**Tutorial on Bayesian optimization**

Loc Nguyen

Loc Nguyen’s Academic Network

Email: ng\_phloc@yahoo.com

Homepage: www.locnguyen.net

**Abstract**

**1. Introduction**

As a convention, lowercase and un-bold letters like *a*, *b*, *c*, *x*, *y*, and *z* denote scalar whereas lowercase and bold letters like ***a***, ***b***, ***c***, ***x***, ***y***, and ***z*** denote vector. In some cases, uppercase and un-bold letters like *A*, *B*, *C*, *X*, *Y*, and *Z* can denote vector. Uppercase and bold / un-bold letters like *A*, *B*, *C*,***A***,***B***,***C***, *X*, *Y*, *Z*,***X***,***Y***, and ***Z*** denote matrix. Variables can be denoted as *x*, *y*, *z*, ***x***, ***y***, ***z***, *X*, *Y*, *Z*, ***X***, ***Y***, and ***Z*** whereas constants can be denoted as *a*, *b*, *c*, ***a***, ***b***, ***c***, *A*, *B*, *C*, ***A***, ***B***, and ***C***. Uppercase and bold / un-bold letters like *X*, *Y*, *Z*,***X***, ***Y***, and ***Z*** can denote random variables. However, scalar, vector, matrix, variables, and random variables are stated explicitly in concrete cases.

**2. Bayesian optimization**

**3. Gaussian process regression**

Nonparametric BO is based on Gaussian process regression (GPR) which, in turn, is based on Gaussian process. Therefore, we start the description of GPR with a concept of Gaussian process. Given a random process ***X*** = (*X*1, *X*2,…, *Xn*) over *n* timepoints in which each *Xt* where *t* belongs to {1, 2,…, *n*} is random variable, if and only if for any finite index set {*t*1, *t*2,…, *tk*} of {1, 2,…, *n*} where *tj* belongs to {1, 2,…, *n*}, the subset considered as *tk*-dimension random vector follows multinormal distribution known as multivariate Gaussian distribution then ***X*** is called *Gaussian random process* or Gaussian process (GP) in brief. Note, each represents one dimension of the *k*-dimension random vector variable . Moreover, please pay attention that any combination of follows normal distribution too. Without loss of generality, we denote the random process as random variable ***X*** = (*X*1, *X*2,…, *Xn*)*T* where {*t*1, *t*2,…, *tk*} = {1, 2,…, *n*} obeying multinormal distribution (multivariate normal distribution or multivariate Gaussian distribution) as follows:

Where *μ*(***X***) and Σ(***X***) are mean vector and covariance matrix of ***X***, respectively. Note, the superscript “*T*” denotes transposition operator of vector and matrix whereas the tilde sign “~” indicates probabilistic distribution of a random variable. GP is known as infinite multinormal distribution which is the generalized form of *n*-dimension multinormal distribution because *n* can approach positive infinity *n* = +∞. Let *f*(***X*** | *μ*(***X***), Σ(***X***) be probability density function (PDF) of ***X*** when ***X*** is continuous, we have:

Indeed, mean *μ*(***X***) and covariance Σ(***X***) are functions of ***X***, respectively. Note, in this research, notation *f*(.|.) often denote PDF, which is not concrete function. In literature, *μ*(***X***) is assumed to be zero as *μ*(***X***) = **0***T* for convenience but it can be defined as the random process ***X*** itself, *μ*(***X***) = ***X***.Besides, *μ*(***X***) can be customized according to concrete applications, for example it can be constant as *μ*(***X***) = ***μ***. Covariance function Σ(***X***) measures the correlation between random variables when the random process “moves” them, in which the closer the given two random variables are, the larger their covariance. It is invertible, positive definite, and symmetric. The two most important properties based on covariance function Σ(***X***) of random process are stationarity and isotropy among four basic properties: stationarity, isotropy, smoothness, and periodicity. Stationarity implies that the PDF *f*(***X*** | *μ*(***X***) of random process ***X*** will not be changed when the process is shifted in time, for example, if new random variable *Xn*+1 raises to be added then means and covariances of old (previous) variables *Xi* where 1 ≤ *i* ≤ *n* in ***X*** will not be changed. It is proved that if GP ***X*** satisfies stationarity, Σ(***X***) will depend only on the deviation *Xi*–*Xj* but the inversed statement is not asserted. However, if Σ(***X***) depends only on the Euclidean distance |*Xi*–*Xj*| then, GP ***X*** will satisfy isotropy. If ***X*** satisfies both stationarity and isotropy, ***X*** is called homogeneous process. In cases where each element of matrix function Σ(***X***) depends only on *Xi* and *Xj* like stationarity case and isotropy, it will be formulated as following matrix.

In these cases, covariance function Σ(***X***) is identified with its element function Σ(*Xi*, *Xj*) if the formulation of Σ(*Xi*, *Xj*) is not changed formally and hence, Σ(*Xi*, *Xj*) is called *kernel function*. The following are some kernel functions.

Where *l* is the characteristic length-scale of the process which reinforces similarity of *Xi* and *Xj*. As a convention, let *Xi*:*j* where *i*≤*j* denote a sequential subset of ***X*** such that *Xi*:*j* = {*Xi*, *Xi*+1,…, *Xj*}. Of course, we have *Xi*:*i* = *Xi*. In general case, *Xi*:*j* is arbitrary subset of distinguish variables. Let Σ(*Xi*, *Xj*:*k*) denote a row covariance vector of *Xi* and *Xj*:*k* as follows:

Let Σ(*Xj*:*k*, *Xi*) denote a column covariance vector of *Xj*:*k* and *Xi* as follows:

As a convention, let Σ(*Xi*:*j*, *Xk*:*l*) denote a pseudo covariance matrix of *Xi*:*j* and *Xk*:*l* as follows:

Note, Σ(*Xi*:*j*, *Xk*:*l*) is not an actual covariance matrix because it may not be squared. Obviously, Σ(*Xi*:*j*, *Xk*:*l*) is a partition of Σ(***X***) and we have Σ(***X***) = Σ(*Xi*:*n*, *Xi*:*n*). If denoting ***X***1 =(*Xi*, *Xi*+1,…, *Xj*)*T* and ***X***2 = (*Xk*, *Xk*+1,…, *Xl*)*T*, we denote:

Gaussian process repression (GPR) is based on Gaussian process (GP) when there is a target function which attaches to each *X* where *X* represents any *Xi* in ***X*** such that *Y* = *f*(*X*). Please distinguish the target function *f* from the formal notations *f*(.|.) of probability density function (PDF). Of course, *Y* or *Yi* is also random variable and we also have *Yi* = *f*(*Xi*). Besides, in context of regression model, the target function *f*(*X*) is not a formal function with arithmetic operators and exactly, it is a mapping between *X* and *Y*. For example, sample ***D*** = {***X***, ***Y***} has two paired datasets ***X*** and ***Y*** in which for every *Xi* belonging to ***X*** there is a *Yi* belong to ***Y*** and hence, the equation *Y* = *f*(*X*) only indicates such mapping. GPR model tries to represent or draw a regressive PDF of ***Y*** from its previous ones (which will be explained later) and ***X***. Suppose we had a GP ***X*** = (*X*1, *X*2,…, *Xn*)*T* and target function ***Y*** = *f*(***X***), assuming that the prior PDF of ***Y*** = (*Y*1, *Y*2,…, *Yn*)*T* given ***X*** is also derived from the multinormal PDF of ***X***.

Therefore, we have:

Where *μ*(.) and Σ(.) denote mean function and covariance function (kernel function) in formality, respectively, and Σ(***X***, ***X***) = Σ(***X***). The equation above implies that the prior PDF of GP ***Y*** is initialized with the PDF of GP ***X***. Mean vector and covariance matrix of ***Y*** are *μ*(***Y***) and Σ(***X***, ***X***) *\**) within the PDF *f*(***Y*** | ***X***, *μ*(.), Σ(.)), respectively. The mean function *μ*(***Y***) is redefined here with variable ***Y*** but it should be the same to the mean function *μ*(***X***) in formulation if *μ*(***X***) can be extended with the lower/higher dimensional space. For instance, if *μ*(***X***) is defined as itself *μ*(***X***) = ***X*** then, *μ*(***Y***) will be defined as itself *μ*(***Y***) = ***Y*** = (*Y*1, *Y*2,…, *Yn*)*T*. In literature, for simplicity, *μ*(***Y***) is set to be zero as *μ*(***Y***) = **0***T*. It is better if setting *μ*(***Y***) = **0***T*.

Now suppose we randomize a new set of variables ***X***\* = (*Xn*+1, *Xn*+2,…, *Xn*+*k*)*T* and obtain their evaluated values ***Y***\* = *f*(***X***\*) = (*Yn*+1, *Yn*+2,…, *Yn*+*k*)*T*. Similarly, mean vector and covariance matrix of ***Y****\** are *μ*(***Y****\**) and Σ(***X****\**, ***X****\**) within the PDF *f*(***Y***\* | ***X***\*, *μ*(.), Σ(.)), respectively. In literature, for simplicity, *μ*(***Y***) and *μ*(***Y****\**) are set to be zero as *μ*(***Y***) = **0***T* and *μ*(***Y****\**) = **0***T*. Consequently, the joint PDF of ***Y*** and ***Y***\* which is of course multinormal distribution is denoted as follows:

Where mean vector and covariance matrix of the joint PDF of ***Y*** and ***Y***\* are and , respectively. As a result, the Gaussian process regression (GPR) model of dependent variable *Y* is conditional PDF of ***Y***\* given ***Y***, ***X***, and ***X***\* such as *f*(***Y***\* | ***Y***, ***X***, ***X***\*, *μ*(.), Σ(.)) which is also predictive PDF of ***Y***\* because it can be used to predict next occurrence of *Y*. From interesting properties of multinormal distribution, it is easy to drawn GPR from the joint PDF *f*(***Y***, ***Y***\* | ***Y***, ***X***, ***X***\*, *μ*(.), Σ(.)) as follows:

Where,

Obviously, the GPR of ***Y***\* distributes normally with mean vector *μ*(***Y***\* | ***Y***, ***X***, ***X***\*) and covariance matrix Σ(***Y***\* | ***X***, ***X***\*). Indeed, the GPR *f*(***Y***\* | ***Y***, ***X***, ***X***\*, *μ*(.), Σ(.)) is posterior PDF of ***Y*** whose prior PDF is *f*(***Y*** | ***X***, *μ*(.), Σ(.)). Initializing such prior PDF by the PDF of ***X*** as *f*(***Y*** | ***X***, *μ*(.), Σ(.)) = *f*(***X*** | *μ*(.), Σ(.)) is not totally correct but implicit biases will be decreased after the posterior PDF *f*(***Y***\* | ***Y***, ***X***, ***X***\*, *μ*(.), Σ(.)) is updated. We have following summary:

Although the GPR of ***Y***\* depends on ***Y***, ***X***, and ***X***\*, the main semantic meaning of regression model here is mentioned mainly that ***Y***\* is determined based on its previous one ***Y*** when both of them are assumed to be based on *X* via the PDF of ***Y*** as *f*(***Y*** | ***X***, *μ*(.), Σ(.)) and the PDF of ***Y***\* as *f*(***Y***\* | ***X***\*, *μ*(.), Σ(.)). This implies that *X* is the intermediate point for the probabilistic dependence of ***Y***\* on ***Y***.

If ***X***\* and ***Y***\* are 1-element vectors such that ***X***\* = *Xn*+1 and ***Y***\* = *Yn*+1 and let ***x***1:*n* = ***X***, ***x*** = ***X***\*, *y*1:*n* = ***Y***, and *y* = ***Y***\*, the GPR becomes:

Where,

The equation above with single variables ***x*** and *y* (single posterior processes) is popular in BO, especially ***x*** is vector and *y* is scalar although ***x*** and *y* can be arbitrary such as scalar, vector, and matrix with note that high dimensional spaces require tensor products for example.

GPR can be executed continuously with new ***X***\* while the old ***X***\* is incorporated into larger GP ***X***. In practice, such continuous execution is implemented as iterative algorithm whose each iteration has two following steps:

|  |
| --- |
| *Step* 1:  Take new ***X***\*  *Step* 2: |

**Table 3.1.** GPR algorithm

In step 1, the covariance matrix Σ(***X***, ***X***) is not recomputed entirely because some of its elements Σ(*Xi*, *Xj*) were determined before.

Because GP ***X*** will get huge when the iterative algorithm repeats many iterations and ***X***\* is incorporated into ***X*** over and over again, it is possible to apply first-order Markov property so that each iteration remembers only one previous *X*. Therefore, we can assign ***X***\* to ***X*** as ***X*** = ***X***\* in step 2 so that the iterative algorithm runs faster and saves more computational resources as follows:

|  |
| --- |
| *Step* 1:  Take new ***X***\*  *Step* 2: |

**Table 3.2.** GPR algorithm with first-order Markov property

However, this advantage of first-order Markov property cannot be applied into BO (by defining how to take new ***X***\* with acquisition functions) because it is necessary to look entire *X* domain forward and backward to find out potential global optimizer. The use of Markov property will be appropriate to tasks of prediction and estimation in which entire time series are split into many smaller time windows where it is necessary to remember *w* windows with *w*-order Markov property. Shortly, the assignment ***X*** = ***X***\* is suitable to forward modeling.

GPR focuses mainly on the regressive (posterior) covariance matrix Σ(***Y***\* | ***X***, ***X***\*) via covariance function Σ(*Xi*, *Xj*) because adjusting the regressive (posterior) mean vector *μ*(***Y***\* | ***Y***, ***X***, ***X***\*) via adjusting the mean function *μ*(***Y***) is unnecessary due to:

As usual, *μ*(***Y***) and *μ*(***Y****\**) are set to be zero as *μ*(***Y***) = **0***T* and *μ*(***Y****\**) = **0***T* so that

Which is indeed an arithmetic regression of ***Y***\* on ***Y***, ***X***, and ***X***\* in addition to the main semantic meaning that the posterior PDF of ***Y***\* given ***Y*** is the main regression. However, it is still better to define exactly the mean function *μ*(***Y***) in cases of predicting more precisely confident intervals of *Y* based on *X* because the equation above:

Which implies variation of *Y* according to variation of *X* from the origin. It is not a real value of *Y*. By another way, if setting *μ*(***Y***) = ***Y*** and *μ*(***Y****\**) = ***Y****\** so that

That is not a regression function, which is impossible to predict *Y*\* based on *X*\* when *Y*\* is unknown. Therefore, I propose a technique based on linear regression model to define *μ*(***Y***\*) with constraint that each element *μ*(*Y*\*) of *μ*(***Y***\*) depend only on *Y*\*, for instance:

Note, *Y*\* represents any *Yi* belonging to the subprocess ***Y***\* = (*Yn*+1, *Yn*+2,…, *Yn*+*k*)*T* and *X*\* represents any *Xi* belonging to the subprocess ***X***\* = (*Xn*+1, *Xn*+2,…, *Xn*+*k*)*T*. Let *φ*(*X*) be the transformation function which transforms *X* space into *Y* space such that *φ*(.) is invertible.

This means that *φ*(.) can be defined only if *X* space and *Y* space have the same dimension. The simplest form of *φ*(.) is identity function.

Let *Z\** be a regressive estimate of *X*\*:

Where ***α****\** is regressive coefficient vector with suppose that ***X***\* is also vector. How to calculate ***α****\** from sample {***X***\*, *φ*–1(***Y***\*)} will be described later. Please pay attention that this linear regression is totally different from the regression meaning of GPR via posterior PDF. Let be an estimate of *Y*\* from ***X***\* with association of linear regression model and transformation function *φ*(.).

Obviously, we have:

Let |***X***|, |***X***\*|, |***Y***|, and |***Y***\*| be cardinalities (also dimensions) of ***X***, ***X***\*, ***Y***, and ***Y***\*. Of course, we have |***X***| = |***Y***| and |***X***\*| = |***Y***\*|, for example, we also have |***X***| = |***Y***| = *n* and |***X***\*| = |***Y***\*| = *k*. Concentration ratio (CR) of a subprocess which is point density of such subprocess is defined by the ratio of cardinality of such subprocess to the cardinality of entire process, for example, CR of ***X***\* over {***X***, ***X***\*} is:

Suppose ***α*** being the regressive coefficient vector estimated from sample {***X***, *φ*–1(***Y***)} was determined before, the mean function *μ*(*Y*\*) is proposed as a weighted estimation with concentration ratios of ***X*** and ***X***\* as follows:

As a result, the two steps of the iterative algorithm for estimating GPR are finetuned as follows:

|  |
| --- |
| *Step* 1:  Take new ***X***\*  Calculate ***α****\** from sample {***X***\*, *φ*–1(***Y***\*)}  *Step* 2: |

**Table 3.3.** GPR algorithm with linear regression

Where,

Note, the vector *φ*–1(***Y***\*) is determined:

For interval estimation of *Y*\* with given ***X***\*, suppose GPR algorithm finished obtaining ***α*** and *μ*(*Y*) after some iterations, we calculate:

The mean function *μ*(*Y*\*) is recalculated with ***X***\* as follows:

As a result, the confident interval of *Y*\* which is the pair {*μ*(***Y***\* | ***X***, ***X***\*), Σ(***Y***\* | ***X***, ***X***\*)} is determined as follows:

Because the GPR algorithm here defines the mean function *μ*(*Y*\*) based on linear regression model (LRM), it is necessary to describe LRM in short. Given a dependent random variable *Z* and a set of independent random variables *X* = (1, *X*1, *X*2,…, *Xn*)*T*, LRM tries to establish linear relationship between *Z* and *X* so that *Z* is sum of a linear combination of *X* and an random error *ε*.

As a convention, *Xj* are called regressors and *Z* is called responsor whereas ***α*** = (*α*0, *α*1, *α*2,…, *αn*)*T* are called regressive coefficient vector. Suppose *ε* distributes normally with mean 0 and variance *σ*2 then *Z* distributes normally with mean ***α****TX* and variance *σ*2 due to:

Note, *E*(.) and *V*(.) denote theoretical expectation and variance, respectively. This implies that the PDF of random variable *Z* is:

LRM of *Z* given *X* is built from sample {***X***, **z**} of size *N* in form of data matrix as follows:

Therefore, *xij* and *yi* is the *i*th instances of regressor *Xj* and responsor *Z* at the *i*th row of matrix (***X***, **z**). As a convention, we can connect datasets ***X*** and ***Z*** of LRM here with the GP ***X\**** and the set *φ*–1(***Y***\*) aforementioned, respectively with note that removing values 1 from datasets ***X*** and ***Z***.

For simplicity, variable *Z* of LRM here is scalar but it can be extended as vector variable for general case when it is possible to build up many scalar LRMs independently to form a vector LRM as a simplest technique.

The essence of LRM is to estimate the regressive coefficient vector ***α*** and the variance *σ*2. By applying maximum likelihood estimation (MLE) method, we obtain estimates and of ***α*** and *σ*2.

**4. Acquisition functions**

**5. Numerical simulation**

**6. Conclusions**

**Appendices**

**References**