A review of the paper: Variational Learning of Inducing Variables in Sparse Gaussian Processes

Mathis Hardion
MVA, Télécom Paris
mathis.hardion@ens-paris-saclay.fr

Lucas Haubert
MVA, Mines Saint-Étienne
lucas.haubert@ens-paris-saclay.fr

Abstract

Gaussian Processes (GPs) are a popular prior over functions in regression tasks, but the standard approach becomes untractable for large datasets which are ubiquitous in modern data science. It is therefore crucial to find reliable, computationally inexpensive approximations for the model to be usable in practice. The studied paper builds upon sparse GPs by introducing a variational framework to select the inducing variables, which enjoys the benefits of keeping the prior exact by only approximating the posterior, and reduced overfitting due to treating the inducing variables separately. This report summarizes the approach and highlights its key strengths and weaknesses. Numerical experiments are conducted to compare the method to classic regression algorithms and thus get a broader perspective on GP models. Extensions in the literature are reviewed and novel tentative directions of further research are suggested.

1 Introduction

This report discusses the main contributions of the studied article [20], their strengths, shortcomings and extensions. The paper introduces a novel (at the time) framework to approximate Gaussian Processes (GPs). GPs are widespread in machine learning [15] as a nonparametric prior over functions. In the context of standard GPs, it may be needed to invert a covariance matrix, which can be computationally heavy, then the exact formulation of the posterior and marginal likelihood become intractable for large datasets. Finding reliable, computationally cheaper estimates is thus of relevance in this context. The paper builds upon sparse GP methods [18, 24, 15] which essentially consist in selecting a small amount of inducing variables to construct a low-rank approximation of the covariance matrix, which is much cheaper to deal with.

The report is organized as follows: Section 2 provides context about sparse Gaussian Process Regression (GPR), which is the point of the studied paper. Section 3 then explains and critically adresses the contributions of the paper. The experiments proposed as part of this project are presented in Section 4. Then, extensions of this work in the existing literature as well as potential further directions of research are discussed in Section 5.

2 Background

2.1 Gaussian Process Regression

Given a space $\mathcal{X}$, $\mathbf{f} = (\mathbf{f}(\mathbf{x}))_{\mathbf{x}\in\mathcal{X}}$ is said to be a GP if any subset of random variables indexed by $\mathcal{X}$ is a Gaussian vector. It is uniquely determined by its mean $(m(\mathbf{x}))_{x\in\mathcal{X}}$ and covariance kernel $(k(\mathbf{x},\mathbf{x}'))_{\mathbf{x}\in\mathcal{X}}$, the latter often being parametrized by a set of hyperparameters θ. In machine learning, one can use such a GP as prior over a function f on $\mathcal{X}$, then extend it as a posterior given the data. Then, we consder as inputs $(\mathbf{x}_i)_{i=1}^n$ and we observe $(y_i)_{i=1}^n$ their noised images through f, i.e.

Project for the course "Bayesian Machine Learning", Master MVA, ENS Paris-Saclay

$y_i = f(x_i) + \varepsilon_i$ where $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$. By considering $\mathbf{f} = (f(\mathbf{x}_i))_i$ as a GP, the induced posterior is also a GP. For exact regression, one would estimate the parameters θ, σ by maximizing the marginal likelihood, that is:

$$p(\mathbf{y}) = \mathcal{N}(\mathbf{y}|0, \sigma I_n + K_{nn}) \tag{1}$$

where $K_{nn} \coloneqq (k(\mathbf{x}_i, \mathbf{x}_j))_{i,j}$ is the covariance matrix on the training inputs. The problem is that maximizing the above likelihood involves inverting a $n \times n$ matrix at each step of the optimization routine since K_{nn} depends on the parameters, which can be too computationally demanding for large large datasets.

2.2 Inducing variables

In order to face the problem of computational complexity with large datasets, the idea of sparse GP methods is to replace K_{nn} by an approximation matrix Q_{nn} which can be inverted quicker. Approaches like [19], [17] use a smaller subset X_m of $m \ll n$ 'inducing variables'[1], consisting in m training samples or 'pseudo-inputs'. Learning this set, as well as the hyperparameters, is crutial to obtain a sparse GP method. This idea is to approximate the marginal likelihood (1) to get these values. To this end, the Projected Process (PP) approximation [17] considers the approximated marginal likelihood by replacing K_{nn} with some matrix Q_{nn}. This substitution is defined by the Nyström approximation:

$$Q_{nn} = K_{nm} K_{mm}^{-1} K_{mn} \tag{2}$$

where K_{nm} is the cross-covariance matrix between training and inducing points, $K_{nm} = K_{mn}^T$, and K_{mm} is the covariance matrix on the inducing inputs[2]. With such an approximation, the Woodbury matrix identity [7] allows for much faster inversion of $\sigma I_n + Q_{nn}$, in $\mathcal{O}(m^2 n)$ rather than $\mathcal{O}(n^3)$.

The main shortcomings of this method are that it ends up modifying the prior, and can be prone to overfitting since X_m adds an extra set of variables to be optimized. To tackle this issue, Titsias [20] suggests a variational approach to determine X_m instead, which is adressed in Section 3.

3 Contributions of the article

3.1 Variational Learning

The main idea of the paper is to directly approximate the posterior distribution rather than modify the prior. To do so, the author starts by observing that the predictive distribution writes the first integral for points $\mathbf{z}$:

$$p(\mathbf{z}|\mathbf{y}) = \int p(\mathbf{z}|\mathbf{f}) p(\mathbf{f}|\mathbf{y}) d\mathbf{f} = \int p(\mathbf{z}|\mathbf{f}_m, \mathbf{f}) p(\mathbf{f}|\mathbf{f}_m, \mathbf{y}) p(\mathbf{f}_m|\mathbf{y}) d\mathbf{f} d\mathbf{f}_m$$

and then uses a set of function points $\mathbf{f}_m$ corresponding to pseudo-inputs X_m, independent from the training inputs, as well as the augmented joint model $p(\mathbf{y}|\mathbf{f}) p(\mathbf{z}, \mathbf{f}_m, \mathbf{f}_m)$, to write the second integral.

Consider the scenario where $\mathbf{f}_m$ is a sufficient statistic for $\mathbf{f}$, i.e. $p(\mathbf{z}|\mathbf{f}_m, \mathbf{f})$ with $p(\mathbf{z}|\mathbf{f}_m)$, then if $q(\mathbf{z}) \coloneqq p(\mathbf{z}|\mathbf{y})$ and $\phi(\mathbf{f}_m) \coloneqq p(\mathbf{f}_m|\mathbf{y})$,

$$q(\mathbf{z}) = \int p(\mathbf{z}|\mathbf{f}_m) \phi(\mathbf{f}_m) d\mathbf{f}_m = \int q(\mathbf{z}, \mathbf{f}_m) d\mathbf{f}_m \tag{3}$$

In practice, it may be difficult to find inducing variables $\mathbf{f}_m$ that are sufficient statistics, then satisfy this scenario. In such case, the author chooses $\phi(\mathbf{f}_m)$ to be a 'free variational Gaussian distribution', different a priori from $p(\mathbf{f}_m|\mathbf{y})$ and discribed by some mean μ and covariance matrix A. The approximate posterior GP mean and covariance are then described by those parameters, as well as auxialiary matrices K [20]. Note that this system induces the form of the posterior which is computed in $\mathcal{O}(m^2 n)$ rather than $\mathcal{O}(n^3)$ in a standard setting.

Now the problem consists in choosing ϕ, i.e. μ and A, and the inducing inputs X_m. The author then introduces a variational approach to learn X_m as a variational parameter by minimizing some

[1] Set of inducing variables: $X_m = \left(\mathbf{x}_i^{(m)}\right)_{i=1}^m$

[2] Auxiliary matrices: $K_{nm} \coloneqq \left(k\left(\mathbf{x}_i, \mathbf{x}_j^{(m)}\right)\right)_{i,j=1}^{n,m} =: K_{mn}^T$ and $K_{mm} \coloneqq \left(k\left(\mathbf{x}_i^{(m)}, \mathbf{x}_j^{(m)}\right)\right)_{i,j=1}^m$

Kullback-Leibler (KL) divergence. The idea is to minimize the 'distance' (KL divergence, which is actually not a distance, but remains a relevant comparison tool) between the variational distribution and the exact posterior distribution over the latent function values. The underlaying problem is then:

$$\min KL(q(\mathbf{f}, \mathbf{f}_m)||p(\mathbf{f}, \mathbf{f}_m|\mathbf{y})) \tag{4}$$

This minimization is also equivalent to the maximization of the following variational lower bound of the true log marginal likelihood, once ϕ is optimal:

$$F_V(X_m) = \log(\mathcal{N}(y|0, \sigma^2 I + Q_{nn})) - \frac{1}{2\sigma^2} Tr(\tilde{K}) \tag{5}$$

where $Q_{nn} = K_{nm}K_{mm}^{-1}K_{mn}$ and $\tilde{K} = K_{nn} - Q_{nn}$. This objective function is relevant, since it contains a regularization trace term, which differentiates it from the logarithm of (1) (Projected Process marginal likelihood). This term represents the total variance of the conditional prior which is also the squared error of predicting the training latent values $\mathbf{f}$ from the inducing variables $\mathbf{f}_m$. Then if it is equal to 0, $\mathbf{f}_m$ are sufficient statistics. Yet in the case of variational learning, the goal is to optimize the whole term.

Two optimization paradigms are to consider: gradient-based optimization, in the case where the objective functions are differentiables, and combinatory approach else. The author notices that some kernel functions, especially in high dimension, may not be differentiables. In such case, the paper suggests a disrecte selection over the training inputs to build X_m. It is based on a greedy algorithm and is comparable to a EM approach (select and adapt). The main result is that it makes the objective function F_V monotically increase, hence the divergence $KL(q(\mathbf{f}, \mathbf{f}_m)||p(\mathbf{f}, \mathbf{f}_m|\mathbf{y}))$ decreases optimally (see more details in [20]). Yet, the comparisons and experiments done in the paper take place in the differentiable setting. Hence, gradient-based optimization approaches are used in what follows.

3.2 Comparisons and experiments

Given the framework, paper [20] provides toy experiments and comparisons over real-world datasets between the variational approach with F_V and PP and SPGP approaches (similar to PP, but here $Q_{nn} = \text{diag}(K_{nn} - K_{nm}K_{mm}^{-1}K_{mn}) + K_{nm}K_{mm}^{-1}K_{mn}$).

First a small one-dimensional dataset with 200 training points is considered from [19]. Sparse models are trained, then compared with exact GP predictions. The predictive predoctions of the 3 methods (variational, PP, SPGP) are built from 15 inducing points. As expected, the variational approach outperforms the two others, leading to a variational lower bound value close to the true log marginal likelihood, by additionally matching the hyperparameters. In a more challenging setting, that is maintaining only 20 training points, the variational GP approach still matches the exact distribution, while PP and SPGP overfit or lack of accuracy. Finally, this toy experiments shows at the same time: the accuracy of the variational approach, as well as its robustness to overfitting, due to its trace regularization term in (5).

In a second time, the paper explores four real-world datasets: Boston-housing, a small one (about 500 training data), and KIN40K, SARCOS and ABALONE, large ones (thousands of training data). The maximization, w.r.t. the hyperparameters and the inducing variables, of the objective functions relative to variational / PP / SPGP approaches, is studied. By initializing the inducing variables in different ways, and considering the standardized mean squared error and the standardized negative log probability density as metrics, we notice again that only the variational approach is able to provide adequate results in terms of accuracy and overfitting and match the exact GP prediction.

3.3 Strengths and limitations of the paper

The above studies demonstrate the strengths of the paper. Indeed, the author suggests a variational approach to Sparse Gaussian Regression by considering the inducing variables in a specific way, which has been influential in the literature as we shall see in section 5. It allows to keep the exact prior by approximating only the posterior which leaves room for strong mathematical rigor, by means of a KL comparison and optimization for instance. Moreover, the underlying algorithms are well-built to be deployed on GPU and enjoy computational efficiency.

Yet, some rooms of improvement can be highlighted. First, in Section 3 'Variational Learning' of the paper, because of practical difficulties to find inducing variables $\mathbf{f_m}$ that are sufficient statistics, ϕ is

taken as a "free" variational Gaussian distribution, which was not directly justified to converge to the exact posterior. Such convergence was more recently investigated in [2] (see section 5).

Also, one can make a few remarks on the datasets involved in the experiments of the paper. The toy experiments are made with a one-dimensional dataset consisting in 200 points. Its nature and the corresponding regression task are then very simple to handle, as expected. Regarding the real-life datasets provided, a simple linear regression can perform well on some of those (see section 4), meaning they can be considered to be quite simple. Thus, the proposed method could have been tested on more complex datasets, in order to further test its capabilities.

Finally, without it being an absolute weak point, the article focuses only on GP methods for regression, and does not mention performance of other paradigms on the same tasks. Yet, it could be informative to make a comparison between the suggested Sparse Gaussian Regression in [20] and other models, in order to highlight the relevance of the Gaussian modeling.

4 Experiments

We now apply the variational SGP method provided by the maximization of (5) to different datasets and compare its performance with other simple regression algorithms (provided by scikit-learn [13]). We do so using the efficient, GPU-accelerated framework provided by the GPyTorch library [6], and use the Standardized Mean Square Error as performance metric as in [20]. The code, data and hyperparameters used are provided in [8]. The results are reported table 1. One can notice that for

	LR	GB	VARSGP
SARCOS [22]	0.075	**0.0258**	0.034
Abalone [12]	0.478	0.574	**0.436**
Forest Fires [3]	**0.999**	1.249	1.009
Electrical Grid Simulated Data [1]	0.359	**0.045**	0.063

Table 1: SMSE of Linear Regression (LR), Gradient Boosting (GB), and variational sparse GP (VARSGP) on four datasets.

the SARCOS and Abalone datasets used in the paper, Linear Regression and Gradient Boosting can achieve somewhat similar or even better performance than the variational sparse GP (and do so with lower run time). The same can be said about the Forest Fires data, although it is a much more complex regression task and all tested methods perform poorly. Finally, the Electrical Grid Simulated Data is complex enough that linear regression is far from the best, but GB still outperforms our VARSGP reimplementation. From these observations, one can put Gaussian Processes in perspective: while they provide an elegant and versatile framework, they may still be outperformed by more basic, simpler to train algorithms depending on the task. In a Machine Learning pipeline, it is indeed best to start with simple models to assess the complexity of the task before using more advanced models like GPs.

5 Extensions

5.1 In the literature

Since the publication of [20], many works have improved the method and expanded upon the variational SGP framework. As suggested at the end of the paper, the next step was the utilization of the method within Gaussian Process Latent Variable Models achieved the next year [21]. The main improvements of more recent literature on the topic is scalability and GPU acceleration, which are important to consider in modern data challenges. Some methods include leveraging Stochastic Gradient Descent with natural gradients [9], scaling GP classification with variational bounds [10], and batch conjugate gradient algorithms with PyTorch and GPU integration yielding the GPyTorch python package [6]. One should also note that modern methods can allow for exact (i.e. non sparse) GP inference even on large datasets [23]. A review of recent scalable methods can be found in [11]. On the theoretical standpoint, the convergence rates of the variational approximation were recently investigated by [2], showing that one can make the KL divergence arbitrarily small with the rate of increase for m slower than the increase in n.

5.2 Other possible directions of further research

We now provide possible extensions that are, to the best of our knowledge, original to this report. Concerning the maximization of (5), a recent contribution involving JKO flows in the Bures-Wasserstein space [4] seems promising and could potentially be utilized within variational SGP to further enhance performance. The following question could also be further investigated: can distances other than KL be utilized to quantify discrepancy between the variational distribution and the true augmented posterior in (4), and still enjoy scalability ? The KL divergence has a relevant relationship with the maximum likelihood estimation, however it suffers from lacking symmetry and the triangle inequality, and from the fact that it goes to $+\infty$ as soon as the supports of the measures are disjoint. Other metrics such as the Wasserstein distance or its computationally cheaper counterpart, the Sinkhorn divergence, have gained traction in the computational mathematics community for their geometric properties [16, 14]. As such, considering the MSE is one of the most widely used evaluation metrics, it would be possible to directly account for it in the variational minimization by using Wasserstein-2 or the like. Despite our best efforts, we have unfortunately not been able to find a tractable way to apply such an idea, due to the following difficulties: first, the simplifications that happen with the KL divergence seem to be specific to it, meaning it is more challenging to get a closed-form formula for a variational bound. Second, even if one was able to apply gradient descent with the Sinkhorn divergence using the framework of [5], it would be required to sample from the true posterior which is precisely what is untractable in our case of interest. Maximum Mean Discrepancy (MMD) norms would most likely encounter similar problems.

6 Conclusion

The studied article 'Variational Learning of Inducing Variables in Sparse Gaussian Processes' [20] is complete and innovative (at its time) when it comes to inference by leveraging Gaussian Processes. In particular, it suggests a variational framework for sparse GP regression that outperforms the state-of-the-art at its time. The idea is to learn jointly the hyperparameters and the inducing variables by minimizing the KL divergence between the exact GP distribution (posterior) and the approximate one. It is at the time accurate, but also designed to avoid overfitting.

Strenghts and limitations of the paper have been pointed as part of this review. First notice that the variational approach differs in nature by comparison to PP and SPGP, in that it keeps the prior unchanged and deals only with the posterior with the data. Also, this way of proceeding can be deployed easily on GPU and enjoy computational efficiency. Yet, some limitations concerning the simplicity of the experiments, as well as the lack of comparison between non-GP based regression methods, have been pointed. This is why we proceeded to comparisons between GP methods, in particular the variational approach, and non-GP based ones, such as Linear Regression and Gradient Boosting, on various datasets. Gradient Boosting could provide better results that the article's approach, meaning that it is relevant not to focus on GP methods only, but first consider the nature of the problem and the relevance / need to exploit more advanced models, such as variational GP.

Finally, some extensions of this work have been highlighted. The most significant ones from the litterature concern the scalability of GP-based methods on GPU, and the advent of exact (non-sparse) GP methods, even on large datasets (which was the main motivation of the studies paper). Moreoverver original extensions were discussed in this report, concerning the mathematical tools to use to build and solve the optimization problems. Discussions on the KL divergence, as well as the Sinkhorn divergence and the Wasserstein distance were given.

References

[1] V. Arzamasov. *Electrical Grid Stability Simulated Data*. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5PG66. 2018.

[2] D. Burt, C. E. Rasmussen, and M. Van Der Wilk. "Rates of Convergence for Sparse Variational Gaussian Process Regression". In: *Proceedings of the 36th International Conference on Machine Learning*. Vol. 97. Proceedings of Machine Learning Research. PMLR, June 2019, pp. 862–871. URL: `https://proceedings.mlr.press/v97/burt19a.html`.

[3] P. Cortez and A. Morais. *Forest Fires*. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5D88D. 2008.

[4] M. Z. Diao et al. "Forward-Backward Gaussian Variational Inference via JKO in the Bures-Wasserstein Space". In: *Proceedings of the 40th International Conference on Machine Learning*. Vol. 202. Proceedings of Machine Learning Research. PMLR, July 2023, pp. 7960–7991. URL: https://proceedings.mlr.press/v202/diao23a.html.

[5] J. Feydy et al. "Interpolating between Optimal Transport and MMD using Sinkhorn Divergences". In: *The 22nd International Conference on Artificial Intelligence and Statistics*. 2019, pp. 2681–2690.

[6] J. Gardner et al. "GPyTorch: Blackbox Matrix-Matrix Gaussian Process Inference with GPU Acceleration". In: *Advances in Neural Information Processing Systems*. Vol. 31. Curran Associates, Inc., 2018. URL: https://proceedings.neurips.cc/paper_files/paper/2018/file/27e8e17134dd7083b050476733207ea1-Paper.pdf.

[7] L. Guttman. "Enlargement Methods for Computing the Inverse Matrix". In: *The Annals of Mathematical Statistics* 17.3 (1946), pp. 336–343. URL: https://doi.org/10.1214/aoms/1177730946.

[8] M. Hardion. *VARSGP repo*. URL: https://github.com/mhardion/VARSGP.

[9] J. Hensman, N. Fusi, and N. D. Lawrence. "Gaussian Processes for Big Data". In: *Uncertainty in Artificial Intelligence*. Vol. 29. AUAI Press, 2013.

[10] J. Hensman, A. Matthews, and Z. Ghahramani. "Scalable Variational Gaussian Process Classification". In: *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*. Vol. 38. Proceedings of Machine Learning Research. San Diego, California, USA: PMLR, May 2015, pp. 351–360. URL: https://proceedings.mlr.press/v38/hensman15.html.

[11] H. Liu et al. *When Gaussian Process Meets Big Data: A Review of Scalable GPs*. 2019. arXiv: 1807.01065 [stat.ML].

[12] W. Nash et al. *Abalone*. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C55C7W. 1995.

[13] F. Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[14] G. Peyré and M. Cuturi. *Computational Optimal Transport*. 2020. arXiv: 1803.00567 [stat.ML].

[15] C. E. Rasmussen and C. K. I. Williams. *Gaussian processes for machine learning*. Adaptive computation and machine learning. MIT Press, 2006, pp. I–XVIII, 1–248. ISBN: 026218253X.

[16] F. Santambrogio. *Optimal Transport for Applied Mathematicians. Calculus of Variations, PDEs, and Modeling*. 1st ed. Progress in Nonlinear Differential Equations and Their Applications. Birkhäuser Cham, 2015, pp. XXVII, 353.

[17] M. W. Seeger, C. K. I. Williams, and N. D. Lawrence. "Fast Forward Selection to Speed Up Sparse Gaussian Process Regression". In: *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics*. Vol. R4. Proceedings of Machine Learning Research. Reissued by PMLR on 01 April 2021. PMLR, Jan. 2003, pp. 254–261. URL: https://proceedings.mlr.press/r4/seeger03a.html.

[18] A. Smola and P. Bartlett. "Sparse Greedy Gaussian Process Regression". In: *Advances in Neural Information Processing Systems*. Vol. 13. MIT Press, 2000. URL: https://proceedings.neurips.cc/paper_files/paper/2000/file/3214a6d842cc69597f9edf26df552e43-Paper.pdf.

[19] E. Snelson and Z. Ghahramani. "Sparse Gaussian Processes using Pseudo-inputs". In: *Advances in Neural Information Processing Systems*. Vol. 18. MIT Press, 2005. URL: https://proceedings.neurips.cc/paper_files/paper/2005/file/4491777b1aa8b5b32c2e8666dbe1a495-Paper.pdf.

[20] M. Titsias. "Variational Learning of Inducing Variables in Sparse Gaussian Processes". In: *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics*. Vol. 5. Proceedings of Machine Learning Research. Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA: PMLR, Apr. 2009, pp. 567–574. URL: https://proceedings.mlr.press/v5/titsias09a.html.

[21] M. Titsias and N. D. Lawrence. "Bayesian Gaussian Process Latent Variable Model". In: *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*. Vol. 9. Proceedings of Machine Learning Research. Chia Laguna Resort, Sardinia, Italy: PMLR, May 2010, pp. 844–851. URL: https://proceedings.mlr.press/v9/titsias10a.html.

[22] S. Vijayakumar. *SARCOS data*. 2000. URL: https://gaussianprocess.org/gpml/data/.

[23] K. Wang et al. "Exact Gaussian Processes on a Million Data Points". In: *Advances in Neural Information Processing Systems*. Vol. 32. Curran Associates, Inc., 2019. URL: https://proceedings.neurips.cc/paper_files/paper/2019/file/01ce84968c6969bdd5d51c5eeaa3946a-Paper.pdf.

[24] C. Williams and M. Seeger. "Using the Nyström Method to Speed Up Kernel Machines". In: *Advances in Neural Information Processing Systems*. Vol. 13. MIT Press, 2000. URL: https://proceedings.neurips.cc/paper_files/paper/2000/file/19de10adbaa1b2ee13f77f679fa1483a-Paper.pdf.