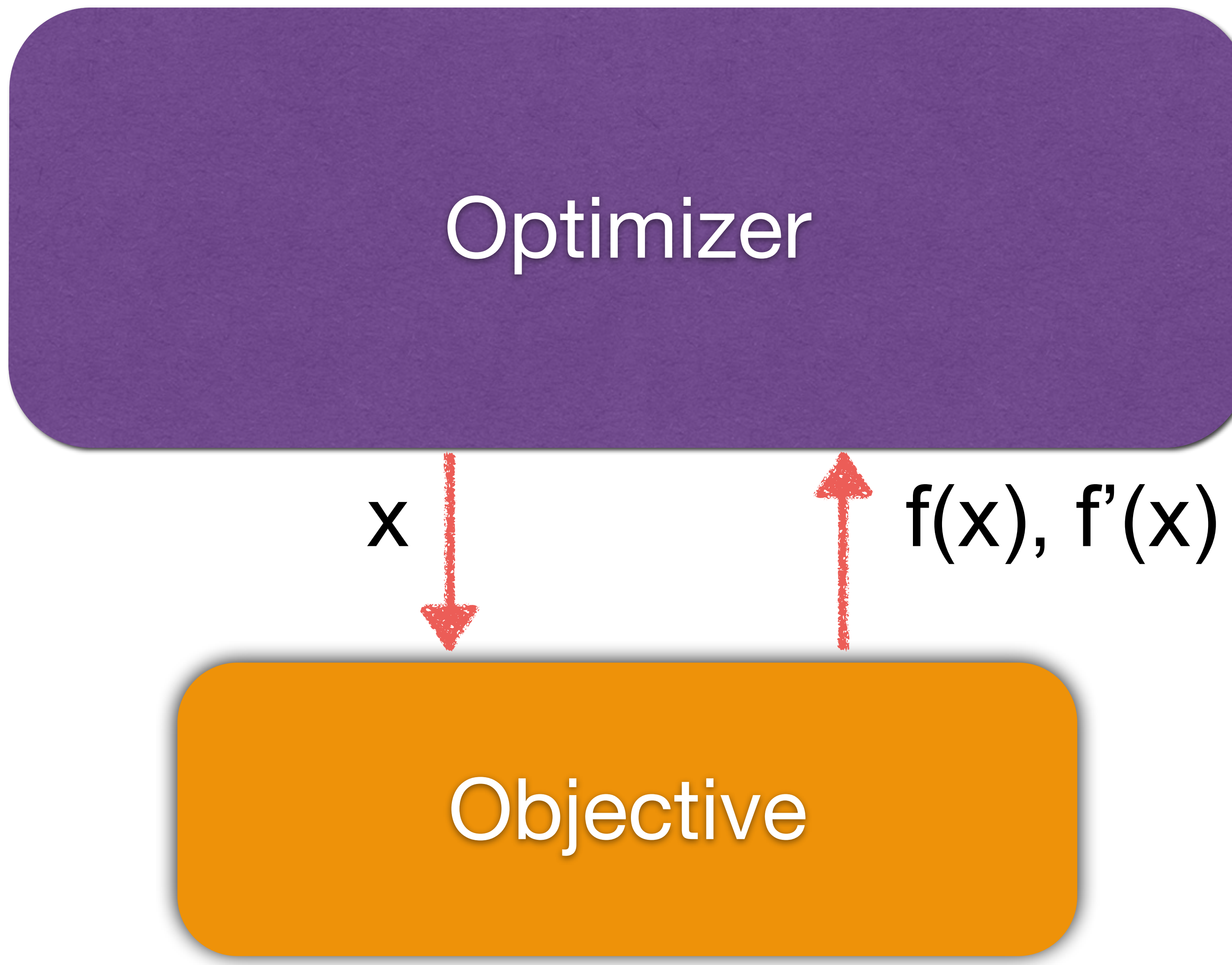
Surrogate Assisted Tuning (SAT) Quickstart

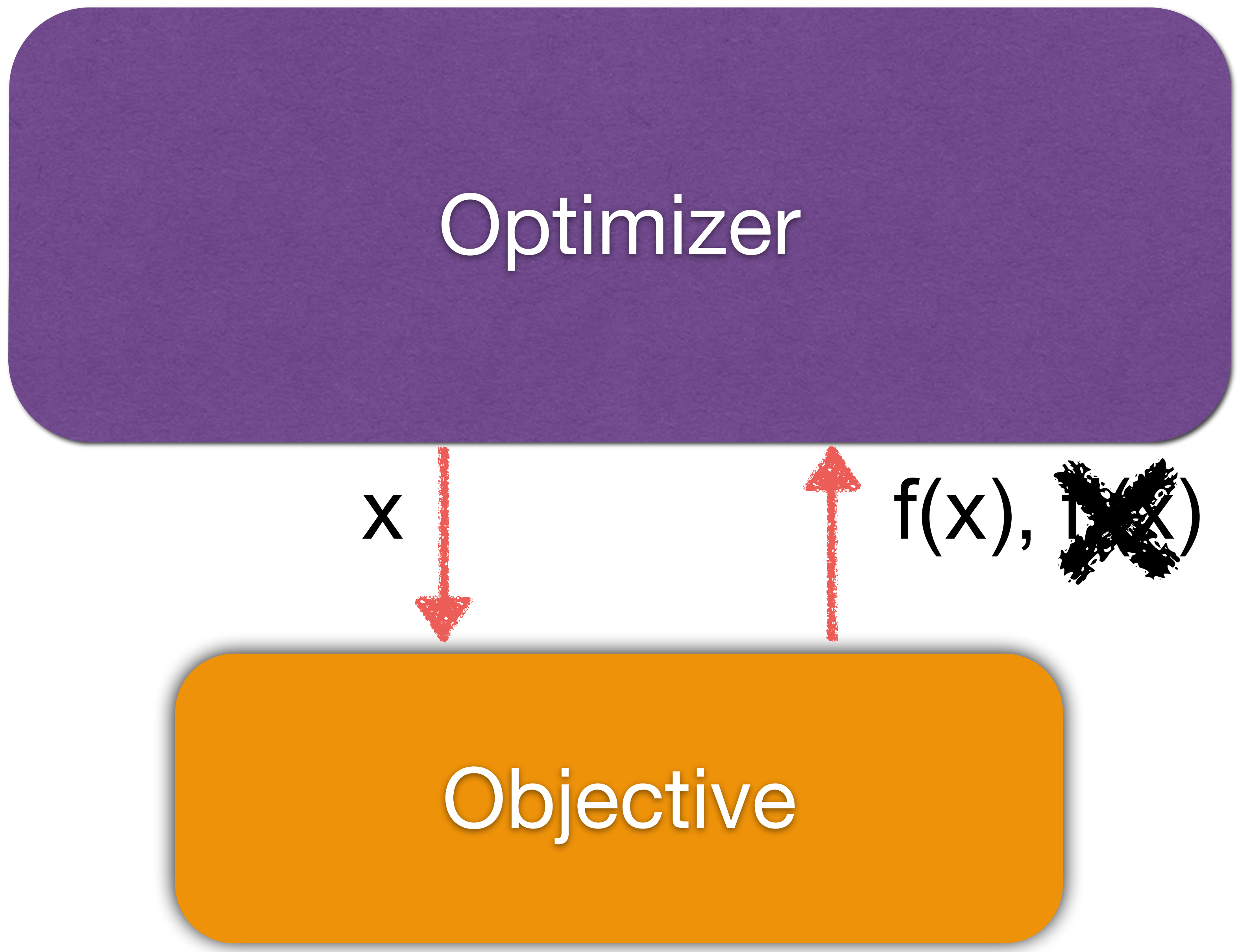
Yi-Chung Hsu 許奕中
Department of Mathematics
National Taiwan University

Spring, 2014

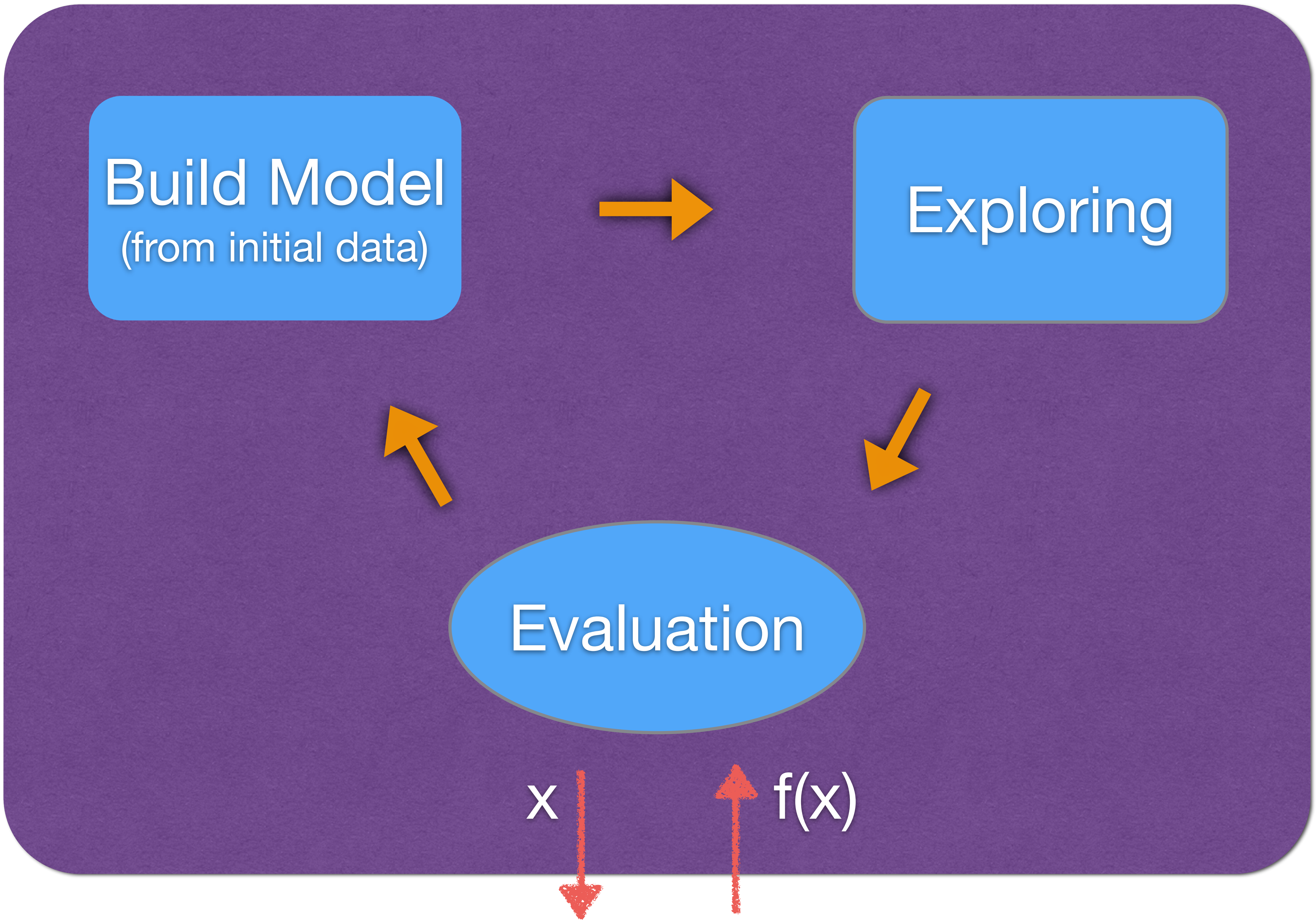
Optimization



Optimization



Surrogate Assisted Tuning



SAT Features

- Modeling
    - Kriging (Gaussian process)
    - Co-Kriging
- Infilling Criteria
    - Expected Improvement (EI)
    - Mean Square Error (MSE)

- MATLAB
    - Reference: Forrester, Alexander, Sobester, Andras, and Keane, Andy (2008) *Engineering design via surrogate modeling: a practical guide*, Chichester, UK, Wiley, 228pp.

Download

http://bit.ly/**1ieBtRs**

Surrogate

- Objective function $f(x)$ on k-dimensional domain

$$X = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}_{n \times k} \quad Y = f(X) = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}_{n \times 1}$$

- Surrogate

$$\hat{f}(X, Y, w, x) \approx f(x)$$

Directories & Routines

- KrigingNd
  - `Kriging_info.m`
  - `Kriging_pred.m`
- Exploring
  - `eiOptimum.m`
- Cokriging
  - `cokriging.m`
  - `cokrigingpredictor.m`
- SamplingPlans
  - `LHD.m`
- Util
  - `grid_cut.m`
  - `visualize.m`

Situation 1

- For the objective function f, and given the data X, Y obtained from the function.
- Now this function routine needs **3 hrs** to compute the value, can you make a guess how much the value is at p in **1 second**?

Kriging in k-Dimensional Domain

- **model** = Kriging_info(X, Y)

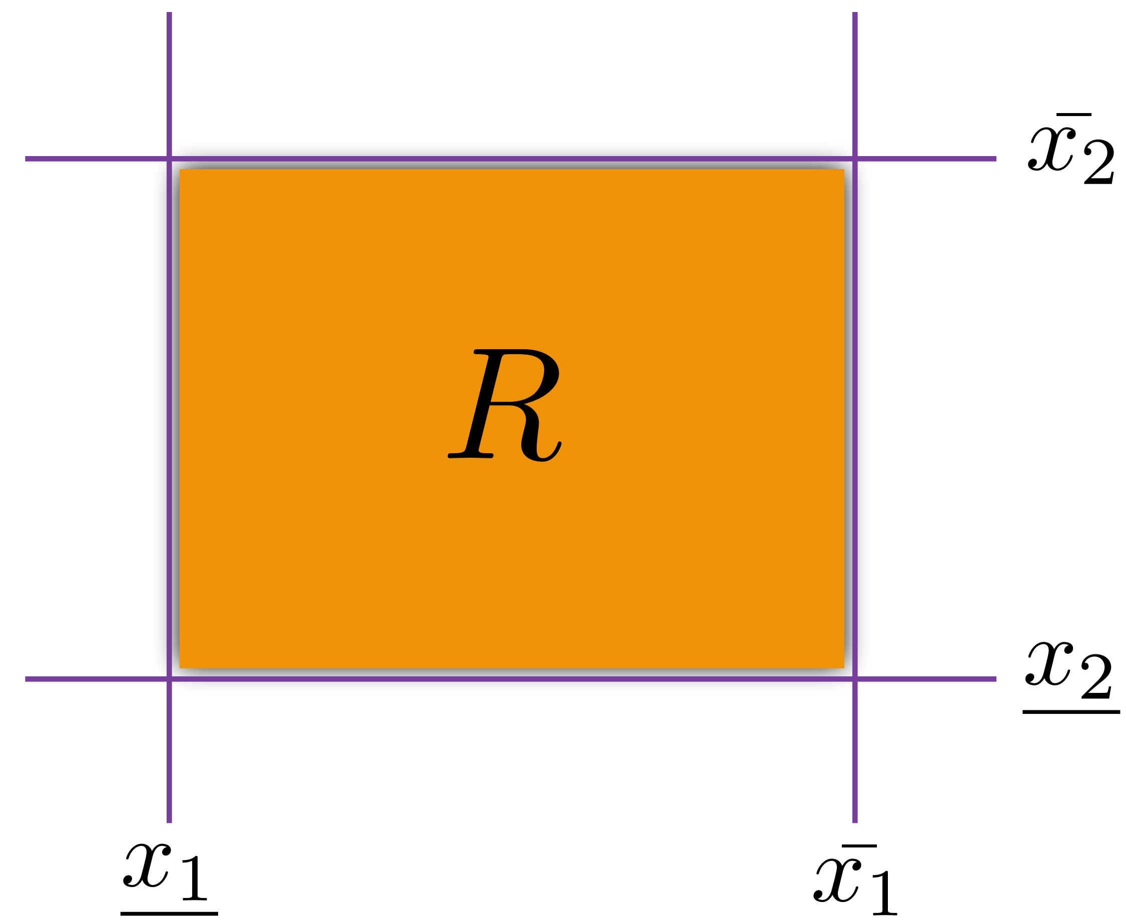
$$X = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}_{n \times k} \qquad Y = f(X) = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}_{n \times 1}$$

- [y_hat, mse] = Kriging_pred(p, **model**)

$$\texttt{y\_hat} = \hat{f}(X, Y, w, p) \approx f(p)$$

Situation 2

- (2D case)Not only at the single point, I want to have a whole visualization of the model in rectangle region R.

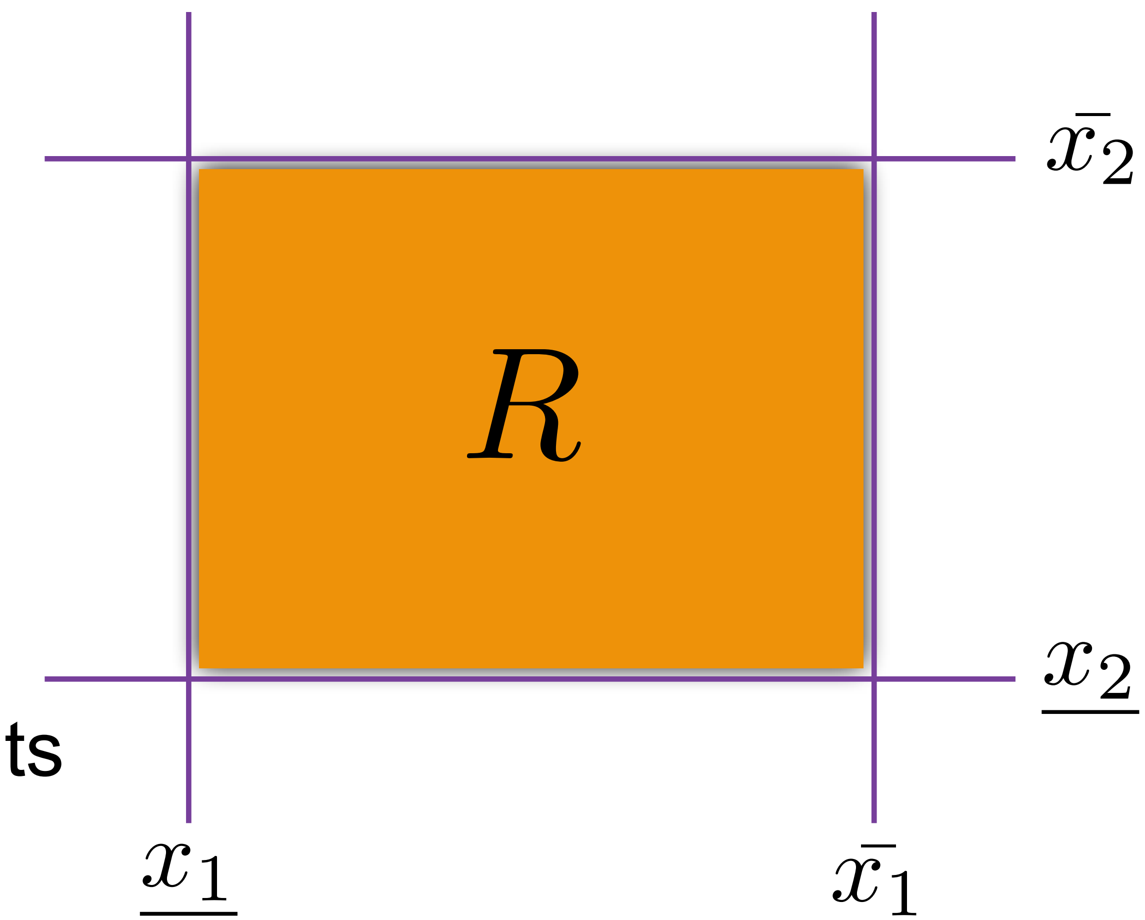


Kriging in k-Dimensional Domain

- `grid = grid_cut(lowerbound, upperbound, gridsize)`

一個維度上所切割的格點數

- Ex: 2-dimensional case
  - `lowerbound` = $[\underline{x_1} \quad \underline{x_2}]$
  - `upperbound` = $[\bar{x}_1 \quad \bar{x}_2]$
  - `gridsize` = 100
- `grid` is a list include $\texttt{gridsize}^k$ points

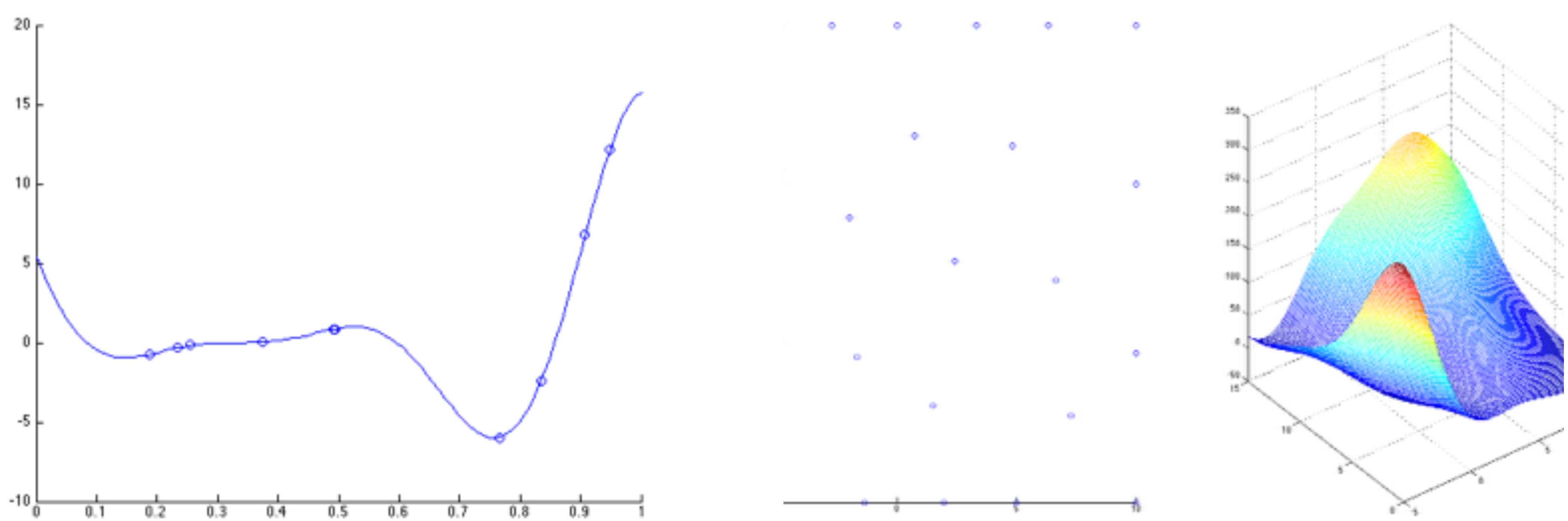


There are $\texttt{gridsize}^k$ points computed in `Kriging_pred`!!

Visualization

- Use G to denote `grid` in mathematics, and also denote $\hat{f}(G)$ with P.

- `visualize(k, gridsize, X, Y, grid, P)`



Situation 3

- We want to find the minimizer of another objective function $h : \mathbb{R}^k \to \mathbb{R}$.

- Suppose the training set $[X, Y]$ is given, find the minimizer in 20 explored **iterations**.

Expected Improvement Measurement

```
G = grid_cut(ub, lb, gridsize);
%% ~~Allocate P, MSE, EI~~
………………………
%% ~~~~~~~~~~~~~~~~~~~~~~~
for iteration = 1:20
   model = Kriging_info(X, Y);
   for i = 1:gridsize^k
         [P(i), MSE(i)] = Kriging_pred(G(i,:), model);
         EI(i)  = eiMinimum(P(i), Y, MSE(i));
   end
```

Expected Improvement Measurement

- `ei = eiMinimum(y_hat, Y, mse)`

- EI means the expectation of being better than current best.
- So that you can locate where has biggest EI w.r.t `grid`, and evaluate the value at it.

Expected Improvement Measurement

```
    for i = 1:gridsize^k
          [P(i), MSE(i)] = Kriging_pred(G(i,:), model);
          EI(i) = eiMinimum(P(i), Y, MSE(i));
    end
    [~, EIMaxIdx] = max(EI);
    x_new = G(EIMaxIdx, :);
    X = [X; x_new]; Y = [Y; h(x_new)];
end             % end of iteration
```

Demonstration

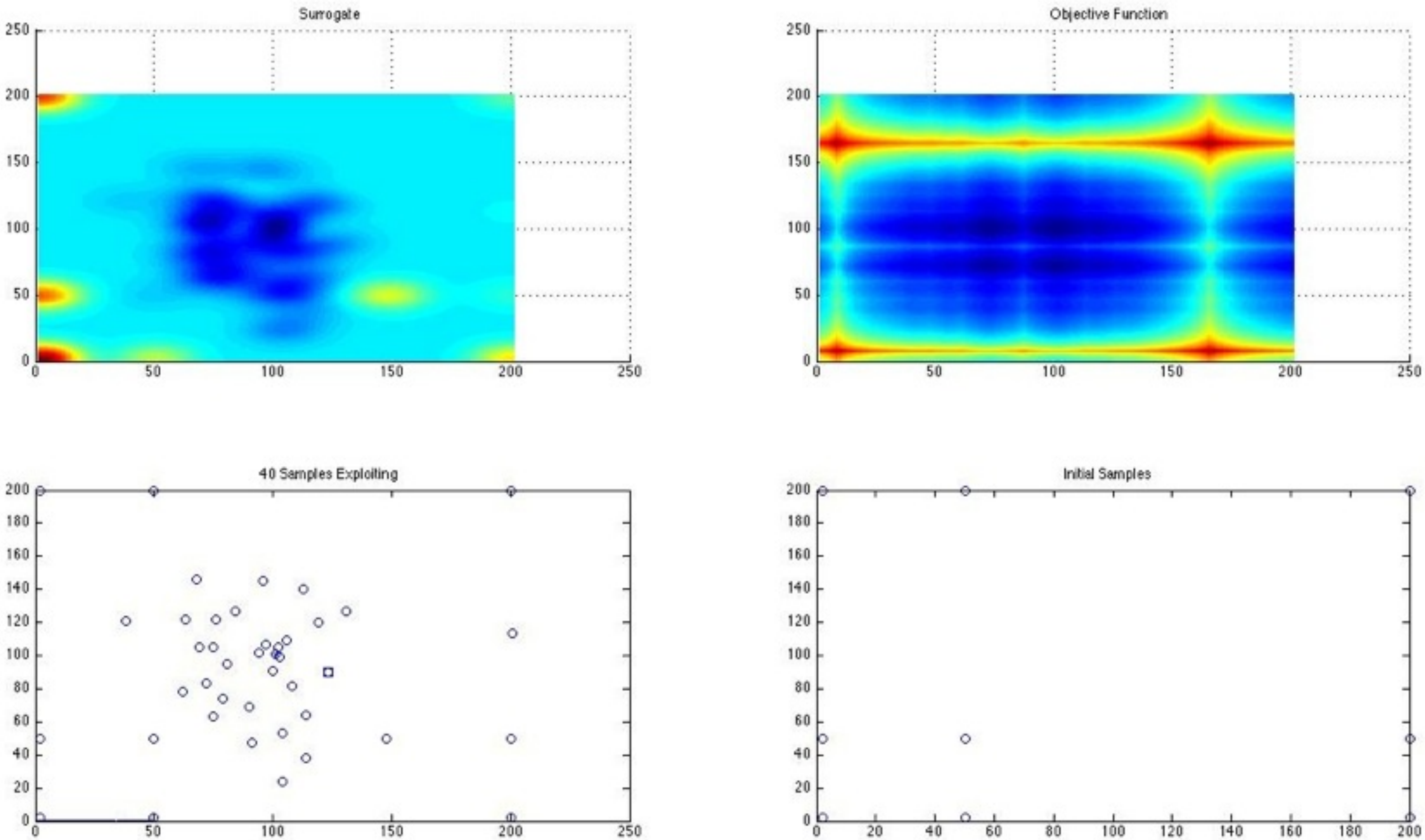
Practice

Practice

- f1: $\mathbb{R}^2 \to \mathbb{R}$
  - Search range $[20, 45] \times [90, 105]$
- f2: $\mathbb{R} \to \mathbb{R}$
  - Search range $[-20, 0]$
- f3: $\mathbb{R} \to \mathbb{R}$
  - Search range $[72, 80]$

Practice

- f4: $\mathbb{R}^2 \to \mathbb{R}$
  - Search range $[1, 200] \times [1, 200]$


Surrogate
Objective Function
40 Samples Exploiting
Initial Samples