Information Sources with Combined Outputs

1 Mathematical Formulation

We consider an optimization problem that combines multiple independent information sources,

$$\max_{x \in A} g\left(x\right) := \max_{x \in A} g\left(f_1\left(x\right), \ldots, f_k\left(x\right)\right)$$

where A is the feasible compact set and $f_1, \ldots, f_k$ are continuous functions (see Figure 1).

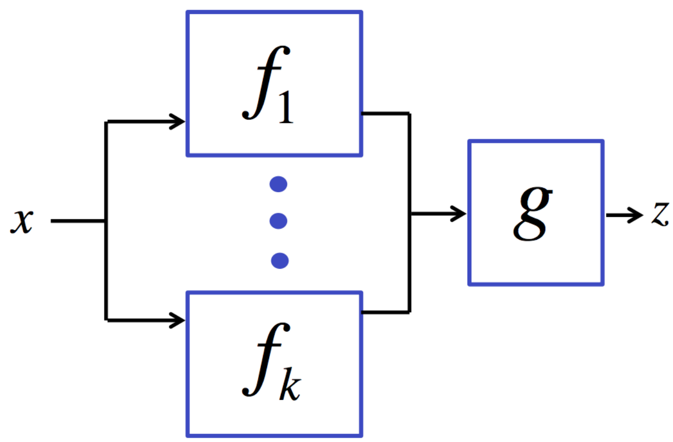

Figure 1: Diagram of the problem

2 Applications

I Machine Learning.

We have many terabytes of training data $\{(x_i, y_i)\}_{i=1}^{N}$ where $\{x_i\}_{i=1}^{N}$ are the inputs and $\{y_i\}_{i=1}^{N}$ are the outputs.

Training Machine Learning Models We have a machine learning model (e.g. logistic regression) that depends on a vector of parameters α. Our data is spread across k disks.

We want to choose α that maximizes the log-likelihood

$$\begin{aligned} g\left(\alpha\right) &= \sum_{s=1}^{k} \sum_{i \in \mathrm{disk}_s} \log p\left(y_i \mid x_i, \alpha\right) \\ &= \sum_{s=1}^{k} f_s\left(\alpha\right) \end{aligned}$$

where

$$f_s(\alpha) = \sum_{i \in \text{disk}_s} \log p(y_i \mid x_i, \alpha)$$

Cross-Validation We have a set of models $f(x, \alpha)$ indexed by the parameter α (e.g. maximum depth of decision trees). In K-Fold cross validation, we split the data into K equally sized sets. We denote by $\hat{f}^{-k}(x)$ the fitted function, estimated with the kth set removed. We want to choose α that minimizes

$$\text{CV}\left(\hat{f}, \alpha\right) = \frac{1}{N} \sum_{i=1}^{N} L\left(y_i, \hat{f}^{-k(i)}(x_i, \alpha)\right)$$

where L is the loss function and $k(i)$ is the set where x_i is for $i \in \{1, \dots, N\}$. Here each $L\left(y_i, \hat{f}^{-k(i)}(x_i, \alpha)\right)$ is an information source.

II Simulation Optimization

We want to solve

$$\max_x \mathbb{E}[f(x, \omega)]$$

where f is a stochastic simulator, ω is the randomness with a known probability distribution p.

We define the information sources as $f_s(x) := f(x, \omega_s)$ and

$$\begin{aligned} g(x) &= g(f_1(x), \dots, f_k(x), \dots) \\ &= \sum_s p(w_s) f_s(x) \end{aligned}$$

if ω takes countable values, $\omega_1, \dots, \omega_k, \dots$.

If $\omega \in C$ takes uncountable infinite values, then

$$g(x) = \int_w p(\omega) f_s(x, \omega) d\omega.$$

.

3 Value of Information Functions

We place Gaussian processes on $f_1, \dots, f_k$. Depending on the problem and the kernels of the Gaussian processes, we may have a Gaussian process on g.

We define the value of information functions as

$$V_n(x, h) = \mathbb{E}[\max_z a_{n+1}(z) - \max_z a_n(z) \mid x_{n+1} = x, h(x)]$$

where $h \in \{f_1, \dots, f_k\}$.

4 Algorithm

1. (First stage of samples) Use maximum likelihood or maximum a posteriori estimation to fit the parameters $(\mu_0^1, \Sigma_0^1), (\mu_0^2, \Sigma_0^2), \dots, \left(\mu_0^k, \Sigma_0^k\right))$ of the GP prior on $f_1, \dots, f_k$, respectively. We denote the parameters of the GP on g by a_0, b_0.

2. (Main stage of samples) For $n \leftarrow 0$ to N do

   (a) Update our joint Gaussian process posterior on g using all samples by time n.

   (b) Solve $(x_{n+1}, h_{n+1}) \in \arg\max_{x \in A, h \in \{f_1,\ldots,f_k\}} V_n(x,h)$.

   (c) Evaluate $h_{n+1}(x_{n+1})$

3. Return $x^* = \arg\max_x a_{N+1}(x)$.