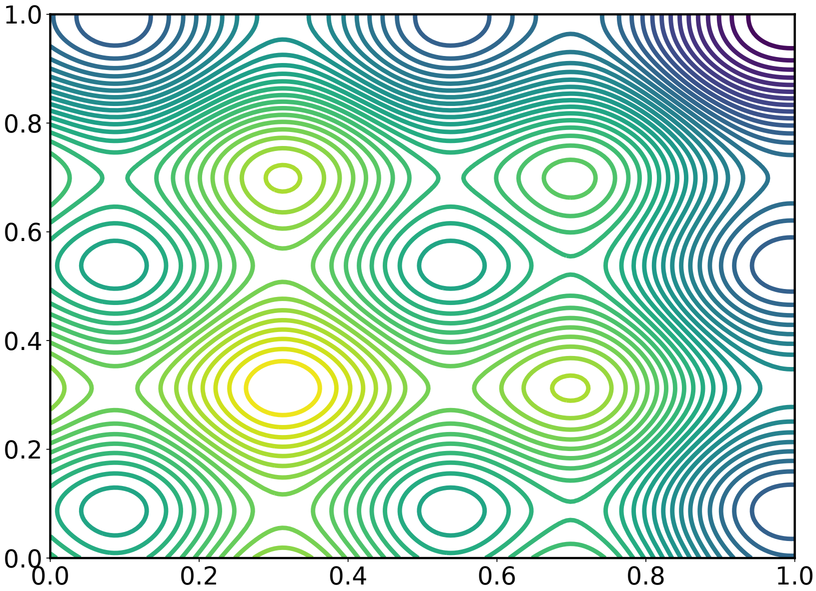
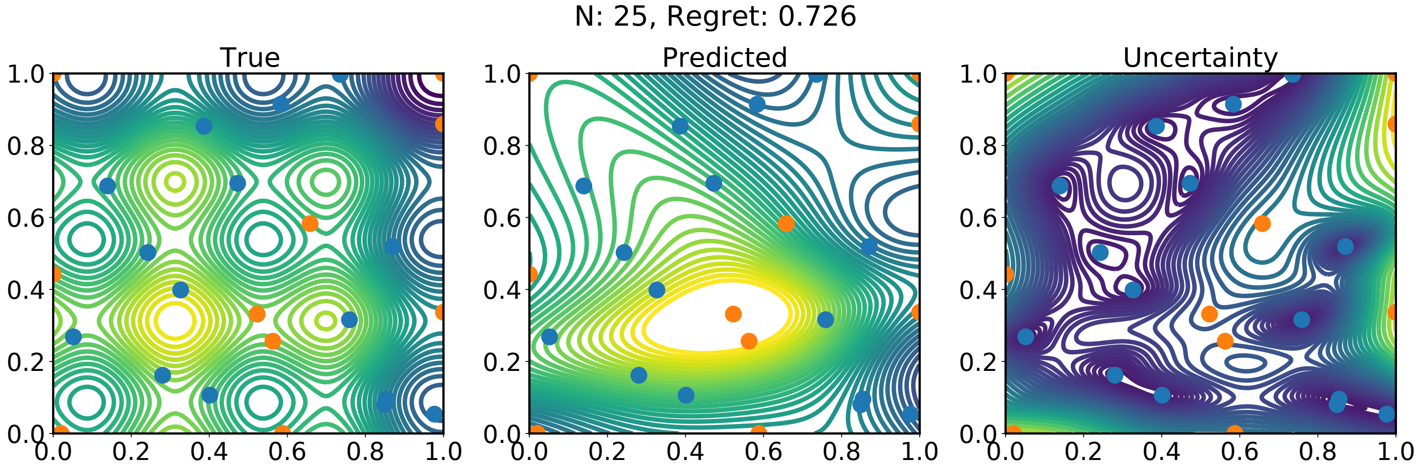
Batch Bayesian optimization example

As an example, consider the following function to optimize,

$$f(x,y) = 1 - (u^2 + v^2 - 0.3\cos(3\pi u) - 0.3\cos(3\pi v)) \quad u = 1.6x - .5 \quad v = 1.6y - .5 \tag{1}$$

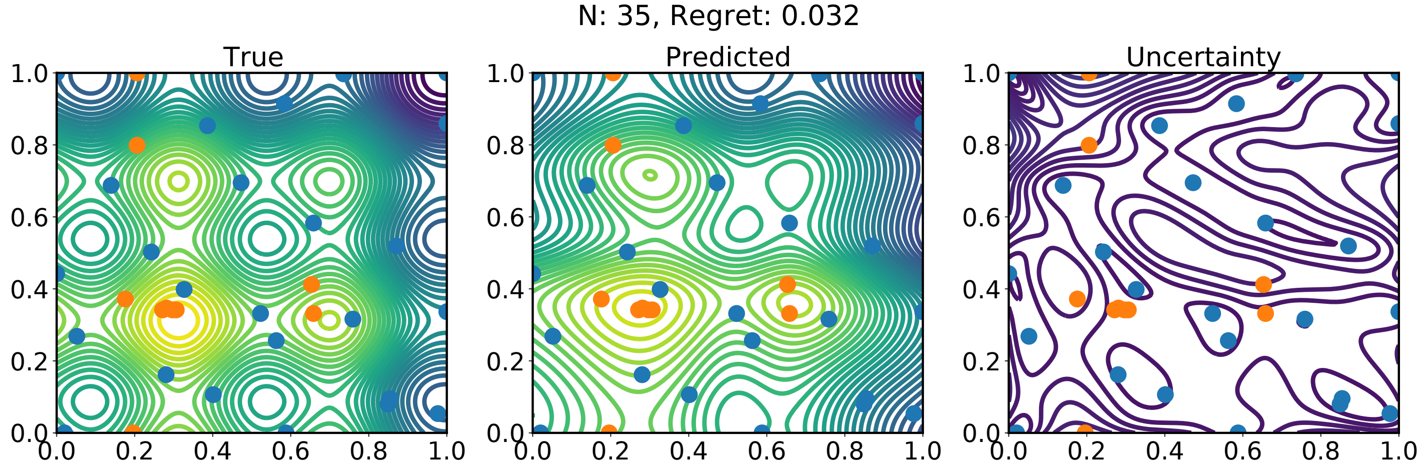


An initial set of 15 points was randomly selected in $[0,1]^2$ and used to train a neural network to predict the objective. Training data are shown as blue dots and model suggestions for the next experiment are shown as ten orange dots. The regret is the difference between the best possible objective value subtracted by the best value found in the proposed data.

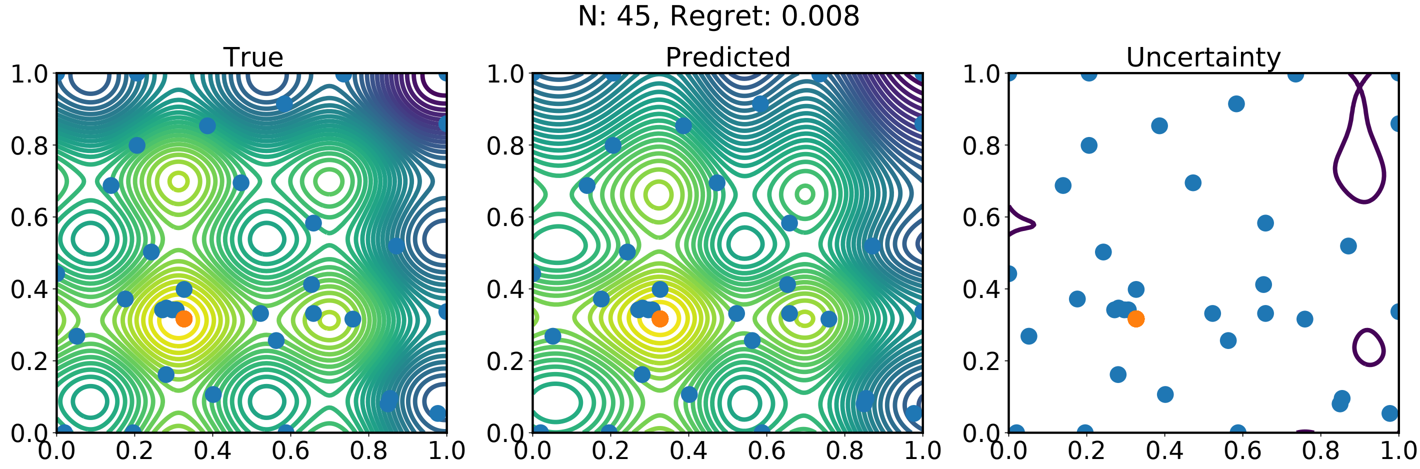


The proposed experimental design includes 10 conditions (orange dots) that exploit and explore the design space. In this initial design, there are several orange dots that occupy the edges/corners of the design space (exploration) and a few orange dots that were sampled where the model predicted higher values for the objective function (exploitation).

All of the orange dots and previously used blue dots were used to assemble a new training set that was used to re-fit the model. In next experimental design, a new set of ten orange dots was proposed. Now that the model is more confident about the edges of the design space, the next experimental design includes many more orange dots in areas where the model expects higher values for the objective function.



Once updated with data from the second experimental design, the model is now very confident in the location of the maximum value of the objective function so the final design does not explore beyond that point.



Method

Prediction uncertainty for nonlinear parametric models

Given variables, $\mathbf{x}$, we wish to predict the distribution of an outcome y conditioned on previous observations, $\mathcal{D} = \{\mathbf{x}_i, y_i\}$. In other words, we want $p(y|\mathbf{x}, \mathcal{D})$. Using a model that depends on parameters θ, the density function can be expanded and rewritten as

$$p(y|\mathbf{x}, \mathcal{D}) = \int_\theta p(y, \theta|\mathbf{x}, \mathcal{D})\mathrm{d}\theta = \int_\theta p(y|\mathbf{x}, \theta)p(\theta|\mathcal{D})\mathrm{d}\theta. \tag{2}$$

The assumption that outcomes can be predicted by a model $f(\mathbf{x}, \theta)$ provides an expression for the data independent density of y,

$$y = f(\mathbf{x}, \theta) + \varepsilon, \qquad \varepsilon \sim \mathcal{N}(0, \sigma^2) \tag{3}$$

$$p(y|\mathbf{x}, \theta) = \mathcal{N}(y|f(\mathbf{x}, \theta), \sigma^2) \tag{4}$$

where ε is a random variable that models measurement noise and σ^2 is a model hyper-parameter. Bayes' theorem is used to estimate the density of model parameters given data,

$$p(\theta|\mathcal{D}) = \frac{p(\mathcal{D}|\theta)p(\theta)}{p(\mathcal{D})} \tag{5}$$

where the likelihood, $p(\mathcal{D}|\theta)$, is defined using equation 4,

$$p(\mathcal{D}|\theta) = \prod_i \mathcal{N}(y_i|f(\mathbf{x}_i), \sigma^2). \tag{6}$$

and the prior, $p(\theta)$, is assumed to be an isotropic Gaussian

$$p(\theta) = \mathcal{N}(\mathbf{0}, \alpha^{-1}\mathbb{I}) \tag{7}$$

where α is also a hyper-parameter. The Laplace approximation uses a second order Taylor series expansion to approximate the unknown density $p(\theta|\mathcal{D})$. This approximation results in a Gaussian centered at a mode of $p(\theta|\mathcal{D})$ and with a covariance matrix equal to the inverse of the matrix of second derivatives of the negative log of $p(\theta|\mathcal{D})$,

$$p(\theta|\mathcal{D}) \approx \mathcal{N}(\theta|\hat{\theta}, \mathbf{H}^{-1}) \tag{8}$$

where

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}} \ln p(\theta|\mathcal{D}) \tag{9}$$

and

$$\mathbf{H} = \nabla_\theta \nabla_\theta \; - \ln \; p(\theta|\mathcal{D})\big|_{\theta=\hat{\theta}}\,. \tag{10}$$

Once values for $\hat{\theta}$ and $\mathbf{H}$ are obtained, model predictions are evaluated using equations 2, 4, and 8:

$$p(y|\mathbf{x}, \mathcal{D}) = \int_\theta p(y|\mathbf{x}, \theta)p(\theta|\mathcal{D})\mathrm{d}\theta \approx \int_\theta \mathcal{N}(y|f(\mathbf{x}, \theta), \sigma^2)\mathcal{N}(\theta|\hat{\theta}, \mathbf{H}^{-1})\mathrm{d}\theta \tag{11}$$

Linearizing $f(\mathbf{x}, \theta)$ about $\hat{\theta}$ allows for the following approximations of the expected value and variance of y given $\mathbf{x}$, where expectations are taken with respect to the parameter posterior

$$p(y|\mathbf{x}, \mathcal{D}) \approx \mathcal{N}(y|f(\mathbf{x}, \hat{\theta}), \sigma^2(\mathbf{x}, \hat{\theta})) \tag{12}$$

where

$$\sigma^2(\mathbf{x}, \hat{\theta}, \mathbf{H}^{-1}) \approx \sigma^2 + \mathbf{g}(\mathbf{x}, \hat{\theta})^T\mathbf{H}^{-1}\mathbf{g}(\mathbf{x}, \hat{\theta}), \qquad \mathbf{g}(\mathbf{x}, \hat{\theta}) = \nabla_\theta \, f(\mathbf{x}, \theta)\big|_{\theta=\hat{\theta}} \tag{13}$$

Approximation of Hessian inverse

The proposed batch Bayesian optimization algorithm is based on the following approximation of the Hessian and specifically its inverse. Evaluating equation 10 gives

$$\begin{aligned} \mathbf{H} &= \alpha\mathbb{I} + \nabla_\theta \nabla_\theta \sum_i \sigma^{-2}(y_i - f(\mathbf{x}_i, \theta))^2 \\ &\approx \alpha\mathbb{I} + \sigma^{-2} \sum_i \mathbf{g}(\mathbf{x}_i, \theta)\mathbf{g}(\mathbf{x}_i, \theta)^T \end{aligned} \tag{14}$$

which is the outer-product approximation of the Hessian. To evaluate the inverse of the outer-product approximated Hessian, the Woodbury identity can be used to give

$$\mathbf{H}_{l+1}^{-1} = \mathbf{H}_l^{-1} - \frac{\mathbf{H}_l^{-1}\mathbf{g}(\mathbf{x}_{l+1}, \theta)\mathbf{g}^T(\mathbf{x}_{l+1}, \theta)\mathbf{H}_l^{-1}}{\sigma^2 + \mathbf{g}^T(\mathbf{x}_{l+1}, \theta)\mathbf{H}_l^{-1}\mathbf{g}(\mathbf{x}_{l+1}, \theta)} \tag{15}$$

which is evaluated sequentially with $\mathbf{H}_0^{-1} = 1/\alpha\mathbf{I}$. Useful features of this equation are that it provides a way to compute the inverse Hessian using a single pass through the training data and that it provides a way to update the inverse Hessian for any input, $\mathbf{x}_{l+1}$, without requiring y_{l+1}!

Bayesian optimization and maximum expected improvement

A common acquisition function for Bayesian optimization of y with respect to $\mathbf{x}$ is the Expeceted Improvement from the best previously observed value, y^*, defined as

$$\begin{aligned} a(\mathbf{x}, \hat{\theta}, \mathbf{H}^{-1}) &= \mathbb{E}_y[\max(0, y - y^*)] \\ &= (f(\mathbf{x}, \hat{\theta}) - y^*)\Phi(f(\mathbf{x}, \hat{\theta})|y^*, \sigma^2(\mathbf{x}, \hat{\theta}, \mathbf{H}^{-1})) + \sigma(\mathbf{x}, \hat{\theta}, \mathbf{H}^{-1})\mathcal{N}(f(\mathbf{x}, \hat{\theta})|y^*, \sigma^2(\mathbf{x}, \hat{\theta}, \mathbf{H}^{-1})). \end{aligned} \tag{16}$$

Batch Bayesian optimization algorithm

The proposed batch Bayesian optimization algorithm iterates between maximizing the Expected Improvement (equation 16) acquisition function to seek an experimental condition $\mathbf{x}_{l+1}$, followed by updating the parameter covariance using equation 15 with $\mathbf{x}_{l+1}$. Updating the parameter covariance, $\mathbf{H}^{-1}$, affects the computation of the variance, given by equation 13. As a result, maximizing the Expected Improvement after conditioning on $\mathbf{x}_{l+1}$ will result in a new $\mathbf{x}_{l+2}$.

Algorithm 1 Select optimal batch of experimental conditions

```
Require: $f$, $\hat{\theta}$, $\mathbf{H}^{-1}$, $N$
  $n \leftarrow 0$
  $\mathcal{X} \leftarrow \emptyset$
  while $n < N$ do
    {Select experimental condition (maximize equation 16)}
    $\mathbf{x}_n = \underset{\mathbf{x}}{\text{argmax}} \quad a(\mathbf{x}, \hat{\theta}, \mathbf{H}^{-1})$
    {Update model parameter covariance (equation 15)}
    $\mathbf{H}^{-1} \leftarrow \mathbf{H}^{-1} - \frac{\mathbf{H}^{-1}\mathbf{g}(\mathbf{x}_n, \hat{\theta})\mathbf{g}^T(\mathbf{x}_n, \hat{\theta})\mathbf{H}^{-1}}{\sigma^2 + \mathbf{g}^T(\mathbf{x}_n, \hat{\theta})\mathbf{H}^{-1}\mathbf{g}(\mathbf{x}_n, \hat{\theta})}$
    {Add $\mathbf{x}_n$ to set of experimental conditions}
    $\mathcal{X} \leftarrow \mathcal{X} \cup \mathbf{x}_n$
    $n \leftarrow n + 1$
  end while
```