Learning to Optimise: Using Bayesian Deep Learning for Transfer Learning in Optimisation

Jonas Langhabel
Department of Computer Science
Technische Universität Berlin
Berlin, DE 10587
mail@langhabel.com

Jannik Wolff
Department of Computer Science
Technische Universität Berlin
Berlin, DE 10587
jannik.wolff@outlook.com

Raphaël Holca-Lamarre
Neural Information Processing Group,
Technische Universität Berlin
Berlin, DE 10587
raphael@bccn-berlin.de

Abstract

When performing global optimisation, the number of samples required to survey a solution space grows exponentially with the dimensionality of the space. This curse of dimensionality limits the use of global optimisation methods in high-dimensional problems. We propose to address these limitations through transfer learning: by allowing an algorithm to learn across multiple tasks, it should address new problems more efficiently (i.e., requiring fewer samples), thereby alleviating the curse of dimensionality for these new problems. In more details, we extend surrogate-based optimisation to learn a function approximation for both the design variables (the solution space of a single task) and the dimensions over which tasks vary (i.e., to interpolate between the solution spaces of different tasks). We use a Bayesian deep neural network for function approximation and test four methods to quantify uncertainty in the network's output: (1) the Euclidean distance between sample points, (2) the Kriging method, (3) an ensemble of networks, and (4) Monte-Carlo dropout. These uncertainty measures are used to solve the exploration-exploitation dilemma, allowing efficient global exploration of the solution space. We find that the three first methods provide a useful measure of uncertainty for exploration. Monte-Carlo dropout, however, yields a measure that is less affected by the density of sample points in the solution space, and it is thus less well suited to guide exploration. We test our algorithm on one-dimensional optimisation problems that are related along a second dimension. As learning progresses, the interpolation learned by the network allows the optimisation method to concentrate its search on promising regions of the solution spaces of new problems. The optimisation then requires fewer samples to converge to the optima. In future work, we intend to examine whether this approach can be used to circumvent the curse of dimensionality in higher-dimensional problems.

Workshop on Bayesian Deep Learning, NIPS 2016, Barcelona, Spain.

In the context of optimisation, searching for globally optimal solutions in high-dimensional spaces is challenging. The number of samples required to survey a solution space grows exponentially with the dimensionality of the space, a problem known as the curse of dimensionality. This curse limits global exploration to relatively small solution spaces. Some gradient-based methods scale well with dimensionality (for instance, the adjoint state method), but these methods converge to local optima. Current optimisation algorithms are therefore limited to either searching low-dimensional spaces for global optima or large spaces for local optima.

We propose to address these limitations through transfer learning. In short, we make use of a deep neural network to interpolate between the solution spaces of related optimisation problems. Based on this interpolation, the algorithm concentrates its search on promising regions of the solution space when faced with a new optimisation problem. The algorithm then requires fewer samples to find the optimum, which could be used to circumvent the curse of dimensionality. In this extended abstract, we present preliminary results we obtained with this approach.

In more details, our approach is an extension of surrogate-based optimisation. Surrogate-based optimisation is a metaheuristic that learns an approximation for the function to be optimised and explores the design space according to this surrogate function. This method typically makes use of probabilistic estimates to quantify uncertainty in the value of the surrogate function to solve the exploration-exploitation dilemma. That is, in order to efficiently explore the solution space, new sample points are taken so as to balance high uncertainty in the function (exploration) and high function value (exploitation). This method allows for the global exploration of a space while usually requiring fewer function evaluations than other algorithms.

We extend this method to learn function approximations over multiple related optimisation problems. Specifically, we increase the dimensionality of the surrogate model to include the dimensions over which the different problem instances vary. For example, consider a one-dimensional problem consisting of optimising the length of the wing of an aircraft. Furthermore, assume that the problem is repeated for multiple aircrafts flying at different speeds (e.g., a commercial plane, a propeller plane, etc.). In traditional surrogate-based optimisation, each problem is treated separately: a distinct one-dimensional approximation of the wing length value function is learned for every plane. In the present algorithm, we extend the dimensionality of the surrogate model to include the variables over which the problem instances differ (in this case, the flying speed). In the current example, the regressor models how both the wing length and the flying speed affect the value function. For a problem at new flying speed, the regressor interpolates the value of the function from prior problems, thereby exhibiting a knowledge of the solution space before any samples are taken.

Surrogate-based optimisation relies on probabilistic regression methods to learn an approximation of the target function and compute uncertainty in this approximation. In this work, we explore the use of a deep neural network for function approximation. We compare four methods to quantify uncertainty in the network's output: (1) the Euclidean distance between sample points, (2) the Kriging method [1], (3) an ensemble of networks [2], and (4) Monte-Carlo dropout [3] (see Fig. 1). We find that the Euclidean distance, the Kriging, and the ensemble of networks methods all provide useful measures of uncertainty to solve the exploration-exploitation dilemma (high uncertainty away from sample points). On the other hand, Monte-Carlo dropout provides high uncertainty estimates where the function exhibits large fluctuations in its value, with less regard to the density of sample points. This method therefore appears less well suited to guide the exploration of a space. The ensemble of networks method is by far the less computationally efficient; simple measures (such as the Euclidean distance between sample points) provide comparable performances while being at least an order of magnitude faster to compute.

We test our learning algorithm on a 1-dimensional artificial optimisation problem that may be understood as the aircraft wing-length problem presented above (Fig. 2). We find that the neural network regressor is able to generalise between optimisation problems, leading to a progressive decrease in the number of samples required to find the function's optimum . These results indicate that, in a simple case, learning from prior problems can improve the convergence of optimisation methods. The question remains open as whether this approach can be scaled up and used to circumvent the curse of dimensionality in real-world scenarios.

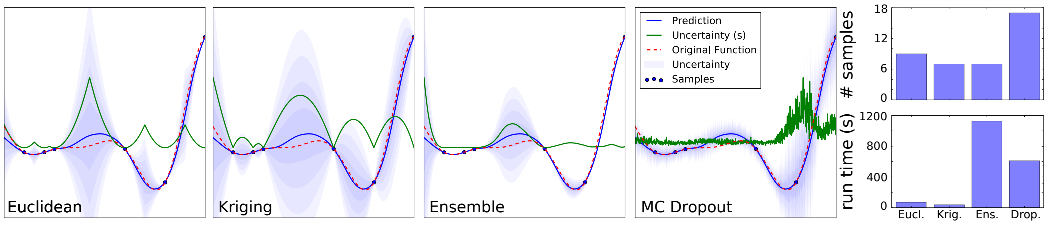

Figure 1: Uncertainty quantification. We compare four methods to quantify uncertainty in the output of the neural network. We use these measure of uncertainty to explore the solution space of a 1-dimensional optimisation problem. For each method, we report the number of samples required to reach the function's optimum and the wall clock run time.

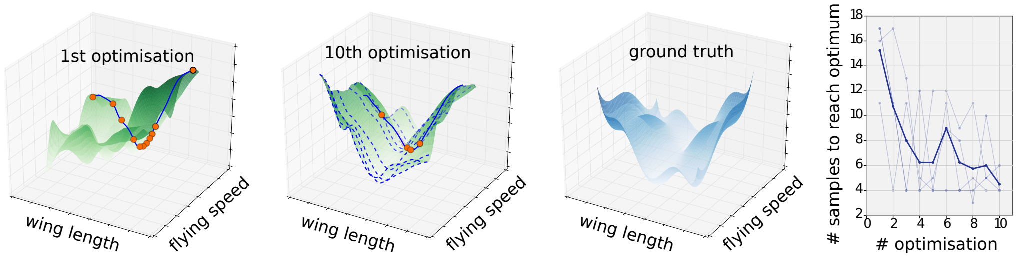

Figure 2: Learning to optimise. Preliminary results obtained by the algorithm on artificial data. The optimisation problem has only one design variable ("wing length"); the problem instances vary along a second dimension ("flying speed"). The blue surface represents the true value function, the green surfaces the approximation learned by the network, the blue traces the optimisations solved for different flying speeds, and the orange points samples of the value function taken for the current problem. As learning progresses, the algorithm requires fewer samples to find the optimum.

References

1. Oliver, M. A. & Webster, R. Kriging: a method of interpolation for geographical information systems. *International Journal of Geographical Information System* **4,** 313–332 (1990).
2. Jiang, Y. *Uncertainty in the output of artificial neural networks* in *2007 International Joint Conference on Neural Networks* (2007), 2551–2556.
3. Gal, Y. & Ghahramani, Z. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. *arXiv preprint arXiv:1506.02142* (2015).