Graph-valued models for dimensionality reduction and regression

Anna Calissano[1], **Aasa Feragen**[2], Simone Vantini[1]

[1] MOX - Modelling and Scientific Computing - Politecnico di Milano
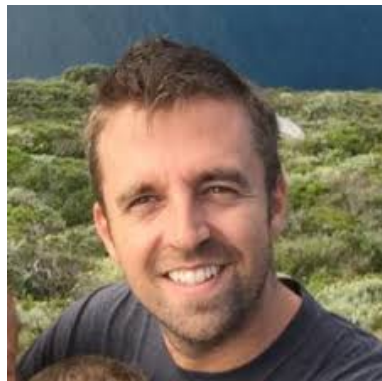[2] Section for Image Analysis & Computer Graphics, DTU Compute
afhar@dtu.dk
`http://www2.compute.dtu.dk/~afhar`

Recent Advances in Statistical Analysis of Imaging Data

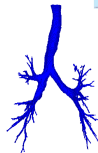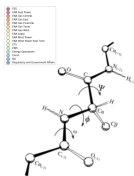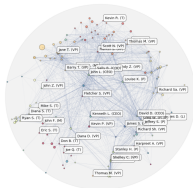Florida State University, 5.12.2020

Danmarks
Tekniske
Universitet

DTU
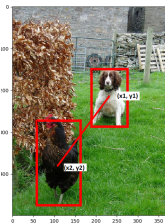
Collaboration with Anna Calissano and Simone Vantini, MOX, Politecnico Milano

Populations of Unlabeled Networks: Graph Space Geometry and Geodesic Principal Components, Preprint, 2019

Graph-valued Regression Models, In Preparation, 2020

Graphs as data objects – they are everywhere!
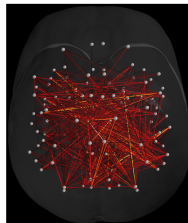
▶ Scene understanding, social networks, chemo/bioinformatics, brain connectivity

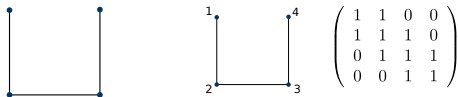# Graphs as data objects – they are everywhere!

▶ Variable nodes, variable edges, attributes on nodes and edges

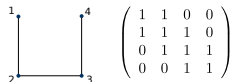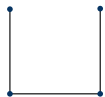# A space of graphs

▶ A (weighted) graph can be represented by its adjacency
matrix $A \in \mathbb{R}^{n \times n} =: \mathcal{A}$

$$\begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{pmatrix}$$

Jain et al, JMLR 2009

A space of graphs

▶ A (weighted) graph can be represented by its adjacency matrix $A \in \mathbb{R}^{n \times n} =: \mathcal{A}$

$$\begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{pmatrix}$$

$$\begin{pmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix}$$

▶ Each graph has multiple adjacency matrix representations

Jain et al, JMLR 2009

A space of graphs

► A (weighted) graph can be represented by its adjacency matrix $A \in \mathbb{R}^{n \times n} =: \mathcal{A}$

$$\begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{pmatrix}$$
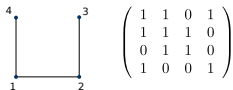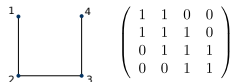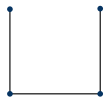
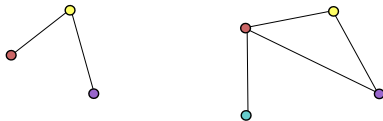$$\begin{pmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix}$$

► Each graph has multiple adjacency matrix representations

► A graph space with unique graph representations: The quotient

$$\mathcal{G} := \mathcal{A}/S_n$$

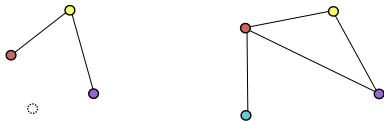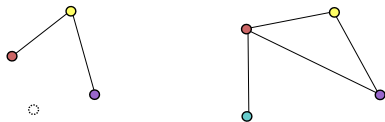with respect to the node permutation group S_n

---

Jain et al, JMLR 2009

A general space of graphs

▶ Easy to accommodate graphs with different numbers of nodes:

A general space of graphs

▶ Easy to accommodate graphs with different numbers of nodes:

A general space of graphs

► Easy to accommodate graphs with different numbers of nodes:

► Easy to extend to vector valued node and edge weights

$$A \in \mathbb{R}^{n \times n} =: \mathcal{A} \quad \rightsquigarrow \quad A \in (\mathbb{R}^d)^{n \times n} =: \mathcal{A}$$

Existing work on statistics in Jain's graph space

- Jain, Obermayer: Structure Spaces. Journal of Machine Learning Research. (2009)

- Jain, Obermayer: Large Sample Statistics in the Domain of Graphs. SSPR/SPR (2010)

- Jain, Obermayer: Maximum Likelihood for Gaussians on Graphs. GbRPR (2011)

- Jain: Maximum likelihood method for parameter estimation of bell-shaped functions on graphs. Pattern Recognition Letters (2012)

- Calissano, Feragen, Vantini: Analysis of Populations of Networks: Structure Spaces and the Computation of Summary Statistics. ICSA (2019)

- Guo, Srivastava, Sarkar: A Quotient Space Formulation for Statistical Analysis of Graphical Data. arXiv preprint arXiv:1909.12907 (2019).

- Kolaczyk, Lin, Rosenberg, Xu, Walters: Averages of Unlabeled Networks: Geometric Characterization and Asymptotic Behavior. arXiv preprint arXiv:1709.02793 (2019).

Graph Space Geometry

Graph Space is a Quotient Space
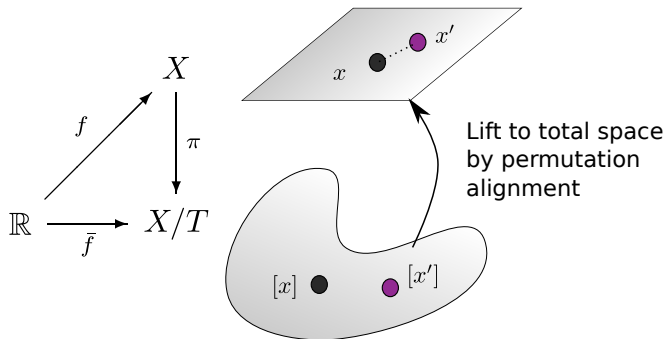
- Graph space X/T inherits a metric from the Euclidean metric on $X = \mathbb{R}^{n \times n}$
- Graph space is a geodesic metric space – any two points joined by a shortest path
- The total space X can be thought of as a "tangent space" where a "log map" at any base point graph $[x]$ is equivalent to alignment to its fixed representative x
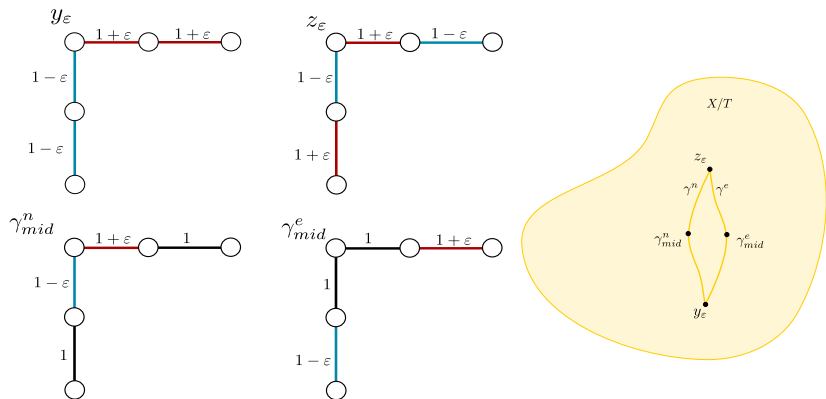
The geometry of graph space

Theorem
Graph-space geodesics are not necessarily unique.
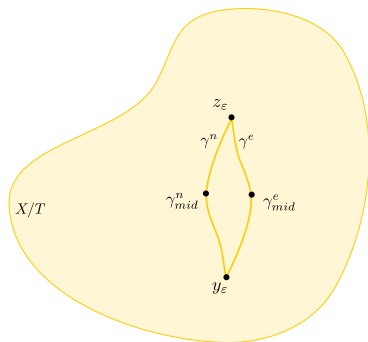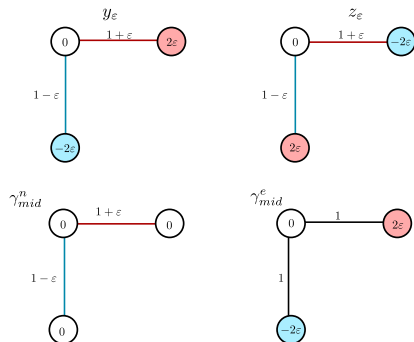
Theorem
Graph-space curvature is unbounded from above.

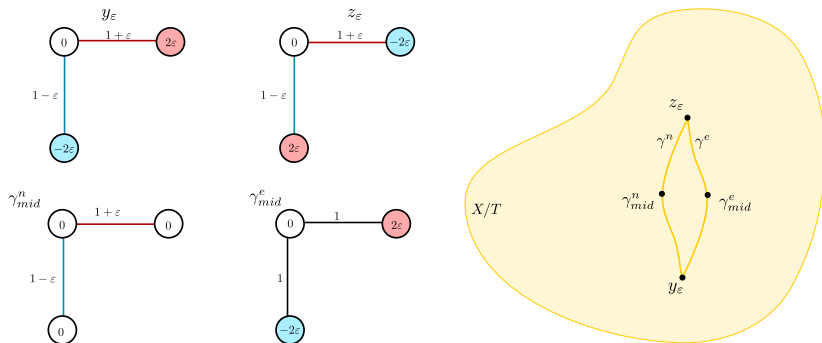# The counterexample is not dependent on having node attributes

Statistics

First statistic: Fréchet mean

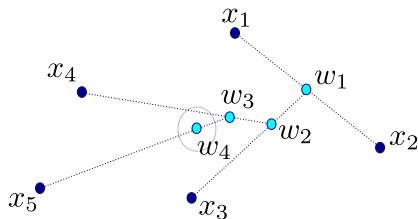$$[m] = \mathrm{argmin}_{[x] \in X/G} \sum_{i=1 \ldots n} d^2_{X/G}([x], [x_i])$$

Theorem

Fréchet means are not generally unique in graph space X/G.

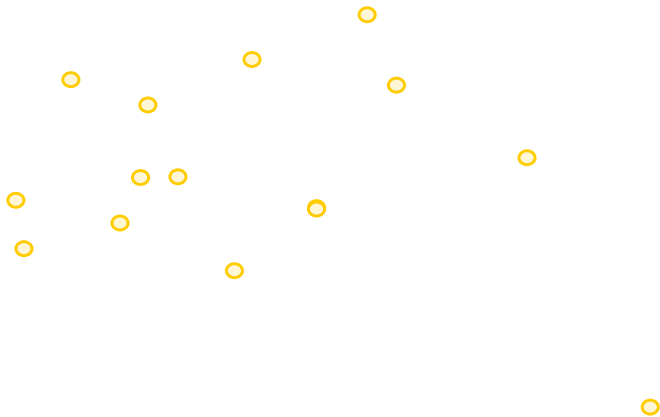# Existing algorithms and heuristics for computing Fréchet means in nonlinear spaces

Iterative weighted midpoints / stochastic gradient descent

Note: Proofs of convergence usually require being able to work in a neighborhood with unique geodesics.
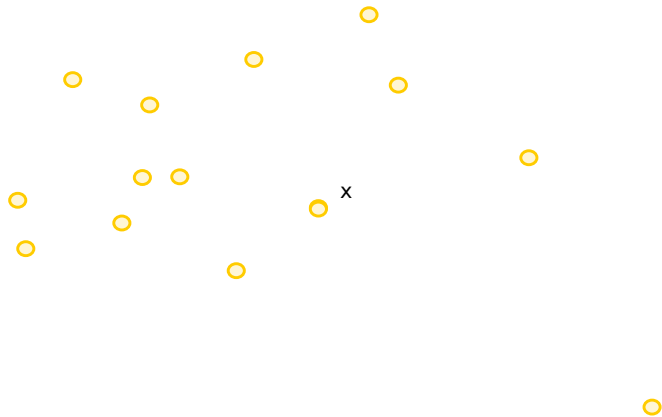
Existing algorithms and heuristics for computing Fréchet means in nonlinear spaces

We choose an analogy with Generalized Procrustes Analysis, and call the general strategy "Align all and compute" (AAC).

Existing algorithms and heuristics for computing Fréchet means in nonlinear spaces

We choose an analogy with Generalized Procrustes Analysis, and call the general strategy "Align all and compute" (AAC).

x

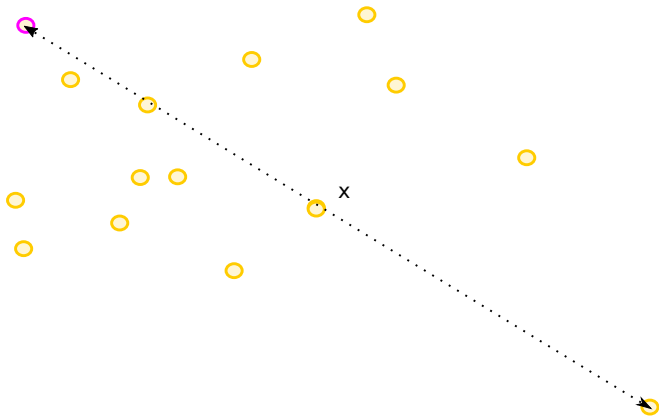# Existing algorithms and heuristics for computing Fréchet means in nonlinear spaces

We choose an analogy with Generalized Procrustes Analysis, and call the general strategy "Align all and compute" (AAC).

Existing algorithms and heuristics for computing Fréchet means in nonlinear spaces

We choose an analogy with Generalized Procrustes Analysis, and call the general strategy "Align all and compute" (AAC).
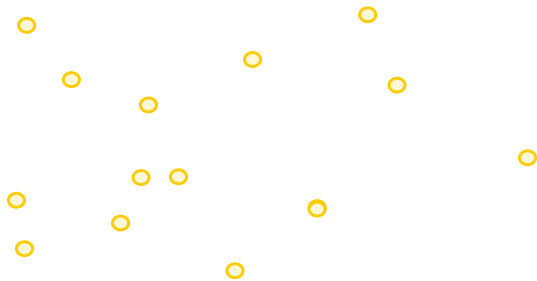
Existing algorithms and heuristics for computing Fréchet means in nonlinear spaces

We choose an analogy with Generalized Procrustes Analysis, and call the general strategy "Align all and compute" (AAC).
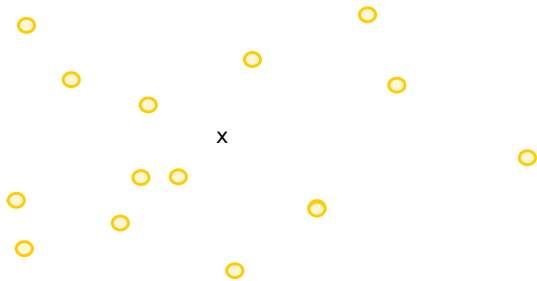
Existing algorithms and heuristics for computing Fréchet means in nonlinear spaces

We choose an analogy with Generalized Procrustes Analysis, and call the general strategy "Align all and compute" (AAC).
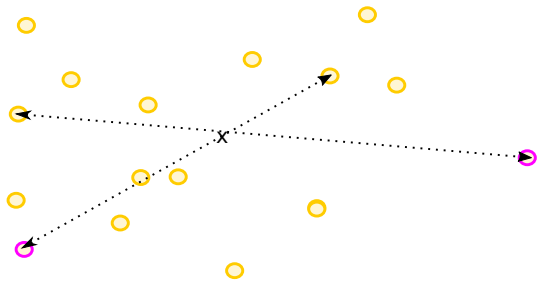
Existing algorithms and heuristics for computing Fréchet means in nonlinear spaces

We choose an analogy with Generalized Procrustes Analysis, and call the general strategy "Align all and compute" (AAC).
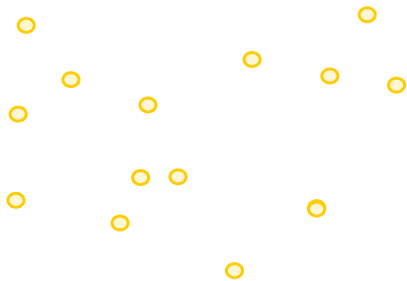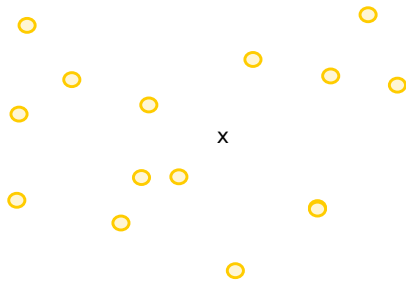
Existing algorithms and heuristics for computing Fréchet means in nonlinear spaces

We choose an analogy with Generalized Procrustes Analysis, and call the general strategy "Align all and compute" (AAC).

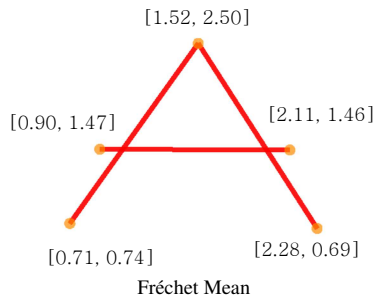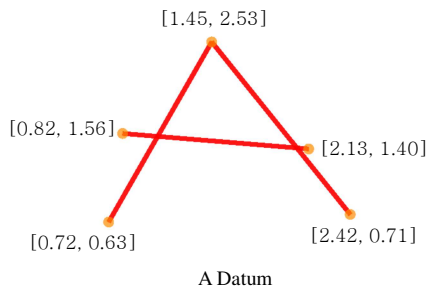Guarantee convergence to local minimum in finite time for generic dataset.

Example: Handwritten Letters

[1.45, 2.53]

[0.82, 1.56]

[2.13, 1.40]

[0.72, 0.63]

[2.42, 0.71]

A Datum

[1.52, 2.50]

[0.90, 1.47]

[2.11, 1.46]

[0.71, 0.74]

[2.28, 0.69]

Fréchet Mean

Example: Mobility networks Lombardia region
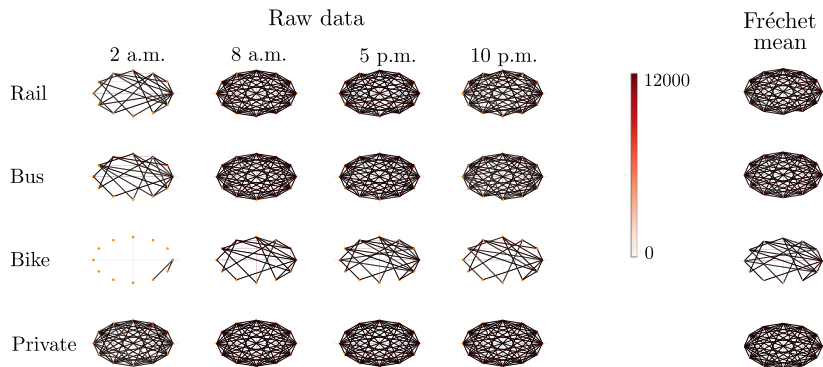
Raw data

Fréchet mean

	2 a.m.	8 a.m.	5 p.m.	10 p.m.
Rail				
Bus				
Bike				
Private				

12000

0

Higher order statistics: "Tangent space" approach

- "Tangent space" approach: Align all points with a representative of the mean and perform statistics in the total space ($\Leftrightarrow$ tangent space statistics in manifolds).

- Guo, Srivastava, Sarkar (2019): Tangent space PCA
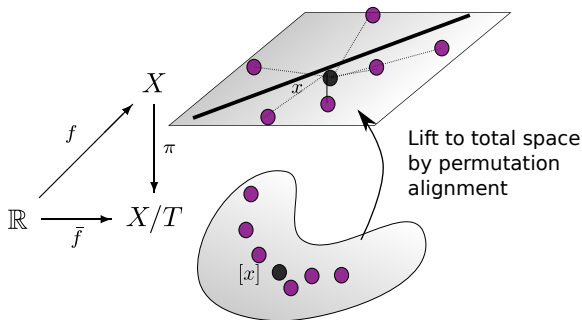

Lift to total space by permutation alignment
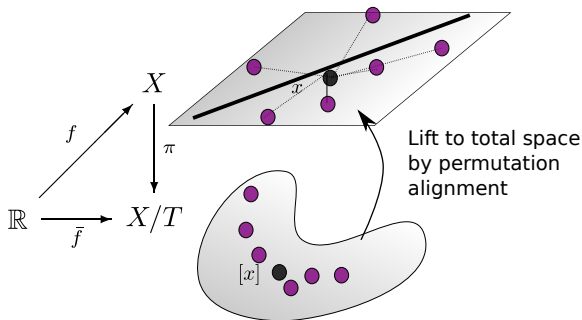
Higher order statistics: "Tangent space" approach

- ▶ "Tangent space" approach: Align all points with a representative of the mean and perform statistics in the total space ($\Leftrightarrow$ tangent space statistics in manifolds).
- ▶ Guo, Srivastava, Sarkar (2019): Tangent space PCA
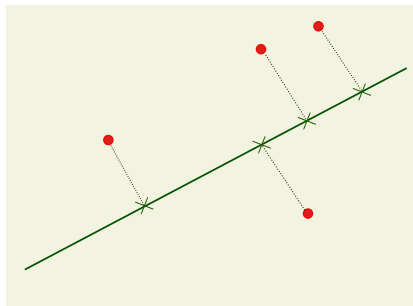- ▶ Risk distorted residuals away from the base point

Intrinsic dimensionality reduction: Generalized Geodesic PCA

Generalized Geodesic Principal Components

- **Data:** $[x_1], \ldots, [x_k] \in X/T$
- **Model:** Optimize over generalized geodesics $\delta = \pi \circ \gamma$ for geodesic γ in space of adjacency matrices X
- **Task:** Find generalized geodesic $\delta_1 \in \Gamma(X/T)$ such that

$$\delta_1 = \mathrm{argmin}_{\delta \in \Gamma(X/T)} \sum_{i=1}^{k} (d_{X/T}^2([x_i], \delta))$$

AAC for Generalized Geodesic Principal components

Algorithm 1: PCA via AAC

Result: Principal components 1-d represented as subspaces of $\mathbb{R}^{n \times n}$

initialize by aligning all data graphs to a random adjacency matrix;

while *While not converged* **do**

  perform PCA in $\mathbb{R}^{n \times n}$;

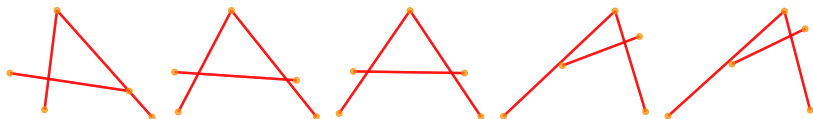 choose representatives of all data graphs in optimal position with the first PC in $\mathbb{R}^{n \times n}$.
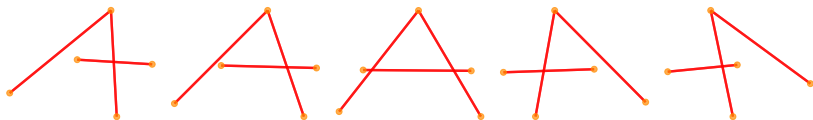
end

Theorem

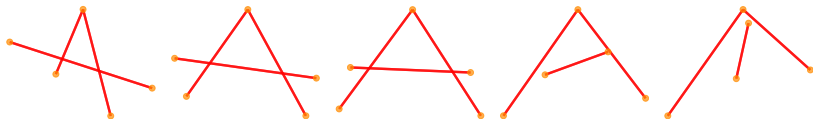The AAC algorithm converges to a local minimum in finite time.

1st GGPC (27⊳3%)
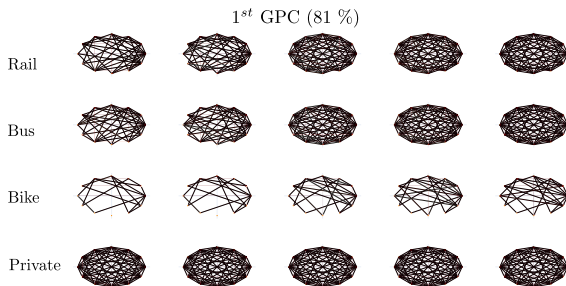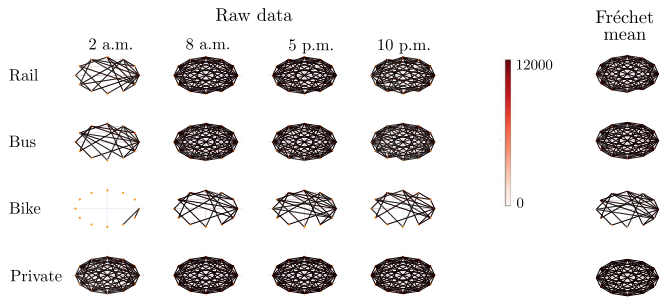
2nd GGPC (18⊳3%)

3rd GGPC (15⊳6%)

Mobility networks Lombardia region

Preliminary example: Baboons

▶ Networks represent grooming interaction/association between baboons; they can have different roles (alpha/beta) over time ⇒ naturally unlabeled networks

▶ Each network summarizes data collected within 30 days before and 90 days after each knockout.

▶ Knockout = when a given alpha or beta male leaves the group.

Image by Noneotuho, https://commons.wikimedia.org/wiki/File:
Macaca_fuscata,_grooming,_Iwatayama,_20090201.jpg

Preliminary example: Baboons

Figure: GPC 1

Image by Noneotuho, https://commons.wikimedia.org/wiki/File:
Macaca_fuscata,_grooming,_Iwatayama,_20090201.jpg

Preliminary example: Baboons
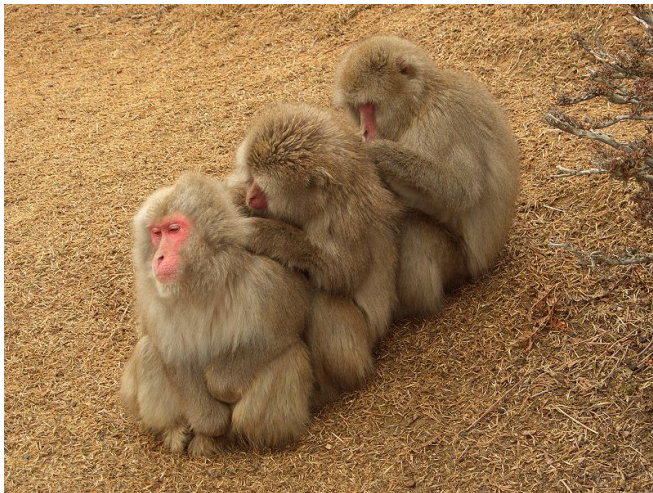

Figure: GPC 2

Image by Noneotuho, https://commons.wikimedia.org/wiki/File:
Macaca_fuscata,_grooming,_Iwatayama,_20090201.jpg

Graph-valued regression models

Regression models taking values in Graph Space

- ▶ **Data:** $(s_1, [x_1]), \ldots, (s_k, [x_k])$, where $(s_i, [x_i]) \in \mathbb{R}^p \times X/T$
- ▶ **Model:** Optimize over $\bar{f} = \pi \circ f$ for linear regression models f on the space X of adjacency matrices
- ▶ **Task:** Describe the relationship $f \colon \mathbb{R}^p \to X/T$ minimizing

$$\sum_{i=1}^{k} d_{X/T}^2([x_i], f(s_i))$$
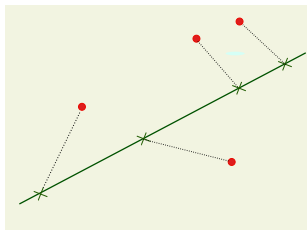
- ▶ **Note:** Residuals measured between predicted and true output.

AAC for Linear Regression models taking values in Graph Space

Algorithm 2: Linear Regression Models in Graph Space via AAC

Result: Predictions $f(s) \in \mathbb{R}^{n \times n}$ for any $s \in \mathbb{R}^d$

initialize by aligning all data graphs to a random adjacency matrix;

while *While not converged* **do**

   estimate linear regression model in $\mathbb{R}^{n \times n}$;

   choose representatives of all data graphs in optimal position with current estimate of the regression line in $\mathbb{R}^{n \times n}$.
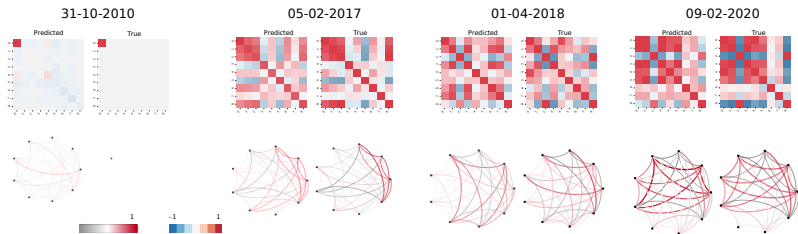
end

Theorem

The AAC algorithm converges to a local minimum in finite time.
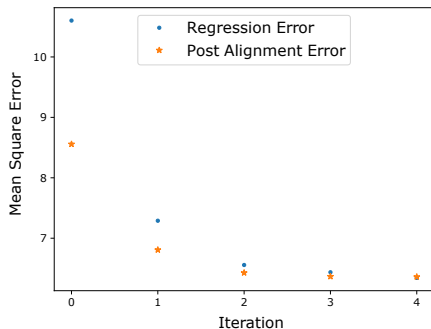
Example: Cryptocurrency correlation networks

Data: Correlation networks for price in USD of Bitcoin, Dash, Digibyte, Dogecoin, Litecoin, Vertcoin, Stellar, Monero, Verge from July 18th 2010 (first record of bitcoin) - until April 3rd 2020, for 20-day time windows.
Task: Predict network from time

Example: Cryptocurrency correlation networks

Convergence:

Example:
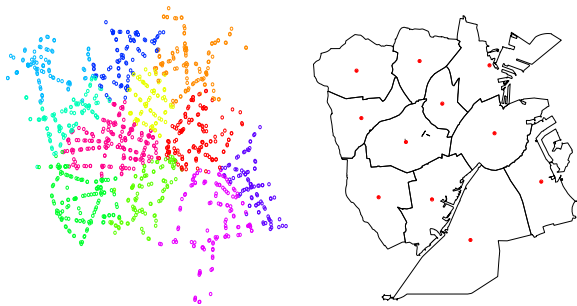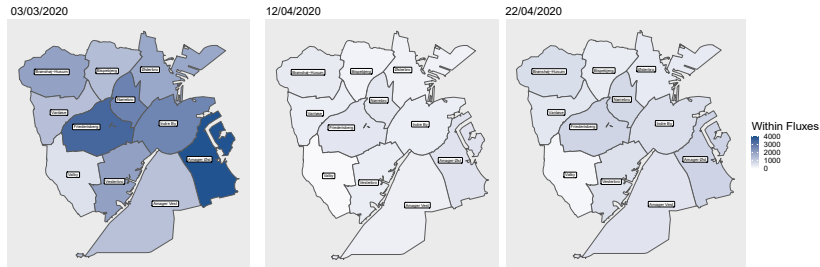# Public Transport and Covid-19 in Copenhagen, Denmark

Figure: Bus stops in the different areas of Copenhagen and Frederiksberg.
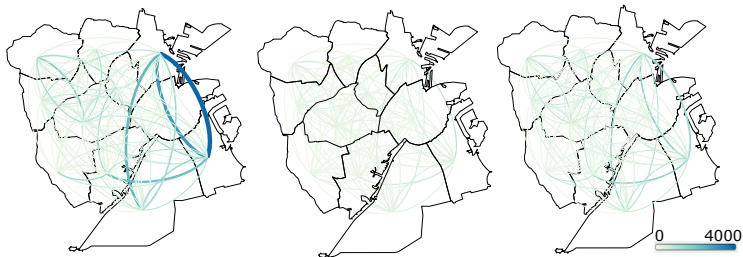
Example:
Public Transport and Covid-19 in Copenhagen, Denmark

Figure: Prediction of the within area fluxes (i.e. the nodes attributes) of three days randomly sampled from the three periods: 03/03/2020, 12/04/2020, and 22/04/2020
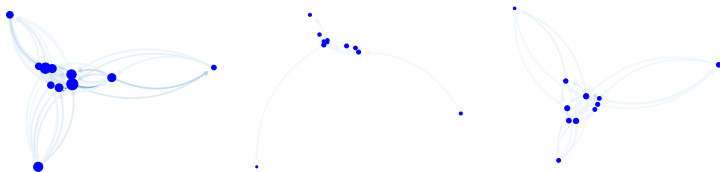
Example:
Public Transport and Covid-19 in Copenhagen, Denmark

Figure: Prediction of the network (i.e. the nodes attributes) of three days randomly sampled from the three periods: 03/03/2020, 12/04/2020, and 22/04/2020

Example:
Public Transport and Covid-19 in Copenhagen, Denmark

Figure: Prediction of the network of three days randomly sampled from the three periods: 03/03/2020, 12/04/2020, and 22/04/2020. The dimension of the node is proportional to the within area flux. The position of the nodes is computed using Spectral Layout of the Networkx python package.

Example:
Public Transport and Covid-19 in Copenhagen, Denmark
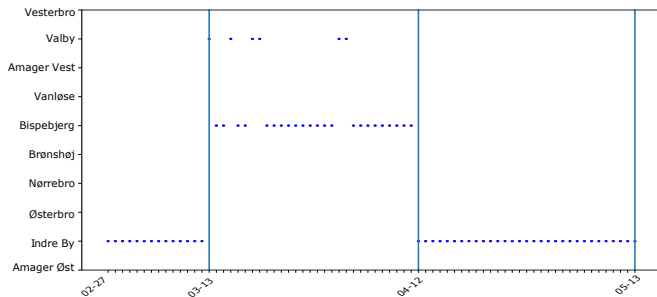
Figure: Most popular optimal matching with the node Indre By as a function time. Vertical Lines describe transitions between different phases of the lockdown.

Discussion

Discussion

Advantages:

- ▶ Total space is $(\mathbb{R}^d)^{n \times n}$, easing generalization of Euclidean methods
- ▶ The AAC approach gives intrinsic statistics while retaining the computational advantages of Euclidean statistics

Limitations:

- ▶ Distances are generally NP-complete due to graph matching problem $\rightsquigarrow$ approximations
- ▶ Graph space geometry is highly non-Euclidean (and it is not a manifold); this is likely to affect statistics in ways we do not yet understand.
- ▶ Due to the interpolation-based nature of statistics, estimators tend to lie in the top dimensional stratum ($=$ be a complete graph), whereas data often lives on its boundary. Might we lean on ideas from logistic regression and/or image segmentation to overcome this limitation?