Dept. of Computer Science, University of Copenhagen

Three cases of uncertainty

Visualization and Processing of Anisotropy
in Imaging, Geometry and Astronomy

Schloss Dagstuhl 02.11.2018

Aasa Feragen, aasa@di.ku.dk

In collaboration with A. Mallasto and S. Hauberg

Case I: Uncertainty in Tractography

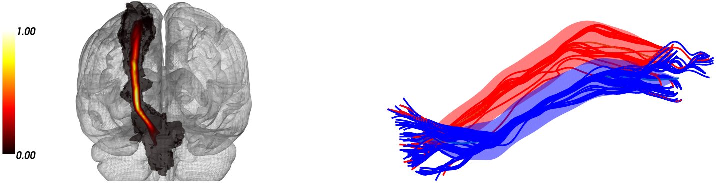
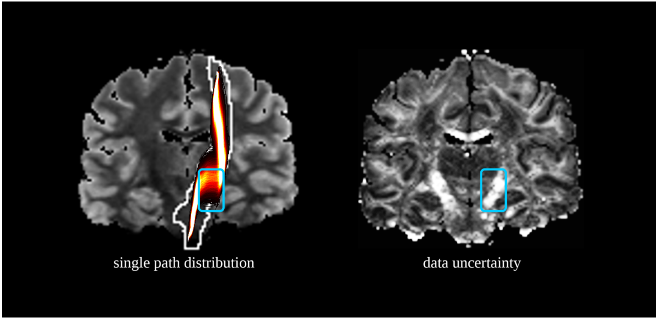


Figure: **Left:** Voxel-wise heatmap, Kasenburg et al NeuroImage'16. **Right:** Envelope of Gaussian Process tract (cluster) estimate, Mallasto and Feragen NIPS'17.

- **What** uncertainty do we quantify? Application dependent.
- **How** do we visualize/communicate uncertainty?
- **How** does it act in population analysis?

What does uncertainty measure?

Example: Tractography[1]



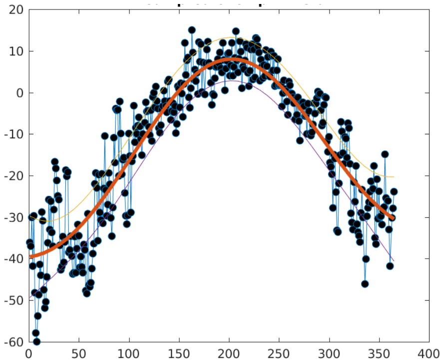
- Uncertainty due to noise
- Signal loss due to anatomical effects
- Poor model fit
- Otherwise lack of support in data

[1] Hauberg et al, MICCAI'15

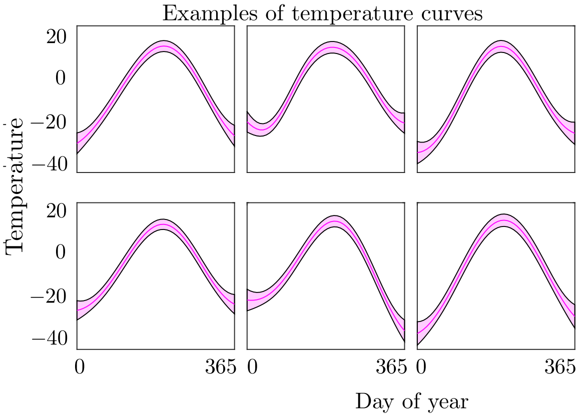
What does uncertainty measure?

Example: Daily temperature in Siberian city

- Data (temperature curve) is estimated from measurements with high level of variation
- Temperature curve should reflect natural variation – should be represented as *distributions over curves*, not as curves



What does uncertainty measure?

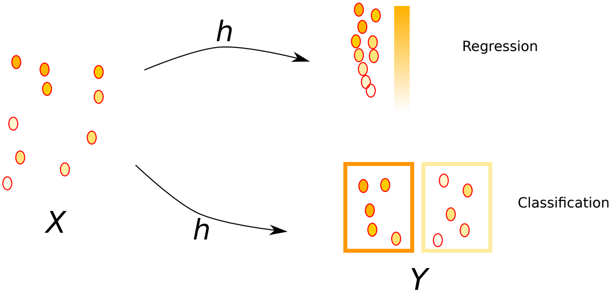


- Uncertainty can also measure of natural variation in the data

Case II: The role of uncertainty when data is estimated from data

The standard machine learning scenario:

- Data resides in some space X; aim to learn a predictive function $h\colon X \to Y$

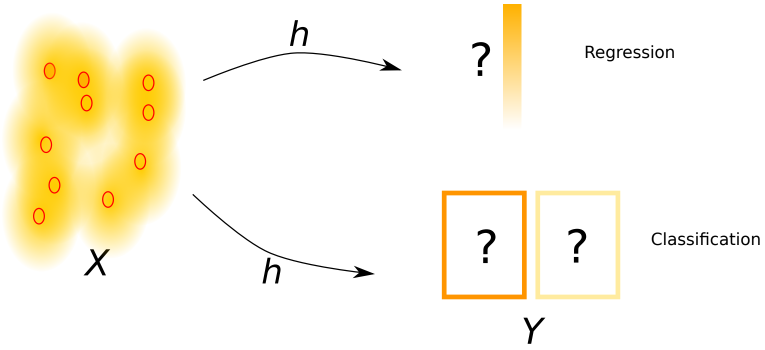


- What when the observation is a distribution, not a point?

Case II: The role of uncertainty when data is estimated from data

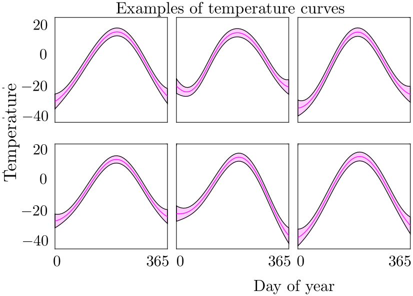
The standard machine learning scenario:

- Data resides in some space X; aim to learn a predictive function $h\colon X \to Y$

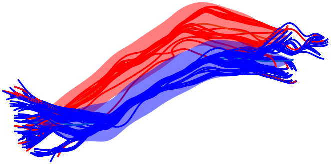


- What when the observation is a distribution, not a point?

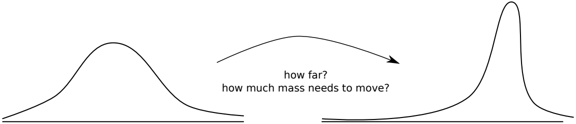
To the learning problem...



- So we want to do statistics on uncertain curves
- First problem: What does it mean for two uncertain curves to be *similar/dissimilar*?



Wasserstein (earth mover's) distance[1]



$$W_2(\mu, \nu) = \sqrt{\inf_{\gamma \in \Gamma(\mu,\nu)} \int_{M \times M} d^2(x, y) d\gamma(x, y)}$$

- Wasserstein distances between GPs are approximated arbitrarily well by Wasserstein distances between approximating GDs
- Wasserstein means for sets of GPs are approximated arbitrarily well by Wasserstein means for sets of approximating GDs
- Wasserstein for GDs is computationally nice (distances: analytical, means: iterative)

[1] Mallasto and Feragen, NIPS'17

Learning with Wasserstein distances between GPs[2]

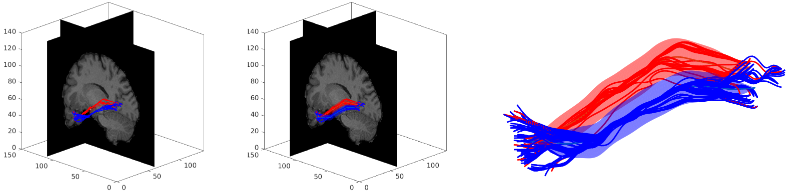


Figure: Clustering, hypothesis testing between populations of GPs

With a distance *d* between GPs, we can define

- mean GPs (cluster means)
- hierarchical clustering
- permutation tests for equal means
- etc

[2] Mallasto and Feragen, NIPS'17

Learning with Wasserstein distances between GPs[2]

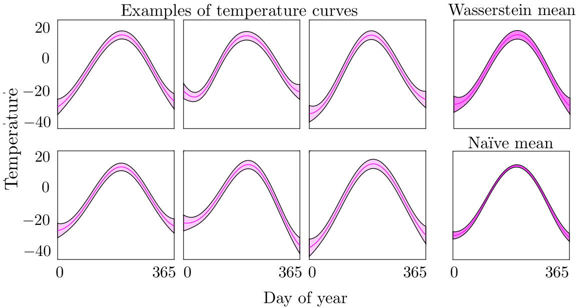


Figure: Wasserstein means utilize the point-wise uncertainty.

NB! Uncertainty is not a bad thing! Can contain highly relevant information. Not clear that we always want to minimize uncertainty in our estimates.

[2] Mallasto and Feragen, NIPS'17

Learning with Wasserstein distances between GPs[2]

- With weighted means for GPs, we can perform kernel regression.

Figure: Kernel regression on GPs: Predicted temperature curve distributions over 30 Russian weather stations in the period 1940-2009

[2] Mallasto and Feragen, NIPS'17

Case III: The effect of geometric constraints

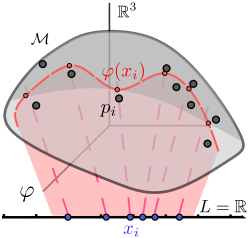
- GPs satisfy geometric constraints: k has to be a covariance function (positive definiteness).
- This led to the question: Can geometric constraints help learn structure?
- $\rightsquigarrow$ Uncertainty quantification in Riemannian submanifold learning

Latent variable models

- The Gaussian Process Latent Variable Model (GPLVM) is a submanifold learning algorithm, where the learned embedding

$$f\colon U \subset \mathbb{R}^n \to \mathbb{R}^N$$

from chart onto submanifold is a GP: $f \sim \mathcal{GP}(\mu, k)$
- For data that lies on a manifold, we thought: Waste of resources to learn what you already know (the manifold)
- We extended GPLVM to take values strictly on the manifold[3]. Let's see what happened...



[3]Mallasto, Hauberg, Feragen, In review'18

Performance of the Wrapped Gaussian Process Latent Variable Model

Riemannian	Femur	Diatoms	Diffusion tensors	Crypto-tensors
GPLVMProj	$(9.22 \pm 0.55) \times 10^{-2}$	$(2.48 \pm 0.25) \times 10^{-2}$	0.582 ± 0.025	21.91 ± 2.26
WGPLVM	$\mathbf{(9.20 \pm 0.53) \times 10^{-2}}$	$\mathbf{(2.39 \pm 0.15) \times 10^{-2}}$	$\mathbf{0.391 \pm 0.035}$	$\mathbf{3.04 \pm 0.26}$

Euclidean	Femur	Diatoms	Diffusion tensors	Crypto-tensors
GPLVM	$(9.21 \pm 0.55) \times 10^{-2}$	$(2.48 \pm 0.25) \times 10^{-2}$	$\mathbf{(6.03 \pm 0.34) \times 10^{-2}}$	$(7.36 \pm 5.27) \times 10^{5}$
GPLVMProj	$(9.21 \pm 0.55) \times 10^{-2}$	$(2.48 \pm 0.25 \times 10^{-2}$	$\mathbf{(6.03 \pm 0.34) \times 10^{-2}}$	$(5.49 \pm 3.17) \times 10^{5}$
WGPLVM	$\mathbf{(9.19 \pm 0.53) \times 10^{-2}}$	$\mathbf{(2.39 \pm 0.15) \times 10^{-2}}$	$(7.54 \pm 0.36) \times 10^{-2}$	$(8.69 \pm 7.12) \times 10^{5}$

Figure: Mean reconstruction errors (top = intrinsic distance, bottom = Euclidean distance)

- **Conclusion:** These datasets were not particularly big, but even in the Euclidean models, the mean function learned the manifold anyway!

Performance of the Wrapped Gaussian Process Latent Variable Model

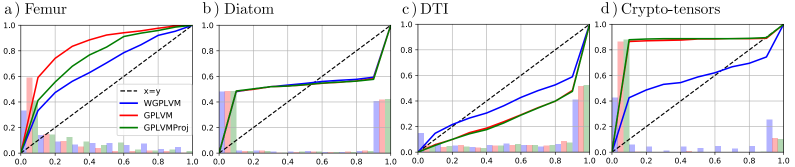


Figure: Comparison of quantiles in the data and the estimated model. The bars represent the frequency of occurances, where the fraction of samples, given by the x-value, lie closer to the mean prediction than a test point. The continuous curves represent the cumulative distributions. Whenever the cumulative distribution lies above $x = y$, we are overestimating the corresponding quantile.

- **Conclusion:** In the Euclidean models, the covariance function does *not* learn the manifold on its own

Performance of the Wrapped Gaussian Process Latent Variable Model

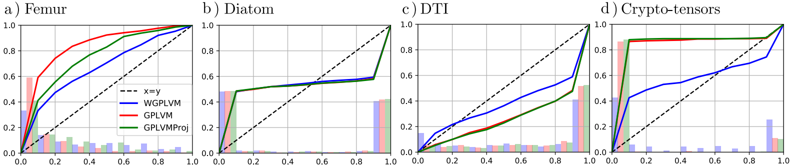


Figure: Comparison of quantiles in the data and the estimated model. The bars represent the frequency of occurances, where the fraction of samples, given by the x-value, lie closer to the mean prediction than a test point. The continuous curves represent the cumulative distributions. Whenever the cumulative distribution lies above $x = y$, we are overestimating the corresponding quantile.

- **Conclusion:** In the Euclidean models, the covariance function does *not* learn the manifold on its own
- **Explanation:** The uncertainty covers up a poor model fit of the parameterized covariance, which assigns positive

Main points

- Uncertainty covers a number of aspects:
  - Sample size
  - Uncertainty due to noise
  - Signal loss due to anatomical effects
  - Poor model fit
  - Otherwise lack of support in data

  Different aspects need different modelling and communication.

Main points

- Uncertainty covers a number of aspects:
  - Sample size
  - Uncertainty due to noise
  - Signal loss due to anatomical effects
  - Poor model fit
  - Otherwise lack of support in data

  Different aspects need different modelling and communication.
- **Incorporating uncertainty in population analysis** – first steps; many to be taken

Main points

- Uncertainty covers a number of aspects:
  - Sample size
  - Uncertainty due to noise
  - Signal loss due to anatomical effects
  - Poor model fit
  - Otherwise lack of support in data

  Different aspects need different modelling and communication.
- **Incorporating uncertainty in population analysis** – first steps; many to be taken
- **Utilizing geometric constraints in submanifold learning:** Uncertainty quantification benefits from prior knowledge