Confidence Bound Minimization for Bayesian optimization with Student's-t Processes

Mr. Conor Clare
School of Computing, Ulster University
Newtownabbey, Northern Ireland
clare-c@ulster.ac.uk

Dr. Glenn Hawe
School of Computing, Ulster University
Newtownabbey, Northern Ireland
gi.hawe@ulster.ac.uk

Dr. Zhiwei Lin
School of Computing, Ulster University
Newtownabbey, Northern Ireland
z.lin@ulster.ac.uk

Professor Sally McClean
School of Computing, Ulster University
Coleraine, Northern Ireland
si.mcclean@ulster.ac.uk

ABSTRACT

Bayesian optimization seeks the global optimum of a black-box, objective function $f(\mathbf{x})$, in the fewest possible iterations. Recent work applied knowledge of the true value of the optimum to the Gaussian Process probabilistic model typically used in Bayesian optimization. This, together with a new acquisition function called Confidence Bound Minimization, resulted in a Gaussian probabilistic posterior in which the predictions were no greater than the known maximum (and no less than for minimum). Our novel work applies Confidence Bound Minimization to Bayesian optimization with Student's-t Processes, a probabilistic alternative which addresses known weaknesses in Gaussian Processes - outliers' probability and the calculation of posterior covariance. The new model is applied to the problem of hyperparameter tuning for an XGBoost classifier. Experiments show superior regret minimization and predictive accuracy, versus the popular Expected Improvement acquisition function. Combining Confidence Bound Minimization with a transformed Student's-t Process probabilistic model and known optima produces superior training regret minimization and posterior predictions for the Six-Hump Camel(2D) and Levy(4D) benchmark problems, which do not fall below true minima.

CCS CONCEPTS

• **Theory of computation** → **Theory and algorithms for application domains**; **Machine learning theory**; **Kernel methods**; **Gaussian processes**;

KEYWORDS

Bayesian Optimization, Student's-t Processes, Confidence Bound Minimization, Expected Improvement, Expected Regret Minimization, Gaussian Processes, XGBoost Classification, Hyperparameter Tuning, Six-Hump Camel(2D), Levy(4D).

APPIS 2020, January 7–9, 2020, Las Palmas de Gran Canaria, Spain

ACM ISBN 978-1-4503-7630-3/20/01…$15.00
https://doi.org/10.1145/3378184.3378193

ACM Reference Format:
Mr. Conor Clare, Dr. Glenn Hawe, Dr. Zhiwei Lin, and Professor Sally McClean. 2020. Confidence Bound Minimization for Bayesian optimization with Student's-t Processes. In *Proceedings of the 3rd International Conference on Applications of Intelligent Systems (APPIS 2020), January 7–9, 2020, Las Palmas de Gran Canaria, Spain.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3378184.3378193

1 INTRODUCTION

Single-objective optimization seeks the global optimum $\mathbf{x}^*$ to maximise a black-box objective function $f(\mathbf{x})$, for input $\mathbf{x}$ in a design-space χ [6]:

$$\mathbf{x}^* = \arg\max_{\mathbf{x}\in\chi} f(\mathbf{x})$$

Bayesian optimization [2, 6, 13] constructs a probabilistic model to efficiently seek the global maximum (or minimum) of a black-box, objective function $f(\mathbf{x})$ [5]. It is widely used in applications that have computationally-expensive non-linear objectives, such as hyperparameter tuning in machine learning [14, 18], aerostructural engineering [17] and nuclear science [3]. The probabilistic model incorporates our prior beliefs about f, updating the prior with observations sampled from $f(\mathbf{x})$, to obtain a posterior distribution that better approximates a black-box objective [9].

The two high-level modelling choices in Bayesian optimization are the probabilistic model and the acquisition function. A multivariate probabilistic model ('surrogate') is assumed to explain the joint behaviour of observations sampled [9]. Gaussian Processes (GPs) are usually chosen and are simply defined, for both prior and posterior distributions, using just the GP mean and the GP covariance function [5, 10].

Bayesian optimization combines the predictions for unknown $\mathbf{x}$ values, with predictive uncertainty through an acquisition function, such as the popular Expected Improvement (EI) acquisition function [8, 19]. The surrogate defines the acquisition, which then determines the next observation sampled for $f(\mathbf{x})$ [9]. The surrogate and acquisition function combine to balance 'exploration' (observation sampling in χ with high uncertainty), versus 'exploitation' (sampling around the current, best observations) [2].

Recently, [9] combined knowledge of known optima f^* with GP surrogates. Since the objective function does not exceed the maximum f^*, than neither should the maximum of the GP posterior's predictions. Two new acquisition functions, Confidence Bound Minimization (CBM) and Expected Regret Minimization (ERM) were derived, with CBM based on the Gaussian Process-Upper Confidence Bound acquisition function [15].

Despite simplicity and flexibility advantages, GP surrogates have two known weaknesses [11, 12, 17]. First, low probability is assigned to remote observations (outliers), despite contrasting, observed data e.g. aerostructural engineering [17]. Secondly, the GP posterior covariance does not depend on the black-box objective function's y_i-values. Instead, only the location of the training $\mathbf{x}_i \in \mathcal{D}_n$ determines the posterior covariance of a GP [11, 12, 17], where the training set of observations $\mathcal{D}_n$ is $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$ for $n = 1, \ldots, N$ iterations [17].

One recently proposed solution to the weaknesses of GPs are Student-t Processes (STPs), based on the multivariate Student's-t distribution [11, 12, 17]. STPs generalize the multivariate Gaussian distribution. STPs have an additional parameter, ν, which defines the 'degrees-of-freedom' of the STP [11, 12, 17] and controls STP kurtosis, influencing the size of the tails and hence, the probability of outliers [1]. This addresses the first weakness of GPs, regarding low probability of outliers. Further, unlike the GP posterior, the STP posterior covariance does depend on the black-box objective function's y_i-values [11, 12, 17], which addresses the second weakness of GPs.

Motivated by these recent advances, the main contributions of this paper are:

(1) to make use of knowledge of a known optimum value in Bayesian optimization with Student's-t Processes;
(2) to demonstrate the utility of this approach on a range of benchmark problems and a hyperparameter tuning problem from machine learning.

2 EXPLOITING KNOWN OPTIMA VALUE FOR BAYESIAN OPTIMIZATION WITH STPS

2.1 Gaussian Processes

A stochastic process $f(\mathbf{x})$ is Gaussian when observations jointly sampled have a multivariate Gaussian probability distribution [2, 10]. GPs are parameterized by two functions. The first is the mean function, $m(\mathbf{x})$, defining the expected value of an input, $\mathbf{x}$. The second is the kernel function $k(\mathbf{x}, \mathbf{x}')$, which calculates the covariance between two different inputs $\mathbf{x}$ and $\mathbf{x}'$ [17]:

$$f(\mathbf{x}) \sim \mathcal{GP}\left(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')\right)$$

The GP posterior covariance $\hat{\Sigma}_{GP}$ is given by [10, 17]:

$$\hat{\Sigma}_{GP} = K_{\mathbf{x}_*,\mathbf{x}_*} - K_{\mathbf{x}_*,\mathbf{x}} K_{\mathbf{x},\mathbf{x}}^{-1} K_{\mathbf{x},\mathbf{x}_*}$$

where $K_{\mathbf{x},\mathbf{x}}$ is the covariance defined by the kernel between the observed locations, $\mathbf{x}_i \in \mathcal{D}_n$; $K_{\mathbf{x}_*,\mathbf{x}}$ is the covariance of the kernel between the unobserved locations and observed locations; and $K_{\mathbf{x}_*,\mathbf{x}_*}$ is the covariance of the unobserved locations [17]. As can be seen, the GP posterior covariance does not depend on the black-box objective function's values [10].

2.2 Student's-t Processes

One recently proposed solution to those weaknesses is to instead use Student-t Processes (STPs), which are based on the multivariate Student's-t probability distribution. A stochastic process $f(\mathbf{x})$ is Student's-t when observations jointly sampled have a multivariate Student's-t probability distribution [11, 12, 17].

$$f(\mathbf{x}) \sim \mathcal{STP}\left(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'), \nu\right)$$

The STP posterior covariance $\hat{\Sigma}_{STP}$ is given by [17]:

$$\hat{\Sigma}_{STP} = \frac{\nu + y^T K_{\mathbf{x},\mathbf{x}}^{-1} y - 2}{\nu + |\mathcal{D}_n| - 2} (K_{\mathbf{x}_*,\mathbf{x}_*} - K_{\mathbf{x}_*,\mathbf{x}} K_{\mathbf{x},\mathbf{x}}^{-1} K_{\mathbf{x},\mathbf{x}_*}) \left(\frac{\nu - 2}{\nu}\right)$$

where $y^T K_{\mathbf{x},\mathbf{x}}^{-1} y$ is the squared Mahalanobis distance of the training $\mathbf{x}_i$ using their covariance. $|\mathcal{D}_n|$ is the number of samples in the training set of observations, $\mathcal{D}_n$ [17]. As can be seen, the STP posterior covariance depends on the black-box objective function's values.

2.3 Exploiting Known Optima

[9] combined known optima f^* with a GP surrogate, to produce GP posterior predictions which do not breach known limits. Their work developed the CBM acquisition function, defined in Eq. 1 for STPs, as $\alpha_n^{CBM+f^*}(\mathbf{x})$, to exploit knowledge about known optima in Bayesian optimization. This paper enhances Bayesian optimization by exploiting known optima f^*, using STPs and the CBM acquisition function, with $\hat{\mu}_{STP}(\mathbf{x})$ and $\hat{\sigma}_{STP}(\mathbf{x})$ the respective STP posterior mean and STP posterior standard deviation; and β_t an exploration/exploitation trade-off parameter [15]:

$$\alpha_n^{CBM+f^*}(\mathbf{x}) = \arg\max_{\mathbf{x} \in \chi} |f^* - \hat{\mu}_{STP}(\mathbf{x})| + \sqrt{\beta_t}\hat{\sigma}_{STP}(\mathbf{x}) \quad (1)$$

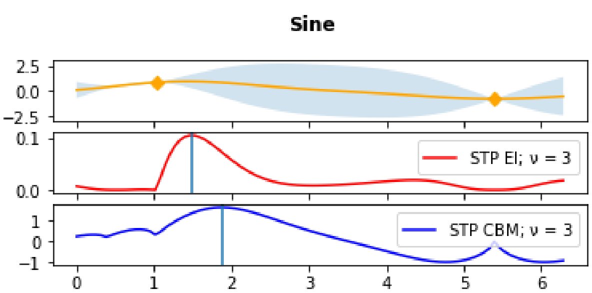

Figure 1: Estimating the Sine function: Bayesian optimization with an STP surrogate ($\nu = 3$) and a squared-exponential kernel [10], randomly initialized using 2 observations (top); with the next observation sampled using EI (middle) versus CBM (bottom).

The Bayesian optimization model is described in Algorithm 1, where the black-box objective function is $f(\mathbf{x})$; the acquisition function at iteration $n-1$ is $\alpha_{n-1}(\mathbf{x})$; and the training set of observations $\mathcal{D}_n$ is $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$, for $n = 1, \ldots, N$ iterations of Bayesian optimization.

Algorithm 1: Bayesian optimization [6]:

(1) **Input:** black-box objective function $f(\mathbf{x})$
(2) Construct $\mathcal{D}_0$, a set of randomly-sampled input-output pairs $(\mathbf{x}_i, y_i)$, where $\mathbf{x}_i \in \chi$, $y_i = f(\mathbf{x}_i)$;
(3) **for:** $n = 1, \ldots, N$ iterations **do**
(4) Train surrogate using $\mathcal{D}_{n-1}$
(5) select: $\mathbf{x}_n = \arg\max_{\mathbf{x} \in \chi} \alpha_{n-1}(\mathbf{x})$;
(6) query the objective f at $\mathbf{x}_n$ to obtain y_n;
(7) augment data: $\mathcal{D}_n = \mathcal{D}_{n-1} \cup \{(\mathbf{x}_n, y_n)\}$;
(8) **end for**
(9) **Return:** $\mathbf{x}_n = \arg\max_{\mathbf{x}_n \in D_n} y_n$

3 APPLICATIONS

Three applications of Bayesian optimization, combining the CBM acquisition function with STPs (ν = 3), are programmed in the Python language, using the 'pyGPGO' package [7]. Application 3.1, 3.2 and 3.3 use Algorithm 1 for model training and training regret comparison purposes, with the difference between known optima f^* and the best y-value defining training regret at each iteration of Bayesian optimization. Application 3.2 and 3.3 also use [9] to transform the surrogate estimate of $f(\mathbf{x})$ and create STP posterior predictions for $f(\mathbf{x})$ which do not fall below known optima, f^* [16]. Algorithm 1 models applications with global minima (rather than global maxima), by simply multiplying both $f(\mathbf{x})$ and f^* by -1, in Applications 3.2 and 3.3.

3.1 Hyperparameter Tuning - XGBoost Classification

Eq. 1 is applied to the "Skin Segmentation"[1] dataset, to tune six XGBoost hyperparameters for classification [9] (Table 1). The dataset is split into 85% for training and 15% for testing. The objective function is logistic, with 5 random initialization [4] and 30 iterations used for training [9]. 3-fold cross-validation of the XGBoost classifier is averaged to measure y. The kernel is Matérn 3/2 [10]. The best, known accuracy is f^* = 100% [9]. The difference between known optima f^* and the best classification accuracy defines XGBoost classification accuracy regret at each iteration. The STP CBM model (ν = 3) minimizes classification accuracy regret (Figure 2).

The hyperparameters chosen to train the STP EI (ν = 3) and STP CBM (ν = 3) XGBoost classifiers, correspond to the 'Best y-value' of 99.4508% for STP EI (ν = 3) versus 99.5891% for STP CBM (ν = 3) (Table 1). GP ERM [9] is also considered, together with STP ERM (ν = 3). Three acquisition functions - CBM, ERM and EI - are used with two surrogates - GP and STP (ν = 3) - to create 6 Bayesian optimization models. The STP CBM model (ν = 3) minimizes XGBoost classification accuracy regret in fewer iterations (marginally) than the other 5 Bayesian optimization models (Figure 2). The remaining 15% of the 'Skin Segmentation' dataset is used to test the posterior predictive accuracy of the 6 Bayesian optimization models, with posterior classification accuracy results shown in Table 2. GP CBM (99.6164%) performs best, followed by GP ERM (99.6137%) with STP CBM (ν = 3) and GP EI next (both 99.6110%). STP ERM (ν = 3) and STP EI (ν = 3) perform the worst (99.5919% and 99.5838%, respectively).

[1] https://archive.ics.uci.edu/ml/datasets/Skin+Segmentation

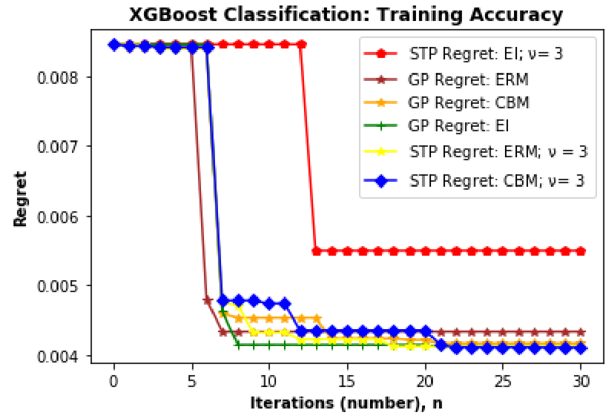

Figure 2: Comparing XGBoost classification accuracy regret on the 'Skin Segmentation' dataset, using Algorithm 1. The best, known accuracy is f^* = 100% [9], with minimized XGBoost classification accuracy regret for STP CBM (ν = 3).

Table 1: Hyperparameter tuning for XGBoost classification with Bayesian optimization [9], using STP EI (ν = 3) versus STP CBM (ν = 3).

Optimal Hyperparameter	STP EI (ν = 3)	STP CBM (ν = 3)
alpha	7.328311	1.788172
gamma	7.938542	3.497682
max_depth	13.999830	9.315171
subsample	0.501892	1.000000
min_child_weight	9.000008	2.475225
colsample	1.000000	1.000000
Best y-value	**99.4508%**	**99.5891%**

Table 2: Posterior predictive accuracy for XGBoost classification, with GP CBM and GP ERM performing best. Although STP CBM (ν = 3) has minimized XGBoost classification accuracy regret, GP CBM and GP ERM achieve comparable results in less iterations (Figure 2).

Model	Posterior Accuracy
GP CBM	99.6164%
GP ERM	99.6137%
STP CBM (ν = 3)	99.6110%
GP EI	99.6110%
STP ERM (ν = 3)	99.5919%
STP EI (ν = 3)	99.5838%

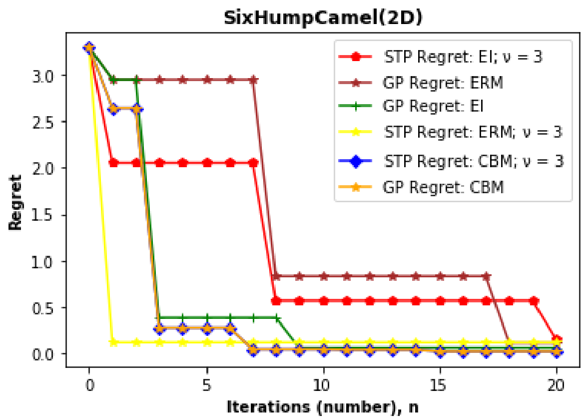

Figure 3: Comparing Six-Hump Camel(2D) training accuracy, using Algorithm 1. The best, known minimum f^* = -1.0316 [16], with minimized training regret for STP CBM (ν = 3) and GP CBM.

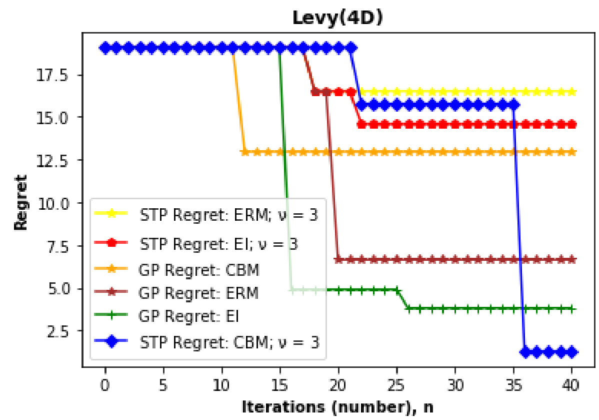

Figure 4: Comparing Levy(4D) training accuracy, using Algorithm 1. The best, known minimum f^* = 0 [16], with minimized training regret for STP CBM (ν = 3).

3.2 Benchmark Problem - Six-Hump Camel(2D)

Bayesian optimization with STP surrogates estimated training regret for the popular Six-Hump Camel(2D) benchmark problem [16]. The difference between known optima f^* and the best y-value, defines training regret at each iteration of Bayesian optimization. Three acquisition functions - CBM, ERM and EI - are combined with two surrogates - GP and STP (ν = 3) - to create 6 Bayesian optimization models. Each experiment has 5 random initialization [4] and 20 iterations (input d-dimension $\times$ 10 = 20) [4]. The kernel is Matérn 5/2 [10]. The 6 Bayesian optimization models are compared, with results shown in Figure 3. Combining the CBM acquisition function with a transformation of the original STP surrogate [9] and the known minimum f^*, produces STP posterior predictions for Six-Hump Camel(2D) which do not fall below the known minimum f^* = -1.0316 [16].

3.3 Benchmark Problem - Levy(4D)

Bayesian optimization with STP surrogates estimated training regret for the popular Levy(4D) benchmark problem [16]. The difference between known optima f^* and the best y-value, defines training regret at each iteration of Bayesian optimization. Three acquisition functions - CBM, ERM and EI - are combined with two surrogates - GP and STP (ν = 3) - to create 6 Bayesian optimization models. Each experiment has 5 random initialization [4] and 40 iterations (input d-dimension $\times$ 10 = 40) [4]. The kernel is Matérn 5/2 [10]. The 6 Bayesian optimization models are compared, with results shown in Figure 4. Combining the CBM acquisition function with a transformation of the original STP surrogate [9] and the known minimum f^*, produces STP posterior predictions for Levy(4D) which do not fall below the known minimum f^* = 0 [16].

4 CONCLUSIONS

Bayesian optimization with STP CBM (ν = 3) is applied to a hyperparameter tuning problem for XGBoost classification [9], showing superior regret minimization (Figure 2) and greater posterior predictive accuracy (Table 2), versus the STP EI (ν = 3) acquisition function. STP CBM (ν = 3) also shows superior training regret minimization versus STP EI (ν = 3) and GP EI for the Six-Hump Camel(2D) and Levy(4D) benchmark problems (Figures 3 and 4). Combining Confidence Bound Minimization with a transformed Student-t Processes' surrogate [9] and known optima, produces Student-t Processes' posterior predictions for the Six-Hump Camel(2D) and Levy(4D) benchmark problems [16] which do not fall below known minima [9].

STP CBM (ν = 3) outperforms both GP ERM and STP ERM (ν = 3) for Application 3.3, with comparable traning performance for Application 3.1 and 3.2. Interestingly, the performance of both STP CBM (ν = 3) and GP CBM is broadly equivalent for hyperparameter tuning (Application 3.1) and Six-Hump Camel(2D) (Application 3.2), suggesting that exploiting known optima f^* is more important than surrogate choice for CBM outperforming EI.

5 FUTURE WORK

The ERM acquisition function [9] for Bayesian optimization with STPs will be explored further by the authors in a separate paper. The CBM and ERM acquisition functions both use known optima f^* as an input, however f^* is not always known and in such scenarios, must be estimated. Future work will consider repeated sampling algorithms (e.g. Markov chain Monte Carlo) to simulate f^* posterior distributions [14] for the CBM and ERM acquisition functions used in Bayesian optimization.

ACKNOWLEDGMENTS

Mr. Conor Clare acknowledges the Northern Ireland Executive's Department for the Economy ('DfE') in funding this research paper.

REFERENCES

[1] Richard G. Brereton. 2015. The t-distribution and its relationship to the normal distribution. *Journal of Chemometrics* 29, 9 (2015), 481–483. https://doi.org/10.1002/cem.2713 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/cem.2713

[2] Eric Brochu, Vlad M. Cora, and Nando de Freitas. 2010. A Tutorial on Bayesian Optimization of Expensive Cost Functions, with Application to Active User Modeling and Hierarchical Reinforcement Learning. arXiv:arXiv:1012.2599

[3] Mark Alan Chilenski. 2016. *Experimental Data Analysis Techniques for Validation of Tokamak Impurity Transport Simulations.* Ph.D. Dissertation. Massachusetts Institute of Technology.

[4] Javier González, Michael Osborne, and Neil Lawrence. 2016. GLASSES: Relieving The Myopia Of Bayesian Optimisation. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics (Proceedings of Machine Learning Research)*, Vol. 51. PMLR, 790–799. http://proceedings.mlr.press/v51/gonzalez16b.html

[5] Phillip Hennig and Christian J. Schuler. 2012. Entropy Search for Information-Efficient Global Optimization. *Journal of Machine Learning Research* 13 (June 2012), 1809–1837.

[6] José Miguel Hernández-Lobato, Matthew W. Hoffman, and Zoubin Ghahramani. 2014. Predictive Entropy Search for Efficient Global Optimization of Black-box Functions. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 1 (NIPS'14)*. 918–926.

[7] José Jiménez and Josep Ginebra. 2017. pyGPGO: Bayesian Optimization for Python. *The Journal of Open Source Software* 2 (11 2017), 431. https://doi.org/10.21105/joss.00431

[8] J Mockus, Vytautas Tiesis, and Antanas Zilinskas. 1978. *The application of Bayesian methods for seeking the extremum.* Vol. 2. 117–129.

[9] Vu Nguyen and Michael A. Osborne. 2019. Knowing The What But Not The Where in Bayesian Optimization. arXiv:arXiv:1905.02685

[10] Carl. E. Rasmussen and Christopher K. I. Williams. 2006. Gaussian Processes for Machine Learning. In *Gaussian Processes for Machine Learning.* MIT Press.

[11] Amar Shah, Andrew G. Wilson, and Zoubin Ghahramani. 2013. Bayesian Optimization using Student-t Processes. *NIPS Workshop on Bayesian Optimization* (2013).

[12] Amar Shah, Andrew G. Wilson, and Zoubin Ghahramani. 2014. Student-t Processes as Alternatives to Gaussian Processes. *The Seventeenth International Conference on Artificial Intelligence and Statistics (AISTATS), 2014* (02 2014).

[13] Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P. Adams, and Nando de Freitas. 2016. Taking the Human Out of the Loop: A Review of Bayesian Optimization. *Proc. IEEE* 104, 1 (Jan 2016), 148–175. https://doi.org/10.1109/JPROC.2015.2494218

[14] Jasper Snoek, Hugo Larochelle, and Ryan P. Adams. 2012. Practical Bayesian Optimization of Machine Learning Algorithms. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 2 (NIPS'12).* 2951–2959.

[15] Niranjan Srinivas, Andreas Krause, Sham Kakade, and Matthias Seeger. 2010. Gaussian Process Optimization in the Bandit Setting: No Regret and Experimental Design. In *Proceedings of the 27th International Conference on International Conference on Machine Learning (ICML'10).* 1015–1022.

[16] Sonja Surjanovic and Derek Bingham. 2019. Virtual Library of Simulation Experiments: Test Functions and Datasets. http://www.sfu.ca/~ssurjano.

[17] Brendan Tracey and David Wolpert. 2018. Upgrading from Gaussian Processes to Student's-t Processes. 2018 AIAA Non-Deterministic Approaches Conference. https://doi.org/10.2514/6.2018-1659

[18] Jian Wu, Matthias Poloczek, Andrew G. Wilson, and Peter Frazier. 2017. Bayesian Optimization with Gradients. In *Advances in Neural Information Processing Systems 30.* 5267–5278.

[19] Antanas Zilinskas. 1992. A review of statistical models for global optimization. *Journal of Global Optimization* 2 (06 1992), 145–153. https://doi.org/10.1007/BF00122051