NDA: Time Series Analysis
Part 2: problem-oriented approach

Lionel Tabourier

LIP6 – CNRS and Sobonne University

first_name.last_name@lip6.fr

January 2021

Summary of Part 1

About the classic approach

- pre-processing (transformation etc)
- decomposition/modeling
  - model for trend
  - model for seasonal component
  - ARMA model of residuals
- post-processing → model for the whole process

summarized description of a time series

Advantages and drawbacks

- powerful ⇒ understanding, allows prediction, test, . . .
- but heavy procedure, can be difficult to achieve

Now we look at time series data as (quasi) standard data

Summary of Part 1

About the classic approach

- pre-processing (transformation etc)
- decomposition/modeling
  - model for trend
  - model for seasonal component
  - ARMA model of residuals
- post-processing → model for the whole process

summarized description of a time series

Advantages and drawbacks

- powerful ⇒ understanding, allows prediction, test, . . .
- but heavy procedure, can be difficult to achieve

Now we look at time series data as (quasi) standard data

Summary of Part 1

About the classic approach

- pre-processing (transformation etc)
- decomposition/modeling
  - model for trend
  - model for seasonal component
  - ARMA model of residuals
- post-processing → model for the whole process

summarized description of a time series

Advantages and drawbacks

- powerful ⇒ understanding, allows prediction, test, . . .
- but heavy procedure, can be difficult to achieve

Now we look at time series data as (quasi) standard data

A problem-oriented approach

Alternative approach problem-oriented:
a specific question to solve $\Rightarrow$ suited techniques
regression, classification, hypothesis testing, clustering...

here: investigate a TSA problem and its resolution

The problem to solve

Provide resources in a local WiFi network (mall, university, . . .)

- forecast the daily demand on the network
- plan and adapt the service in consequence
- raise an alert of exceptionally high demand

Data available

- data traffic usage per application during 4 months

A problem-oriented approach

Alternative approach problem-oriented:
a specific question to solve $\Rightarrow$ suited techniques
regression, classification, hypothesis testing, clustering...

here: investigate a TSA problem and its resolution

The problem to solve

Provide resources in a local WiFi network (mall, university, . . .)

- forecast the daily demand on the network
- plan and adapt the service in consequence
- raise an alert of exceptionally high demand

Data available

- data traffic usage per application during 4 months

A problem-oriented approach

Alternative approach problem-oriented:
a specific question to solve $\Rightarrow$ suited techniques
regression, classification, hypothesis testing, clustering...

here: investigate a TSA problem and its resolution

The problem to solve

Provide resources in a local WiFi network (mall, university, . . .)

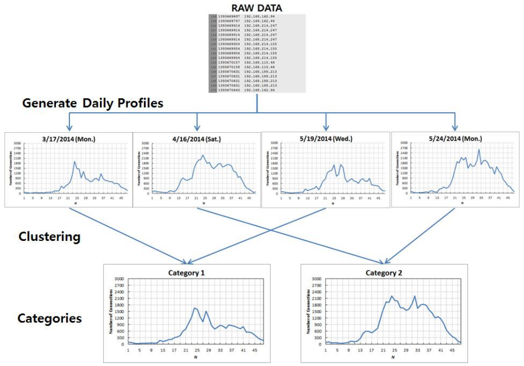
- forecast the daily demand on the network
- plan and adapt the service in consequence
- raise an alert of exceptionally high demand

Data available

- data traffic usage per application during 4 months

Approach chosen

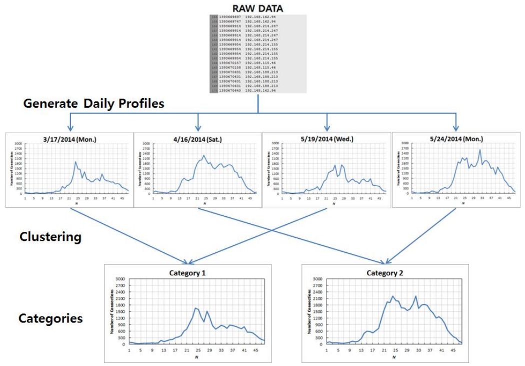
- create classes of usage profiles for each application
  $\rightarrow$ clustering of past data
- analyze streaming data to attribute a class
  $\rightarrow$ classification of incoming data



Not a unique solution $\rightarrow$ need evaluation and comparison

Approach chosen

- create classes of usage profiles for each application
  → clustering of past data
- analyze streaming data to attribute a class
  → classification of incoming data



Not a unique solution → need evaluation and comparison

Step 1: clustering

Agglomerative hierarchical clustering

cf. course on Clustering

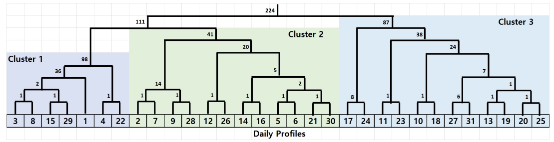
UPGMA method *(Unweighted Pair Group Method with Arithmetic Mean)*

- initialization: each profile is one class
- compute the distances $d(i,j)$ between pairs

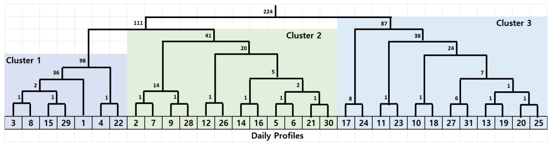
1. find the lowest distance pair, group them in one class
2. recompute distances by arithmetic mean:

$$d(A,B) = \sum_{i \in A} \sum_{j \in B} d(i,j)$$

- iterate 1 and 2 until having one class

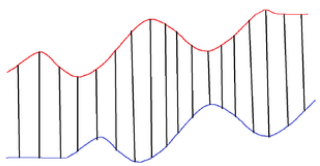


Step 1: clustering



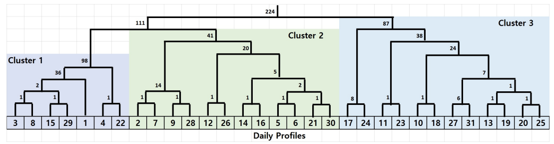
Definition of the distance

Need appropriate notion of distance for time series
euclidean distance inappropriate



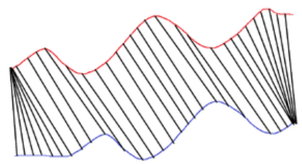
$d(S_1, S_2) = \sum_t |S_1(t) - S_2(t)|$

Step 1: clustering



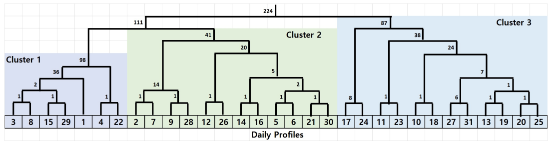
Definition of the distance

Need appropriate notion of distance for time series
dynamic time warping



$d(S_1, S_2) = \sum_t |S_1(t) - S_2(t')|$ s.t. $S_2(t')$ is the *best* match for $S_1(t)$

Step 1: clustering



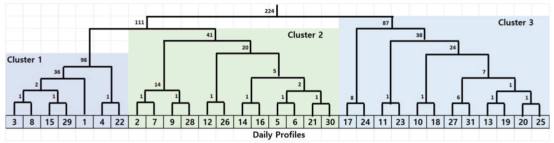
Criterion to decide the clusters

Several options depending on the application:

- predefined number of clusters
- distance threshold Δ between clusters

here: we want homogeneous clusters $\Rightarrow$ threshold parameter Δ to set experimentally

Step 1: clustering



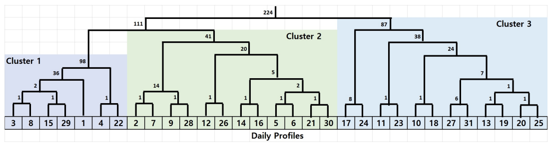
Criterion to decide the clusters

Several options depending on the application:

- predefined number of clusters
- distance threshold Δ between clusters

here: we want homogeneous clusters $\Rightarrow$ threshold parameter Δ to set experimentally

Step 1: clustering

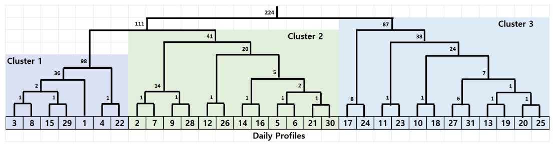


Computational complexity

- for each distance computation using DTW: ts of size n, for each point n comparisons $\Rightarrow O(n^2)$
- UPGMA with S profiles: simple implementations in $O(S^3)$, most efficient in $O(S^2)$

$\rightarrow$ heavy computational costs
manageable because offline processing (and small dataset)

Step 1: clustering



Computational complexity

- for each distance computation using DTW: ts of size n, for each point n comparisons $\Rightarrow O(n^2)$
- UPGMA with S profiles: simple implementations in $O(S^3)$, most efficient in $O(S^2)$

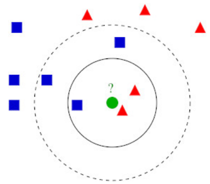
$\rightarrow$ heavy computational costs
manageable because offline processing (and small dataset)

Step 2: classification

cf. course on Classification

K-nearest neighbors: context

- Each data point is located in a space of features
- Each data point has a class *(ex: red triangles, blue squares)*



K-nearest neighbors: principle

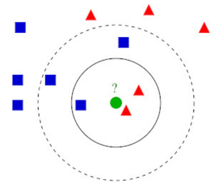
- For a new unlabeled data point *(ex: green circle)*: compute its distance to all labeled data points
- Prediction = dominant class among its k nearest neighbors

Step 2: classification

cf. course on Classification

K-nearest neighbors: context

- Each data point is located in a space of features
- Each data point has a class *(ex: red triangles, blue squares)*



K-nearest neighbors: principle

- For a new unlabeled data point *(ex: green circle)*: compute its distance to all labeled data points
- Prediction = dominant class among its k nearest neighbors

Step 2: classification

Here

simple: DTW distance to all existing data points
$\rightarrow$ what is the class of the closest profile? (k=1)

Computational complexity

- for each distance computation using DTW: ts of size n, for each point n comparisons $\Rightarrow O(n^2)$
- comparisons with S profiles:$O(S)$

computational costs relatively light
but must be done in streaming

Step 2: classification

Here

simple: DTW distance to all existing data points
$\rightarrow$ what is the class of the closest profile? (k=1)

Computational complexity

- for each distance computation using DTW: ts of size n, for each point n comparisons $\Rightarrow O(n^2)$
- comparisons with S profiles:$O(S)$

computational costs relatively light
but must be done in streaming

Step 2: classification

Here

simple: DTW distance to all existing data points
$\rightarrow$ what is the class of the closest profile? (k=1)

Computational complexity

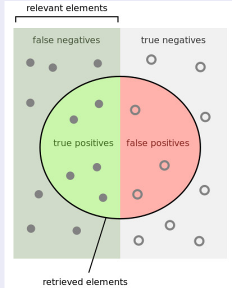
- for each distance computation using DTW:
  ts of size n, for each point n comparisons $\Rightarrow O(n^2)$
- comparisons with S profiles: $O(S)$

computational costs relatively light
but must be done in streaming

Quality evaluation

Classification task

- observe data during part of the day to classify
- $\Rightarrow$ possible misclassifications



$\rightarrow$ here we compute prediction accuracy
proportion of accurate classifications

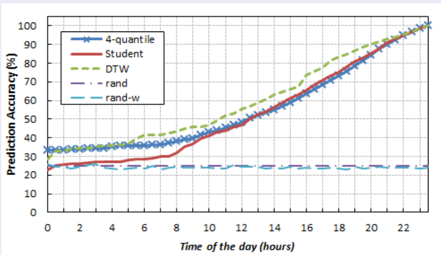
What about the practical efficiency?

Quality evaluation

Classification task

- observe data during part of the day to classify
- $\Rightarrow$ possible misclassifications

prediction accuracy (as a function of learning duration)



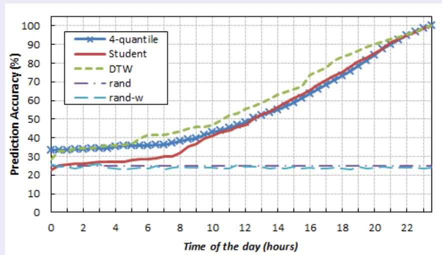
What about the practical efficiency?

Quality evaluation

Classification task

- observe data during part of the day to classify
- $\Rightarrow$ possible misclassifications

prediction accuracy (as a function of learning duration)



What about the practical efficiency?

Discussion

Complete workflow

- acquire data
- offline processing: clustering (classes definition)
- streaming processing: on-the-fly classification
- evaluation of classification

Choices

- choice of resolution
- methods (of clustering, of classification)
- implementation details: (distance, threshold Δ, . . .)

under practical constraints $\rightarrow$ computational demand, . . .

TS analysis is also a craft:
choices guided by knowledge of the methods and data
important to identify why something works or not. . .

Discussion

Complete workflow

- acquire data
- offline processing: clustering (classes definition)
- streaming processing: on-the-fly classification
- evaluation of classification

Choices

- choice of resolution
- methods (of clustering, of classification)
- implementation details: (distance, threshold Δ, . . .)

under practical constraints $\rightarrow$ computational demand, . . .

TS analysis is also a craft:
choices guided by knowledge of the methods and data
important to identify why something works or not. . .

8/8

Discussion

Complete workflow

- acquire data
- offline processing: clustering (classes definition)
- streaming processing: on-the-fly classification
- evaluation of classification

Choices

- choice of resolution
- methods (of clustering, of classification)
- implementation details: (distance, threshold Δ, . . .)

under practical constraints $\rightarrow$ computational demand, . . .

TS analysis is also a craft:
choices guided by knowledge of the methods and data
important to identify why something works or not. . .

8/8