset)

## serie event  
## 1 1.0000000 FALSE  
## 2 0.9689124 FALSE  
## 3 0.8775826 FALSE  
## 4 0.7316889 FALSE  
## 5 0.5403023 FALSE  
## 6 0.3153224 FALSE

# Plot the raw time series  
har\_plot(harbinger(), dataset$serie)



# Configure DTW-clustering for sequence discords (seq = 3)  
model <- hanct\_dtw(3)

# Fit the detector  
model <- fit(model, dataset$serie)

# Run detection  
detection <- detect(model, dataset$serie)

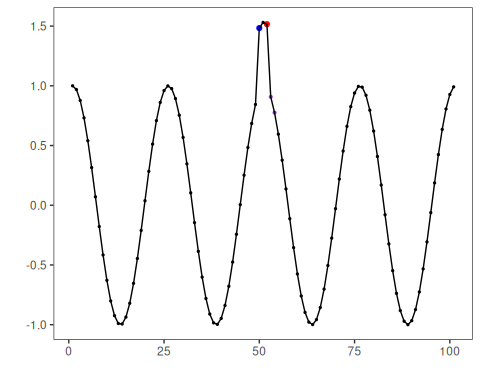
# Show detected discord starts  
print(detection |> dplyr::filter(event == TRUE))

## idx event type seq seqlen  
## 1 52 TRUE discord 3 3

# Evaluate detections against labels  
evaluation <- evaluate(model, detection$event, dataset$event)  
print(evaluation$confMatrix)

## event   
## detection TRUE FALSE  
## TRUE 0 1   
## FALSE 1 99

# Plot discords vs. ground truth  
har\_plot(model, dataset$serie, detection, dataset$event)



# Plot residual magnitude and decision thresholds  
har\_plot(model, attr(detection, "res"), detection, dataset$event, yline = attr(detection, "threshold"))

