Visualizing Qualitative Patterns in Multivariate Time Series

Luka Stopar
Supervisor: prof. dr. Dunja Mladenić
Co-supervisor: prof. dr. Primož Škraba
Outline

• Motivation
• Approach
• Hypotheses
• Scientific contributions
• Evaluation
• Conclusions
• Publications
Motivation

• Multivariate time series are abundant in scientific and engineering applications
• Common visualizations like common axis plots and parallel views are unsatisfactory
  – Clutter when visualizing large datasets,
  – Fail to highlight the interactions between the variables
  – Difficult to interpret even for data of medium dimensionality
• Users must zoom into interesting intervals, isolate informative variables and remove clutter manually
Approach

• An approach for the abstraction and visualization of multivariate time series
• Data is abstracted as states and transitions
  – Allowing for the visualization of large datasets
• The abstraction is hierarchical
  – Allowing the users to observe the data on several levels of detail (i.e. scales)
Hypotheses

**H1:** Combining techniques from *machine learning, data mining* and statistics into a **tool for data abstraction and visualization** can highlight the **long-term behavior** of a dataset and expose recurrent patterns while mitigating cluttering effects typical when visualizing large datasets.

**H2:** Long-term and recurrent **patterns can emerge on different spatial granularities** (i.e. scales). Traversing the scales interactively provides an effective way to identify scales where such patterns emerge.

**H3:** The **visual abstractions** that summarize the structure and dynamics of the dataset **can be mapped back to the data** and interpreted as domain-specific concepts by interacting with **quantitative data-driven tools**.
Scientific Contributions

**C1:** A methodology for multivariate time series abstraction, visualization, multi-scale exploration and interpretation.

**C2:** A novel visualization approach which allows the user to explore the structure and long-term behavior of the dataset.

**C3:** A fully interactive, web-based visualization tool called StreamStory, which integrates our methodology and is publicly available.
Methodology
C1: Methodology
C1: Methodology – Static Time Series Embedding

- Input: a time series
  \[(t_0, x_0), (t_1, x_1), \ldots, (t_i, x_i), \ldots, \text{ where } x_i \in \mathbb{R}^d\]

- Output: a point cloud
  \[y_1, y_2, \ldots, y_n, \text{ where } y_i \in \mathbb{R}^m\]

- Time is not explicitly represented
- The representation is augmented with historical and temporal information
C1: Methodology – Static Time Series Embedding

- We define a three-component embedding function
  
  \[ f(t_i, x_i) = (g(x_i), \tau(t_i), h(x_1, x_2, \ldots, x_{i-1})) \]

  - \( g \): the **value embedding function** adds information about the current measurement
  - \( \tau \): the **time embedding function** adds information about the current timestamp
  - \( h \): the **history embedding function** adds information about the past trajectory of the time series
C1: Methodology – Constructing the States

- Next, we abstract the measurements into states
  - A state represents a typical configuration of the temporal signal
  - Summarizes the structure around it’s centroid
- We partition the space into Voronoi cells

- Output: a matrix of centroids
  \[ C \in \mathbb{R}^{m \times k} \]
- Each point is assigned to the state \( j \) with the closest centroid
  \[ j = \arg \min_{j \in \mathbb{N}_k} d(y_i, c_j) \]
C1: Methodology – Modelling Transitions

• Temporal dynamics re-introduced as transitions between the states
  – They model the trajectory of the time series through the ambient space

• We model transitions as a continuous-time Markov chain
  – The process generating the transitions is memoryless

• Assumptions:
  – The Markov chain is non-explosive
  – The Markov chain is recurrent
C1: Methodology – Modelling Transitions

- Advantages of Markov chains
  - Compact representation
    \[ Q \in \mathbb{R}^{m \times m} \]
  - We can compute several long-term properties with linear algebra operations, e.g. the stationary distribution:
    \[ \pi Q = 0 \]

- Output: the matrix of transition rates \( Q \)
Constructing a hierarchical partition of the space

We aggregate states by merging adjacent partitions
We compute the transition rate between two aggregated states A and B as:

\[ q_{AB} = \frac{\sum_{i \in A} \pi_i \sum_{j \in B} q_{ij}}{\sum_{i \in A} \pi_i} \]
We support two algorithms to compute the hierarchy

a) **Mean-linkage agglomerative clustering**
   - A distance based approach
   - Iteratively merges the pair of states with the nearest centroids

b) **Iterative min-cut based splitting**
   - A transition based approach
   - Assumes that each state has a Markov chain of its own
   - Starts with a single state
   - Iteratively splits a state by removing the transitions in minimal cut
Visual Representation
C2: Visual Representation

- Overview
- Visual elements
- State layout
- Hierarchical Relationships & Colour coding
- Automatic State Labelling
C2: Visual Representation - Overview

- Time series drawn as a static graph with auxiliary views
  - States are nodes in the graph
  - Transitions are edges
- Hierarchical structure is shown at a single scale in the center
  - The scale can be switched interactively
C2: Visual Representation – Visual Elements

- States summarize regions of the ambient space
- They are visualized as circles with properties encoded in the following attributes
  - **Radius**: the area of the state is linearly proportional to the time that the process spent in the associated region
  - **Position**: reflects the states’ position in the ambient space relative to the other states
  - **Colour**: encodes the hierarchical relationships between the states
  - **Label**: highlights spatial and temporal properties that are typical for the state
  - **Border**: a blue border highlights the currently selected state
C2: Visual Representation – Visual Elements

- Transitions summarize trajectory of the time series
- They are drawn as edges with the following visual properties:
  - **Label**: the label encodes the probability of the associated transition
  - **Colour**: reduces clutter by highlighting dominant transitions
  - **Shape**: further reduces clutter by blurring the low probability transitions with dashed lines
C2: Visual Representation – Visual Elements

- Icicle plot visualizes the historical sequence of states on a linear time axis
- Interactive and linked to the graph-based representation by colour
- Can be used to analyse known historical phenomena
C2: Visual Representation – Visual Elements

- The time histograms visualize when a state typically occurs
  - e.g. in which month of the year
- We visualize four temporal granularities
  - Daily
  - Weekly
  - Monthly
  - Yearly
C2: Visual Representation – State Layout

- State layout reflects the positions of states relative to each other in the ambient space
- Lowest-scale states projected via MDS
- Position of the coarser states computed as a weighted average of their children

\[
p(s) = \frac{\sum_{i=1}^{k} p(s_i) \pi(s_i)}{\sum_{i=1}^{k} \pi(s_i)}
\]

- We follow up with a cross-scale repulsive scheme which ensures that the states do not overlap
• We use colour to encode the states’ position in the hierarchy
• Hue encodes the proximity of the states on the same scale
• Saturation encodes the states’ scale (i.e. the scale where the state first appears)
C2: Visual Representation – Automatic State Labelling

• To help interpretation we automatically label the states
• Two types of labels
  – Spatial labels highlight a characteristic attribute
  – Temporal labels highlight temporal characteristics
Visualization Tool
C3: Visualization Tool

- Demonstration
Evaluation
Evaluation – by Demonstration

• We use four datasets to demonstrate:
  – How to find and interpret the long-term behavior of the data
  – How to map elements of the abstract representation back to domain concepts
  – How traversing the scales can reveal structure in the data
Evaluation – Comparison to TimeCurves

- We compared the representation of two datasets: Weather, Traffic

- Large datasets produce clutter
- We cannot map elements of the visualization to domain concepts
- Removes information by abstracting into states
Evaluation – Feedback from Domain Experts

• Gathered feedback from domain experts in two stages of development
• Early feedback showed that a representation with only states and transitions is hard to interpret
• Adding automatic labels and the icicle plot greatly improved interpretability
• Machine learning tools like decision trees were not useful for domain experts
Conclusion

• Novel methodology for the abstraction of multivariate time series
• Novel visualization approach
• Demonstrated its utility on four real-world datasets
• Confirmed hypotheses through experiments
Publications
Our Previous Work

• We developed a search engine extension, SearchPoint, that groups and reorders documents by topic

• Extended SearchPoint to news corpora adding ranking by sentiment and geographical location

• We contributed to the open source analytics library QMiner

(1) Jaguar USA - Official Site
https://www.jaguarusa.com
Jaguar would like to use cookies to store information on your computer, to improve our website. One of the cookies we use is essential for parts of the site to operate and has already been set.

(2) Jaguar - Wikipedia
The jaguar (Panthera onca) is a wild cat species and the only extant member of the genus Panthera native to the Americas. The jaguar's present range extends from Southwestern United States and Mexico in North America, across much of Central America, ...
Publications

Journal Articles


Conference Papers


Thank You