The handle [http://hdl.handle.net/1887/65632](http://hdl.handle.net/1887/65632) holds various files of this Leiden University dissertation.

**Author:** Stein, B. van  
**Title:** Data driven modeling & optimization of industrial processes  
**Issue Date:** 2018-09-20
The Kriging model, as stated in Chapter 5, is heavily exploited by the *Efficient Global Optimization* framework, a popular optimization algorithm for optimizing expensive functions, such as the production of high quality steel in the PROMI-MOOC project.

Applying Efficient Global Optimization with Cluster Kriging is one solution in solving the time and space complexity bottleneck of standard Kriging while using the Global Optimization algorithm. However, many data sets do not obey the assumptions that the Kriging model is based on. Other machine learning techniques such as Artificial Neural Networks, Random Forests and Support Vector Machines might perform much better as surrogate models. Unfortunately, these techniques do not provide a prediction variance out of the box.

In this chapter a novel uncertainty measure is proposed to allow the use of these different data driven models in the *Efficient Global Optimization* framework. With the help of this measure, global optimization with complex deep neural networks, as well as other machine learning techniques such as Random Forests, as the surrogate model, becomes feasible.
In this chapter the challenges of applying global optimization on arbitrary machine learning models is discussed. In order to use the models in the model-based optimization procedure of the framework (Section 2.3), an uncertainty measure, the \(k\)-NN uncertainty is presented. It is shown that the \(k\)-NN uncertainty allows for combinations of EGO and models that outperform the classical EGO using Kriging.

This chapter is primarily based on the publication [100].
6.1 Background

As discussed in Chapter 2, the main objective of the industrial partner Tata Steel, is to classify surface defects on incoming steel slabs and optimize process parameters to reduce the number of defects occurring. To achieve this objective within the framework presented in Section 2.3, global optimization algorithms like EGO have to be applied using a large scale of different predictive models as surrogate model. Using different surrogate models can significantly improve the accuracy of the final predictive model and therefore greatly increase the performance of the EGO algorithm.

However, there is one major issue, the infill-criteria of EGO requires a prediction uncertainty or prediction variance, which most statistical models do not provide out of the box. To overcome this issue, a novel uncertainty measure is introduced. This measure allows us to apply EGO in combination with any predictive model such as deep neural networks that are capable of predicting surface defects on the Tata Steel data set.

6.2 kNN Uncertainty Measure for EGO

For most regression models, their overall accuracy can be estimated with help of various error measures. However, in some applications it is important to provide not only point predictions, but also to estimate the “uncertainty” of the prediction, e.g., in terms of confidence intervals, variances, or interquartile ranges. There are very few statistical modeling techniques able to achieve this. For instance, the Kriging method is equipped with a theoretical mean squared error. In this section we address this problem by introducing a heuristic method to estimate the uncertainty of the prediction, based on the error information from the \( k \)-nearest neighbors. This heuristic, called the \( k \)-\( NN \) uncertainty measure, is computationally much cheaper than other approaches (e.g., bootstrapping) and can be applied regardless of the underlying regression model. To validate and demonstrate the usefulness of the proposed heuristic, it is combined with various models...
and plugged into the well-known EGO algorithm. This allows for the usage of regression models other than the Kriging model.

Such a measure of the prediction uncertainty should aim at the following objectives: 1) It should operate independently of the modeling assumptions. 2) It should be exploitable by the Efficient Global Optimization algorithm, making it possible to use any regression model in the EGO framework.

In the nonparametric settings, when estimating the mean squared error of the predictor, the available information are the data set \( D := \{ \mathbf{x}, y \} \) and the prediction \( \hat{f}(\mathbf{x}) \) at \( \mathbf{x} \). Intuitively, this empirical uncertainty measure should be zero at correctly predicted known observations and increase for the data points that are far from the observations. Given these preferred properties, a distance-weighted measure \( \hat{U}_{k,NN} \) is proposed as follows:

\[
\hat{U}_{k,NN} = \sum_{i \in N(\mathbf{x})} w_i^k \left| \hat{f}(\mathbf{x}) - y_i \right| + \min_{i \in N(\mathbf{x})} d(\mathbf{x}_i, \mathbf{x}) \frac{\max_{x_i, x_j \in \mathbf{x}} d(\mathbf{x}_i, \mathbf{x}_j)}{\sigma},
\]

(6.1)

\[
w_i = 1 - \frac{d(\mathbf{x}_i, \mathbf{x})}{\sum_{i \in N(\mathbf{x})} d(\mathbf{x}_i, \mathbf{x})}, \quad \hat{\sigma} = \sqrt{\text{Var} \left[ \{ y_i \}_{i \in N(\mathbf{x})} \cup \{ \hat{f}(\mathbf{x}) \} \right]}.
\]

Note that \( N(\mathbf{x}) \) collects the indices of \( k \)-nearest neighbors to \( \mathbf{x} \) and \( d(\cdot, \cdot) \) denotes the Euclidean distance metric. \( \hat{\sigma} \) is computed as the standard deviation of the observations in the neighborhood with the prediction \( \hat{f}(\mathbf{x}) \).

The proposed uncertainty quantification consists of two components: 1) the empirical error of the prediction and 2) the variability of the observed outputs \( y \). Intuitively, less empirical prediction error leads to higher certainty of the prediction. Moreover, when comparing two different regression tasks, a large variability of the observations \( y \) could contribute to the high uncertainty of the prediction, even if the predictor \( \hat{f} \) were making the same empirical error on both tasks. The empirical error is computed from the difference between the prediction \( \hat{f}(\mathbf{x}) \) and the observations at the \( k \)-nearest neighbors. Such differences are linearly scaled
where the weights are inversely proportional to Euclidean distances to the neighbors. This heuristic is based on the intuition that the closer neighbors have more influence than neighbors that are further away. To quickly diminish the effect of far-away neighbors, the exponent \( k \) (the number of neighbors) is applied on the weights. The variability of the observation is estimated by calculating the standard deviation of the observations at the nearest neighbors and the predicted point. The resulting value is then rescaled by the distance to the nearest neighbor. Using the distances to scale the heuristic error prediction, we make sure that the uncertainty goes to zero at correctly predicted known points and that it increases when predicting points further away from the known observations.

A good number of neighbors is depending on the number of known points and the data dimensionality. For most of the experiments \( k \) is set to 20, more neighbors will provide a more smooth but also slightly more pessimistic prediction error while less neighbors make the expected prediction error more optimistic and less smooth.

To illustrate the behaviour of \( k \)-NN uncertainty, a 1-D function \( f(x) = x \sin(x) \) is used in Figure 6.1. Note that \( k \)-NN uncertainty (green area) progresses very similarly to the Kriging uncertainty quantification (blue area). It can also be observed that at the known observations the prediction error given by the \( k \)-NN uncertainty algorithm is exactly the error between the prediction and the known observation. Note that the SVR model is badly fitted, and different hyperparameters would result in a much better fit. This is on purpose to illustrate how the \( k \)-NN uncertainty would look like using less fitted models. Lastly, it can be observed that when the to be predicted point is far away from the known data points the uncertainty of the prediction increases.

### 6.3 Experimental Setup

Two different experimental setups are used to demonstrate the properties and effectiveness of \( k \)-NN uncertainty in Efficient Global Optimization. First, we validate \( k \)-NN uncertainty by visual inspection of plotted two and five dimensional benchmark functions that are often used in the field of optimization (in this case
Figure 6.1: Best viewed in color. Visualization of $k$-NN uncertainty heuristic. The dotted red line is the real function $f(x) = x \sin(x)$, the red dots are the observed points. The blue line is the predicted mean of the Kriging model with the shaded blue area showing the standard Kriging variance and the shaded green area is $k$-NN uncertainty on the same Kriging model. The yellow line shows the predictions of a Support Vector Regression (SVR) model with default hyper-parameters ($C = 1$, RBF kernel, $\epsilon = 0.1$) with the shaded yellow area denoting $k$-NN uncertainty using the SVR model. The number of neighbors for $k$-NN uncertainty is set to 4 in this case.

the Ackley and Schaffer function). In Figure 6.2 it can be observed that $k$-NN uncertainty is quite similar to the Kriging variance as shown in the lower subplots of Figure 6.2a. $k$-NN uncertainty is a bit less optimistic than the Kriging variance but shows roughly the same areas with higher variance. When looking at the Random Forest bootstrapping variance and $k$-NN uncertainty it can be observed that the bootstrapping variance is very blocky, due to the Random Forest model assumptions. $k$-NN uncertainty however does not use the individual tree predictions of the Random Forest model and because of the interpolation effect using the distances to the known observations, it creates a much more smooth surface. In Figure 6.3 the models are trained on samples of the Schaffer function in the space of $-50$ to 50 for both dimensions while tested on the complete range of $-100$ to 100.
to 100. It can be observed that the $k$-NN uncertainty gradually increases when moving away from the known observations, while the Kriging variance almost immediately explodes to a flat high value.

When looking at the same benchmark function but now in five dimensions, we can plot a one-dimensional slice of the function (using the first dimension) to show the local behaviour of $k$-NN uncertainty versus the Kriging variance in Figure 6.4. Here it can be observed that the Kriging variance is over-optimistic and actually wrong, while $k$-NN uncertainty is much more pessimistic and actually captures very nicely the shape of the underlying function.

![Image of plots](image)

**Figure 6.2:** a) Upper-left plot is the Ackley function in two dimensions, upper-right shows the Kriging prediction of this function using 100 data points for training. Lower-left plot shows the Kriging variance and bottom-right shows $k$-NN uncertainty using the same Kriging model. b) Upper-left plot is the same as a), upper-right shows a Random Forest predictor with 50 trees using 100 data points for training. Lower-left plot shows the variance given by the tree regressors of the Random Forest and bottom-right shows $k$-NN uncertainty using the same Random Forest model. The number of nearest neighbors for $k$-NN uncertainty is set to 20.

The second experiment is more quantitative as we compare the performance of $k$-NN uncertainty in the setting of Efficient Global Optimization. We compare
6. ARBITRARY MODEL EFFICIENT GLOBAL OPTIMIZATION

(a) Kriging variance versus \(k\)-NN uncertainty

(b) Random Forest bootstrapping variance versus \(k\)-NN uncertainty

**Figure 6.3:** Same as in Figure 6.2 but now using the *Schaffer* benchmark function.

**Figure 6.4:** The green dots are unseen observations of the *Ackley* function in five dimensions (slice with the last four dimensions set at zero), the blue line is the predicted mean of a Kriging model, the blue shaded area is the Kriging variance and the green shaded area \(k\)-NN uncertainty with 20 nearest neighbors using the same Kriging model.
the convergence speed of the original EGO with Kriging, EGO with Kriging using $k$-NN uncertainty instead of the Kriging variance, EGO with a Random Forest model using $k$-NN uncertainty as the prediction variance and finally EGO with a multi-layer perceptron using $k$-NN uncertainty (using two hidden layers of size 100 and 50 nodes respectively).

The experiment is carried out using three different benchmark functions Ackley, Rastrigin and Schaffer with implementations from the DEAP [90] python package. For each function, the experiments are repeated using 100 initial samples using a Latin hypercube sampling strategy, and in 2, 5 and 10 dimensions ($d$). The EGO algorithm is run for $10 \cdot d$ evaluations and each experiment is repeated 40 times. The number of nearest neighbors for $k$-NN uncertainty is set to 20.

From Figure 6.5 we can observe that in most cases the convergence is very similar for all four EGO setups. In the two dimensional cases the Random Forest setup seems to be slightly worse performing than the Kriging setups, on the other hand, in the ten dimensional cases the Random Forest setup seems to outperform the Kriging setups. For the Kriging models, the original Kriging variance seems to perform slightly better than the heuristic $k$-NN uncertainty, however in most cases this is only marginal. Interesting is to note the performance of the neural network using $k$-NN uncertainty, which performs very well and even outperforms the standard EGO procedure with Kriging in the five dimensional cases. Further investigation showed us that the neural network fits the underlying global trend of the function much more accurate than the Kriging or Random Forest model, allowing the EGO procedure to quickly converge to the global optimum.

6.4 Conclusion

An uncertainty quantification measure, the $k$-NN uncertainty measure is proposed. The proposed heuristic works independently of the modeling assumptions and can therefore be used in combination with any regression model. It is shown that the heuristic function obeys the preferred properties: 1) it ensures exactitude; on known observations a correct prediction gives zero prediction variance. 2) it
Figure 6.5: Convergence on the three benchmark functions. Displaying, as blue dots, the original EGO procedure with Kriging variance, the red stars show the convergence of EGO with a Random Forest model using $k$-NN uncertainty, the green diamonds illustrate the convergence of EGO with Kriging using $k$-NN uncertainty and the yellow triangles illustrate the convergence of EGO with a multi-layer perceptron using $k$-NN uncertainty. The shaded areas show the 95% confidence interval over 40 runs.
increases with the dispersion of the known observations. 3) It is exactly the prediction error when predicting known data points. The behaviour of the $k$-NN uncertainty is verified by plotting the surface of several predictors on two benchmark functions and by running a wide set of experiments using the Efficient Global Optimization framework. Results of the EGO experiments show that the heuristic can be used in such optimization settings and that the performance in both high and low dimensions, using different statistical models, can even outperform the original EGO concept that uses Kriging. It also shows that different regression models can be used in Efficient Global Optimization using such a heuristic as prediction variance, making EGO more widely applicable. It is shown that the proposed heuristic is robust with respect of its parameter $k$, the number of neighbors, and a recommendation of $k = 20$ is given.