Modern Gaussian Processes: Scalable Inference and Novel Applications

(Part III) Applications, Challenges & Opportunities

Edwin V. Bonilla and Maurizio Filippone

CSIRO’s Data61, Sydney, Australia and EURECOM, Sophia Antipolis, France

July 14th, 2019
1. Multi-task Learning

2. The Gaussian Process Latent Variable Model (GPLVM)

3. Bayesian Optimisation

4. Deep Gaussian Processes

5. Other Interesting GP/DGP-based Models
Multi-task Learning
Data Fusion and Multi-task Learning (1)

- Sharing information across tasks/problems/modalities
- Very little data on test task
- Can model dependencies \textit{a priori}
- Correlated GP prior over latent functions
Multi-task GP (Bonilla et al, NeurIPS, 2008)

- \( \text{Cov}(f_\ell(x), f_m(x')) = K_{\ell m}^{f} \kappa(x, x') \)
- \( K \) can be estimated from data
- Kronecker-product covariances
  - ‘Efficient’ computation
- Robot inverse dynamics (Chai et al, NeurIPS, 2009)
Multi-task GP (Bonilla et al, NeurIPS, 2008)

- $\text{Cov}(f_\ell(x), f_m(x')) = K_{\ell m}^f \kappa(x, x')$
- $K$ can be estimated from data
- Kronecker-product covariances
  - ‘Efficient’ computation
- Robot inverse dynamics (Chai et al, NeurIPS, 2009)

Generalisations and other settings:

- Convolution formalism (Alvarez and Lawrence, JMLR, 2011)
- GP regression networks (Wilson et al, ICML, 2012)
- Many more ...
The Gaussian Process Latent Variable Model (GPLVM)
The **Gaussian Process Latent Variable Model (GPLVM; Lawrence, NeurIPS, 2004):**

- Probabilistic non-linear dimensionality reduction
- Use independent GPs for each observed dimension
- Estimate latent projections of the data via maximum likelihood
Style-Based Inverse Kinematics: Given a set of constraints, produce the most likely pose

- High dimensional data derived from pose information
  - joint angles, vertical orientation, velocity and accelerations
- GPLVM used to learn low-dimensional trajectories
- GPLVM predictive distribution used in cost function for finding new poses with constraints

Fig. and cool videos at http://grail.cs.washington.edu/projects/styleik/
Bayesian Optimisation
Optimisation of black-box functions:

- Do not know their implementation
- Costly to evaluate
- Use GPs as surrogate models
Optimisation of black-box functions:

- Do not know their implementation
- Costly to evaluate
- Use GPs as surrogate models

Vanilla BO iterates:

1. Get a few samples from true function
Optimisation of black-box functions:

- Do not know their implementation
- Costly to evaluate
- Use GPs as surrogate models

Vanilla BO iterates:

1. Get a few samples from true function
2. Fit a GP to the samples
Optimisation of black-box functions:

- Do not know their implementation
- Costly to evaluate
- Use GPs as surrogate models

Vanilla BO iterates:

1. Get a few samples from true function
2. Fit a GP to the samples
3. Use GP predictive distribution along with acquisition function to suggest new sample locations
Optimisation of black-box functions:

- Do not know their implementation
- Costly to evaluate
- Use GPs as surrogate models

Vanilla BO iterates:

1. Get a few samples from true function
2. Fit a GP to the samples
3. Use GP predictive distribution along with **acquisition function** to suggest new sample locations

What are sensible acquisition functions?
Bayesian Optimisation (2)

A taxonomy of algorithms proposed by D. R. Jones (2001)

- $\mu(x^*_\star), \sigma^2(x^*_\star)$: pred. mean, variance
- $I \stackrel{\text{def}}{=} f(x^*_\star) - f_{\text{best}}$: pred. improvement

Fig. from Boyle (2007)
Bayesian Optimisation (2)

A taxonomy of algorithms proposed by D. R. Jones (2001)

- $\mu(x_\star), \sigma^2(x_\star)$: pred. mean, variance
- $I \overset{\text{def}}{=} f(x_\star) - f_{\text{best}}$: pred. improvement
- **Expected improvement:**

$$EI(x_\star) = \int_0^\infty I p(I) dI$$

- Simple ‘analytical form’
- Exploration-exploitation

Fig. from Boyle (2007)
Bayesian Optimisation (2)

A taxonomy of algorithms proposed by D. R. Jones (2001)

- $\mu(x_\star), \sigma^2(x_\star)$: pred. mean, variance
- $I \overset{\text{def}}{=} f(x_\star) - f_{\text{best}}$: pred. improvement
- Expected improvement:

$$EI(x_\star) = \int_0^\infty I p(I) dI$$

- Simple ‘analytical form’
- Exploration-exploitation

Main idea: Sample $x_\star$ so as to maximize the EI
Many cool applications of BO and probabilistic numerics:

- Optimisation of ML algorithms (Snoek et al, NeurIPS, 2012)
- Multi-task BO (Swersky et al, NeurIPS, 2013)
- Bayesian Quadrature

See http://probabilistic-numerics.org/ and references therein
Deep Gaussian Processes
The Deep Learning Revolution

- Large representational power
- *Big data* learning through stochastic optimisation
- Exploit GPU and distributed computing
- Automatic differentiation
- Mature development of regularization (e.g., dropout)
- Application-specific representations (e.g., convolutional)
Can we exploit what made Deep Learning successful for practical and scalable learning of Gaussian processes?
Deep Gaussian Processes

- Composition of Processes

\[(f \circ g)(x)??\]
• Composition of processes: Deep Gaussian Processes
Learning Deep Gaussian Processes

- Inference requires calculating integrals of this kind:

\[
p(Y|X, \theta) = \int p(Y|F^{(N_h)}, \theta^{(N_h)}) \times \\
p(F^{(N_h)}|F^{(N_h-1)}, \theta^{(N_h-1)}) \times \ldots \times \\
p(F^{(1)}|X, \theta^{(0)}) \, dF^{(N_h)} \ldots dF^{(1)}
\]

- Extremely challenging!
Inference for DGPs

- Inducing-variable approximations
  - VI+Titsias
    - Damianou and Lawrence (AISTATS, 2013)
    - Hensman and Lawrence, (arXiv, 2014)
    - Salimbeni and Deisenroth, (NeurIPS, 2017)
  - EP+FITC: Bui et al. (ICML, 2016)
  - MCMC+Titsias
    - Havasi et al (arXiv, 2018)
- VI+Random feature-based approximations
  - Gal and Ghahramani (ICML 2016)
  - Cutajar et al. (ICML 2017)
Inference for DGPs

- Inducing-variable approximations
  - VI+Titsias
    - Damianou and Lawrence (AISTATS, 2013)
    - Hensman and Lawrence, (arXiv, 2014)
    - Salimbeni and Deisenroth, (NeurIPS, 2017)
  - EP+FITC: Bui et al. (ICML, 2016)
  - MCMC+Titsias
    - Havasi et al (arXiv, 2018)

- VI+Random feature-based approximations
  - Gal and Ghahramani (ICML 2016)
  - Cutajar et al. (ICML 2017)
Example: DGPs with Random Features are Bayesian DNNs

Recall RF approximations to GPs (part II-a). Then we have:
Stochastic Variational Inference

- Define \( \Psi = (\Omega^{(0)}, \ldots, \mathcal{W}^{(0)}, \ldots) \)
- Lower bound for \( \log [p(Y|X, \theta)] \)

\[
\mathbb{E}_{q(\psi)} \left( \log [p(Y|X, \psi, \theta)] \right) - \text{DKL} [q(\psi) || p(\psi|\theta)],
\]

where \( q(\psi) \) approximates \( p(\psi|Y, \theta) \).
- DKL computable analytically if \( q \) and \( p \) are Gaussian!

Optimize the lower bound wrt the parameters of \( q(\psi) \)
Stochastic Variational Inference

• Assume that the likelihood factorizes

\[ p(Y|X, \psi, \theta) = \prod_{k} p(y_k|x_k, \psi, \theta) \]

• Doubly stochastic **unbiased** estimate of the expectation term
Stochastic Variational Inference

• Assume that the likelihood factorizes

\[ p(Y | X, \psi, \theta) = \prod_k p(y_k | x_k, \psi, \theta) \]

• Doubly stochastic \textbf{unbiased} estimate of the expectation term
  
  ▶ Mini-batch

\[ \mathbb{E}_{q(\psi)} (\log [p(Y | X, \psi, \theta)]) \approx \frac{n}{m} \sum_{k \in I_m} \mathbb{E}_{q(\psi)} (\log [p(y_k | x_k, \psi, \theta)]) \]
Stochastic Variational Inference

• Assume that the likelihood factorizes

\[ p(Y|X, \psi, \theta) = \prod_k p(y_k|x_k, \psi, \theta) \]

• Doubly stochastic unbiased estimate of the expectation term
  ▶ Mini-batch

\[ \mathbb{E}_{q(\psi)} \left( \log [p(Y|X, \psi, \theta)] \right) \approx \frac{n}{m} \sum_{k \in \mathcal{I}_m} \mathbb{E}_{q(\psi)} \left( \log [p(y_k|x_k, \psi, \theta)] \right) \]

▶ Monte Carlo

\[ \mathbb{E}_{q(\psi)} \left( \log [p(y_k|x_k, \psi, \theta)] \right) \approx \frac{1}{N_{MC}} \sum_{r=1}^{N_{MC}} \log[p(y_k|x_k, \tilde{\psi}_r, \theta)] \]

with \( \tilde{\psi}_r \sim q(\psi) \).
Stochastic Variational Inference

- Reparameterization trick

\[
(\tilde{\mathbf{W}}^{(l)}_{r})_{ij} = \sigma_{ij}^{(l)} \varepsilon_{rij}^{(l)} + \mu_{ij}^{(l)},
\]

with \( \varepsilon_{rij}^{(l)} \sim \mathcal{N}(0, 1) \)

- \ldots same for \( \Omega \)

- Variational parameters

\[
\mu_{ij}^{(l)}, (\sigma^2)_{ij}^{(l)} \ldots
\]

\ldots and the ones for \( \Omega \)

- Optimization with automatic differentiation in TensorFlow

Kingma and Welling, *ICLR*, 2014
Other Interesting GP/DGP-based Models
Other Interesting GP/DGP-Based Models (1)

**Convolutional GPs and DGPs**
- Wilson et al (NeurIPS, 2016)
- van der Wilk et al (NeurIPS, 2017)
- Bradshaw et al (Arxiv, 2017)
- Tran et al (AISTATS, 2019)

**Structured Prediction**
- Galliani et al (AISTATS, 2017)

**Network-structure discovery**
- Linderman and Adams (ICML, 2014)
- Dezfooli, Bonilla and Nock (ICML, 2018)
Other Interesting GP/DGP-Based Models (2)

Autoencoders


Constrained dynamics

- Lorenzi and Filippone, (ICML), 2018

Reinforcement Learning

- Rasmussen & Kauss (NIPS, 2004); Engel et al (ICML, 2005)
- Deisenroth and Rasmussen (ICML, 2011)
- Martin and Englot (Arxiv, 2018)

Doubly stochastic Poisson processes

- Adams et al (ICML, 2009); Lloyd et al (ICML, 2015)
- John and Hensman (ICML, 2018)
- Aglietti, Damoulas and Bonilla (AISTATS, 2019)
Applications and extensions of GP models by using more complex priors (e.g. coupled, compositions) and likelihoods

- Multi-task GPs by using correlated priors
- Dimensionality reduction via the GPLVM
- Probabilistic numerics, e.g. Bayesian optimisation
- Deep GPs
- Convolutional GPs
- Other settings such as RL, structured prediction, Poisson point processes
Interested in working at the cutting edge of research in ML and AI? contact

Richard Nock: richard.nock@data61.csiro.au
or
Edwin Bonilla: edwin.bonilla@data61.csiro.au