Online Clustering of High-Dimensional Trajectories under Concept Drift

2011-09-07, ECMLPKDD 2011 Athens, Greece

Georg Krempl[1,2] Zaigham Siddiqui[2] Myra Spiliopoulou[2]

[1] University of Graz
georg.krempl@uni-graz.at

[2] University of Magdeburg
{myra,siddiqui,krempl}
@iti.cs.uni-magdeburg.de

Outline

- Problem Description
  - Motivation and Objectives
  - Modeling Trajectories as Gaussian Mixtures
  - Trajectory Clustering with Expectation Maximization (offline)
- TRACER Algorithm (online)
  - Overview
  - Initialisation
  - Update, Clustering and Prediction
- Experiments
  - Settings
  - Results
- Conclusion

Outline

- Problem Description
  - Motivation and Objectives
  - Modeling Trajectories as Gaussian Mixtures
  - Trajectory Clustering with Expectation Maximization (offline)
- TRACER Algorithm (online)
  - Overview
  - Initialisation
  - Update, Clustering and Prediction
- Experiments
  - Settings
  - Results
- Conclusion

- **CRM Application**
  - Customers are shopping online
  - Money is spent on different product groups in a basket
  - Multiple visits per customer
  - Behaviour changing over time (recession, new product)
  - Can we cluster customers ?
    Can we predict values in the next basket ?

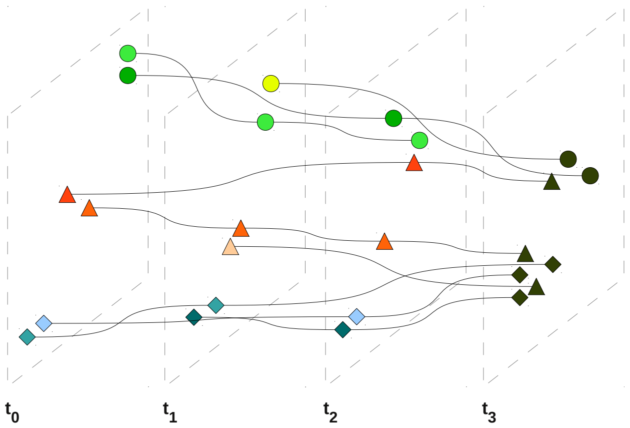
- **CRM Application**
  - Customers are shopping online
  - Money is spent on different product groups in a basket
  - Multiple visits per customer
  - Behaviour changing over time (recession, new product)
  - Can we cluster customers ?
    Can we predict values in the next basket ?

- **Trajectory Clustering Problem**
  - Customers: Population of *individuals*
  - Each visit: *Measurement*,
    Money spent in all product groups: *Measurement vector*
  - Customer history: *Trajectory*
  - Subpopulations of customers: *Clusters*

  - Multiple measurements per individual
  - Measurements are not taken at equi-distant times
  - Distribution of measurements is subject to drift

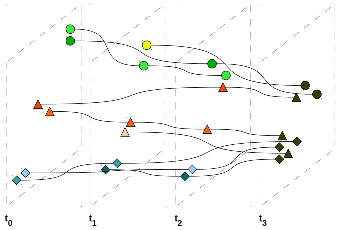
Clustering Trajectories under Drift: Objective

- Cluster individuals
- Track clusters over time
- Predict/Extrapolate cluster movements



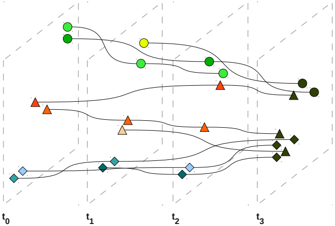
Clustering Trajectories under Drift: Objective

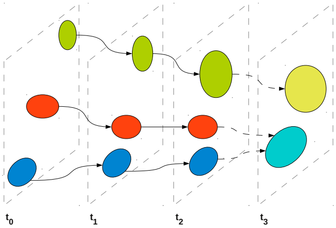
- Cluster individuals
- Track clusters over time
- Predict/Extrapolate cluster movements



Clustering Trajectories under Drift: Objective

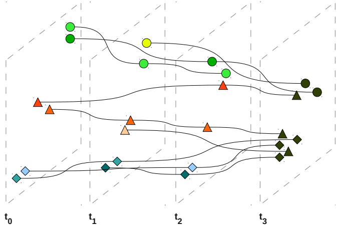
- Cluster individuals
- Track clusters over time
- Predict/Extrapolate cluster movements

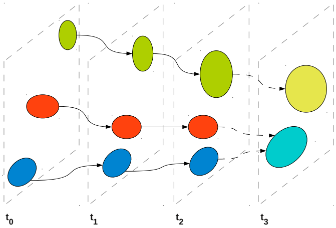


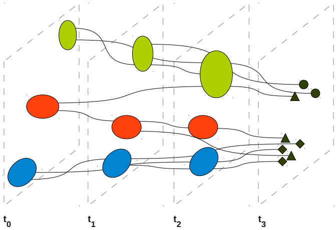


Clustering Trajectories under Drift: Objective

- Cluster individuals
- Track clusters over time
- Predict/Extrapolate cluster movements







Clustering Trajectories under Drift

- Formulation as Gaussian Mixture Model
- $z_i = z_{i1}, z_{i2}, \cdots, z_{in_i}$ are the n_i observations of i-th individual
- K clusters, with
  - mixing proportions α_k
  - distribution parameters θ_k
    mean depends on time via regression coefficients β_k,
    covariance matrix Σ_k is static

  for the k-th cluster.

Clustering Trajectories under Drift

- Formulation as Gaussian Mixture Model
- $z_i = z_{i1}, z_{i2}, \cdots, z_{in_i}$ are the n_i observations of i-th individual
- K clusters, with
  - mixing proportions α_k
  - distribution parameters θ_k
    mean depends on time via regression coefficients β_k,
    covariance matrix Σ_k is static

  for the k-th cluster.
- Likelihood of observing trajectory of individual i:

$$p(z_i; \Theta) = \prod_{l=1}^{n_i} \sum_{k=1}^{K} \alpha_k p(z_{il}; \theta_k) \tag{1}$$

EM Trajectory Clustering

- EM algorithm for general likelihood maximisation problem: Dempster et al., 1977
- Offline EM Trajectory Clustering algorithm:
  - Gaffney and Smyth, 1999
  - Provides an initial clustering
  - Problem:
    Offline algorithm, how to use in a stream?
    How robust against sudden change (non-smooth trajectories)

Outline

TRACER Algorithm

Overview

- Make an initial clustering using EM
- Update clustering:
  - Estimate new position of clusters
  - Assign new individuals to clusters
- Assumptions:
  - Static number of clusters, K
  - Static covariance matrices, Σ_k

TRACER Algorithm

Overview

- Make an initial clustering using EM
- Update clustering:
  - Estimate new position of clusters
  - Assign new individuals to clusters
- Assumptions:
  - Static number of clusters, K
  - Static covariance matrices, Σ_k
- Approach: Kálmán filter (Kálmán, 1959)

Kálmán filter

- State transition: New state x_s

$$x_s = Ax_{s-1} + w_s \tag{2}$$

- State-to-signal: Measurement $z \in \mathcal{R}^D$

$$z_s = Hx_s + v_s \tag{3}$$

Kálmán filter

- State transition: New state x_s

$$x_s = Ax_{s-1} + w_s \tag{2}$$

- State-to-signal: Measurement $z \in \mathcal{R}^D$

$$z_s = Hx_s + v_s \tag{3}$$

- States: True (unobservable) cluster centroids, vector of length $D * (O + 1)$
- Kálmán filter computes at each discrete time step s:
  State estimate for each cluster: $\hat{x}_s$
  Error estimate on cluster state: P_s
- Questions:
  - How to chose $\hat{x}_0$, A, Q, H, R ?
  - How to assign individuals to clusters ?

TRACER Initialisation

Initial State of Each Cluster

State is initialised from β-coefficients obtained via EM

- State vector μ_0 of size $(D * (O + 1)x1)$ at $t = 0$:

$$f(t) = (f_1(0), \cdots, f_D(0))$$

- d-th coordinate estimate:

$$f_d^{(0)}(t) = \beta_{d0} + t\beta_{d1} + \cdots + t^o\beta_{do}$$

- Covariance matrix Σ_0: Identity matrix

TRACER Initialisation

State Transition Matrix A

- Matrix $A = [a_{ij}]$ with

$$a_{i,j} = \begin{cases} \delta_q = \frac{\Delta^q}{q!} & \textit{if} \quad \exists \quad q \in \mathbb{N}_0 : \quad i - j + D * q = 0 \\ 0 & \textit{otherwise} \end{cases}$$

TRACER Initialisation

State Transition Matrix A

- Matrix $A = [a_{ij}]$ with

$$a_{i,j} = \begin{cases} \delta_q = \frac{\Delta^q}{q!} & \text{if} \quad \exists \quad q \in \mathbb{N}_0 : \quad i - j + D * q = 0 \\ 0 & \text{otherwise} \end{cases}$$

- Example for $D = 2$ and $O = 2$:

$$A = \begin{pmatrix} a_0 & 0 & a_1 & 0 & a_2 & 0 \\ 0 & a_0 & 0 & a_1 & 0 & a_2 \\ 0 & 0 & a_0 & 0 & a_1 & 0 \\ 0 & 0 & 0 & a_0 & 0 & a_1 \\ 0 & 0 & 0 & 0 & a_0 & 0 \\ 0 & 0 & 0 & 0 & 0 & a_0 \end{pmatrix}$$

with $a_0 = 1$, $a_1 = \Delta$, $a_2 = \frac{\Delta^2}{2}$

TRACER Initialisation

Process Noise Covariance Matrix Q

- Identity matrix multiplied by process noise factor $\hat{q}$:

$$Q = I * \hat{q}$$

TRACER Initialisation

Process Noise Covariance Matrix Q

- Identity matrix multiplied by process noise factor $\hat{q}$:

$$Q = I * \hat{q}$$

Measurement (or state-to-signal) Matrix H

- Set equal to the identity matrix, $H = I$

TRACER Initialisation

Process Noise Covariance Matrix Q

- Identity matrix multiplied by process noise factor $\hat{q}$:

$$Q = I * \hat{q}$$

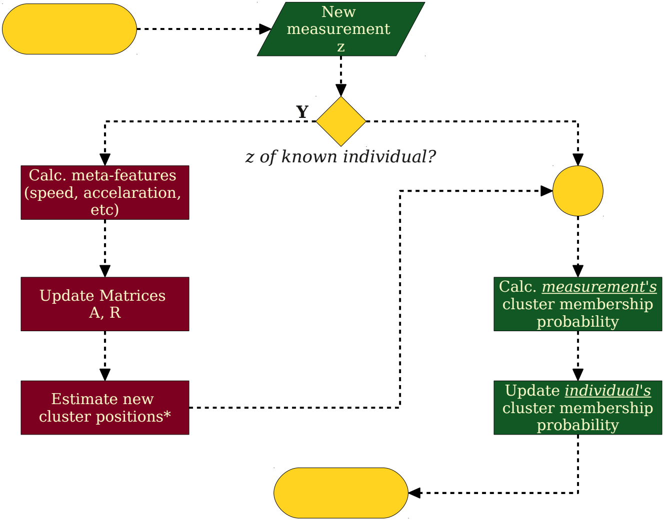
Measurement (or state-to-signal) Matrix H

- Set equal to the identity matrix, $H = I$

Measurement Noise Covariance Matrix R

- Computed as covariance matrix of EM clustering

TRACER Update and Clustering



Outline

Objective

- Similar clustering quality of EM and TRACER?
- Robustness against sudden shift
- Speed and suitability for online processing

Objective

- Similar clustering quality of EM and TRACER?
- Robustness against sudden shift
- Speed and suitability for online processing

Synthetic Data Streams with Drift

- 5 types of synthetic data sets:
  - Different state transition noise (A : high, C low)
  - Different number of dimensions ($A, \cdots, C$: one; D, E: two)
- 10 data sets per type
- 1500 individuals, on average 2 measurements / individual,
  1000 measurements for training,
  1000 for test before shift, 1000 for test after shift

Update Strategies

	Method	Description
EM	EM	Expectation Maximisation (multivariate variant of [Gaffney and Smyth, 1999])
Kalman	K-1	Confidence prop. to squared membership probability
	K-2	Confidence $\in \{0; 1\}$, winner-takes-all
	K-3	Confidence prop. to membership probability
	K-4	As K1, but 10x higher ST noise factor estimate
	K-5	As K1, but 10x smaller ST noise factor estimate
	K-6	As K1, but use of speed and acceleration as meta-features for membership probability estimation p

Measure

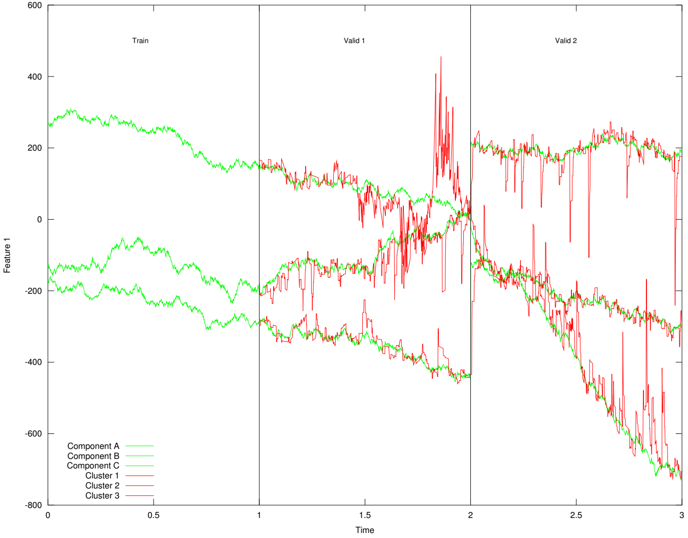
- Cluster Purity:

$$purity = \frac{1}{N} \sum_{j=1}^{K} \max_{i=1}^{K} C_{ij}$$

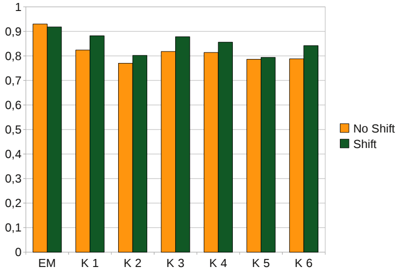
C_{ij} Number of elements in the i-th true and j-th pred. cluster
N Total number of elements

- Wilcoxon signed rank sum test:
  Significance of differences in clustering quality

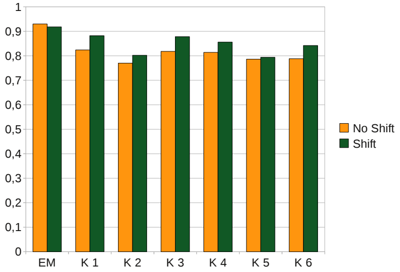
Accuracy of State Estimation over Time

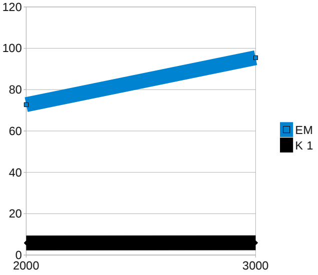


Dependence of Purity : Shift and Speed : Dataset Size



Dependence of Purity : Shift and Speed : Dataset Size





	Purity		Time		
	No Shift	Shift	2000	3000	Method Description
EM	**0.93**	**0.91**	**72.74**	**95.42**	Offline Expectation Maximisation
K 1	**0.82**	**0.88**	**5.84**	**5.92**	Squared membership prob. $c = 1/p^2$
K 2	0.77	0.80	5.54	5.68	Winner-takes-all
K 3	0.82	0.88	5.82	6.10	Membership prob. as weights, $c = 1/p$
K 4	0.81	0.86	5.76	5.92	As K1, but ST noise estimated 10x higher
K 5	0.77	0.79	5.72	6.12	As K1, but ST noise estimated 10x lower
K 6	0.79	0.84	5.84	5.92	As K1, but speed and acceleration as features for p estimation

Outline

- Problem Description
  - Motivation and Objectives
  - Modelling Trajectories as Gaussian Mixtures
  - Trajectory Clustering with Expectation Maximisation (offline)
- TRACER Algorithm (online)
  - Overview
  - Initialisation
  - Update, Clustering and Prediction
- Experiments
  - Settings
  - Results
- Conclusion

Conclusion

Summary

- Trajectory clustering: e.g. customers with purchase histories
- TRACER Algorithm: *Online* trajectory clustering and tracking
- Compared to offline EM: Competitive quality, much faster, robust against shift
  Of particular interest when clustering streams

Conclusion

Summary

- Trajectory clustering: e.g. customers with purchase histories
- TRACER Algorithm: *Online* trajectory clustering and tracking
- Compared to offline EM: Competitive quality, much faster, robust against shift
  Of particular interest when clustering streams

Outlook

- Real-world application and experiments
- Dynamic covariance matrices (changing R over time), dynamic number of clusters (changing K over time)
- Smoothness of prediction
- Consider case where individuals change their cluster membership over time

Conclusion

Questions ?

Thank you!

Sourcecode available online:
https://bitbucket.org/geos/tracer-trajectory-tracking/overview

Bibliography

A. P. Dempster, N. M. Laird, and D. Rubin.
Maximum likelihood from incomplete data via the EM algorithm.
Journal of the Royal Statistical Society, Series B, 39:1–38, 1977.

S. Gaffney and P. Smyth.
Trajectory clustering with mixtures of regression models.
In *KDD '99*, pages 63–72. ACM, 1999.

Y. Han, J. de Veth, and L. Boves.
Trajectory clustering for automatic speech recognition, 2005.

X. Jiang and N. Petkov, editors.
Computer Analysis of Images and Patterns, 13th International Conference, CAIP 2009, Münster, Germany, September 2-4, 2009. Proceedings, volume 5702 of *Lecture Notes in Computer Science*. Springer, 2009.

R. E. Kalman.
A New Approach to Linear Filtering and Prediction Problems.
Trans. of the ASME – Journal of Basic Engineering, 82(Series D):35–45, 1960.

G. Xiong, C. Feng, and L. Ji.
Dynamical gaussian mixture model for tracking elliptical living objects.
Pattern Recognition Letters, 27:838–842, May 2006.