Uncertainty-Aware Modeling of Learned Human Driver Steering Behaviors on High-Difficulty Maneuvers: Comparing BNNs and GPs*

Harry Fieldhouse[1] and David Cole[1]

Abstract— **We present a comparative analysis of two Bayesian approaches to representing internal models in human driver steering control: Gaussian Processes (GPs) and Bayesian Neural Networks (BNNs). We apply these methods within a recently proposed driver model that combines an uncertainty-aware learned internal model of the vehicle dynamics with a Model Predictive Control (MPC) framework, enabling representation of a range of driver behaviors. The GP and BNN approaches are evaluated with a focus on their ability to represent observed human behaviors, along with computational efficiency, prediction accuracy, and robustness. The GP model is shown to be particularly effective and robust in low-data regimes, providing interpretable uncertainty estimates leading to control that aligns well with human behavior observed during repeated driving maneuvers. BNNs, while computationally intensive to train, offer superior scalability and flexibility when dealing with high-dimensional systems or more complex and nonlinear dynamics, though predictions and learning are less interpretable.**

I. Introduction

Predicting accurately how human drivers may behave in a given circumstance is of high value when developing driver aids and autonomous vehicles that will interact with human drivers and other human driven vehicles on the road.

This paper explores the predictive modeling of steering behaviors, particularly as evaluated during demanding maneuvers involving nonlinear vehicle dynamics. Modeling human drivers' steering control is an area of active research, in which focus is shifting towards capturing nuances of human perception, adaptation, and learning to refine individual driver predictions [1]. Recent advances allow for more accurately modeling individual drivers' styles and for tracking evolution of driving behavior over time.

It is widely believed that when controlling a familiar system, humans rely on what they have learned of the system from experience to predict how it will respond to inputs, with this predictive model referred to as the *internal model* [2]. In computational driver modeling, the *internal model* represents the driver's understanding of the vehicle's behavior, and their expectation of the vehicle's response to inputs.

Recent driver models have used optimal control frameworks such as Model Predictive Control (MPC), that minimize the cost associated with predicted future states and control based on the internal model predictions [3], [4], [5]. Traditionally, the internal model has often been assumed to be accurate and fixed. However, to accurately model real drivers, internal models need to adapt, learn and track confidence in understanding, as humans do. This is especially apparent under challenging unfamiliar or nonlinear conditions [6], [7], [4], [5]. The cost applied to the future states aims to replicate the human incentives to proceed in the desired direction and avoid obstacles. Additional cost function elements are included to capture other aspects that human drivers seek or avoid, such as penalties on control actions: steering angle, angular velocity, angular acceleration, or on steering torque which relate to control effort and to the biological incentive to minimize the metabolic cost of implementing control actions. In [4], an additional cost on the uncertainty in predictions that drivers would have in underexplored or unpredictable regions of the vehicle dynamics was considered — it seems likely that humans would innately learn to include this when selecting control actions. In testing, it was shown that by varying this cost on predictive variance, different observed human driver behaviors could be replicated: cautiousness, where drivers are averse to uncertainty, with a high cost on variance; and adventurousness, where drivers are exploratory and unafraid of uncertainty, with a low cost [4].

Being able to model different driving behaviors like this has significant appeal; however making these variance predictions and optimizing over them can come at a high computational cost.

Two prominent Bayesian approaches to uncertainty-aware modeling of this internal knowledge are:

- **Gaussian Processes (GPs)** — Nonparametric models that directly estimate a distribution over functions, providing both predictive means and variances [8].
- **Bayesian Neural Networks (BNNs)** — Neural networks with distributions over weights that yield predictions with uncertainty estimates [9].

In previous work [4], [5], a GP model was used, building on the cautious nonlinear MPC approach using GPs developed by Hewing *et al.* in [10]; but including variance in the MPC cost to provide behavioral incentives, rather than as a probabilistic constraint, and with more limited assumed system knowledge. Many other works demonstrate various approaches to control using neural networks (NNs) [11], [12], [13]. This report provides a comparative analysis between these approaches, focusing primarily on the ability to model human driver action, and addressing computational efficiency and prediction accuracy.

II. Driver Model Overview

Internal models will be compared within a driver model architecture as introduced and described fully in [5]. The

*This work was supported by Toyota Motor Europe (TME), grant number G116038. The industrial supervisor is Xabier Carrera Akutain of TME.

[1]Department of Engineering, University of Cambridge, UK
hamf2@cam.ac.uk, djc13@cam.ac.uk

driver model contains a learned internal model of the vehicle dynamics, trained on a store of prior vehicle responses, which predicts future states of the vehicle given current observations and planned control inputs. Nonlinear MPC is used to optimise the steering control inputs over the prediction horizon. The cost function includes penalty terms on path following error (lateral and yaw displacement), control effort (first and second time-derivatives of steering angle) and on the variance of internal model predictions. State estimation is performed using an extended Kalman filter (EKF) based on observations and the internal model predictions. This model has been shown to be capable of representing a range of individual driving styles, behaviors and driver skill levels [4]. Particular focus was directed to replicating cautious and adventurous control styles, where cautious control was identified as control where the driver avoids situations with high internal model prediction variance and can be observed as gentle low magnitude early control inputs, where magnitude can increase over time when the driver's region of high confidence in predictions grows as they learn the dynamics, and achieved in the driver model with a high penalty on prediction variance in the cost function. Adventurous control is the opposite, observed as variable, large magnitude, and exploratory early control inputs.

Previous work with this driver model exclusively used GPs for the learned internal model with comparison made to an accurate, fixed internal model.

A different software implementation is used here to that described previously, with CasADi [14] as a symbolic framework to solve the constrained nonlinear MPC optimization problem. PyTorch [15] is used for GPU acceleration of model learning, with L4CasADi [16] used to interface the models with the CasADi solver.

A. Gaussian Process Internal Model

A GP defines a distribution over functions $\boldsymbol{f}(\boldsymbol{x})$ such that any finite set of function values follows a multivariate Gaussian distribution. Given training data $D = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^N$, the GP provides a predictive distribution for a new input $\boldsymbol{x}_*$ with mean and variance given by:

$$\mu_* = \boldsymbol{k}_*^T \left(K + \sigma_n^2 I\right)^{-1} y, \tag{1}$$

$$\sigma_*^2 = k(\boldsymbol{x}_*, \boldsymbol{x}_*) - \boldsymbol{k}_*^T \left(K + \sigma_n^2 I\right)^{-1} \boldsymbol{k}_*, \tag{2}$$

where K is the kernel matrix computed from training inputs, $k(\cdot,\cdot)$ is the kernel function (for the covariance between two points), $\boldsymbol{k}_*$ is the vector of kernel evaluations between $\boldsymbol{x}_*$ and each training point, and σ_n^2 is the noise variance [8].

The kernel used encodes assumptions about the structure and smoothness of the function — choosing appropriately is critical for performance [17]. The kernel hyperparameters are optimized in the learning process.

For predicting multi-dimensional outputs, when outputs are independent, GP regressions can be stacked in parallel with efficiency gains possible when the distribution and smoothness are similar across output dimensions so that the same kernel calculations can be reused [8].

B. Bayesian Neural Network Internal Model

A BNN extends on a standard NN by treating the network weights W as random variables with a posterior distribution given by:

$$p(W \mid D) = \frac{p(D \mid W)\, p(W)}{p(D)}. \tag{3}$$

The predictive distribution for a new input x_* is then obtained by marginalizing over the weight posterior:

$$p(\boldsymbol{y}_* \mid \boldsymbol{x}_*, D) = \int p(\boldsymbol{y}_* \mid \boldsymbol{x}_*, W)\, p(W \mid D)\, dW. \tag{4}$$

Based on this, BNNs can predict epistemic uncertainty in the output $\boldsymbol{y}_*$. Aleatory uncertainty can be modeled separately by adding a Gaussian noise term to the output layer for each output dimension.

Approximate inference methods (e.g., Monte Carlo sampling) are employed for predicting mean and variance since the integral is typically intractable [9]. The variance can also be propagated forwards using a maximum a posteriori (MAP) estimate of the weights and approximate moment matching, yielding a Gaussian approximation to the posterior [18]. This approximation also allows automatic differentiation of the variance with respect to the input, important for efficient MPC optimization, and is used here for this reason.

BNNs can capture complex nonlinear mappings and, when sufficiently trained, yield a predictive mean and uncertainty analogous to GPs.

Figure 1 compares a learned BNN representation of the vehicle dynamics with that of a GP. The models are learnt from 500 datapoints of simulated nonlinear vehicle dynamics with additive Gaussian noise (std. dev. 0.03 rad/s for the yaw rate output as plotted). The BNN architecture used here and throughout the paper is a 2-layer fully connected network with 64 hidden units per layer and rectified linear unit (ReLU) activations. The GPs use a squared exponential kernel with automatic relevance determination and have a zero mean function and Gaussian likelihood [8]. The GP and BNN models are trained on the same data. Predictions are shown in **Figure 1** for a forward vel. of 18 m/s with zero for all other initial state variables, over a range of input steering angles.

III. Comparative Analysis

A. Computational Efficiency

- **GPs:** Exact GP training has a computational complexity of $\mathcal{O}(N^3)$ (with N being the number of data points), and each inference is $\mathcal{O}(N)$ for the mean and $\mathcal{O}(N^2)$ for the variance [8]. Sparse GP approximations can reduce this cost (typically to $\mathcal{O}(NM^2)$, $\mathcal{O}(M)$, $\mathcal{O}(M^2)$ for training, mean prediction and variance prediction respectively, where M is the number of inducing points), making GPs suitable for small to moderate data sizes [19], [20], [21]. Intelligent selection of covariance functions can be used to exploit structure in the dynamics and minimize the number of training/inducing points needed [8], [22].
- **BNNs:** BNNs training is approximately $\mathcal{O}(N)$. Inference involves multiple forward passes (e.g., using

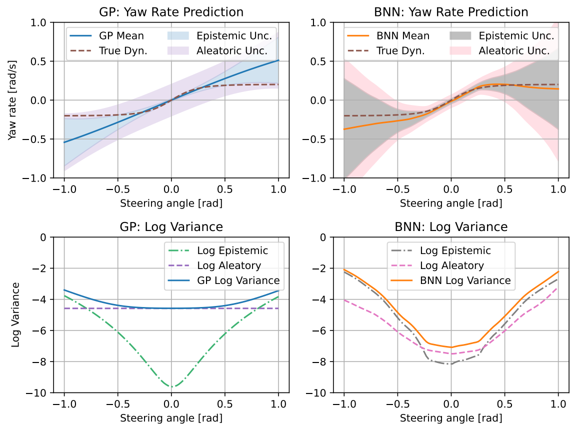

Fig. 1: GP and BNN internal model predictions for vehicle dynamics. In the top right figure uncertainties are stacked.

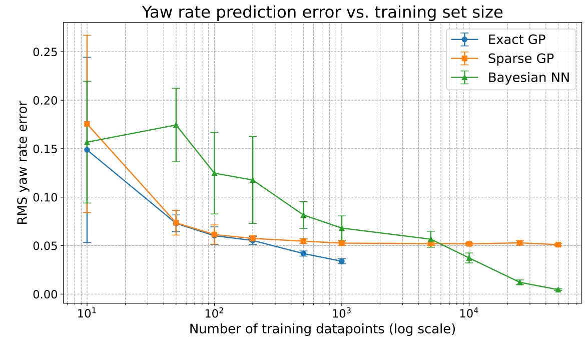

Fig. 2: Prediction accuracy against dataset size.

Monte Carlo dropout) [9], or additional operations to track variance propagation which adds overhead [18].

Thus, while BNNs scale better to higher dimensional/more complex dynamics with large datasets, for small datasets the overheads in training and evaluation can be more costly.

B. Prediction Accuracy

- **GPs** have demonstrated excellent performance in low-data regimes [23]. In the GP-based driver model [4], prediction accuracy improved over a series of maneuvers, as evidenced by decreasing path error.
- **BNNs** can capture more complex dynamics given sufficient data. In tasks involving high-dimensional inputs or extensive driving scenarios, a BNN is likely to outperform a GP [11].

Figure 2 shows a plot of prediction error versus number of training datapoints for BNN and GP methods (with the same model setups as for **Figure 1**). For the GP, a variational sparse GP [20] (with 100 inducing points and the same covariance, mean and likelihood functions as the exact GPs) is also shown. The data plotted comes from 100 repeats for each data point. These results show, when data is limited, GPs achieve much better modeling accuracy. However, once data is plentiful, BNN accuracy continues to improve toward very small modeling errors, while exact GP computation becomes impractical due to computational cost and sparse GPs saturate their ability to learn features of the dynamics with the number of inducing points they have available.

C. Robustness

When the internal model is used in a predictive control framework, reasonable responses in out-of-distribution regions and robustness to noise and disturbances are critical. Human drivers are known to be robust to disturbances and noise in the vehicle dynamics, and this is a key requirement for any driver model. GPs and BNNs have different strengths in this regard with MPC:

- **GPs** with common covariance kernels (e.g., squared exponential) have helpful properties for robustness with MPC — uncertainty is guaranteed to increase with deviation from training data and predictions are guaranteed to be smoothly varying (over a known, often infinite, number of derivatives) [8]. Poorly selected kernels or poor hyperparameter initialization can significantly impact robustness due to overfitting and local optima [17].
- **BNNs** performance heavily depends on training data and on convergence during training [9]. Extrapolation to unseen regions of the input space has no guarantees and can be unpredictable [24].

D. Ability to Represent Observed Human Behaviors

The ability to represent human behaviors is crucial for driver models.

- **GP** models have been shown to replicate human driver behaviors effectively, particularly in low-data regimes, where it can learn and adapt quickly [4].
- **BNN** models, while capable of representing complex behaviors, generally require larger datasets to achieve similar performance. Differences in how uncertainty is handled and represented can influence the simulated driver's behavior (cautious versus adventurous).

Simulations here are set up to replicate human driver experiments performed by Keen [7] (further analysed in [4]). In that experiment, participants drove an instrumented vehicle through a series of 12 elk test maneuvers on a test track at a constant speed (18 m/s). Their control was analysed and evidence of learning and different behaviours were observed. Performance of an adventurous-driving participant is shown in **Figure 3** from [7], [4]. Results are plotted for path following error and control effort across the maneuvers, showing both the trade-off between these objectives and the overall learning which occurs. Associated steering angle traces and contrasting cautious control are shown in [4].

In **Figure 4**, simulated results from the driver model with both GP and BNN models are shown over the same series of maneuvers. The same driver and vehicle model parameters were used for both internal models (replicating the experimental conditions identified in [7], with cost set for adventurous control with a low penalty on prediction variance). Datasets were initialised with 150 datapoints from the dynamics at normally distributed inputs. For each maneuver an additional 70 observed datapoints were added to

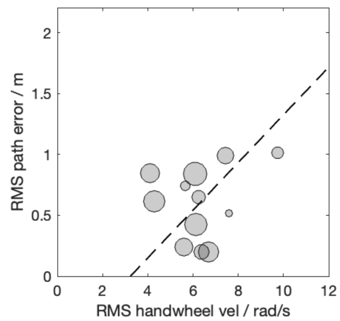

Fig. 3: Human results on elk test with adventurous driver [7], [4]. Maneuver number shown by increasing marker size.

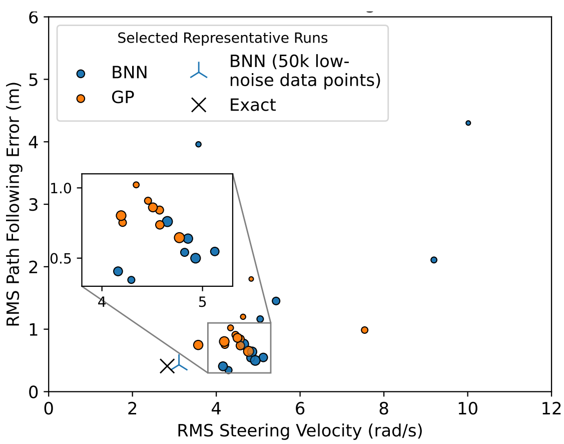

Fig. 4: Driver model elk test results with GP and BNN. The markers show statistically selected[1] representative maneuver series of 12 elk test runs from 100 repeats of this with each model. Maneuver number is indicated by marker size, with the first maneuver the smallest and the last (12th maneuver) the largest. The same vehicle/cost/noise parameters and dataset sizes are used for both models.

the dataset. Model training was performed using the full current dataset prior to each maneuver. A trend of improving performance is seen through the maneuvers, with the high run-to-run variability expected of adventurous control. With the BNN for the initial maneuvers the driver model performed poorly, not reliably maintaining control in maneuvers until approximately the fourth repeat, as indicated by very high path-following errors, demonstrating the impact of low BNN accuracy with limited data. The GP model on average displays lower control effort, believed to be a result of a smoother learned approximation to the vehicle dynamics due to the prior beliefs encoded in the GP kernel, leading to more smoothly varying control actions.

Adventurous and cautious steering control traces (with low and high weightings on the variance penalty in the cost function) are shown in **Figure 5** for the same repeated elk test maneuvers as simulated using the driver model with a BNN internal model. This shows the variance information from the BNN can be used to adjust the driving style. Large magnitude and highly variable control are seen in the early maneuvers for the adventurous control, before converging to a learned pattern, compared to consistent, lower magnitude control by the cautious driver with more gradual convergence. Equivalent control traces with a GP internal model and from human drivers are shown in [4].

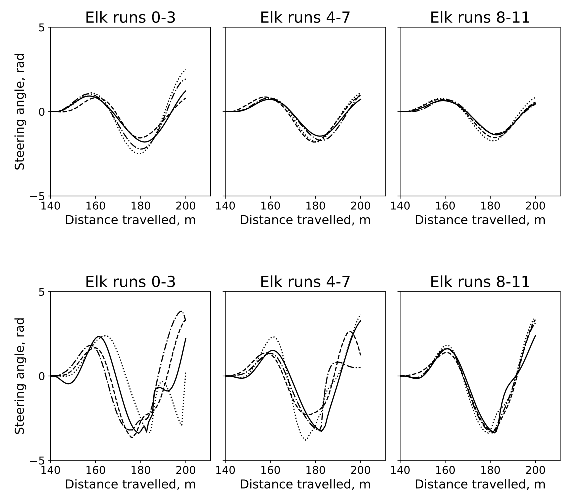

Fig. 5: Driver model steering traces with BNN internal model for repeated elk tests with high (top 3 plots) and low (bottom 3 plots) penalties on prediction variance to represent cautious and adventurous control respectively.

IV. Conclusion

Both Gaussian Processes and Bayesian Neural Networks offer viable frameworks for simulating human driver actions incorporating learning and uncertainty. GPs are particularly effective in data-sparse scenarios, providing interpretable uncertainty estimates that align well with human behavior observed during repeated driving maneuvers. Using BNNs makes achieving fine-grained interpretability more difficult, but they offer superior scalability and flexibility when dealing with high-dimensional data or more complex scenarios.

The choice between the two approaches depends on the specific application context:

- For small-scale applications or when interpretability is key, a GP-based internal model is attractive.
- For large-scale or high-dimensional problems, BNNs perform well.

Future work may explore hybrid approaches that combine the strengths of both methods — utilizing GPs initially for their accuracy and efficiency in low-data regimes, before switching to BNNs for increased internal model accuracy when datasets are larger.

[1]The representative selection method used finds the series where the multivariate Gaussian distribution over the metrics from that series has the minimum KL divergence from an average distribution of metrics over all the series repeats [4]. Selection is used to show the variability between runs, important to the modeling here, which is lost when averaging.

V. References

References

[1] J. Engström, J. Bärgman, D. Nilsson, B. Seppelt, G. Markkula, G. B. Piccinini, and T. Victor, "Great expectations: a predictive processing account of automobile driving," *Theoretical Issues in Ergonomics Science*, vol. 19, pp. 156–194, 2017. [Online]. Available: https://doi.org/10.1080/1463922X.2017.1306148

[2] D. M. Wolpert, Z. Ghahramani, and M. I. Jordan, "An internal model for sensorimotor integration," *Science*, vol. 269, pp. 1880–1882, 9 1995. [Online]. Available: https://doi.org/10.1126/science.7569931

[3] G. Markkula, E. Boer, R. Romano, and N. Merat, "Sustained sensorimotor control as intermittent decisions about prediction errors: computational framework and application to ground vehicle steering," *Biological Cybernetics*, vol. 112, pp. 181–207, 2018. [Online]. Available: https://doi.org/10.1007/s00422-017-0743-9

[4] H. Fieldhouse, S. Keen, and D. Cole, "Measurement and modelling of driver learning of steering control during successive obstacle avoidance manoeuvres," *Vehicle System Dynamics*, pp. 1–25, 2024. [Online]. Available: https://doi.org/10.1080/00423114.2024.2442462

[5] H. Fieldhouse and D. Cole, "State estimation and sensorimotor noise in a driver steering model with a gaussian process internal model," in *16th International Symposium on Advanced Vehicle Control (AVEC 2024): Lecture Notes in Mechanical Engineering*. Springer Science and Business Media Deutschland GmbH, 2024, pp. 64–70. [Online]. Available: https://doi.org/10.1007/978-3-031-70392-8_10

[6] A. Y. Ungoren and H. Peng, "An adaptive lateral preview driver model," *Vehicle System Dynamics*, vol. 43, pp. 245–259, 2005. [Online]. Available: https://doi.org/10.1080/00423110412331290419

[7] S. Keen, "Modeling driver steering behavior using multiple-model predictive control," *PhD Thesis at University of Cambridge Department of Engineering*, 2008.

[8] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. MIT Press, 2006. [Online]. Available: www.GaussianProcess.org/gpml

[9] C. Blundell, J. Cornebise, K. Kavukcuoglu, and D. Wierstra, "Weight uncertainty in neural networks," *32nd International Conference on Machine Learning, ICML 2015*, vol. 2, pp. 1613–1622, 5 2015. [Online]. Available: https://doi.org/10.48550/arXiv.1505.05424

[10] L. Hewing, J. Kabzan, and M. N. Zeilinger, "Cautious model predictive control using gaussian process regression," *IEEE Transactions on Control Systems Technology*, vol. 28, pp. 2736–2743, 2020. [Online]. Available: https://doi.org/10.1109/TCST.2019.2949757

[11] N. A. Spielberg, M. Brown, N. R. Kapania, J. C. Kegelman, and J. C. Gerdes, "Neural network vehicle models for high-performance automated driving," *Science Robotics*, vol. 4, 3 2019. [Online]. Available: https://doi.org/10.1126/scirobotics.aaw1975

[12] T. Salzmann, E. Kaufmann, J. Arrizabalaga, M. Pavone, D. Scaramuzza, and M. Ryll, "Real-time neural mpc: Deep learning model predictive control for quadrotors and agile robotic platforms," *IEEE Robotics and Automation Letters*, 2023. [Online]. Available: https://doi.org/10.1109/LRA.2023.3246839

[13] W. Gu, S. Primatesta, and A. Rizzo, "Physics-informed neural network for quadrotor dynamical modeling," *Robotics and Autonomous Systems*, vol. 171, p. 104569, 1 2024. [Online]. Available: https://doi.org/10.1016/j.robot.2023.104569

[14] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, "Casadi: a software framework for nonlinear optimization and optimal control," *Math. Prog. Comp*, vol. 11, pp. 1–36, 2019. [Online]. Available: https://doi.org/10.1007/s12532-018-0139-4

[15] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. K. Xamla, E. Yang, Z. Devito, M. R. Nabla, A. Tejani, S. Chilamkurthy, Q. Ai, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," *arXiv*, 2019. [Online]. Available: https://doi.org/10.48550/arXiv.1912.01703

[16] T. Salzmann, J. Arrizabalaga, J. Andersson, M. Pavone, and M. Ryll, "Learning for casadi: Data-driven models in numerical optimization," in *Learning for Dynamics and Control Conference (L4DC)*, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2312.05873

[17] A. G. Wilson and R. P. Adams, "Gaussian process kernels for pattern discovery and extrapolation," *30th International Conference on Machine Learning, ICML 2013*, pp. 2104–2112, 2 2013. [Online]. Available: https://doi.org/10.48550/arXiv.1302.4245

[18] A. Wu, S. Nowozin, E. Meeds, R. E. Turner, J. M. Hernández-Lobato, and A. L. Gaunt, "Deterministic variational inference for robust bayesian neural networks," *7th International Conference on Learning Representations, ICLR 2019*, 10 2018. [Online]. Available: https://doi.org/10.48550/arXiv.1810.03958

[19] E. Snelson and Z. Ghahramani, "Sparse gaussian processes using pseudo-inputs," *Advances in Neural Information Processing Systems 18*, 2005.

[20] M. K. Titsias, "Variational learning of inducing variables in sparse gaussian processes," *PMLR*, pp. 567–574, 4 2009.

[21] M. Bauer, M. V. D. Wilk, and C. E. Rasmussen, "Understanding probabilistic sparse gaussian process approximations," *NeurIPS (30th Conference on Neural Information Processing Systems) Proceedings*, 2016. [Online]. Available: https://doi.org/10.48550/arXiv.1606.04820

[22] J. Quinonero-Candela and C. Rasmussen, "A unifying view of sparse approximate gaussian process regression," *Journal of Machine Learning Research, v.6, 1935-1959 (2005)*, vol. 6, 12 2005.

[23] J. Snoek, H. Larochelle, and R. P. Adams, "Practical bayesian optimization of machine learning algorithms," *Advances in Neural Information Processing Systems*, vol. 4, pp. 2951–2959, 6 2012. [Online]. Available: https://doi.org/10.48550/arXiv.1206.2944

[24] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, "Understanding deep learning requires rethinking generalization," *Communications of the ACM*, vol. 64, pp. 107–115, 11 2016. [Online]. Available: https://doi.org/10.1145/3446776