Destination Prediction by Trajectory Distribution-Based Models

PGMO Days

Philippe Besse, Brendan Guillouet, Jean-Michel Loubes, François Royer
14 November 2017
Introduction

1 - A new structure for the data

2 - Model’s Exploitation

Discussion
Introduction
Objectives

Location of Taxi in San Francisco.

Geo-referenced dataset, $p \in \mathcal{P}$, t.q. $p = (\text{lon}, \text{lat}, t, \text{Id}_{\text{traj}})$. 
Objectives

Location of Taxi in San Francisco.

Geo-referenced dataset, $p \in \mathcal{P}$, t.q. $p = (lon, lat, t, Id_{traj})$.

- Extract information hidden in the data.
Objectives

Geo-referenced dataset, $p \in \mathcal{P}$, t.q. $p = (\text{lon}, \text{lat}, t, Id_{\text{traj}})$.

- **Extract** information hidden in the data.
- **Use** this information to predict and analyse future trajectories.

Location of Taxi in San Francisco.
Objectives

Geo-referenced dataset, $p \in \mathcal{P}$, t.q. $p = (lon, lat, t, ld_{\text{traj}})$.

- Extract information hidden in the data.
- Use this information to predict and analyse future trajectories.

⇒ Create a Learning method.

Location of Taxi in San Francisco.
Build a **MODEL**:

- **Adaptable**: which use only location, date and time, independent from additional (road network, etc.).
- **Interpretable**: easy to explain, avoid «black box» model.
- **Instantaneous**: allows real time prediction, no re-learning.
Constraints

Build a **MODEL**:

- **Adaptable**:
  - *which use only location, date and time,*
  - *independent from additional (road network, etc.)*
Build a **MODEL**:

- **Adaptable**:
  - *which use only location, date and time,*
  - *independent from additional (road network, etc.)*

- **Interpretable**:
  - *easy to explain,*
  - *avoid « black box » model.*
Constraints

Build a **model**:

- **Adaptable**:
  - *which use only location, date and time,*
  - *independent from additional (road network, etc.)*

- **Interpretable**:
  - *easy to explain,*
  - *avoid « black box » model.*

- **Instantaneous**:
  - *allows real time prediction,*
  - *no re-learning.*
• From Caltrain station, San Francisco, USA [Piorkowski et al., 2009].
• 4,127 Trajectories.
• 1 observation/60 seconds.

• From Sao Bento station, Porto, Portugal [Kaggle, 2015].
• 19,423 Trajectories.
• 1 observation/15 seconds.

San Francisco Dataset.

Porto Dataset.
1 - A NEW STRUCTURE FOR THE DATA
Definition (Trajectory)

\[ T^i : \left( (p^i_1, t^i_1), \ldots, (p^i_{n^i}, t^i_{n^i}) \right), \text{with } p^i_k \in \mathbb{R}^2, \forall k \in [1 \ldots n^i]. \]

where \( n^i \) is the number of locations which compose the trajectory \( T^i \).
\[ C(\mathcal{P}) = \left\{ \mathcal{P}_1, \ldots, \mathcal{P}_{k_1}, \ldots, \mathcal{P}_1, \ldots, \mathcal{P}_{k_K} \right\} \]

\[ C(\mathcal{T}) = \{ \mathcal{T}^1, \ldots, \mathcal{T}^K \} \]
\( \mathcal{P} \) Set of points

\[
\mathcal{C}(\mathcal{P}) = \left\{ \mathcal{P}_1^1, \ldots, \mathcal{P}_{K_1}^1, \ldots, \mathcal{P}_1^K, \ldots, \mathcal{P}_{K_1}^K \right\}
\]

Clusters of points

\( \{\Theta_{\text{ML}}^1, \ldots, \Theta_{\text{ML}}^K\} \) Set of Gaussian mixture model

Preprocessing

\( \mathcal{T} \) Set of trajectories

\[
\mathcal{C}(\mathcal{T}) = \{\mathcal{T}^1, \ldots, \mathcal{T}^K\}
\]

Clusters of trajectories

Trajectory Clustering

\[ C(\mathcal{P}) = \{\mathcal{P}^1, \ldots, \mathcal{P}^K\} \]

Gaussian mixture model

Learning
Set of points $\mathcal{P}$

Preprocessing

Set of trajectories $\mathcal{T}$

Trajectory Clustering

$\mathcal{C}(\mathcal{P}) = \{\mathcal{P}_1^1, \ldots, \mathcal{P}_{R_1}^1, \ldots, \mathcal{P}_1^K, \ldots, \mathcal{P}_{R_K}^K\}$

Clusters of trajectories

$\mathcal{C}(\mathcal{T}) = \{\mathcal{T}_1^1, \ldots, \mathcal{T}_K^K\}$

Clusters of trajectories

Set of Gaussian mixture model

Gaussian mixture model

Set of points

$\{\Theta_{ML}^1, \ldots, \Theta_{ML}^K\}$

Set of Gaussian mixture model
The diagram illustrates the process of trajectory clustering using Gaussian mixture models.

1. **Set of points** \( \mathcal{P} \)
   - Preprocessing
   - Clusters of points \( C(\mathcal{P}) = \{ \mathcal{P}_1^1, \ldots, \mathcal{P}_1^K, \ldots, \mathcal{P}_K^1, \ldots, \mathcal{P}_K^K \} \)
   - Set of Gaussian mixture model \( \{ \Theta_1^1_{ML}, \ldots, \Theta_K^K_{ML} \} \)

2. **Set of trajectories** \( \mathcal{T} \)
   - Trajectory Clustering
   - Clusters of trajectories \( C(\mathcal{T}) = \{ \mathcal{T}_1^1, \ldots, \mathcal{T}_K^K \} \)

The diagram represents the learning process where points are preprocessed into clusters, and trajectories are clustered into Gaussian mixture models.
1 - A new structure for the data

Distance between trajectories
A NEW DISTANCE BETWEEN TRAJECTORIES

Definition (Distance from a point to a trajectory)

\[ D_{pt}(p_{i_1}^1, T^2) = \min_{i_2 \in [0, \ldots, n_2-1]} D_{ps}(p_{i_1}^1, s_{i_2}^2). \]

Definition (Segment-Path Distance (SPD))

\[ D_{SPD}(T^1, T^2) = \frac{1}{n_1} \sum_{i_1=1}^{n_1} D_{pt}(p_{i_1}^1, T^2). \]
Symmetrized Segment-Path Distance (SSPD)

A new distance: SSPD [Besse et al., 2016a].

**Definition (Symmetrized-SPD (SSPD))**

\[
D_{SSPD}(T^1, T^2) = \frac{D_{SPD}(T^1, T^2) + D_{SPD}(T^2, T^1)}{2}
\]

The SSPD distance is a « symmetric » [Deza and Deza, 2009].
A new distance: SSPD [Besse et al., 2016a].

**Definition (Symmetrized-SPD (SSPD))**

\[
D_{SSPD}(T^1, T^2) = \frac{D_{SPD}(T^1, T^2) + D_{SPD}(T^2, T^1)}{2}.
\]

The **SSPD** distance is a « symmetric » [Deza and Deza, 2009].

**Advantages:**

- compare trajectory as a whole,
A new distance: SSPD [Besse et al., 2016a].

**Definition (Symmetrized-SPD (SSPD))**

\[ D_{SSPD}(T^1, T^2) = \frac{D_{SPD}(T^1, T^2) + D_{SPD}(T^2, T^1)}{2}. \]

The **SSPD** distance is a « symmetric » [Deza and Deza, 2009].

**ADVANTAGES :**

- compare trajectory as a whole,
- take into account their shape and their geographical distance,
A new distance: SSPD [Besse et al., 2016a].

**Definition (Symmetrized-SPD (SSPD))**

\[
D_{\text{SSPD}}(T^1, T^2) = \frac{D_{\text{SPD}}(T^1, T^2) + D_{\text{SPD}}(T^2, T^1)}{2}.
\]

The SSPD distance is a « symmetric » [Deza and Deza, 2009].

**Advantages:**

- compare trajectory as a whole,
- take into account their shape and their geographical distance,
- independent from time indexing.
1 - A NEW STRUCTURE FOR THE DATA

TRAJECTORY CLUSTERING
Clustering methods

Choice of the method:

- **Existing methods:**
  - Spectral Clustering and K-means:
    - Can’t be used on this distance.
  - k-medoid and DBSCAN:
    - No optimised for « symmetric ».
  - The Affinity propagation:
    - Does not allow to control the number of clusters.

Selected method: 

- *La Hierarchical clustering*.
  - Takes ‘symmetric’ into account,
  - Produce different levels of classification.
  - Produce a new partition of $K$ clusters of trajectories: $C(T) = \{T_1, \ldots, T_K\}$. 
Clustering methods

Choice of the method:

- Existing methods:
  - Spectral Clustering and K-means: can't be used on this distance.
  - k-medoid and DBSCAN: no optimised for « symmetric ».
  - The Affinity propagation: does not allow to control the number of clusters.

Selected method: La Hierarchical clustering.
- Takes 'symmetric' into account.
- Produce different levels of classification.

Produce a new partition of $K$ clusters of trajectories: $C(T) = \{T_1, \ldots, T_K\}$. 
Clustering methods

Choice of the method:

• Existing methods:
  • Spectral Clustering and K-means:
    • Can’t be used on this distance.

• k-medoid and DBSCAN:
  • No optimised for “symmetric”.

• The Affinity propagation:
  • Does not allow to control the number of clusters.

• Selected method ➔ Hierarchical clustering.
  • Takes ‘symmetric’ into account.
  • Produces different levels of classification.

Produce a new partition of \( K \) clusters of trajectories:
\[ C(T) = \{T_1, \ldots, T_K\} \]
**Clustering methods**

**Choice of the method:**

- Existing methods:
  - **Spectral Clustering** and **K-means**:
    - *Can’t be used on this distance.*
  - **k-medoid** and **DBSCAN**:
    - *no optimised for « symmetric ».*

Selected method:
- **Hierarchical clustering**.
  - Takes ‘symmetric’ into account,
  - Produce different levels of classification.

Produce a new partition of $K$ clusters of trajectories:

$$C(T) = \{T_1, \ldots, T_K\}.$$
Clustering methods

Choice of the method:

- Existing methods:
  - Spectral Clustering and K-means:
    - Can’t be used on this distance.
  - k-medoid and DBSCAN:
    - no optimised for « symmetric ».
  - The Affinity propagation:
    - does not allow to control the number of clusters.

Selected method: Hierarchical clustering.

Takes 'symmetric' into account,

Produce different levels of classification.

Produce a new partition of $K$ clusters of trajectories: $C(T) = \{T_1, \ldots, T_K\}$. 
Clustering methods

Choice of the method:

- Existing methods:
  - **Spectral Clustering** and **K-means**:
    - Can’t be used on this distance.
  - **k-medoid** and **DBSCAN**:
    - no optimised for « symmetric ».
  - The **Affinity propagation**:
    - does not allow to control the number of clusters.

- Selected method ⇒ La **Hierarchical clustering**.
  - Takes ’symmetric’ into account,
  - Produce different levels of classification.
Clustering methods

Choice of the method:

- Existing methods:
  - **Spectral Clustering** and **K-means**:
    - Can’t be used on this distance.
  - **k-medoid** and **DBSCAN**:
    - No optimised for « symmetric ».
  - The **Affinity propagation**:
    - Does not allow to control the number of clusters.

- Selected method ⇒ La **Hierarchical clustering**.
  - Takes ‘symmetric’ into account,
  - Produce different levels of classification.

Produce a new partition of $K$ clusters of trajectories:

$$C(T) = \{T^1, \ldots, T^K\}.$$
Trajectory clustering in San Francisco.

\[ \mathcal{T} \Rightarrow C(\mathcal{T}) = \{\mathcal{T}^1, \ldots, \mathcal{T}^K\} \]

Trajectories of taxis.

25 classes trajectory clustering.