Data-Efficient Machine Learning with Multiple Output Types and High Input Dimensions

Zhang Yehong
(B.Eng., HIT)

A thesis submitted for the degree of
Doctor of Philosophy

Department of Computer Science, School of Computing
National University of Singapore

Aug 2017

Supervisors:
Dr Low Kian Hsiang, Main Supervisor
Professor Mohan S Kankanhalli, Co-Supervisor

Examiners:
Professor Leong Tze Yun
Dr Harold Soh Soon Hong
Professor Patrick Jaillet, Massachusetts Institute of Technology

Data-efficient machine learning with multiple output types and high input dimensions

by

Zhang Yehong

Declaration

I hereby declare that the thesis is my original work and it has been written by me in its entirety. I have duly acknowledged all the sources of information which I have been used in the thesis. This thesis has also not been submitted for any degree in any university previously.

Zhang Yehong August 24, 2017

Abstract

Recent research works in machine learning (ML) have focused on learning some target variables of interest to achieve competitive (or state-of-the-art) predictive performance in less time but without requiring large quantities of data, which is known as *data-efficient ML*. This thesis focuses on two highly related data-efficient ML approaches: *active learning* (AL) and *Bayesian optimization* (BO) which, instead of learning passively from a given set of data, need to select and gather the *most informative* observations for learning the target variables of interest accurately given some budget constraints. In particular, this thesis aims to (a) exploit the auxiliary types of outputs which correlate with the target variables for improving the learning performance of the target output type in both AL and BO algorithms and (b) scale up the state-of-the-art BO algorithm to high input dimensions.

To achieve above-mentioned objectives, an AL algorithm of *multi-output Gaussian process* (MOGP) is first developed for minimizing the predictive uncertainty (i.e., posterior joint entropy) of the target output type. In contrast to existing works, our AL problems involve selecting not just the most informative sampling inputs to be observed but also the types of outputs at each selected input for improving the learning performance of *only* the target output type given a sampling budget. Unfortunately, such an entropy criterion scales poorly in the numbers of candidate inputs and selected observations when optimized. To resolve this issue, we exploit a structure common to sparse MOGP models for deriving a novel AL criterion. Furthermore, we exploit a relaxed form of submodularity property of our new criterion for devising a polynomial-time approximation algorithm that guarantees a constant-factor approximation of that achieved by the optimal set of selected observations. Empirical evaluation on real-world datasets shows that our proposed approach outperforms existing algorithms for AL of MOGP and single-output GP models.

Secondly, to boost the BO performance by exploiting the cheaper or less noisy observations of some auxiliary functions with varying fidelities, we proposed a novel generalization of *predictive entropy search* (PES) for multi-fidelity BO called *multi-fidelity PES* (MF-PES). In contrast to existing multi-fidelity BO algorithms, our proposed MF-PES algorithm can naturally trade off between exploitation vs. exploration over the target and auxiliary functions with varying fidelities without needing to manually tune any such parameters. To achieve this, we first model the unknown target and auxiliary functions jointly as a *convolved multi-output Gaussian process* (CMOGP) whose convolutional structure is then exploited for deriving an efficient approximation of MF-PES. Empirical evaluation on synthetic and real-world experiments shows that MF-PES outperforms the state-of-the-art multi-fidelity BO algorithms.

Lastly, to improve the BO performance in real-world applications with high input dimensions (e.g., computer vision, biology), we generalize PES for high-dimensional BO by exploiting an additive structure of the target function. New practical constraints are proposed and approximated efficiently such that the proposed acquisition function of *additive PES* (add-PES) can be optimized independently for each local and low-dimensional input component. The empirical results show that our add-PES considerably improves the performance of the state-of-the-art high-dimensional BO algorithms by using a simple and common setting for optimizing different tested functions with varying input dimensions, which makes it a superior alternative to existing high-dimensional BO algorithms.

Acknowledgements

I would like to first express my sincere and deepest gratitude to my advisors, A/Prof. Low Kian Hsiang and Prof. Mohan S Kankanhalli, for providing continuous support, inspiring guidance and valuable advices during my whole Ph.D. study. I would also like to thank the thesis committee members, Prof. Leong Tze Yun, A/Prof. Harold Soh Soon Hong, Prof Patrick Jaillet and Prof. Lee Wee Sun, for their valuable feedbacks.

I would like to thank all my friends and lab mates. My Ph.D. journey would not have been as enjoyable without your companionship. I would like to especially thank Dr. Hoang Trong Nghia for his valuable time and great suggestions shared with me.

I acknowledge Sensor-enhanced Social Media Centre (SeSaMe) for offering me financial support.

Lastly, I would like to thank my parents, for all the love, understanding and encouragement they give me during my study abroad.

This thesis is dedicated to my family.

Their unconditional love and support is invaluable.

Contents

List of Symbols

$\langle x, i \rangle$ An input tuple (i.e., a tuple of input and its corresponding output type)

$\mu^{(i)}(x^{(i)})$ Prior mean function of the i^{th} component in an additive model

$\mu^{(i)}_{A^{(i)}}$ A vector of prior mean for any $A^{(i)} \subset D^{(i)}$

μ_A A vector defined by $\mu_A \triangleq (\mu_{\langle x,i \rangle})^{\top}_{\langle x,i \rangle \in A}$ for any $A \subset D^{+}$

$\mu_{\langle x,i \rangle}$ Prior mean for the output of $\langle x, i \rangle$

$\mu_{A|B}$ Posterior mean vector for the output of A given the observations of B for any $A, B \subset D^{+}$

$\phi(x)$ A feature vector of input x

$\phi^{(i)}(x^{(i)})$ An m-dimensional feature vector for the i^{th} component in an additive model

$\phi_i(x)$ An m-dimensional feature vector of input x for output type i

ρ_i The fidelity of the i^{th} (auxiliary) function with respect to the target function

$\sigma^2_{n_i}$ The noise variance of output type i

$\sigma^2_{s_i}$ The signal variance of output type i

$\sigma^{(i)}(x^{(i)}_p, x^{(i)}_q)$ Covariance function of the i^{th} component in an additive model

$\Sigma^{(i)}_{AA'}$ A covariance matrix of the i^{th} component in an additive model for any $A, A' \subset D^{(i)}$

$\Sigma^{(i)}_{AA|B}$ A posterior covariance matrix of the i^{th} component in an additive model for any $A \subset D^{(i)}$ and $B \subset D$

$\sigma_{\langle x,i\rangle\langle x',j\rangle}$ Covariance between the output measurements of $\langle x, i\rangle$ and $\langle x', j\rangle$

$\Sigma_{AA'}$ A covariance matrix defined by $\Sigma_{AA'} \triangleq (\sigma_{\langle x,i\rangle\langle x',j\rangle})_{\langle x,i\rangle\in A,\langle x',j\rangle\in A'}$ for any $A, A' \subset D^+$

$\Sigma_{AA|B}$ Posterior covariance matrix of input tuples A given the observations of B for any $A, B \subset D^+$

ε_i An additive noise of output type i

$a^{[s]}$ The s^{th} sample of a where a could be any variable or function

C The number of local functions in an additive model

D A d-dimensional input domain, $D \subset \mathbb{R}^d$

d The dimension of input x

D^+ The domain of input tuples $\langle x, i\rangle$ with $i = 1, ..., M$

$D^{(i)}$ The input domain of the i^{th} local function in an additive model, $D^{(i)} \subset \mathbb{R}^{d_i}$

d_i The dimension of i^{th} input component $x^{(i)}$ in an additive model

D^+_i A subset of D^+ for output type i

$f(x)$ An unknown function when only one type of output exists (i.e., M=1)

$f^{(i)}(x^{(i)})$ The i^{th} local function of $f(x)$ in an additive model

$f_i(x)$ An unknown function of output type i

$K_i(x)$ A kernel function of output type i

$L(x)$ A latent function

L_U A vector of latent measurements evaluated at the inducing set U

M The number of output types

m The dimension of random features

N The number of observations to be selected (i.e.,sampling budget)

t The index of the primary type of output or target function

U A set of inducing inputs

V A finite set of input tuples of all types of outputs

V_i A finite set of input tuples of output types i

X A set of input tuples $\langle x, i \rangle$ (when $M > 1$) or inputs x (when $M = 1$)

x A column input vector, $x \in D$

$X^{(i)}$ A set of $x^{(i)}$ in an additive model

$x^{(i)}$ The i^{th} input component of x

$x_*^{(i)}$ The global optimizer of the i^{th} local function in an additive model

X_i A subset of X with output type i

x_{i*} The global optimizer of the i^{th} function $f_i(x)$

Y_A A vector of random output measurements for any $A \subset D^+$

y_A	A vector of realized output measurements for any $A \subset D^+$ or $A \subset D$
$Y_{\langle x,i \rangle}$	A random output measurement when x is unobserved for output type i
$y_{\langle x,i \rangle}$	A realized output measurement when x is observed for output type i

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

For many budget-constrained applications (i.e., in terms of computations and/or sensing/data gathering) in the real-world, data-efficient machine learning (ML) [ICMLws, 2016] is an attractive, frugal alternative to learning from a massive amount of data (hence, prohibitively costly). Different from the latter, data-efficient ML needs to address the challenge of learning about some target variables of interest in a complex domain accurately and efficiently without requiring large quantities of data. For example, *semi-supervised learning* aims to build good classifiers given only a small amount of labeled data by exploiting the large amount of unlabeled data [Zhu, 2005]. *Transfer learning* improves the performance of a target task by transferring knowledge from related task domains when the target task has insufficient training data [Pan and Yang, 2010].

Instead of learning passively from a given small set of data as in the examples above, some data-efficient ML approaches (e.g., active learning and Bayesian optimization) need to select and gather the *most informative* observations for learning

the target variables of interest more accurately given some budget constraints (e.g., mission time). This thesis will focus on such approaches and aims to generalize their strategies to more practical settings: multiple types of output and high input dimensions.

Firstly, most existing data-efficient ML algorithms are designed to select and gather the observations from only one type of output. In practice, however, the primary type of output represented by the target variable often coexists and correlates well with some auxiliary type(s) of outputs which may be less noisy (e.g., due to higher-quality sensors), and/or less tedious to sample (e.g., due to greater availability/quantity, higher sampling rate, and/or lower sampling cost of these type(s)) and can consequently be exploited for improving the learning performance of the target variables. Such multiple types of correlated outputs exist in many real-world application domains, for example:

- *Environmental sensing and monitoring.* Environmental sensing refers to the task of sensing, modeling, and predicting large-scale, spatially correlated environmental phenomena. In many environmental sensing applications, measurements from auxiliary type(s) of phenomena can be exploited to improve the prediction of the target phenomenon. For example, to monitor soil pollution by some heavy metals (e.g., Cadmium), its complex and time-consuming extraction from soil samples can be alleviated by supplementing its prediction with correlated auxiliary types of soil measurements (e.g., pH) that are easier to sample [Goovaerts, 1997]. Similarly, to monitor algal bloom in the coastal ocean, plankton abundance correlates well with auxiliary types of ocean measurements (e.g., chlorophyll a, temperature, and salinity) that can be sampled more readily [Apple *et al.*, 2008].

- *Automatic ML.* Recently, a growing number of works have focused on develop-

ing ML methods that select features, workflows, ML paradigms, algorithms and their hyperparameters automatically such that they can be used easily without expert knowledge [ICMLws, 2015]. This is usually achieved by optimizing an unknown target function whose input and output are constituted by the settings to be selected and the validation accuracy, respectively. However, the target function is sometimes very expensive to evaluate due to large training dataset or/and complex model structure. In practice, the expensive-to-evaluate target function often correlates well with some auxiliary function(s) of varying fidelities (i.e., degrees of accuracy in reproducing the target function) that may be less noisy and/or cheaper to evaluate and can thus be exploited to boost the automatic ML performance. For example, automatically tuning the hyperparameters of a sophisticated ML model (e.g., deep neural network) is usually time-consuming as it may incur several hours to days to evaluate the validation accuracy of the ML model at each selected setting of hyperparameters when training with a massive dataset. To accelerate this process, one may consider a low-fidelity auxiliary function with the same inputs (i.e., hyperparameters) and output (i.e., validation accuracy) as the target function except that its validation accuracy is evaluated by training the ML model with a small subset of the dataset, hence incurring less time [Swersky *et al.*, 2013]. Similarly, the parameter setting/configuration of a real robot [Lizotte *et al.*, 2007; Tesch *et al.*, 2013] can be calibrated faster by simulating its motion in a low-fidelity but low-cost and noise-free simulation environment [Cutler *et al.*, 2015].

Other examples of real-world applications with multiple output types include remote sensing [Atkinson *et al.*, 2000], traffic monitoring [Chen *et al.*, 2012b], monitoring of groundwater [Passarella *et al.*, 2003], monitoring of indoor environmental quality, and precision agriculture [Webster and Oliver, 2007] as well as that pertaining to the Web

such as natural language processing [Reichart *et al.*, 2008] and recommender systems [Zhao *et al.*, 2013], among others.

All of the above practical applications motivate the need to design and develop data-efficient ML algorithms that can exploit the correlation between different types of outputs such that the target variables of interest can be learned more accurately and/or more efficiently, which is one focus of this thesis.

Secondly, some approaches for data-efficient ML succeeded only in low-dimensional (usually < 10) input space [Shahriari *et al.*, 2016] and cannot perform well in the real-world applications which have high-dimensional input space. For example, in the *automatic ML* problems mentioned above, the dimension of the (hyper)parameters to be automatically tuned are usually high in applications such as computer vision [Bergstra *et al.*, 2013], biology [González *et al.*, 2014] and robotics control [Calandra, 2017]. Since the number of samples needed for globally optimizing a target function usually grows exponentially in the input (i.e., hyperparameters) dimensions, this poses a significant technical challenge to apply automatic ML algorithms efficiently to such applications with high input dimensions, which is another focus of this thesis.

1.2 Objective

Among all the data-efficient ML approaches, two specific problems which need to actively select and gather observations are discussed in this thesis:

1. *Active learning (AL) for environmental sensing.* In this problem, the AL algorithm aims to select and gather the most informative observations for modeling and predicting the spatially varying phenomenon given some sampling budget constraints (e.g., quantity of deployed sensors, energy consumption).

2. *Bayesian optimization (BO) for data-efficient black-box optimization.* BO is a

global optimization algorithm which has recently demonstrated with notable success in optimizing an unknown (possibly noisy, non-convex, and/or with no closed-form expression/derivative) target function by sampling a finite budget of often expensive function evaluations [Shahriari *et al.*, 2016].

To achieve the above-mentioned goal for these two data-efficient ML problems, the works in this thesis will first attempt to exploit the correlations between multiple types of outputs for improving the performance of AL and BO algorithms, and then try to generalize a state-of-the-art BO algorithm to high-dimensional input space. To be specific, there are four key objectives in this thesis that pertain to the need to account for multiple output types and high input dimensions for AL and BO:

- *Modeling data with multiple output types or high input dimensions.* In order to learn the target variables with observations from multiple types of outputs or high input dimensions, models that can capture such structures of the data are required. The outputs of these models should be used to design algorithms for the AL and BO problems that we are interested. More importantly, for the problems that are difficult/expensive to solve with an exact algorithm, we would like to exploit the structure of the model for (a) deriving a tractable and efficient approximation algorithm and (b) guaranteeing the performance of our approximation algorithm if possible.

- *Being effective in terms of achieving good multi-output AL performance.* In a multi-output AL algorithm, we aim to design and develop an AL criterion that can be used to select not just the *most informative* sampling inputs to be observed but also the types of outputs at each selected input for minimizing the predictive uncertainty of unobserved areas of the primary type of output. Note that our focus here differs from multivariate spatial sampling algorithms [Bueso *et al.*, 1999; Le *et al.*, 2003] that aim to improve the prediction of *all*

types of outputs, for which existing AL algorithms for sampling observations only from the target variables can be extended and applied straightforwardly, as detailed in Section 4.2. In addition, since multiple types of outputs increase the search space for AL, the multi-output AL criterion is expected to scale well in the number of candidate sampling outputs and even more so in the number of observations to be selected.

- *Designing multi-fidelity acquisition function of BO.* Conventionally, a BO algorithm relies on some choice of acquisition function as a heuristic to guide its search for the global target maximizer by sampling observations from *only* the target function. Existing acquisition functions include improvement-based [Shahriari *et al.*, 2016] such as probability of improvement or *expected improvement* (EI) over currently found maximum, information-based [Villemonteix *et al.*, 2009] such as *entropy search* (ES) [Hennig and Schuler, 2012] and *predictive entropy search* (PES) [Hernández-Lobato *et al.*, 2014], and *upper confidence bound* (UCB) [Srinivas *et al.*, 2010]. To boost the BO performance by exploiting multiple types of outputs from auxiliary functions of varying fidelities (i.e., degrees of accuracy in reproducing the target function) that may be less noisy and/or cheaper to evaluate, this thesis aims to design and develop a *multi-fidelity* BO acquisition function that can be used to select not just the most informative inputs but also the target and/or auxiliary functions with varying fidelities to be evaluated at each selected input for finding or improving the belief of the global target maximizer. Note that we use multi-fidelity BO instead of multi-output BO to be consistent with the existing BO literature related to multiple output types [Kandasamy *et al.*, 2016].

- *Designing BO acquisition function for high input dimensions.* PES is a state-of-the-art BO acquisition function which has been shown to outperform the other

BO algorithms in low-dimensional (up to 8) input spaces [Hernández-Lobato *et al.*, 2014]. However, the computational cost for evaluating and optimizing the PES acquisition function grows exponentially in the number of dimensions for input space, as detailed in Section 6.1. To make progress in high-dimensional BO, we aim to extend the state-of-the-art BO algorithm (i.e., PES) to high-dimensional input setting such that it scales well in the number of input dimensions.

1.3 Contributions

To achieve the above-mentioned objectives, this thesis models the data using some form of *Gaussian process* (GP)-based probabilistic regression models which can characterize the structures of multiple output types or high input dimensions. Unlike the non-probabilistic regression methods (e.g., multivariate linear regression [Izenman, 1975], multi-output support vector regression [Sánchez-Fernández *et al.*, 2004], regularization methods [Evgeniou and Pontil, 2004] and neural network) which can handle multiple output types and/or high input dimensions, the probabilistic GP-based models allow the predictive uncertainty of the outputs to be formally quantified (e.g., based on entropy or mutual information criterion) and consequently exploited for developing the AL and BO algorithms. The novel contributions for this thesis are summarized below:

1.3.1 Multi-output active learning

In the environmental sensing problem, all types of correlated outputs (i.e., target and auxiliary) is firstly modeled jointly as a *multi-output Gaussian process*[1] (MOGP)

[1]One may argue for a simpler alternative of using the observations of the auxiliary output types as additional input features to a single-output GP modeling the primary output. This is, however,

[Álvarez and Lawrence, 2011; Bonilla *et al.*, 2007b; Osborne *et al.*, 2008; Teh and Seeger, 2005; Williams *et al.*, 2009], which allows the spatial correlation structure of each type of output and the cross-correlation structure between different types of outputs to be formally characterized.

To the best of our knowledge, this work is the first to present an efficient algorithm for active learning of a MOGP model (Chapter 4). We consider utilizing the entropy criterion to measure the predictive uncertainty of the primary output, which is widely used for active learning of a single-output GP model. Unfortunately, for the MOGP which models multiple types of output, such a criterion scales poorly in the number of candidate sampling inputs of the primary output type (Section 4.2) and even more so in the number of selected observations (i.e., sampling budget) when optimized (Section 4.3). To resolve this scalability issue, we first exploit a structure common to a unifying framework of sparse MOGP models (Section 3.2.2) for deriving a novel active learning criterion (Section 4.2). Then, we define a relaxed notion of submodularity[2] called ϵ-submodularity and exploit the ϵ-submodularity property of our new criterion for devising a polynomial-time approximation algorithm that guarantees a constant-factor approximation of that achieved by the optimal set of selected observations (Section 4.3). Then, we empirically evaluate the performance of our proposed algorithm using three real-world datasets (Section 4.4).

1.3.2 Multi-fidelity Bayesian optimization (BO)

We present a novel generalization of PES for multi-fidelity BO, which we call *multi-fidelity PES* (MF-PES) (Chapter 5). In existing BO algorithms, the chosen acqui-

not feasible in practice: Such observations have to be known/specified for GP prediction, which is not the case since they need to be sampled, just like that of the target variables.

[2]The original notion of submodularity has been used in [Krause and Golovin, 2014; Krause and Guestrin, 2007; Krause *et al.*, 2008] to theoretically guarantee the performance of their algorithms for active learning of a *single-output* GP model.

sition function is always able to trade off between sampling at or near to a likely target maximizer based on a GP belief of the unknown target function (exploitation) vs. improving the GP belief (exploration) until the budget is expended. In contrast to the state-of-the-art multi-fidelity BO algorithms such as multi-fidelity GP-UCB [Kandasamy *et al.*, 2016], multi-fidelity sequential kriging optimization [Huang *et al.*, 2006], and multi-task ES [Swersky *et al.*, 2013], our proposed MF-PES algorithm can jointly and naturally optimize such a non-trivial exploration-exploitation trade-off without needing to manually tune any such parameters or that of EI to perform well in different real-world applications, as detailed in Sections 5.2 and 5.3.

To achieve this, we model the unknown target and auxiliary functions jointly as a *convolved MOGP* (CMOGP) [Álvarez and Lawrence, 2011] whose convolutional structure is exploited to formally characterize the fidelity of each auxiliary function through its cross-correlation with the target function (Section 5.1). Although the exact acquisition function of MF-PES cannot be computed in closed form, the main contribution of our work here is to show that it is in fact possible to derive an efficient approximation of MF-PES via (a) a novel *multi-output random features* (MRF) approximation of the CMOGP model whose cross-correlation structure between the target and auxiliary functions can be exploited for improving the belief of the target maximizer using the observations from evaluating these functions (Section 5.3.1), and (b) practical constraints relating the global target maximizer to that of the auxiliary functions (Section 5.3.2). We empirically evaluate the BO performance of our MF-PES algorithm on synthetic and real-world experiments (Section 5.4).

1.3.3 High-dimensional BO

To scale the state-of-the-art BO algorithm to high input dimensions, we introduce a novel generalization of PES to high-dimensional BO, which we call *additive PES*

(add-PES) (Chapter 6). Compared to *additive GP-UCB* (add-GP-UCB) [Kandasamy *et al.*, 2015] which is a state-of-the-art high-dimensional BO algorithm, our add-PES is considerably less tedious (manually) for finding an appropriate exploration-exploitation trade-off to achieve better BO performance for different functions with high input dimensions, as detailed in Sections 6.1 and 6.3. In contrast to the high-dimensional *max-value ES* (MES) [Wang and Jegelka, 2017] which approximates the PES acquisition function by replacing the uncertainty of target maximizer as that of the maximal function value, our add-PES algorithm avoids such further approximation and is defined according to the original PES acquisition function which aims to reduce the uncertainty of the target maximizer, and thus promising to achieve better accuracy (Section 6.2).

To achieve this, the unknown target function with high input dimensions is modeled using additive GP (Section 3.3) whose additive structure is exploited to improve the scalability of PES in the number of input dimensions by optimizing some local functions independently (Section 6.2). Unfortunately, the approximation of original PES cannot be applied to each function component straightforwardly since some of the constraints used for approximating PES can only be defined over the sum of all function components. To resolve this issue, we propose novel steps for approximating the add-PES acquisition function efficiently (Section 6.2.2). Finally, we empirically evaluate the performance of our proposed add-PES algorithm using synthetic functions with varying high dimensions of input (Section 6.3).

1.4 Organization

The remaining chapters of this thesis are organized as follows. Section 2.1 will first briefly review existing data-efficient ML approaches including AL and BO that we are interested. Then, data-efficient ML literature that is related to multiple output

types and high input dimensions will be presented in Section 2.2 and Section 2.3, respectively. Chapter 3 provides the technical details of modeling data with multiple types of outputs and high-dimensional input with MOGP and additive GP, respectively. A brief review of existing MOGP models is also included to identify the advantage of CMOGP which is selected in this thesis. Chapter 4 and 5 will focus on designing AL and BO algorithms for multiple output types. In particular, a novel near-optimal multi-output active learning method and its experimental results will be presented in Chapter 4. The technical and experimental results of our proposed multi-fidelity PES algorithm will be shown in Chapter 5. Then, the state-of-the-art BO acquisition function (i.e., PES) is generalized for optimizing unknown functions with high-dimensional input in Chapter 6. Finally, the conclusion and the future works of this thesis will be presented in Chapter 7.

Chapter 2

Related Works

This chapter reviews the literature related to data-efficient ML. First, existing approaches that belongs to data-efficient ML family will be briefly introduced in Section 2.1. Then, we will focus on the literature that is closely related to the research problems proposed in this thesis: active learning (AL) and Bayesian optimization (BO) algorithms with multiple output types and high input dimensions. In particular, Section 2.2.1 focuses on the literature of multi-output AL. Section 2.2.2 reviews existing multi-fidelity (i.e., multi-output) BO algorithms and discusses the advantage of our proposed method compared to the state-of-the-art ones. Then, the BO algorithms designed for high input dimensions will be briefly reviewed in Section 2.3. Compared to all the literature reviewed in this chapter, our proposed algorithms are either designed for different problem settings or technically much easier to apply for different real-world applications, as detailed later.

2.1 Data-Efficient Machine Learning

Instead of learning from massive amount of data, data-efficient ML is a family of ML approaches which aims to learn about some target variables of interest in a complex

domain accurately and efficiently with limited (labeled) data [ICMLws, 2016]. Such approaches are required in many real-world application domains such as environmental sensing, automatic ML and personalized learning which are difficult to collect a large amount of observations due to the given budget constraints (e.g., energy consumption, mission time, no. of participants).

A wide range of existing ML approaches have been designed to resolve the small-data issue. For example, *semi-supervised learning* aims to build good classifiers using a small set of labeled data by exploiting the large amount of unlabeled data [Zhu, 2005]. *Transfer learning* techniques improve the learning performance of a target task by transferring knowledge from related task domains when the target task has insufficient training data [Pan and Yang, 2010]. *One-shot learning* is a problem that aims to learn the object categories from only one, or a handful, of training images [Fei-Fei *et al.*, 2006]. The other approaches that can belong to the family of data-efficient ML include bootstrapping, data augmentation, Bayesian deep learning, non-parametric methods, etc [ICMLws, 2016].

All above-mentioned approaches learn the target variables *passively* from a given small set of data. In some problems, however, the observations or labeled data are not available at the beginning. It requires the algorithm to *actively* select and gather the most informative observations for learning the target variables accurately and efficiently given a budget constraint. *Active learning* (AL) (sometimes known as "optimal experimental design in statistics literature) and *Bayesian optimization* (BO) have been designed to achieve this goal, and thus, are important approaches of the data-efficient ML family and will be the focus of this thesis.

A comprehensive literature review of AL can be found in Settles (2010). This thesis will focus on AL of environmental sensing and monitoring applications and aim to develop AL algorithm for multiple types of correlated environmental phenomena (i.e., multi-output AL). Different from the other application domains (e.g., image annota-

tion and retrieval [Wang and Hua, 2011], recommendation [Zhao *et al.*, 2013]) that are rarely modeled using GP, the spatially varying target phenomenon in the interested applications can be formally characterized by a probabilistic GP regression model whose predictive uncertainty will be exploited for deriving efficient AL algorithms. The literature related to multi-output AL will be reviewed in Section 2.2.1.

BO has been shown to succeed in globally optimizing a black-box function whose derivatives and convexity properties are unknown [Brochu *et al.*, 2010; Snoek *et al.*, 2012]. Usually, the target function optimized by BO is very expensive to evaluate such that only a small number of observations can be sampled, which is same as the key issue of data-efficient ML. To achieve this, a BO algorithm conventionally optimizes some choice of acquisition function to iteratively select the next input to evaluate the unknown function. Examples of acquisition functions include *probability of improvement* (PI), *expected improvement* (EI) [Brochu *et al.*, 2010], *entropy search* (ES) [Hennig and Schuler, 2012], *predictive entropy search* (PES) [Hernández-Lobato *et al.*, 2014], and *upper confidence bound* (UCB) [Srinivas *et al.*, 2010]. A comprehensive introduction and literature reviews of BO can be found in Brochu *et al.* (2010) and Shahriari *et al.* (2016). The remaining sections of this chapter will focus on BO algorithms designed for multiple output types or high input dimensions, whose related literature will be reviewed in Section 2.2.2 and Section 2.3, respectively.

2.2 Data-Efficient Multi-Output Machine Learning

In this section, we will review AL and BO literature that related to the main focus of this work: multiple output types and high input dimensions.

2.2.1 Multi-output active learning

Firstly, as has been mentioned in Chapter 1.2, our multi-output AL (MOAL) algorithm is designed to select the most informative observations from all output types

for minimizing the predictive uncertainty of *only* the primary type of output. This objective differs from multivariate spatial sampling algorithms [Bueso *et al.*, 1999; Le *et al.*, 2003] in geostatistical literature which aim to improve the predictive accuracy of *all* types of outputs.

Then, existing works on AL with multiple output types are not driven by the MOGP model and have not formally characterized the cross-correlation structure between different types of outputs: Some spatial sampling algorithms [Bueso *et al.*, 1998; Angulo and Bueso, 2001] have simply modeled the auxiliary output as a noisy perturbation of the primary output that is assumed to be latent, which differs from our work here. *Multi-task active learning* (MTAL) and *active transfer learning* (ATL) algorithms have considered the prediction of each type of output as one task and used the auxiliary tasks to help learn the target task. However, the MTAL algorithm of Zhang (2010) requires the relations between different classification tasks to be manually specified, which is highly non-trivial to achieve in practice and not applicable to MOGP regression. The ATL algorithm of Wang *et al.* (2014) has used only a single-output GP to model and predict the offset between different tasks, which may not represent a complex cross-correlation structure well (e.g., each type of coexisting phenomena is an additive combination of blurred versions of some latent ones). Some other ATL and AL algorithms [Roth and Small, 2006; Shi *et al.*, 2008; Zhao *et al.*, 2013; Zhu *et al.*, 2011] have used active learning strategies (e.g., margin-based criterion) specific to their classification or recommendation tasks that cannot be readily tailored to MOGP regression.

Another research area related to multi-output AL is the *optimal sensor scheduling* [Mourikis and Roumeliotis, 2006; Hero and Cochran, 2011; Wu *et al.*, 2014; Tzoumas *et al.*, 2016; Han *et al.*, 2017] in control community, which is designed to schedule the usage of multiple types of sensors due to the resources (i.e., communication bandwidth, battery power) limitation. The algorithms in this area differ from

our proposed method in both aspects mentioned above: Firstly, the sensor scheduling algorithms usually fuse the measurements of all selected sensors for some particular objectives such as localization, tracking or decision making [Hero and Cochran, 2011]. To achieve this, the observed or predictive measurements of all types of sensors instead of just the target phenomena as in our method are considered in formalizing the objective. Secondly, to the best of our knowledge, the sensor scheduling algorithms have assumed independent observation models for each type of sensor [Wu *et al.*, 2014; Tzoumas *et al.*, 2016; Han *et al.*, 2017]. The cross-correlations between different sensors are not formally characterized such that the measurements of the selected sensors cannot be exploited for improving the predictive performance of the correlated sensor measurements that are not observed at the current step.

2.2.2 Multi-fidelity BO

In this thesis, we aim to develop a multi-fidelity BO algorithm that can select observations from both the target and auxiliary function(s) with varying fidelities for optimizing *only* the unknown target function accurately and efficiently. The state-of-the-art multi-fidelity BO algorithms whose objective is same as this thesis include multi-fidelity EI [Huang *et al.*, 2006; Forrester *et al.*, 2007], multi-task ES [Swersky *et al.*, 2013], and multi-fidelity GP-UCB [Kandasamy *et al.*, 2016]. However, all these algorithms require heuristically setting some parameters to trade off between exploitation vs. exploration over the target and auxiliary functions with varying fidelities, which makes them not easy to perform well in different real-world applications. Detailed descriptions of such parameters for each algorithm will be shown in Sections 5.2 and 5.3. Moreover, to approximate the designed acquisition function tractably and efficiently, the multi-task ES algorithm in [Swersky *et al.*, 2013] has to reduce the search space to a small set of input candidates selected by applying EI to only

the target function. Forrester *et al.* (2007) and Poloczek *et al.* (2016) also discretized the input domain using Latin Hypercube. Such discretization steps artificially constrain the exploration of the unknown functions especially if the input dimension is high. In contrast, the multi-fidelity BO algorithm proposed in this thesis avoid both of the above issues (i.e., artificial parameter setting and input discretization) and is much easier to achieve good performance for different functions or different real-world applications (Section 5.4).

To capture the cross-correlation between the target and auxiliary functions, recent multi-fidelity BO works [Kandasamy *et al.*, 2017; Klein *et al.*, 2017] have assumed that the target and auxiliary functions have *given* fidelity values (e.g., data size in hyperparameter tuning problem) which is used to compute the cross-correlation between multiple functions through a kernel function. The collaborative hyperparameter tuning algorithms [Bardenet *et al.*, 2013; Yogatama and Mann, 2014] have similarly, presented the correlated datasets used for different functions with a vector of numerical descriptors. In practice, it is usually difficult to manually and correctly specify such fidelity values or descriptors for any application, especially since the true fidelity value of an auxiliary function is usually unknown. For example, one application of multi-fidelity BO is to accelerate the parameters calibration of robot using the low-fidelity simulations [Marco *et al.*, 2017], where the cross-correlation between the real robot and the simulation is difficult to be measured using any given values/descriptors. Our work instead defines a principled fidelity measure using CMOGP (Section 5.3) that can be learned from data for any application and hence easily exploited by future developments of multi-fidelity BO algorithms.

All the other existing BO works that used multiple types of outputs are of different objective from ours: The algorithms in [Bardenet *et al.*, 2013; Yogatama and Mann, 2014; Feurer *et al.*, 2015] have either assumed the optimizers or observations of the auxiliary tasks to be known or tried to optimize both the target and auxil-

iary functions simultaneously. Multi-objective BO algorithms [Zuluaga *et al.*, 2013; Hernández-Lobato *et al.*, 2016] have tried to simultaneously optimize multiple tasks whose objectives may conflict with each other. Since there is usually no single optimal solution that excels in all tasks, multi-objective BO is therefore interested in identifying an entire set of Pareto-optimal solutions with optimal trade-offs of different tasks, which is not the focus of our work in this thesis.

2.3 High-Dimensional BO

As has been mentioned in Section 1.1, high-dimension input is an important but difficult issue of BO for advancing the state-of-the-art because the number of samples required for globally optimizing a function usually grows exponentially in the input dimensions. To resolve this issue, the early works of high-dimensional BO usually assumed "low effective dimensionality" (i.e., a low dimensional subspace that the unknown function varies in) and used methods such as *random embeddings* [Wang *et al.*, 2013], *sequential likelihood ratio test* [Chen *et al.*, 2012a] and *low rank matrix recovery* [Djolonga *et al.*, 2013] to select the relevant/active ones among all the input dimensions. Such assumption, however, is sometimes too strong and results in bad performance in real-world applications whose unknown function varies along all the input dimensions [Kandasamy *et al.*, 2015].

Recent research works [Kandasamy *et al.*, 2015; Wang and Jegelka, 2017; Wang *et al.*, 2017] have focused on resolving the high-dimensional issues by assuming an additive structure of the unknown function, which is more general than the "low effective dimensionality" assumption. Rana *et al.* (2017) proposed to optimize the acquisition function globally in high-dimension by choosing large to small length-scales for GP without requiring any assumption on the structure of the underlying function. However, most of the above algorithms [Kandasamy *et al.*, 2015; Wang *et*

al., 2017; Rana *et al.*, 2017] are based on UCB and/or EI acquisition functions which are difficult to parameterize to better trade off between exploration vs. exploration in BO process. For example, UCB-based algorithm is very easy to trap into some local optimizers by using the parameter recommended in the paper, as observed from our experiments (Section 6.3). Moreover, the exact parameter for a good exploration-exploitation trade-off is not clear for a given application since multiple variables (e.g., input dimension, no. of iterations, etc.) which related to the application are involved.

As will be shown later in Chapter 6, PES-based BO algorithm could obtain a very good exploratory behavior by simply setting a common parameter for different applications, which is important for optimizing a function *globally* in high input dimensions. Although Wang and Jegelka (2017) have proposed an additive *max-value entropy search* (add-MES) algorithm for approximating the state-of-the-art PES acquisition function in high dimensions, their algorithm doesn't perform comparable to the real PES (Section 6.3) since an additional approximation is introduced by replacing the target maximizer with the maximal value, which makes their objective deviated from the original PES (i.e., reducing the predictive uncertainty of the target maximizer). In contrast, our proposed high-dimensional BO algorithm will be a generalization of the real/original PES acquisition function which is easy to decide its parameter for achieving good exploration-exploitation trade-off, and thus, much easier to perform well in different real-world applications.

2.4 Summary

As can be seen from above sections, even though there are already some AL and BO literature about multiple output types and high input dimensions, they are always (a) of different objective, (b) bad in exploiting the input/output structures of the unknown function(s) due to some strong (independent) assumptions or (c) technically

difficult to perform well in different real-world applications due to the requirement of artificially setting parameters. To advance the state-of-the-art of data-efficient ML, AL and BO techniques that can fill in all above gaps about multiple output types and high input dimensions are required. To achieve this, we will first introduce the technical details of the underlying models which can formally characterize the input/output structure of our problems (Chapter 3), and thus, are used in this thesis. Then, novel AL and BO algorithms for multiple output types and high input dimensions will be proposed in Chapters 4, 5 and 6, as detailed later.

Chapter 3

Background and Notation

In order to achieve the objectives mentioned in Section 1.2, this thesis will model the data using Gaussian process (GP)-based regression models whose predictive uncertainty can be exploited for deriving efficient AL and BO algorithms. This chapter starts by introducing the general idea of GP (Section 3.1). Then, we will describe the technical details of *convolved multi-output GP* (CMOGP) (Section 3.2.1) and *additive GP* (Section 3.3) which are used to model multiple output types and high input dimensions, respectively, in our proposed algorithms. To elaborate the advantage of the selected model, other existing multi-output GP algorithms will be briefly introduced and compared with CMOGP in Section 3.2.3.

3.1 Gaussian Process (GP)

A *Gaussian process* (GP) is a Bayesian non-parametric model that can be used to model an unknown function $f(x)$ as follows: Let $D \subset \mathbb{R}^d$ denote a set of sampling inputs. Then, f_D can be modeled using a GP, that is, every finite subset of $\{f(x)\}_{x \in D}$ has a multivariate Gaussian distribution [Rasmussen and Williams, 2006]. Such a GP is fully specified by its prior mean $\mu_x \triangleq \mathbb{E}[f(x)]$ and covariance $\sigma_{xx'} \triangleq \text{cov}[f(x), f(x')]$

for all $x, x' \in D$, the latter of which characterizes the correlation structure of $f(x)$ and can be defined using a covariance function. Even though a wide range of covariance functions have been studied [Rasmussen and Williams, 2006], most of them are only defined over a *single* type of output. Therefore, the key issue for modeling multiple types of outputs using GP is how to specify the prior covariance $\sigma_{xx'}$ when x and x' are sampled for different types of outputs, which will be discussed next.

3.2 Multi-Output Gaussian Process (MOGP)

A number of MOGP models such as co-kriging [Webster and Oliver, 2007], parameter sharing [Skolidis, 2012], and *linear model of coregionalization* (LMC) [Teh and Seeger, 2005; Bonilla *et al.*, 2007b] have been proposed to handle multiple types of correlated outputs. In this section, we will first introduce the *convolved MOGP* (CMOGP) model which has been empirically demonstrated in [Álvarez and Lawrence, 2011] to outperform the others and will be used in our algorithms. Then, the other multi-output prediction models will be briefly reviewed and compared with CMOGP.

3.2.1 Convolved MOGP (CMOGP)

CMOGP regression was first proposed by Boyle and Frean (2004) and then approximated with a sparse structure by Álvarez and Lawrence (2009). Let M types of outputs be defined over D such that each input $x \in D$ is associated with a noisy realized (random) output $y_{\langle x,i \rangle}$ ($Y_{\langle x,i \rangle}$) if x is observed (unobserved) for type i for $i = 1, \ldots, M$. Let $D_i^+ \triangleq \{\langle x, i \rangle\}_{x \in D}$ and $D^+ \triangleq \bigcup_{i=1}^{M} D_i^+$. Then, the output $Y_{\langle x,i \rangle}$ of type i at any input $x \in D$ can be defined as an unknown function $f_i(x)$ corrupted by an additive noise $\varepsilon_i \sim \mathcal{N}(0, \sigma_{n_i}^2)$ with noise variance $\sigma_{n_i}^2$:

$$Y_{\langle x,i \rangle} \triangleq f_i(x) + \varepsilon_i \ .$$

In a CMOGP regression model, the function $f_i(x)$ is defined as a convolution between a smoothing kernel $K_i(x)$ and a latent function $L(x)$[1]:

$$f_i(x) \triangleq \int_{x' \in D} K_i(x - x')\ L(x')\ \mathrm{d}x'. \tag{3.1}$$

As shown in [Álvarez and Lawrence, 2011], if $\{L(x)\}_{x \in D}$ is a GP, then $\{Y_{\langle x,i \rangle}\}_{\langle x,i \rangle \in D^+}$ is also a GP, that is, every finite subset of $\{Y_{\langle x,i \rangle}\}_{\langle x,i \rangle \in D^+}$ follows a multivariate Gaussian distribution. Similar to the single-output case, such a GP is also fully specified by its *prior* mean $\mu_{\langle x,i \rangle} \triangleq \mathbb{E}[Y_{\langle x,i \rangle}]$ and covariance $\sigma_{\langle x,i \rangle \langle x',j \rangle} \triangleq \mathrm{cov}[Y_{\langle x,i \rangle}, Y_{\langle x',j \rangle}]$ for all $\langle x, i \rangle, \langle x', j \rangle \in D^+$, the latter of which characterizes the inter-correlation structure for each type of output (i.e., $i = j$) and the cross-correlation structure between different types of outputs (i.e., $i \neq j$).

Specifically, let $\{L(x)\}_{x \in D}$ be a GP with prior covariance $\sigma_{xx'} \triangleq \mathcal{N}(x - x' | \underline{0}, P_0^{-1})$ and $K_i(x) \triangleq \sigma_{s_i} \mathcal{N}(x | \underline{0}, P_i^{-1})$ where $\sigma_{s_i}^2$ is the signal variance controlling the intensity of measurements of type i, P_0 and P_i are diagonal precision matrices controlling, respectively, the degrees of correlation between latent measurements and cross-correlation between latent and type i measurements, and $\underline{0}$ denotes a column vector comprising components of value 0. Then, the covariance of $\{Y_{\langle x,i \rangle}\}_{\langle x,i \rangle \in D^+}$ can be computed as follows:

$$\sigma_{\langle x,i \rangle \langle x',j \rangle} = \sigma_{s_i} \sigma_{s_j} \mathcal{N}(x - x' | \underline{0}, P_0^{-1} + P_i^{-1} + P_j^{-1}) + \delta_{xx'}^{ij} \sigma_{n_i}^2 \tag{3.2}$$

where $\delta_{xx'}^{ij}$ is a Kronecker delta of value 1 if $i = j$ and $x = x'$, and 0 otherwise.

Supposing a column vector $y_X \triangleq (y_{\langle x,i \rangle})_{\langle x,i \rangle \in X}^\top$ of realized outputs is available for some set $X \triangleq \bigcup_{i=1}^M X_i$ of observed input tuples where $X_i \subset D_i^+$, a CMOGP regression

[1]For the ease of exposition, we consider a single latent function. Note, however, that multiple latent functions can be used to improve the fidelity of modeling, as shown in [Álvarez and Lawrence, 2011]. More importantly, our proposed algorithm and theoretical results remain valid with multiple latent functions.

model can exploit these observations to provide a Gaussian predictive distribution $\mathcal{N}(\mu_{Z|X}, \Sigma_{ZZ|X})$ of the outputs for any set $Z \subseteq D^+ \setminus X$ of unobserved input tuples with the following *posterior* mean vector and covariance matrix:

$$\begin{aligned} \mu_{Z|X} &\triangleq \mu_Z + \Sigma_{ZX}\Sigma_{XX}^{-1}(y_X - \mu_X) \\ \Sigma_{ZZ|X} &\triangleq \Sigma_{ZZ} - \Sigma_{ZX}\Sigma_{XX}^{-1}\Sigma_{XZ} \end{aligned} \tag{3.3}$$

where $\Sigma_{AA'} \triangleq (\sigma_{\langle x,i\rangle\langle x',j\rangle})_{\langle x,i\rangle\in A,\langle x',j\rangle\in A'}$ and $\mu_A \triangleq (\mu_{\langle x,i\rangle})_{\langle x,i\rangle\in A}^\top$ for any $A, A' \subseteq D^+$.

3.2.2 Sparse CMOGP regression

A limitation of the CMOGP model is its poor scalability in the number $|X|$ of observations: Computing its Gaussian predictive distribution (3.3) requires inverting Σ_{XX}, which incurs $\mathcal{O}(|X|^3)$ time. To improve its scalability, a unifying framework of sparse CMOGP regression models such as the deterministic training conditional, fully independent training conditional, and *partially independent training conditional* (PITC) approximations [Álvarez and Lawrence, 2011] exploit a vector $L_U \triangleq (L(x))_{x\in U}^\top$ of inducing outputs for some small set $U \subset D$ of inducing inputs (i.e., $|U| \ll |D|$) to approximate each output $Y_{\langle x,i\rangle}$:

$$Y_{\langle x,i\rangle} \approx \int_{x'\in D} K_i(x - x')\ \mathbb{E}[L(x')|L_U]\ \mathrm{d}x' + \varepsilon_i\ .$$

They also share two structural properties that can be exploited for deriving our active learning criterion and in turn an efficient approximation algorithm in Chapter 4:

P1. Measurements of different types (i.e., $Y_{D_i^+}$ and $Y_{D_j^+}$ for $i \neq j$) are conditionally independent given L_U;

P2. Y_X and Y_Z are conditionally independent given L_U.

PITC will be used as the sparse CMOGP regression model in our work here since the others in the unifying framework impose further assumptions. With the above structural properties, PITC can utilize the observations to provide a Gaussian predictive distribution $\mathcal{N}(\mu^{\text{PITC}}_{Z|X}, \Sigma^{\text{PITC}}_{ZZ|X})$ where

$$\begin{aligned}\mu^{\text{PITC}}_{Z|X} &\triangleq \mu_Z + \Gamma_{ZX}(\Gamma_{XX} + \Lambda_X)^{-1}(y_X - \mu_X)\\ \Sigma^{\text{PITC}}_{ZZ|X} &\triangleq \Gamma_{ZZ} + \Lambda_Z - \Gamma_{ZX}(\Gamma_{XX} + \Lambda_X)^{-1}\Gamma_{XZ}\end{aligned} \tag{3.4}$$

such that $\Gamma_{AA'} \triangleq \Sigma_{AU}\Sigma^{-1}_{UU}\Sigma_{UA'}$ for any $A, A' \subseteq D^+$, Σ_{AU} (Σ_{UU}) is a covariance matrix with covariance components

$$\sigma_{\langle x,i\rangle x'} = \sigma_{s_i}\mathcal{N}(x - x'|\underline{0}, P_0^{-1} + P_i^{-1})$$

for all $\langle x, i\rangle \in A$ and $x' \in U$ ($\sigma_{xx'}$ for all $x, x' \in U$), $\Sigma_{UA'}$ is the transpose of $\Sigma_{A'U}$, and Λ_A is a block-diagonal matrix constructed from the M diagonal blocks of $\Sigma_{AA|U} \triangleq \Sigma_{AA} - \Gamma_{AA}$ for any $A \subseteq D^+$, each of which is a matrix $\Sigma_{A_iA_i|U}$ for $i = 1, \ldots, M$ where $A_i \subseteq D^+_i$ and $A \triangleq \bigcup_{i=1}^M A_i$. Note that computing (3.4) does not require the inducing inputs U to be observed. Also, the covariance matrix Σ_{XX} in (3.3) is approximated by a reduced-rank matrix Γ_{XX} summed with the resulting sparsified residual matrix Λ_X. So, by using the matrix inversion lemma to invert the approximated covariance matrix $\Gamma_{XX} + \Lambda_X$ and applying some algebraic manipulations, computing (3.4) incurs $\mathcal{O}(|X|(|U|^2 + (|X|/M)^2))$ time [Álvarez and Lawrence, 2011] in the case of $|U| \leq |X|$ and evenly distributed observations among all M types.

3.2.3 Related works

There are some other methods that can be used to model multiple types of correlated outputs. Similar to CMOGP, the common idea of all MOGP models is to assume

that any finite number of realized ($y_{\langle x,i\rangle}$) or random ($Y_{\langle x,i\rangle}$) outputs have a joint multivariate Gaussian distribution which is specified by a mean vector μ and covariance matrix Σ. Usually, the covariance matrix Σ need to be computed using a covariance function that is determined according to both the inter-correlation of one output type and the cross-correlation between different output types, where the cross-correlations are usually captured by allowing different output types to share a certain structure over $f_i(x)$ and/or ε_i for $i = 1, ..., M$. Several approaches have been proposed for achieving this, as detailed later.

3.2.3.1 Parameter transfer models

Parameter transfer refers to the methods that model each type of output with a single-output GP and transfer information between different types of outputs by sharing some parameters. The basic assumption is that different output types are conditionally independent given some parameters:

$$p(f_1(x), ..., f_M(x)|\theta) = \prod_{i=1}^{M} p(f_i(x)|\theta)p(\theta) \tag{3.5}$$

where θ can be the hyperparameters (i.e., $\sigma_{n_i}^2$ and some parameters in the covariance function) and/or the parameters (i.e., mean μ and covariance matrix Σ of the multivariate Gaussian distribution) of the model [Skolidis, 2012].

Specifically, there are several approaches that consider θ as the variables related to the GP hyper-parameters: Multi-task informative vector machine (MT-IVM) of [Lawrence and Platt, 2004] assumed that θ in equation (3.5) is the GP hyper-parameters. The semi-supervised multi-task regression algorithm of [Zhang and Yeung, 2009] captured the cross-correlations between different types of outputs by imposing a common prior distribution over the hyper-parameters such that θ is the parameters of the prior distribution. Zhang (2010) even placed a common prior over only

the noise variance $\sigma^2_{n_i}$ of each output type to capture the cross-correlations between different output types. Some other works focus on transferring information between different output types by sharing the parameters of GP [Schwaighofer *et al.*, 2004; Yu *et al.*, 2005; Birlutiu *et al.*, 2010]. Usually, they assumed that the model parameters (i.e., mean vector μ and covariance matrix Σ) are drawn from a common hyper-prior, and placed a normal-inverse Wishart prior distribution over the covariance matrix of the multivariate Gaussian distribution :

$$p(\mu, \Sigma) = \mathcal{N}(\mu|\mu_0, \frac{1}{\pi}\Sigma)\mathcal{IW}(\Sigma|\tau, \Sigma_0) \ .$$

The shared parameters of the prior distribution (i.e., μ_0, Σ_0 and τ) and the functional values are estimated with an EM algorithm.

However, all these parameter transfer models assume that all types of outputs are highly correlated with each other, which is a very strong assumption and makes such models less effective in situations where some types of outputs are less correlated or even independent of the others.

3.2.3.2 Linear model of coregionalization (LMC)

LMC refers to models that share information between different output types via a set of latent functions. The reviews of LMC in this section will follow that of Álvarez and Lawrence (2011). Specifically, LMC models assume that the function $f_i(x)$ of each output type can be expressed as a linear combination of some independent latent functions $L(x)$:

$$f_i(x) \triangleq \sum_{q=1}^{Q} \sum_{r=1}^{R_q} a^r_{i,q} L^r_q(x) \tag{3.6}$$

where $\{L^r_q(x)\}_{r=1}^{R_q}$ represent the latent functions that share the same covariance function. If all the independent latent functions are GP, then $\{f_i(x)\}_{\langle x,i\rangle \in D^+}$ will also

be a joint GP specified by a mean function and a positive semi-definite covariance matrix Σ. In LMC, Σ can be represented as a Kronecker product of two separate matrices: (a) Σ^t that is used to model the cross-correlations between different output types, and (b) Σ^x that is used to capture the inter-correlations between the data points in one output type [Skolidis, 2012]. Usually, the kernel function for Σ^x can be chosen from the wide variety of kernel functions that are used for the single-output GP. Several approaches have been proposed to construct Σ^t in different MOGP literatures [Teh and Seeger, 2005; Bonilla *et al.*, 2007a; Bonilla *et al.*, 2007b; Osborne *et al.*, 2008]:

Semiparametric latent factor models (SLFM) proposed by Teh and Seeger (2005) turns out to be a simplified version of equation (3.6) by assuming $R_q = 1$ and some linear relationships between $a^r_{i,q}$. In particular, the covariance matrix in SLFM can be computed with:

$$\Sigma \triangleq (A \otimes I)\Sigma^x(A^\top \otimes I)$$

where I is an identity matrix, A is a matrix of parameters that related to $a^r_{i,q}$ and can be treated as Σ^t. An empirical Bayes estimation is derived to estimate the parameters, and informative vector machine (IVM) [Lawrence *et al.*, 2003] is used to reduce the computational complexity.

Some other works which turn out to be LMC intuitively assumed that the covariance matrix of a MOGP model can be expressed as:

$$\Sigma \triangleq \Sigma^t \otimes \Sigma^x + \Sigma^n \otimes I$$

where Σ^n is a diagonal matrix in which the $(i,i)^{th}$ element is the noise variance $\sigma^2_{n_i}$. Different methods have been proposed to estimate Σ^t: Bonilla *et al.* (2007a) computed Σ^t with a kernel function $k^t(t_i, t_j)$ where t_i is the task-descriptor features for the i^{th} type of output. Bonilla *et al.* (2007b) applied Cholesky decomposition to the cross-

correlation matrix: $\Sigma^t \triangleq LL^T$ and estimate L with an EM algorithm. Osborne *et al.* (2008) decompose Σ^t using completely general spherical parameterisation: $\Sigma^t \triangleq \text{diag}(g)\ s^\top s\ \text{diag}(g)$ where g gives an intuitive length scale for each environmental variable, and $s^\top s$ is the correlation matrix. Details about the relationship between these MOGP models and LMC can be found in Álvarez and Lawrence (2011) and Álvarez *et al.* (2012).

All above-mentioned LMC methods have modeled the cross-correlations between different output types via instantaneous mixing of independent latent functions. Such assumption, however, makes LMC methods difficult to model correlated output types that are blurredly correlated with each other, which is the limitation of LMC. Interestingly, the CMOGP (Section 3.2.1) used in this thesis resolved this issue by introducing convolutions to capture the cross-correlations between multiple output types [Higdon, 2002; Boyle and Frean, 2004; Álvarez and Lawrence, 2011], which is equivalent to LMC when the smoothing kernels $K_i(x - x') \triangleq a_i\delta(x - x')$ where $\delta(x)$ is the Dirac delta function.

3.2.3.3 Other models for multi-output prediction

In addition to previous MOGP models that shared certain structure or parameters to achieve multi-output prediction, researchers from computer vision community have also proposed some approaches to exploit the internal dependencies within the high-dimensional vision outputs. Twin GP [Bo and Sminchisescu, 2010] estimated multi-outputs by minimizing the Kullback-Leibler divergence between two GPs which is used to model finite index sets of training and testing examples. Rudovic and Pantic (2011) proposed shape-constrained GP based on a face-shape model to do the head-pose normalization. Although such methods are computationally efficient for high-dimensional outputs, they can only model a system where all output types must be observed at a given input, which makes them not suitable for our problem where

each type of output can be sampled from different set of inputs[2].

Except for MOGP, there are also other methods that can be used to model multiple output types: Multivariate linear regression [Izenman, 1975; Reinsel and Velu, 1998; Obozinski *et al.*, 2008; Rohde and Tsybakov, 2011], principal component analysis (PCA) for output dimension reduction [Higdon *et al.*, 2008; Brooks *et al.*, 2008] and landmark selection [Balasubramanian and Lebanon, 2012] are examples of *parametric* methods that can be used to model multiple types of correlated outputs. Such models parametrized the functions using finite number of parameters (e.g., finite number of weights) and attempted to infer these parameters from the observations, which makes the complexity of the model to be bounded even if the amount of information in the data is unbounded [Ghahramani, 2012]. In contrast, by placing a prior distribution (i.e., GP) directly over the functions $f_i(x)$ for $i = 1, ..., M$ rather than the parameters, non-parametric regression models such as MOGP allows their parameters (i.e., mean vector μ and covariance matrix Σ) to be in infinite dimension and refined with growing sample size, which makes the model more flexible.

Furthermore, the non-probabilistic multivariate regression methods (e.g., multivariate linear regression [Izenman, 1975], multi-output support vector regression [Weston *et al.*, 2002; Sánchez-Fernández *et al.*, 2004; Tuia *et al.*, 2011; Xu *et al.*, 2013], regularization methods [Evgeniou and Pontil, 2004]) usually provide *only* the most likely regression value for a given input. In contrast, the MOGP regression model that we selected is probabilistic, which allows the predictive uncertainty of the output type(s) to be formally quantified (e.g., based on entropy or mutual information criterion) and consequently exploited for selecting the next sampled input in the AL and BO algorithms.

[2]They are known as *isotopic* and *heterotopic* system respectively in geostatistics.

		Model properties			Data properties	
		Non-parametric	Probabilistic	USS	Non-IMC	HS
MLR						$\surd$
DR with PCA			$\surd-$			
LS			$\surd-$			
MOSVR		$\surd-$				$\surd$
MOGP	PT	$\surd$	$\surd$			$\surd$
	LMC	$\surd-$	$\surd$			$\surd$
	Twin GP	$\surd$	$\surd$		$\surd$	
	CMOGP	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$

Table 3.1: Comparison of selected multi-output prediction methods. (MLR: Multivariate linear regression. DR: Dimension reduction. LS: landmark selection. MOSVR: Multi-output support vector regression. PT: Parameter transfer. USS: Unifying sparse structure. Non-IMC: not an instantaneous mixed cross-correlation (i.e., cross-correlation that cannot be simply modeled via instantaneous mixing of independent latent functions, as discussed in Section 3.2.3.2). HS: Heterotopic system. The symbol '$\surd$' means that this type of model has the required model property or can deal with the data property. The symbol '$\surd-$' means that some variations of this model have the required model property and the others do not have.

3.2.4 Summary

A brief summary of multi-output regression models mentioned above is shown in Table 3.1. As can be seen from the table, we propose to use CMOGP for modeling multiple types of outputs in our proposed AL and BO algorithms because:

1. CMOGP is a non-parametric model which is more flexible than parametric models.

2. Probabilistic CMOGP allows the predictive uncertainty of the outputs to be formally quantified and will be consequently exploited for designing our data-efficient ML algorithms.

3. CMOGP can capture non-trivial/blurred cross-correlations (i.e., non-IMC) between different output types and has been shown to empirically outperform the

other MOGP models [Álvarez and Lawrence, 2011].

4. The unifying sparse structure (USS) of CMOGP can be exploited for deriving our multi-output AL criterion and in turn an efficient approximation algorithm, as detailed in Section 4.2.1 and Section 4.3.

5. In the proposed BO algorithm with multiple output types, the convolutional structure of CMOGP can be exploited to (a) formally characterize the fidelity of each auxiliary functions through its cross-correlation with the target function and (b) derive efficient approximation algorithm of our proposed multi-fidelity acquisition function (Section 5.3).

3.3 Additive GP for High Input Dimensions

Additive GP is a method that can be used to model an unknown function with high input dimensions by decomposing the function into a sum of low-dimensional local functions, each depending on only a subset of the input variables [Duvenaud *et al.*, 2011; Kandasamy *et al.*, 2015]. Specifically, let $D \subset \mathbb{R}^d$ be a set representing the input domain such that each input $x \in D$ is associated with a noisy output $y_x \sim \mathcal{N}(f(x), \sigma_n^2)$, $x^{(i)}$ for $i = 1, ..., C$ be d_i-dimensional *disjoint* input components of x and $d_i \ll d$. Then, $x = \oplus_{i=1}^{C} x^{(i)}$ for each $x \in D$. An additive GP assumes that the function $f(x)$ defined over the input domain D can be decomposed into a sum of local functions:

$$f(x) = f^{(1)}(x^{(1)}) + f^{(2)}(x^{(2)}) + ... + f^{(C)}(x^{(C)}) \tag{3.7}$$

where $f^{(i)}$ are independent for $i = 1, ..., C$. Let $D^{(i)} \subset \mathbb{R}^{d_i}$ denote the i^{th} component of D for $i = 1, ..., C$. If we assume that each $\{f^{(i)}(x^{(i)})\}_{x^{(i)} \in D^{(i)}}$ is a GP with prior

mean $\mu^{(i)}(x^{(i)}) \triangleq \mathbb{E}[f^{(i)}(x^{(i)})]$ and covariance $\sigma^{(i)}(x_p^{(i)}, x_q^{(i)}) \triangleq \mathrm{cov}[f^{(i)}(x_p^{(i)}), f^{(i)}(x_q^{(i)})]$ as shown in Section 3.1 for $i = 1, ..., C$. Then, $\{f(x)\}_{x \in D}$ is also a GP specified by the following additive prior mean and covariance

$$\mu(x) \triangleq \sum_{i=1}^{C} \mu^{(i)}(x^{(i)}), \qquad \sigma(x_p, x_q) \triangleq \sum_{i=1}^{C} \sigma^{(i)}(x_p^{(i)}, x_q^{(i)}).$$

Note that even though the additive assumption in (3.7) may not be true for some unknown function $f(x)$, Kandasamy *et al.* (2015) have argued that the additive model still has advantages compared to the original GP in such cases: additive GP is a simpler model compared to the original GP such that it is easier to fit a *small* set of observations as in many BO applications. They also empirically showed that the additive GP works well when the additive structure used in (3.7) is different from the true additive/non-additive structure of $f(x)$.

Supposing a column vector $y_X \triangleq (y_x)^\top_{x \in X}$ of noisy outputs are observed by evaluating the function $f(x)$ at a set $X \subset D$ of inputs, an additive GP model can provide a predictive distribution $\mathcal{N}(\mu^{(i)}_{Z^{(i)}|X}, \Sigma^{(i)}_{Z^{(i)}Z^{(i)}|X})$ of the local outputs of $f_{Z^{(i)}} \triangleq (f^{(i)}(x^{(i)}))^\top_{x^{(i)} \in Z^{(i)}}$ for any set $Z^{(i)} \subset D^{(i)}$ of local inputs for $i = 1, ..., C$ with the following *posterior* mean vector and covariance matrix:

$$\begin{aligned} \mu^{(i)}_{Z^{(i)}|X} &= \mu^{(i)}_{Z^{(i)}} + \Sigma^{(i)}_{Z^{(i)}X^{(i)}}(\Sigma_{XX} + \sigma_n^2 I)^{-1}(y_X - \mu_X) \\ \Sigma^{(i)}_{Z^{(i)}Z^{(i)}|X} &= \Sigma^{(i)}_{Z^{(i)}Z^{(i)}} - \Sigma^{(i)}_{Z^{(i)}X^{(i)}}(\Sigma_{XX} + \sigma_n^2 I)^{-1}\Sigma^{(i)}_{X^{(i)}Z^{(i)}} \end{aligned} \tag{3.8}$$

where $\mu^{(i)}_{A^{(i)}} \triangleq (\mu^{(i)}(x^{(i)}))^\top_{x^{(i)} \in A^{(i)}}$, $\Sigma^{(i)}_{A^{(i)}B^{(i)}} \triangleq (\sigma^{(i)}(x_p^{(i)}, x_q^{(i)}))_{x_p^{(i)} \in A^{(i)},\ x_q^{(i)} \in B^{(i)}}$ for any $A^{(i)}, B^{(i)} \subseteq D^{(i)}$ and $\mu_A \triangleq (\mu(x))^\top_{x \in A}$, $\Sigma_{AB} \triangleq (\sigma(x_p, x_q))_{x_p \in A, x_q \in B}$ for any $A, B \subseteq D$.

Chapter 4

Near-Optimal Active Learning of MOGPs

In this chapter, we will focus on the first data-efficient ML problem proposed in Section 1.2: active learning for environmental sensing where the target phenomena have been shown to coexist and correlate well with some auxiliary type(s) of phenomena in many cases (Section 1.1). Specifically, we aim to design and develop an active learning algorithm that selects not just the most informative sampling locations to be observed but also the types of measurements (i.e., target and/or auxiliary) at each selected location for minimizing the predictive uncertainty of unobserved areas of a target phenomenon given a sampling budget[1].

To achieve this, we model all types of coexisting phenomena jointly as a CMOGP (Section 4.1) and develop a novel efficient algorithm for active learning of this model in this chapter. In Section 4.2, we first consider utilizing the entropy criterion to measure the predictive uncertainty of a target phenomenon. However, due to its poor

[1]In this chapter, we use the vocabulary of environmental sensing (i.e., "location" as the "input" and "phenomenon" as the "output"). However, all of our results hold for any real-world application domains that have multiple types of correlated outputs and the same active learning objective (i.e., minimize the predictive uncertainty of *only* the target output type(s)).

scalability for solving active learning of CMOGP model, we exploit the structure of sparse CMOGP (Section 3.2.2) for deriving a novel active learning criterion. Then, in order to solve our problem using the novel criterion in a polynomial-time, we relax the notion of submodularity with a ϵ term and exploit the ϵ-submodularity property of our new criterion for devising an approximation algorithm with performance guarantee (Section 4.3). Three real-world datasets are used to empirically evaluate the performance of our proposed algorithm in Section 4.4.

4.1 Modeling Coexisting Phenomena with CMOGP

Recall from Section 1.1 that, in practice, the target phenomenon often coexists and correlates well with some auxiliary type(s) of phenomena whose measurements may be more spatially correlated, less noisy (e.g., due to higher-quality sensors), and/or less tedious to sample (e.g., due to greater availability/quantity, higher sampling rate, and/or lower sampling cost of deployed sensors of these type(s)) and can consequently be exploited for improving its prediction. To capture the cross-correlations between the target and auxiliary phenomena, the CMOGP method can be used to jointly model all types of coexisting phenomena. Specifically, let the sampling locations be the inputs and the measurement of type i be the i^{th} type of output of a CMOGP model. The random output measurements $\{Y_{\langle x,i\rangle}\}_{\langle x,i\rangle\in D^+}$ will be a GP using the results of Section 3.2.1. Then, the covariance in (3.2) can be used to characterize the spatial correlation structure for each type of phenomenon (i.e., $i = j$) and the cross-correlation structure between different types of phenomena (i.e., $i \neq j$), and the predictive distribution of the measurements for any set Z of unobserved locations and their corresponding measurement types can be computed using (3.3) and (3.4) for CMOGP and sparse CMOGP, respectively.

4.2 Active Learning of CMOGP

As has been shown in many existing literature [Bueso *et al.*, 1999; Krause and Guestrin, 2007; Chen *et al.*, 2012b], the entropy criterion can be used to measure the predictive uncertainty of the unobserved areas of a target phenomenon. Using the CMOGP model, the Gaussian posterior joint entropy (i.e., predictive uncertainty) of the measurements Y_Z for any set $Z \subseteq D^+ \setminus X$ of tuples of unobserved locations and their corresponding measurement types can be expressed in terms of its posterior covariance matrix $\Sigma_{ZZ|X}$ (3.3) which is independent of the realized measurements y_X:

$$H(Y_Z|Y_X) \triangleq \frac{1}{2}\log(2\pi e)^{|Z|}|\Sigma_{ZZ|X}| \ . \tag{4.1}$$

Let index t denote the type of measurements of the target phenomenon[2]. Then, active learning of a CMOGP model involves selecting an optimal set $X^* \triangleq \bigcup_{i=1}^{M} X_i^*$ of N tuples (i.e., sampling budget) of sampling locations and their corresponding measurement types to be observed that minimize the posterior joint entropy of type t measurements at the remaining unobserved locations of the target phenomenon:

$$X^* \triangleq \arg\min_{X:|X|=N} H(Y_{V_t \setminus X_t}|Y_X) \tag{4.2}$$

where $V_t \subset D_t^+$ is a finite set of tuples of candidate sampling locations of the target phenomenon and their corresponding measurement type t available to be selected for observation. However, evaluating the $H(Y_{V_t \setminus X_t}|Y_X)$ term in (4.2) incurs $\mathcal{O}(|V_t|^3 + N^3)$ time, which is prohibitively expensive when the target phenomenon is spanned by a large number $|V_t|$ of candidate sampling locations. If auxiliary types of phenomena are missing or ignored (i.e., $M = 1$), then such a computational difficulty can be

[2]Our proposed algorithm can be extended to handle multiple types of target phenomena, as demonstrated in Section 4.4.

eased by instead solving the well-known *maximum entropy sampling* (MES) problem [Shewry and Wynn, 1987]:

$$X_t^* = \underset{X_t:|X_t|=N}{\arg\max}\, H(Y_{X_t})$$

which can be proven to be equivalent to (4.2) by using the chain rule for entropy $H(Y_{V_t}) = H(Y_{X_t}) + H(Y_{V_t \setminus X_t}|Y_{X_t})$ and noting that $H(Y_{V_t})$ is a constant. Evaluating the $H(Y_{X_t})$ term in MES incurs $\mathcal{O}(|X_t|^3)$ time, which is independent of $|V_t|$. Such an equivalence result can in fact be extended and applied to minimizing the predictive uncertainty of *all* M types of coexisting phenomena, as exploited by multivariate spatial sampling algorithms [Bueso *et al.*, 1999; Le *et al.*, 2003]:

$$\underset{X:|X|=N}{\arg\max}\, H(Y_X) = \underset{X:|X|=N}{\arg\min}\, H(Y_{V \setminus X}|Y_X), \tag{4.3}$$

where $V \triangleq \bigcup_{i=1}^{M} V_i$ and V_i is defined in a similar manner to V_t but for measurement type $i \neq t$. This equivalence result (4.3) also follows from the chain rule for entropy $H(Y_V) = H(Y_X) + H(Y_{V \setminus X}|Y_X)$ and the fact that $H(Y_V)$ is a constant. Unfortunately, it is not straightforward to derive such an equivalence result for our active learning problem (4.2) in which a target phenomenon of interest coexists with auxiliary types of phenomena (i.e., $M > 1$): If we consider maximizing $H(Y_X)$ or $H(Y_{X_t})$, then it is no longer equivalent to minimizing $H(Y_{V_t \setminus X_t}|Y_X)$ (4.2) as the sum of the two entropy terms is not necessarily a constant.

4.2.1 Exploiting sparse CMOGP model structure

We derive a new equivalence result by considering instead a constant entropy $H(Y_{V_t}|L_U)$ that is conditioned on the inducing measurements L_U used in sparse CMOGP regression models (Section 3.2.2). Then, by using the chain rule for entropy and structural property **P2** shared by sparse CMOGP regression models in the unifying framework

[Álvarez and Lawrence, 2011] described in Section 3.2.2, (4.2) can be proven (see Appendix A.1) to be equivalent to

$$X^* \triangleq \underset{X:|X|=N}{\arg\max}\, H(Y_{X_t}|L_U) - I(L_U; Y_{V_t\setminus X_t}|Y_X) \tag{4.4}$$

where

$$I(L_U; Y_{V_t\setminus X_t}|Y_X) \triangleq H(L_U|Y_X) - H(L_U|Y_{X\cup V_t\setminus X_t}) \tag{4.5}$$

is the conditional mutual information between L_U and $Y_{V_t\setminus X_t}$ given Y_X. Our novel active learning criterion in (4.4) exhibits an interesting exploration-exploitation trade-off: The inducing measurements L_U can be viewed as latent structure of the sparse CMOGP model to induce conditional independence properties **P1** and **P2**. So, on one hand, maximizing the $H(Y_{X_t}|L_U)$ term aims to select tuples X_t of locations with the most uncertain measurements Y_{X_t} of the target phenomenon and their corresponding type t to be observed given the latent model structure L_U (i.e., exploitation). On the other hand, minimizing the $I(L_U; Y_{V_t\setminus X_t}|Y_X)$ term (4.5) aims to select tuples X to be observed (i.e., possibly of measurement types $i \neq t$) so as to rely less on measurements $Y_{V_t\setminus X_t}$ of type t at the remaining unobserved locations of the target phenomenon to infer latent model structure L_U (i.e., exploration) since $Y_{V_t\setminus X_t}$ won't be sampled.

Supposing $|U| \leq |V_t|$, evaluating our new active learning criterion in (4.4) incurs $\mathcal{O}(|U|^3 + N^3)$ time for every $X \subset V$ and a *one-off* cost of $\mathcal{O}(|V_t|^3)$ time (Appendix A.2). In contrast, computing the original criterion in (4.2) requires $\mathcal{O}(|V_t|^3 + N^3)$ time for every $X \subset V$, which is more costly, especially when the number N of selected observations is much less than the number $|V_t|$ of candidate sampling locations of the target phenomenon due to, for example, a tight sampling budget or a large sampling domain that usually occurs in practice. The trick to achieving such a computational advantage can be inherited by our approximation algorithm to be

described next.

4.3 Approximation Algorithm

Our novel active learning criterion in (4.4), when optimized, still suffers from poor scalability in the number N of selected observations (i.e., sampling budget) like the old criterion in (4.2) because it involves evaluating a prohibitively large number of candidate selections of sampling locations and their corresponding measurement types (i.e., exponential in N). However, unlike the old criterion, it is possible to devise an efficient approximation algorithm with a theoretical performance guarantee to optimize our new criterion, which is the main contribution of our work in this chapter.

The key idea of our proposed approximation algorithm is to greedily select the next tuple of sampling location and its corresponding measurement type to be observed that maximally increases our criterion in (4.4), and iterate this till N tuples are selected for observation. Specifically, let

$$F(X) \triangleq H(Y_{X_t}|L_U) - I(L_U; Y_{V_t \setminus X_t}|Y_X) + I(L_U; Y_{V_t}) \tag{4.6}$$

denote our active learning criterion in (4.4) augmented by a positive constant $I(L_U; Y_{V_t})$ to make $F(X)$ non-negative. Such an additive constant $I(L_U; Y_{V_t})$ is simply a technical necessity for proving the performance guarantee and does not affect the outcome of the optimal selection (i.e., $X^* = \arg\max_{X:|X|=N} F(X)$). Then, our approximation algorithm greedily selects the next tuple $\langle x, i\rangle$ of sampling location x and its corresponding measurement type i that maximizes $F(X \cup \{\langle x, i\rangle\}) - F(X)$:

$$\begin{aligned} \langle x, i\rangle^+ &\triangleq \arg\max_{\langle x,i\rangle \in V \setminus X} F(X \cup \{\langle x, i\rangle\}) - F(X) \\ &= \arg\max_{\langle x,i\rangle \in V \setminus X} H(Y_{\langle x,i\rangle}|Y_X) - \delta_i H(Y_{\langle x,i\rangle}|Y_{X \cup V_t \setminus X_t}) \end{aligned} \tag{4.7}$$

Algorithm 1 Greedy Active Learning Algorithm of MOGP

Input: A set V of candidate sampling locations;

1: $X \leftarrow \emptyset$;
2: **for** $n = 1$ to N **do**
3: select $\langle x, i\rangle^+ = \arg\max_{\langle x,i\rangle \in V \setminus X} H(Y_{\langle x,i\rangle}|Y_X) - \delta_i H(Y_{\langle x,i\rangle}|Y_{X \cup V_t \setminus X_t})$;
4: $X \leftarrow X \cup \{\langle x, i\rangle^+\}$;
5: **end for**
6: **return** X

where δ_i is a Kronecker delta of value 0 if $i = t$, and 1 otherwise. The derivation of (4.7) is in Appendix A.3. Our algorithm updates $X \leftarrow X \cup \{\langle x, i\rangle^+\}$ and iterates the greedy selection (4.7) and update till $|X| = N$ (i.e., sampling budget is depleted), which is shown in Algorithm 1. The intuition to understanding (4.7) is that our algorithm has to choose between observing a sampling location with the most uncertain measurement (i.e., $H(Y_{\langle x,t\rangle}|Y_X)$) of the target phenomenon (i.e., type t) vs. that for an auxiliary type $i \neq t$ inducing the largest reduction in predictive uncertainty of the measurements at the remaining unobserved locations of the target phenomenon since $H(Y_{\langle x,i\rangle}|Y_X) - H(Y_{\langle x,i\rangle}|Y_{X \cup V_t \setminus X_t}) = I(Y_{\langle x,i\rangle}; Y_{V_t \setminus X_t}|Y_X) = H(Y_{V_t \setminus X_t}|Y_X) - H(Y_{V_t \setminus X_t}|Y_{X \cup \{\langle x,i\rangle\}})$.

It is also interesting to figure out whether our approximation algorithm may avoid selecting tuples of a certain auxiliary type $i \neq t$ and formally analyze the conditions under which it will do so, as elucidated in the following result:

Proposition 1. *Let* $V_{\text{-}t} \triangleq \bigcup_{i \neq t} V_i$, $X_{\text{-}t} \triangleq \bigcup_{i \neq t} X_i$, $\rho_i \triangleq \sigma_{s_i}^2/\sigma_{n_i}^2$, *and* $R(\langle x, i\rangle, V_t \setminus X_t) \triangleq \sum_{\langle x',t\rangle \in V_t \setminus X_t} \mathcal{N}(x - x'|\underline{0}, P_0^{-1} + P_i^{-1} + P_t^{-1})^2$. *Assuming absence of suppressor variables,* $H(Y_{\langle x,i\rangle}|Y_X) - H(Y_{\langle x,i\rangle}|Y_{X \cup V_t \setminus X_t}) \leq 0.5 \log(1 + 4\rho_t \rho_i R(\langle x, i\rangle, V_t \setminus X_t))$ *for any* $\langle x, i\rangle \in V_{\text{-}t} \setminus X_{\text{-}t}$.

Its proof (Appendix A.4) relies on the following assumption of the absence of suppressor variables which holds in many practical cases [Das and Kempe, 2008]:

Conditioning does not make $Y_{\langle x,i\rangle}$ and $Y_{\langle x',t\rangle}$ more correlated for any $\langle x,i\rangle \in V_{\text{-}t} \setminus X_{\text{-}t}$ and $\langle x',t\rangle \in V_t \setminus X_t$. Proposition 1 reveals that when the signal-to-noise ratio ρ_i of auxiliary type i is low (e.g., poor-quality measurements due to high noise) and/or the cross-correlation (3.2) between measurements of the target phenomenon and auxiliary type i is small due to low $\sigma^2_{s_t}\sigma^2_{s_i}R(\langle x,i\rangle, V_t \setminus X_t)$, our greedy criterion in (4.7) returns a small value, hence causing our algorithm to avoid selecting tuples of auxiliary type i.

Theorem 1 (Time Complexity). *Our approximation algorithm incurs* $\mathcal{O}(N(|V||U|^2 + N^3) + |V_t|^3)$ *time.*

Its proof is in Appendix A.5. So, our approximation algorithm only incurs quartic time in the number N of selected observations and cubic time in the number $|V_t|$ of candidate sampling locations of the target phenomenon.

4.3.1 Performance guarantee

To theoretically guarantee the performance of our approximation algorithm, we will first motivate the need to define a relaxed notion of *submodularity*. A submodular set function exhibits a natural diminishing returns property: When adding an element to its input set, the increment in its function value decreases with a larger input set. To maximize a nondecreasing and submodular set function, the work of Nemhauser *et al.* (1978) has proposed a greedy algorithm guaranteeing a $(1-1/e)$-factor approximation of that achieved by the optimal input set.

The main difficulty in proving the submodularity of $F(X)$ (4.6) lies in its mutual information term being conditioned on X. Some works [Krause and Guestrin, 2005; Renner and Maurer, 2002] have shown the submodularity of such conditional mutual information by imposing conditional independence assumptions (e.g., Markov chain). In practice, these strong assumptions (e.g., $Y_A \perp Y_{\langle x,i\rangle}|Y_{V_t\setminus X_t}$ for any $A \subseteq X$

and $\langle x, i\rangle \in V_{\text{-}t} \setminus X_{\text{-}t}$) severely violate the correlation structure of multiple types of coexisting phenomena and are an overkill: The correlation structure can in fact be preserved to a fair extent by relaxing these assumptions, which consequently entails a relaxed form of submodularity of $F(X)$ (4.6); a performance guarantee similar to that of Nemhauser *et al.* (1978) can then be derived for our approximation algorithm.

Definition 1. *A function $G : 2^B \rightarrow \mathbb{R}$ is submodular if*

$$G(A' \cup \{a\}) - G(A') \leq G(A \cup \{a\}) - G(A)$$

for any $A \subseteq A' \subseteq B$ and $a \in B \setminus A'$.

Definition 2. *A function $G : 2^B \rightarrow \mathbb{R}$ is ϵ-submodular if*

$$G(A' \cup \{a\}) - G(A') \leq G(A \cup \{a\}) - G(A) + \epsilon$$

for any $A \subseteq A' \subseteq B$ and $a \in B \setminus A'$.

Lemma 1. *Let $\sigma^2_{n^*} \triangleq \min_{i\in\{1,\ldots,M\}} \sigma^2_{n_i}$. Given $\epsilon_1 \geq 0$, if*

$$\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x,i\rangle|\widetilde{X}\cup V_t\setminus X_t} - \Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x,i\rangle|X\cup V_t\setminus X_t} \leq \epsilon_1 \tag{4.8}$$

for any $\widetilde{X} \subseteq X$ and $\langle x, i\rangle \in V_{\text{-}t} \setminus X_{\text{-}t}$, then $F(X)$ is ϵ-submodular where $\epsilon = 0.5\log(1+\epsilon_1/\sigma^2_{n^*})$.

Its proof is in Appendix A.6. Note that (4.8) relaxes the above example of conditional independence assumption (i.e., assuming $\epsilon_1 = 0$) to one which allows $\epsilon_1 > 0$. In practice, ϵ_1 is expected to be small: Since further conditioning monotonically decreases a posterior variance [Xu *et al.*, 2014], an expected large set $V_t \setminus X_t$ of tuples of remaining unobserved locations of the target phenomenon tends to be informative enough to make $\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x,i\rangle|\widetilde{X}\cup V_t\setminus X_t}$ small and hence the non-negative variance reduction term and ϵ_1 in (4.8) small.

Furthermore, (4.8) with a given small ϵ_1 can be realized by controlling the discretization of the domain of candidate sampling locations. For example, by refining

the discretization of V_t (i.e., increasing $|V_t|$), the variance reduction term in (4.8) decreases because it has been shown in [Das and Kempe, 2008] to be submodular in many practical cases. We give another example in Lemma 2 to realize (4.8) by controlling the discretization such that every pair of selected observations are sufficiently far apart.

It is easy to derive that $F(\emptyset) = 0$. The "information never hurts" bound for entropy [Cover and Thomas, 1991] entails a nondecreasing $F(X)$:

$$\begin{aligned} F(X \cup \{\langle x,i\rangle\}) - F(X) &= H(Y_{\langle x,i\rangle}|Y_X) - \delta_i H(Y_{\langle x,i\rangle}|Y_{X\cup V_t\setminus X_t}) \\ &\geq H(Y_{\langle x,i\rangle}|Y_X) - H(Y_{\langle x,i\rangle}|Y_{X\cup V_t\setminus X_t}) \geq 0 \ . \end{aligned}$$

The first inequality requires $\sigma^2_{n^*} \geq (2\pi e)^{-1}$ so that $H(Y_{\langle x,i\rangle}|Y_A) = 0.5\log 2\pi e \Sigma^{\mathrm{PITC}}_{\langle x,i\rangle\langle x,i\rangle|A} \geq 0.5\log 2\pi e\sigma^2_{n^*} \geq 0$,[3] which is reasonable in practice due to ubiquitous noise. Combining this result with Lemma 1 yields the performance guarantee:

Theorem 2. *Given $\epsilon_1 \geq 0$, if (4.8) holds, then our approximation algorithm is guaranteed to select X s.t. $F(X) \geq (1-1/e)(F(X^*) - N\epsilon)$ where $\epsilon = 0.5\log(1 + \epsilon_1/\sigma^2_{n^*})$.*

Its proof (Appendix A.7) is similar to that of the well-known result of Nemhauser *et al.* (1978) except for exploiting ϵ-submodularity of $F(X)$ in Lemma 1 instead of submodularity.

Finally, we present a discretization scheme that satisfies (4.8): Let ω be the smallest discretization width of V_i for $i = 1,\ldots,M$. Construct a new set $V^- \subset V$ of tuples of candidate sampling locations and their corresponding measurement types such that every pair of tuples are at least a distance of $p\omega$ apart for some $p > 0$; each candidate location thus has only one corresponding type. Such a construction V^- constrains our algorithm to select observations sparsely across the spatial domain so that any

[3] $\Sigma^{\mathrm{PITC}}_{\langle x,i\rangle\langle x,i\rangle|A} \geq \sigma^2_{n^*}$ is proven in Lemma 3 in Appendix A.4.

$\langle x, i\rangle \in V_{\text{-}t} \setminus X_{\text{-}t}$ has sufficiently many neighboring tuples of remaining unobserved locations of the target phenomenon from $V_t \setminus X_t$ to keep $\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x,i\rangle|\widetilde{X}\cup V_t\setminus X_t}$ small and hence the variance reduction term and ϵ_1 in (4.8) small. Our previous theoretical results still hold if V^- is used instead of V. The result below gives the minimum value of p to satisfy (4.8):

Lemma 2. *Let* $\sigma^2_{s^*} \triangleq \max_{i\in\{1,\ldots,M\}} \sigma^2_{s_i}$, ℓ *be the largest first diagonal component of* $P_0^{-1} + P_i^{-1} + P_j^{-1}$ *for all* $i, j = 1, \ldots, M$, *and* $\xi \triangleq \exp(-\omega^2/(2\ell))$. *Given* $\epsilon_1 > 0$ *and assuming absence of suppressor variables, if*

$$p^2 > \log\left\{\frac{1}{2\sigma^2_{s^*}}\min\left(\frac{\sigma^2_{n^*}}{N}, \frac{1}{2}\left(\sqrt{\epsilon_1^2 + \frac{4\epsilon_1\sigma^2_{n^*}}{N}} - \epsilon_1\right)\right)\right\} \Big/ \log\xi \ ,$$

then (4.8) *holds. See Appendix A.8 for its proof.*

4.4 Experimental Results

This section evaluates the predictive performance of our approximation algorithm (m-Greedy) empirically on three real-world datasets:

(a) *Jura* dataset [Goovaerts, 1997] contains concentrations of 7 heavy metals collected at 359 locations in a Swiss Jura region;

(b) *Gilgai* dataset [Webster, 1977] contains electrical conductivity and chloride content generated from a line transect survey of 365 locations of Gilgai territory in New South Wales, Australia;

(c) *Indoor environmental quality* (IEQ) dataset [Bodik *et al.*, 2004] contains temperature (°F) and light (Lux) readings taken by 43 temperature sensors and 41 light sensors deployed in the Intel Berkeley Research lab.

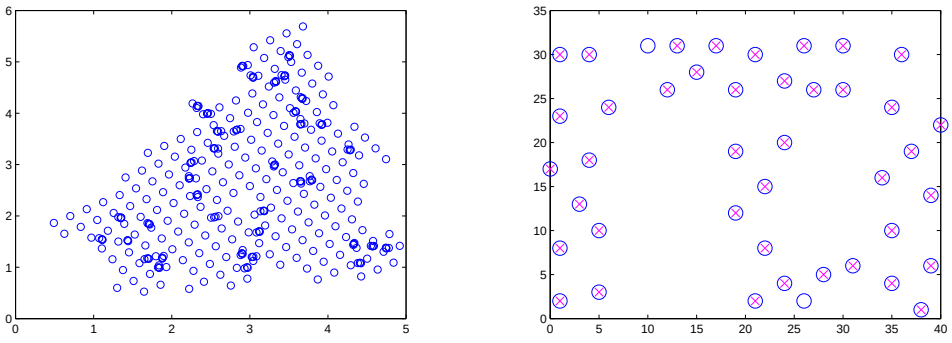


Figure 4.1: Sampling locations for the (a) Jura (km) and (b) IEQ (m) datasets where '$\circ$' and '$\times$' denote locations of temperature and light sensors, respectively.

The sampling locations for the Jura and IEQ datasets are shown in Figs 4.1.

The performance of m-Greedy is compared to that of the (a) maximum variance/entropy (m-Var) algorithm which greedily selects the next location and its corresponding measurement type with maximum posterior variance/entropy in each iteration; and (b) greedy maximum entropy (s-Var) [Shewry and Wynn, 1987] and mutual information (s-MI) [Krause *et al.*, 2008] sampling algorithms for gathering observations *only* from the target phenomenon.

For all experiments, k-means is used to select inducing locations U by clustering all possible locations available to be selected for observation into $|U|$ clusters such that each cluster center corresponds to an element of U. The hyper-parameters (i.e., $\sigma^2_{s_i}$, $\sigma^2_{n_i}$, P_0 and P_i for $i = 1, \ldots, M$) of MOGP and single-output GP models are learned using the data via maximum likelihood estimation [Álvarez and Lawrence, 2011]. For each dataset, observations (i.e., 100 for Jura and Gilgai datasets and 10 for IEQ dataset) of type t are randomly selected to form the test set T; the tuples of candidate sampling locations and corresponding type t therefore become less than that of auxiliary types. The *root mean squared error* (RMSE) metric $\sqrt{|T|^{-1} \sum_{x \in T} (y_{\langle x,t \rangle} - \mu_{\langle x,t \rangle | X})^2}$ is used to evaluate the performance of the tested al-

gorithms. All experimental results are averaged over 50 random test sets. For a fair comparison, the measurements of all types are normalized before using them for training, prediction, and active learning.

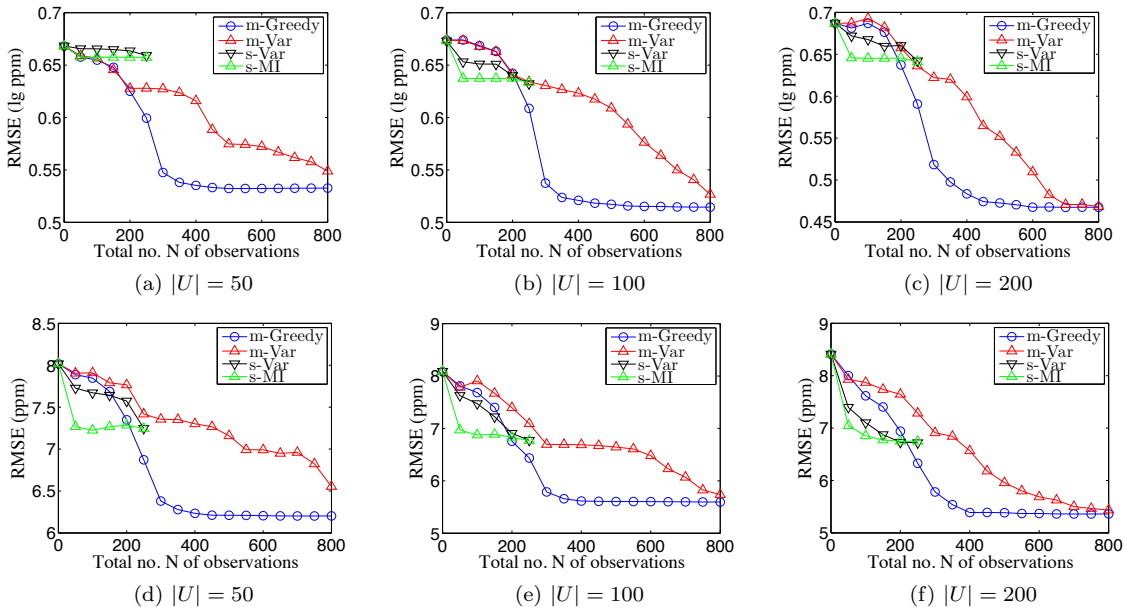
4.4.1 Jura dataset.

Three types of correlated lg-Cd, Ni, and lg-Zn measurements are used in this experiment; we take the log of Cd and Zn measurements to remove their strong skewness, as proposed as a standard statistical practice in [Webster and Oliver, 2007]. The measurement types with the smallest and largest signal-to-noise ratios (respectively, lg-Cd and Ni; see Table 4.1) are each set as type t.

	lg-Cd	Ni	lg-Zn
$\sigma^2_{s_i}$	2.2204	8.8280	2.3198
$\sigma^2_{n_i}$	0.0853	0.1130	0.0596
ρ_i	**26.0305**	**78.1239**	38.9228

Table 4.1: Signal-to-noise ratios ρ_i of lg-Cd, Ni, and lg-Zn measurements for Jura dataset with $|U| = 100$.

Figs. 4.2a-c and 4.2d-f show, respectively, results of the tested algorithms with lg-Cd and Ni as type t. It can be observed that the RMSE of m-Greedy decreases more rapidly than that of m-Var, especially when observations of auxiliary types are selected after about $N = 200$. This is because our algorithm selects observations of auxiliary types that induce the largest reduction in predictive uncertainty of the measurements at the remaining unobserved locations of the target phenomenon (Section 4.3). In contrast, m-Var may select observations that reduce the predictive uncertainty of auxiliary types of phenomena, which does not directly achieve the aim of our active learning problem. With increasing $|U|$, both m-Greedy and m-Var reach



(a) $|U| = 50$ (b) $|U| = 100$ (c) $|U| = 200$

(d) $|U| = 50$ (e) $|U| = 100$ (f) $|U| = 200$

Figure 4.2: Graphs of RMSEs vs. no. N of observations with (a-c) lg-Cd and (d-f) Ni as type t and varying no. $|U| = 50, 100, 200$ of inducing locations for Jura dataset.

smaller RMSEs, but m-Greedy can achieve this faster with much less observations. As shown in Figs. 4.2a-f, m-Greedy performs much better than s-Var and s-MI, which means observations of correlated auxiliary types can indeed be used to improve the prediction of the target phenomenon. Note that there are only limited number of observations for s-Var and s-MI since the candidate observations of the target output type are limited in the dataset. Finally, by comparing the results between Figs. 4.2a-c and 4.2d-f, the RMSE of m-Greedy with Ni as type t decreases faster than that with lg-Cd as type t, especially in the beginning (i.e., $N \leq 200$) due to higher-quality Ni measurements (i.e., larger signal-to-noise ratio).

4.4.2 Gilgai dataset.

In this experiment, the lg-Cl contents at depth 0-10cm and 30-40cm are used jointly as two types of target phenomena while the log of electrical conductivity, which is easier

to measure at these depths, is used as the auxiliary type. Fig. 4.3a shows results of the average RMSE over the two lg-Cl types with $|U| = 100$. Similar to the results of the Jura dataset, with two types of target phenomena, the RMSE of m-Greedy still decreases more rapidly with increasing N than that of m-Var and achieves a much smaller RMSE than that of s-Var and s-MI; the results of s-Var and s-MI are also averaged over two independent single-output GP predictions of lg-Cl content at the two depths.

4.4.3 IEQ dataset.

Fig. 4.3b shows results with light as type t and $|U| = 40$. The observations are similar to that of the Jura and Gilgai datasets: RMSE of m-Greedy decreases faster than that of the other algorithms. More importantly, with the same number of observations, m-Greedy achieves much smaller RMSE than s-Var and s-MI that can sample only from the target phenomenon. This is because m-Greedy selects observations of the auxiliary type (i.e., temperature) that are less noisy ($\sigma^2_{n_i} = 0.13$) than that of light ($\sigma^2_{n_t} = 0.23$), which demonstrates its advantage over s-Var and s-MI when type t measurements are noisy (e.g., due to poor-quality sensors).

4.5 Summary

This work describes a novel efficient algorithm for active learning of a MOGP model. To resolve the issue of poor scalability in optimizing the conventional entropy criterion, we exploit a structure common to a unifying framework of sparse MOGP models for deriving a novel active learning criterion (4.4). Then, we exploit the ϵ-submodularity property of our new criterion (Lemma 1) for devising a polynomial-time approximation algorithm (4.7) that guarantees a constant-factor approximation of that achieved by the optimal set of selected observations (Theorem 2). Empiri-

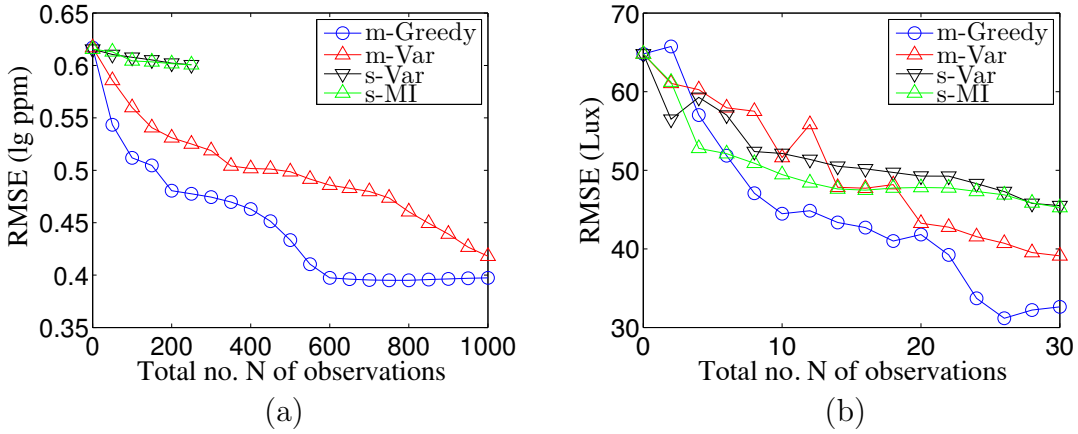


Figure 4.3: Graphs of RMSEs vs. no. N of observations with (a) lg-Cl as types t for Gilgai dataset and (b) light as type t for IEQ dataset.

cal evaluation on three real-world datasets shows that our approximation algorithm m-Greedy outperforms existing algorithms for active learning of MOGP and single-output GP models, especially when measurements of the target phenomenon are noisier than that of the auxiliary types.

Chapter 5

Predictive Entropy Search (PES) for Multi-Fidelity BO

Chapter 4 has proposed an active learning algorithm for multiple correlated output types. Motivated by the *automatic ML* examples in Section 1.1, this chapter aims to generalize another data-efficient ML method (i.e., BO) to multiple types of correlated outputs (i.e., multiple correlated functions). In particular, we aim to design and develop a *multi-fidelity* BO algorithm that selects not just the most informative inputs but also the target and/or auxiliary function(s) with varying fidelities and costs to be evaluated at each selected input for finding or improving the belief of the global target maximizer.

To achieve this, we first model the unknown target and auxiliary functions jointly as a CMOGP (Section 5.1). Then, a novel generalization of PES for multi-fidelity BO called *multi-fidelity PES* (MF-PES) is proposed in Section 5.3. In contrast to the state-of-the-art multi-fidelity BO algorithms reviewed in Section 2.2.2, our proposed MF-PES algorithm can naturally trade off between exploration vs. exploitation over the target and auxiliary functions with varying fidelities without needing to manually

tune any such parameters (Section 5.2). More importantly, to compute the acquisition function of MF-PES in closed form, an efficient approximation of MF-PES is derived in Section 5.3 via (a) a novel *multi-output random features* (MRF) approximation of the CMOGP model whose cross-correlation (i.e., multi-fidelity) structure between the target and auxiliary functions can be exploited for improving the belief of the target maximizer using the observations from evaluating these functions (Section 5.3.1), and (b) practical constraints relating the global target maximizer to that of the auxiliary functions (Section 5.3.2). Several benchmark functions with varying optimization difficulties and two real-world hyperparameters tuning problems are used to empirically demonstrate the advantages of our MF-PES algorithm (Section 5.4).

5.1 Multi-Fidelity Modeling with CMOGP

In order to exploit the cross-correlations between multiple functions, we first model the unknown target and auxiliary functions jointly using a CMOGP (Section 3.2.1). Specifically, let M unknown functions $f_1, \ldots, f_M$ with varying fidelities be jointly modeled as a CMOGP over a bounded input domain $D \subset \mathbb{R}^d$ such that each input $x \in D$ is associated with a noisy output

$$y_{\langle x,i \rangle} \sim \mathcal{N}(f_i(x), \sigma_{n_i}^2) \tag{5.1}$$

for $i = 1, \ldots, M$. Recall from Section 3.2.1 that CMOGP defines each i-th function f_i as a convolution between a smoothing kernel K_i and a latent function [1] L:

$$f_i(x) \triangleq \int_{x' \in D} K_i(x - x')\ L(x')\ \mathrm{d}x'\ . \tag{5.2}$$

[1] To ease exposition, we consider a single latent function. Note, however, multiple latent functions can improve multi-fidelity prediction. More importantly, our proposed MF-PES algorithm can be easily generalized to handle multiple latent functions, as shown in Appendix B.5.

If $\{L(x)\}_{x\in D}$ is a GP, then $\{f_i(x)\}_{\langle x,i\rangle\in D^+}$ is also a GP, that is, every finite subset of $\{f_i(x)\}_{\langle x,i\rangle\in D^+}$ follows a multivariate Gaussian distribution. Such a GP is fully specified by its *prior* mean $\mu_{\langle x,i\rangle} \triangleq \mathbb{E}[f_i(x)]$ and covariance $\sigma_{\langle x,i\rangle\langle x',j\rangle} \triangleq \mathrm{cov}[f_i(x), f_j(x')]$ for all $\langle x,i\rangle, \langle x',j\rangle \in D^+$, the latter of which characterizes both the correlation structure within each function (i.e., $i = j$) and the cross-correlation between different functions (i.e., $i \neq j$). By assuming the same kernel functions of $L(x)$ and $K_i(x)$ as that of Section 3.2.1, the covariance

$$\sigma_{\langle x,i\rangle\langle x',j\rangle} = \sigma_{s_i}\sigma_{s_j}\mathcal{N}(x - x'|\underline{0}, P_0^{-1} + P_i^{-1} + P_j^{-1}) \ . \tag{5.3}$$

Notice that (5.3) leaves out the noise variance term in (3.3) since we focus on the noiseless functions f_i instead of the noisy outputs $Y_{\langle x,i\rangle}$ in this chapter due to the BO objective (i.e., finding the global maximizer of the *target function*).

Let t be the index of the target function and x_{*_i} be the maximizer of function f_i. Interestingly, the fidelity of an auxiliary function f_i with respect to target function f_t in the context of BO can naturally be characterized by the following normalized covariance between $f_i(x_{*_i})$ and $f_t(x_{*_t})$:

$$\rho_i \triangleq \sigma_{\langle x_{*_i},i\rangle\langle x_{*_t},t\rangle}/(\sigma'_{s_i}\sigma'_{s_t}) \in [0,1] \tag{5.4}$$

where $\sigma'_{s_i} \triangleq \sigma_{s_i}/(2\pi|P_0^{-1} + 2P_i^{-1}|)^{1/4}$. Note that our defined fidelity measure ρ_i tends to 1 (i.e., higher fidelity of f_i) when (a) the convolutional structure of f_i parametrized by P_i becomes more similar to that of f_t (i.e., P_t) and (b) the maximizer x_{*_i} of f_i is closer to the target maximizer x_{*_t}. We will show in our experiments (Section 5.4) that an auxiliary function f_i with a higher fidelity ρ_i improves the BO performance of our MF-PES algorithm.

Similar to (3.3), a CMOGP model can provide a predictive belief/distribution

$\mathcal{N}(\mu_{Z|X}, \Sigma_{ZZ|X})$ of the outputs of $f_Z \triangleq (f_i(x))^\top_{\langle x,i\rangle \in Z}$ (instead of Y_Z) for any set Z of input tuples with the following *posterior* mean vector and covariance matrix:

$$\begin{aligned} \mu_{Z|X} &\triangleq \mu_Z + \Sigma_{ZX}(\Sigma_{XX} + \Sigma_\varepsilon)^{-1}(y_X - \mu_X) \\ \Sigma_{ZZ|X} &\triangleq \Sigma_{ZZ} - \Sigma_{ZX}(\Sigma_{XX} + \Sigma_\varepsilon)^{-1}\Sigma_{XZ} \end{aligned} \tag{5.5}$$

where $\Sigma_{AA'} \triangleq (\sigma_{\langle x,i\rangle\langle x',j\rangle})_{\langle x,i\rangle \in A, \langle x',j\rangle \in A'}$ for any $A, A' \subseteq D^+$ with $\sigma_{\langle x,i\rangle\langle x',j\rangle}$ defined in (5.3) , and Σ_ε is a diagonal matrix with diagonal components $\sigma^2_{n_i}$ occurring $|X_i|$ times for $i = 1, \ldots, M$.

5.2 Multi-Fidelity BO

A multi-fidelity BO algorithm repeatedly selects the next input tuple $\langle x, i\rangle$ for evaluating the i-th function f_i at x that maximizes a choice of multi-fidelity acquisition function $\alpha(y_X, \langle x, i\rangle)$ given the past observations (X, y_X):

$$\langle x, i\rangle^+ \triangleq \underset{\langle x,i\rangle \in D^+ \setminus X}{\arg\max}\ \alpha(y_X, \langle x, i\rangle)$$

and updates $X \leftarrow X \cup \{\langle x, i\rangle^+\}$ until the budget is expended. The general algorithm of multi-fidelity BO is shown in Algorithm 2. Intuitively, the acquisition function α should be constructed to enable the multi-fidelity BO algorithm to jointly and naturally optimize the non-trivial trade-off between exploitation vs. exploration over the target and auxiliary functions with varying fidelities for finding or improving the belief of the global target maximizer x_{*_t} by utilizing information from the CMOGP predictive belief of these functions (5.5).

To do this, one may, at first glance, be tempted to consider a (a) direct application or a (b) straightforward generalization of UCB and improvement-based (e.g., PI and EI) acquisition functions that enable the conventional BO algorithms to optimize

Algorithm 2 General Multi-Fidelity BO Algorithm

Input: A budget B; Input domain D^+; A set of random initializations $(X_{\text{init}}, y_{X_{\text{init}}})$

1: $X \leftarrow X_{\text{init}}$;
2: **while** $B \geq 0$ **do**
3: select $\langle x, i\rangle^+ \leftarrow \arg\max_{\langle x,i\rangle \in D^+ \setminus X} \alpha(y_X, \langle x, i\rangle)$;
4: evaluate function f_i at the selected input x to get the observation $y_{\langle x,i\rangle^+}$;
5: $X \leftarrow X \cup \{\langle x, i\rangle^+\}$;
6: $B \leftarrow B - \text{cost}_i(x)$;
7: **end while**
8: **return** $\tilde{x}_{t_*} \leftarrow \arg\max_{x \in D} \mu_{\{\langle x,t\rangle\}|X}$

the trade-off between exploitation vs. exploration using the GP predictive/posterior mean and variance, respectively. (a) The former would, however, waste the informative observations from evaluating a high-fidelity auxiliary function (i.e., convolutional structures and maximizers of the target and auxiliary functions are similar or close due to a positive cross-correlation[2]) that is less noisy and/or cheaper to evaluate than the target function. (b) The latter can be achieved by, for example, plugging in the averaged predictive means and variances over all (i.e., target and auxiliary) functions which, unfortunately, satisfies a different objective of maximizing an average of these functions (see Section 3.2 in [Swersky *et al.*, 2013]) instead of the target function directly.

To resolve such issues, a *multi-fidelity GP-UCB* (MF-GP-UCB) algorithm [Kandasamy *et al.*, 2016] has recently been proposed and requires heuristically setting parameters to trade off between exploitation vs. exploration over the target and auxiliary functions with varying fidelities[3] for practical implementation. In particular,

[2]Like the work of [Swersky *et al.*, 2013] (Section 2.2), we assume the cross-correlation between the target and auxiliary functions to be positive. An auxiliary function that is negatively correlated with the target function can be easily transformed to be positively correlated by negating all its outputs.

[3]For MF-GP-UCB, the fidelity of each auxiliary function is characterized by the tightness of a heuristically specified bound on the supremum norm between the target function and itself.

the MF-GP-UCB algorithm will only evaluate the target function at a selected input if the square root of the predictive variance (at this input) of every GP modeling a separate auxiliary function is smaller than its corresponding threshold parameter. As shall be seen in Section 5.4, the performance of MF-GP-UCB is highly sensitive to the choice of these parameters which have to be manually tuned to make it work well in optimizing different target functions. Multi-fidelity BO algorithms based on EI [Forrester *et al.*, 2007; Huang *et al.*, 2006] have also been proposed but do not perform as well as MF-GP-UCB (see Section 6 and Appendix D.1 in [Kandasamy *et al.*, 2016]); the multi-fidelity sequential kriging optimization algorithm [Huang *et al.*, 2006] also relies on heuristically setting a single shared parameter in EI to control the exploration-exploitation trade-off over all target and auxiliary functions. In contrast, we will propose a *multi-fidelity predictive entropy search* (MF-PES) algorithm that can jointly and naturally optimize the exploration-exploitation trade-off without needing to manually tune any such parameters or that of EI to be discussed next.

5.3 Multi-Fidelity PES

Information-based acquisition functions (e.g., ES [Hennig and Schuler, 2012] and PES [Hernández-Lobato *et al.*, 2014]) have been constructed to enable the conventional BO algorithms to improve the belief of the maximizer of an unknown target function. In multi-fidelity BO, we can similarly define a belief of the maximizer x_{*_i} of each i-th function f_i as

$$p(x_{*_i}|y_X) \triangleq p(f_i(x_{*_i}) = \max_{x\in D} f_i(x)|y_X)$$

for $i = 1, ..., M$. To achieve the objective of maximizing *only* the target function in multi-fidelity BO, ES can be directly used to measure the information gain of *only* the target maximizer x_{*_t} from selecting the next input tuple $\langle x, i\rangle$ for evaluating the

i-th function f_i (i.e., possibly auxiliary) at x given the past observations (X, y_X):

$$\alpha(y_X, \langle x, i\rangle) \triangleq H(x_{*_t}|y_X) - \mathbb{E}_{p(y_{\langle x,i\rangle}|y_X)}[H(x_{*_t}|y_{X\cup\{\langle x,i\rangle\}})]. \tag{5.6}$$

The *multi-task ES* (MT-ES) algorithm [Swersky *et al.*, 2013] has used Monte Carlo sampling to approximate (5.6) but faced two critical limitations: (a) Computing (5.6) incurs cubic time in the size of the discretized input domain and is thus expensive to evaluate with a large input domain (or risks being approximated poorly), and (b) to reduce the considerable time in evaluating (5.6) over the entire discretized input domain and perform competitively, MT-ES heuristically prunes this search space to a small set of input candidates that are selected by applying EI to *only* the target function, hence artificially constraining the exploration of auxiliary functions and requiring a parameter in EI (i.e., to control the exploration-exploitation trade-off) to be manually tuned to fit different real-world applications.

To circumvent the above-mentioned issues, we can exploit the symmetric property of conditional mutual information and rewrite (5.6) as

$$\alpha(y_X, \langle x, i\rangle) = H(y_{\langle x,i\rangle}|y_X) - \mathbb{E}_{p(x_{*_t}|y_X)}[H(y_{\langle x,i\rangle}|y_X, x_{*_t})] \tag{5.7}$$

which we call *multi-fidelity PES* (MF-PES). Intuitively, the selection of an input tuple $\langle x, i\rangle$ to maximize (5.7) has to trade off between exploration of every target and auxiliary function (hence inducing a large Gaussian predictive entropy $H(y_{\langle x,i\rangle}|y_X)$) vs. exploitation of the current belief $p(x_{*_t}|y_X)$ of the target maximizer x_{*_t} to choose a nearby input x of a high-fidelity function f_i (i.e., convolutional structures and maximizers of the target and auxiliary functions are similar or close (Section 5.1)) to be evaluated (hence inducing a small expected predictive entropy $\mathbb{E}_{p(x_{*_t}|y_X)}[H(y_{\langle x,i\rangle}|y_X, x_{*_t})]$) to yield a highly informative observation that in turn improves the belief of x_{*_t}.

Due to (5.5), the first Gaussian predictive/posterior entropy term in (5.7) can be computed analytically:

$$H(y_{\langle x,i\rangle}|y_X) \triangleq 0.5\log(2\pi e(\sigma^2_{\langle x,i\rangle|X} + \sigma^2_{n_i})) \tag{5.8}$$

where $\sigma^2_{\langle x,i\rangle|X} \triangleq \Sigma_{\{\langle x,i\rangle\}\{\langle x,i\rangle\}|X}$ (5.5). Unfortunately, the second term in (5.7) cannot be evaluated in closed form. Although this second term appears to resemble that in PES [Hernández-Lobato *et al.*, 2014], their approximation method, however, cannot be applied straightforwardly here since it cannot account for the complex cross-correlation structure between the target and auxiliary functions. To achieve this, we will first propose a novel multi-output random features approximation of the CMOGP model whose cross-correlation (i.e., multi-fidelity) structure between the target and auxiliary functions can be exploited for sampling the target maximizer x_{*_t} more accurately using the past observations (X, y_X) from evaluating these functions (especially when the target function is noisy and/or sparsely evaluated due to cost), which is in turn used to approximate the expectation in (5.7). Then, we will formalize some practical constraints relating the global target maximizer to that of the auxiliary functions, which are used to approximate the second entropy term within the expectation in (5.7).

5.3.1 Multi-output random features (MRF) for sampling the target maximizer

To approximate the expectation in (5.7) efficiently by averaging over samples of the target maximizer from $p(x_{*_t}|y_X)$ in a continuous input domain, we will derive an analytic sample of the unknown target function[4] f_t given the past observations (X, y_X),

[4]Note that (5.5) gives an analytic expression of the CMOGP predictive mean of $f_t(x)$ but not of $f_t(x)$ itself. So, maximizing its predictive mean over x is not equivalent to maximizing $f_t(x)$.

which is differentiable and can be optimized by any existing gradient-based optimization method to search for its maximizer. Unlike the work of Hernández-Lobato *et al.* (2014) that achieves this in PES using the *single-output random features* (SRF)[5] method [Rahimi and Recht, 2007], we have to additionally consider how the complex cross-correlation (i.e., multi-fidelity) structure between the target and auxiliary functions can be exploited for sampling the target maximizer x_{*_t} more accurately, which is in turn used to approximate the expectation in (5.7). To address this, we will now present a novel *multi-output random features* (MRF) approximation of the CMOGP model by first deriving an analytic form of the latent function L with SRF and then an analytic approximation of f_i using the convolutional structure of the CMOGP model.

Using the results of Rahimi and Recht (2007), the prior covariance of the GP modeling L (Section 5.1) can be rewritten as

$$\begin{aligned}\sigma_{xx'} &= \alpha \int p(w)\ e^{-jw^\top (x-x')}\ \mathrm{d}w \\ &= 2\alpha\ \mathbb{E}_{p(w,b)}[\cos(w^\top x + b)\cos(w^\top x' + b)]\end{aligned} \tag{5.9}$$

where $p(w) \triangleq s(w)/\alpha$, $s(w)$ is the Fourier dual of $\sigma_{xx'}$, and $b \sim \mathcal{U}[0, 2\pi]$. Let $\phi(x)$ denote a random vector of an m-dimensional feature mapping of the input x:

$$\phi(x) \triangleq \sqrt{2\alpha/m}\ \cos(W^\top x + B) \tag{5.10}$$

where $W \triangleq (w_q)_{q=1,\ldots,m}$ and $B \triangleq (b_q)^\top_{q=1,\ldots,m}$ with w_q and b_q sampled from $p(w)$ and $p(b)$, respectively. From (5.9) and (5.10), the prior covariance $\sigma_{xx'}$ can be approximated by $\sigma_{xx'} \approx \phi(x)^\top \phi(x')$ and the latent function L can be approximated by a

[5]SRF has also been used by Lázaro-Gredilla *et al.* (2010) to derive a sparse spectrum (single-output) GP approximation.

linear model:

$$L(x) \approx \phi(x)^\top \theta \tag{5.11}$$

where $\theta \sim \mathcal{N}(\underline{0}, I)$ is an m-dimensional vector of weights. The derivations of (5.9) and (5.11) are shown in Appendix A of [Hernández-Lobato *et al.*, 2014]. Then, interestingly, by exploiting the convolutional structure of the CMOGP model in (5.2) and the SRF results in (5.10) and (5.11), $f_i(x)$ can also be approximated analytically by a linear model:

$$f_i(x) \approx \phi_i(x)^\top \theta \tag{5.12}$$

where the random vector

$$\phi_i(x) \triangleq \sigma_{s_i} \operatorname{diag}(e^{-\frac{1}{2}W^\top P_i^{-1} W}) \; \phi(x) \tag{5.13}$$

can be interpreted as input features of $f_i(x)$[6] and function $\operatorname{diag}(A)$ returns a diagonal matrix with the same diagonal components as A. The derivation of (5.12) is in Appendix B.1.

Using (5.12), we will now show how a sample of f_t can be constructed from a linear combination of samples of features ϕ_t and from the posterior of weights θ given the past observations (X, y_X). It follows from (5.1) and (5.12) that y_{X_i} is conditionally independent of $f_{X \setminus X_i}$, W, and B given f_{X_i} for $i = 1, \ldots, M$ and $f_{X_1}, \ldots, f_{X_M}$ are conditionally independent given θ, W, and B, respectively. Then,

$$p(y_X|\theta, W, B) = \prod_{i=1}^{M} \int p(y_{X_i}|f_{X_i}) \; p(f_{X_i}|\theta, W, B) \; \mathrm{d}f_{X_i} = \mathcal{N}(y_X|\Phi^\top\theta, \Sigma_\varepsilon)$$

where $\Phi \triangleq (\phi_j(x))_{\langle x,j\rangle \in X}$.

[6]The approximated covariance $\sigma_{\langle x,i\rangle\langle x',j\rangle} \approx \phi_i(x)^\top \phi_j(x')$ then characterizes the correlation within each function (i.e., $i = j$) and the cross-correlation between different functions (i.e., $i \neq j$).

As a result, the posterior distribution of θ is

$$p(\theta|y_X, W, B) = \mathcal{N}(\theta|A^{-1}\Phi\Sigma_\varepsilon^{-1}y_X, A^{-1}) \tag{5.14}$$

where $A = \Phi\Sigma_\varepsilon^{-1}\Phi^\top + I$. Let $\phi_i^{[s]}$ and $\theta^{[s]}$ denote vectors of features and weights sampled from (5.13) and (5.14), respectively. The i-th function f_i can then be approximated by

$$f_i^{[s]}(x) \triangleq \phi_i^{[s]}(x)^\top\theta^{[s]} \ . \tag{5.15}$$

Consequently, the expectation in (5.7) can be approximated by averaging over S samples of the target maximizer $x_{*_t}^{[s]}$ of $f_t^{[s]}$ to yield the following approximation of MF-PES:

$$\alpha(y_X, \langle x, i\rangle) \approx H(y_{\langle x,i\rangle}|y_X) - S^{-1}\sum_{s=1}^{S} H(y_{\langle x,i\rangle}|y_X, x_{*_t}^{[s]}) \tag{5.16}$$

where, for $i = 1, \ldots, M$,

$$x_{*_i}^{[s]} \triangleq \arg\max_{x\in D} f_i^{[s]}(x) \ . \tag{5.17}$$

Drawing a sample of $x_{*_i}^{[s]}$ incurs $\mathcal{O}(m^3 + m^2|X|)$ time if $m \leq |X|$ and $\mathcal{O}(|X|^3 + |X|^2 m)$ time if $m > |X|$, which is more efficient than using Thompson sampling [Chapelle and Li, 2011] to sample f_i over a discretized input domain that incurs cubic time in its size since a sufficiently fine discretization of the entire input domain is typically larger in size than the number $|X|$ of observations.

5.3.2 Approximating the predictive entropy conditioned on the target maximizer

In this subsection, we will discuss how the second entropy term in (5.16) is approximated. Firstly, the posterior distribution of $y_{\langle x,i\rangle}$ given the past observations and

target maximizer is computed by

$$p(y_{\langle x,i\rangle}|y_X, x_{*_t}) = \int p(y_{\langle x,i\rangle}|f_i(x))\ p(f_i(x)|y_X, x_{*_t})\ \mathrm{d}f_i(x) \tag{5.18}$$

where $p(y_{\langle x,i\rangle}|f_i(x))$ is a Gaussian distribution and $p(f_i(x)|y_X, x_{*_t})$ will be approximated by *expectation propagation* (EP), as detailed later. As shown in Section 5.1, the Gaussian predictive distribution $p(f_i(x)|y_X)$ can be computed analytically using (5.5). Then, $p(f_i(x)|y_X, x_{*_t})$ can be considered as a constrained version of $p(f_i(x)|y_X)$ by further conditioning on the target maximizer x_{*_t}. It is intuitive that the posterior distribution of $f_i(x)$ is constrained by

$$f_i(x) \leq f_i(x_{*_i}), \forall \langle x, i\rangle \in D^+ \ .$$

However, since only the target maximizer x_{*_t} is of interest, how should the value of $f_i(x)$ be constrained by x_{*_t} instead of x_{*_i} if $i \neq t$? To resolve this, we introduce a slack variable c_i to formalize the relationship between maximizers of the target and auxiliary functions:

$$f_i(x) \leq f_i(x_{*_t}) + c_i \quad \forall x \in D, i \neq t \tag{5.19}$$

where $c_i \triangleq \mathbb{E}_{p(x_{*_i}|y_X)}[f_i(x_{*_i})] - \mathbb{E}_{p(x_{*_t}|y_X)}[f_i(x_{*_t})]$ measures the gap between the expected maximum of f_i and the expected output of f_i evaluated at x_{*_t} and can, surprisingly, be approximated efficiently using the result of MRF even though f_i is unknown, as detailed later. To capture above-mentioned constraints analytically, the following simplified constraints instead of (5.19) are used to approximate $p(f_i(x)|y_X, x_{*_t})$:

*C*1. $f_i(x) \leq f_i(x_{*_t}) + \delta_i c_i$ for a given $\langle x, i\rangle \in D^+$ where δ_i equals to 0 if $i = t$, and 1 otherwise.

*C*2. $f_j(x_{*_t}) + \delta_j c_j \geq y_{\max_j} + \varepsilon_j$ for $j = 1, \ldots, M$ where $y_{\max_j} \triangleq \max_{\langle x,i\rangle \in X_j} y_{\langle x,i\rangle}$ is

the largest among the noisy outputs observed by evaluating f_j at X_j.

The first constraint $C1$ keeps the influence of x_{*_t} to the next input tuple $\langle x, i\rangle$ to be selected by MF-PES. Instead of constraining all unknown functions over the entire input domain, $C2$ relaxes (5.19) to be valid only for the noisy outputs observed from evaluating these functions. Using these constraints, we will first derive a tractable approximation of the posterior distribution $p(f_i(x_{*_t})|y_X, C2)$ which does not depend on the next selected input x. Note that such terms can be computed once and reused in the approximation of $p(f_i(x)|y_X, x_{*_t})$ in (5.18) which depends on x, as detailed later.

5.3.2.1 Approximating terms independent of x

Let $f_j^* \triangleq f_j(x_{*_t})$ and $f^* \triangleq (f_j^*)_{j=1,\ldots,M}^\top$. We can use the cdf of a standard Gaussian distribution to represent the probability of $C2$ and constrain the posterior distribution $p(f^*|y_X)$ with $C2$ by

$$p(f^*|y_X, C2) \propto p(f^*|y_X) \prod_{j=1}^{M} \Phi_{\text{cdf}}\left((f_j(x_{*_t}) + c_j - y_{\max_j})/\sigma_{n_j}\right). \tag{5.20}$$

Interestingly, by sampling the target and auxiliary maximizers x_{*_t} and x_{*_j} using the method proposed in Section 5.3.1, the value of c_j in (5.20) can be approximated in practice by Monte Carlo sampling[7]:

$$c_j = \mathbb{E}_{p(x_{*_j}|y_X)}[f_j(x_{*_j})] - \mathbb{E}_{p(x_{*_t}|y_X)}[f_j(x_{*_t})] \approx S^{-1} \sum_{s=1}^{S} \left(f_j^{[s]}(x_{*_j}^{[s]}) - f_j^{[s]}(x_{*_t}^{[s]})\right).$$

With the multiplicative form of (5.20) , $p(f^*|y_X, C2)$ can be approximated to be a multivariate Gaussian distribution $\mathcal{N}(f^*|\mu, \Sigma)$ using EP by approximating each non-

[7]When $j = t$, c_j is equal to 0 since $x_{*_j} = x_{*_t}$.

Gaussian factor (i.e., Φ_{cdf}) in (5.20) to be a Gaussian, as detailed in Appendix B.2. Consequently, the posterior distribution $p(f_i^*|y_X, C2)$ can be approximated by a Gaussian $\mathcal{N}(f_i^*|\mu_i, \tau_i)$ where μ_i is the i-th component of μ and τ_i is the i-th diagonal component of Σ.

5.3.2.2 Approximating terms that depend on x

In $C2$, f_i^* is the only term that is related to $C1$. It follows that $f_i(x)$ is conditionally independent of $C2$ given f_i^*. Let $f^+ \triangleq [f_i(x_{*_t}); f_i(x)]$, then

$$p(f^+|y_X, C2) = p(f_i(x)|y_X, f_i^*)\ p(f_i^*|y_X, C2) = \mathcal{N}(f^+|\mu^+, \Sigma^+) \tag{5.21}$$

where μ^+ and Σ^+ can be computed analytically using μ_i, τ_i and (5.5), as detailed in Appendix B.3.

To involve $C1$, an indicator function $\mathbb{I}(f_i(x) \leq f_i(x_{*_t}) + \delta_i c_i)$ is used to represent the probability that $C1$ holds. Then, $p(f_i(x)|y_X, x_{*_t}) \approx \int p(f^+|y_X, C1, C2)\ \mathrm{d}f_i^*$ where

$$p(f^+|y_X, C1, C2) \approx Z'^{-1} p(f^+|y_X, C2)\ \mathbb{I}(f_i(x) \leq f_i(x_{*_t}) + \delta_i c_i) \tag{5.22}$$

Since the posterior of $f_i(x_{*_t})$ has been updated according to $C2$ (5.20), c_i in (5.22) is updated likewise:

$$c_i \approx S^{-1} \sum_{s=1}^{S} \left(f_i^{[s]}(x_{*_i}^{[s]}) - \mu_i^{[s]} \right)$$

where $\mu_i^{[s]}$ is computed in (5.20) using a sampled $x_{*_t}^{[s]}$. Similar to that in [Hernández-Lobato *et al.*, 2014], a one-step EP can be used to approximate (5.22) as a multivariate Gaussian distribution with posterior covariance matrix

$$\Sigma_{f^+} \triangleq \Sigma^+ - v^{-1}\gamma(\gamma - (\eta - \delta_i c_i)/\sqrt{v})\ \Sigma^+ a a^\top \Sigma^+ \tag{5.23}$$

where $a = [-1; 1]$, $\gamma \triangleq \phi((\delta_i c_i - \eta)/\sqrt{v})/\Phi_{\text{cdf}}((\delta_i c_i - \eta)/\sqrt{v})$, $\eta \triangleq a^\top \mu^+$, and $v \triangleq a^\top \Sigma^+ a$. The derivation of (5.23) is in Appendix B.4. So, the posterior variance of $p(f_i(x)|y_X, x_{*_t})$ can be approximated using the $(2,2)$-th component of Σ_{f^+} denoted by v_{f_i} and its posterior entropy can consequently be approximated by

$$H(y_{\langle x,i\rangle}|y_X, x_{*_t}) \approx 0.5 \log(2\pi e(v_{f_i} + \sigma_{n_i}^2)) \tag{5.24}$$

due to (5.18). Using (5.8) and (5.16), it follows that MF-PES (5.7) can be approximated by

$$\alpha(y_X, \langle x, i\rangle) \approx \frac{1}{2} \log\left(\sigma_{\langle x,i\rangle|X}^2 + \sigma_{n_i}^2\right) - \frac{1}{2S} \sum_{s=1}^{S} \log\left(v_{f_i}^{[s]} + \sigma_{n_i}^2\right).$$

When the costs of evaluating target vs. auxiliary functions differ, we use the following cost-sensitive MF-PES instead: $\alpha_{\text{cost}}(y_X, \langle x, i\rangle) \triangleq \alpha(y_X, \langle x, i\rangle)/\text{cost}(i)$ which can be interpreted as the information gain of the target maximizer per cost of evaluating the i-th function f_i from selecting the next input tuple $\langle x, i\rangle$. Since such a cost (e.g., time incurred to train a ML model) is usually not known, we need a method to estimate it in real-world applications, which will be discussed later in Section 5.4.

5.4 Experiments and Discussion

This section empirically evaluates the multi-fidelity BO performance of our MF-PES algorithm against that of (a) PES [Hernández-Lobato *et al.*, 2014], (b) MT-ES [Swersky *et al.*, 2013] performing Monte Carlo approximation of (5.6), (c) MF-GP-UCB with all parameters trading off between exploitation vs. exploration set according to that recommended in [Kandasamy *et al.*, 2016], (d) MF-GP-UCB*: MF-GP-UCB

with carefully fine-tuned parameters[8], and (e) MF-rand: an additional baseline which selects the value of i from $1, ..., M$ randomly and then use PES over the selected function f_i to choose x, which is used to elaborate the advantage of MF-PES in automatically selecting the type of function (i.e., target or auxiliary function) to evaluate.

For a fair comparison, CMOGP is used to model multiple functions in *all* tested algorithms since the tested algorithms (MT-ES and MF-GP-UCB) can cater to any GP model and their performance would thus be improved by using CMOGP which has been empirically demonstrated by [Álvarez and Lawrence, 2011] to outperform the models used by MT-ES and MF-GP-UCB. Note that the multi-fidelity sequential kriging optimization algorithm [Huang *et al.*, 2006] is not evaluated here since MF-GP-UCB outperforms it in both synthetic and real-world experiments, as empirically demonstrated in [Kandasamy *et al.*, 2016].

In all experiments, we use $m \triangleq 200$ random features and $S \triangleq 50$ samples of the target maximizer in MF-PES. The CMOGP hyperparameters are learned via maximum likelihood estimation [Álvarez and Lawrence, 2011]. The performance of the tested algorithms are evaluated using *immediate regret* (IR) $|f_t(x_{t_*}) - f_t(\tilde{x}_{t_*})|$ where $\tilde{x}_{t_*} \triangleq \arg\max_{x \in D} \mu_{\{\langle x,t \rangle\}|X}$ is their recommended target maximizer. In each experiment, one observation of the target function is randomly selected as the initialization. The standard error is computed as the error bar in all the results. Costs of evaluating the target and auxiliary functions are assumed to be, respectively, 10 and 1 in all synthetic experiments.

[8]Using the parameters recommended in [Kandasamy *et al.*, 2016], MF-GP-UCB does not perform well in most of our experiments. To achieve a fair comparison, we carefully fine-tune its parameters to make it perform well in every experiment.

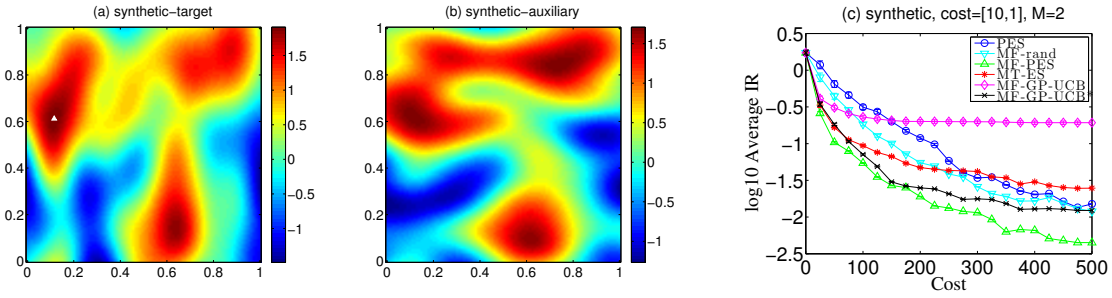


Figure 5.1: (a-b) Examples of the synthetic functions where '△' is the global target maximizer. (c) Graphs of $\log_{10}$(averaged IR) vs. cost incurred by tested algorithms for synthetic functions.

5.4.1 Synthetic experiments

The performance of the tested algorithms are first evaluated using the following synthetic and benchmark functions.

Synthetic functions. The synthetic functions are generated with different fidelities using $M \triangleq 2$ and $D \triangleq [0,1]^2$. To do this, the CMOGP hyperparameters with one latent function are first fixed as $P_0 \triangleq \text{diag}[100, 100], P_1 \triangleq \text{diag}[2000, 100], P_2 \triangleq \text{diag}[100, 2000], \sigma_{s_1} \triangleq \sigma_{s_2} \triangleq 1, \sigma_{n_1}^2 \triangleq 0.01$, and $\sigma_{n_2}^2 \triangleq 0.001$ which are also used in the tested algorithms as optimal hyperparameters. Then, a set X of 450 input tuples is uniformly sampled from D^+ and their corresponding outputs are sampled from the CMOGP prior. The target and auxiliary functions are set to be the predictive mean $\mu_{\{\langle x,i\rangle\}|X}$ of the CMOGP model with $i = 1$ and $i = 2$, respectively. An example of the synthetic functions can be found in Fig. 5.1a-b. Ten pairs (i.e., one target and one auxiliary) of synthetic functions are generated using the above procedure. An averaged IR is obtained by optimizing the target function in each of them with 10 different initializations for each tested algorithm.

Hartmann-6D function. In these experiments, the original Hartmann-6D function is used as target function and $M \triangleq 2$ or 3. Similar to that in [Kandasamy *et al.*, 2016], three auxiliary functions of varying degrees of fidelity are constructed by

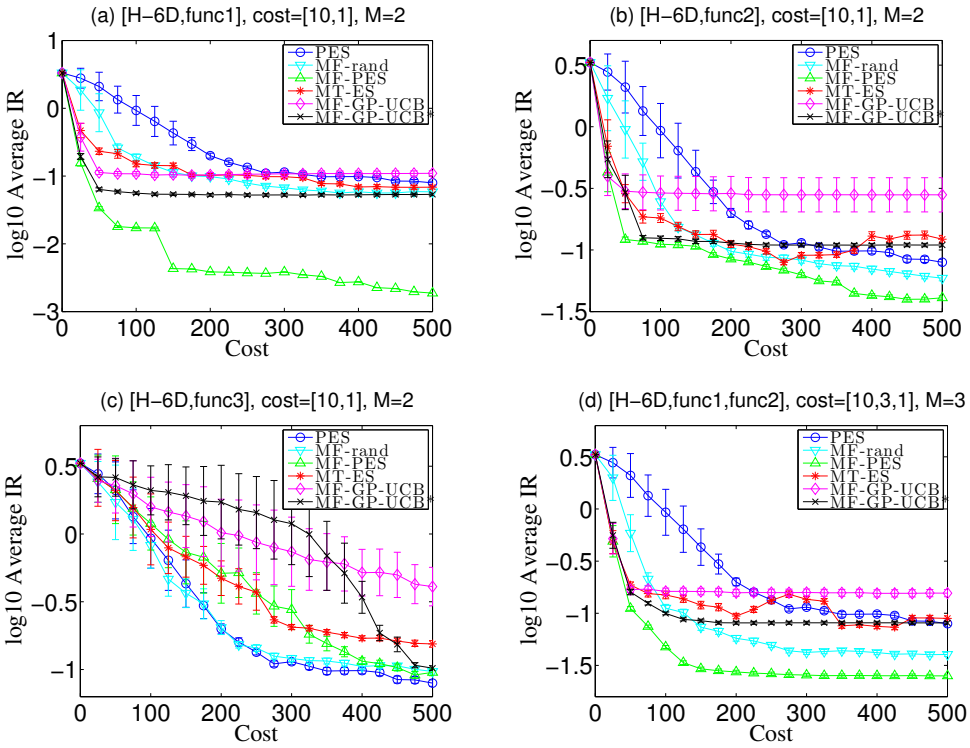


Figure 5.2: Graphs of $\log_{10}$(averaged IR) vs. cost incurred by tested algorithms for Hartmann-6D (H-6D) function and its auxiliary functions func1, func2, and func3 with the respective fidelities ρ^1, ρ^2, and ρ^3 computed using (5.4) where $\rho^1 > \rho^2 > \rho^3$. The type, cost, and number of the functions used in each experiment are shown in the title of each graph. The number Q (B.18) of latent functions used in CMOGP to model the target and auxiliary functions is $Q = 1$, $Q = 2$, $Q = 2$ and $Q = 2$, respectively, for (a)-(d).

tweaking the Hartmann-6D function, as detailed in Appendix B.6. The experiments are run with 10 different initializations.

Figs. 5.1c and 5.2 show results of all tested algorithms for synthetic and Hartmann functions, respectively, with a cost budget of 500. It can be observed from Figs. 5.1c, 5.2a-b and 5.2d that MF-PES can achieve a much lower averaged IR with considerably less cost than PES, which implies that the BO performance can be improved by auxiliary function(s) of sufficiently high fidelity and low evaluation cost and noise. Both the Hartmann and synthetic functions are difficult to optimize due

to their multimodal nature (e.g., induced by large values in P_0) and/or large input domain, which causes MT-ES and MF-GP-UCB to be trapped easily in some local maximum and hence perform not as well. We have dedicated time to carefully fine-tune the parameters of MF-GP-UCB* such that it explores more to perform better than MF-GP-UCB but is still outperformed by MF-PES. In contrast, MF-PES is rarely trapped in a local maximum and performs significantly better than all the other tested algorithms by naturally exploring more over these multimodal functions. Moreover, Fig. 5.2c shows that when the fidelity of the auxiliary function is very low ($\rho^3 = 0.0037$, as shown in Appendix B.6), MF-PES can achieve a comparable performance to PES, hence demonstrating its robustness to a low-fidelity auxiliary function. Lastly, as shown in Figs. 5.1, 5.2a-b and 5.2d, the performance of MF-rand is between that of PES and MF-PES when the auxiliary function(s) has sufficiently high fidelity. Conversely, when the auxiliary function has very low fidelity (Fig. 5.2c), the performance of MF-rand is similar to that of PES. Such observations showed that MF-PES naturally provides better strategies for selecting the type of function compared to the random method.

Branin-Hoo function. The performance of MF-PES is also evaluated using auxiliary functions with different fidelities based on the well-known benchmark Branin-Hoo function as the target function. Three auxiliary functions func1, func2, and func3 with the corresponding fidelities ρ^1, ρ^2, and ρ^3 are constructed by, respectively, (a) decreasing only the noise variance, (b) shifting Branin-Hoo along both axes, and (c) using the Currin exponential function to yield $\rho^1 > \rho^2 > \rho^3$, as detailed in Appendix B.6.

Fig. 5.3 shows results of the averaged IR over 50 different initializations. Similar to the results for Hartmann-6D, MF-PES achieves a much lower averaged IR with less cost than PES when the auxiliary function is of a sufficiently high fidelity and low evaluation cost (i.e., Fig. 5.3a-b). Also, the performance of MF-PES relative

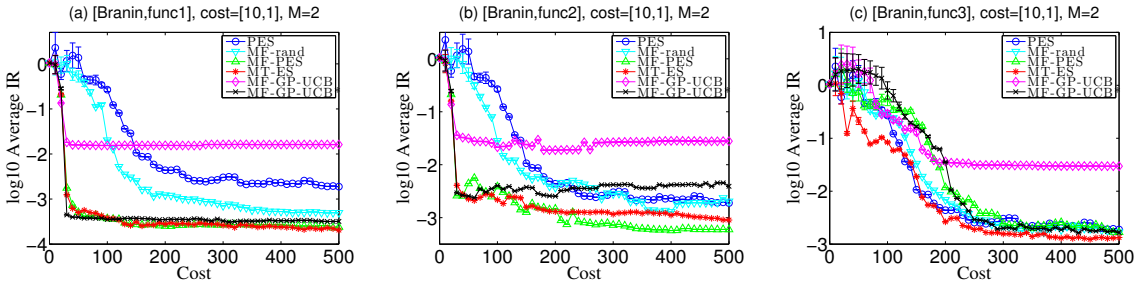
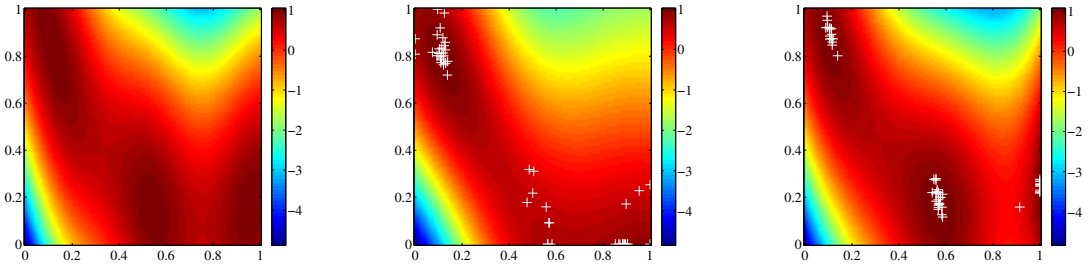


Figure 5.3: Graphs of $\log_{10}$(averaged IR) vs. cost incurred by tested algorithms for Branin function and its auxiliary functions func1, func2, and func3 with the respective fidelities ρ^1, ρ^2, and ρ^3 computed using (5.4) where $\rho^1 > \rho^2 > \rho^3$. The type, cost, and number of the functions used in each experiment are shown in the title of graph. The number Q (B.18) of latent functions used in CMOGP to model the target and auxiliary functions is $Q = 1$, $Q = 1$, $Q = 2$, respectively, for (a)-(c).

to that of PES reveals that the BO performance can be improved by an auxiliary function func1 less noisy than (but identical to) the target function due to a smaller noise variance. As shown in Fig. 5.3c, the performance of MF-PES is initially hurt by an auxiliary function with an extremely low fidelity (i.e., func3 with $\rho_2^3 = 0.0683$) but eventually converges to the same performance as PES with enough observations. Finally, the performance of MF-PES is similar to that of MT-ES and MF-GP-UCB* in Fig. 5.3. This is because Branin function which has three global maxima and no local maximum is very easy to be optimized. This easy-to-optimize Branin function makes MF-PES loses its advantage of rarely being trapped in a local maximum as mentioned in Section 5.4.

We have also investigated the effectiveness of the MRF approximation in improving the belief of the target maximizer of the Branin-Hoo function over the SRF method as shown in Fig. 5.4. In particular, it can be observed from Fig. 5.4 that MRF can sample the target maximizer more accurately than SRF.



(a) Branin-Hoo (b) SRF (c) MRF

Figure 5.4: Surface plots of the (a) true Branin-Hoo function as the target function (note that the auxiliary function is constructed by shifting 10% of this function along both axes), (b) predicted target function by SRF with 10 observations from evaluating the target function, and (c) predicted target function by MRF with 10 and 50 observations from evaluating the target and auxiliary functions, respectively. Note that '$+$' denotes a location of the sampled target maximizer.

5.4.2 Real-world experiments

In this subsection, the tested algorithms are used to automatically tune the hyperparameters of ML models in two image classification tasks:

- **Logistic regression (LR) with MNIST dataset.** The four LR hyperparameters to be tuned in our experiments are the learning rate of *stochastic gradient descent* (SGD) in the range of $[10^{-5}, 1]$, l_2 regularization parameter in the range of $[10^{-5}, 1]$, batch size in the range of $[20, 1000]$, and number of learning epochs in the range of $[5, 100]$.

- **Convolutional neural network (CNN) with CIFAR-10 dataset.** The six CNN[9] hyperparameters to be tuned in our experiments are the learning rate of SGD in the range of $[10^{-5}, 1]$, three dropout rates in the range of $[0, 1]$, batch size in the range of $[100, 1000]$, and number of learning epochs in the range of $[100, 1000]$.

[9]We use the same CNN structure as the example code of keras: `https://github.com/fchollet/keras/blob/master/examples/cifar10_cnn.py` and switch the optimizer in their code to SGD.

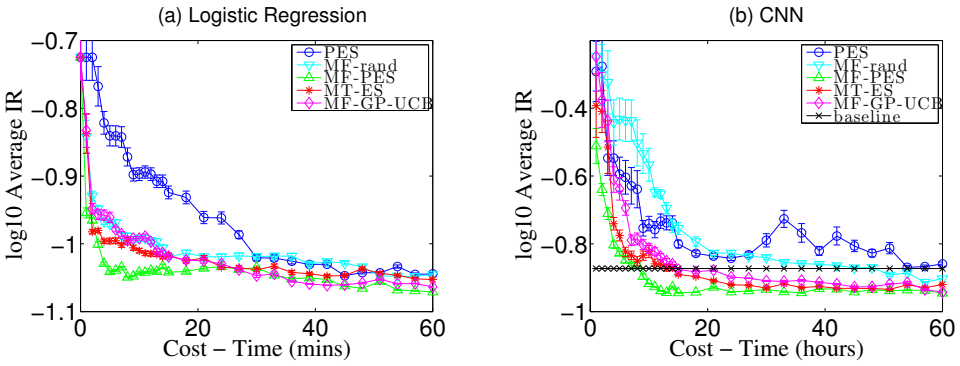


Figure 5.5: Graphs of $\log_{10}$(averaged IR) vs. cost incurred by tested algorithms for (a) LR and (b) CNN.

For each task, we use training and validation data of size 50000 and 10000, respectively. The unknown target function to be maximized is the validation accuracy evaluated by training the ML model with all the training data. The unknown auxiliary function is also the validation accuracy evaluated by training the same ML model but with a smaller fixed dataset of size 10000 randomly selected from the original training data. The evaluation cost should be the time for training the ML model. However, since the training time is not known and varies with different settings of hyperparameters, the costs for evaluating the target and auxiliary functions are estimated to be, respectively, 5 and 1 according to their training data size. The actual total training time is shown in the results of all our experiments. Furthermore, since the optimal hyperparameters (i.e., global target maximizer x_{*_t}) is now known in these real-world problems, $f_t(x_{t_*}) = 1$ is used to compute IR and $f_t(\tilde{x}_{t_*})$ is evaluated by training the ML model with $\tilde{x}_{t_*}$ for the tested algorithms.

Fig. 5.5a-b shows the results of the tested algorithms with 10 (5) different initializations for the image classification task using LR (CNN). It can be observed that MF-PES converges faster to a smaller IR than other tested algorithms. Also, MF-PES improves the performance of CNN compared to the baseline achieved using the default hyperparameters in the existing code, which shows that MF-PES is promising

in finding more competitive hyperparameters of complex ML models.

5.5 Summary

This chapter describes a novel MF-PES algorithm for multi-fidelity BO that can naturally trade off between exploitation vs. exploration over the target and auxiliary functions with varying fidelities without needing to manually tune any such parameters. Our MF-PES algorithm utilizes a novel MRF approximation of the CMOGP model whose cross-correlation (i.e., multi-fidelity) structure between the target and auxiliary functions can be exploited for sampling the target maximizer more accurately using the observations from evaluating these functions. Empirical evaluation on synthetic functions, benchmark functions and image classification tasks using LR and CNN with real-world MNIST and CIFAR-10 datasets shows that MF-PES outperforms the state-of-the-art multi-fidelity BO algorithms.

However, one limitation of the experiments is that of the prior trained hyperparameters: The results in Figs 5.5 are achieved with *fixed* CMOGP/GP hyperparameters which are learned via maximum likelihood estimation using prior collected observations. Such step is used to make a fair comparison of all tested multi-fidelity BO algorithms by avoiding the effect of the non-stable CMOGP hyperparameters learning. Compared to the single-fidelity BO algorithm (i.e., PES), 18.5% and 21.1% additional time is used to collect the auxiliary observations as training data for LR and CNN, respectively. Therefore, the training time of CMOGP in multi-fidelity BO is not much larger than that of single-fidelity BO. Such prior trained CMOGP/GP hyperparameters, however, is still a little bit unfair for PES, which will be resolved in our future work by developing stable CMOGP hyperparameters training algorithm such that the CMOGP hyperparameters could be updated on-the-fly in all tested BO algorithms.

Chapter 6

PES for High-Dimensional BO

In Chapter 5, we have generalized the PES acquisition function to multi-fidelity BO. This chapter will focus on another technical challenge of BO (i.e., high input dimensions) and present a novel generalization of PES for high-dimensional BO. In particular, to improve the scalability of PES in the number of input dimensions, we exploit the structure of additive GP (Section 3.3) such that the original PES can be decomposed into a sum of *local* PESs, each of which depends only on a subset of low input dimensions, and thus, could be optimized independently. However, such an *additive PES* (add-PES) cannot be computed by applying the approximation algorithm of PES [Hernández-Lobato *et al.*, 2014] straightforwardly to each local component since the constraints required for making an efficient approximation of each local PES are not independent. To resolve this issue, novel approximation steps of add-PES are derived in Section 6.2. More interestingly, compared to the state-of-the-art high-dimensional BO algorithms, we empirically demonstrate that our add-PES can easily achieve an appropriate exploration-exploitation trade-off without the effort in tuning such parameter in all tested functions (Section 6.3), which makes it very promising to work well in different real-world applications.

6.1 High-Dimensional BO

Let $f(x)$ be a black-box function defined over a bounded d-dimensional input domain $D \subset \mathbb{R}^d$ such that each input $x \in D$ is associated with a noisy output $y_x \sim \mathcal{N}(f(x), \sigma_n^2)$. Recall that BO is used to globally optimize such a function $f(x)$ when its derivatives are unknown and its evaluation at some $x \in D$ is very expensive. Let $y_X \triangleq (y_x)^\top_{x \in X}$ denote a vector of noisy outputs observed by evaluating f at a set $X \subset D$ of inputs. Conventionally, a BO algorithm iteratively selects the next input x^+ for evaluating the function f at x^+ by maximizing some choice of acquisition function given the past observations (X, y_X):

$$x^+ \triangleq \arg\max_{x \in D} \alpha(y_X, x) \tag{6.1}$$

and updates $X \leftarrow X \cup \{x\}$ until the budget is expended. Although it's usually easy to evaluate $\alpha(y_X, x)$ and its gradients, the computational cost of (6.1), however, grows exponentially in d and is very expensive for high input dimensions (i.e., large d).

To resolve this issue, we will exploit the structure of additive GP (Section 3.3) for improving the scalability of (6.1) in the number of input dimensions. Recall from Section 3.3 that an additive GP assumes that the function $f(x)$ can be decomposed into a sum of independent local functions:

$$f(x) = f^{(1)}(x^{(1)}) + f^{(2)}(x^{(2)}) + ... + f^{(C)}(x^{(C)}) \tag{6.2}$$

where $f^{(i)}(x^{(i)})$ depends only on a d_i-dimensional input $x^{(i)}$, $d_i \ll d$ and $x^{(i)}$ are *disjoint* components of x for $i = 1, ..., C$. Let each $\{f^{(i)}(x^{(i)})\}_{x^{(i)} \in D^{(i)}}$ be an independent GP for $i = 1, ..., C$. Then, the additive GP model can provide a Gaussian predictive distribution $\mathcal{N}(\mu^{(i)}_{x^{(i)}|X}, \Sigma^{(i)}_{x^{(i)}x^{(i)}|X})$ of $f^{(i)}(x^{(i)})$ for any $x^{(i)} \in D^{(i)}$ using (3.8). Given such a *local* predictive distribution, Kandasamy *et al.* (2015) has generalized

UCB acquisition function to high-dimensional BO using $\alpha(y_X, x) \triangleq \sum_{i=1}^{C} \mu^{(i)}_{x^{(i)}|X} + \sqrt{\beta \Sigma^{(i)}_{x^{(i)}x^{(i)}|X}}$. Their algorithm, however, requires this parameter β to be carefully set for achieving good exploration-exploitation trade-off, which is not easy in practice since setting beta requires the consideration of several variables like input dimension and time step[1]. In contrast, we will propose an *additive PES* (add-PES) algorithm which is less tedious in finding an appropriate exploration-exploitation trade-off to achieve good BO performance for different functions with varying input dimensions, as will be discussed next.

6.2 Additive PES for High-Dimensional BO

Recall that in conventional BO algorithms, information-based acquisition functions (e.g., ES [Hennig and Schuler, 2012] and PES [Hernández-Lobato *et al.*, 2014]) has been constructed to improve the belief $p(x_*|y_x)$ of the target maximizer x_* where $x_* \triangleq \arg\max_{x \in D} f(x)$. To make such acquisition functions scale well in the number of input dimensions, we exploit the independent structure in additive GP and construct an alternative acquisition function which improves the belief of the maximizer for each local function $f^{(i)}$ independently:

$$\alpha(x, y_X) = \sum_{i=1}^{C} \left(H(x^{(i)}_* | y_X) - \mathbb{E}_{p(f^{(i)}(x^{(i)}))|y_X)}[H(x^{(i)}_* | y_X, f^{(i)}(x^{(i)}))] \right) \tag{6.3}$$

where $x^{(i)}_* \triangleq \arg\max_{x^{(i)} \in D^{(i)}} f^{(i)}(x^{(i)})$. However, it is very expensive to approximate (6.3) since $p(x^{(i)}_*|y_X)$ is analytically intractable [Hennig and Schuler, 2012]. To resolve this issue, Wang and Jegelka (2017) proposed an *additive max-value ES* (add-MES)

[1]Although Kandasamy *et al.* (2015) has provided an expression of β to achieve no regret in the limit, such an expression of β either (a) typically yields a value that does not achieve a good BO performance fast enough in practice due to a limited sampling budget, or (b) is too complicated to be computed in practice. So, they suggested a heuristic equation for computing β in their experiments which unfortunately still doesnt perform as well as expected, as will be shown in Section 6.3.

algorithm which simplified (6.3) by replacing all $x_*^{(i)}$ in (6.3) with $f^{(i)}(x_*^{(i)})$. In other words, their algorithm aims to improve the belief of the maximal function value $f(x_*)$ instead of the maximizer x_*. As will be demonstrated in Section 6.3, such a simplifying assumption by add-MES can result in a false sense of certainty about the maximizer x_* (when, in reality, it should be highly uncertain) due to a high degree of certainty about the maximum $f(x_*)$ arising from the presence of local maxima. When this happens, add-MES can be trapped easily in a local maximum since it wrongly perceives that it does not need to explore more. Our add-PES algorithm avoids above issue of add-MES and is designed to approximate the original form of (6.3) directly, as will be shown next.

Similar to Section 5.3, we exploit the symmetric property of conditional mutual information and rewrite (6.3) as

$$\alpha(x, y_X) = \sum_{i=1}^{C} \Big(H(f^{(i)}(x^{(i)})|y_X) - \mathbb{E}_{p(x_*^{(i)}|y_X)}[H(f^{(i)}(x^{(i)})|y_X, x_*^{(i)})] \Big) \tag{6.4}$$

Since $\max \alpha(x, y_X) = \max \sum_{i=1}^{C} \alpha^{(i)}(x^{(i)}, y_X) = \sum_{i=1}^{C} \max \alpha^{(i)}(x^{(i)}, y_X)$ where

$$\alpha^{(i)}(x^{(i)}, y_X) \triangleq H(f^{(i)}(x^{(i)})|y_X) - \mathbb{E}_{p(x_*^{(i)}|y_X)}[H(f^{(i)}(x^{(i)})|y_X, x_*^{(i)})] \tag{6.5}$$

and depends on only a *disjoint* subset of x, we can select the next input x to be evaluated for $f(x)$ by maximizing each local acquisition function $\alpha^{(i)}(x^{(i)}, y_X)$ for $i = 1, ..., C$ independently: $x^+ \triangleq \arg\max_{x \in D} \alpha(x, y_X) = \oplus_{i=1}^{C} \arg\max_{x^{(i)} \in D^{(i)}} \alpha^{(i)}(x^{(i)}, y_X)$. Due to (3.8), the first Gaussian predictive entropy term in (6.5) can be computed analytically:

$$H(f^{(i)}(x^{(i)})|y_X) \triangleq 0.5 \log(2\pi e \Sigma^{(i)}_{x^{(i)}x^{(i)}|X}) \tag{6.6}$$

To approximate the second term, we will first approximate the expectation using

an additive form of random features for sampling $x^{(i)}$, and then, propose some novel practical constraint for approximating the conditional entropy term, as detailed later.

6.2.1 Additive random features for sampling the high-dimensional target maximizer

Recall from Section 5.3.1 that an unknown function $f(x)$ modeled using GP can be approximated analytically by a linear model using *single-output random features* (SRF) [Rahimi and Recht, 2007] method. The original PES algorithm [Hernández-Lobato *et al.*, 2014] has exploited this result to sample x_* by maximizing samples of the linear model. Such a sampling method, however, also scales poorly in the number of input dimensions due to the optimization step. To resolve this issue, Wang and Jegelka (2017) has derived an *additive random features* (add-RF) method which will be used in this work for approximating the expectation in (6.5) efficiently.

Using the result of SRF [Rahimi and Recht, 2007] which has been reviewed in Section 5.3.1, each local function $f^{(i)}(x^{(i)})$ can be approximated by a linear model:

$$f^{(i)}(x^{(i)}) \approx \phi^{(i)}(x^{(i)})^\top \theta^{(i)} \tag{6.7}$$

where $\phi^{(i)}(x^{(i)}) \triangleq \sqrt{2\alpha/m}\cos(W_i^\top x^{(i)} + B_i)$ for $i = 1, ..., C$, W_i and B_i are defined same as in (5.10). Then, due to (6.2) and (6.7), $f(x)$ can also be approximated by a linear model:

$$f(x) = \sum_{i=1}^{C} f^{(i)}(x^{(i)}) \approx \sum_{i=1}^{C} \phi^{(i)}(x^{(i)})^\top \theta^{(i)} = \phi(x)^\top \theta \tag{6.8}$$

where $\phi(x) \triangleq (\phi^{(i)}(x^{(i)})^\top)^\top_{i=1,...,C}$, $\theta \triangleq (\theta^{(i)\top})^\top_{i=1,...,C}$. Then, due to (6.8) and $y_x \sim$

$\mathcal{N}(f(x), \sigma_n^2)$,

$$p(y_X|\theta, W, B) = \int p(y_X|f_X)p(f_X|\theta, W, B)\mathrm{d}f_X = \mathcal{N}(y_X|\Phi^\top\theta, \sigma_n^2 I)$$

where $\Phi \triangleq (\phi(x))_{x\in X}$. As a result, the posterior of θ is

$$p(\theta|y_X, W, B) = \mathcal{N}(\theta|\sigma_n^{-2}A^{-1}\Phi y_X, A^{-1}) \tag{6.9}$$

where $A \triangleq \sigma_n^{-2}\Phi\Phi^\top + I$. Let $\theta^{[s]}$ denote a sample of θ from (6.9). The i^{th} component of $\theta^{[s]}$ can be treated as a sample of $\theta^{(i)}$, and thus, be used with a sample of $\phi^{(i)}(x^{(i)})$ to approximated a sample of $f^{(i)}(x^{(i)})$ using (6.7). As a result, a sample of $x^{(i)}$ can be achieved by maximizing the sample of $f^{(i)}(x^{(i)})$ for $i = 1, ..., C$ independently. Let $X_*^{(i)}$ denote a set of samples of $x^{(i)}$ for $i = 1, ..., C$. Then, the expectation in (6.5) can be approximated by averaging over the samples in $X_*^{(i)}$:

$$\alpha^{(i)}(x^{(i)}, y_X) \approx H(f^{(i)}(x^{(i)})|y_X) - \frac{1}{|X_*^{(i)}|}\sum_{x_*^{(i)}\in X_*^{(i)}} H(f^{(i)}(x^{(i)})|y_X, x_*^{(i)}) \ . \tag{6.10}$$

6.2.2 Approximating the additive PES

In this subsection, we will discuss how the second entropy them in (6.10) is approximated. As has been mentioned in Section 5.3.2, for any $i = 1, .., C$, the conditional probability $p(f^{(i)}(x^{(i)})|y_X, x_*^{(i)})$ can be considered as a constrained version of $p(f^{(i)}(x^{(i)})|y_X)$ by further conditioning on the local maximizer $x_*^{(i)}$, where $p(f^{(i)}(x^{(i)})|y_X)$ has been shown to be Gaussian and can be computed analytically using (3.8). Intuitively, the posterior distribution of $f^{(i)}(x^{(i)})$ is constrained by

$$f^{(i)}(x^{(i)}) \leq f^{(i)}(x_*^{(i)}), \quad \forall x^{(i)} \in D^{(i)} \tag{6.11}$$

Note that (6.11) appears to resemble that in the original PES. To formally characterize such a constraint, Hernández-Lobato *et al.* (2014) has simplified their constraint as (a) only the function value of the next input to be selected by PES is smaller than $f^{(i)}(x_*^{(i)})$, and (b) $f^{(i)}(x_*^{(i)})$ is larger than the noisy outputs in the past observations, the latter of which, however, cannot be applied to our add-PES acquisition function because the noisy output y_x can only be observed for $f(x)$ instead of each local function $f^{(i)}(x^{(i)})$. To resolve this issue, we propose the following simplified constraints of (6.11) to approximate $p(f^{(i)}(x^{(i)})|y_X, x_*^{(i)})$:

$C1$. $f^{(i)}(x^{(i)}) \leq f^{(i)}(x_*^{(i)})$ for a given $x^{(i)} \in D^{(i)}$ for $i = 1, ..., C$.

$C2$. $\sum_{i=1}^{C} f^{(i)}(x_*^{(i)}) \geq y_{\max} + \epsilon$ where $y_{\max} \triangleq \max_{x \in X} y_x$ is the largest noisy output observed by evaluating $f(x)$ at X.

Note that $C2$ is defined jointly for $\{f^{(i)}(x_*^{(i)})\}_{i=1,...,C}$, which fortunately, will not be an issue for optimizing (6.4) independently for each $i = 1, ..., C$ since $C2$ is independent of $x^{(i)}$ and can be computed once jointly for $\{f^{(i)}(x_*^{(i)})\}_{i=1,...,C}$ and reused in the approximation of $C1$ for each $i = 1, ..., C$, as detailed later.

6.2.2.1 Approximating constraint $C2$ which is independent of $x^{(i)}$

Let $f_*^{(i)} \triangleq f^{(i)}(x_*^{(i)})$ and $f_* \triangleq (f_*^{(i)})_{i=1,...,C}^\top$. We can use the cdf of a standard Gaussian distribution to represent the probability of $C2$ and constrain the posterior joint distribution $p(f_*^{(1)}, ..., f_*^{(C)}|y_X)$ with $C2$ by

$$\begin{aligned} p(f_*^{(1)}, ..., f_*^{(C)}|y_X, C2) &\approx \frac{1}{Z} p(f_*^{(1)}, ..., f_*^{(C)}|y_X) \Phi_{\text{cdf}}(\frac{a^\top f_* - y_{\max}}{\sigma_n}) \\ &= \frac{1}{Z} \mathcal{N}(f_*|\mu, \Sigma) \Phi_{\text{cdf}}(\frac{a^\top f_* - y_{\max}}{\sigma_n}) \end{aligned} \tag{6.12}$$

where a is a C-dimensional vector with all the entries of value 1, μ and Σ can be computed analytically using (3.8). Then, the result of (6.12) can be approximated as

a multivariate Gaussian distribution $\mathcal{N}(f^*|\mu', \Sigma')$ using a one step EP. Let $m \triangleq a^\top \mu$, $v \triangleq a^\top \Sigma a$, $t \triangleq (m - y_{\max})/\sqrt{\sigma_n^2 + v}$ and $\gamma \triangleq \phi_{\text{pdf}}(t)/\Phi_{\text{cdf}}(t)$. Then,

$$\mu' = \mu + \frac{\gamma}{\sqrt{\sigma_n^2 + v}} \Sigma a \ , \quad \Sigma' = \Sigma - \frac{\gamma^2 + \gamma t}{\sigma_n^2 + v} \Sigma a a^\top \Sigma \tag{6.13}$$

The derivation of μ' and Σ' is in Appendix C.1. As a result,

$$p(f_*^{(i)}|y_X, C2) = \int p(f_*|y_X, C2) \mathrm{d}f_*^{(1)} ... \mathrm{d}f_*^{(i-1)} \mathrm{d}f_*^{(i+1)} \mathrm{d}f_*^{(C)} = \mathcal{N}(f_*^{(i)}|\mu_i', v_i')$$

where $\mu_i' \triangleq [\mu']_i$ and $v_i' \triangleq [\Sigma']_{ii}$.

6.2.2.2 Approximating constraint $C1$ which depends on $x^{(i)}$

Next, we approximate $C1$ using the similar steps to that in Section 5.3.2.2. In particular, let $f_+ \triangleq [f_*^{(i)}; f^{(i)}(x^{(i)})]$. Then,

$$p(f_+|y_X, C2) = p(f^{(i)}(x^{(i)})|y_X, f_*^{(i)}) p(f_*^{(i)}|y_X, C2) = \mathcal{N}(f_+|\mu_+, \Sigma_+)$$

where μ_+ and Σ_+ can be computed analytically using μ_i' and the same steps as in Appendix B.3. Let $a_1 \triangleq [-1; 1]$, $m' \triangleq a_1^\top \mu_+$, $v' \triangleq a_1^\top \Sigma_+ a_1$, $t' \triangleq -m'/\sqrt{v'}$ and $\gamma' \triangleq \phi_{\text{pdf}}(t)/\Phi_{\text{cdf}}(t)$. We can represent $C1$ using an indicator function and approximate $p(f^{(i)}(x^{(i)})|y_X, C1, C2)$ as follows:

$$\begin{aligned} p(f^{(i)}(x^{(i)})|y_X, C1, C2) &\approx \int \frac{1}{Z'} p(f_+|y_X, C2) \mathbb{I}(f^{(i)}(x^{(i)}) \leq f_*^{(i)}) \mathrm{d}f_*^{(i)} \\ &\approx \int \mathcal{N}(f_+|\mu_+', \Sigma_+') \mathrm{d}f_*^{(i)} = \mathcal{N}(f^{(i)}(x^{(i)})|\mu^{(i)}, v^{(i)}) \end{aligned}$$

where $v^{(i)} \triangleq [\Sigma_+']_{22}$ and

$$\Sigma_+' = \Sigma_+ - \frac{1}{v'} \gamma'(\gamma' + t') \Sigma_+ a_1 a_1^\top \Sigma_+ \ . \tag{6.14}$$

The derivation of Σ'_+ is same as that in Appendix B.4 with $c_i = 0$. Let $v^{(i)}(x_*^{(i)})$ denote $v^{(i)}$ computed using a sample $x_*^{(i)} \in X_*^{(i)}$ of the local target maximizer of $f^{(i)}(x^{(i)})$. Consequently, using (6.6), (6.10) and (6.14), the add-PES acquisition function can be approximated by

$$\alpha^{(i)}(x^{(i)}, y_X) \approx \frac{1}{2}\log(\Sigma^{(i)}_{x^{(i)}x^{(i)}|X}) - \frac{1}{2|X_*^{(i)}|}\sum_{x_*^{(i)}\in X_*^{(i)}} \log(v^{(i)}(x_*^{(i)})) \ . \tag{6.15}$$

The steps of add-PES is presented in Algorithm 3.

6.3 Experiments and Discussion

This section evaluates the high-dimensional BO performance of our add-PES algorithm against that of (a) PES [Hernández-Lobato *et al.*, 2014], (b) add-MES [Wang and Jegelka, 2017] which maximizes the information gain of the maximal value $f^{(i)}(x^{(i)})$ instead of the maximizer $x^{(i)}$ and (c) add-GP-UCB with β set according to that recommended in [Kandasamy *et al.*, 2015].

The synthetic functions with varying input dimensions are generated with $(d, d_i) \triangleq (10, 2), (24, 3), (32, 4), (50, 5)$ and $D \triangleq [0, 1]^d$. To do this, we used the following mean and covariance functions for the GP of each local function

$$\mu^{(i)}(x^{(i)}) \triangleq 0, \quad \sigma^{(i)}(x_p^{(i)}, x_q^{(i)}) \triangleq \sigma_s^2 \exp\left(-\frac{||x_p^{(i)} - x_q^{(i)}||^2}{2l^2}\right)$$

for $i = 1, ..., C$ and set $l^2 \triangleq 0.01$ when $d = 10$, $l^2 \triangleq 0.1$ when $d = 24, 32, 50$ and $\sigma_s^2 \triangleq 1$ for all synthetic functions. All above hyperparameters are also used in the tested algorithms to compute their acquisition functions. Then, a set X of 500 inputs is sampled randomly from D and their corresponding outputs are sampled from the GP prior with mean function $\mu(x) \triangleq 0$ and covariance function $\sigma(x_p, x_q) \triangleq \sum_{i=1}^{M} \sigma^{(i)}(x_p^{(i)}, x_q^{(i)})$.

The synthetic function is set to be the predictive mean function $\mu_{x|X}$ of this GP.

First, we show the difference between $p(x_*^{(i)}|y_X)$ and $p(f^{(i)}(x_*^{(i)})|y_X)$ by sampling $f^{(1)}(x_*^{(1)})$ of the synthetic function with $d = 10$, which is used to demonstrate the advantage of add-PES compared to add-MES as mentioned in Section 6.2. An example of 3 samples achieved using add-RF given 50 observations is shown in Figs. 6.1(a)-(c). As can be seen, even though the locations of the maximizer vary a lot (i.e., large uncertainty in $x_*^{(1)}$) in these 3 samples, the maximal values (i.e., 2.1622, 2.1555 and 2.1706) of the sampled functions are very similar to each other (i.e., small uncertainty in $f^{(1)}(x_*^{(1)})$). In such case, maximizing the information gain of $x_*^{(1)}$ using add-PES is expected to achieve a lot more information than maximizing the information gain of $f^{(1)}(x_*^{(1)})$ using add-MES. To further demonstrate this, we present the acquisition functions $\alpha^{(1)}(x^{(1)}, y_X)$ of add-MES and add-PES computed using above three samples in Figs. 6.1(e) and 6.1(f), respectively. The ground true of local function $f^{(1)}(x^{(1)})$ and its observations are also shown in Fig. 6.1(d). As can be seen, although the local maximum area highlighted using a black box has been well exploited, add-MES still gives large acquisition function values for this area, which makes it to trap in this local maximum for longer time. In contrast, add-PES gives small acquisition function values in this well-exploited area, and thus, shows better exploratory behavior than add-MES as we have expected.

Next, we compare the performance of all tested algorithms by optimizing each synthetic function with a budget of 500 observations. As has been shown in Wang and Jegelka (2017) and Hernández-Lobato *et al.* (2017), a small number of samples could help to improve the exploratory behavior of the algorithm, which is important to globally optimize a function with high input dimensions (i.e., very larger search space). To demonstrate this, we used 1 (i.e., add-PES-1 and add-MES-1) and 50 (i.e., add-PES-50 and add-MES-50) samples to approximate the acquisition functions of both add-PES and add-MES. The performance of the tested algorithms are evaluated

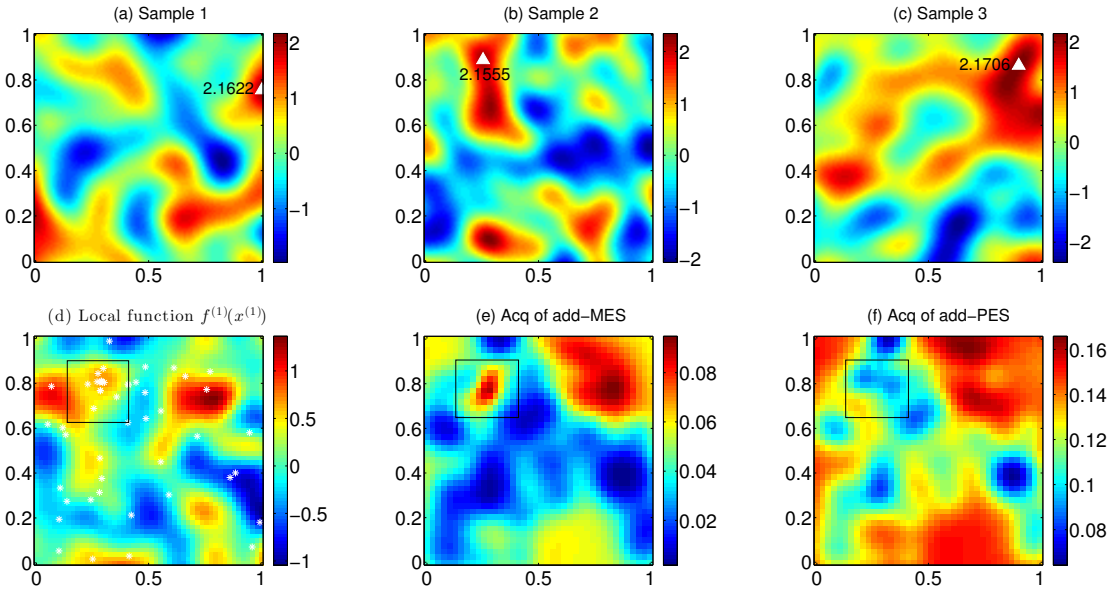


Figure 6.1: Surface plots of (a-c) samples of the first local function $f^{(1)}(x^1)$ $(d_1 = 2)$ of the synthetic function with $d = 10$ where '$\triangle$' is the sample of local maximizer $x_*^{(i)}$ and the number next to '$\triangle$' is the sampled value of $f^{(i)}(x_*^{(i)})$, (d) the true local function $f^{(1)}(x^1)$ where '$+$' is the local input $x^{(1)}$ of existing observations and (e)-(f) the acquisition function $\alpha^{(1)}(x^{(1)}, y_X)$ of add-MES and add-PES, respectively.

using *immediate regret* (IR) $|f(x_*) - f(\tilde{x}_*)|$ where $\tilde{x}_* \triangleq \oplus_{i=1}^{C} \arg\max_{x^{(i)} \in D^{(i)}} \mu^{(i)}_{x^{(i)}|X}$ is the recommended maximizer of the additive methods and $\tilde{x}_* \triangleq \arg\max_{x \in D} \mu_{x|X}$ is the recommended maximizer of PES. In each experiment, 10 observations are randomly selected as the initialization. An averaged IR is obtained by optimizing each synthetic function with 10 different initializations for each tested algorithm.

As shown in Fig. 6.2, add-PES can achieve a much lower averaged IR than PES, which means that BO performance with high input dimensions can be improved considerably by exploiting an additive structure of the function. For all tested functions, add-GP-UCB and add-MES are shown to be trapped very easily in some local maximum such that they cannot achieve a small averaged IR as that of add-PES-1. Compared to add-PES-50, add-PES-1 can always perform better by using only

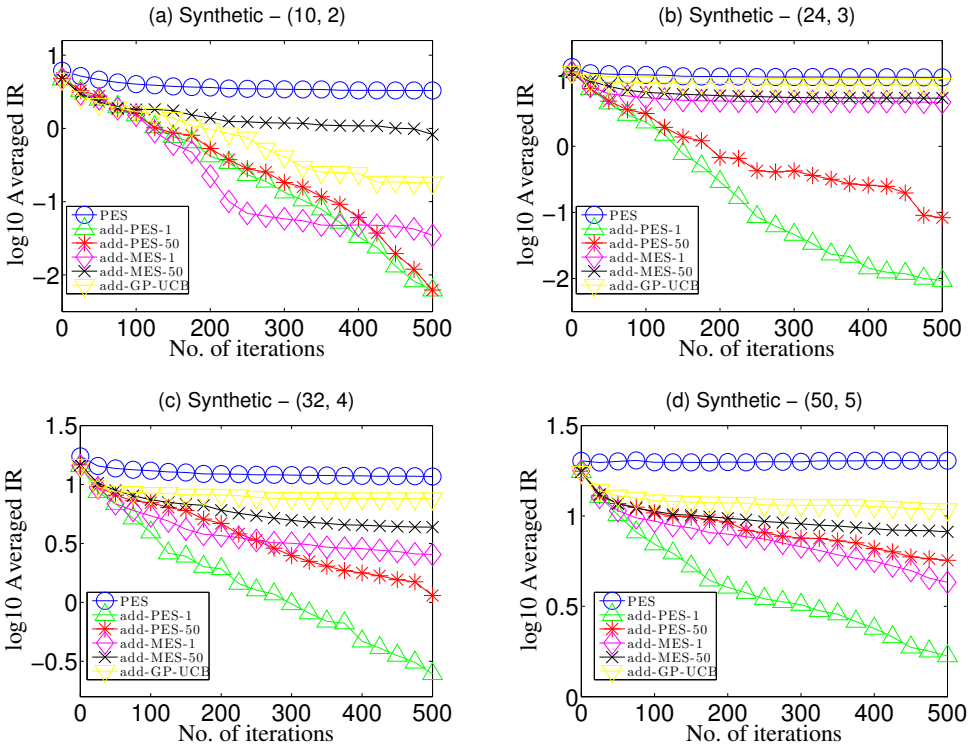


Figure 6.2: Graphs of $\log_{10}$(averaged IR) vs. no. of iterations by tested algorithms for synthetic functions with varying input dimensions. The input dimension d and d_i of x and $x^{(i)}$, respectively, are shown in the title of each graph as (d, d_i).

a single sample. Furthermore, when the number of input dimensions is increased from Fig. 6.2(a) to 6.2(d), the performance advantage arising from strong exploration behavior through the use of a single sample of x_* becomes obvious. This clear performance advantage of simply choosing a single sample eliminates the need to carefully set the number of samples to make add-PES perform well.

6.4 Summary

This chapter describes an add-PES algorithm for high-dimensional BO which is a generalization of the PES by assuming an additive structure of the unknown func-

tion. Novel constraint is proposed and approximated efficiently for achieving good scalability of add-PES in the number of input dimensions. The empirical evaluation on synthetic functions with varying input dimensions shows that add-PES outperforms the state-of-the-art high-dimensional BO algorithms and is very easy to decide its parameter for achieving a good exploration-exploitation trade-off, which makes it promising to perform well in different real-world applications.

However, the additive structure of the synthetic function is assumed to be known in all above experiments, which is not true in the real-world applications. To learn the additive structure when it is unknown, *maximum likelihood estimation* can be used to select the best decomposition among a set of randomly selected ones [Kandasamy *et al.*, 2015]. More synthetic/real-world functions with unknown additive structure will be used to test the performance of our algorithm in the future work.

Algorithm 3 Additive PES (add-PES) for High-Dimensional BO

Input: A budget B; Input domain D; A set of random initializations $(X_{\text{init}}, y_{X_{\text{init}}})$

$X \leftarrow X_{\text{init}}$;
while $B \geq 0$ **do**
  Sample a set of $X_*^{(i)}$ for $i = 1, ..., C$;
  for $\{x_*^{(i)}\}_{i=1,...,C}$ in $\{X_*^{(i)}\}_{i=1,...,C}$ **do**
    Compute μ' and Σ' using (6.13); ▷ Approximate constraint $C2$.
  end for
  $x^+ \leftarrow []$; ▷ Select the next input to evaluate $f(x)$.
  for $i = 1$ to C **do**
    $\alpha^{(i)}(\cdot\ , y_X) \leftarrow \text{ADDPES}(X, y_X, X_*^{(i)}, \mu', \Sigma')$
    $x^{(i)} \leftarrow \arg\max_{x^{(i)} \in D^{(i)}} \alpha^{(i)}(x^{(i)}, y_X)$
    $x^+ \leftarrow x^+ \oplus x^{(i)}$; ▷ Connect the selected input component.
  end for
  evaluate function $f(x)$ at the selected input x^+ to get the observation y_{x^+};
  $X \leftarrow X \cup \{x^+\}$;
  $B \leftarrow B - \text{cost}(x)$;
end while
$\tilde{x}_* \leftarrow []$; ▷ Compute the maximizer of predictive mean for each local function.
for $i = 1$ to C **do**
  $\tilde{x}_*^{(i)} \leftarrow \arg\max_{x^{(i)} \in D^{(i)}} \mu^{(i)}(x^{(i)}|y_X)$; ▷ $\mu^{(i)}(\cdot\ |y_X)$ is constructed using (3.8).
  $\tilde{x}_* \leftarrow \tilde{x}_* \oplus \tilde{x}_*^{(i)}$;
end for
return $\tilde{x}_*$;

procedure ADDPES(X, y_X, , $X_*^{(i)}, \mu', \Sigma'$) ▷ Approximate constraint $C1$.
  for each $x_*^{(i)}$ in $X_*^{(i)}$ **do**
    $\mu'_i \leftarrow [\mu']_i$, $v'_i \leftarrow [\Sigma']_{ii}$;
    Compute μ_+ and Σ_+ using Appendix B.3;
    Compute Σ'_+ using (6.14);
    $v^{(i)}(x_*^{(i)}) \leftarrow [\Sigma'_+]_{22}$
  end for
  return $\alpha^{(i)}(\cdot\ , y_X)$ computed using (6.15)
end procedure

Chapter 7

Conclusion and Future Work

7.1 Conclusion

This thesis generalizes two data-efficient ML approaches (i.e., AL and BO) to multiple output types and high input dimensions. By exploiting the structure of some form of GP-based probabilistic regression models, all the proposed algorithms in this thesis have achieved better performance in selecting and gathering the most information observations for learning the target variable(s) of interest more accurately and efficiently given some budget constraints. The specific contributions for each work are listed below:

1. Active learning of MOGP [Zhang *et al.*, 2016]

- *Novel active learning criterion for MOGP model.* To resolve the scalability issue in optimizing the conventional entropy criterion, we exploit a structure common to a unifying framework of sparse MOGP models for deriving a novel active learning criterion.

- *Approximation algorithm with performance guarantee.* To approximately op-

timize the new criterion in a polynomial time, the ϵ-submodularity property of our new criterion is exploited for devising a polynomial time approximation algorithm that guarantees a constant factor approximation of that achieved by the optimal set of selected observations.

- *Empirical evaluation.* Via evaluating the performance of our proposed algorithm using three real-world datasets, we empirically show that our approximation algorithm m-Greedy outperforms existing algorithms for active learning of MOGP and single-output GP models, especially when measurements of the primary output types are more noisy than that of the auxiliary types.

2. PES for multi-fidelity BO

- *Generalizing PES to multi-fidelity BO.* To exploit the less noisy and/or cheaper auxiliary function(s) of varying fidelities for accelerating the optimization of the target function, we generalize the PES to multi-fidelity BO by modeling the unknown target and auxiliary functions jointly as a CMOGP whose covariance structure, interestingly, is used to formalize the fidelity of each auxiliary function. More importantly, the proposed MF-PES algorithm can naturally trade off between exploration vs. exploitation of the target and auxiliary functions without needing to manually tune any such parameters, which makes it a superior alternative among the limited selection of multi-fidelity BO algorithms

- *Approximation of MF-PES.* Since the proposed MF-PES acquisition function is analytically intractable, we derive an efficient approximation of MF-PES via a novel MRF approximation of the CMOGP model. In particular, MRF is first used to improve the belief of the target maximizer, and then, is exploited to approximate our newly proposed practical constraints for relating the global target maximizer to that of auxiliary functions.

- *Empirical evaluation.* We empirically evaluate and verify the superior performance of our MF-PES algorithm over that of the state-of-the-art multi-fidelity BO algorithms in both synthetic experiment and real-world hyperparameters tuning applications.

3. PES for high-dimensional BO

- *Additive PES (add-PES) for high-dimensional BO.* To scale up the state-of-the-art BO algorithm to high input dimensions, we proposed an additive form of PES (i.e., add-PES) which selects each local and low-dimensional input component independently for achieving the next input to evaluate the function. Interestingly, although the practical constraints for approximating add-PES can only be defined jointly over all input components, we show that it is still possible to optimize an add-PES over each local input component independently using some new EP steps.

- *Empirical evaluation.* We empirically demonstrate that our add-PES algorithm achieves much better BO performance than the state-of-the-art high-dimensional BO algorithms by simply using a single sample of the target maximizer for synthetic functions with varying input dimensions. Such results show that our add-PES is much easier to decide its parameter (e.g., one sample) for achieving appropriate exploration-exploitation trade-off, which makes it promising to perform well in different real-world applications.

7.2 Future Work

There are a few directions that can be pursued as continuation to the works in this thesis.

Firstly, in our works about data-efficient ML for multiple output types, we have assumed that the hyperparameters (i.e., σ_{s_i}, σ_{n_i}, P_0 and P_i for $i = 1, ..., M$ in (3.2)) of CMOGP are point-based values and learned via maximum likelihood estimation [Álvarez and Lawrence, 2011], which might not be efficient enough in practice when the observations are too sparse to find the correct hyperparameters or too dense such that updating CMOGP hyperparameters with a new observation requires long training time. To resolve above issues, we would like to consider using probabilistic hyperparameters instead of the point-based ones and marginalizing them for computing the *integrated acquisition function* [Snoek *et al.*, 2012] for multi-fidelity BO, which can make the algorithm more robust to the sparse and/or noisy observations, and thus, allows the CMOGP hyperparameters to be updated more accurately on the fly of the AL/BO process. Furthermore, it's also worth to investigate whether the CMOGP hyperparameters can be updated incrementally at every iteration of the proposed algorithms.

Secondly, another direction of multi-fidelity BO is to explore how to automatically select the fidelity of the auxiliary function for trading off between the accuracy in reproducing the target function and the cost. Using the hyperparameters tuning application (Section 5.4) as an example: The auxiliary function in this problem is constructed by training the ML model with a small subset of training data which has a fixed size and is randomly selected . Given a time budget constraint, how to select this subset of data for constructing auxiliary function(s) with specified fidelities which able to achieve good BO performance has not been studied. To address this issue, we can consider fusing our AL algorithm with multi-fidelity BO such that the auxiliary function(s) can be more accurately constructed and the BO performance can be improved as a consequence.

Thirdly, as has been mentioned at the end of Section 6.3, we will consider applying our add-PES algorithm to some real-world applications such as the configuration set-

ting of mobility-on-demand systems [Chen *et al.*, 2013] and swarm robotics [Brambilla *et al.*, 2013]. In such applications, the parameters to be tuned are usually in super high dimensions where a good high-dimensional BO algorithm is required. However, for constructing an accurate additive model in this case, the optimal decomposition of the high input dimensions need to be learned from the observations or some specific features of the application, which is very difficult in practice due to the sparsity of observations collected using BO, and thus, is a non-trivial and interesting issue to be exploited in the future.

Finally, although both MF-PES and add-PES are generalizations of the PES algorithm, they cannot be applied together straightforwardly since the additive structure assumption used in add-PES has not been generalized to any multi-output GP model. To resolve this issue, we can consider assuming an additive structure for either the latent function $L(x)$ or the original target and auxiliary functions $f_i(x)$ in CMOGP. Then, a combination of the approximation steps in MF-PES (Chapter 5) and add-PES (Chapter 6) will be considered for developing an algorithm of multi-fidelity BO with high-dimensional input, which is a more general BO algorithm and may have interesting real-world applications.

Bibliography

[Álvarez and Lawrence, 2009] Mauricio A. Álvarez and Neil D Lawrence. Sparse convolved Gaussian processes for multi-output regression. In *Proc. NIPS*, pages 57–64, 2009.

[Álvarez and Lawrence, 2011] M. A. Álvarez and N. D. Lawrence. Computationally efficient convolved multiple output Gaussian processes. *JMLR*, 12:1459–1500, 2011.

[Álvarez *et al.*, 2012] Mauricio A. Álvarez, Lorenzo Rosasco, and Neil D. Lawrence. Kernels for vector-valued functions: A review. *Found. Trends Mach. Learn.*, 4(3), 2012.

[Angulo and Bueso, 2001] J. M. Angulo and M. C. Bueso. Random perturbation methods applied to multivariate spatial sampling design. *Environmetrics*, 12(7):631–646, 2001.

[Apple *et al.*, 2008] J. K. Apple, E. M. Smith, and T. J. Boyd. Temperature, salinity, nutrients, and the covariation of bacterial production and chlorophyll-a in estuarine ecosystems. *J. Coastal Res.*, 25(sp1):59–75, 2008.

[Atkinson *et al.*, 2000] P. M. Atkinson, G. M. Foody, P. J. Curran, and D. S. Boyd. Assessing the ground data requirements for regional scale remote sensing of tropical forest biophysical properties. *International Journal of Remote Sensing*, 21(13-14):2571–2587, 2000.

[Balasubramanian and Lebanon, 2012] Krishnakumar Balasubramanian and Guy Lebanon. The landmark selection method for multiple output prediction. In *Proc. ICML*, pages 983–990, New York, NY, USA, 2012.

[Bardenet *et al.*, 2013] Rémi Bardenet, Mátyás Brendel, Balázs Kégl, and Michele Sebag. Collaborative hyperparameter tuning. In *Proc. ICML*, pages 199–207, 2013.

[Bergstra *et al.*, 2013] James Bergstra, Daniel Yamins, and David Cox. Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In *Proc. ICML*, pages 115–123, 2013.

[Birlutiu *et al.*, 2010] Adriana Birlutiu, Perry Groot, and Tom Heskes. Multi-task preference learning with an application to hearing aid personalization. *Neurocomputing*, 73(7):1177–1185, 2010.

[Bo and Sminchisescu, 2010] Liefeng Bo and Cristian Sminchisescu. TwinGaussian processes for structured prediction. *International Journal of Computer Vision*, 87(1-2):28–52, 2010.

[Bodik *et al.*, 2004] P. Bodik, C. Guestrin, W. Hong, S. Madden, M. Paskin, and R. Thibaux. http://www.select.cs.cmu.edu/data/labapp3/index.html, 2004.

[Bonilla *et al.*, 2007a] Edwin V Bonilla, Felix V Agakov, and Christopher Williams. Kernel multi-task learning using task-specific features. In *Proc. AISTATS*, pages 43–50, 2007.

[Bonilla *et al.*, 2007b] Edwin V Bonilla, Kian Ming Adam Chai, and Christopher KI Williams. Multi-task Gaussian process prediction. In *Proc. NIPS*, pages 153–160, 2007.

[Boyle and Frean, 2004] Phillip Boyle and Marcus Frean. Dependent Gaussian processes. In *Proc. NIPS*, pages 217–224, 2004.

[Brambilla *et al.*, 2013] Manuele Brambilla, Eliseo Ferrante, Mauro Birattari, and Marco Dorigo. Swarm robotics: a review from the swarm engineering perspective. *Swarm Intelligence*, 7(1):1–41, 2013.

[Brochu *et al.*, 2010] 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. Technical report, 2010.

[Brooks *et al.*, 2008] Alex Brooks, Alexei Makarenko, and Ben Upcroft. Gaussian process models for indoor and outdoor sensor-centric robot localization. *Robotics, IEEE Transactions on*, 24(6):1341–1351, Dec 2008.

[Bueso *et al.*, 1998] M. C. Bueso, J. M. Angulo, and F. J. Alonso. A state-space model approach to optimum spatial sampling design based on entropy. *Environmental and Ecological Statistics*, 5(1):29–44, 1998.

[Bueso *et al.*, 1999] M. C. Bueso, J. M. Angulo, J. Cruz-Sanjulián, and J. L. García-Aróstegui. Optimal spatial sampling design in a multivariate framework. *Math. Geology*, 31(5):507–525, 1999.

[Calandra, 2017] Roberto Calandra. *Bayesian Modeling for Optimization and Control in Robotics*. PhD thesis, Technische Universität Darmstadt, 2017.

[Cao *et al.*, 2013] N. Cao, K. H. Low, and J. M. Dolan. Multi-robot informative path planning for active sensing of environmental phenomena: A tale of two algorithms. In *Proc. AAMAS*, pages 7–14, 2013.

[Chapelle and Li, 2011] Olivier Chapelle and Lihong Li. An empirical evaluation of Thompson sampling. In *Proc. NIPS*, pages 2249–2257, 2011.

[Chen *et al.*, 2012a] Bo Chen, Rui Castro, and Andreas Krause. Joint optimization and variable selection of high-dimensional Gaussian processes. In *Proc. ICML*, 2012.

[Chen *et al.*, 2012b] J. Chen, K. H. Low, C. K.-Y. Tan, A. Oran, P. Jaillet, J. M. Dolan, and G. S. Sukhatme. Decentralized data fusion and active sensing with mobile sensors for modeling and predicting spatiotemporal traffic phenomena. In *Proc. UAI*, pages 163–173, 2012.

[Chen *et al.*, 2013] J. Chen, K. H. Low, and C. K.-Y. Tan. Gaussian process-based decentralized data fusion and active sensing for mobility-on-demand system. In *Proc. RSS*, 2013.

[Cover and Thomas, 1991] T. Cover and J. Thomas. *Elements of Information Theory*. John Wiley & Sons, Inc., 1991.

[Cutler *et al.*, 2015] M. Cutler, T. J. Walsh, and J. P. How. Real-world reinforcement learning via multi-fidelity simulators. *IEEE Transactions on Robotics*, 31(3):655–671, 2015.

[Das and Kempe, 2008] A. Das and D. Kempe. Algorithms for subset selection in linear regression. In *Proc. STOC*, pages 45–54, 2008.

[Djolonga *et al.*, 2013] Josip Djolonga, Andreas Krause, and Volkan Cevher. High-dimensional Gaussian process bandits. In *Proc. NIPS*, pages 1025–1033, 2013.

[Duvenaud *et al.*, 2011] David K Duvenaud, Hannes Nickisch, and Carl E Rasmussen. Additive Gaussian processes. In *Proc. NIPS*, pages 226–234, 2011.

[Evgeniou and Pontil, 2004] T. Evgeniou and M. Pontil. Regularized multi-task learning. In *Proc. SIGKDD*, pages 109–117, 2004.

[Fei-Fei *et al.*, 2006] Li Fei-Fei, Rob Fergus, and Pietro Perona. One-shot learning of object categories. *IEEE transactions on pattern analysis and machine intelligence*, 28(4):594–611, 2006.

[Feurer *et al.*, 2015] Matthias Feurer, Jost Tobias Springenberg, and Frank Hutter. Initializing Bayesian hyperparameter optimization via meta-learning. In *Proc. AAAI*, pages 1128–1135, 2015.

[Forrester *et al.*, 2007] A. I. J. Forrester, A. Sóbester, and A. J. Keane. Multi-fidelity optimization via surrogate modelling. *Proc. R. Soc. A*, 463(2088):3251–3269, 2007.

[Ghahramani, 2012] Zoubin Ghahramani. Probabilistic modelling, machine learning, and the information revolution. *Presentation at MIT CSAIL*, 2012.

[Golub and Van Loan, 1996] G. H. Golub and C.-F. Van Loan. *Matrix Computations*. Johns Hopkins Univ. Press, 3rd edition, 1996.

[González *et al.*, 2014] Javier González, Joseph Longworth, David C James, and Neil D Lawrence. Bayesian optimization for synthetic gene design. In *NIPS Workshop on Bayesian Optimization in Academia and Industry*, 2014.

[Goovaerts, 1997] P. Goovaerts. *Geostatistics for Natural Resources Evaluation*. Oxford Univ. Press, 1997.

[Han *et al.*, 2017] Duo Han, Junfeng Wu, Huanshui Zhang, and Ling Shi. Optimal sensor scheduling for multiple linear dynamical systems. *Automatica*, 75:260–270, 2017.

[Hennig and Schuler, 2012] Philipp Hennig and Christian J Schuler. Entropy search for information-efficient global optimization. *JMLR*, 13(Jun):1809–1837, 2012.

[Hernández-Lobato *et al.*, 2014] José Miguel Hernández-Lobato, Matthew W Hoffman, and Zoubin Ghahramani. Predictive entropy search for efficient global optimization of black-box functions. In *Proc. NIPS*, pages 918–926, 2014.

[Hernández-Lobato *et al.*, 2016] Daniel Hernández-Lobato, José Miguel Hernández-Lobato, Amar Shah, and Ryan P Adams. Predictive entropy search for multi-objective Bayesian optimization. 2016.

[Hernández-Lobato *et al.*, 2017] José Miguel Hernández-Lobato, James Requeima, Edward O Pyzer-Knapp, and Alán Aspuru-Guzik. Parallel and distributed thompson sampling for large-scale accelerated exploration of chemical space, 2017.

[Hero and Cochran, 2011] Alfred O Hero and Douglas Cochran. Sensor management: Past, present, and future. *IEEE Sensors Journal*, 11(12):3064–3075, 2011.

[Higdon *et al.*, 2008] Dave Higdon, James Gattiker, Brian Williams, and Maria Rightley. Computer model calibration using high-dimensional output. *Journal of the American Statistical Association*, 103(482), 2008.

[Higdon, 2002] Dave Higdon. Space and space-time modeling using process convolutions. In *Quantitative methods for current environmental issues*, pages 37–56. Springer, 2002.

[Huang *et al.*, 2006] D. Huang, T. T. Allen, W. I. Notz, and R. A. Miller. Sequential kriging optimization using multiple-fidelity evaluations. *Struct. Multidisc. Optim.*, 32(5):369–382, 2006.

[ICMLws, 2015] ICMLws. https://sites.google.com/site/automlwsicml15/, 2015.

[ICMLws, 2016] ICMLws. https://sites.google.com/site/dataefficientml/, 2016.

[Izenman, 1975] Alan Julian Izenman. Reduced-rank regression for the multivariate linear model. *J. Multivariate Analysis*, 5(2):248–264, 1975.

[Kandasamy *et al.*, 2015] Kirthevasan Kandasamy, Jeff Schneider, and Barnabás Póczos. High dimensional Bayesian optimisation and bandits via additive models. In *Proc. ICML*, pages 295–304, 2015.

[Kandasamy *et al.*, 2016] K. Kandasamy, G. Dasarathy, J. B. Oliva, J. Schneider, and B. Póczos. Gaussian process bandit optimisation with multi-fidelity evaluations. In *Proc. NIPS*, pages 992–1000, 2016.

[Kandasamy *et al.*, 2017] Kirthevasan Kandasamy, Gautam Dasarathy, Jeff Schneider, and Barnabas Poczos. Multi-fidelity Bayesian optimisation with continuous approximations. In *Proc. ICML*, pages 1799–1808, 2017.

[Klein *et al.*, 2017] Aaron Klein, Stefan Falkner, Simon Bartels, Philipp Hennig, and Frank Hutter. Fast Bayesian optimization of machine learning hyperparameters on large datasets. In *Proc. AISTATS*, 2017.

[Krause and Golovin, 2014] A. Krause and D. Golovin. Submodular function maximization. In L. Bordeaux, Y. Hamadi, and P. Kohli, editors, *Tractability: Practical Approaches to Hard Problems*, pages 71–104. Cambridge Univ. Press, 2014.

[Krause and Guestrin, 2005] A. Krause and C. Guestrin. Near-optimal nonmyopic value of information in graphical models. In *Proc. UAI*, 2005.

[Krause and Guestrin, 2007] Andreas Krause and Carlos Guestrin. Nonmyopic active learning of Gaussian processes: An exploration-exploitation approach. In *Proc. ICML*, pages 449–456, 2007.

[Krause *et al.*, 2008] A. Krause, A. Singh, and C. Guestrin. Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies. *JMLR*, 9:235–284, 2008.

[Lawrence and Platt, 2004] Neil D Lawrence and John C Platt. Learning to learn with the informative vector machine. In *Proc. ICML*, page 65, 2004.

[Lawrence *et al.*, 2003] Neil Lawrence, Matthias Seeger, and Ralf Herbrich. Fast sparse Gaussian process methods: The informative vector machine. In *Proc. NIPS*, pages 625–632, 2003.

[Lázaro-Gredilla *et al.*, 2010] M. Lázaro-Gredilla, J. Quiñonero-Candela, C. E. Rasmussen, and A. R. Figueiras-Vidal. Sparse spectrum Gaussian process regression. *JMLR*, 11:1865–1881, 2010.

[Le *et al.*, 2003] N. D. Le, L. Sun, and J. V. Zidek. Designing networks for monitoring multivariate environmental fields using data with monotone pattern. Technical Report #2003-5, Statistical and Applied Mathematical Sciences Institute, Research Triangle Park, NC, 2003.

[Lizotte *et al.*, 2007] D. Lizotte, T. Wang, M. Bowling, and D. Schuurmans. Automatic gait optimization with Gaussian process regression. In *Proc. IJCAI*, pages 944–949, 2007.

[Marco *et al.*, 2017] Alonso Marco, Felix Berkenkamp, Philipp Hennig, Angela P Schoellig, Andreas Krause, Stefan Schaal, and Sebastian Trimpe. Virtual vs. real: Trading off simulations and physical experiments in reinforcement learning with Bayesian optimization. In *Proc. ICRA*, pages 1557–1563, 2017.

[Minka, 2001] T. P. Minka. *A family of algorithms for approximate Bayesian inference*. PhD thesis, Massachusetts Institute of Technology, 2001.

[Mourikis and Roumeliotis, 2006] Anastasios I Mourikis and Stergios I Roumeliotis. Optimal sensor scheduling for resource-constrained localization of mobile robot formations. *IEEE Transactions on Robotics*, 22(5):917–931, 2006.

[Nemhauser *et al.*, 1978] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functions - I. *Mathematical Programming*, 14(1):265–294, 1978.

[Obozinski *et al.*, 2008] Guillaume R Obozinski, Martin J Wainwright, and Michael I Jordan. High-dimensional support union recovery in multivariate regression. In *Proc. NIPS*, pages 1217–1224, 2008.

[Osborne *et al.*, 2008] Michael A Osborne, Stephen J Roberts, Alex Rogers, Sarvapali D Ramchurn, and Nicholas R Jennings. Towards real-time information processing of sensor network data using computationally efficient multi-output Gaussian processes. In *Proc. IPSN*, 2008.

[Pan and Yang, 2010] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2010.

[Passarella *et al.*, 2003] G. Passarella, M. Vurro, V. D'Agostino, and M. J. Barcelona. Cokriging optimization of monitoring network configuration based on fuzzy and non-fuzzy variogram evaluation. *Environmental Monitoring and Assessment*, 82:1–21, 2003.

[Petersen and Pedersen, 2012] K. B. Petersen and M. S. Pedersen. *The Matrix Cookbook*. 2012.

[Poloczek *et al.*, 2016] Matthias Poloczek, Jialei Wang, and Peter I Frazier. Multi-information source optimization. *arXiv preprint arXiv:1603.00389*, 2016.

[Rahimi and Recht, 2007] Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *Proc. NIPS*, pages 1177–1184, 2007.

[Rana *et al.*, 2017] Santu Rana, Cheng Li, Sunil Gupta, Vu Nguyen, and Svetha Venkatesh. High dimensional Bayesian optimization with elastic gaussian process. In *Proc. ICML*, pages 2883–2891, 2017.

[Rasmussen and Williams, 2006] C. E. Rasmussen and C. K. Williams. *Gaussian processes for machine learning*. MIT Press, 2006.

[Reichart *et al.*, 2008] R. Reichart, K. Tomanek, U. Hahn, and A. Rappoport. Multi-task active learning for linguistic annotations. In *Proc. ACL*, pages 861–869, 2008.

[Reinsel and Velu, 1998] G. C. Reinsel and R. P. Velu. *Multivariate Reduced-Rank Regression: Theory and Applications*. Springer, 1998.

[Renner and Maurer, 2002] Renato Renner and Ueli Maurer. About the mutual (conditional) information. In *Proc. IEEE ISIT*, 2002.

[Rohde and Tsybakov, 2011] Angelika Rohde and Alexandre B Tsybakov. Estimation of high-dimensional low-rank matrices. *The Annals of Statistics*, 39(2):887–930, 2011.

[Roth and Small, 2006] Dan Roth and Kevin Small. Margin-based active learning for structured output spaces. In *Proc. ECML*, pages 413–424, 2006.

[Rudovic and Pantic, 2011] Ognjen Rudovic and Maja Pantic. Shape-constrained Gaussian process regression for facial-point-based head-pose normalization. In *Proceedings of International Conference on Computer Vision (ICCV-11)*, pages 1495–1502. IEEE, 2011.

[Sánchez-Fernández *et al.*, 2004] M. Sánchez-Fernández, M. de-Prado-Cumplido, J. Arenas-García, and F. Pérez-Cruz. SVM multiregression for nonlinear channel estimation in multiple-input multiple-output systems. *IEEE Trans. Signal Processing*, 52(8):2298–2307, 2004.

[Schön and Lindsten, 2011] Thomas B Schön and Fredrik Lindsten. Manipulating the multivariate Gaussian density. Technical report, Linkoeping University, 2011.

[Schwaighofer *et al.*, 2004] Anton Schwaighofer, Volker Tresp, and Kai Yu. Learning Gaussian process kernels via hierarchical bayes. In *Proc. NIPS*, pages 1209–1216, 2004.

[Settles, 2010] Burr Settles. Active learning literature survey. *University of Wisconsin, Madison*, 52(55-66):11, 2010.

[Shahriari *et al.*, 2016] Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P Adams, and Nando de Freitas. Taking the human out of the loop: A review of Bayesian optimization. *Proc. of the IEEE*, 104(1):148–175, 2016.

[Shewry and Wynn, 1987] M. C. Shewry and H. P. Wynn. Maximum entropy sampling. *J. Applied Stat.*, 14(2):165–170, 1987.

[Shi *et al.*, 2008] Xiaoxiao Shi, Wei Fan, and Jiangtao Ren. Actively transfer domain knowledge. In *Proc. ECML*, pages 342–357, 2008.

[Skolidis, 2012] G. Skolidis. *Transfer Learning with Gaussian Processes*. PhD thesis, University of Edinburgh, 2012.

[Snoek *et al.*, 2012] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical Bayesian optimization of machine learning algorithms. In *Proc. NIPS*, pages 2951–2959, 2012.

[Srinivas *et al.*, 2010] Niranjan Srinivas, Andreas Krause, Sham Kakade, and Matthias Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. In *Proc. ICML*, pages 1015–1022, 2010.

[Stewart and Sun, 1990] G. W. Stewart and J.-G. Sun. *Matrix Perturbation Theory*. Academic Press, 1990.

[Swersky *et al.*, 2013] Kevin Swersky, Jasper Snoek, and Ryan P Adams. Multi-task Bayesian optimization. In *Proc. NIPS*, pages 2004–2012, 2013.

[Teh and Seeger, 2005] Y. W. Teh and M. Seeger. Semiparametric latent factor models. In *Proc. AISTATS*, pages 333–340, 2005.

[Tesch *et al.*, 2013] M. Tesch, J. Schneider, and H. Choset. Expensive function optimization with stochastic binary outcomes. In *Proc. ICML*, pages 1283–1291, 2013.

[Tuia *et al.*, 2011] D. Tuia, J. Verrelst, L. Alonso, F. Perez-Cruz, and G. Camps-Valls. Multioutput support vector regression for remote sensing biophysical parameter estimation. *Geoscience and Remote Sensing Letters, IEEE*, 8(4):804–808, 2011.

[Tzoumas *et al.*, 2016] Vasileios Tzoumas, Ali Jadbabaie, and George J Pappas. Near-optimal sensor scheduling for batch state estimation: Complexity, algorithms, and limits. In *Decision and Control (CDC), 2016 IEEE 55th Conference on*, pages 2695–2702. IEEE, 2016.

[Villemonteix *et al.*, 2009] J. Villemonteix, E. Vazquez, and E. Walter. An informational approach to the global optimization of expensive-to-evaluate functions. *J. Glob. Optim.*, 44(4):509–534, 2009.

[Wang and Hua, 2011] Meng Wang and Xian-Sheng Hua. Active learning in multimedia annotation and retrieval: A survey. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(2):10, 2011.

[Wang and Jegelka, 2017] Zi Wang and Stefanie Jegelka. Max-value entropy search for efficient Bayesian optimization. In *Proc. ICML*, pages 3627–3635, 2017.

[Wang *et al.*, 2013] Ziyu Wang, Masrour Zoghi, Frank Hutter, David Matheson, Nando De Freitas, et al. Bayesian optimization in high dimensions via random embeddings. In *Proc. IJCAI*, pages 1778–1784, 2013.

[Wang *et al.*, 2014] Xuezhi Wang, Tzu-Kuo Huang, and Jeff Schneider. Active transfer learning under model shift. In *Proc. ICML*, 2014.

[Wang *et al.*, 2017] Zi Wang, Chengtao Li, Stefanie Jegelka, and Pushmeet Kohli. Batched high-dimensional Bayesian optimization via structural kernel learning. In *Proc. ICML*, pages 3656–3664, 2017.

[Webster and Oliver, 2007] R. Webster and M. Oliver. *Geostatistics for Environmental Scientists*. John Wiley & Sons, Inc., 2nd edition, 2007.

[Webster, 1977] R. Webster. Spectral analysis of Gilgai soil. *Australian Journal of Soil Research*, 15(3):191–204, 1977.

[Weston *et al.*, 2002] Jason Weston, Olivier Chapelle, Vladimir Vapnik, André Elisseeff, and Bernhard Schölkopf. Kernel dependency estimation. In *Proc. NIPS*, pages 873–880, 2002.

[Williams *et al.*, 2009] Christopher Williams, Stefan Klanke, Sethu Vijayakumar, and Kian M Chai. Multi-task Gaussian process learning of robot inverse dynamics. In *Proc. NIPS*, pages 265–272, 2009.

[Wu *et al.*, 2014] Xiang Wu, Kanjian Zhang, and Changyin Sun. Optimal scheduling of multiple sensors in continuous time. *ISA transactions*, 53(3):793–801, 2014.

[Xu *et al.*, 2013] Shuo Xu, Xin An, Xiaodong Qiao, Lijun Zhu, and Lin Li. Multi-output least-squares support vector regression machines. *Pattern Recognition Letters*, 34(9):1078–1084, 2013.

[Xu *et al.*, 2014] Nuo Xu, Kian Hsiang Low, Jie Chen, Keng Kiat Lim, and Etkin Baris Ozgul. Gp-localize: Persistent mobile robot localization using online sparse Gaussian process observation model. In *Proc. AAAI*, pages 2585–2592, 2014.

[Yogatama and Mann, 2014] Dani Yogatama and Gideon Mann. Efficient transfer learning method for automatic hyperparameter tuning. page 10771085, 2014.

[Yu *et al.*, 2005] Kai Yu, Volker Tresp, and Anton Schwaighofer. Learning Gaussian processes from multiple tasks. In *Proc. ICML*, pages 1012–1019, 2005.

[Zhang and Yeung, 2009] Yu Zhang and Dit-Yan Yeung. Semi-supervised multi-task regression. In *Proc. ECML-PKDD*, pages 617–631, 2009.

[Zhang *et al.*, 2016] Yehong Zhang, Trong Nghia Hoang, Kian Hsiang Low, and Mohan Kankanhalli. Near-optimal active learning of multi-output Gaussian processes. In *Proc. AAAI*, pages 2351–2357, 2016.

[Zhang, 2010] Y. Zhang. Multi-task active learning with output constraints. In *Proc. AAAI*, pages 667–672, 2010.

[Zhao *et al.*, 2013] L. Zhao, S. J. Pan, E. W. Xiang, E. Zhong, Z. Lu, and Q. Yang. Active transfer learning for cross-system recommendation. In *Proc. AAAI*, pages 1205–1211, 2013.

[Zhu *et al.*, 2011] Z. Zhu, X. Zhu, Y. Ye, Y. Guo, and X. Xue. Transfer active learning. In *Proc. CIKM*, pages 2169–2172, 2011.

[Zhu, 2005] Xiaojin Zhu. Semi-supervised learning literature survey. 2005.

[Zuluaga *et al.*, 2013] Marcela Zuluaga, Guillaume Sergent, Andreas Krause, and Markus Püschel. Active learning for multi-objective optimization. pages 462–470, 2013.

Appendix A

Appendix of Chapter 4

A.1 Derivation of Novel Active Learning Criterion in Equation (4.4)

$$
\begin{aligned}
&\arg\min_{X:|X|=N} H(Y_{V_t\setminus X_t}|Y_X) \\
&= \arg\max_{X:|X|=N} H(Y_{V_t}|L_U) - H(Y_{V_t\setminus X_t}|Y_X) \\
&= \arg\max_{X:|X|=N} H(Y_{V_t}|L_U) - H(Y_{V_t\setminus X_t}|L_U) + H(Y_{V_t\setminus X_t}|L_U) - \\
&\quad H(Y_{V_t\setminus X_t}|Y_X, L_U) + H(Y_{V_t\setminus X_t}|Y_X, L_U) - H(Y_{V_t\setminus X_t}|Y_X) \\
&= \arg\max_{X:|X|=N} H(Y_{X_t}|L_U, Y_{V_t\setminus X_t}) + I(Y_{V_t\setminus X_t}; Y_X|L_U) - I(L_U; Y_{V_t\setminus X_t}|Y_X) \\
&= \arg\max_{X:|X|=N} H(Y_{X_t}|L_U) - I(L_U; Y_{V_t\setminus X_t}|Y_X) \ .
\end{aligned}
$$

The first equality follows from the fact that $H(Y_{V_t}|L_U)$ is a constant. The third equality is due to the chain rule for entropy $H(Y_{V_t}|L_U) = H(Y_{V_t\setminus X_t}|L_U) + H(Y_{X_t}|L_U, Y_{V_t\setminus X_t})$ as well as the definition of conditional mutual information

$$
I(Y_{V_t\setminus X_t}; Y_X|L_U) \triangleq H(Y_{V_t\setminus X_t}|L_U) - H(Y_{V_t\setminus X_t}|Y_X, L_U)
$$

and

$$I(L_U; Y_{V_t \setminus X_t} | Y_X) \triangleq H(Y_{V_t \setminus X_t} | Y_X) - H(Y_{V_t \setminus X_t} | Y_X, L_U).$$

The last equality follows from structural property **P2** shared by sparse CMOGP regression models in the unifying framework [Álvarez and Lawrence, 2011] described in Section 3.2.2, which results in $H(Y_{X_t}|L_U, Y_{V_t \setminus X_t}) = H(Y_{X_t}|L_U)$ and $I(Y_{V_t \setminus X_t}; Y_X | L_U) = 0$.

A.2 Time Complexity of Evaluating Active Learning Criterion in Equation (4.4)

Due to (4.1), the first term of (4.4) can be written as

$$H(Y_{X_t}|L_U) = \frac{1}{2} \log(2\pi e)^{|X_t|} |\Sigma_{X_t X_t|U}|$$

where $\Sigma_{X_t X_t|U} = \Sigma_{X_t X_t} - \Sigma_{X_t U} \Sigma_{UU}^{-1} \Sigma_{U X_t}$ by definition (see last paragraph of Section 3.2.2). So, evaluating $H(Y_{X_t}|L_U)$ incurs $\mathcal{O}(|U|^3 + N^3)$ time for every $X \subset V$; this worst-case time complexity occurs when all the tuples in X are of measurement type t (i.e., $X = X_t$). Then, the second term of (4.4) can be written as

$$\begin{aligned} I(L_U; Y_{V_t \setminus X_t} | Y_X) &= H(L_U|Y_X) - H(L_U|Y_{X \cup V_t \setminus X_t}) \\ &= \frac{1}{2} \log \frac{|\Sigma_{UU|X}|}{|\Sigma_{UU|X \cup V_t \setminus X_t}|} \\ &= \frac{1}{2} \log \frac{|\Sigma_{UU|X}|}{|\Sigma_{UU|\bigcup_{i \neq t} X_i \cup V_t}|} \end{aligned}$$

where

$$\Sigma_{UU|A} = \Sigma_{UU}(\Sigma_{UU} + \Sigma_{UA} \Lambda_A^{-1} \Sigma_{AU})^{-1} \Sigma_{UU}$$

for any $A \subset D^+$, as derived in [Álvarez and Lawrence, 2011]. Therefore, evaluating $|\Sigma_{UU|X}|$ incurs $\mathcal{O}(|U|^3 + N^3)$ time for every $X \subset V$; this worst-case time complexity

occurs when all the tuples in X are of one measurement type.

Let $A \triangleq \bigcup_{i \neq t} X_i \cup V_t$. Then, by the definition of Λ_A (see last paragraph of Section 3.2.2),

$$\Sigma_{UA}\Lambda_A^{-1}\Sigma_{AU} = \sum_{i \neq t} \Sigma_{UX_i}\Sigma_{X_iX_i|U}^{-1}\Sigma_{X_iU} + \Sigma_{UV_t}\Sigma_{V_tV_t|U}^{-1}\Sigma_{V_tU} \ .$$

Evaluating the $\sum_{i \neq t} \Sigma_{UX_i}\Sigma_{X_iX_i|U}^{-1}\Sigma_{X_iU}$ term incurs $\mathcal{O}(|U|^3 + N^3)$ time for every $X \subset V$; this worst-case time complexity occurs when all the tuples in X are of one measurement type. Note that the $\Sigma_{UV_t}\Sigma_{V_tV_t|U}^{-1}\Sigma_{V_tU}$ term remains the same for every $X \subset V$ (i.e., since it is independent of X) and hence only needs to be computed once in $\mathcal{O}(|V_t|^3)$ time. Therefore, evaluating $|\Sigma_{UU|\bigcup_{i \neq t} X_i \cup V_t}| = |\Sigma_{UU|A}|$ incurs $\mathcal{O}(|U|^3 + N^3)$ time for every $X \subset V$ and a *one-off* cost of $\mathcal{O}(|V_t|^3)$ time. Consequently, evaluating $I(L_U; Y_{V_t \setminus X_t}|Y_X)$ incurs $\mathcal{O}(|U|^3 + N^3)$ time for every $X \subset V$ and a *one-off* cost of $\mathcal{O}(|V_t|^3)$ time. So, evaluating our active learning criterion in (4.4) incurs $\mathcal{O}(|U|^3 + N^3)$ time for every $X \subset V$ and a *one-off* cost of $\mathcal{O}(|V_t|^3)$ time.

A.3 Derivation of Greedy Criterion in Equation (4.7)

If $i = t$, then

$$\begin{aligned}
&F(X \cup \{\langle x,t\rangle\}) - F(X)\\
&= H(Y_{X_t \cup \{\langle x,t\rangle\}}|L_U) - (H(L_U|Y_{X \cup \{\langle x,t\rangle\}}) - H(L_U|Y_{X \cup \{\langle x,t\rangle\} \cup V_t \setminus (X_t \cup \{\langle x,t\rangle\})}))\\
&\quad -(H(Y_{X_t}|L_U) - (H(L_U|Y_X) - H(L_U|Y_{X \cup V_t \setminus X_t})))\\
&= H(Y_{X_t \cup \{\langle x,t\rangle\}}|L_U) - H(Y_{X_t}|L_U) + (H(L_U|Y_X) - H(L_U|Y_{X \cup \{\langle x,t\rangle\}})) \qquad \text{(A.1)}\\
&= H(Y_{\langle x,t\rangle}|Y_{X_t}, L_U) + H(Y_{\langle x,t\rangle}|Y_X) - H(Y_{\langle x,t\rangle}|Y_X, L_U)\\
&= H(Y_{\langle x,t\rangle}|L_U) + H(Y_{\langle x,t\rangle}|Y_X) - H(Y_{\langle x,t\rangle}|L_U)\\
&= H(Y_{\langle x,t\rangle}|Y_X) \ .
\end{aligned}$$

The first equality follows from (4.4) and (4.6). The second equality is due to $H(L_U|Y_{X\cup\{\langle x,t\rangle\}\cup V_t\setminus(X_t\cup\{\langle x,t\rangle\})}) = H(L_U|Y_{X\cup V_t\setminus X_t})$. The third equality is due to the chain rule for entropy $H(Y_{X_t\cup\{\langle x,t\rangle\}}|L_U) = H(Y_{X_t}|L_U) + H(Y_{\langle x,t\rangle}|Y_{X_t}, L_U)$ as well as the definition of conditional mutual information $I(L_U; Y_{\langle x,t\rangle}|Y_X) \triangleq H(L_U|Y_X) - H(L_U|Y_{X\cup\{\langle x,t\rangle\}}) = H(Y_{\langle x,t\rangle}|Y_X) - H(Y_{\langle x,t\rangle}|Y_X, L_U)$. The second last equality follows from structural property **P2** shared by sparse CMOGP regression models in the unifying framework [Álvarez and Lawrence, 2011] described in Section 3.2.2.

Otherwise (i.e., $i \neq t$),

$$
\begin{aligned}
&F(X\cup\{\langle x,i\rangle\}) - F(X)\\
&= H(Y_{X_t}|L_U) \ - (H(L_U|Y_{X\cup\{\langle x,i\rangle\}}) - H(L_U|Y_{X\cup\{\langle x,i\rangle\}\cup V_t\setminus X_t}))\\
&\quad -(H(Y_{X_t}|L_U) - (H(L_U|Y_X) - H(L_U|Y_{X\cup V_t\setminus X_t})))\\
&= H(Y_{X_t}|L_U) - H(Y_{X_t}|L_U) + (H(L_U|Y_X) -\\
&\quad H(L_U|Y_{X\cup\{\langle x,i\rangle\}})) \ + H(L_U|Y_{X\cup V_t\setminus X_t\cup\{\langle x,i\rangle\}}) - H(L_U|Y_{X\cup V_t\setminus X_t})\\
&= H(Y_{\langle x,i\rangle}|Y_X) - H(Y_{\langle x,i\rangle}|L_U, Y_X) +\\
&\quad H(Y_{\langle x,i\rangle}|Y_{X\cup V_t\setminus X_t}, L_U) - H(Y_{\langle x,i\rangle}|Y_{X\cup V_t\setminus X_t})\\
&= H(Y_{\langle x,i\rangle}|Y_X) - H(Y_{\langle x,i\rangle}|L_U) + H(Y_{\langle x,i\rangle}|L_U) - H(Y_{\langle x,i\rangle}|Y_{X\cup V_t\setminus X_t})\\
&= H(Y_{\langle x,i\rangle}|Y_X) - H(Y_{\langle x,i\rangle}|Y_{X\cup V_t\setminus X_t}) \ .
\end{aligned}
\tag{A.2}
$$

The first equality follows from (4.4) and (4.6). The third equality is due to the definition of conditional mutual information

$$I(L_U; Y_{\langle x,i\rangle}|Y_X) \triangleq H(L_U|Y_X) - H(L_U|Y_{X\cup\{\langle x,i\rangle\}}) = H(Y_{\langle x,i\rangle}|Y_X) - H(Y_{\langle x,i\rangle}|L_U, Y_X)$$

and

$$
\begin{aligned}
I(L_U; Y_{\langle x,i\rangle}|Y_{X\cup V_t\setminus X_t}) &\triangleq H(L_U|Y_{X\cup V_t\setminus X_t}) - H(L_U|Y_{X\cup V_t\setminus X_t\cup\{\langle x,i\rangle\}})\\
&= H(Y_{\langle x,i\rangle}|Y_{X\cup V_t\setminus X_t}) - H(Y_{\langle x,i\rangle}|Y_{X\cup V_t\setminus X_t}, L_U).
\end{aligned}
$$

The second last equality follows from structural properties **P1** and **P2** shared by sparse CMOGP regression models in the unifying framework [Álvarez and Lawrence, 2011] described in Section 3.2.2. Therefore, (4.7) results.

A.4 Proof of Proposition 1

Before proving Proposition 1, the following lemmas are needed:

Lemma 3. *For all* $X \subset V$ *and* $\langle x, i\rangle \in V \setminus X$, $\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x,i\rangle|X} \geq \sigma^2_{n_i}$.

Its proof follows closely to that of Lemma 6 in [Cao *et al.*, 2013].

Lemma 4. *Assuming absence of suppressor variables, for all* $X \subset V$ *and* $\langle x, i\rangle, \langle x', j\rangle \in V \setminus X$, $|\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x',j\rangle|X}| \leq 2|\sigma_{\langle x,i\rangle\langle x',j\rangle}|$.

Proof. If $i = j$, then

$$|\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x',j\rangle}| = |\sigma_{\langle x,i\rangle\langle x',j\rangle}| \leq 2|\sigma_{\langle x,i\rangle\langle x',j\rangle}| \ . \tag{A.3}$$

If $i \neq j$, then

$$\begin{aligned} |\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x',j\rangle}| &= |\Gamma_{\langle x,i\rangle\langle x',j\rangle}| \\ &= |\sigma_{\langle x,i\rangle\langle x',j\rangle} - \Sigma_{\langle x,i\rangle\langle x',j\rangle|U}| \\ &\leq |\sigma_{\langle x,i\rangle\langle x',j\rangle}| + |\Sigma_{\langle x,i\rangle\langle x',j\rangle|U}| \\ &\leq 2|\sigma_{\langle x,i\rangle\langle x',j\rangle}| \ . \end{aligned} \tag{A.4}$$

The first equality is due to (3.4) while the second equality follows from the definition of $\Gamma_{\langle x,i\rangle\langle x',j\rangle}$ (see last paragraph of Section 3.2.2). The last inequality follows from the practical assumption of absence of suppressor variables [Das and Kempe, 2008]: $|\Sigma_{\langle x,i\rangle\langle x',j\rangle|U}| \leq |\sigma_{\langle x,i\rangle\langle x',j\rangle}|$. Then,

$$|\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x',j\rangle|X}| \leq |\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x',j\rangle}| \leq 2|\sigma_{\langle x,i\rangle\langle x',j\rangle}| \ .$$

The first inequality follows from the practical assumption of absence of suppressor variables [Das and Kempe, 2008]. The second inequality is due to (A.3) and (A.4). □

Main Proof. Let $B \triangleq V_t \setminus X_t$. Using the spectral theorem, $(\Sigma^{\text{PITC}}_{BB|X})^{-1} = WQW^\top$ where the columns of W are the eigenvectors of $(\Sigma^{\text{PITC}}_{BB|X})^{-1}$ and Q is a diagonal matrix comprising the eigenvalues of $(\Sigma^{\text{PITC}}_{BB|X})^{-1}$. Let $\lambda_{\max}(A)$ and $\lambda_{\min}(A)$ denote, respectively, the maximum and minimum eigenvalues of matrix A, and $\alpha \triangleq W^\top \Sigma^{\text{PITC}}_{B\langle x,i\rangle|X}$.

$$
\begin{aligned}
&\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x,i\rangle|X} - \Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x,i\rangle|X\cup V_t\setminus X_t}\\
&= \Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x,i\rangle|X} - \left(\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x,i\rangle|X} - \Sigma^{\text{PITC}}_{\langle x,i\rangle B|X}(\Sigma^{\text{PITC}}_{BB|X})^{-1}\Sigma^{\text{PITC}}_{B\langle x,i\rangle|X}\right)\\
&= \Sigma^{\text{PITC}}_{\langle x,i\rangle B|X}(\Sigma^{\text{PITC}}_{BB|X})^{-1}\Sigma^{\text{PITC}}_{B\langle x,i\rangle|X}\\
&= \Sigma^{\text{PITC}}_{\langle x,i\rangle B|X}WQW^\top\Sigma^{\text{PITC}}_{B\langle x,i\rangle|X}\\
&= \alpha^\top Q\alpha\\
&\leq \lambda_{\max}((\Sigma^{\text{PITC}}_{BB|X})^{-1})\alpha^\top\alpha\\
&= \frac{\Sigma^{\text{PITC}}_{\langle x,i\rangle B|X}WW^\top\Sigma^{\text{PITC}}_{B\langle x,i\rangle|X}}{\lambda_{\min}(\Sigma^{\text{PITC}}_{BB|X})}\\
&= \frac{\|\Sigma^{\text{PITC}}_{\langle x,i\rangle B|X}\|_2^2}{\lambda_{\min}(\Sigma^{\text{PITC}}_{BB|X})}\\
&= \frac{\sum_{\langle x',t\rangle\in B}|\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x',t\rangle|X}|^2}{\lambda_{\min}(\Sigma^{\text{PITC}}_{BB|X})}\\
&\leq \frac{\sum_{\langle x',t\rangle\in B}4|\sigma_{\langle x,i\rangle\langle x',t\rangle}|^2}{\lambda_{\min}(\Sigma^{\text{PITC}}_{BB|X})}\\
&\leq \frac{4\sigma_{s_i}^2\sigma_{s_t}^2\sum_{\langle x',t\rangle\in B}\mathcal{N}(x-x'|\underline{0},P_0^{-1}+P_i^{-1}+P_t^{-1})^2}{\sigma_{n_t}^2}\\
&= 4\rho_t\sigma_{s_i}^2 R(\langle x,i\rangle,B)\ .
\end{aligned}
\tag{A.5}
$$

The first equality is due to the incremental update formula of GP posterior variance (see Appendix C in [Xu *et al.*, 2014]). The first inequality is due to the fact that Q is a diagonal matrix comprising the eigenvalues of $(\Sigma^{\text{PITC}}_{BB|X})^{-1}$. The fifth equality is due

to a property of eigenvalues that $\lambda_{\max}(A^{-1}) = 1/\lambda_{\min}(A)$. The sixth equality follows from the fact that $WW^\top = I$. The second inequality follows from Lemma 4. The third inequality is due to (3.2) and the fact that $\lambda_{\min}(\Sigma^{\text{PITC}}_{BB|X}) = \lambda_{\min}(\Sigma^{\text{PITC}}_{BB|X} - \sigma^2_{n_t} I + \sigma^2_{n_t} I) = \lambda_{\min}(\Sigma^{\text{PITC}}_{BB|X} - \sigma^2_{n_t} I) + \sigma^2_{n_t} \geq \sigma^2_{n_t}$ since $\lambda_{\min}(\Sigma^{\text{PITC}}_{BB|X} - \sigma^2_{n_t} I) \geq 0$ (i.e., $\Sigma^{\text{PITC}}_{BB|X} - \sigma^2_{n_t} I$ is a positive semi-definite matrix). Then,

$$
\begin{aligned}
& H(Y_{\langle x,i\rangle}|Y_X) - H(Y_{\langle x,i\rangle}|Y_{X\cup V_t\setminus X_t}) \\
&= \frac{1}{2}\log\frac{\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x,i\rangle|X}}{\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x,i\rangle|X\cup B}} \\
&\leq \frac{1}{2}\log\frac{\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x,i\rangle|X\cup B} + 4\rho_t\sigma^2_{s_i}R(\langle x,i\rangle, B)}{\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x,i\rangle|X\cup B}} \\
&\leq \frac{1}{2}\log\left(1 + \frac{4\rho_t\sigma^2_{s_i}R(\langle x,i\rangle, B)}{\sigma^2_{n_i}}\right) \\
&= \frac{1}{2}\log(1 + 4\rho_t\rho_i R(\langle x,i\rangle, B)) \ .
\end{aligned}
$$

The first inequality is due to (A.5) while the second inequality follows from Lemma 3.

A.5 Proof of Theorem 1

If $i = t$, then

$$
H(Y_{\langle x,t\rangle}|Y_X) = \frac{1}{2}\log(2\pi e)\Sigma^{\text{PITC}}_{\langle x,t\rangle\langle x,t\rangle|X}
$$

where $\Sigma^{\text{PITC}}_{\langle x,t\rangle\langle x,t\rangle|X}$ is previously defined in (3.4). So, evaluating $H(Y_{\langle x,t\rangle}|Y_X)$ incurs $\mathcal{O}(|U|^2)$ time for every $\langle x,t\rangle \in V_t \setminus X_t$ and $\mathcal{O}(|U|^3 + N^3)$ time in each iteration; this worst-case time complexity occurs when all the tuples in X are of one measurement type.

Otherwise (i.e., $i \neq t$),

$$H(Y_{\langle x,i\rangle}|Y_X) - H(Y_{\langle x,i\rangle}|Y_{X\cup V_t\setminus X_t}) = \frac{1}{2}\log\frac{\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x,i\rangle|X}}{\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x,i\rangle|X\cup V_t\setminus X_t}} = \frac{1}{2}\log\frac{\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x,i\rangle|X}}{\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x,i\rangle|\bigcup_{i\neq t} X_i\cup V_t}}$$

where $\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x,i\rangle|X}$ and $\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x,i\rangle|\bigcup_{i\neq t} X_i\cup V_t}$ are previously defined in (3.4). Therefore, evaluating $\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x,i\rangle|X}$ incurs $\mathcal{O}(|U|^2)$ time for every $\langle x,i\rangle \in V_{\text{-}t}\setminus X_{\text{-}t}$ and $\mathcal{O}(|U|^3+N^3)$ time in each iteration; this worst-case time complexity occurs when all the tuples in X are of one measurement type.

Let $A \triangleq \bigcup_{i\neq t} X_i \cup V_t$. Then, by the definition of Λ_A (see last paragraph of Section 3.2.2),

$$\Sigma_{UA}\Lambda_A^{-1}\Sigma_{AU} = \sum_{i\neq t}\Sigma_{UX_i}\Sigma^{-1}_{X_iX_i|U}\Sigma_{X_iU} + \Sigma_{UV_t}\Sigma^{-1}_{V_tV_t|U}\Sigma_{V_tU}\ .$$

Evaluating the $\sum_{i\neq t}\Sigma_{UX_i}\Sigma^{-1}_{X_iX_i|U}\Sigma_{X_iU}$ term incurs $\mathcal{O}(|U|^3+N^3)$ time in each iteration; this worst-case time complexity occurs when all the tuples in X are of one measurement type. Note that the $\Sigma_{UV_t}\Sigma^{-1}_{V_tV_t|U}\Sigma_{V_tU}$ term remains the same in each iteration (i.e., since it is independent of X) and hence only needs to be computed once in $\mathcal{O}(|V_t|^3)$ time in our approximation algorithm. As a result, evaluating $\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x,i\rangle|\bigcup_{i\neq t} X_i\cup V_t} = \Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x,i\rangle|A}$ (specifically, its efficient formulation exploiting $\Sigma_{UA}\Lambda_A^{-1}\Sigma_{AU}$, as shown in [Álvarez and Lawrence, 2011]) incurs $\mathcal{O}(|U|^2)$ time for every $\langle x,i\rangle \in V_{\text{-}t}\setminus X_{\text{-}t}$ and $\mathcal{O}(|U|^3+N^3)$ time in each iteration, and a *one-off* cost of $\mathcal{O}(|V_t|^3)$ time. Consequently, evaluating $H(Y_{\langle x,i\rangle}|Y_X) - H(Y_{\langle x,i\rangle}|Y_{X\cup V_t\setminus X_t})$ incurs $\mathcal{O}(|U|^2)$ time for every $\langle x,i\rangle \in V_{\text{-}t}\setminus X_{\text{-}t}$ and $\mathcal{O}(|U|^3+N^3)$ time in each iteration, and a *one-off* cost of $\mathcal{O}(|V_t|^3)$ time.

Since $|U| \leq |V_t| < |V|$, our approximation algorithm thus incurs $\mathcal{O}(N(|V||U|^2 + N^3) + |V_t|^3)$ time.

A.6 Proof of Lemma 1

To prove that $F(X)$ is ϵ-submodular, we have to show that

$$F(X' \cup \{\langle x, i\rangle\}) - F(X') \leq F(X \cup \{\langle x, i\rangle\}) - F(X) + \epsilon$$

for any $X \subseteq X' \subseteq V$ and $\langle x, i\rangle \in V \setminus X'$. Before doing this, the following lemma is needed:

Lemma 5. *Suppose that $\epsilon_1 \geq 0$ is given. For any $\langle x, i\rangle \in V_{\text{-}t} \setminus X'_{\text{-}t}$, if $\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x,i\rangle|X\cup V_t\setminus X'_t} - \Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x,i\rangle|X'\cup V_t\setminus X'_t} \leq \epsilon_1$, then $I(Y_{\langle x,i\rangle}; Y_{V_t\setminus X'_t}|Y_{X'}) \leq I(Y_{\langle x,i\rangle}; Y_{V_t\setminus X'_t}|Y_X) + \epsilon$ where $\epsilon = 0.5\log(1 + \epsilon_1/\sigma^2_{n^*})$.*

Proof. Let $\bar{X} \triangleq X' \setminus X$. Then,

$$\begin{aligned}
&I(Y_{\langle x,i\rangle}; Y_{\bar{X}}|Y_{X\cup V_t\setminus X'_t}) \\
&= H(Y_{\langle x,i\rangle}|Y_{X\cup V_t\setminus X'_t}) - H(Y_{\langle x,i\rangle}|Y_{\bar{X}\cup X\cup V_t\setminus X'_t}) \\
&= \frac{1}{2}\log\frac{\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x,i\rangle|X\cup V_t\setminus X'_t}}{\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x,i\rangle|\bar{X}\cup X\cup V_t\setminus X'_t}} \\
&\leq \frac{1}{2}\log\frac{\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x,i\rangle|\bar{X}\cup X\cup V_t\setminus X'_t} + \epsilon_1}{\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x,i\rangle|\bar{X}\cup X\cup V_t\setminus X'_t}} \\
&= \frac{1}{2}\log\left(1 + \frac{\epsilon_1}{\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x,i\rangle|\bar{X}\cup X\cup V_t\setminus X'_t}}\right) \\
&\leq \frac{1}{2}\log\left(1 + \frac{\epsilon_1}{\sigma^2_{n_i}}\right) \\
&\leq \frac{1}{2}\log\left(1 + \frac{\epsilon_1}{\sigma^2_{n^*}}\right).
\end{aligned} \tag{A.6}$$

The first inequality is due to the sufficient condition. The second inequality follows from Lemma 3. Then, by the definition of conditional mutual information,

$$
\begin{aligned}
&I(Y_{\langle x,i\rangle}; Y_{V_t\setminus X'_t}|Y_{\bar{X}\cup X}) + I(Y_{\langle x,i\rangle}; Y_{\bar{X}}|Y_X)\\
&= H(Y_{\langle x,i\rangle}|Y_{\bar{X}\cup X}) - H(Y_{\langle x,i\rangle}|Y_{\bar{X}\cup X\cup V_t\setminus X'_t}) + H(Y_{\langle x,i\rangle}|Y_X) - H(Y_{\langle x,i\rangle}|Y_{\bar{X}\cup X})\\
&= H(Y_{\langle x,i\rangle}|Y_X) - H(Y_{\langle x,i\rangle}|Y_{\bar{X}\cup X\cup V_t\setminus X'_t})\\
&= H(Y_{\langle x,i\rangle}|Y_X) - H(Y_{\langle x,i\rangle}|Y_{X\cup V_t\setminus X'_t}) + H(Y_{\langle x,i\rangle}|Y_{X\cup V_t\setminus X'_t}) - H(Y_{\langle x,i\rangle}|Y_{\bar{X}\cup X\cup V_t\setminus X'_t})\\
&= I(Y_{\langle x,i\rangle}; Y_{V_t\setminus X'_t}|Y_X) + I(Y_{\langle x,i\rangle}; Y_{\bar{X}}|Y_{X\cup V_t\setminus X'_t})\ .
\end{aligned}
$$

Therefore,

$$
\begin{aligned}
I(Y_{\langle x,i\rangle}; Y_{V_t\setminus X'_t}|Y_{X'}) &= I(Y_{\langle x,i\rangle}; Y_{V_t\setminus X'_t}|Y_{\bar{X}\cup X})\\
&= I(Y_{\langle x,i\rangle}; Y_{V_t\setminus X'_t}|Y_X) + I(Y_{\langle x,i\rangle}; Y_{\bar{X}}|Y_{X\cup V_t\setminus X'_t}) - I(Y_{\langle x,i\rangle}; Y_{\bar{X}}|Y_X)\\
&\le I(Y_{\langle x,i\rangle}; Y_{V_t\setminus X'_t}|Y_X) + I(Y_{\langle x,i\rangle}; Y_{\bar{X}}|Y_{X\cup V_t\setminus X'_t})\\
&\le I(Y_{\langle x,i\rangle}; Y_{V_t\setminus X'_t}|Y_X) + 0.5\log\left(1+\frac{\epsilon_1}{\sigma^2_{n^*}}\right)\ .
\end{aligned}
$$

The first inequality is due to the fact that conditional mutual information is non-negative. The last inequality follows from (A.6). □

Main Proof. To prove that $F(X)$ is ϵ-submodular, we have to show that $H(Y_{\langle x,i\rangle}|Y_{X'}) - \delta_i H(Y_{\langle x,i\rangle}|Y_{X'\cup V_t\setminus X'_t}) \le H(Y_{\langle x,i\rangle}|Y_X) - \delta_i H(Y_{\langle x,i\rangle}|Y_{X\cup V_t\setminus X_t}) + \epsilon$ for any $X \subseteq X' \subseteq V$ and $\langle x,i\rangle \in V\setminus X'$.

If $i = t$, then $H(Y_{\langle x,i\rangle}|Y_{X'}) \le H(Y_{\langle x,i\rangle}|Y_X) \le H(Y_{\langle x,i\rangle}|Y_X) + \epsilon$ for any $\epsilon \ge 0$ due to the "information never hurts" bound for entropy [Cover and Thomas, 1991].

Otherwise (i.e., $i \ne t$),

$$
\begin{aligned}
&H(Y_{\langle x,i\rangle}|Y_{X'}) - H(Y_{\langle x,i\rangle}|Y_{X'\cup V_t\setminus X'_t})\\
&\le H(Y_{\langle x,i\rangle}|Y_X) - H(Y_{\langle x,i\rangle}|Y_{X\cup V_t\setminus X'_t}) + \epsilon\\
&\le H(Y_{\langle x,i\rangle}|Y_X) - H(Y_{\langle x,i\rangle}|Y_{X\cup V_t\setminus X_t}) + \epsilon
\end{aligned}
$$

where $\epsilon = 0.5 \log(1 + \epsilon_1/\sigma^2_{n^*})$. The first inequality is due to Lemma 5. The second inequality follows from the "information never hurts" bound for entropy [Cover and Thomas, 1991]: $H(Y_{\langle x,i\rangle}|Y_{X\cup V_t\setminus X'_t}) \geq H(Y_{\langle x,i\rangle}|Y_{X\cup V_t\setminus X_t})$ since $(V_t \setminus X'_t) \subseteq (V_t \setminus X_t)$.

A.7 Proof of Theorem 2

Our proof here is similar to that of Theorem 1.5 in [Krause and Golovin, 2014] which is a generalization of the well-known result of Nemhauser *et al.* (1978). The key difference is that we exploit ϵ-submodularity of $F(X)$ (i.e., Lemma 1) instead of submodularity, as shown below for completeness.

Let $X^* \triangleq \{\langle x_1, s_1\rangle^*, \ldots, \langle x_N, s_N\rangle^*\}$ be the optimal set of selected observations, X^k be the set of tuples selected by our approximation algorithm in iteration $k = 1, \ldots, N$, $X^0 \triangleq \emptyset$, and $\Delta(\langle x, i\rangle|X) \triangleq F(X \cup \{\langle x, i\rangle\}) - F(X)$. Then,

$$
\begin{aligned}
& F(X^*) \\
& \leq F(X^* \cup X^k) \\
& = F(X^k) + \sum_{j=1}^{N} \Delta\left(\langle x_j, s_j\rangle^* \middle| \bigcup_{r=1}^{j-1} \{\langle x_r, s_r\rangle^*\} \cup X^k\right) \\
& \leq F(X^k) + \sum_{j=1}^{N} \left(\Delta(\langle x_j, s_j\rangle^*|X^k) + \epsilon\right) \\
& \leq F(X^k) + \sum_{j=1}^{N} \left(F(X^{k+1}) - F(X^k) + \epsilon\right) \\
& \leq F(X^k) + N\left(F(X^{k+1}) - F(X^k) + \epsilon\right).
\end{aligned}
$$

The first inequality follows from the nondecreasing property of $F(X)$. The first equality is a straightforward telescoping sum. The second inequality follows from the ϵ-submodularity of $F(X)$, as proven in Lemma 1. The third inequality follows from

(4.7). Then,

$$F(X^*) - F(X^k) \leq N\left(F(X^{k+1}) - F(X^k) + \epsilon\right). \tag{A.7}$$

Let $\zeta_k \triangleq F(X^*) - F(X^k)$. Then, (A.7) can be rewritten as $\zeta_k \leq N(\zeta_k - \zeta_{k+1} + \epsilon)$ which can be rearranged to yield

$$\zeta_{k+1} \leq \left(1 - \frac{1}{N}\right)\zeta_k + \epsilon\ . \tag{A.8}$$

Then, by recursion of (A.8), it is straightforward to get

$$\zeta_k \leq \left(1 - \frac{1}{N}\right)^k \zeta_0 + N\left(1 - \left(1 - \frac{1}{N}\right)^k\right)\epsilon\ . \tag{A.9}$$

Then, by substituting $\zeta_k = F(X^*) - F(X^k)$ and $\zeta_0 = F(X^*) - F(X^0) = F(X^*)$, (A.9) can be rearranged to

$$\begin{aligned} F(X^k) &\geq \left(1 - \left(1 - \frac{1}{N}\right)^k\right)(F(X^*) - N\epsilon) \\ &\geq (1 - e^{-k/N})(F(X^*) - N\epsilon)\ . \end{aligned}$$

The second inequality follows from the well-known inequality $e^{-x} \geq 1 - x$. Finally, Theorem 2 is obtained when $k = N$ and $\epsilon = 0.5\log(1+\epsilon_1/\sigma^2_{n^*})$, as defined in Lemma 1.

A.8 Proof of Lemma 2

Let $B \triangleq \widetilde{X} \cup V_t \setminus X_t$ and $A \triangleq X \setminus \widetilde{X}$. From the incremental update formula of GP posterior variance (see Appendix C in [Xu *et al.*, 2014]),

$$\begin{aligned}
&\Sigma^{\mathrm{PITC}}_{\langle x,i\rangle\langle x,i\rangle|B} - \Sigma^{\mathrm{PITC}}_{\langle x,i\rangle\langle x,i\rangle|B\cup A} \\
&= \Sigma^{\mathrm{PITC}}_{\langle x,i\rangle\langle x,i\rangle|B} - \big(\Sigma^{\mathrm{PITC}}_{\langle x,i\rangle\langle x,i\rangle|B} - \Sigma^{\mathrm{PITC}}_{\langle x,i\rangle A|B}(\Sigma^{\mathrm{PITC}}_{AA|B})^{-1}\Sigma^{\mathrm{PITC}}_{A\langle x,i\rangle|B}\big) \\
&= \Sigma^{\mathrm{PITC}}_{\langle x,i\rangle A|B}(\Sigma^{\mathrm{PITC}}_{AA|B})^{-1}\Sigma^{\mathrm{PITC}}_{A\langle x,i\rangle|B} \ .
\end{aligned} \tag{A.10}$$

Let $\Sigma^{\mathrm{PITC}}_{AA|B} \triangleq C+E$ where C is defined as a matrix with the same diagonal components as $\Sigma^{\mathrm{PITC}}_{AA|B}$ and off-diagonal components 0 while E is defined as a matrix with diagonal components 0 and the same off-diagonal components as $\Sigma^{\mathrm{PITC}}_{AA|B}$. Then,

$$\begin{aligned}
\|C^{-1}\|_2 &= \lambda_{\max}(C^{-1}) \\
&= \frac{1}{\lambda_{\min}(C)} \\
&= \frac{1}{\min_{\langle x,i\rangle\in A}\Sigma^{\mathrm{PITC}}_{\langle x,i\rangle\langle x,i\rangle|B}} \\
&\leq \frac{1}{\sigma^2_{n_i}} \leq \frac{1}{\sigma^2_{n^*}} \ .
\end{aligned} \tag{A.11}$$

The first equality is due to a property of matrix norm in Section 10.4.5 in [Petersen and Pedersen, 2012]. The second equality is due to a property of eigenvalues that $\lambda_{\max}(C^{-1}) = 1/\lambda_{\min}(C)$. The third equality is due to the diagonal property of C. The first inequality is due to Lemma 3.

Matrix E comprises off-diagonal components $\Sigma^{\mathrm{PITC}}_{\langle x,i\rangle\langle x',j\rangle|B}$ for all $\langle x,i\rangle, \langle x',j\rangle \in A$

such that $\langle x, i\rangle \neq \langle x', j\rangle$, each of which has an absolute value not more than $2\sigma_{s^*}^2\xi^{p^2}$:

$$
\begin{aligned}
&|\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x',j\rangle|B}| \\
&\leq 2|\sigma_{\langle x,i\rangle\langle x',j\rangle}| \\
&= 2|\sigma_{s_i}\sigma_{s_j}|\mathcal{N}(x - x'|\underline{0}, P_0^{-1} + P_i^{-1} + P_j^{-1}) \\
&= 2|\sigma_{s_i}\sigma_{s_j}| \exp\left\{-\frac{1}{2}\sum_{v=1}^{d}\frac{(x_v - x'_v)^2}{\ell_v^{ij}}\right\} \\
&\leq 2|\sigma_{s_i}\sigma_{s_j}| \exp\left\{-\frac{(x_1 - x'_1)^2}{2\ell_1^{ij}}\right\} \\
&\leq 2|\sigma_{s_i}\sigma_{s_j}| \exp\left\{-\frac{p^2\omega^2}{2\ell}\right\} \\
&= 2|\sigma_{s_i}\sigma_{s_j}|\xi^{p^2} \\
&\leq 2\sigma_{s^*}^2\xi^{p^2}
\end{aligned}
$$

where x_v is the v-th component of a d-dimensional location vector x and ℓ_v^{ij} denotes the v-th diagonal component of $P_0^{-1} + P_i^{-1} + P_j^{-1}$. The first inequality follows from Lemma 4. The second equality is due to the precision matrices being diagonal. The third inequality follows from $\ell \triangleq \max_{i,j\in\{1,\ldots,M\}} \ell_1^{ij}$ and the fact that the distance between x_1 and x'_1 of any $\langle x, i\rangle, \langle x', j\rangle \in A$ must be at least $p\omega$ due to the construction of V^-. Therefore,

$$\|E\|_2 \leq 2N\sigma_{s^*}^2\xi^{p^2} \tag{A.12}$$

due to a property that the 2-norm of a matrix is at most its largest absolute component multiplied by its dimension [Golub and Van Loan, 1996].

Similarly, $\Sigma^{\text{PITC}}_{\langle x,i\rangle A|B}$ comprises components $\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x',j\rangle|B}$ for all $\langle x', j\rangle \in A$, each of which has an absolute value not more than $2\sigma_{s^*}^2\xi^{p^2}$:

$$|\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x',j\rangle|B}| \leq 2|\sigma_{\langle x,i\rangle\langle x',j\rangle}| \leq 2\sigma_{s^*}^2\xi^{p^2}. \tag{A.13}$$

Now,

$$\begin{aligned}
&\Sigma^{\text{PITC}}_{\langle x,i\rangle A|B}(C+E)^{-1}\Sigma^{\text{PITC}}_{A\langle x,i\rangle|B} - \Sigma^{\text{PITC}}_{\langle x,i\rangle A|B}C^{-1}\Sigma^{\text{PITC}}_{A\langle x,i\rangle|B} \\
&= \Sigma^{\text{PITC}}_{\langle x,i\rangle A|B}\left\{(C+E)^{-1} - C^{-1}\right\}\Sigma^{\text{PITC}}_{A\langle x,i\rangle|B} \\
&\leq \|\Sigma^{\text{PITC}}_{\langle x,i\rangle A|B}\|_2^2\|(C+E)^{-1} - C^{-1}\|_2 \\
&\leq \sum_{\langle x',j\rangle\in A}|\Sigma^{\text{PITC}}_{\langle x,i\rangle\langle x',j\rangle|B}|^2\frac{\|C^{-1}\|_2\|E\|_2}{\frac{1}{\|C^{-1}\|_2} - \|E\|_2} \\
&\leq 4N\sigma_{s^*}^4\xi^{2p^2}\frac{\|C^{-1}\|_2\|E\|_2}{\frac{1}{\|C^{-1}\|_2} - \|E\|_2}\ .
\end{aligned} \tag{A.14}$$

The first inequality is due to Cauchy-Schwarz inequality and submultiplicativity of the matrix norm [Stewart and Sun, 1990]. The second inequality follows from an important result in the perturbation theory of matrix inverses (in particular, Theorem III.2.5 in [Stewart and Sun, 1990]). It requires the assumption $\|C^{-1}E\|_2 < 1$. Using (A.11), (A.12), and the matrix norm property in Section 10.4.2 in [Petersen and Pedersen, 2012], this assumption can be satisfied by

$$\|C^{-1}E\|_2 \leq \|C^{-1}\|_2\|E\|_2 \leq \frac{2N\sigma_{s^*}^2\xi^{p^2}}{\sigma_{n^*}^2} < 1.$$

Then,

$$p^2 > \log\left(\frac{\sigma_{n^*}^2}{2N\sigma_{s^*}^2}\right)\Big/\log\xi\ . \tag{A.15}$$

The last inequality in (A.14) is due to (A.13).

Then, from both (A.10) and (A.14),

$$
\begin{aligned}
&\Sigma^{\mathrm{PITC}}_{\langle x,i\rangle\langle x,i\rangle|B} - \Sigma^{\mathrm{PITC}}_{\langle x,i\rangle\langle x,i\rangle|B\cup A} \\
&= \Sigma^{\mathrm{PITC}}_{\langle x,i\rangle A|B}(C+E)^{-1}\Sigma^{\mathrm{PITC}}_{A\langle x,i\rangle|B} \\
&\leq \Sigma^{\mathrm{PITC}}_{\langle x,i\rangle A|B}C^{-1}\Sigma^{\mathrm{PITC}}_{A\langle x,i\rangle|B} + 4N\sigma_{s^*}^4\xi^{2p^2}\frac{\|C^{-1}\|_2\|E\|_2}{\frac{1}{\|C^{-1}\|_2}-\|E\|_2} \\
&\leq \|\Sigma^{\mathrm{PITC}}_{\langle x,i\rangle A|B}\|_2^2\|C^{-1}\|_2 + 4N\sigma_{s^*}^4\xi^{2p^2}\frac{\|C^{-1}\|_2\|E\|_2}{\frac{1}{\|C^{-1}\|_2}-\|E\|_2} \\
&\leq 4N\sigma_{s^*}^4\xi^{2p^2}\|C^{-1}\|_2 + 4N\sigma_{s^*}^4\xi^{2p^2}\frac{\|C^{-1}\|_2\|E\|_2}{\frac{1}{\|C^{-1}\|_2}-\|E\|_2} \\
&= 4N\sigma_{s^*}^4\xi^{2p^2}\|C^{-1}\|_2\left(1+\frac{\|E\|_2}{\frac{1}{\|C^{-1}\|_2}-\|E\|_2}\right) \\
&= \frac{4N\sigma_{s^*}^4\xi^{2p^2}}{\frac{1}{\|C^{-1}\|_2}-\|E\|_2} \\
&\leq \frac{4N\sigma_{s^*}^4\xi^{2p^2}}{\sigma_{n^*}^2-2N\sigma_{s^*}^2\xi^{p^2}} \ .
\end{aligned}
$$

The first inequality is due to (A.14). The second inequality is due to Cauchy-Schwarz inequality. The third inequality is due to (A.13). The last inequality follows from (A.11) and (A.12).

To satisfy (4.8) in Lemma 1, let

$$
\frac{4N\sigma_{s^*}^4\xi^{2p^2}}{\sigma_{n^*}^2-2N\sigma_{s^*}^2\xi^{p^2}} \leq \epsilon_1 \ .
$$

Then,

$$
p^2 \geq \log\left\{\frac{1}{4\sigma_{s^*}^2}\left(\sqrt{\epsilon_1^2+\frac{4\epsilon_1\sigma_{n^*}^2}{N}}-\epsilon_1\right)\right\}\Bigg/\log\xi \ . \tag{A.16}
$$

Finally, from both (A.15) and (A.16), Lemma 2 results.

Appendix B

Appendix of Chapter 5

B.1 Derivation of (5.12)

Firstly, let A be a $d \times d$ positive-definite diagonal matrix and x, x', w, and b be d-dimensional vectors. Then the following convolutional result can be derived to be used in our derivation of (5.12):

$$\begin{aligned}
&\int_{x' \in D} e^{-\frac{1}{2}(x-x')^\top A(x-x')} e^{j(w^\top x'+b)} \, \mathrm{d}x' \\
&= e^{jb} \int_{x' \in D} e^{-\frac{1}{2}(x^\top Ax - 2x^\top Ax' + x'^\top Ax') + jw^\top x'} \, \mathrm{d}x' \\
&= e^{-\frac{1}{2}x^\top Ax + jb} \int_{x' \in D} e^{-\frac{1}{2}x'^\top Ax' + (x^\top A + jw^\top)x'} \, \mathrm{d}x' \\
&= \sqrt{\frac{(2\pi)^d}{|A|}} e^{-\frac{1}{2}x^\top Ax + jb} e^{\frac{1}{2}(x^\top A + jw^\top)A^{-1}(x^\top A + jw^\top)^\top} \\
&= \sqrt{\frac{(2\pi)^d}{|A|}} e^{-\frac{1}{2}x^\top Ax + jb + \frac{1}{2}x^\top Ax + jx^\top w - \frac{1}{2}w^\top A^{-1}w} \\
&= \sqrt{\frac{(2\pi)^d}{|A|}} e^{j(b + x^\top w) - \frac{1}{2}w^\top A^{-1}w}.
\end{aligned} \tag{B.1}$$

The third equality follows from a result generalizing the Gaussian integral described at https://en.wikipedia.org/wiki/Gaussian_integral#Generalizations.

From (5.2),

$$
\begin{aligned}
f_i(x) &= \int_{x' \in D} K_i(x - x')\ L(x')\ \mathrm{d}x' \\
&\approx \int_{x' \in D} K_i(x - x')\ \phi(x')^\top \theta\ \mathrm{d}x' \\
&= \sqrt{2\alpha/m} \times \theta^\top \left(\int_{x' \in D} K_i(x - x') \cos(w_q^\top x' + b_q)\ \mathrm{d}x' \right)^\top_{q=1,\ldots,m} \\
&= \sigma_{s_i} \sqrt{\frac{2\alpha}{m(2\pi)^d |P_i^{-1}|}} \times \theta^\top \left(\int_{x' \in D} e^{-\frac{1}{2}(x-x')^\top P_i (x-x')} \cos(w_q^\top x' + b_q)\ \mathrm{d}x' \right)^\top_{q=1,\ldots,m} \\
&= \sigma_{s_i} \sqrt{\frac{2\alpha}{m(2\pi)^d |P_i^{-1}|}} \times \theta^\top \left(\frac{1}{2} \int_{x' \in D} e^{-\frac{1}{2}(x-x')^\top P_i (x-x')} \left(e^{j(w_q^\top x' + b_q)} + e^{-j(w_q^\top x' + b_q)} \right)\ \mathrm{d}x' \right)^\top_{q=1,\ldots,} \\
&= \frac{1}{2} \sigma_{s_i} \sqrt{\frac{2\alpha}{m(2\pi)^d |P_i^{-1}|}} \times \sqrt{\frac{(2\pi)^d}{|P_i|}} \times \theta^\top \left(e^{j(b_q + x^\top w_q) - \frac{1}{2} w_q^\top P_i^{-1} w_q} + e^{-j(b_q + x^\top w_q) - \frac{1}{2} w_q^\top P_i^{-1} w_q} \right)^\top_{q=1,..} \\
&= \sigma_{s_i} \sqrt{\frac{2\alpha}{m}} \times \theta^\top \left(\frac{1}{2} e^{-\frac{1}{2} w_q^\top P_i^{-1} w_q} \left(e^{j(b_q + x^\top w_q)} + e^{-j(b_q + x^\top w_q)} \right) \right)^\top_{q=1,\ldots,m} \\
&= \sigma_{s_i} \sqrt{2\alpha/m} \times \theta^\top \left(e^{-\frac{1}{2} w_q^\top P_i^{-1} w_q} \cos(w_q^\top x + b_q) \right)^\top_{q=1,\ldots,m} \\
&= \sigma_{s_i} \sqrt{2\alpha/m} \times \theta^\top \mathrm{diag}(e^{-\frac{1}{2} W^\top P_i^{-1} W}) \cos(W^\top x + B) \\
&= \sigma_{s_i} \theta^\top \mathrm{diag}(e^{-\frac{1}{2} W^\top P_i^{-1} W}) \phi(x).
\end{aligned}
$$

where w_q is the q-th column of W and b_q is the q-th component of B. The first approximation is due to (5.11). The second and last equalities follow from (5.10). The third equality is due to the definition of the convolved kernel: $K_i(x) \triangleq \sigma_{s_i} \mathcal{N}(x | \underline{0}, P_i^{-1})$. The fourth and third last equalities follow from the fact that $\cos(x) = \frac{1}{2}(e^{jx} + e^{-jx})$ which can be derived from the Euler's formula. The fifth equality is due to (B.1).

B.2 EP Approximation for (5.20)

Let $t_j(f_j^*) \triangleq \Phi_{\text{cdf}}((f_j(x_{*_t}) + c_j - y_{\max_j})/\sigma_{n_j})$ for $j = 1, \ldots, M$. Then, $p(f^*|y_X, C2)$ can be approximated by a multivariate Gaussian $q(f^*)$ such that each non-Gaussian factor is replaced by a Gaussian factor, that is, $t_j(f_j^*) \approx \tilde{t}_j(f_j^*) \triangleq \mathcal{N}(f_j^*|\tilde{\mu}_j, \tilde{\tau}_j)$. Let $\tilde{\mu} \triangleq (\tilde{\mu}_j)_{j=1,\ldots,M}^\top$ and $\widetilde{\Sigma}$ be a $M \times M$ diagonal matrix with $\widetilde{\Sigma}_{jj} \triangleq \tilde{\tau}_j$ for $j = 1, \ldots, M$. Then,

$$\begin{aligned} p(f^*|y_X, C2) &= \frac{1}{Z} p(f^*|y_X) \prod_{j=1}^{M} t_j(f_j^*) \approx q(f^*) \triangleq \mathcal{N}(f^*|\mu, \Sigma) \\ &= \frac{1}{Z} \mathcal{N}(f^*|\mu_0, \Sigma_0) \prod_{j=1}^{M} \mathcal{N}(f_j^*|\tilde{\mu}_j, \tilde{\tau}_j) \end{aligned} \tag{B.2}$$

where $\mu \triangleq \Sigma(\widetilde{\Sigma}^{-1}\tilde{\mu} + \Sigma_0^{-1}\mu_0)$ and $\Sigma \triangleq (\widetilde{\Sigma}^{-1} + \Sigma_0^{-1})^{-1}$ can be obtained using Gaussian identities, and μ_0 and Σ_0 are, respectively, the posterior mean vector and covariance matrix of the Gaussian predictive distribution $p(f^*|y_X)$ computed analytically using (5.5). With the multiplicative form of (B.2), EP [Minka, 2001] can be used to compute the Gaussian factors $\tilde{t}_j(f_j^*) = \mathcal{N}(f_j^*|\tilde{\mu}_j, \tilde{\tau}_j)$ for $j = 1, \ldots, M$ in (B.2). Briefly speaking, EP will start from some initial values for $(\tilde{\mu}_j, \tilde{\tau}_j)$ and iteratively refine them, as shown in next subsection.

From (B.2), the posterior distribution $p(f_i(x_{*_t})|y_X, C2)$ can be approximated by

$$\begin{aligned} p(f_i(x_{*_t})|y_X, C2) &= \int p(f^*|y_X, C2)\ \mathrm{d}f_1^* \ldots \mathrm{d}f_{i-1}^* \mathrm{d}f_{i+1}^* \ldots \mathrm{d}f_M^* \\ &\approx \int q(f^*)\ \mathrm{d}f_1^* \ldots \mathrm{d}f_{i-1}^* \mathrm{d}f_{i+1}^* \ldots \mathrm{d}f_M^* = \mathcal{N}(f_i(x_{*_t})|\mu_i, \tau_i) \end{aligned} \tag{B.3}$$

where μ_i is the i-th component of μ and τ_i is the i-th diagonal component of Σ.

B.2.1 Steps for EP approximation

EP is a procedure that starts from some initial values for the parameters $(\tilde{\mu}_j, \tilde{\tau}_j)$ of the Gaussian factors $\tilde{t}_j(f_j^*) = \mathcal{N}(f_j^*|\tilde{\mu}_j, \tilde{\tau}_j)$ for $j = 1, ..., M$ and iteratively refines these quantities. At each iteration, for every Gaussian factor $\tilde{t}_j(f_j^*)$, its contribution is removed to form the cavity distribution

$$q_{-j}(f^*) \propto q(f^*)/\tilde{t}_j(f_j^*) = \mathcal{N}(f^*|\mu_{-j}, \Sigma_{-j}) \ .$$

Then, the cavity distribution $q_{-j}(f_j^*)$ follows a Gaussian distribution $\mathcal{N}(f_j^*|\bar{\mu}_j, \bar{\tau}_j)$ with mean $\bar{\mu}_j \triangleq \bar{\tau}_j(\tau_j^{-1}\mu_j - \tilde{\tau}_j^{-1}\tilde{\mu}_j)$ and variance $\bar{\tau}_j \triangleq (\tau_j^{-1} - \tilde{\tau}_j^{-1})^{-1}$.

Let $\hat{q}(f_j^*) \triangleq \mathcal{N}(f_j^*|\hat{\mu}_j, \hat{\tau}_j) \propto q_{-j}(f_j^*)t_j(f_j^*)$ denote a new Gaussian distribution whose j-th Gaussian factor $\tilde{t}_j(f_j^*)$ is replaced by its corresponding real factor $t_j(f_j^*)$. It is well-known that when $q(f^*)$ is Gaussian, the distribution that minimizes $\mathrm{KL}(\hat{q}(f_j^*)||q(f_j^*))$ is one whose first and second moments match that of $\hat{q}(f_j^*)$. Let

$$\overline{Z}_j \triangleq \log \int \mathcal{N}(f_j^*|\bar{\mu}_j, \bar{\tau}_j) \ t_j(f_j^*) \ \mathrm{d}f_j^* \ . \tag{B.4}$$

Then, the moments can be updated to

$$\hat{\mu}_j \triangleq \bar{\mu}_j + \bar{\tau}_j \frac{\partial \overline{Z}_j}{\partial \bar{\mu}_j} \quad \text{and} \quad \hat{\tau}_j \triangleq \bar{\tau}_j - \bar{\tau}_j^2 \left(\left[\frac{\partial \overline{Z}_j}{\partial \bar{\mu}_j} \right]^2 - 2 \frac{\partial \overline{Z}_j}{\partial \bar{\tau}_j} \right). \tag{B.5}$$

The parameters of the Gaussian factor $\tilde{t}_j(f_j^*) = \mathcal{N}(f_j^*|\tilde{\mu}_j, \tilde{\tau}_j)$ can be computed with

$$\tilde{\mu}_j = \tilde{\tau}_j(\hat{\tau}_j^{-1}\hat{\mu}_j - \bar{\tau}_j^{-1}\bar{\mu}_j) \quad \text{and} \quad \tilde{\tau}_j = (\hat{\tau}_j^{-1} - \bar{\tau}_j^{-1})^{-1} \ . \tag{B.6}$$

By applying the results in Appendix B.2 in [Hernández-Lobato *et al.*, 2014] to (B.4), (B.5),

and (B.6), the parameters of $\tilde{t}_j(f_j^*)$ can be refined to

$$\tilde{\mu}_j = \bar{\mu}_j + \kappa^{-1} \quad \text{and} \quad \tilde{\tau}_j = \beta^{-1} - \bar{\tau}_j$$

where

$$\alpha \triangleq \frac{\bar{\mu}_j + c_j - y_{\max_j}}{\sqrt{\bar{\tau}_j + \sigma_{n_j}^2}}, \beta \triangleq \frac{\phi(\alpha)}{\Phi_{\text{cdf}}(\alpha)} \left[\frac{\phi(\alpha)}{\Phi_{\text{cdf}}(\alpha)} + \alpha \right] \frac{1}{\bar{\tau}_j + \sigma_{n_j}^2}, \text{and } \kappa \triangleq \left[\frac{\phi(\alpha)}{\Phi_{\text{cdf}}(\alpha)} + \alpha \right] \frac{1}{\sqrt{\bar{\tau}_j + \sigma_{n_j}^2}}$$

for $j = 1, \ldots, M$.

B.3 Derivation of Posterior Distribution $p(f^+|y_X, C2)$

Let $X^\dagger \triangleq X \cup \{\langle x_{*_t}, i\rangle\}$,

$$p(f^+|y_X, C2) = p(f_i(x)|y_X, f_i^*)\ p(f_i^*|y_X, C2) = \mathcal{N}(f^+|\mu^+, \Sigma^+) \tag{B.7}$$

with posterior mean vector $\mu^+ \triangleq [\mu_i; \Psi[y_X; \mu_i]]$ and covariance matrix

$$\Sigma^+ \triangleq \begin{bmatrix} \tau_i & \tau_i\psi \\ \psi\tau_i & \sigma^2_{\langle x,i\rangle|X^\dagger} + \psi^2\tau_i \end{bmatrix}$$

where $\Psi \triangleq \Sigma_{\{\langle x,i\rangle\}X^\dagger}\Sigma^{-1}_{X^\dagger X^\dagger}$ and ψ is the last component of Ψ. Next, we will give the derivation of μ^+ and Σ^+.

First, the following lemma is needed.

Lemma 6. *Let a, b, c be three random vectors with dimension n_a, n_b, n_c and*

$$p(a|c) = \mathcal{N}(a|\mu_a, \Sigma_a)$$

$$p(b|a,c) = \mathcal{N}(b|\mu_{b|a,c}, \Sigma_{b|a,c})$$

where $\mu_{b|a,c} \triangleq M_1 a + M_2 c + s = [M_1, M_2][a; c] + s$. *Then, the conditional joint distribution of* a *and* b *given* c *is*

$$p(a,b|c) = \mathcal{N}([a;b]|\mu_{a,b|c}, \Sigma_{a,b|c})$$

where

$$\mu_{a,b|c} \triangleq \begin{bmatrix} \mu_a \\ [M_1, M_2][\mu_a; c] + s \end{bmatrix} \quad \textit{and} \quad \Sigma_{a,b|c} \triangleq \begin{bmatrix} \Sigma_a & \Sigma_a M_1^\top \\ M_1\Sigma_a & \Sigma_{b|a,c} + M_1\Sigma_a M_1^\top \end{bmatrix}.$$

Proof. From the definition of multivariate Gaussian distribution,

$$p(a,b|c) = p(a|c)\; p(b|a,c) = \frac{(2\pi)^{-(n_a+n_b)/2}}{\sqrt{|\Sigma_{b|a,c}||\Sigma_a|}} e^{-\frac{1}{2}E} \tag{B.8}$$

where $E \triangleq (b - \mu_{b|a,c})^\top \Sigma_{b|a,c}^{-1}(b - \mu_{b|a,c}) + (a - \mu_a)^\top \Sigma_a^{-1}(a - \mu_a)$.

Let $f \triangleq b - M_1\mu_a - M_2 c - s$ and $e \triangleq a - \mu_a$. Then,

$$\begin{aligned} E &= (b - M_1 a - M_2 c - s)^\top \Sigma_{b|a,c}^{-1}(b - M_1 a - M_2 c - s) + (a - \mu_a)^\top \Sigma_a^{-1}(a - \mu_a) \\ &= (f - M_1 e)^\top \Sigma_{b|a,c}^{-1}(f - M_1 e) + e^\top \Sigma_a^{-1} e \\ &= \begin{bmatrix} a - \mu_a \\ b - M_1\mu_a - M_2 c - s \end{bmatrix}^\top R^{-1} \begin{bmatrix} a - \mu_a \\ b - M_1\mu_a - M_2 c - s \end{bmatrix} \end{aligned} \tag{B.9}$$

where

$$R = \begin{bmatrix} M_1^\top \Sigma_{b|a,c}^{-1} M_1 + \Sigma_a^{-1} & -M_1^\top \Sigma_{b|a,c}^{-1} \\ -\Sigma_{b|a,c}^{-1} M_1 & \Sigma_{b|a,c}^{-1} \end{bmatrix}^{-1} = \begin{bmatrix} \Sigma_a & \Sigma_a M_1^\top \\ M_1\Sigma_a & \Sigma_{b|a,c} + M_1\Sigma_a M_1^\top \end{bmatrix}.$$

The last equality of (B.9) can be computed from equation 50 in [Schön and Lindsten, 2011] and the second equality of R is due to equation 9d in [Schön and Lindsten, 2011]. Also,

$$\frac{1}{|R|} = \frac{1}{|\Sigma_a||\Sigma_{b|a,c}|}$$

due to equation 51 in [Schön and Lindsten, 2011]. Therefore, (B.8) can be written as

$$\begin{aligned}
&p(a,b|c) \\
&= \frac{(2\pi)^{-(n_a+n_b)/2}}{\sqrt{|R|}} \exp\left(-\frac{1}{2}\begin{bmatrix} a-\mu_a \\ b-M_1\mu_a-M_2c-s \end{bmatrix}^\top R^{-1} \begin{bmatrix} a-\mu_a \\ b-M_1\mu_a-M_2c-s \end{bmatrix}\right) \\
&= \mathcal{N}\left([a;b]\middle|\begin{bmatrix} \mu_a \\ [M_1,M_2][\mu_a;c]+s \end{bmatrix}, R\right).
\end{aligned} \tag{B.10}$$

□

Then, in (B.7), we know that $p(f_i(x)|y_X, f_i^*) = \mathcal{N}(f_i(x)|\mu_{\langle x,i\rangle|X^\dagger}, \sigma^2_{\langle x,i\rangle|X^\dagger})$ with $\mu_{\langle x,i\rangle|X^\dagger} \triangleq \Sigma_{\{\langle x,i\rangle\}X^\dagger}\Sigma^{-1}_{X^\dagger X^\dagger}[y_X; f_i^*]$ and $p(f_i^*|y_X, C2) = \mathcal{N}(f_i^*|\mu_i, \tau_i)$ (B.3). Therefore, (B.7) can be easily obtained by replacing a, b, and c in Lemma 6 with f_i^*, $f_i(x)$, and y_X, respectively.

B.4 Derivation of Posterior Covariance Matrix in (5.23)

Let $r \triangleq a^\top f^+$. From (B.7) and (5.22),

$$\begin{aligned}
Z' &= \int \mathcal{N}(f^+|\mu^+, \Sigma^+)\ \mathbb{I}(f_i(x) - f_i(x_{*_t}) \le \delta_i c_i)\ \mathrm{d}f^+ \\
&= \int \mathcal{N}(r|\eta, v)\ \mathbb{I}(r \le \delta_i c_i)\ \mathrm{d}r = \Phi_{\mathrm{cdf}}\left(\frac{\delta_i c_i - \eta}{\sqrt{v}}\right).
\end{aligned} \tag{B.11}$$

Let $\overline{Z}' \triangleq \log Z'$. Then, the derivative of $\overline{Z}'$ with respect to the posterior mean vector μ^+ and covariance matrix Σ^+ can be computed as follows:

$$\frac{\partial \overline{Z}'}{\partial \mu^+} = \frac{\partial \overline{Z}'}{\partial \eta}\frac{\partial \eta}{\partial \mu^+} = \frac{1}{\Phi_{\text{cdf}}((\delta_i c_i - \eta)/\sqrt{v})}\ \phi\left(\frac{\delta_i c_i - \eta}{\sqrt{v}}\right)\left(-\frac{1}{\sqrt{v}}\right) a = -\frac{\gamma}{\sqrt{v}}a$$

$$\frac{\partial \overline{Z}'}{\partial \Sigma^+} = \frac{\partial \overline{Z}'}{\partial v}\frac{\partial v}{\partial \Sigma^+} = \frac{1}{\Phi_{\text{cdf}}((\delta_i c_i - \eta)/\sqrt{v})}\ \phi\left(\frac{\delta_i c_i - \eta}{\sqrt{v}}\right)\frac{\eta - \delta_i c_i}{2v\sqrt{v}} aa^\top = \frac{\gamma(\eta - \delta_i c_i)}{2v\sqrt{v}} aa^\top.$$

Then,

$$\begin{aligned}
\Sigma_{f^+} &= \Sigma^+ - \Sigma^+ \left(\left[\frac{\partial \overline{Z}'}{\partial \mu^+}\right]\left[\frac{\partial \overline{Z}'}{\partial \mu^+}\right]^\top - 2\frac{\partial \overline{Z}'}{\partial \Sigma^+}\right)\Sigma^+ \\
&= \Sigma^+ - \Sigma^+ \left(\frac{\gamma^2}{v} aa^\top - \frac{\gamma(\eta - \delta_i c_i)}{v\sqrt{v}} aa^\top\right)\Sigma^+ \\
&= \Sigma^+ - \frac{\gamma}{v}\left(\gamma - \frac{\eta - \delta_i c_i}{\sqrt{v}}\right)\Sigma^+ aa^\top \Sigma^+.
\end{aligned} \tag{B.12}$$

The first equality is due to (B.5).

B.5 Generalizing to Multiple Latent Functions

B.5.1 CMOGP with multiple latent functions

Let $\{L_q(x)\}_{q=1,\ldots,Q}$ denote a set of Q independent latent functions. Then, CMOGP defines each i-th function f_i as

$$f_i(x) \triangleq \sum_{q=1}^{Q} \int_{x' \in D} K_{iq}(x - x')\ L_q(x')\ \mathrm{d}x'\ . \tag{B.13}$$

Similar to CMOGP with only one latent function, the work of [Álvarez and Lawrence, 2011] has shown that if every $\{L_q(x)\}_{x \in D}$ is an independent GP for $q = 1, \ldots, Q$, then $\{f_i(x)\}_{\langle x,i\rangle \in D^+}$ is also a GP. Specifically, let $\{L_q(x)\}_{x \in D}$ be a GP with prior covariance

$\sigma^q_{xx'} \triangleq \mathcal{N}(x - x'|\underline{0}, P_{0q}^{-1})$ and $K_{iq}(x) \triangleq \sigma_{s_i q}\mathcal{N}(x|\underline{0}, P_i^{-1})$. Then,

$$\sigma_{\langle x,i\rangle\langle x',j\rangle} = \sum_{q=1}^{Q} \sigma_{s_i q}\sigma_{s_j q}\mathcal{N}(x - x'|\underline{0}, P_{0q}^{-1} + P_i^{-1} + P_j^{-1}) \ . \tag{B.14}$$

The Gaussian predictive distribution in (5.5) and the subsequent results in Section 5.3 related to CMOGP remain valid by computing its posterior covariance matrix with (B.14) instead of (5.3).

Similar to that in Section 5.1, the fidelity of an auxiliary function f_i with respect to target function f_t can be computed using (B.14) as

$$\rho_i \triangleq \sigma_{\langle x_{*_i},i\rangle\langle x_{*_t},t\rangle}/(\sigma'_{s_i}\sigma'_{s_t}) \tag{B.15}$$

where $\sigma'_{s_i} \triangleq \left(\sum_{q=1}^{Q} \sigma^2_{s_i q}/(2\pi|P_{0q}^{-1} + 2P_i^{-1}|)^{1/2}\right)^{1/2}$. Note that (B.15) reduces to (5.4) when $Q = 1$.

B.5.2 MRF approximation with multiple latent functions

In this subsection, we will extend the MRF approximation described in Section 5.3.1 to approximate the CMOGP model with multiple latent functions.

Similar to that in Section 5.3.1, the covariance function of the GP modeling L_q can be written as

$$\begin{aligned}\sigma^q_{xx'} &= \alpha_q \int p(w_q)\ e^{-jw_q^\top(x-x')}\ \mathrm{d}w_q \\ &= 2\alpha_q\ \mathbb{E}_{p(w_q,b_q)}[\cos(w_q^\top x + b_q)\cos(w_q^\top x' + b_q)]\end{aligned}$$

where $p(w_q) \triangleq s(w_q)/\alpha_q$, $s(w_q)$ is the Fourier dual of $\sigma^q_{xx'}$, and $b_q \sim \mathcal{U}[0, 2\pi]$. Then,

each latent function L_q can be approximated by a linear model:

$$L_q(x) \approx \phi_q(x)^\top \theta_q \tag{B.16}$$

where $\phi_q(x) \triangleq \sqrt{2\alpha_q/m}\ \cos(W_q^\top x + B_q)$ for $q = 1, \ldots, Q$, and W_q and B_q consist of m stacked samples from $p(w_q)$ and $p(b_q)$, respectively. Let

$$f_{iq}(x) \triangleq \int_{x' \in D} K_{iq}(x - x')\ L_q(x')\ \mathrm{d}x'\ . \tag{B.17}$$

Then,

$$f_i(x) = \sum_{q=1}^{Q} f_{iq}(x) = \sum_{q=1}^{Q} \phi_{iq}(x)^\top \theta_q = \Phi_i(x)^\top \theta \tag{B.18}$$

where $\theta \triangleq (\theta_q^\top)_{q=1,\ldots,Q}^\top$, $\Phi_i(x) \triangleq (\phi_{iq}(x)^\top)_{q=1,\ldots,Q}^\top$, and

$$\phi_{iq}(x) \triangleq \sigma_{s_i q}\ \mathrm{diag}(e^{-\frac{1}{2} W_q^\top P_i^{-1} W_q})\ \phi_q(x)$$

can be interpreted as the input features of function $f_i(x)$ corresponding to the latent function $L_q(x)$. The first equality is due to (B.13) and (B.17). The second equality is due to (5.12), (B.16), and (B.17).

Since (B.18) has exactly the same form as (5.12), all the results in Section 5.3.1 will remain valid for MRF approximation with multiple latent functions.

B.6 Details of the Benchmark Functions

Let $x_{(i)}$ be the i-th component of an input x. The following benchmark functions are used in our experiments:

Hartmann-6D function. $D \triangleq [0,1]^6$, $f_i(x) \triangleq \sum_{j=1}^{4} \beta_j^{(i)} \exp(\sum_{k=1}^{6} A_{jk}(x_{(k)} - P_{jk}))$ where $A, P \in \mathbb{R}^{4\times 6}$ are fixed matrices:

$$A \triangleq \begin{bmatrix} 10 & 3 & 17 & 3.5 & 1.7 & 8 \\ 0.05 & 10 & 17 & 0.1 & 8 & 14 \\ 3 & 3.5 & 1.7 & 10 & 17 & 8 \\ 17 & 8 & 0.05 & 10 & 0.1 & 14 \end{bmatrix},$$

$$P \triangleq 10^{-4} \times \begin{bmatrix} 1312 & 1696 & 5569 & 124 & 8283 & 5886 \\ 2329 & 4135 & 8307 & 3736 & 1004 & 9991 \\ 2348 & 1451 & 3522 & 2883 & 3047 & 6650 \\ 4047 & 8828 & 8732 & 5743 & 1091 & 381 \end{bmatrix}$$

and $\beta_j^{(i)}$ is the j-th component of the vector $\beta^{(i)}$ which varies for different functions, as shown in Table B.1 below.

Name	Function expression	$\sigma_{n_i}^2$	Degree ρ_i of fidelity
target	$f_1(x)$ with $\beta^{(1)} \triangleq [1.0, 1.2, 3.0, 3.2]$	10^{-3}	1
func1	$f_i(x)$ with $\beta^{(i)} \triangleq \beta^{(1)} + [0.1, -0.1, -0.01, 0.01]$	10^{-4}	0.9995
func2	$f_i(x)$ with $\beta^{(i)} \triangleq \beta^{(1)} + [1, 1, -1, -1]$	10^{-4}	0.8759
func3	$f_i(x)$ with $\beta^{(i)} \triangleq \beta^{(1)} + [4, 4, -4, -4]$	10^{-4}	0.0037

Table B.1: The target and 3 different auxiliary functions for Hartmann-6D. The fidelity ρ_i of the auxiliary function is calculated using the generalized expression (B.15) for multiple latent functions.

Branin-Hoo function. For all the target and auxiliary functions, $D \triangleq [0, 1]^2$. The noise variance $\sigma_{n_i}^2$ is set to be 10^{-3} and 10^{-4} for the outputs of Branin function (i.e., target) and auxiliary function (i.e., func1, func2 or func3), respectively.

Name	Function expression	Degree ρ_i of fidelity
target	$f_1(x) \triangleq -\frac{1}{51.95}\left[\left(\bar{x}_2 - \frac{5.1\bar{x}_1^2}{4\pi^2} + \frac{5\bar{x}_1}{\pi} - 6\right)^2 + \left(10 - \frac{10}{8\pi}\right)\cos(\bar{x}_1) - 44.81\right]$ where $\bar{x}_1 \triangleq 15x_{(1)} - 5$, $\bar{x}_2 \triangleq 15x_{(2)}$	1
func1	$f_2(x) \triangleq f_1(x)$	0.9997
func2	$f_2(x) \triangleq -\frac{1}{51.95}\left[\left(\bar{x}_2 - \frac{5.1\bar{x}_1^2}{4\pi^2} + \frac{5\bar{x}_1}{\pi} - 6\right)^2 + \left(10 - \frac{10}{8\pi}\right)\cos(\bar{x}_1) - 44.81\right]$ where $\bar{x}_1 \triangleq 15(x_{(1)} + 0.01) - 5$, $\bar{x}_2 \triangleq 15(x_{(2)} + 0.01)$	0.9992
func3	$f_2(x) \triangleq \left(1 - \exp(\frac{-1}{2x_{(2)}})\right)\left(\frac{2300x_{(1)}^3 + 1900x_{(1)}^2 + 2092x_{(1)} + 60}{100x_{(1)}^3 + 500x_{(1)}^2 + 4x_{(1)} + 20}\right)$ (i.e., Currin exponential function)	0.0683

Table B.2: The target and 3 different auxiliary functions for Branin-Hoo. The fidelity ρ_i of the auxiliary function is calculated using the generalized expression (B.15) for multiple latent functions. The fidelity of auxiliary function func1 is not exactly 1 due to the trained hyperparameters $P_1 \neq P_2$ because the limited training data/observations gathered from evaluating the target and auxiliary functions are corrupted by different noises and correspond to different sets of inputs such that the trained CMOGP model cannot achieve the true cross-correlation between these functions.

Appendix C

Appendix of Chapter 6

C.1 Derivation of (6.13)

Let $r \triangleq a^\top f^*$. The normalization term Z in (6.12) can be computed as

$$\begin{aligned} Z &= \int \mathcal{N}(f_*|\mu, \Sigma)\Phi_{\text{cdf}}(\frac{a^\top f_* - y_{\max}}{\sigma_n})\mathrm{d}f_* \\ &= \int \mathcal{N}(r|m, v)\Phi_{\text{cdf}}(\frac{r - y_{\max}}{\sigma_n})\mathrm{d}r = \Phi(t) \end{aligned}$$

The last equality is due to equation (3.82) in [Rasmussen and Williams, 2006]. Let $Z' \triangleq \log Z$. Then, the derivative of Z' with respect to the posterior mean vector μ and covariance matrix Σ can be computed as follows:

$$\frac{\partial Z'}{\partial \mu} = \frac{\partial Z'}{\partial m}\frac{\partial m}{\partial \mu} = \frac{1}{\Phi_{\text{cdf}}(t)}\phi_{\text{pdf}}(t)\frac{1}{\sqrt{\sigma_n^2 + v}}a = \frac{\gamma a}{\sqrt{\sigma_n^2 + v}}$$

$$\frac{\partial Z'}{\partial \Sigma} = \frac{\partial Z'}{\partial v}\frac{\partial v}{\partial \Sigma} = \frac{1}{\Phi_{\text{cdf}}(t)}\phi_{\text{pdf}}(t)(m - y_{\max})(-\frac{1}{2})(\sigma_n^2 + v)^{-\frac{3}{2}}aa^\top = -\frac{\gamma t}{2(\sigma_n^2 + v)}aa^\top$$

where $\gamma \triangleq \phi_{\text{pdf}}(t)/\Phi_{\text{cdf}}(t)$. Then, using the result in (B.5), we can approximate the distribution of $p(f_*^{(1)}, ..., f_*^{(C)}|y_X, C2)$ as a multivariate Gaussian with the following

posterior mean vector and covariance matrix:

$$\begin{aligned}
\mu' &= \mu + \Sigma\frac{\partial Z'}{\partial \mu} = \mu + \frac{\gamma}{\sqrt{\sigma_n^2 + v}}\Sigma a \\
\Sigma' &= \Sigma - \Sigma\left(\left[\frac{\partial Z'}{\partial \mu}\right]\left[\frac{\partial Z'}{\partial \mu}\right]^\top - 2\frac{\partial Z'}{\partial \Sigma}\right)\Sigma \\
&= \Sigma - \Sigma\left(\frac{\gamma^2}{\sigma_n^2 + v}aa^\top + \frac{\gamma t}{\sigma_n^2 + v}aa^\top\right)\Sigma \\
&= \Sigma - \frac{\gamma^2 + \gamma t}{\sigma_n^2 + v}\Sigma aa^\top\Sigma
\end{aligned}$$