Multi-objective Bayesian Optimization Method in the Allocation of Asset Portfolio

Student Name: [redacted]
Supervisor Name: [redacted]
Submitted as part of the degree of M.Sc. MISCADA to the
Board of Examiners in the Department of Computer Sciences, Durham University

Abstract

Many real-world applications require the optimization of multiple conflicting objectives. Bayesian global optimization (BGO) methods are very applicable for the optimization of black-box functions that are expensive to evaluate, and can be extended to Multi-objective Bayesian Optimization (MOBO) methods to solve the problem with conflicting goals. BGO models the objectives via probabilistic surrogates and define an acquisition function that evaluating the objective at new designs. The Expected Improvement (EI) is a well established criterion in optimization with costly function evaluations in BGO, which has recently been generalized to multi-objective optimization. Formulating the expected improvement based on the hypervolume indicator is a promising approach to solve multi-objective problem. Given the bayesian model of the optimization, the Expected Hyper-Volume Improvement (EHVI) computes the expected gain in attained hypervolume for a given input point. In this paper, I use MOBO method based on gaussian process regression and EHVI criterion to identify the pareto front with a relatively smaller number of simulations. Numerical examples illustrate the performance of the hypervolume-based EI in the multiobjective BGO framework. In real-world problems, I will use this method to find the pareto set for best weights of different assets in an asset portfolio. The goal is to choose design variables that optimize two conflicting objectives: maximizing returns and minimizing risk. The MOBO method based on EHVI criterion exhibits good data efficiency in thus problems.

Keywords: Bayesian Optimization, Gaussian Process, Hypervolume, Expected Improvement, Multi-Objective Optimization.

Contents

1 Introduction

In many scientific and real-world applications, we often need to deal with optimization problem. When the function is expensive to be evaluated, methods relying on a large number of evaluations of the objective function will lead to inefficiency. In mathematics, the optimization problem can be defined as finding the parameter values that return the minimal (or maximal) value of a objective function. In most real-world problems, there are usually more than one objective functions which are conflicting to be optimized at the same time: optimizing one objective means that other objectives will be non-optimal. So it is desirable to find a trade-off that satisfies the different objectives as much as possible. In this case, multi-objective optimization is expected to be pareto efficient, where no one objective can be further improved without hurting the others. As for multi-objective optimization problem, there exist many multi-objective optimization algorithms, like the non-dominated sorting based genetic algorithm (Deb et al. [2000]), and the multi-objective evolutionary algorithm based on decomposition (Zhang and Li [2007]). These above approaches rely on a large number of evaluations, which become inefficient when the objective functions are costly in simulation.

Bayesian global optimization (BGO) methods are proposed to optimize black-box functions, where a surrogate model (gaussian process regression) is built for predicting the performance of a set of parameters. The trade-off between exploration and exploitation is executed by an acquisition function to obtain a optimal solution. Bayesian optimization mainly relies on two steps: (1) train statistical surrogate models using previous experiments, and then (2) using an acquisition function to find input points which yield the best added value for the optimization. In this way, modeling will be more accurate in the vicinity of the pareto front of the problem, and the number of simulations is expected to be reduced.

This paper considers the problem of the allocation of an asset portfolio, which has two conflicting objectives in real-application: (1) maximizing returns and (2) minimizing risk. The objectives depend on the weights of different assets, which can be regard as parameters of the objectives. I introduce the multi-objective bayesian optimization (MOBO) method and use actual yield data of various financial products with different returns and risks to conduct numerical simulation to obtain the optimal asset portfolio, so as to provide suggestions for investors. The experimental results will show that the proposed method can generate well-representative solutions.

2 Existing methods

The black-box objective function optimization problems is costly to evaluate. Since the gradient of the objective function is difficult to unknown or evaluate, gradient descent methods are not applicable in such problems. In this case, bayesian optimization replaces the objective function by gaussian process regression and split the original problem by a sequence of simpler optimization problems, this is why gaussian process regression and bayesian optimization are closely related. In this sec-

tion, I will introduce the bayesian framework to simultaneously perform function optimization for black-box objective functions. When it is extended to multi-objective optimization problems, we can define the optimal solution by the concept of pareto front.

2.1 Gaussian Process Regression

Before starting to optimize the black-box function $f(x)$, we need to consider a formulating prior belief about $f(x)$ first. Let us assume that there are N points have been gathered, before starting to select new points, a surrogate probabilistic model for $f(x)$ should be calculated. Let $\mu(x)$ be the prior expectation for all x, thus the function $\boldsymbol{\mu}$ is the prior expectation of $f(x)$. Then let the prior covariance between $\mu(x)$ and $\mu(x_*)$ be $k(x, x_*)$ for all x and x_*, in particular $var(f(x)) = k(x, x)$. Let the marginal distribution of $f(x)$ be normal for every x and the joint distribution of $f(x)$ and $f(x_*)$ be bivariate normal for all x and x_*. So the joint distribution of $f(x)$ at any p specific points is a p-dimensional multivariate normal distribution.

This is typically a Gaussian Process (GP), which is a typical surrogate model for bayesian optimazition. GP is a collection of random variables, any finite subset of these variables are jointly gaussian distributed. The random variables are the values of the objective functions. We place a GP prior $GP(\mu(\cdot), k(\cdot, \cdot))$ on the unknown function $f(x)$, where $\mu(\cdot)$ and $k(\cdot, \cdot)$ are the mean and covariance functions respectively. GP prior is fully specified by $\mu(\cdot)$ and $k(\cdot, \cdot)$ (Olofsson et al. [2019])

The GP prior implies that when we observe finite number of function values $\boldsymbol{f} = [f(x_1), ..., f(x_n)]$ at points $X = [x_1, ..., x_n]$, we can derive the posterior distribution of $f(x_*)$. The joint distribution of the observed function values $\boldsymbol{f}$ at locations X and $f(x_*)$ at a query point x_* can be writen as:

$$\begin{bmatrix} \boldsymbol{f} \\ f(x_*) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu} \\ \mu(x_*) \end{bmatrix}, \begin{bmatrix} \boldsymbol{K} & \boldsymbol{k}_*^\top \\ \boldsymbol{k}_* & k(x_*, x_*) \end{bmatrix} \right) \tag{1}$$

where

$$\boldsymbol{\mu} = (\mu(x_1), \mu(x_2)..., \mu(x_n))^T$$

$$\boldsymbol{k}_* = (k(x_1, x_*), k(x_2, x_*)..., k(x_n, x_*))^T$$

$$\boldsymbol{K} = \begin{pmatrix} k(x_1, x_1), & k(x_1, x_2), & ... & k(x_1, x_n) \\ k(x_2, x_1), & k(x_2, x_2), & ... & k(x_2, x_n) \\ ... & ... & ... & ... \\ k(x_n, x_1), & k(x_n, x_2), & ... & k(x_n, x_n) \end{pmatrix}$$

We can get an expression for the posterior predictive distribution of gaussian process regression with bayesian method:

$$f(x_*)|(\boldsymbol{f}, X) \sim N(\mu_*, \sigma_*^2) \tag{2}$$

where

$$\begin{cases} \mu_* = \boldsymbol{k}_*^\top \boldsymbol{K}^{-1} \boldsymbol{f} = \mu(x_*) + \boldsymbol{k}_*^\top \boldsymbol{K}^{-1}(\boldsymbol{f} - \boldsymbol{\mu}) \\ \sigma_*^2 = k(x_*, x_*) - \boldsymbol{k}_*^\top \boldsymbol{K}^{-1} \boldsymbol{k}_* \end{cases} \tag{3}$$

But in many real-world problems, the objective function should be evaluated through a "noisy" procedure, which means $y_i = f(x_i) + \varepsilon_i$. A typical hypothesis assuming independent gaussian centered noise is: $\varepsilon_i \sim N(0, \sigma_\varepsilon^2)$, which are i.i.d. noise terms. In this case, equation (3) becomes:

$$\begin{bmatrix} \boldsymbol{f} \\ f(x_*) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu} \\ \mu(x_*) \end{bmatrix}, \begin{bmatrix} \boldsymbol{K} + \sigma_\varepsilon^2 I & \boldsymbol{k}_*^\top \\ \boldsymbol{k}_* & k(x_*, x_*) \end{bmatrix} \right)$$

Gaussian process regression modeling naturally adapts to this case. Assuming independent identically distributed gaussian noise ε_i with variance σ_ε^2, the posterior predictive mean and variance are:

$$\begin{cases} \mu_* = \mu(x_*) + \boldsymbol{k}_*^\top (\boldsymbol{K} + \sigma_\varepsilon^2 I)^{-1} (\boldsymbol{f} - \boldsymbol{\mu}) \\ \sigma_*^2 = k(x_*, x_*) - \boldsymbol{k}_*^\top (\boldsymbol{K} + \sigma_\varepsilon^2 I)^{-1} \boldsymbol{k}_* \end{cases} \tag{4}$$

The number of points for evaluation is relatively small in this sequential decision making setting, so the predictions based on gaussian process regression are relatively easy to compute. One advantage of GP is that it can estimate the confidence region for test points. This feature is really useful for global optimization, because validation points can be evaluated according to the confidence of the model (Brochu et al. [2010]).

The covariance function is central in the analysis of GP, which called the kernel function. The most popular kernel function is the squared exponential kernel, but in this case the divergences of all features of x affect the covariance equally. Typically, the covariance functions will have some free parameters. For generalization, it is necessary to generalize by adding hyperparameters. A popular choice is the squared-exponential covariance function with a vector of automatic relevance determination (ARD) hyperparameters Λ(Mockus [2012]), which has the following form:

$$k(\boldsymbol{x}, \boldsymbol{x}') = exp(-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{x}')^\top \boldsymbol{\Lambda}^{-1}(\boldsymbol{x} - \boldsymbol{x}')) \tag{5}$$

where $\Lambda = diag(\lambda_1^2, ..., \lambda_D^2), x \in R_D$, is a diagonal matrix of squared length scales λ_d, it parameterizes the length scales of the inputs and allows us to scale each dimension of the inputs separately. Our aim is to explore the effects of varying the hyperparameters on GP prediction: the parameters

Λ influence how the GP model fit the observed data. In this form, the hyperparameters means if one of the element in hyperparameters Λ is small, the effect on the covariance of x corresponding to it will be small too. There are various methods for determining the hyperparameters from training data, an usual way for selecting parameters is to maximize the likelihood function $L(\Lambda) = p(y|\Lambda)$. The underlying idea is to maximize the probability of the sample data y. For a given model, in the case of the GP regression, $\Lambda = (\Lambda_K, \sigma_\varepsilon)$ consists of the parameters Λ_K of the kernel function and the standard deviation σ_ε of the noise. Let $z = f(x)$ represents the GP. We have:

$$p(y|\Lambda) = \int p(y|\Lambda, z)p(z|\Lambda)dz$$

It is common to maximize the log marginal likelihood function:

$$\hat{\Lambda} = \arg\max_{\Lambda} \boldsymbol{l}(\Lambda) = \arg\max_{\Lambda} \ln p(y|\Lambda).$$

In the case of gaussian noise, we have $Y \sim N(0, \boldsymbol{K}(\Lambda_K) + \sigma_\varepsilon^2 I)$. where $\boldsymbol{K}(\Lambda_K)$ is the kernel matrix that depends on the parameters Λ_K. The log marginal likelihood function can be writen as:

$$\boldsymbol{l}(\Lambda) = -\frac{n}{2}\ln(2\pi) - \frac{1}{2}\ln \mid \boldsymbol{K}(\Lambda_K) + \sigma_\varepsilon^2 I \mid -\frac{1}{2}Y^T(\boldsymbol{K}(\Lambda_K) + \sigma_\varepsilon^2 I)^{-1}Y$$

This kind of problem can usually be solved by gradient-descent algorithms when is possible to compute analytically the gradient of $\boldsymbol{K}(\Lambda_K)$. When $\boldsymbol{l}(\Lambda)$ is not always convex, and sometimes will suffer local maxima, where requires to use Monte Carlo methods, more details in (Gonzalvez et al. [2019]).

2.2 Bayesian Optimization

First, consider the minimization of a single objective function $f(x)$. Bayesian optimization construct gaussian process regression to evaluate the objective funtion, which exhibits more certainty in well-explored regions than in unexplored regions. We can obtain the optimal point for the next evaluation by maximizing an acquisition function, which makes bayesian optimization more data efficient by fewer evaluations. So it is practical to deal with black-box functions that are costly to evaluate.

Bayesian optimization mainly consists of two parts: probabilistic surrogate model and acquisition function. First, we build a prior probabilistic model (gaussian process regression) for the objective function $f(x)$, given the gaussian process regression model, we need to determine the optimal location to sample next point, and then update the model with samples drawn from$f(x)$ to get a posterior probability distribution. When selecting the next point, we need to consider the trade-off between exploitation and exploration, which is determined by maximizing an acquisition function. Exploitation means sampling where the surrogate model prediction objective gain is high and exploration means sampling where predicts a higher uncertainty (Frazier [2018]). Assume the

Algorithm 1 Bayesian optimization steps

Input: $f, \mathcal{X}, \mathcal{S}, GP$

$D \leftarrow$**InitSamples**$(f, \mathcal{X})$

for $i \leftarrow \mid D \mid$ to T do :

- $p(y \mid x, D) = \text{FitModel}(GP, D)$
- $x_i = \arg\max_{x \in \mathcal{X}} S(x, p(y \mid x, D))$
- $y_i = f(x_i)$
- $D \leftarrow D \cup (x_i, y_i)$

end for

conditional distribution $f(x_*))|y, X \sim N(\mu_*, \sigma_*^2)$ is given for a point x_*. The following expressions are different choice of acquisition functions for single-objective bayesian optimization:
upper condence bound:

$$UCB(x) = \mu_* + \kappa\sigma_*(\kappa > 0)$$

probability of improvement:

$$PI(x) = P(f(x_*) < y_{min}) = \Phi(\tilde{\mu})$$

and expected improvement:

$$EI(x) = E[y_{min} - f(x_*)] = \sigma_*(\tilde{\mu}\Phi(\tilde{\mu}) + \varphi(\tilde{\mu}))$$

where $\tilde{\mu} = \frac{(y_{min} - \mu_*)}{\sigma_*}$, $y_{min} = \min_i y_i$ is the best value of f observed so far, $\varphi(\cdot)$ is the zero-mean gaussian probability density function with unit variance and $\Phi(\cdot)$ the corresponding cumulative distribution function (Brochu et al. [2010]). The improvement is defined as the difference between the current minimum of the observations and the new function value.

The general idea of bayesian optimization is summarized in the reference to Algorithm 1 .

Remark. f is the black-box function to be optimized, $\mathcal{X}$ is the search space of x, D represents a data set composed of several pairs of data, each pair of array is represented as (x, y), $\mathcal{S}$ is acquisition function, and T is the number of cycles to choose x.

2.3 Multi-Objective Optimization: Pareto Front

Assume a D-dimensional input space and m conflicting objective functions $f_i : R^D \rightarrow R, i = 1, ..., m$, unknown functions expressing the relation between each objective and explanatory variable are denoted as $F(x) = (f_1(x), f_2(x), ...f_m(x))^T$To find an optimal trade-off between the objective

functions, a popular way to deal with thus problem is the concept of pareto front.

Our goal is to minimize objective functions $F(x)$ over continuous space $X \subseteq R^D$,each evaluation of an input $x \subseteq X$ produces a vector of objective values$y = (y_1, y_2, \ldots, y_m)$ where $y_i = f_i(x)$ for all $i \in \{1, 2, \ldots, m\}$. pareto-optimality is a kind of state that on one objective function can be improved without impairing another. The optimal solution of multi-objective optimization problem is a set of points $X^* \subseteq X$ such that no point $x_0 \in X/X_*$ pareto-dominates a point$x \in X_*$. The solution set X_* is called the optimal pareto set, and the corresponding set of function values $Y_* = (f_1(X^*), f_2(X^*), \ldots, f_m(X^*))$ is called pareto front, which is useful for selecting solutions. The pareto set is usually not finite, optimization is to provide a finite set that represents X_*well.

The pareto front is approximated by the set of non-dominated function observations. We aim to approximate X^* by minimizing the number of function evaluations. Since the pareto front is unknown, when calculating actual optimization problem, a new point x is added to the surface formed by the non-dominated solution set D_n^* , which defined by a set satisfying the following condition:

$$\forall x, (x, f(x)) \in D_n^* \subset D_n, \forall x\prime, (x\prime, f(x\prime)) \in D_n,$$

$$\exists i \in \{1, 2, \ldots, m\}, f_i(x) \leq f_i(x\prime)$$

Based on the current observation data set $D_n = \{(x_1, F(x_1)), \cdots, (x_n, F(x_n))\}$. The solutions are searched based on the amount of improvement when adding a new point to update the observation dataset as $D_{n+1} = \{D_n, (x_{n+1}, F(x_{n+1}))\}$ (Wada and Hino [2019]).

3 Method for Sampling in Simulation

3.1 Design of Acquisition Function

Multi-objective bayesian optimization focus on methods where gaussian process regression models are fitted to each objective. How to sample next points for query is an important question. In the single-objective case, the expected improvement(EI) is the conditional expectation of the improvement provided by a new observation $Y(x)$, which evaluates the potential gain of an additional point based on the expected increase over the best observation so far:

$$EI(x) = E[max(0, \min_{1 \leq i \leq n} (y_i - Y(x)) \mid Y(x_1) = y_1, ..., Y(x_n) = y_n] \tag{6}$$

When multiple objectives are considered, the improvement function can be defined by estimating the expected "progress" brought by a new observation (relatively to the current set of non-dominated observations D_n^*), such as the hypervolume have been proposed. Specifically, the hypervolume improvement (HVI) is the increment of the volume contained between the pareto front and a reference

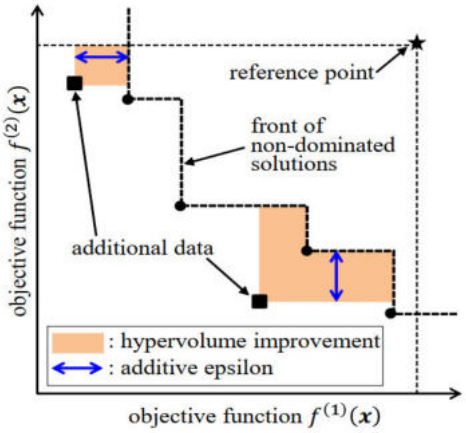


Figure 1
Illustration of of hypervolume improvement and additive epsilon for multi-objective optimization with two variables. (Wada and Hino [2019])

point in the objective space, when a non-dominated point is added. Suppose $HV(D_n^*)$ is the hypervolume of the region dominated by D_n^* with a reference point (a user-defined parameter to specify the upper limit of the pareto solution to be searched) as an upper bound. Then, HVI is defined as:

$$HVI(F(x_{n+1})) = HV(D_{n+1}^*) - HV(D_n^*). \tag{7}$$

An illustrative example is given in Figure 1.

Selecting EHVI as the acquisition function has many advantages. By contrast, probability of hypervolumn improvement(PHVI) is the probability of a point not to be dominated by the current pareto set. Region with non-zero probability in the design space represents the potential improvement of the pareto front. This sequential infill criteria only consider the current set of non-dominated observations insdead of providing a continuous representation of the pareto front. In contrast to PHVI criteria, EHVI prefer new points that dominate most points within the pareto set. When no such points are found, the acquisition function based on EHVI encourages the selection of points that extend the pareto set in an uniform way (Mickael et al. [2019]). So the design for the acquisition function is crucial in identifying an optimal and uniform pareto set.

3.2 Calculation of Expected Hyper-Volume Improvement

We have known that gaussian process regression have the property of providing a prediction and uncertainty for the objective function values $f_i(x), i = 1, ..., m$, which are defined by a multivariate normal distribution. What we are interested in is to find the maximum value of the improvement brought by a new point. A good theoretical overview of different types of EI is given by (Wagner et al.

[2010]). In this section I refer to the method proposed by (Emmerich et al. [2011]) and specialize this method on evaluating the hypervolume-based EI efficiently in two-objectives condition and discuss on computation procedure for the expected hypervolume improvement (EHVI).

One method to calculate EHVI and in multi-objective optimization by multi-point searching has been proposed by (Wada and Hino [2019]), which searches for q candidate points at each iterative search. Thereafter, q candidate points $x_1, \cdots, x_q$ are arranged to a vector $X = [x_1, \cdots, x_q]^T \in R^q$. The acquisition function can be writen as :

$$\mathbb{E}_{\mathrm{HV}}(I) = E_{p(F_q(X)|D_n)}[HVI(F_q(X))]$$

where $F_q(X) = [F(x_1)^\top, \cdots, F(x_q)^\top]^\top$, $HVI(F_q(X))$ is the improvement of the hypervolume when q candidates $F(x_1), \cdots, F(x_q)$ are added, and $p(F_q(X)|D_n)$ is the posterior of $F_q(X)$ given the observation dataset D_n.

Each of the objective functions $f_m(x)$ are assumed to follow independent gaussian processes as $p(F_q(X)|D_n) = p(f_1(x_1), \cdots, f_1(x_q)|D_n), \cdots, p(f_m(x_1), \cdots, f_m(x_q)|D_n)$. When calculating expectation, this acquisition function requires an integral with respect to multivariate distribution, so it is difficult to calculate it analytically. When we consider multi-point search, computation would be much more difficult. Approximation using Monte Carlo sampling is performed:

$$\mathbb{E}_{\mathrm{HV}}(I)_{MC} = \frac{1}{N}\sum_{n=1}^{N} HVI(F_{q,n}(X))$$

where N is the number of Monte Carlo samplings, $F_{q,n}(X)$ is the n-th sample point that follows the distribution $p(F_q(X)|D_n)$.

But there are many significant challenges when optimizing this acquisition function by this method: a) requires lots of simulations, which can be potentially sub-optimal; b) optimization is expensive; c) performance depend on the number of Monte-Carlo samples. In comparison, a simplified version of the hypervolume-based EI is proposed. Emmerich et al. [2011] proposed a method to calculate it by decomposing the non-dominated region into a much smaller set of cells as is also done in this work, which provides a simplified computation procedure for the integral expression. This will be useful to enhance both accuracy and speed of computation. The expected improvement at a point x with respect to the approximation set $\mathcal{P}_\alpha$, which is the integral of EHVI (Emmerich [2005]). $\mathbb{E}_{\mathrm{HV}}(I)$ can be defined as:

$$\mathbb{E}_{\mathrm{HV}}(I) = \int_R I_V(\boldsymbol{y}, \mathcal{P}_\alpha) PDF_{\boldsymbol{x}}(\boldsymbol{y}) d\boldsymbol{y} \tag{8}$$

where $I_V(\boldsymbol{y}, \mathcal{P}_\alpha) = HV(\mathcal{P}_\alpha \cup \boldsymbol{y}) - HV(\mathcal{P}_\alpha)$, the integration region R is $R^2$when we have two objectives. $I_V(\boldsymbol{y}, \mathcal{P}_\alpha)$is the hypervolume improvement that a point y would yield to the area bounded by $\mathcal{P}_\alpha$, calculated with respect to some reference point. Note that $I_V(\boldsymbol{y}, \mathcal{P}_\alpha) = 0$ if y is dominated

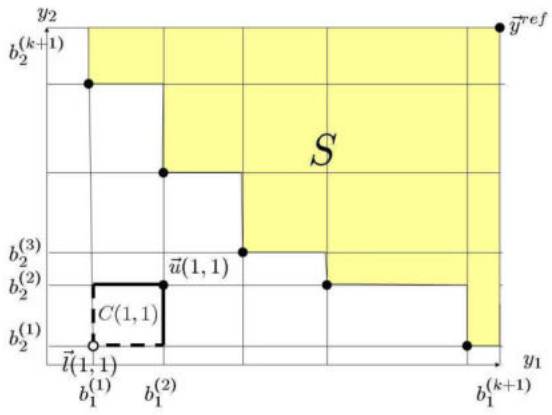


Figure 2
The black points are the points of the population, except the point in the upper right cornerwhich is the reference point for the hypervolume. The yellow region deflnes the measured hypervolume S. The grid coordinates are indicated by $b_1^{(i)}$ and $b_2^{(i)}$ for the flrst and second coordinate, respectively. Grid-cell $C(1,1)$ is highlighted by a thick black boundary.(Emmerich et al. [2011])

by any point in $\mathcal{P}_\alpha$.

Improvement Contribution (IC) denotes the improvement contribution of each cell. Figure 2 exhibit an intuition on the grid-variables and areas in two-objectives that will be introduced in the following. To calculate the improvement contribution of each cell, Let T_i denote the i-th step and $y = (y_1, y_2) \in R^2$. Let $r = (r_1, r_2) \in R^2$ be a reference point. Then $I_V(\boldsymbol{y}, \mathcal{P}_\alpha) = \sum_{i=1}^{k+1} IC(y, T_i)$, where k is the number of points in currently approximation set $\mathcal{P}_\alpha$. Thus EHVI is the sum of all improvement contribution integral over the set of cells. The integral of EHVI can be written in closed form as:

$$\mathbb{E}_{\mathrm{HV}}(I) = \int_{R^2} \sum_{i=1}^{k+1} IC(y, T_i) PDF_{\boldsymbol{x}}(\boldsymbol{y}) d\boldsymbol{y} \tag{9}$$

Let $\mathcal{P} = (y_{(1)}, \cdots, y_{(k)}) \in \mathcal{P}_\alpha$. The integration region R can be partitioned into a set of interval boxes. Let $b_i^{(1)}, b_i^{(2)}, ..., b_i^{(k)}$ denote the sorted list of all the i-th coordinates of the vectors $y_{(1)} \in R, \cdots, y_{(k)} \in R$. In case of two-objectives problems, $i = 1, 2$. We define $b_i^{(0)} = -\infty, b_i^{(k+1)} = r_i, b_i^{(k+2)} = +\infty$. The grid-coordinates $b_i^{(i)}$give rise to a partitioning into grid cells. For each$(i_1,...,i_n)$, where $i_s \in \{0, ..., k+1\}$, the grid cell named $C(i_1,...,i_n)$ is determined by $(b_1^{(i_1)}, ..., b_2^{(i_n)})^T$. We call $(b_1^{(i_1)}, ..., b_2^{(i_n)})^T$ the lower corner of $C(i_1,...,i_n)$ denoted by $\mathbf{l}(i_1,...,i_n)$, likewise we call $(b_1^{(i_1+1)}, ..., b_2^{(i_n+1)})^T$ the upper corner of of $C(i_1,...,i_n)$ denoted by $\mathbf{u}(i_1,...,i_n)$. With this notation we can denote $C(i_1,...,i_n)$ by $[\mathbf{l}(i_1,...,i_n), \mathbf{u}(i_1,...,i_n)]$. As shown in Figure 2.

But there are many cells the integration over which adds a contribution of zero to the integral (Emmerich and Klinkenberg [2008]). These are cells that:

(1) have lower corners $\mathbf{l}(i_1,...,i_n)$ that are dominated or equal to points in $\mathcal{P}$, i.e.$\mathcal{P} \preceq \mathbf{l}(i_1,...,i_n)$, or

(2) have upper corners $\mathbf{u}(i_1,...,i_n)$ that do not dominate the reference point, i.e. $\mathbf{u}(i_1,...,i_n) \npreceq \boldsymbol{r}$. where $\boldsymbol{r}$ is the reference point and the mean value $\hat{\boldsymbol{y}} \leq \boldsymbol{r}$ and $\mathcal{P} \preceq \boldsymbol{r}$.

Cells which meet the criterion (1) or (2) is regarded as inactive cells, while the other cells are active cells. Obviously, the expected improvement is the sum of all contributions of the improvement integral over the set of active cells C^+, i.e.

$$\mathbb{E}_{\mathrm{HV}}(I) = \sum_{C(i_1,...i_n) \in C^+} \int_{\boldsymbol{y} \in [\mathbf{l}(i_1,...,i_n), \mathbf{u}(i_1,...,i_n)]} I(\boldsymbol{y}, \mathcal{P}) PDF_{\boldsymbol{x}}(\boldsymbol{y}) d\boldsymbol{y} \tag{10}$$

Now, I specify the calculation process in the two-objectives problem. The following equations will discuss how the contribution of each cell is computed.

$$\int_{\boldsymbol{y} \in [\mathbf{l}(i_1,...,i_n), \mathbf{u}(i_1,...,i_n)]} I(\boldsymbol{y}, \mathcal{P}) PDF_{\boldsymbol{x}}(\boldsymbol{y}) d\boldsymbol{y}$$

$$= (\prod_{j=1}^{2} (\int_{y_j \in [l_j(i_1,...,i_n), u_j(i_1,...,i_n)]} I(y_j, \mathcal{P}) PDF_x(y_j) dy_j) - Vol(S^-) \prod_{j=1}^{2} (\Phi(\frac{u_i - \mu_i}{\sigma_i}) - \Phi(\frac{l_i - \mu_i}{\sigma_i})) \tag{11}$$

where the term $\int_{y_j \in [l_j(i_1,...,i_n), u_j(i_1,...,i_n)]} I(y_j, \mathcal{P}) PDF_x(y_j) dy_j$ equals to:

$$(\Psi(v_j(i_1,...,i_n), u_j(i_1,...,i_n), \mu_j, \sigma_j) - \Psi(v_j(i_1,...,i_n), l_j(i_1,...,i_n), \mu_j, \sigma_j))$$

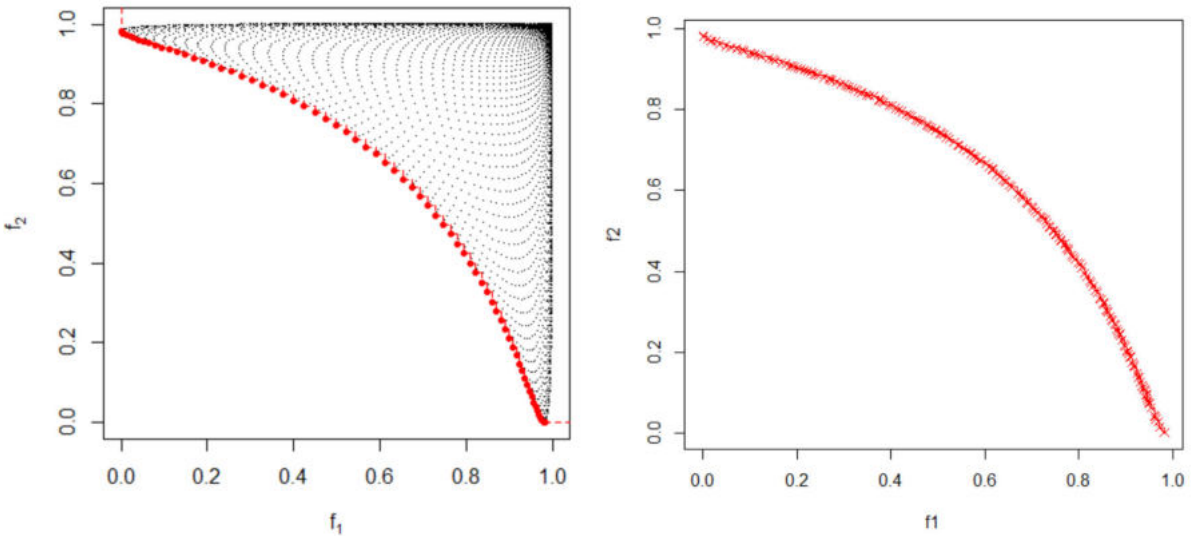
where vector $\boldsymbol{v}(i_1,...,i_m) \in R$, The j-th coordinate of $\boldsymbol{v}(i_1,...,i_m)$ is the j-th coordinate of the intersection point, and $\Psi(.)$ is the integration of the marginal normal distributions:

$$\Psi(a, b, \mu, \sigma) = \sigma * \phi(\frac{b - \mu}{\sigma}) + (a - \mu) * \Phi(\frac{b - \mu}{\sigma})$$

where $\phi(.)$ represents the probability density function of the standard normal distribution, and $\Phi(.)$ represents the cumulative probability density function of the normal distribution.

The second term $Vol(S^-)$ denotes the hypervolume measure for points in the subset of reference point $\boldsymbol{v}(i_1,...,i_m)$ and $\mathcal{P}$ dominated or equal to $\mathbf{u}(i_1,...,i_m)$:

$$S^- = \left\{ \overrightarrow{\mathbf{z}} \in R^m \mid \mathcal{P}(\mathbf{u}) \preceq \overrightarrow{\mathbf{z}} \preceq \mathbf{v} \right\}, \text{where} P(\mathbf{u}) = \left\{ \overrightarrow{\mathbf{p}} \in \mathcal{P} \mid \mathbf{u} \preceq \overrightarrow{\mathbf{p}} \right\}$$



(a) (b)

Figure 3: Plot of pareto front

(a) Plot of pareto front using Multi-objective Bayesian Optimization method based on EHVI criterion, cost 0.3442709s. (b) Plot of pareto front by Non-dominated Sorting Based Genetic algorithm, cost 0.386719s.

4 Numerical Experiment

First we consider a two objectives, one-dimension example, we aim to minimize the following two functions:

$$\begin{cases} f_1 = 1 - \exp((1-x)^2) \\ f_2 = 1 - \exp((1+x)^2) \end{cases} \tag{12}$$

with $x_1 \in [0,1]$ and $x_2 \in [0,1]$, and calculated by Gpareto package in R. In this case, we first build an initial set of observations and two gaussian process regression models and then choose the next point to evaluate using the EHVI criterion. As competitors, I chose non-dominated sorting based genetic algorithm (NSGA-II) from mco package in R, (Roustant et al. [2012]), and then run on the same desktop. The pareto front is shown in Figure 3. These two methods get similar results, and bayesian optimization methods has less evaluations which results in higher efficiency.

Then consider a two objectives, two dimensions example, we aim to minimize the following two functions:

$$\begin{cases} f_1 = (x_2 - 5.1(\frac{x_1}{2\pi})^2 + \frac{5}{\pi}x_1 - 6)^2 + 10((1-\frac{1}{8\pi})\cos(x_1) + 1) \\ f_2 = -\sqrt{(10.5 - x_1)(x_1 + 5.5)(x_2 + 0.5)} - \frac{(x_2 - 5.1(\frac{x_1}{2\pi})^2 - 6)^2}{30} - \frac{(1-\frac{1}{8\pi})\cos(x_1)+1}{3} \end{cases} \tag{13}$$

with $x_1 \in [-5, 10]$ and $x_2 \in [0, 15]$ (re-scaled to $[0,1]^2$) and calculated by Gpareto, and the result is

shown in Figure 4. In this case, We can clearly see that the value of EHVI become largest around the pareto set.The white point represents the point sampled for evaluation , and the change of color represents the EHVI value. The orange area represents the best range for sampling.

5 Application

5.1 Problem Formulation

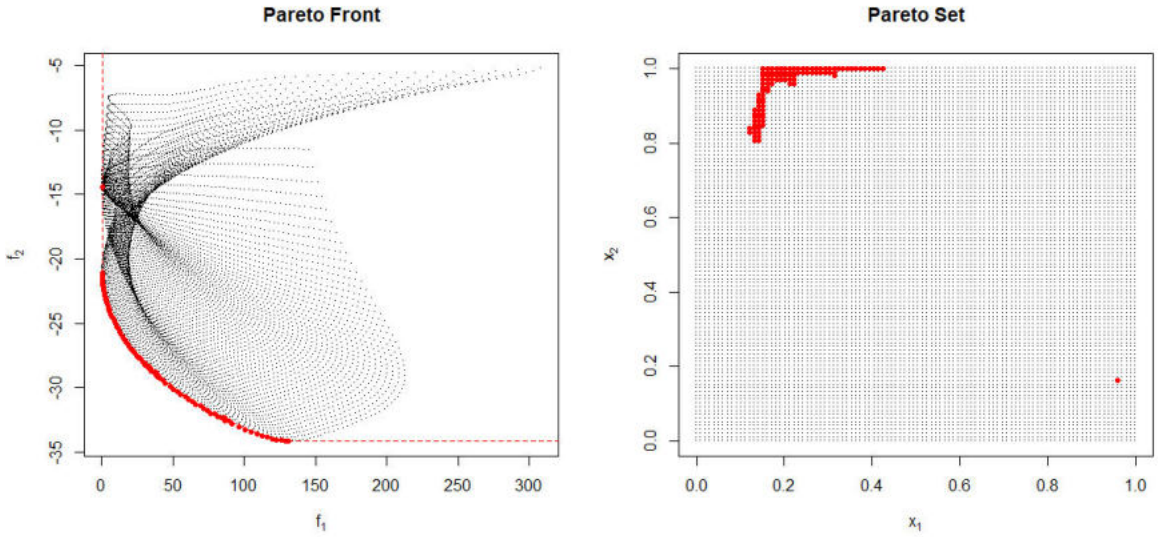
Investment portfolio can be regarded as a process of pursuing both high returns and safety. This is also a multi-objective optimization process, we need to apply a multi-objective optimization model for asset allocation. Based on the constant relative risk aversion : CRRA utility function ,we have two goals: maximize the end-of-period wealth utility and minimize the risk in the asset selection process to pursue the optimal allocation of financial assets under the multi-objective system. Assume the wealth W is allocated to N types of assets with different risks. The relative risk aversion coefficient (γ_i) is constant number, which are different for different types of risk assets, and can be fixed before calculation. The objectives can be expressed as minimizing the following two functions:

$$\begin{cases} f_1 = -\sum_{i=1}^{N} \alpha_i \frac{(1+\mu_i)^{1-\gamma_i}}{1-\gamma_i} \\ f_2 = \sum_{i=1}^{N} \alpha_i (\frac{(\mu_i - \frac{1}{2}\sigma_i^2)t}{\sigma_i \sqrt{t}} + \Phi^{-1}(c_i)). \end{cases} \tag{14}$$

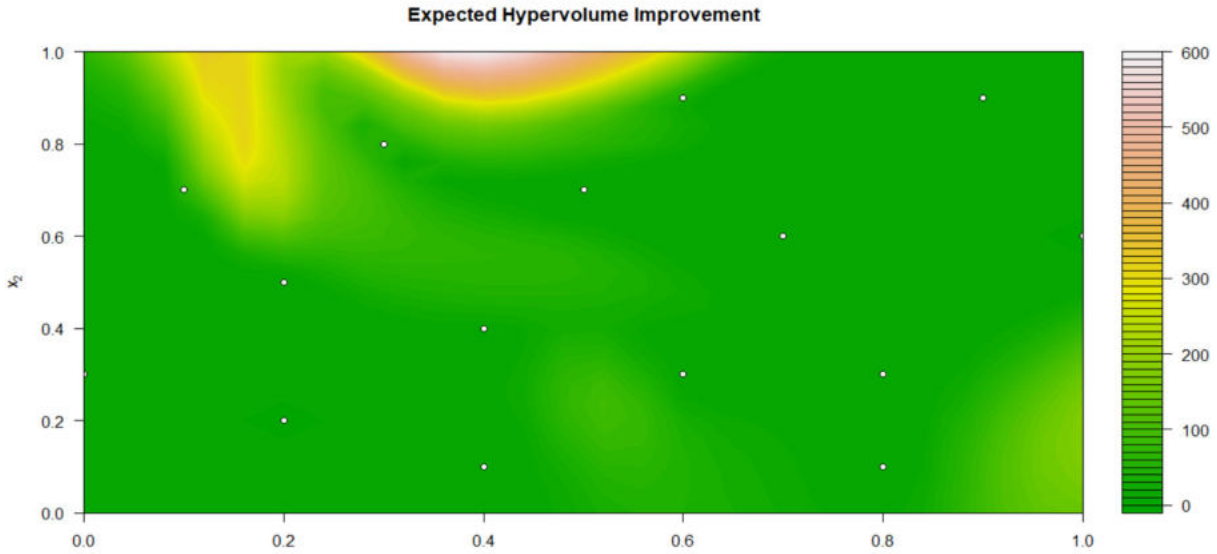
where f_1 is the opposite number of multiple of return relative to the beginning of the period, f_2 is the VaR value, which is a estimation of the maximum loss of asset value caused by adverse market changes in a certain confidence degree in a period of time, and $\alpha_i, (i = 1..n)$ represents the allocation weight of different assets (re-scaled to $[0, 0.5]^2$, where we assume each asset cannot exceed half of the portfolio), μ_i and σ_i^2 represent the average nominal rate of return and standard deviation of various assets respectively, c is the confidence level and t is the holding period of the asset portfolio. the derivation is given in the appendix

5.2 Method and Data

In this section I apply the above bayesian optimization method based on EHVI into the bi-objective optimization problem in allocation of assets in portfolio. I choose gold, NASDAQ index and the stock of Apple company as the target assets in the portfolio. We obtained the daily closing price of these assets for nearly a year from the Yahoo Finance. To get the expected rate of return μ_i of each asset A_i, we calculate the growth rate of each day relative to the previous day and get the average value of these rate, then multiply 365 days to get the annualized yield. And σ_i^2 is the annualized variance of daily return. For the relative risk aversion coefficient γ_i, we set the γ_i of stock is the highest, while that of gold is the lowest. We assume confidence c_i equals to 0.05 for all assets and the holding period Δt is 1. The calculation results are shown in Table 1.



(a) Plot of pareto front and pareto set



(b) Expected hypervolume improvement

Figure 4

A_i	μ_i	σ_i^2	γ_i	c_i
gold	0.394	0.0587	0.1	0.05
NASDAQ index	0.456	0.16389	0.25	0.05
stock of apple company	0.857	0.23871	0.35	0.05

Table 1: Related parameters

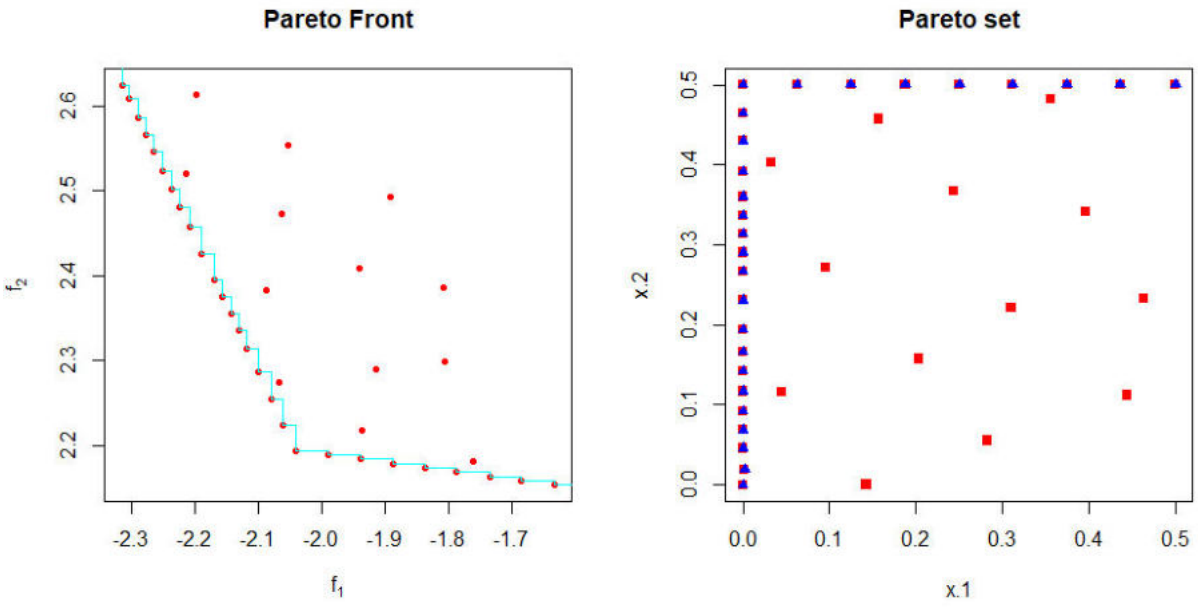
According to the relevant parameters in Table 1, Gpareto is used to solve the target equation (14). We control the weight of each asset within 0.5, and the sum of the weights of these three assets is 1. Let f_1 represents the opposite number of return, and f_2 represents the value of risk, the running results are shown in Figure 5.

We can get the pareto front and pareto set of the portfolio using the method proposed above. In the picture of pareto set above, x_1 represents the optimal value of the weight of gold, and x_2 represents the optimal value of the weight of NASDAQ index, the sum of weights for all asset equals to 1, so the remaining weights are assigned to the stock of apple company. The result shown in Fig.5 (b) illustrates the weights of different assets we should allocate at the same time, which shows the weight of the NASDAQ index should account for half of the total assets because of its good profitability and relatively low volatility. We can conclude that the best trade-off between return and risk can be found by allocating our wealth in different assets.

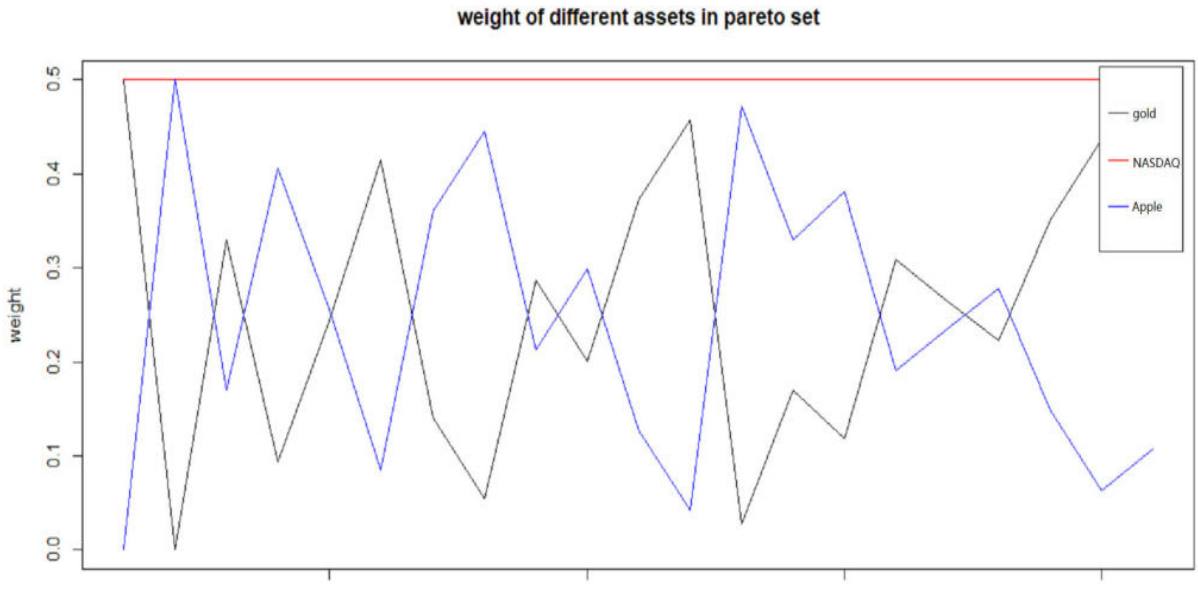
However, in the actual investment of financial assets, due to the large range of various asset weights, it will cause the arbitrariness of arrangement for asset weights and increase the difficulty of asset allocation. Therefore, this kind of portfolio allocation scheme based on interval is not convenient to operate and has limited reference value in the reality. To this end, we need to find a more concise, clear and easier asset allocation scheme. In order to reduce the difficulty of investment decision, we calculate the average weight of all kinds of assets by arithmetic average of each optimal solution set, and consider it as the result of optimal allocation of financial assets. The results show that the weight of gold, Apple company and NASDAQ index are 25.66%, 24.34%, 50% respectively in the this assets portfolio. The optimized configuration results show that, in the allocation of financial assets, investors should pursue the multiple objectives of both profitability and security.

6 Discussion and Conclusions

The procedure for computation of the expected improvement in hypervolume is accurate and efficient in the case of more than one objectives. This paper use the acquisition functions EHVI for multi-objective bayesian optimization of blackbox function, the described procedure use expressions that can be directly computed insdead of complex integration. The given procedure can be considered as a more accurate and often faster computation alternative to the Monte Carlo integration method. The pareto front yields an interaction between the conflicting objectives, which can make an a posteriori trade-off decision preferable to an a priori one. We showed that constructing a probabilistic surrogate model of the expensive black box enables us to construct a probabilistic pareto front. The benefit of gaussian process regression is that they act as cheap simulators for black-box functions with confidence bounds which provides us with a measure of confidence for selecting unexplored optimum. The result demonstrate successful exploitation of multi-objective problems and prove that the method used in this paper is data efficiency. This paper mainly describes the multi-objective bayesian optimization method in the condition of low dimension. It is a challenge to apply this



(a) Plot of pareto front and pareto set



(b) Plot of weights of different assets in pareto set

Figure 5

(a) Outputs of optimization are points on blue line, which represent pareto front, and the blue points in pareto set is the best points we should choose. The red points represent the results of all sampled points. (b) The weights of there different assets in pareto set.

method to higher dimension to get credible and demonstrable results.

Supplementary material

Online supplementary material associated to the dissertation is available from:
mco: https://CRAN.R-project.org/package=mco.
Gpareto: https://CRAN.R-project.org/package= Gpareto.
The codes in this paper is available on: https://github.com/liyihangaaa/Multi-objective-Bayesian-Optimization-method-in-the-allocation-of-Asset-portfolio

References

Eric Brochu, Vlad M. Cora, and Nando de Freitas. A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *CoRR*, abs/1012.2599, 2010. URL `http://arxiv.org/abs/1012.2599`.

Kalyanmoy Deb, Samir Agrawal, Amrit Pratap, and Tanaka Meyarivan. A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: Nsga-ii. In *International conference on parallel problem solving from nature*, pages 849–858. Springer, 2000.

Michael Emmerich. Single-and multi-objective evolutionary design optimization assisted by gaussian random field metamodels. *Dissertation, LS11, FB Informatik, Universität Dortmund, Germany*, 2005.

Michael Emmerich and Jan-willem Klinkenberg. The computation of the expected improvement in dominated hypervolume of pareto front approximations. *Rapport technique, Leiden University*, 34:7–3, 2008.

Michael TM Emmerich, André H Deutz, and Jan Willem Klinkenberg. Hypervolume-based expected improvement: Monotonicity properties and exact computation. In *2011 IEEE Congress of Evolutionary Computation (CEC)*, pages 2147–2154. IEEE, 2011.

Peter I Frazier. A tutorial on bayesian optimization. *arXiv preprint arXiv:1807.02811*, 2018.

Joan Gonzalvez, Edmond Lezmi, Thierry Roncalli, and Jiali Xu. Financial applications of gaussian processes and bayesian optimization. *arXiv preprint arXiv:1903.04841*, 2019.

Binois Mickael, Saint-Ãtienne Mines, and Victor Picheny. Gpareto: An r package for gaussian-process-based multi-objective optimization and analysis. *Journal of Statistical Software*, 30(1): 1–30, 2019.

Jonas Mockus. *Bayesian approach to global optimization: theory and applications*, volume 37. Springer Science & Business Media, 2012.

S. Olofsson, M. Mehrian, R. Calandra, L. Geris, M. P. Deisenroth, and R. Misener. Bayesian multiobjective optimisation with mixed analytical and black-box functions: Application to tissue engineering. *IEEE Transactions on Biomedical Engineering*, 66(3):727–739, 2019.

Olivier Roustant, David Ginsbourger, and Yves Deville. Dicekriging, diceoptim: Two r packages for the analysis of computer experiments by kriging-based metamodeling and optimization. *Journal of Statistical Software*, 51(1):1–55, 2012.

Takashi Wada and Hideitsu Hino. Bayesian optimization for multi-objective optimization and multi-point search. *arXiv preprint arXiv:1905.02370*, 2019.

Tobias Wagner, Michael Emmerich, André Deutz, and Wolfgang Ponweiser. On expected-improvement criteria for model-based multi-objective optimization. In *International Conference on Parallel Problem Solving from Nature*, pages 718–727. Springer, 2010.

Q. Zhang and H. Li. Moea/d: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on Evolutionary Computation*, 11(6):712–731, 2007.

A Section of the appendix

A.1 Construction of two-objectives in assest allocation problem

First we construct an utility maximization portfolio model. Suppose the original wealth at the beginning is W_0 and the wealth at the end of the period is W. The W_0 is allocated to n types of assets with different risks, and $A_i(t), (i = 1..n)$ is used to represent this type of asset. CRRA utility function is a power exponential utility function where the relative risk aversion coefficient (γ_i) is constant number.which are different for different types of risk assets,and can be fixed before calculation.CRRA utility function is additive,and can be expressed as:

$$U(W) = \frac{W^{1-\gamma_i}}{1-\gamma_i}(0 < \gamma_i \leq 1). \tag{15}$$

$\alpha_i(t), (i = 1..n)$is used to represent the allocation weights of various assets. We use $r_i(t), (i = 1..n)$ to represent the nominal rate of return of various types of different risk assets, and t is the holding period of the asset portfolio. The objective function is as follows: The objective function is $max(E[U(W)])$,where:

$$U(W) = \sum_{i=1}^{N} U(A_i(t)) = \sum_{i=1}^{N} \frac{(A_i(t))^{1-\gamma_i}}{1-\gamma_i}. \tag{16}$$

the expected utility maximization objective function of the end of period wealth W can be expressed as:

$$\max(\sum_{i=1}^{N} \alpha_i E(U(A_i(t)))) = \max(\sum_{i=1}^{N} \alpha_i \frac{(A_i(t))^{1-\gamma_i}}{1-\gamma_i}).$$

which equals to

$$\min(-\sum_{i=1}^{N} \alpha_i E(U(A_i(t)))) = \min(-\sum_{i=1}^{N} \alpha_i \frac{(A_i(t))^{1-\gamma_i}}{1-\gamma_i}) = \min(-\sum_{i=1}^{N} \alpha_i \frac{(\mu_i)^{1-\gamma_i}}{1-\gamma_i}). \tag{17}$$

Next we construct a VaR minimized portfolio model. VaR is a estimation of the maximum loss of asset value caused by adverse market changes in a certain confidence degree in a period of time. After comprehensively considering various market risk sources, the VaR method obtains a highly generalized risk measurement value through calculation. Therefore, we choose the VaR method to analyze and measure various market risks of financial asset portfolios. Assume W^* is the minimum wealth value of the asset portfolio at the end of the confidence level c, r_p is the rate of return of the entire portfolio, r_p^* is the minimum rate of return. The VaR of the asset portfolio can be expressed as follows:

$$VaR = W_0[E(r_p) - r_p^*] \tag{18}$$

If the return and loss of the asset portfolio during the holding period follows a continuous distribution with a probability density function $f(W)$, then the confidence c for calculating the VaR of the asset portfolio can be expressed as:

$$c = \int_W^{\infty} f(W)dW = \int_{-\infty}^{VaR_{ci}} f(A_i(\Delta t))d\Delta t \tag{19}$$

where

$$A_i(\Delta t) \sim N((\mu_i - \frac{1}{2}\sigma_i^2)\Delta t, \sigma_i^2\Delta t)$$

$ln(VaR_{ci})$ represents the corresponding VaR value of $lnA_i(\Delta t)$ under confidence c_i:

$$VaR_{ci} = \frac{(\mu_i - \frac{1}{2}\sigma_i^2)\Delta t}{\sigma_i\sqrt{\Delta t}} + \Phi^{-1}(c_i) \tag{20}$$

$\sigma_i(t)(i = 1..n)$represents the standard deviation of the yield corresponding to $A_i(t)$. Introduce the weight of various assets in the asset portfolio$\alpha_i(t), (i = 1..n)$,

the VaR objective function of the asset portfolio we need to minimize is:

$$\min \sum_{i=1}^{N} \alpha_i VaR_i = \min \sum_{i=1}^{N} \alpha_i (\frac{(\mu_i - \frac{1}{2}\sigma_i^2)t}{\sigma_i\sqrt{t}} + \Phi^{-1}(c_i)). \tag{21}$$

Finally,we get two objective functions (17) and (21). What we concern about is how to choose the α_i. The other variables in the equation can be got and calculated in advance.