Efficient Gaussian Processes for data-driven decision making

Vincent Dutordoir

Department of Engineering
University of Cambridge

Report submitted to be registered for the PhD Degree
First-Year-Report

Jesus College August 2021

Abstract

This is where you write your abstract ...

Table of contents

Chapter 1

Introduction

1. Neural networks and Gaussian processes: complementry strengths and weaknesses
2. Ideally we have a single model that can handle low and high dimensional inputs, make robust uncertainty-aware predictions and can be used in big and small data regimes.
3. Gaussian processes and Bayesian Neural networks: connections [Neal, 1992; 1995; Williams and Rasmussen, 1996]
4. Problem with BNN is that (approximate) Bayesian inference is challenging
5. Deep Gaussian processes [Damianou and Lawrence, 2013]
6. Require accurate and scalable approximate Bayesian inference procedures

1.1 Contributions and Layout of this Report

This report represents my learning and the research that I conducted during the first year of my PhD degree. Most notably, we developed a novel sparse approximation for (deep) Gaussian processes based on the decomposition of the kernel in Spherical harmonics. In chapter 2 we cover the necessary theoretical background.

Chapter 3 In this chapter we introduce a new class of inter-domain variational GPs where data is mapped onto the unit hypersphere in order to use spherical harmonic representations. The inference scheme is comparable to Variational Fourier Features, but it does not suffer from the curse of dimensionality, and leads to diagonal covariance matrices between inducing variables. This enables a speed-up in inference, because it bypasses the need to invert large covariance matrices. The experiments show that our model is able to fit a regression model for a dataset with 6 million entries two orders of magnitude faster compared to standard sparse GPs, while retaining state of the art accuracy.

The content of this chapter is largely based on:

Vincent Dutordoir, Nicolas Durrande, and James Hensman [2020]. “Sparse Gaussian Processes with Spherical Harmonic Features”. In: *Proceedings of the 37th International Conference on Machine Learning (ICML)*,

with the exception of the algorithm for computing the spherical harmonics in high dimensions.

Chapter 3 Following up on the previous chapter, we use the decomposition of zonal kernels to design an interdomain inducing variable that mimics the behaviour of activation functions is neural network layers.

The content of this chapter is largely based on:

Vincent Dutordoir, James Hensman, Mark van der Wilk, Carl Henrik Ek, Zoubin Ghahramani, and Nicolas Durrande [2021a]. “Deep Neural Networks as Point Estimate for Deep Gaussian Processes”. In: *submission to NeurIPS*.

Chapter 4 In the last chapter of the report we will shed a light on what the future work will focus on: “Gaussian Decision Systems with Geometric Gaussian processes".

Chapter 2

Theoretical Framework

This chapter discusses Gaussian processes (GP) and deep Gaussian processes (DGPs), the composite model obtained by stacking multiple GP models on top of each other. We also review how to perform approximate Bayesian inference in these models, with a particular attention to Variational Inference. We also cover the theory of postive definite kernels and the Reproducing Kernel Hilbert Spaces (RKHS) they characterise.

2.1 Gaussian Processes

Gaussian processes (GPs) [Rasmussen and Williams, 2006] are non-parametric distributions over functions similar to Bayesian Neural Networks (BNNs). The core difference is that neural networks represent distributions over functions through distributions on weights, while a Gaussian process specifies a distribution on function values at a collection of input locations. This representation allows us to use an infinite number of basis functions, while still enables Bayesian inference [Neal, 1995].

Following from the Kolmogorov extension theorem, we can construct a real-valued stochastic process (i.e. function) on a non-empty set $\mathcal{X}$, $f : \mathcal{X} \to \mathbb{R}$, if there exists on all finite subsets $\{x_1, \dots x_N\} \subset \mathcal{X}$, a *consistent* collection of finite-dimensional marginal distributions over $f(\{x_1, \dots, x_n\})$. For a Gaussian process, in particular, the marginal distribution over every finite-dimensional subset is given by a multivariate normal distribution. This implies that, in order to fully specify a Gaussian process, it suffice to only define the mean and covariance (kernel) function because they are the sufficient statistics for every finite-dimensional marginal distribution. We can therefore denote the GP as

$$f \sim \mathcal{GP}(\mu, k), \tag{2.1}$$

where $\mu : \mathcal{X} \to \mathbb{R}$ is the mean function, which encodes the expected value of f at every x, $\mu(x) = \mathbb{E}_f[f(x)]$, and $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is the covariance (kernel) function that describes the covariance between function values, $k(x, x') = \mathrm{Cov}(f(x), f(x'))$. The covariance function has to be a symmetric, positive-definite function. The Gaussianity, and the fact that we can manipulate function values at some finite points of interest without taking the behaviour at any other points into account (the marginalisation property) make GPs particularly convenient to manipulate and use as priors over functions in Bayesian models – as we will show next.

Throughout this report, we will consider f to be the complete function, and intuitively manipulate it as an infinitely long vector. Moreover, $f(\boldsymbol{x}) \in \mathbb{R}^N$ denotes the function evaluated at a finite set of points, whereas $f^{\setminus \boldsymbol{x}}$ denotes another infinitely long vector similar to f but excluding $f(\boldsymbol{x})$. From the marginalisation property it follows that integrating out over the infinitely many points that are not included in $\boldsymbol{x}$, we obtain a valid finite-dimensional density for $f(\boldsymbol{x})$

$$p(f(\boldsymbol{x})) = \int p(f) \,\mathrm{d}f^{\setminus \boldsymbol{x}}. \tag{2.2}$$

In the case of GPs, this finite-dimensional marginal is given by a multivariate Gaussian distribition, fully characterised by the mean μ and the covariance function k

$$p(f(\boldsymbol{x})) = \mathcal{N}(\boldsymbol{\mu}_{\mathbf{f}}, \mathbf{K}_{\mathbf{ff}}), \quad \text{where} \quad [\boldsymbol{\mu}_{\mathbf{f}}]_i = \mu(x_i) \text{ and } [\mathbf{K}_{\mathbf{ff}}]_{i,j} = k(x_i, x_j). \tag{2.3}$$

Conditioning the GP at this finite set of points leads to a conditional distribution for $f^{\setminus \boldsymbol{x}}$, which is given by another Gaussian process

$$p(f^{\setminus \boldsymbol{x}} \,|\, f(\boldsymbol{x}) = \mathbf{f}) = \mathcal{GP}\big(\mathbf{k}_{\mathbf{f}\cdot}^\top \mathbf{K}_{\mathbf{ff}}^{-1}(\mathbf{f} - \boldsymbol{\mu}_{\mathbf{f}}), \;\; k(\cdot,\cdot) - \mathbf{k}_{\mathbf{f}\cdot}^\top \mathbf{K}_{\mathbf{ff}}^{-1} \mathbf{k}_{\mathbf{f}\cdot}\big), \tag{2.4}$$

where $[\mathbf{k}_{\mathbf{f}\cdot}]_i = k(x_i, \cdot)$. The conditional distribution over the whole function $p(f \,|\, f(\boldsymbol{x}) = \mathbf{f})$ has the exact same form as in eq. (2.4). This is mathematically slightly confusing because the random variable $f(\boldsymbol{x})$ is included both on the left and right-hand-side of the conditioning, but the equation is technically correct [Matthews et al., 2016].

2.1.1 The Beauty of Gaussian Process Regression: Exact Bayesian Inference

One of the key advantages of Gaussian processes for regression is that we can perform exact Bayesian inference. Assume a supervised learning setting where $x \in \mathcal{X}$ (typically, $\mathcal{X} = \mathbb{R}^d$) and $y \in \mathbb{R}$, and we are given a dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$ of input and corresponding output pairs. For convenience, we sometimes group the inputs in $\boldsymbol{x} = \{x_i\}_{i=1}^N$ into a single design matrix and outputs $\boldsymbol{y} = \{y_i\}_{i=1}^N$ into a vector. We further assume that the data is generated by an unknown function $f : \mathcal{X} \to \mathbb{R}$, such that the outputs are perturbed versions of functions evaluations at the corresponding inputs: $y_i = f(x_i) + \epsilon_i$. In the case of regression we assume a Gaussian noise model $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$. We are interested in learning the function f that generated the data.

[General introduction to Bayesian modelling] The key idea in Bayesian modelling is to specify a prior distribution over the quantity of interest. The prior encodes what we know at that point in time about the quantity. In general term, this can be a lot or a little. We encode this information in the form of a distribution. Then, as more data becomes available, we use the rules of probability, an in particlar Bayes' rule, to update our prior beliefs and compute a posterior distribution (see **bisschop**; MacKay [2003] for a thorough introduction).

Following the Bayesian approach, we specify a *prior* over the parameters of interests, which in the case of GPs is the function itself. The prior is important because it characterises the search space over possible solutions for f. Through the prior, we can encode strong assumptions, such as linearity, differentiability, periodicity, etc. or any combination thereof, which makes it possible to generalise well from very limited data. Compared to (Bayesian) parametric models, it is much more convenient and intuitive to specify priors directly in *function-space*, rather than on the weights of a parametric model [Rasmussen and Williams, 2006].

Following eq. (2.1) the prior over function evaluations at the datapoints is defined by the covariance function k. As we assume a à-priori zero mean function μ (without loss of generality) this can be written as:

$$p(f(\boldsymbol{x})) = \mathcal{N}(\mathbf{0}, \mathbf{K}_{\mathbf{ff}}), \quad \text{where} \quad [\mathbf{K}_{\mathbf{ff}}]_{i,j} = k(x_i, x_j). \tag{2.5}$$

Given the function f the likelihood factorises over datapoints and is given by a Gaussian:

$$p(\boldsymbol{y} \mid f) = \prod_{i=1}^{N} p(y_i \mid f) = \prod_{i=1}^{N} \mathcal{N}\Big(y_i \mid f(x_i), \sigma^2\Big) \tag{2.6}$$

We can obtain the posterior over the function using Bayes' rule and the marginalisation property

$$\begin{aligned} p(f \mid \boldsymbol{y}) &= \frac{p(f)\, p(\boldsymbol{y} \mid f)}{p(\boldsymbol{y})} && (2.7) \\ &= p(f^{\setminus \boldsymbol{x}} \mid f(\boldsymbol{x})) \frac{p(f(\boldsymbol{x})) \prod_{i=1}^{N} \mathcal{N}(y_i \mid f(x_i), \sigma^2)}{p(\boldsymbol{y})} && (2.8) \\ &= \mathcal{GP}(\mathbf{k}_{\mathbf{f}\cdot}^{\top} \mathbf{K}_{\mathbf{ff}}^{-1} \mathbf{f}, \;\; k(\cdot, \cdot) - \mathbf{k}_{\mathbf{f}\cdot}^{\top} \mathbf{K}_{\mathbf{ff}}^{-1} \mathbf{k}_{\mathbf{f}\cdot}), && (2.9) \end{aligned}$$

The marginal likelihood (model evidence)

$$p(\boldsymbol{y}) = \mathcal{N}\Big(\boldsymbol{y} \mid \mathbf{0}, \mathbf{K}_{\mathbf{ff}} + \sigma^2 \mathbf{I}\Big) \tag{2.10}$$

1. Plot: Prior, Data, Posterior

2. occam's razor

Problems A common criticism for GPs is that any modification to this approach breaks the Gaussian assumption.

1. Non-Gaussian likelihoods

2. Large datasets

3. Transformations: log or square transform

Solutions

1. Laplace
2. Expectation Propagation
3. Sparse Variational Inference

2.2 Approximate Inference with Sparse Gaussian Processes

1. General introduction to Variational inference [Blei et al., 2017] variational inference (VI), where the problem of Bayesian inference is cast as an optimization problem—namely, to maximize a lower bound of the logarithm of the marginal likelihood.
2. Sparse approximations [Snelson and Ghahramani, 2005; Quiñonero-Candela and Rasmussen, 2005]

2.3 Interdomain Inducing Variables

2.3.1 Example: heavyside inducing variable

$f \sim \mathcal{GP}$ defined on $\mathbb{S}^1$ (the unit circle), $f : [-\pi, \pi] \to \mathbb{R}, \theta \mapsto f(\theta)$.

kernel (Arc Cosine order 0):

$$k(\theta, \theta') = \kappa(\rho) = \pi - |\theta - \theta'| \tag{2.11}$$

$$\frac{\mathrm{d}}{\mathrm{d}\theta}\Big|_{\theta=\theta_m} k(\theta, \theta') = \tag{2.12}$$

$$u_m = \mathcal{L}_m(f) \tag{2.13}$$

$$\mathcal{L}_m = \frac{\mathrm{d}}{\mathrm{d}\theta}\Big|_{\theta=\theta_m} + \int \mathrm{d}\theta \tag{2.14}$$

$$\begin{aligned} \mathrm{Cov}(u_m, f(\theta')) &= \mathbb{E}_f[\mathcal{L}_m(f)\, f(\theta')] \qquad (2.15)\\ &= \mathcal{L}_m k(\theta', \cdot) \qquad (2.16)\\ &= \frac{\mathrm{d}}{\mathrm{d}\theta}\Big|_{\theta=\theta_m} k(\theta', \theta) + \int_{-\pi}^{\pi} k(\theta', \theta)\mathrm{d}\theta \qquad (2.17) \end{aligned}$$

2.4 Deep Gaussian Processes

Vincent Dutordoir, Hugh Salimbeni, Eric Hambro, John McLeod, Felix Leibfried, Artem Artemev, Mark van der Wilk, James Hensman, Marc P Deisenroth, and ST John [2021b]. “GPflux: A Library for Deep Gaussian Processes”. In: *arXiv preprint arXiv:2003.01115*

2.5 Covariance Functions

1. Positive Definite and Symmetry
2. RKHS
3. Bochner’s theorem
4. Mercer Decomposition
5. Examples of RKHS
6. RKHS through Spectral Decomposition
7. Representer Theorem
8. Show how sparse approximation links anchor points

Chapter 3

Spherical Harmonic Variational Gaussian Processes

1. Related work: Variational Fourier Features
2. Zonal kernels
    (a) Examples
    (b) Decomposition in spherical harmonics
    (c) Laplace Beltrami operator and Zonal kernel operator commute (this is why they share the same eigenfeatures)
3. How to compute the Spherical Harmonics: greedy algorithm
4. Experiments

Chapter 4

Deep Neural Networks as Point Estimates for Deep Gaussian Processes

1. Introduction
2. Related work: connection between BNN and GPs
3. Activated Interdomain features
4. Experiments

Chapter 5

Future Research

5.1 Geometric Gaussian Decision Systems

Problem setting

1. Black Box Functions $f : \mathcal{X} \to \mathcal{Y}$
2. We want to estimate a computable property $\mathcal{O}_{\mathcal{A}}(f)$
3. $\mathcal{A}$ is an algorithm $\mathcal{O}_{\mathcal{A}}(f) = \mathcal{A}(f)$
4. Evaluating f is *very* expensive (we can only evaluate it a limited amount of times)

Examples

1. Bayesian Optimisation: $\mathcal{A}(f) = \mathrm{argmax}_{x \in \mathcal{X}}\, f(x)$, which implies $\mathcal{O}_{\mathcal{A}}(f) = x^*$.
2. Sensor Placement (Active Learning): $\mathcal{O}_{\mathcal{A}}(f) = \mathrm{argmax}_{X \subset \mathcal{X}, |X| = T}\, \mathrm{MI}(f, f(X))$.
3. Level sets: $\mathcal{O}_{\mathcal{A}}(f) = \{X \subset \mathcal{X} : f(x) > C, \forall x \in X\}$.
4. Shortest path: $\mathcal{O}_{\mathcal{A}}(f) =$ shortest path between two nodes in a graph.

Model We model f by a Gaussian process

$$f \sim \mathcal{GP} \tag{5.1}$$

1. Low dimensions
2. Prior knowledge
3. Limited and very expensive data

Basically settings where DNNs are never going to be competitive with GPs - low-dim, very data-efficient, high-cost - not even if someone figures out how to do DNN uncertainty right, due to GP regret guarantees (conjecturally under reasonable assumptions) matching the best possible regret achievable by any model/decision system.

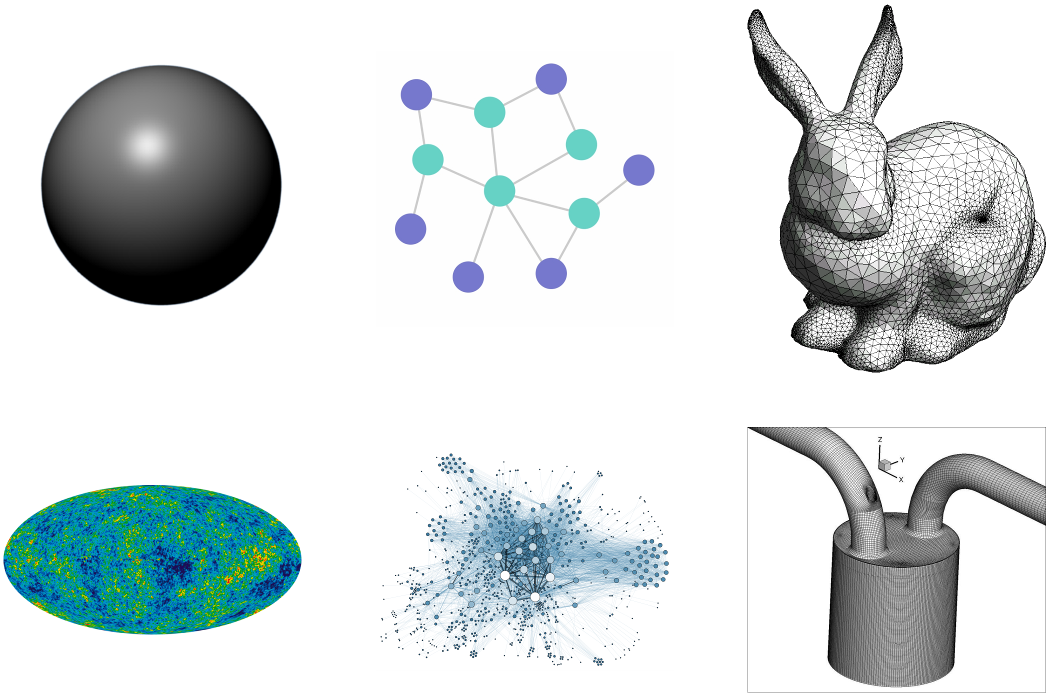

Fig. 5.1 Domains (top) and applications (bottom)

Objectives

1. Theory and analysis
2. Getting these ideas into people's hands to get applications off the ground
   (a) Graph GPs for combinatorial optimization use cases
   (b) Manifold GPs for scientific use cases

Collaborators

1. Alex Terenin (Imperial College London)
2. Willie Neiswanger (Stanford University)

5.2 Projects in Progress

1. "Pay Attention to Deep Gaussian Processes"
   Transformer Layer Gaussian Processes using an explicit feature representation of the attention operation.
   $$\exp(\boldsymbol{x}^\top \boldsymbol{y}) = \Phi^\top(\boldsymbol{x})\Phi(\boldsymbol{y})$$

2. A Unifying Theory for Interdomain Gaussian Processes.

3. VISH-PI: Probabilistic Integration with Variational Inducing Spherical Harmonics.

References

David M Blei, Alp Kucukelbir, and Jon D McAuliffe (2017). "Variational Inference: A Review for Statisticians". In: *Journal of the American Statistical Association.*

Andreas Damianou and Neil D. Lawrence (2013). "Deep Gaussian Processes". In: *Proceedings of the 16th International Conference on Artificial Intelligence and Statistics (AISTATS).*

Vincent Dutordoir, Nicolas Durrande, and James Hensman (2020). "Sparse Gaussian Processes with Spherical Harmonic Features". In: *Proceedings of the 37th International Conference on Machine Learning (ICML).*

Vincent Dutordoir, James Hensman, Mark van der Wilk, Carl Henrik Ek, Zoubin Ghahramani, and Nicolas Durrande (2021a). "Deep Neural Networks as Point Estimate for Deep Gaussian Processes". In: *submission to NeurIPS.*

Vincent Dutordoir, Hugh Salimbeni, Eric Hambro, John McLeod, Felix Leibfried, Artem Artemev, Mark van der Wilk, James Hensman, Marc P Deisenroth, and ST John (2021b). "GPflux: A Library for Deep Gaussian Processes". In: *arXiv preprint arXiv:2003.01115.*

David J. C. MacKay (2003). *Information Theory, Inference and Learning Algorithms.* Cambridge University Press.

Alexander G. de G. Matthews, James Hensman, Turner E. Richard, and Zoubin Ghahramani (2016). "On Sparse Variational Methods and the Kullback-Leibler Divergence between Stochastic Processes". In: *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics (AISTATS).*

Radford M. Neal (1992). "Bayesian Mixture Modeling". In: *Maximum Entropy and Bayesian Methods.*

Radford M. Neal (1995). *Bayesian Learning for Neural Networks.* Springer.

Joaquin Quiñonero-Candela and Carl E. Rasmussen (2005). "A Unifying View of Sparse Approximate Gaussian Process Regression". In: *Journal of Machine Learning Research.*

Carl E. Rasmussen and Christopher K. I. Williams (2006). *Gaussian Processes for Machine Learning.* MIT Press.

Edward Snelson and Zoubin Ghahramani (2005). "Sparse Gaussian Processes using Pseudo-inputs". In: *Advances in Neural Information Processing Systems 4 (NIPS 2005).*

Christopher K. I. Williams and Carl E. Rasmussen (1996). "Gaussian processes for regression". In: