Hyperparameter Optimization

Jee-Hyong Lee

Sungkyunkwan Univ.

Unique Origin Unique Future

Hyperparameter Optimization

- **There are too many hyperparameters to be set by expert**
  - Learning rate, Momentum rate, Dropout, Normalization, Number of layers, number of nodes,

- **Is there any efficient way to set them?**
  - No

Hyperparameter Optimization

- **We can regard it as function optimization**
  - I will use 3-layer perceptron
  - I want to find out the optimal hyperparameters
    - Learning rate: η
    - Momentum rate: r
    - Dropout probability: p
    - Number of nodes in the first hidden layer: m
    - Number of nodes in the second hidden layer: n
  - Then, the accuracy of my NN is a function of η, r, p, m, n

$$Accuracy = f(\eta, r, p, m, n)$$

Hyperparameter Optimization

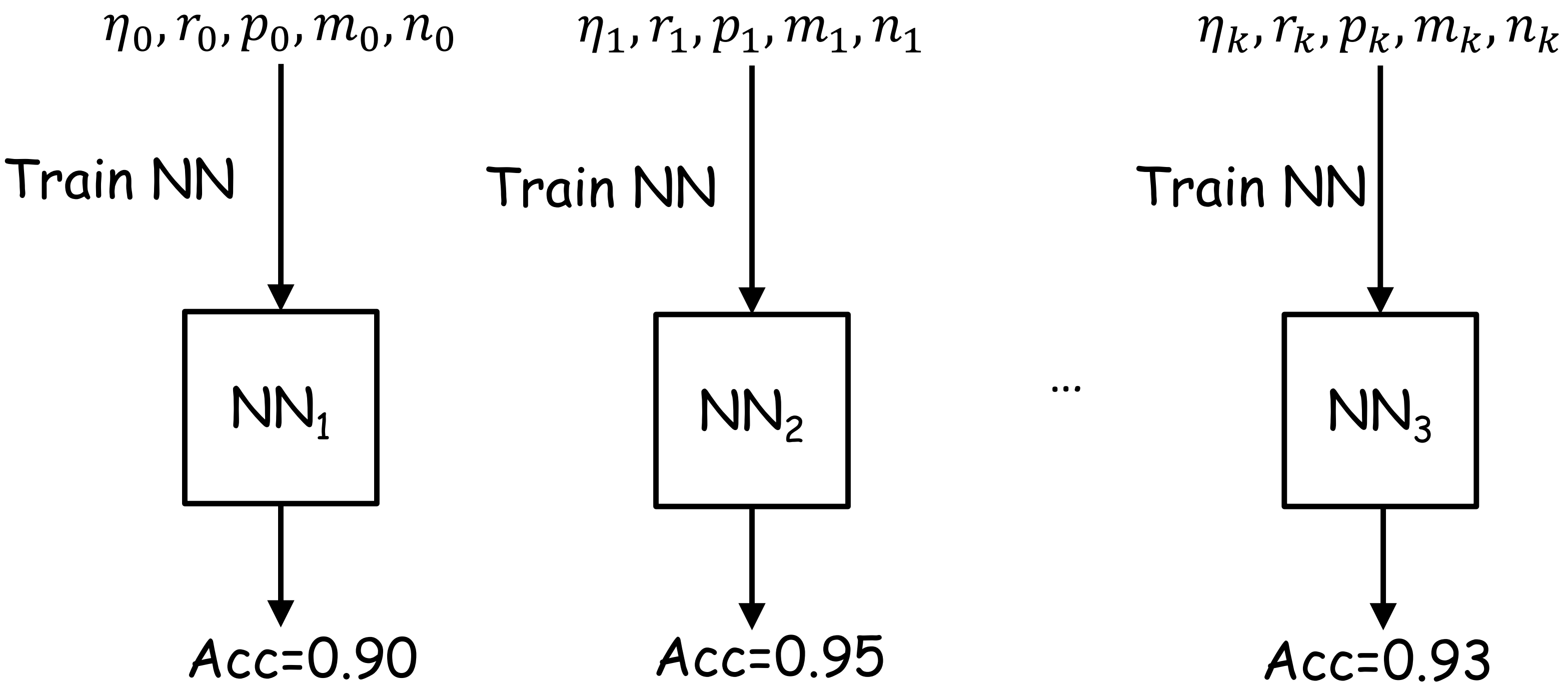
- **We can regard it as function optimization**
  - I need to solve

$$\underset{\eta,r,p,m,n}{\operatorname{argmax}} f(\eta, r, p, m, n)$$

  - Problem
    - I do not know what $f(\eta, r, p, m, n)$ is
    - But I can query. That is, for a some setting we can evaluate f
    - However, the evaluation is very expensive because we need to train a neural network !!

Hyperparameter Optimization

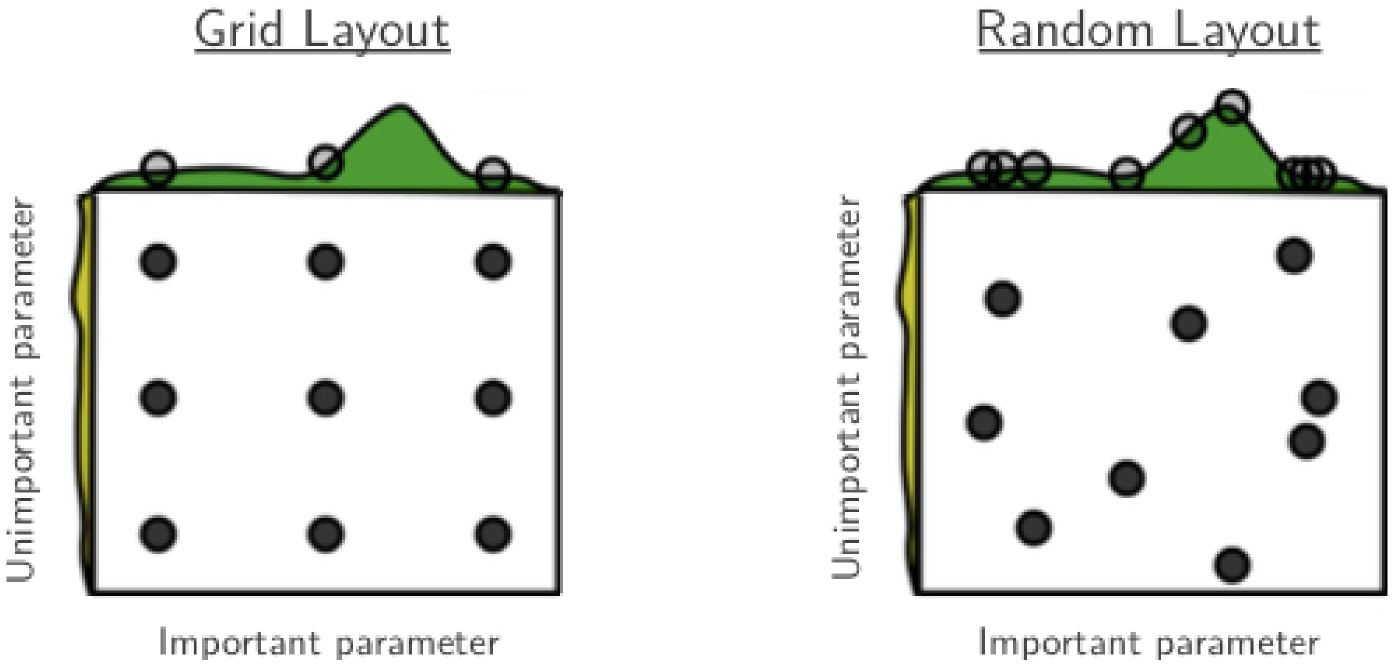
Choose settings



$\eta_0, r_0, p_0, m_0, n_0$
Train NN
NN$_1$
Acc=0.90
$\eta_1, r_1, p_1, m_1, n_1$
Train NN
NN$_2$
Acc=0.95
…
$\eta_k, r_k, p_k, m_k, n_k$
Train NN
NN$_3$
Acc=0.93

Some Simple Search Algorithm

- **Random search, Grid search**



Some Simple Search Algorithm

- **It is not efficient**
  - It does not utilize the previous tries.
  - It would be better to less search the area with low potential and more search the area with high potential

  => Can we choose a better next point based on previous search results?

- **It is too costly**
  - To evaluate how a set of hyperparameters is good, we need to train a neural network
    - we need to train a large neural network every time we try a new set of parameters

  => Can we gradually train NNs as search goes on?

Bayesian Optimization

- **Definition**

$$\underset{\mathrm{x}}{\arg\max}\, f(x)$$

  - You don't know anything about $f(x)$
  - You can query but it is very expensive

  - Any good idea??

Bayesian Optimization

- **Any Good Idea??**
  - No information on $f(x)$ ...

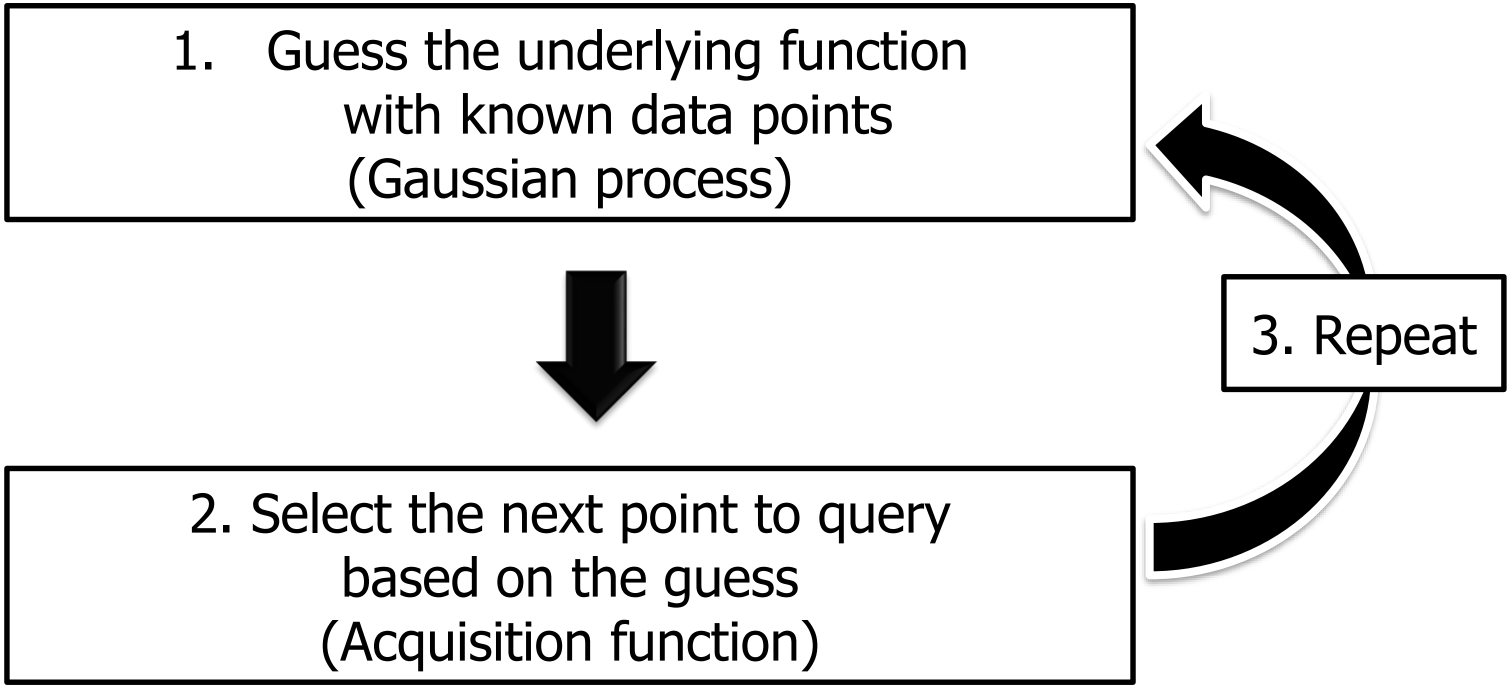
  - First choose a random point, x_1, and evaluate $f(x_1)$
  - Guess the shape of $f(x)$ based on $\left(x_1, f(x_1)\right)$

  - Based on the guess,
    choose the next point, x_2, and evaluate $f(x_2)$
  - Guess shape of $f(x)$ based on $\left\{\left(x_1, f(x_1)\right), \left(x_2, f(x_2)\right)\right\}$

  - Repeat those steps

Bayesian Optimization

- **Overall Description**



1. Guess the underlying function with known data points (Gaussian process)

2. Select the next point to query based on the guess (Acquisition function)

3. Repeat