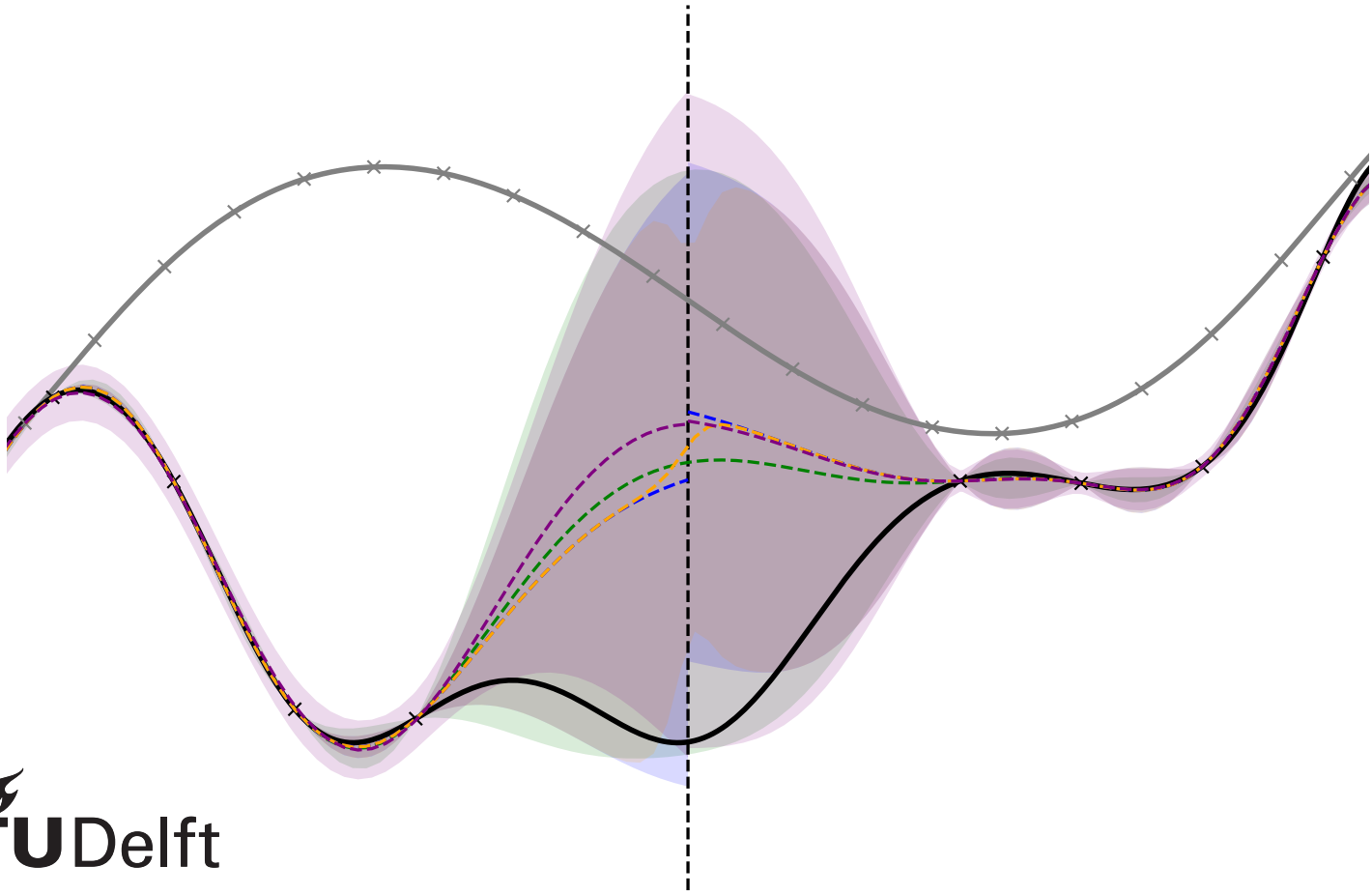
Splitting and Stitching Gaussian Process Regression Models with Multi-fidelity

H. Hendriks



TU Delft

Splitting and Stitching Gaussian Process Regression Models with Multi-fidelity

by

H. Hendriks

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Monday May 8, 2023 at 15:00.

Student number:	4899512
Project duration:	February, 2021 – April, 2023
Thesis committee:	Dr.Ir. F.P. (Frans) van der Meer
	M.A. (Miguel) Bessa PhD.
	Dr. I. (Iuri) B.C.M. Rocha
	O.T. (Taylan) Turan MSc.

An electronic version of this thesis is available at http://repository.tudelft.nl/.

TU Delft

Abstract

The increasing demand for sustainable energy, results in more wind turbines being built offshore. The blades of wind turbines consist of composites which makes them difficult to design. Because composites consist of a micro- and macro-structure of which their interplay determines the global behavior under loads. Therefore, traditional methods of analysis are often infeasible: full-scale experiments take too long and are too costly, small-scale experiments can not capture the interaction between the micro- and macro-structure, and analytical theories are often only tractable for simple cases.

FE^2 is one such method that can analyze the behavior of multi-scale materials, such as composites. It consists of a macro finite element model where the constitutive behavior of each integration point is obtained by homogenizing representative volume elements (RVE) that represent the microstructure. Although this method is capable of accurately predicting composites, the computational cost is high, as for each integration point another finite element model must be computed.

This process can be sped up by replacing the RVE with a surrogate that is more efficient to compute. Recently, Gaussian Process Regression (GPR), a probabilistic machine learning model, is used as a surrogate for the RVEs. It uses prior knowledge and observations of the RVE to make its predictions. It is based on Gaussian Processes, meaning that the predictions exhibit a multivariate Gaussian distribution which provides an uncertainty measure besides its prediction. This is useful in determining where new observations must be collected from the RVE.

To fully capture the RVE a lot of observations are needed from it. This is still a computational bottleneck for the surrogate as they are expensive to compute. The GPR model can be enhanced with observations from a low-fidelity model, for example, linear elastic, which is called GPR with multi-fidelity. The low-fidelity observations, inexpensive and inaccurate, enhance the prediction of the high-fidelity model, which uses high-fidelity observations that are expensive but accurate. This is shown to decrease the number of observations needed for the high-fidelity model. Thus, fewer observations need to be collected from the RVE. However, the correlation between the low- and high-fidelity is often assumed to be constant in these models. This presents a problem as the correlation in the case of surrogate modeling is often non-linear. The literature investigates several extensions of the GPR with multi-fidelity that assume a non-linear correlation, but these models become more complex and lose their simplicity as their predictions are not Gaussian anymore.

Instead, this thesis keeps the constant correlation assumption but improves the correlation inference by splitting the model. Splitting means that multiple independent GPR with multi-fidelity models are used in different regions. As splitting results in discontinuous predictions, the thesis also investigates two stitching methods to remove these discontinuities. The first, the constrained boundary (CB) method, constrains the predictive mean and predictive variance of to neighboring models to be equal at their respective boundaries. These constraints to the optimization procedure of the hyperparameters of the local GPR with multi-fidelity models. The second stitching method, the random variable mixture (RVM) uses a weighing procedure based on a mixture of experts approach. However, this method mixes the random variables instead of the probability density distributions. This creates a much simpler and analytically tractable method. For both of these stitching methods, only the high-fidelity uses the stitching procedure while the low-fidelity is split.

This thesis aims to make the first step in using splitting and stitching of GPRs with multi-fidelity in the correlation inference of non-linear correlations. Therefore, these methods are tested on synthetic one-dimensional datasets with two regions that are pre-determined. This provides a simple procedure to check if this approach is viable and to see under which correlation conditions it improves the prediction accuracy. The methods are tested on three different cases each having a different correlation: constant, discontinuous, and linearly varying. Where the first assumes a linear relation, while the other two have a non-linear relation between the fidelities. The prediction accuracy is investigated in both the interpolation and extrapolation regime, and it is measured across different numbers of low- and high-fidelity observations and two different sampling strategies for the inputs: linearly spaced

and sampled from a uniform distribution. This is to see if the trends of the results hold under different dataset conditions.

The results of the experiments are as follows: under the constant case, splitting and stitching decrease the performance, because it loses the global spatial correlation. RVM performs slightly better than splitting, while CB performs worse. Under the discontinuous case, splitting and stitching improve the performance, because it is able to capture the non-linear correlation, where RVM is slightly worse compared to splitting as the underlying function is discontinuous and it makes it continuous. However, CB performs worse than not splitting. Under the linearly varying case, all methods perform almost similarly in the interpolation regime, but splitting and stitching perform slightly better in the extrapolation regime, while CB is a bit worse. Overall, in non-linear cases splitting and stitching might be beneficial in capturing the non-linear correlation compared to not doing it. These trends hold in the constant case when the number of observations is changed and under the two different sampling strategies.

Thus, this thesis takes the first step in using splitting and stitching to capture the non-linear correlation between fidelities. This can ultimately help to reduce the computational cost of surrogates of the RVEs of the FE^2 method, which in turn enhances the design of structures built out of composites, such as wind turbines, by speeding up the mechanical analysis. However, the splitting and stitching methods must undergo some changes before they can be applied as surrogates. The methods must be changed to handle an input space that has a dimension higher than one, it must be able to handle more than two local models, and the clustering of the regions of the local models must be embedded in these methods.

Acknowledgements

In this section, I would like to thank a few people who helped me during the MSc thesis.

First, I would like to thank my supervisors for the helpful feedback and support, Frans van der Meer, Miguel Bessa, Iuri Rocha, and Taylan Turan. Special thanks to Iuri and Taylan, who helped me immensely with their feedback and conversations during the weekly meetings.

I want to thank my friends, Mark and Esther for their support and good conversations, and for helping me to improve and practice my final presentation.

I want to thank, Sijmen, for the good and helpful conversations at the TU during our coffee breaks.

Last, I want to thank my family, Perry, Geraldine, Jelle, and Evelien, for their support and good conversations, especially during the time that I had a broken leg.

H. Hendriks

Den Haag, May 2023

Contents

1 Introduction

1.1. Motivation and Background

Composite materials are a combination of two or more materials that outperform the constituents in isolation [39]. For instance, fiber-reinforced composite materials are lightweight and have a high strength-to-weight ratio by leveraging the properties of fibers and a matrix material [36]. These attributes make them useful in constructions that are susceptible to weight-dependent fatigue damage, such as wind turbine blades [40]. The behavior of such structures is influenced by characteristics at multiple scales: the design of the construction (macro-scale) and the physical and chemical processes in the composite (micro-scale) [40]. This makes them particularly difficult to analyze with conventional methods, for example, full-scale experiments are expensive and take a long time, smaller-scale experiments cannot capture the macroscopic behavior, and analytical theories are often only tractable for simple cases. Therefore, the analysis of their behavior requires high-fidelity multi-scale numerical approaches.

FE^2 (concurrent finite element analysis) is one of such methods [10, 28, 19]. It averages micro-mechanical models into a homogeneous medium, with which it can predict at the macro-scale while accounting for micro-scale processes. Specifically, each integration point in the macro-model is another finite element model that represents the micro-scale structure resulting in a high computational cost. Several methods are investigated by the literature that replaces these costly micro-models with cheaper surrogates, for example, mesoscale models, and machine learning models such as neural networks; a brief overview of these techniques is given in [45].

More recently, Gaussian Process regression models are investigated as surrogates for constitutive micro-mechanical models [41, 45]. In this case, these models take the strains and stresses as observational data and infer the constitutive relation. The inference takes a Bayesian approach where the underlying relation is inferred in Gaussian processes, which generalizes multivariate Gaussian distributions to the functional space [52]. The model provides uncertainty information regarding its predictions due to its probabilistic nature. This is useful in detecting the extrapolation regime which is limited in providing accurate results [45]. To overcome this issue the interpolation regime can be extended by adding more observations outside its domain. However, this results in a higher computational cost of model inference and optimization that scales cubically. Also, the cost of data collection increases which is especially significant for composites because their observations must be computed with an expensive micro-mechanical model.

The number of observations could be reduced by extending Gaussian process regression models with multi-fidelity information [17, 21]. It uses extra observations from a low-fidelity (inexpensive and inaccurate) model to inform the functional relation of the high-fidelity (expensive and accurate) model, in our case the micro-mechanical model, by inferring a correlation between them. In literature, the correlation is often assumed to be linear and defined using a constant correlation coefficient [17, 21, 45]. These models are able to reduce the number of high-fidelity observations while approximately maintaining the same prediction accuracy [17, 20], and they improve the predictions in the high-fidelity extrapolation regime (only low-fidelity observations are present) [45]. Several studies use these meth-

ods as a surrogate of physical models to provide accurate predictions [34, 20, 45]. However, in cases where the underlying correlation is highly non-linear or weakly correlated and enough high-fidelity observations are available, a non-linear correlation assumption outperforms the linear one [1]. For example, when considering the prediction of a micro-mechanical model, the different loading cases often have different correlations between the fidelities. Therefore, when a prediction is needed from a particular loading case, it is often better with linearly correlated multi-fidelity GPR models to use one that is only trained on that particular case instead of using one that is trained on all cases [45].

In literature, multi-fidelity GPR methods with a non-linear correlation assumption between successive fidelities are investigated [1, 35, 6, 13]. Consequently, these models become more complex, which means that more observations are needed, and often Monte Carlo-based inference is used because the predictive distribution of the fidelities is not Gaussian. Therefore, the computational cost is increased significantly. This makes it interesting to investigate methods that are capable of inferring non-linear correlated data that are simpler and less expensive.

1.2. Research Questions

This thesis investigates the correlation coefficient inference by splitting the Gaussian process regression model with multi-fidelity into multiple local models, each active in a disjoint region of the input space; thus, creating a model that has a piece-wise linear correlation. This provides an alternative to the models with a non-linear correlation that is simpler and has a reduced computational cost due to the division of the input space. The act of splitting creates discontinuities in the predictions at the boundaries between the local models. Therefore, this thesis investigates two methods that remove them by stitching the local models: adding continuity constraints and taking a weighted sum of the local predictions.

Thus, the following research questions are investigated regarding the splitting and stitching of Gaussian process regression models with multi-fidelity:

- **RQ1**: What is the effect of splitting and stitching on the prediction accuracy of GPR methods with multi-fidelity in a linear and non-linear correlated setting in the interpolation and extrapolation regime?
- **RQ2**: What is the influence of the ratio of the number of low and high-fidelity observations on the prediction accuracy of splitting and stitching methods?
- **RQ3**: What is the influence of the sampling strategy of the input space on the prediction accuracy of splitting and stitching methods?

1.3. Scope

This thesis aims to investigate the effect of splitting and stitching GPRs with multi-fidelity with a linear correlation assumption on the correlation inference of different correlated settings. Therefore, the experiments use synthetic datasets, as this allows for precise control of the correlation, such that different correlated settings can be tested.

To keep things simple, the datasets only have one-dimensional inputs and targets, and each method only uses two local models with their regions being predetermined.

Due to time constraints, the methods are not tested on real datasets and are not compared to GPR with multi-fidelity methods that assume a non-linear correlation between their fidelities.

1.4. Outline

The thesis is organized into the following chapters:

- Chapter 2 discusses the literature on Gaussian process regression, local approximation methods (splitting and stitching), GPR with non-linearly correlated multi-fidelity, and the differentiating factor of this thesis from the literature that also combines local approximation methods with GPR with linearly correlated multi-fidelity.

- Chapter 3 mathematically defines the Gaussian process regression models with and without linearly correlated multi-fidelity, and its split versions, it defines and discusses the origin of the two stitching methods, and it discusses how stitching methods are incorporated in the GPR model with multi-fidelity.
- Chapter 4 presents the methodology by specifying the methods under investigation, the cases on which the methods are evaluated, the dataset creation of each case, and how the performance of each method is measured on the datasets.
- Chapter 5 presents and discusses the results of the experiments regarding the splitting and stitching of Gaussian process regression models with linearly correlated multi-fidelity in linear- and non-linear correlation settings.
- Chapter 6 presents the conclusions of this thesis and provides points for future work.

1.5. Mathematical Notation

This thesis adopts the matrix notation: scalars are represented with lower-case letters, e.g., a, vectors are represented with bold-faced lower-case letters, e.g., $\mathbf{v}$, and matrices are represented with bold-faced upper-case letters, e.g., $\mathbf{I}$. The absolute value of a scalar a is denoted as $|a|$ and the euclidean norm of a vector $\mathbf{v}$ is denoted as $\|\mathbf{v}\|$.

2

Literature Review

This thesis investigates a different approach to modeling Gaussian process regression with multi-fidelity. It aims to infer the underlying function of multi-fidelity datasets with non-linear correlations by using local approximation methods instead of increasing the complexity of the correlation assumption between successive fidelities. Therefore, this literature review starts by giving an overview of the field of Gaussian process regression, discusses the most important local approximation methods, introduces GPR with multi-fidelity, discusses its extensions that can handle non-linearly correlated multi-fidelity datasets, and states what this thesis contributes to the field.

2.1. Gaussian Process Regression

Gaussian process regression (GPR) is a machine learning method based on Bayesian inference that uses Gaussian processes, which generalize multivariate Gaussian distributions to the functional space [52]. The method incorporates prior information through the Bayesian formalism, meaning that it expresses a prior belief about the underlying relation of the observations. This provides a natural way of incorporating domain knowledge into the model. As this regression is in a probabilistic framework: predictions are probability distributions. With this, not only are predictions made by calculating its mean, but the variance of this distribution provides a measure for the uncertainty of its prediction [52]. This method is also more robust against overfitting as it balances the fit against the model's complexity.

It became popular in the fields of geostatistics [27, 16] - where it is called Kriging - and meteorology [47, 5], after which its potential was realized in general regression [52]. Currently, it is applied to a variety of tasks, for example, optimization [9] and surrogate modeling [45, 41]. Even though it is widely used, its most prominent shortcoming is its cubic computational cost. This makes it highly inefficient for large-scale datasets [24, 25]. This cost is associated with the non-parametric nature of the GPR method. Unlike neural networks, it operates on the complete dataset when making predictions instead of discarding them after the method is fit.

2.2. Local Approximations with GPR

One way to reduce this large computational cost is to make use of local approximation methods [24, 25]. They partition the input space and assign a local GPR model to each region, distributing the optimization and prediction across multiple models; essentially splitting the model. Most often, the partitioning and the training of the local models is separated. For this, several non-overlapping partition schemes are used in literature, for example, Voronoi tesselation [18] and trees [11, 12]. Aside from the reduced cost, they also exhibit the ability to capture non-stationary features [24, 25]. However, these methods often ignore global patterns, are prone to overfitting, as they have fewer observations, and their predictions are discontinuous at the partition's boundaries [24, 25].

In a line of three papers, Park et al. [32, 31, 30] developed three methods that alleviate the discon-

tinuity problem by constraining the local models to be equal at their boundaries. In their first paper [32], they reformulated local GPR models into an equivalent optimization problem with added continuity constraints for the predictive mean in finite points at the boundary. This method is shown to sometimes result in negative variances due to issues with numerical instabilities. They improved upon this method by transforming the optimization problem to a variational one and solving it with the finite element method [31]. This resulted in a more numerically stable method and allowed continuity to be imposed across the whole boundary. In their third paper [30], they stopped using the equivalent optimization approach and opted to put the constraints as pseudo-observations into the standard GPR framework. Compared to the previous two, this method is mathematically simpler and improves the accuracy of the predictive variance, while having comparable and sometimes even better computational efficiency and accuracy of the predictive mean [30]. One major downside of these three methods is that they are only applicable to low-dimensional problems as the number of constraints increases significantly with the dimensionality of the input space [25]. These are the only approaches known, to the writer of this thesis, to enforce continuity between the local models by adding constraints. Although there are more methods to enforce constraints on GPR [44], they have yet to be used to stitch local models together.

Instead of constraining the local models, the other approach to make local approximation methods continuous is by means of model averaging [25]. Product-of-experts (PoE) is one such method that aggregates the local models by multiplying their probability distributions [15]. When using GPRs as the experts, the inference becomes analytically tractible as multiplying Gaussians results in a Gaussian. This method produces a poor predictive mean and an overconfident predictive variance when increasing the number of local models due to weak experts contributing equally [23]. Therefore, the generalized PoE (GPoE) [3] is developed which weakens the local models in areas where their predictions are poor using weights in the multiplication. However, this produces explosive prediction variance in the extrapolation regime [23]. This issue can be addressed by imposing that the sum of the weights is equal to one [3, 7].

The performance of PoE is improved by imposing a conditional independence assumption, creating a method called the Bayesian committee machine (BCM) [48]. Although this assumption helps to recover the prior in the extrapolation regime [24, 25], it requires that all local models share the same hyperparameters making it less favorable with non-stationary datasets [24, 25]. As with GPoE there is also a more robust version of the BCM called robust BCM [7]. It uses the same weights in a similar fashion to produce more robust predictions in the interpolation regime [24, 25].

The other important modeling averaging technique is the mixture-of-experts [53, 26], which takes a weighted sum using gating functions of the probability distribution of the local models [25]. This is in contrast with the PoE methods that use multiplication. It is used for datasets that have non-stationary features, but it comes with the cost of having an intractable inference [24]. In the original description, the gating functions are parametric. When applying this method to GPR experts, this is unfavorable as GPR is non-parametric. Therefore, Tresp [49] extended it to the non-parametric case by modeling the gating function, the mean, and the noise variance of each expert by a GP. However, this results in an exceedingly high computational cost which is not in favor of large datasets [25]. Three threads of solutions are given in the literature to reduce this cost. The first is the localization of the experts, which localizes the likelihood, see the infinite mixture of GP experts [38]. The second thread applies sparse approximation methods to the GPR experts. Last, the third thread pre-clusters the dataset and thereafter the experts are optimized. Thus, separating the clustering of the dataset and the optimization of the experts. This is denoted as a mixture of explicitly localized experts (MELE), while the former two threads that do not separate the two processes are denoted as a mixture of implicitly localized experts (MILE). MELE misses the interaction between the experts while MILE does, however, it suffers from some failed experts due to the zero-coefficient problem [25].

Nguyen-Tuong et al. [29] created a method in a similar fashion to a mixture-of-experts that can be applied to the online setting. This means that data is continuously coming and the model must be updated along the way. This favors a mathematically simpler method, therefore the predictions are weighed compared to their probability distributions; the inference becomes tractable as weighing and adding independent GPs results in a GP. However, they produce discontinuities as the weighted sum is only taken over a subset of the local models closest to the prediction input [46]. Terry et al. [46] created a similar model that uses all local models in its weighing procedure.

2.3. Gaussian Process Regression with Multi-fidelity

Gaussian process regression with multi-fidelity was first presented by Kennedy et al. [17]. This method incorporates low-fidelity data (inexpensive) with high-fidelity data (expensive) to improve the performance compared to a GPR model that only uses one of the two. Prior beliefs are placed on each fidelity in the form of a Gaussian process and a linear correlation is assumed between successive ones by multiplying the previous fidelity by a constant correlation coefficient. This assumption is also often defined as a polynomial regression that is linear in terms of the correlation coefficients. The method comes with a large computational cost. Gratiet [22] decreases the time complexity by defining a recursive formulation, in which the problem is transformed from one GPR model to multiple independent GPR models each corresponding to a fidelity.

One paper that enhances GPR with multi-fidelity to increase the capability of inferring non-linear correlation between fidelities is the Deep Multi-fidelity GPR model [37]. This is both an extension of the multi-fidelity GPR model by Kennedy et al. [17] and a generalization of the ManifoldGP model [2]. The latter transforms the input space to a manifold, meaning that the observations are first transformed by a mapping and then parsed into the standard GPR model. This extension to multi-fidelity uses a multi-layered neural network as its transformation. Its weights are jointly optimized with the hyperparameters of the multi-fidelity GPR model. It is shown that this method is capable of capturing complex and even discontinuous correlations between fidelities.

Another enhancement to improve the inference of non-linear correlation compared to the model of Gratiet is the method called Non-linear Auto-Regressive multi-fidelity Gaussian Process (NARGP) [35]. Their motivation is that in some engineering problems, in their case computational fluid dynamics, the correlation is highly space-dependent. They adopt a functional composition approach inspired by deep learning, where each successive fidelity is defined as a GPR that has the previous predictive distribution as additional input. Hence, they also create a new kernel that combines these two inputs that have different characteristics. They place the same assumptions on the data and model as Gratiet's recursive model and therefore the optimization of the hyperparameters has a similar computational cost. The predictive distribution is non-Gaussian, hence it is computed using Monte Carlo integration but they state that the additional computational cost is negligible.

Cutajar et al. [4] expands on the NARGP model by removing their structural assumptions and constraints creating the multi-fidelity Deep Gaussian Process (DGP). In essence, they keep the deep learning formulation but abandoned the recursive formulation; all fidelities must be jointly optimized. This leads to more sensible and conservative uncertainty estimates because the model is optimized holistically.

2.4. Contributions

Most of the current method development regarding multi-fidelity GPR aims to infer more complex correlations by increasing the complexity of the correlation assumption between successive fidelities. Instead, this thesis approximates the non-linear correlation using local approximation methods where each local model keeps the simple linear correlation assumption. This results in a linear piece-wise approximation of the correlation. Besides this benefit, it also gains the advantages of local approximation methods: the ability to capture non-stationary features and a decrease in computational cost.

Although no other literature presents such a study, as far as the author knows, there is a study by Rumpfkeil et al. [43, 42] that enhances Gratiet's recursive multi-fidelity GPR formulation [22] with an explicit mixture-of-experts. Their aim of incorporating the local approximation technique is to improve the inference on non-stationary datasets and not on improving the correlation inference. Therefore, they do not mention the effect of the local approximation technique on correlation inference. This leaves a gap for this thesis to fill.

The thesis explores two local approximation methods in the context of GPR with multi-fidelity. One is based on constraining the local models at the boundary between them and the other is based on model averaging by weighing the local predictions.

3

Method Description

The following sections explain the mathematical details of the models used in the investigation of inferring non-linear correlations between fidelities. It starts by stating the mathematical background, and the model selection and prediction process of GPR following the notation in Rasmussen & Williams [52]. Then, the local approximation methods are explained, starting with the description of the naive local experts: local independent GPR models. The thesis uses this method as a base case in the investigation and it forms the basis with which the two local approximation methods based on model averaging techniques are explained: the constrained boundary GPR (CB-GPR), constrains the predictive mean and variance at the boundaries between the locally independent GPRs, and the random variable mixture GPR (RVM-GPR), weighs multiple locally independent GPRs. After this, the mathematical details of GPR with multi-fidelity are given following the recursive formulation of Gratiet et al. [22]. And at last, the combination of using local approximation methods with GPR (with multi-fidelity) is explained.

3.1. Gaussian Process Regression

Before defining GPR, it is important to first define Gaussian processes and the observational data. A Gaussian process (GP) is defined as a collection of random variables, any finite number of which have a joint Gaussian distribution [52]. The random variables are indexed by a d-dimensional subset of R^d; this thesis only focuses on the 1-dimensional case. This definition is equivalent to defining a GP by a mean function $m(\mathbf{x})$ and a covariance function $k(\mathbf{x}, \mathbf{x}')$, showing that a GP is a generalization of Gaussian distributions to the functional space [52].

With GPs now defined, it is important to also first define the notation of the observational data that the GPR model uses. The observational training data of GPR is written as $\mathfrak{D} = \{(\mathbf{x}_i, y_i) \mid i = 1, ..., n\}$ where n is the number of observations, $\mathbf{x}_i$ are the d-dimensional inputs and y_i are the scalar targets. The inputs and targets are aggregated into the $d \times n$-dimensional design matrix $\mathbf{X}$ and the n-dimensional target vector $\mathbf{y}$, respectively. Therefore, the training data $\mathfrak{D}$ is also written as $\mathfrak{D}(\mathbf{X}, \mathbf{y})$.

GPR aims to infer the relation between the observational inputs and targets using the Bayesian approach. The GPR model is defined as

$$y_i = f(\mathbf{x}_i) + \epsilon \tag{3.1}$$

where $f(.)$ is the latent function, $f(\mathbf{x}_i)$ is the function value at $\mathbf{x}_i$, and $\epsilon \sim \mathcal{N}(0, \sigma_n^2)$ is the additive i.i.d. Gaussian noise with noise variance σ_n^2. The function values are also written as f_i and the corresponding aggregated n-dimensional vector is written as $\mathbf{f}$. The inference of the relation starts by placing a prior Gaussian process (GP) on the latent function which is seen as a distribution over functions. Then, the prior is conditioned with the observations to obtain the posterior GP with which new predictions are made.

When the prior GP is placed on the latent function $f(\mathbf{x})$, then this is written as $f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$. The mean function of all prior GPs in the experiments in this thesis is assumed to be zero, therefore the

inclusion of the mean function in the mathematical derivations is left out. This assumption is also often made in the literature for simplicity [52]. The interested reader is referred to section 2.7 of Rasmussen & Williams [52] for more information on how to incorporate a non-zero mean function.

The covariance function of the prior GPs in this thesis is chosen to be the squared exponential function

$$k(\mathbf{x},\mathbf{x}') = \sigma_f^2 \, exp\left(-\frac{1}{2l^2}\|\mathbf{x}-\mathbf{x}'\|^2\right). \tag{3.2}$$

It is the most used covariance function, it provides a notion that close inputs have similar outputs, it is infinitely differentiable which results in smooth GPR predictions and it is stationary meaning that it is invariant under rigid motions (translation and permutation of the inputs). The function has two parameters, namely the signal variance σ_f^2 that determines the variance of the signal without noise and the length-scale l^2 that determines the scale at which two points are correlated [52]; these are referred to as hyperparameters of the GPR model. The covariance function can be freely defined as long as it is positive semidefinite [52]. For a more detailed overview of covariance functions refer to chapter 4 of Rasmussen & Williams [52] and chapter 2 of Duvenaud [8].

The training data is finite therefore the covariance function is more often defined as a matrix. Suppose $\mathbf{X}_1 \in \mathscr{R}^{n_1 \times d}$ and $\mathbf{X}_2 \in \mathscr{R}^{n_2 \times d}$ are aggregated matrices of n_1 and n_2 inputs, respectively, then the kernel of the aggregated matrices with covariance function $k(.,.)$ is defined as $\mathbf{K}(\mathbf{X}_1,\mathbf{X}_2) \in \mathscr{R}^{n_1 \times n_2} := \{k(\mathbf{x}_1,\mathbf{x}_2) \,|\, \forall \mathbf{x}_1 \in \mathbf{X}_1, \forall \mathbf{x}_2 \in \mathbf{X}_2\}$.

Prediction Predictions with a GPR model at n_* new inputs $\mathbf{X}_* \in \mathscr{R}^{n_* \times d}$ are made by calculating the predictive distribution of the corresponding function values $\mathbf{f}_*$. The predictive distribution of $\mathbf{f}_*$ is defined as the conditional distribution of $\mathbf{f}_*$ given the observational training data $\mathfrak{D}(\mathbf{X},\mathbf{y})$ which is written as $p(\mathbf{f}_*|\mathbf{X},\mathbf{y},\mathbf{X}_*)$. The conditional distribution is calculated by aplying the rules of conditioning Gaussian distributions and the joint distribution of the observed targets $\mathbf{y}$ and the function values $\mathbf{f}_*$ at new inputs $\mathbf{X}_*$. The joint distribution is calculated using the prior and is given by

$$\begin{bmatrix}\mathbf{y}\\ \mathbf{f}_*\end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix}\mathbf{0}\\ \mathbf{0}\end{bmatrix}, \begin{bmatrix}\mathbf{K}(\mathbf{X},\mathbf{X}) + \sigma_n^2\mathbf{I} & \mathbf{K}(\mathbf{X},\mathbf{X}_*)\\ \mathbf{K}(\mathbf{X}_*,\mathbf{X}) & \mathbf{K}(\mathbf{X}_*,\mathbf{X}_*)\end{bmatrix}\right) \tag{3.3}$$

where the $\sigma_n^2\mathbf{I}$ term accounts for the assumed additive i.i.d. Gaussian noise on the observed targets, see equation 3.1. Rearranging the joint distribution results in the conditional distribution on the training data, also called the predictive distribution

$$\mathbf{f}_*|\mathbf{X},\mathbf{y},\mathbf{X}_* \sim \mathcal{N}\left(\mathbb{E}\left[\mathbf{f}_*|\mathbf{X},\mathbf{y},\mathbf{X}_*\right], \mathrm{cov}\left[\mathbf{f}_*|\mathbf{X},\mathbf{y},\mathbf{X}_*\right]\right) \quad \text{with} \tag{3.4}$$

$$\mathbb{E}\left[\mathbf{f}_*|\mathbf{X},\mathbf{y},\mathbf{X}_*\right] = \mathbf{K}(\mathbf{X}_*,\mathbf{X})\,\mathbf{R}^{-1}\mathbf{y} \quad \text{and} \tag{3.5}$$

$$\mathrm{cov}\left[\mathbf{f}_*|\mathbf{X},\mathbf{y},\mathbf{X}_*\right] = \mathbf{K}(\mathbf{X}_*,\mathbf{X}_*) - \mathbf{K}(\mathbf{X}_*,\mathbf{X})\,\mathbf{R}^{-1}\mathbf{K}(\mathbf{X},\mathbf{X}_*) \tag{3.6}$$

where $\mathbf{R} = \mathbf{K}(\mathbf{X},\mathbf{X}) + \sigma_n^2\mathbf{I}$. The predictive distribution of the noisy test targets $\mathbf{y}_*$ at prediction inputs $\mathbf{X}_*$ is simply computed by adding $\sigma_n^2\mathbf{I}$ to $\mathrm{cov}\left[\mathbf{f}_*\right]$ [52].

The posterior GP is defined as the GP with its mean and covariance equal to that of equation 3.5 and 3.6.

Figure 3.1 shows five samples from a prior GP and figure 3.2 shows five samples from the posterior GP derived by conditioning the prior with two observations, denoted by the black dots. Note, that all samples of the posterior GP go through the observations, as no signal noise is assumed on the posterior else this will not necessarily happen, and they deviate more when moving further away from the observations. Actually, in the extrapolation regime, the prior is dominant, meaning that the mean and variance of the posterior go to the mean and variance of the prior. This causes its prediction capabilities to be limited in these areas.

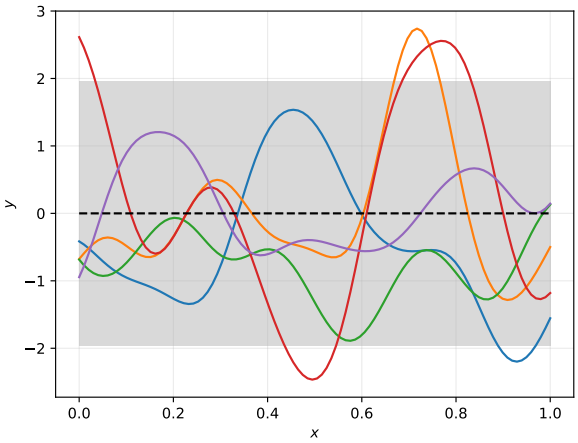


Figure 3.1: 5 Samples from a prior with zero mean

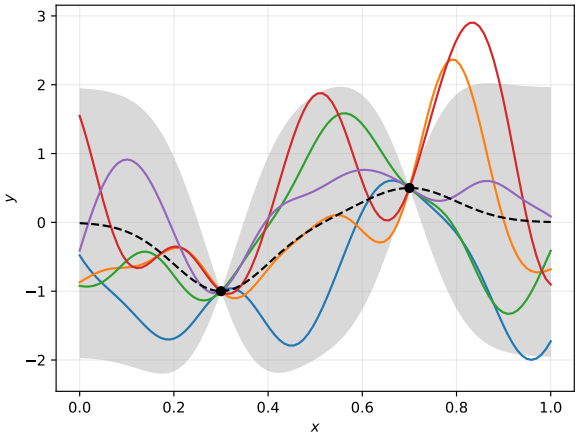


Figure 3.2: 5 Samples from a posterior conditioned on 2 observations

Model Selection The hyperparameters of a GPR model consist of the noise variance σ_n^2 and the parameters of the covariance function. In literature, their determination is referred to as model selection of which several methods exist [52], for example, Bayesian model selection, cross-validation, and marginal likelihood maximization. The latter is the most popular as it uses all observations and is less computationally expensive. It is defined as the maximization of the likelihood that the targets originate from the inputs given the hyperparameters: $p(\mathbf{y}|\mathbf{X},\boldsymbol{\theta})$. In practice, the negative log marginal likelihood (NLML) $-\log p(\mathbf{y}|\mathbf{X},\boldsymbol{\theta})$ is minimized, as it reduces the equation to a much simpler form while preserving the hyperparameters

$$\log p(\mathbf{y}|\mathbf{X},\boldsymbol{\theta}) = -\frac{1}{2}\mathbf{y}^T\mathbf{R}^{-1}\mathbf{y} - \frac{1}{2}\log|\mathbf{R}| - \frac{n}{2}\log 2\pi \tag{3.7}$$

where $\mathbf{R} = \mathbf{K}(\mathbf{X},\mathbf{X}) + \sigma_n^2\mathbf{I}$. Its first term represents the data-fit and the second is the complexity penalty. Therefore, the minimization automatically balances both which results in a less over-fit model that makes this model selection approach a good fit.

Minimization algorithms require the gradients with respect to the parameters of the function it minimizes. Therefore, the derivative of the LML with respect to the hyperparameters θ_j must be calculated. It is equal to

$$\frac{\partial}{\partial\theta_j}\log p(\mathbf{y}|\mathbf{X},\boldsymbol{\theta}) = \frac{1}{2}\mathrm{tr}\left(\left(\boldsymbol{\alpha}\boldsymbol{\alpha}^T - \mathbf{R}^{-1}\right)\frac{\partial\mathbf{R}}{\partial\theta_j}\right) \tag{3.8}$$

where $\boldsymbol{\alpha} = \mathbf{R}^{-1}\mathbf{y}$. This derivative is analytically tractable as long as the derivative of the covariance function with respect to all its hyperparameters is.

The minimization algorithms require the use of multiple restarts because the NLML often has multiple local minima. As an example, figure 3.3 shows an NLML contour plot of a GPR model with 20 observations and the signal variance set to 1.0. It shows two local minima when varying the noise variance and the length-scale of the local model. Therefore, this thesis uses 250 restarts in the minimization algorithm to increase the chance of finding the global minimum.

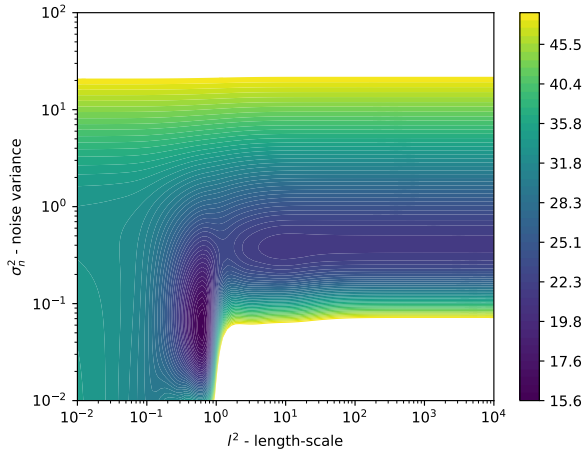


Figure 3.3: GPR Negative Log Marginal Likelihood

3.2. Local Approximation Methods

This section explains the most basic local approximation method and discusses two model-averaging techniques that are used in this thesis. These are methods that combine a set of local models with the aim of reducing the computational cost and increasing the effectiveness of capturing non-stationary features. Also, these techniques solve the issue of discontinuities in the predictions that the general local approximation methods have.

Note, the thesis predetermines the regions of all local approximation methods that are considered. This means specifically, that these regions are not determined by optimization with the observations.

Training Data In the following section, the training data $\mathfrak{D}$ is partitioned via m disjoint regions Ω_k. Each region defines a subset of the training data as $\mathfrak{D}_k := \{(\mathbf{x}, y) \,|\, \forall (\mathbf{x}, y) \in \mathfrak{D}, \mathbf{x} \in \Omega_k\}$, where n_k is the number of observations in $\mathfrak{D}_k$. In turn, the aggregated $d \times n_k$-dimensional input matrix and the n_k-dimensional target vector are defined as $\mathbf{X}_k$ and $\mathbf{y}_k$, respectively. The k'th local training data can be written as $\mathfrak{D}_k = (\mathbf{X}_k, \mathbf{y}_k)$.

Local Regression Problem In literature, the most basic local approximation technique is the Inductive Naive Local Experts (INLE) as described by Liu et al. [25]. The method partitions the input space such that each partition only uses a particular GPR model. This means that GPR takes place in a given $\mathcal{GP}_k$ for partitioned region Ω_k. The optimization and prediction processes are sparsified because each model is independent of the other. Therefore, this results in m independent GPR problems each using their respective partitioned training data $\mathfrak{D}_k$

$$y_{k_i} = f_k(\mathbf{x}_{k_i}) + \epsilon_k \qquad (\mathbf{x}_{k_i}, y_{k_i}) \in \mathfrak{D}_k \tag{3.9}$$

where i refers to the i'th observation $(\mathbf{x}_{k_i}, y_{k_i})$ of partitioned dataset $\mathfrak{D}_k$. As a result of each model's independence the covariance of two targets from different partitions of the dataset is equal to zero: $\text{cov}\left[y_{k_i}, y_{l_j}\right] = 0$ when $k \neq l$.

The local GPR models result in discontinuous predictions and miss global spatial correlations [25]. The following two sections introduce the CB-GPR method and the RVM-GPR method that overcomes this issue by means of model averaging.

3.2.1. Constrained Boundary GPR

The constrained boundary GPR model (CB-GPR) consists of a set of disjoint local GPR models that are constrained to be equal to their neighbors at their respective boundaries. This is achieved by adding continuity constraints for the predictive mean and predictive variance to the LML in the optimization

process of the hyperparameters. The idea of adding constraints to GPR via the LML in the optimization process originates from Pensoneault et al. [33]. An example of predicting with the CB-GPR model is shown in Figure 3.4.

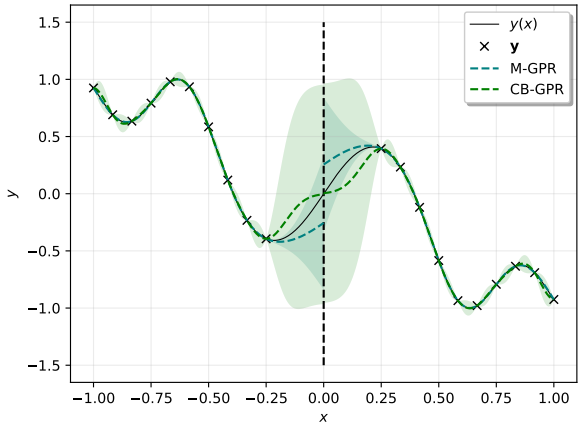


Figure 3.4: Example CB-GPR Method

Constraints The prediction process of the CB-GPR method is similar to predicting with the local GPR models, but the optimization of the hyperparameters differs slightly due to the added constraints. The CB-GPR method constrains the predictive mean and predictive variance of all pairs of local models at their respective boundaries. Assume local models from partition k and l then their constraints are placed at inducing points $\mathbf{x}_{kl_*} \in \Gamma_{kl}$ where Γ_{kl} denotes the boundary between the local models. This method only constrains the local models at a finite number of points, for input spaces that are larger than 1-dimensional, this means that the boundaries between their local models are discontinuous except at the inducing points.

For a given pair of local models, k and l, the constraints at inducing point $\mathbf{x}_{kl_*}$ placed on the local models' function values $f_{k_*}(\mathbf{x}_{kl_*})$ and $f_{l_*}(\mathbf{x}_{kl_*})$, respectively, are written as

$$|\mathbb{E}\left[f_{k_*}(\mathbf{x}_{kl_*})\right] - \mathbb{E}\left[f_{l_*}(\mathbf{x}_{kl_*})\right]| \leq \epsilon_E \quad \text{and} \tag{3.10}$$

$$|\text{var}\left[f_{k_*}(\mathbf{x}_{kl_*})\right] - \text{var}\left[f_{l_*}(\mathbf{x}_{kl_*})\right]|^{\frac{1}{2}} \leq \epsilon_{\text{var}} \tag{3.11}$$

where ϵ_E and ϵ_{var} are upper bounds for the constraints that are both set to 0.01 in this thesis.

Model Selection All hyperparameters are optimized simultaneously by minimizing the global negative log marginal likelihood because the constraints create a dependency between the local models. This is defined as

$$\log p\left(\mathbf{y}|\mathbf{X},\boldsymbol{\theta}\right) = \sum_{k=1}^{m} \log p\left(\mathbf{y}_k|\mathbf{X}_k,\boldsymbol{\theta}_k\right) \tag{3.12}$$

where $\log p\left(\mathbf{y}_k|\mathbf{X}_k,\boldsymbol{\theta}_k\right)$ is equal to the log marginal likelihood of the individual local models, see equation 3.7. The gradients with respect to the hyperparameters are easy to compute using equation 3.8, which results in

$$\frac{\partial}{\partial\theta_j}\log p\left(\mathbf{y}|\mathbf{X},\boldsymbol{\theta}\right) = \sum_{k=1}^{m} \frac{\partial}{\partial\theta_j}\log p\left(\mathbf{y}_k|\mathbf{X}_k,\boldsymbol{\theta}_k\right) = \frac{\partial}{\partial\theta_j}\log p\left(\mathbf{y}_{\hat{k}_j}|\mathbf{X}_{\hat{k}_j},\boldsymbol{\theta}_{\hat{k}_j}\right) \tag{3.13}$$

$$= \frac{1}{2}\text{tr}\left((\boldsymbol{\alpha}_{\hat{k}_j}\boldsymbol{\alpha}_{\hat{k}_j}^T - \mathbf{R}_{\hat{k}_j}^{-1})\frac{\partial\mathbf{R}_{\hat{k}_j}}{\partial\theta_j}\right) \tag{3.14}$$

with $\hat{k}_j$ being the index such that $\theta_j \in \boldsymbol{\theta}_{\hat{k}_j}$ holds. Also, notice that the optimization step is not sparsified due to the constraints making the models dependent on each other in the optimization process.

The minimum solution of the NLML and the constraint NLML are usually not equal. Figure 3.5 shows a contour plot of the NLML, the contours of the predictive mean constraint with $\epsilon_E \in [0.01, 0.1, 1.0]$, and the minimum NLML and the minimum constraint NLML with $\epsilon_E = 0.01$ of a CB model with two local models. The length-scale of both local models are varied while the other hyper-parameters are constants. The plot shows that the minimum of the NLML and of the constraint NLML differ significantly based on the value of ϵ_E.

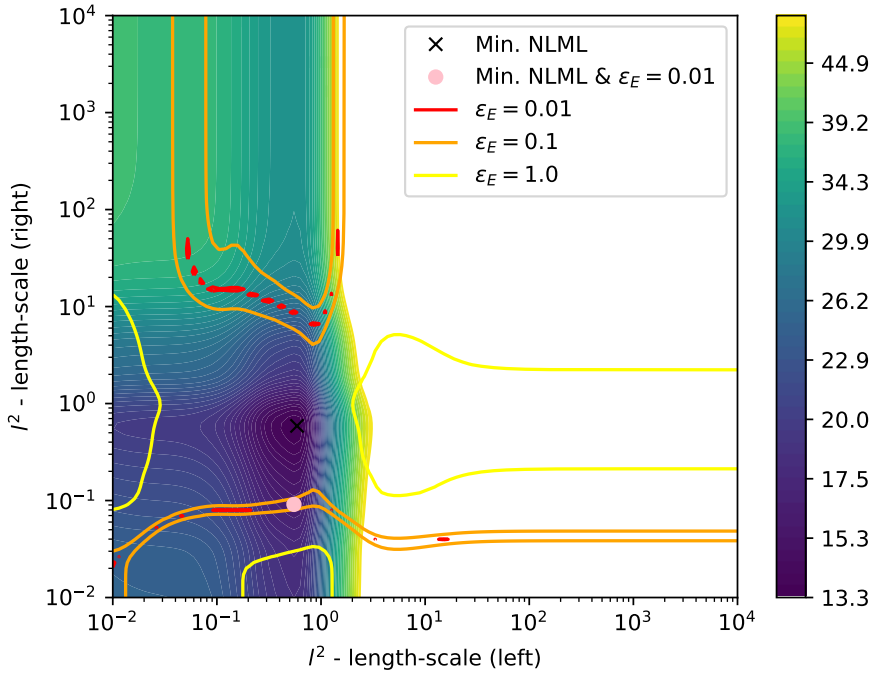


Figure 3.5: CB-GPR - NLML

3.2.2. Random Variable Mixture GPR

The random variable mixture GPR model (RVM-GPR) weighs the predictive function values of multiple independent GPR models each responsible for its own region. Figure 3.6 shows an example of a prediction of an RVM-GPR model. The predictive function values of the two local models and their respective weights are also shown.

This method is based on an idea presented by Vijayakumar et al. [50] and by Nguyen-Tuong et al. [29]: local models are weighed using distance measures.

Definition The RVM-GPR does not use the full Bayesian treatment, as do the other methods, but follows the probabilistic curve-fitting approach: predictions are directly made with the distribution that is assumed on target y given input $\mathbf{x}$. Although the observations are not needed for the prediction process, they are necessary for the determination of the hyperparameters by means of maximizing the log-likelihood of the observations under this model. Suppose m disjoint partitions $\mathfrak{D}_k$ of dataset $\mathfrak{D}$, then the distribution on target y given input $\mathbf{x}$ for the RVM-GPR model is assumed as the weighted sum of the posterior GPs $f_{k_*}(\mathbf{x})$ of locally independent GPR models that each is conditioned and optimized on one of the partitioned datasets. This distribution is defined as

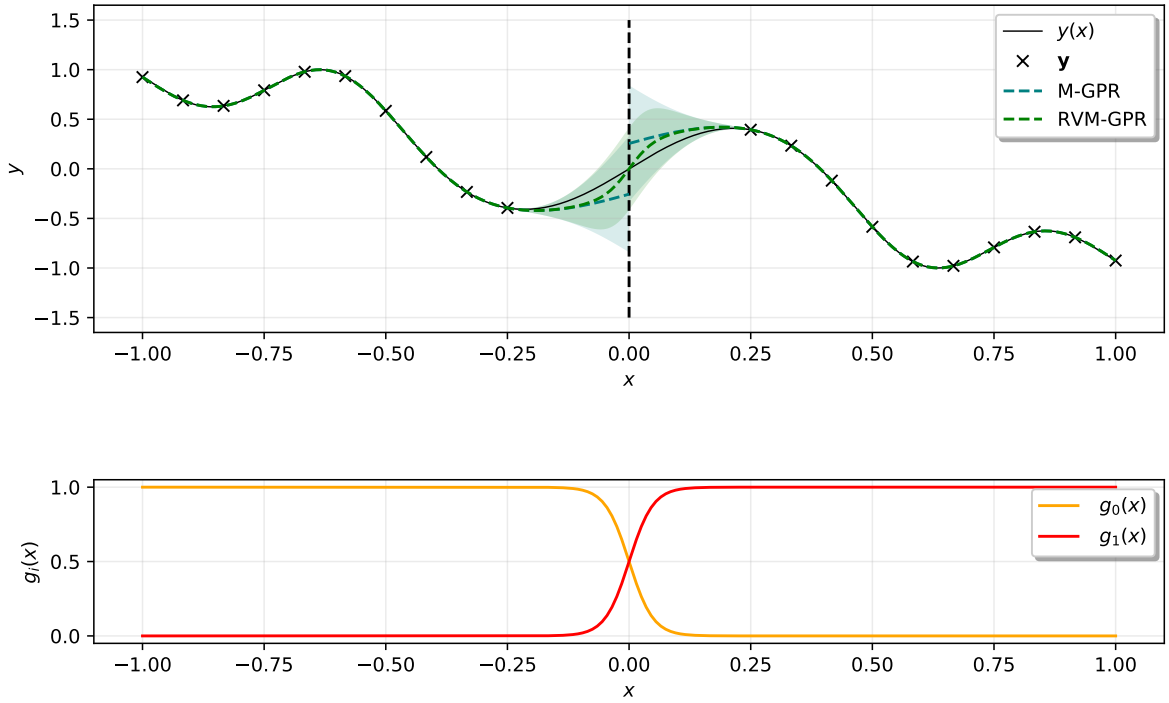


Figure 3.6: A prediction example of an RVM-GPR with two local models, in regions $[0,1]$ and $[1,2]$, (hyperparameters are handpicked) for $f(x) = sin(2x) + \mathcal{N}(x-1)$ and 6 observations linearly spaced across region $[0,2]$.

$$y \sim \underbrace{\left(\sum_{k=1}^{m} g_k(\mathbf{x}) f_{k_*}(\mathbf{x})\right)}_{f(\mathbf{x})} + \epsilon \tag{3.15}$$

with additive i.i.d. Gaussian noise $\epsilon \sim \mathcal{N}(0, \sigma_n^2)$, $g_k(\mathbf{x})$ the k'th weight function value at $\mathbf{x}$, $f_{k_*}(\mathbf{x})$ the predictive distribution of the function value at $\mathbf{x}$ of the k'th optimized locally independent GPR on dataset $\mathfrak{D}_k$, and $f(\mathbf{x})$ the function value of the RVM-GPR model.

This distribution defines a GP on targets y given inputs $\mathbf{x}$. This property follows from the fact that adding two independent GPs results in a new GP likewise for multiplying a GP by a scalar. Note, that the weight functions are deterministic, therefore they act as scalars. The mean and covariance function of this GP is easily derived using the basic rules for adding random variables and multiplying random variables by scalars:

$$m(\mathbf{x}) = \mathbb{E}[y] = \sum_{k=1}^{m} g_k(\mathbf{x}) \mathbb{E}[f_{k_*}(\mathbf{x})] \quad \text{and} \tag{3.16}$$

$$k(\mathbf{x}, \mathbf{x}') = \operatorname{cov}[y, y'] = \left[\sum_{k=1}^{m} g_k(\mathbf{x}) g_k(\mathbf{x}') \operatorname{cov}[f_{k_*}(\mathbf{x}), f_{k_*}(\mathbf{x}')]\right] + \sigma_n^2 \mathbf{I}, \tag{3.17}$$

respectively. The function value $f(\mathbf{x})$ follows an almost equivalent GP, except that the $\sigma_n^2 \mathbf{I}$ term is not present in the covariance function.

Weight function Following Nguyen-Tuong et al. [29], the weight functions $g_k(\mathbf{x})$ are defined as the normalized distance measure of distance measures $w_k(\mathbf{x})$

$$g_k(\mathbf{x}) = \frac{w_k(\mathbf{x})}{\sum_{l=1}^{m} w_l(\mathbf{x})}. \tag{3.18}$$

The thesis only explores cases with 1-dimensional inputs and two local models. A natural transition between the two local models is made by defining the two distance measures $w_k(x)$ such that $g_k(x)$ are sigmoid functions which means that

$$w_1(x) = \exp\left(-\frac{1}{2l^2}(x-c)\right) \quad \text{and} \tag{3.19}$$

$$w_2(x) = \exp\left(\frac{1}{2l^2}(x-c)\right) \tag{3.20}$$

where l^2 is the length-scale of the weights that determines the size of the transition zone between the two local models and c determines the center of that transition zone. Figure 3.7 shows the effect of weighing two constant functions ($f_1(x) = 1.0$ and $f_2(x) = -1.0$) with these distance measures. The resulting weighted function is shown two times, once with a length-scale of 0.1 and one with 0.25, where the boundary is defined at $c = 0.5$. This shows that increasing the length-scale flattens the transition between the weighted functions.

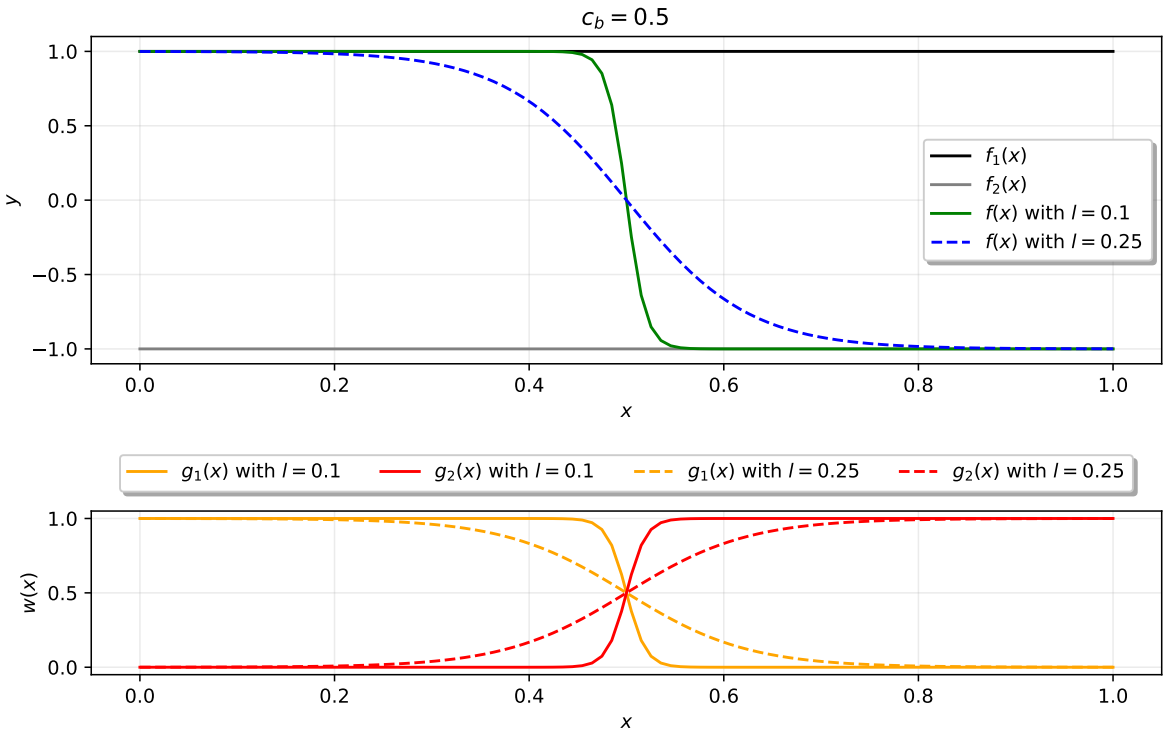


Figure 3.7: RVM-GPR - Length-scale

Model Selection The hyperparameters of the RVM-GPR are optimized in a two-step process: first, the hyperparameters of the locally independent GPR models are optimized using the log-likelihood of its partitioned dataset, and thereafter the hyperparameters of the weight functions and the noise variance σ_n^2 are optimized using the log-likelihood of the complete dataset given the posteriors of the locally independent GPR models. The log-likelihood of the complete dataset on the RVM-GPR model given the posteriors is equal to[1]

$$\log p\left(\mathbf{y}|\mathbf{X},\boldsymbol{\theta}\right) = -\frac{1}{2}(\mathbf{y} - \mathbb{E}\left[\mathbf{f}(\mathbf{X})\right])^T \mathbf{R}_{\text{rvm}}^{-1}(\mathbf{y} - \mathbb{E}\left[\mathbf{f}(\mathbf{X})\right]) - \frac{1}{2}\log|\mathbf{R}_{\text{rvm}}| - \frac{n}{2}\log 2\pi \tag{3.21}$$

with $\mathbf{R}_{\text{rvm}} = \text{cov}\left[\mathbf{f}(\mathbf{X}), \mathbf{f}(\mathbf{X})\right] + \sigma_n^2\mathbf{I}$, $\mathbf{y}$ the aggregated observational targets, $\mathbf{X}$ the aggregated observational inputs, and $\mathbf{f}(\mathbf{X})$ the aggregated distributions of the function values at the observational inputs.

[1] The conditioned posteriors are left out of the left side of the equation, else the equation becomes too large to fit on the page.

The log-likelihood is calculated using the fact that the distribution follows a multivariate normal distribution which follows from equations 3.16 and 3.17

$$\mathbf{y}|\mathbf{X},\boldsymbol{\theta} \sim \mathcal{N}\left(\mathbb{E}\left[\mathbf{f}(\mathbf{X})\right], \operatorname{cov}\left[\mathbf{f}(\mathbf{X}),\mathbf{f}(\mathbf{X})\right] + \sigma_n^2\mathbf{I}\right). \tag{3.22}$$

The determination of the hyperparameters of the weight functions is not sparsified. Meaning that the optimization of the hyperparameters of a specific weight function can not be reduced to optimizing with only its partitioned dataset. Because, when $\mathbf{x}_p$ and $\mathbf{x}_q$ are inputs from two different partitions, then the covariance of their function values $\operatorname{cov}\left[f(\mathbf{x}_p), f(\mathbf{x}_q)\right]$ is by its definition non-zero (see equation 3.17).

Prediction Predictions using this model are made by inserting the new input and substituting the optimized hyperparameters and the posterior function values of the locally independent GPR models into equations 3.16 and 3.17. The covariance matrix between predicted function values of new inputs is not sparsified for the same reason that the log-likelihood of the RVM-GPR is not sparsified.

3.3. Multi-GPR

So far, this chapter has explained the single-fidelity GPR method and the two stitching methods. In this section, the multi-fidelity GPR method is explained so that a mathematical basis is formed to explain the extension of the stitching methods to multi-fidelity. This is important because the thesis investigates the correlation inference of splitting and stitching in the non-linearly correlated setting between fidelities.

Multi-fidelity refers to the usage of multiple fidelities that each represents a different complexity of the function to be modeled. In most cases, observations from a higher complex fidelity require a higher computational cost to obtain but are more accurate. This makes the usage of multi-fidelity preferable, as usually, the single-fidelities methods use the most complex fidelity. Thus multi-fidelity strikes a balance between accuracy and the computational cost of obtaining the observations.

This thesis uses the recursive approach of L. Le Gratiet [21] as the definition of the multi-fidelity GPR. This approach is preferred, because, the prediction and optimization process is processed as separate GPRs instead of a single one that incorporates all fidelities at once. This reduces the computational cost significantly because the computational cost of a GPR scales cubically with the number of observations.

The theory presented in this section is given for n fidelities but keep in mind that this thesis only uses two.

Regression The multi-fidelity regression problem with i.i.d. Gaussian noises $\epsilon_t \sim \mathcal{N}\left(0, \sigma_{n_t}^2\right)$, for each fidelity t, is defined [21] as

$$y_{t_i} = f_t\left(\mathbf{x}_{t_i}\right) + \epsilon_t \quad \text{and} \tag{3.23}$$

$$y_{t_i} = \underbrace{\rho_{t-1} f_{t-1}(\mathbf{x}_{t_i}) + \delta_t(\mathbf{x}_{t_i})}_{f_t(\mathbf{x}_{t_i})} + \epsilon_t = f_t\left(\mathbf{x}_{t_i}\right) + \epsilon_t \tag{3.24}$$

where $\sigma_{n_t}^2$ is called the noise variance of fidelity t. Equation 3.23 defines each fidelity t and equation 3.24 defines the relation between fidelity $t \neq 1$ and the previous fidelity $t-1$. The scalar ρ_{t-1} is called the correlation factor.

The goal of the regression problem is to infer the latent functions f_t between the observations $\mathbf{x}_t$ and y_t. This is achieved by inferring the latent function f_t for $t = 1$ and the functions δ_t for $t \neq 1$ in GPs: $\mathcal{GP}\left(m_t(\mathbf{x}), k_t(\mathbf{x},\mathbf{x}')\right)$. In most cases, the mean function $m_t(\mathbf{x})$ is assumed to be the zero mean function, and the covariance functions $k_t(\mathbf{x},\mathbf{x}')$ is assumed to be the squared exponential function.

Model Selection The optimization process of the Multi-GPR is as follows:

- Create fidelity model $t = 1$ and optimize its hyperparameters by minimizing its negative log marginal likelihood using the training data $\mathfrak{D}_{t=1}$.
- For each fidelity $t > 1$:
  - Create fidelity model t and optimize its hyperparameters by minimizing its negative log marginal likelihood using the training data $\mathfrak{D}_t$ and the predictive function values of fidelity model $t-1$.

The log marginal likelihood of the first fidelity is calculated with equation 3.7 and the log marginal likelihood for the other fidelities $t \neq 1$ is equal to

$$\log p\left(\mathbf{y}_t|\mathbf{X}_t, \mathbf{f}_{t-1_*}\right) = -\frac{1}{2}(\mathbf{y}_t - \rho_{t-1}\mathbb{E}\left[\mathbf{f}_{t-1_*}(\mathbf{X}_t)\right])^T\mathbf{R}_t^{-1}(\mathbf{y}_t - \rho_{t-1}\mathbb{E}\left[\mathbf{f}_{t-1_*}(\mathbf{X}_t)\right]) - \frac{1}{2}\log|\mathbf{R}_t| - \frac{n_t}{2}\log 2\pi \quad (3.25)$$

where $\mathbf{R}_t = \mathbf{K}_t(\mathbf{X}_t, \mathbf{X}_t) + \sigma_{n_t}^2\mathbf{I}$. The minimum value of ρ_{t-1} with respect to the negative log marginal likelihood is analytically tractable, therefore it can be decoupled from the optimization process. The minimum value is calculated by solving the following equation for ρ_{t-1}:

$$\nabla_{\rho_{t-1}} \log p\left(\mathbf{y}_t|\mathbf{X}_t, \mathbf{f}_{t-1_*}\right) = 0. \quad (3.26)$$

Solving for ρ_{t-1} results in

$$\hat{\rho}_{t-1} = \left(\mathbb{E}\left[\mathbf{f}_{t-1_*}(\mathbf{X}_t)\right]^T\mathbf{R}_t^{-1}\mathbb{E}\left[\mathbf{f}_{t-1_*}(\mathbf{X}_t)\right]\right)^{-1}\mathbb{E}\left[\mathbf{f}_{t-1_*}(\mathbf{X}_t)\right]\mathbf{R}_t^{-1}\mathbf{y}_t. \quad (3.27)$$

Prediction Prediction with Multi-GPR follows the same procedure as GPR with the exception that each fidelity is predicted recursively. The conditional distribution of the function values $\mathbf{f}_t$ on the training data $\mathfrak{D}_t = (\mathbf{X}_t, \mathbf{y}_t)$ and the training data of the previous fidelities is written as $p\left(\mathbf{f}_{t_*}|\mathbf{X}_1, \mathbf{y}_1 ... \mathbf{X}_t\mathbf{y}_t, \mathbf{X}_*\right)$. The conditional distribution of fidelity $t = 1$ is equal to the conditional distribution of GPR without multi-fidelity, see equation 3.4. The conditional distribution on the training data for the other fidelities $f_{t\neq 1}$ is found to be

$$\mathbf{f}_{t_*}|\mathbf{X}_1, \mathbf{y}_1 ... \mathbf{X}_t\mathbf{y}_t, \mathbf{X}_* = \mathcal{N}\left(\mathbb{E}\left[\mathbf{f}_{t_*}|\mathbf{X}_1, \mathbf{y}_1 ... \mathbf{X}_t\mathbf{y}_t, \mathbf{X}_*\right], \operatorname{cov}\left[\mathbf{f}_{t_*}|\mathbf{X}_1, \mathbf{y}_1 ... \mathbf{X}_t\mathbf{y}_t, \mathbf{X}_*\right]\right) \quad \text{with} \quad (3.28)$$

$$\mathbb{E}\left[\mathbf{f}_{t_*}|\mathbf{X}_1, \mathbf{y}_1 ... \mathbf{X}_t\mathbf{y}_t, \mathbf{X}_*\right] = \rho_{t-1}\mathbb{E}\left[\mathbf{f}_{t-1_*}(\mathbf{X}_{t_*})\right] + \mathbf{K}_t(\mathbf{X}_{t_*}, \mathbf{X}_t)\mathbf{R}_t^{-1}(\mathbf{y}_t - \rho_{t-1}\mathbb{E}\left[\mathbf{f}_{t-1_*}(\mathbf{X}_t)\right]) \quad \text{and} \quad (3.29)$$

$$\operatorname{cov}\left[\mathbf{f}_{t_*}|\mathbf{X}_1, \mathbf{y}_1 ... \mathbf{X}_t\mathbf{y}_t, \mathbf{X}_*\right] = \rho_{t-1}^2\operatorname{cov}\left[\mathbf{f}_{t-1_*}\right] + \mathbf{K}_t(\mathbf{X}_{t_*}, \mathbf{X}_{t_*}) - \mathbf{K}_t(\mathbf{X}_{t_*}, \mathbf{X}_t)\mathbf{R}_t^{-1}\mathbf{K}_t(\mathbf{X}_t, \mathbf{X}_{t_*}) \quad (3.30)$$

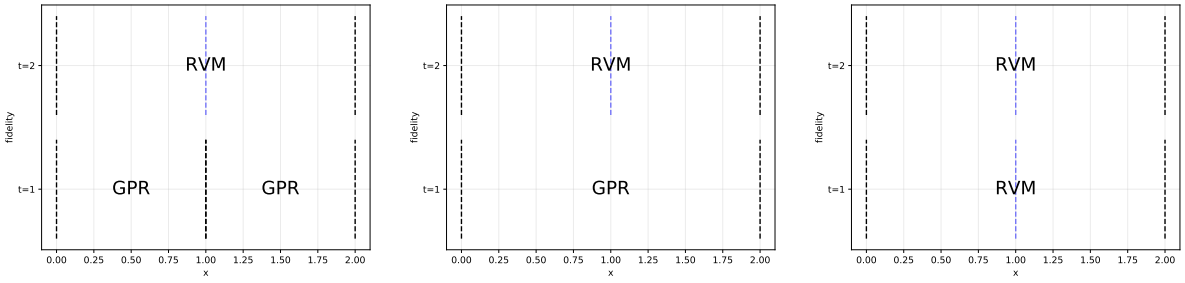
where $\mathbf{R}_t = \mathbf{K}_t(\mathbf{X}, \mathbf{X}) + \sigma_{n_t}^2\mathbf{I}$. The predictive distribution of the test targets $\mathbf{y}_{t_*}$ at prediction inputs $\mathbf{X}_*$ is simply computed by adding $\sigma_{n_t}^2\mathbf{I}$ to $\operatorname{cov}\left[\mathbf{f}_{t_*}\right]$.

3.4. Stitching with Multi-fidelity

A natural way of extending the two stitching methods to multi-fidelity is to replace their locally independent GPR models to locally independent GPR with multi-fidelity models. In essence, the predictive function values of the locally independent GPR models are replaced by the predictive function values of the highest fidelity of the locally independent GPR with multi-fidelity models. The process of prediction then follows that of GPR with multi-fidelity: for each locally independent model calculate the predictive distribution of the function values recursively for each fidelity and, for the RVM-GPR, calculate the weighed prediction at the end. The optimization process of the hyperparameters follows the same procedure as the GPR with multi-fidelity except that for the CB-GPR the constraints are only added to the highest fidelity and for the RVM-GPR the parameters of the weight functions are optimized after the optimization of each fidelity. This extension is referred to as the L-CB-GPR or L-RVM-GPR, where the pre-abbreviation "L" (local) represents that the lower fidelities are modeled as locally independent GPRs.

The local extension is not the only way to extend the stitching methods to multi-fidelity. This thesis also investigates two other extensions that model the low-fidelity differently: either as a global GPR or with the same stitching method. These two options are referred to with the pre-abbreviations "G" (global) and "LS" (locally stitched), respectively. These two extensions' prediction and optimization processes follow the same recursive procedure as for the local extension. Note, that for the locally-stitched extension, the stitching procedures must be applied at each fidelity. Of course, when considering more than two fidelities, one might define each lower fidelity using one of the three extensions, thus multiplying the possibilities. As this thesis only considers two fidelities, this is not further investigated and thus declared as out of scope.

For explanatory purposes, figure 3.8 figuratively shows the definition of the three extensions of the RVM-GPR to multi-fidelity, though the same visualization applies to the CB-GPR method. These definitions consider two fidelities, a one-dimensional input space, and two local models in regions $[0,1]$ and $[1,2]$. Each extension's definition shows what part of each fidelity is modeled by either a GPR or an RVM-GPR. The black lines divide these parts and the blue dotted lines denote the boundaries of the locally independent GPR models of the RVM-GPR at that specific fidelity.



(a) L-RVM-GPR (b) G-RVM-GPR (c) LS-RVM-GPR

Figure 3.8: A figurative definition of the three extensions of the RVM-GPR method to multi-fidelity.

4

Methodology

This chapter describes the methodology to answer the first three research questions in this thesis. Namely, the effect of splitting and stitching on the correlation inference between fidelities, and what effect changing the number of observations and sampling strategy of the input space has on the investigated methods.

This chapter provides a summary of each method that is used in answering these research questions. These methods are investigated in three cases each with a different correlation: one linear and two non-linear. For simplicity, these cases consist of two fidelities and have a one-dimensional input space, and the splitting and stitching methods always have two pre-clustered local models. This chapter also describes the generation of the different datasets that differ in the number of low- and high-fidelity observations and sampling strategy, and lastly, a description is given of how the performance of each method is measured across the datasets.

4.1. Methods

This thesis investigates the two stitching methods, CB-GPR and RVM-GPR, in the multi-fidelity setting using the three defined extensions with pre-abbreviations: "L" (local), "G" (global), and "LS" (locally stitched). For comparison, the single and multi-fidelity GPR are also considered as their split counterparts. The multi-fidelity models are pre-abbreviated with "MF" (multi-fidelity) and the split versions with "M" (multiple). This results in the following 10 methods shown in Table 4.1 (from now on the abbreviations are used to denote the methods).

The splitting and stitching methods all have two local models that are pre-clustered. This means that the determination of the region of each local model is not part of the optimization process, but instead is set beforehand. The regions of the two local models are $[-1,1]$ and $[1,3]$, and they correspond to the symmetric "boundary" that is present in two of the three cases that are defined further on.

The GP priors of all methods use a zero-mean function and the squared exponential function (equation 3.2) as the covariance function. Specifically for the CB methods, the upper bounds in all constraints are set to: $\epsilon_E = 0.01$ and $\epsilon_{var} = 0.01$.

The methods are implemented in a custom Python package that uses the Numpy and Autograd packages.

Optimization The optimization algorithm used for minimizing the negative log marginal likelihood to obtain the hyperparameters of the models is the conjugate gradient (CG) algorithm. The hyperparameters are found using 250 resets each with different initial hyperparameters that are sampled from the uniform distribution $\mathcal{U}(-10,10)$. The kernel hyperparameters are optimized in the log space, meaning that $\log(\theta_j)$ is optimized instead of θ_j. This overcomes the problem with negative values and the inability to optimize to small values for parameters that are squared, for example, σ_f^2, l^2, and σ_n^2.

The CB method uses the same procedure but uses the sequential least squares programming (SLSQP)

Logo	Abbreviation	Method Name
	GPR	Gaussian Process Regression
	M-GPR	Multiple Gaussian Process Regression
	MF-GPR	Multi-fidelity Gaussian Process Regression
	M-MF-GPR	Multiple Multi-fidelity Gaussian Process Regression
	L-RVM	Local Random Variable Mixture of Experts MF-GPR
	G-RVM	Global Random Variable Mixture of Experts MF-GPR
	LS-RVM	Locally Stitched Random Variable Mixture of Experts MF-GPR
	L-CB	Local Constrained Boundary MF-GPR
	G-CB	Global Constrained Boundary MF-GPR
	LS-CB	Locally Stitched Constrained Boundary MF-GPR

Table 4.1: Methods under investigation in this thesis.

algorithm for minimizing the negative log marginal likelihood because it is able to handle constraints on the loss function.

4.2. Cases

The performance of the methods is measured across three cases: constant ρ, discontinuous ρ, and linearly varying ρ. They each define a different correlation between the fidelities so that each method is investigated in one linear and two different non-linear correlated settings. The cases are realized by sampling functions from a multi-fidelity GP. Per case, the performance is measured across five functions to average out biases inherent to the sampled function. The multi-fidelity GP samples are taken by first sampling the low-fidelity GP where after it is multiplied by the case-specific correlation function $\rho(x)$, and the high-fidelity function sample is obtained by sampling the additive term from GP, thus

$$f_h(x) = \rho(x) f_l(x) + \delta(x) \quad \text{with} \tag{4.1}$$
$$f_l(x) \sim \mathcal{GP}\left(0, k_{f_l}(\mathbf{x}, \mathbf{x}')\right) \quad \text{and} \tag{4.2}$$
$$\delta(x) \sim \mathcal{GP}\left(0, k_{\delta}(\mathbf{x}, \mathbf{x}')\right). \tag{4.3}$$

Both GPs have a zero mean function and use the squared exponential kernel (equation 3.2) as the covariance function; the values of the hyperparameters are presented in table 4.2. The functions are sampled using 2000 linearly spaced points in $[-1, 3]$.

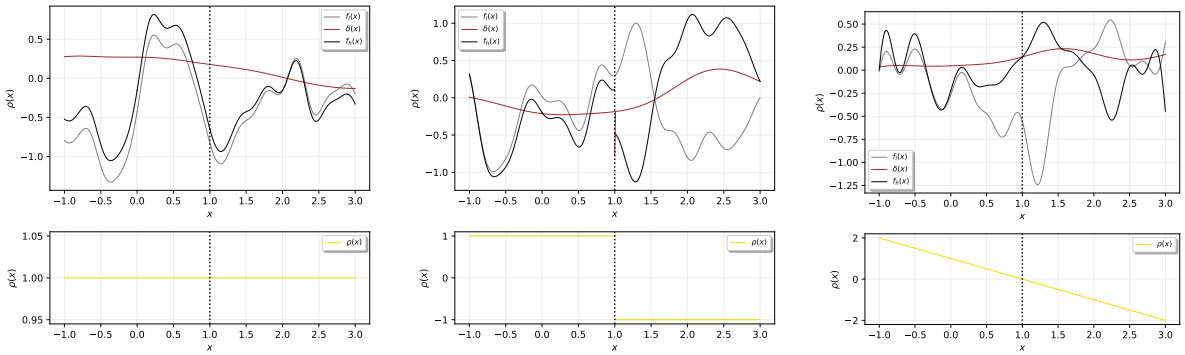
GP	σ_f^2	l^2
$f_l(x)$	0.25	0.2
$\delta(x)$	0.02	0.75

Table 4.2: Hyper Parameter values of sampled MultiGP.

The correlation function $\rho(x)$ of each case is defined in table 4.3, where $\mathscr{H}(x)$ is the Heaviside function, which is equal to 1.0 if $x > 0$ else it is equal to 0. Note, that the non-linear correlation functions are chosen such that they have symmetric properties with respect to the vertical line $x = 1$, as the splitting and stitching methods are all split and stitched on this boundary. Figure 4.1 shows one sampled function of each case.

Logo	Abbreviation	Case Name	$\rho(x)$
	Case 1	Constant ρ Case	1
	Case 2	Discontinuous ρ Case	$1-2\mathscr{H}(x-1)$
	Case 3	Linearly Varying ρ Case	$1.0-x$

Table 4.3: Case-specific correlation function



(a) Constant ρ (b) Discontinuous ρ (c) Linearly varying ρ

Figure 4.1: One sampled function from MultiGP per case.

The additive part $\delta(x)$ is chosen to be non-zero to help against overfitting and to create more realistic experiments, as real data often has small variations in the correlation.

4.3. Data-sets

Each dataset consists of low-fidelity observations sampled from regions $[-1,0]$, $[0,1]$, $[1,2]$, and $[2,3]$, and of high-fidelity observations sampled from regions $[0,1]$ and $[1,2]$. All methods use this complete dataset in the optimization and prediction process. This setup is chosen because it features an interpolation regime $[0,2]$ and two extrapolation regimes $[-1,0]$ and $[2,3]$ with low- and no high-fidelity observations. These regions are symmetric around the line $x = 1$ because the cases are too.

Per sampled function, twelve types of datasets are created each with 20 datasets, the latter is to average out the biases inherent to sampling inputs. These types differ by their number of low- and high-fidelity observations and by the sampling strategy of the inputs. The different numbers of low-fidelity observations per region are 21 and 101, the different numbers of high-fidelity observations per region are 5, 10, and 20, and the inputs are either all linearly spaced or sampled from a uniform

distribution in each region. The targets of the low- and high-fidelity observations have an added i.i.d. Gaussian distribution $\epsilon \sim \mathcal{N}(0, 0.001)$. Figure 4.2 shows an example of a sampled function with one of its data sets from the constant ρ case.

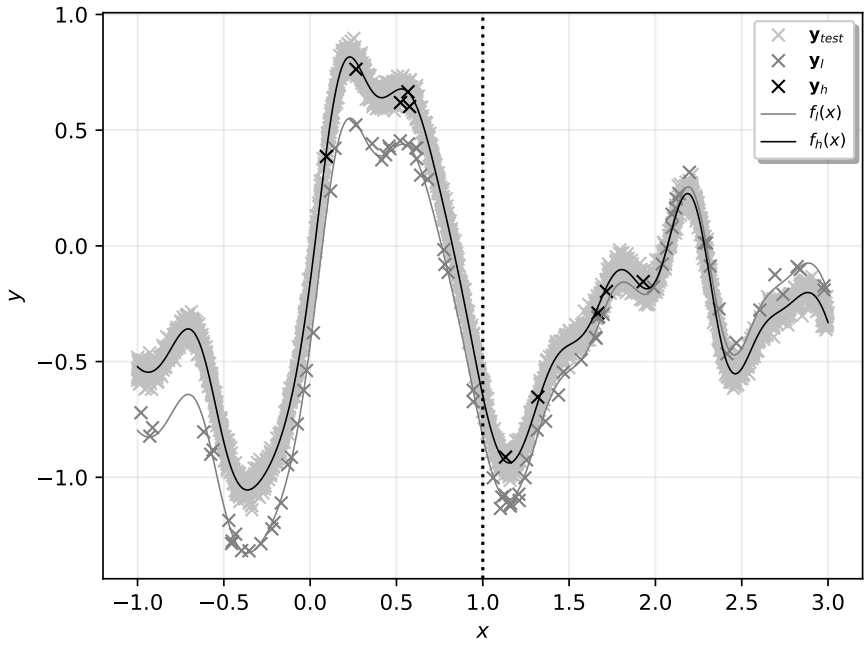


Figure 4.2: Sampled function and training and test dataset of the constant ρ case with the inputs sampled from a uniform distribution.

4.4. Performance

The average performance of a dataset type of a sampled function is defined as the expectation of the test error over its 20 datasets, see section 7.2 of Hastie et al. [14]. The chosen test error is the mean squared error

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \bar{y}_i)^2. \tag{4.4}$$

It is calculated using a test dataset $(\bar{\mathbf{x}}_i, \bar{y}_i) \in \mathfrak{D}_{\text{test}}$ that consists of 1001 observations per region, which are either linearly spaced or sampled from a uniform distribution (based on the sampling choice of the training dataset). In order to keep the training and testing procedures homogeneous, the i.i.d. Gaussian distribution $\epsilon \sim \mathcal{N}(0, 0.001)$ of the training dataset is also added to the test dataset. Figure 4.2 shows an example of a sampled function and a test dataset.

The performance is calculated across four different region divisions of the test datasets, regions: $[0, 2]$, $[-1, 0] \cup [2, 3]$, $[-1, 0]$, and $[2, 3]$. They, respectively, account for the interpolation regime and the total, left, and right extrapolation regime.

Further on in this thesis, the performance is shown in a compact plot similar to the one in Figure 4.3. Each row of this plot represents a method, and the x-axis represents the expectation of the mean squared error of a dataset type of a sampled function which is denoted by the three colored symbols. The colored symbols represent the number of high-fidelity observations: purple filled dot equals 20, blue dot equals 10, and green triangle equals 5. The letters denote the order of the sampled functions so that the performance of each sampled function can be compared across the methods. The case, the number of low-fidelity observations, and the sampling strategy are denoted in the caption of the figures and sometimes in the rows themselves.

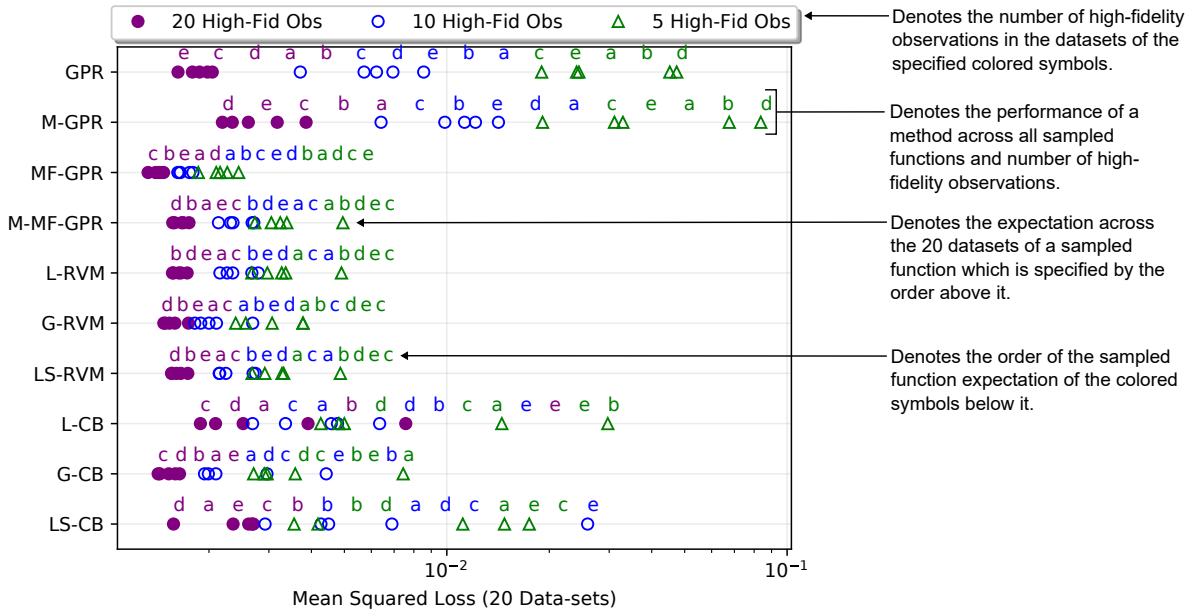


Figure 4.3: Example performance plot: the case, the number of low-fidelity observations, and the sampling strategy are specified in its caption.

5

Results & Discussion

This chapter shows the results of the experiments on the methods and discusses them. This chapter only looks at the MF-GPR, M-MF-GPR, RVM, and CB methods. First, the results across the three cases are shown in both the interpolation and extrapolation regimes. These two sections, only look at the experiments with 21 low-fidelity observations and 5 high-fidelity observations, because a lower number of observations shows the biggest difference between the methods. After this, the results of increasing the number of observations are shown and discussed. At last, the two different sampling strategies of the inputs are compared and discussed, namely, linearly spaced and sampling from a uniform distribution.

This chapter only looks at the local extension of the splitting and stitching methods to multi-fidelity, because their performance is similar when looking across the results. This is due to two reasons: the low-fidelity has enough observations and the underlying function of the low-fidelity is completely stationary for each case. Specifically, the local extension is chosen because it is the most natural extension, i.e., splitting each fidelity and stitching the highest.

In the figures, the number of low- and high-fidelity observations are abbreviated to "LFOs" and "HFOs" and the case names are renamed to:

- Constant ρ case → case 1
- Discontinuous ρ case → case 2
- Linearly varying ρ case → case 3

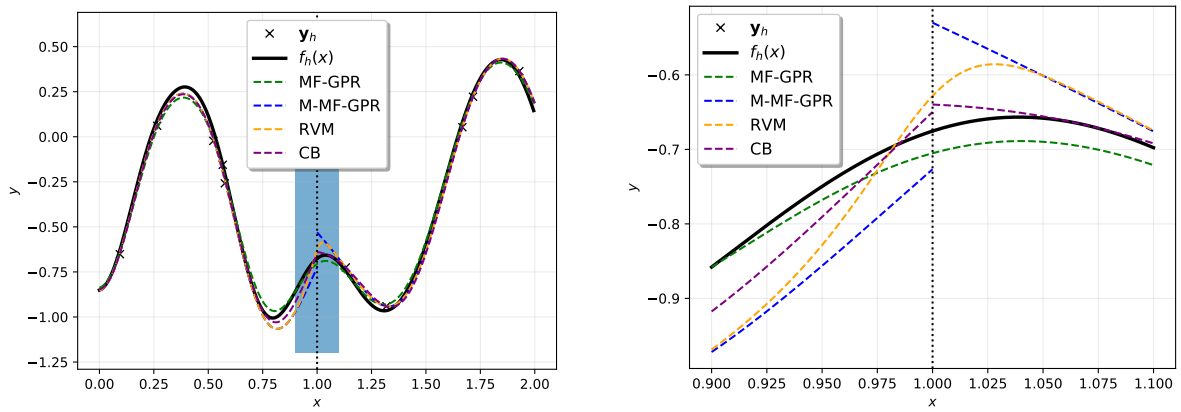
Appendix A shows the results of all experiments.

5.1. Interpolation Regime

This section discusses the results of the experiments across the three cases regarding the interpolation regime, which is region $[0, 2]$.

5.1.1. Constant ρ Case

Figure 5.1 shows the prediction of the four methods in the interpolation regime for a given dataset and sampled function. All methods show similar behavior across the regime except near the boundary where the differences between them become apparent. Figure 5.1b shows the predictions zoomed-in, the MF-GPR shows a continuous prediction, the M-MF-GPR is discontinuous, the RVM weighs the local models of the M-MF-GPR, and the CB constraints two independent models to be equal at the boundary of which still a small discontinuity is seen that represents the error term in the constraints. From these results, no general conclusions can be made even when looking at the actual mean squared error of each prediction, as it is a single representation of the case. However, it does confirm that the methods behave according to their expected behavior in stationary settings.

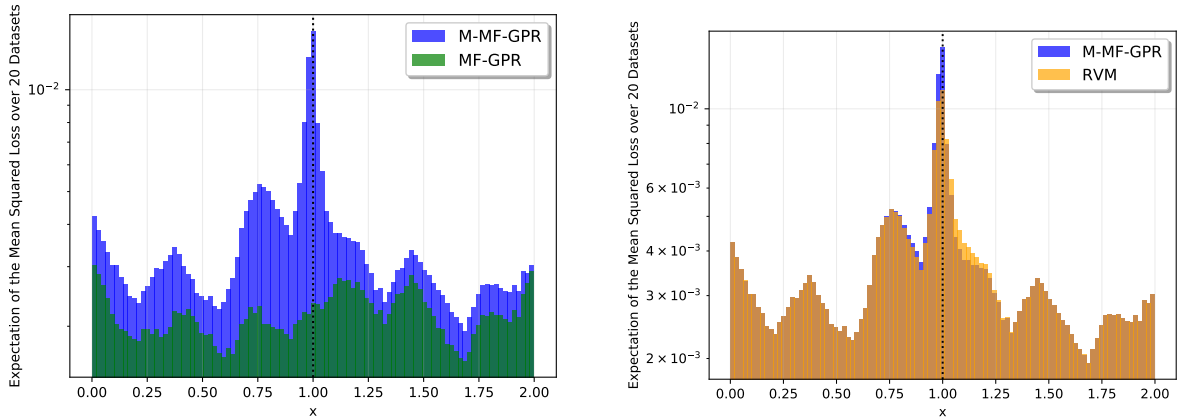


(a) interpolation regime: $[0,2]$.

(b) zoomed-in interpolation regime: $[0.9,1.1]$.

Figure 5.1: prediction comparison of splitting and stitching methods in constant ρ case: 21 LFOs, 5 HFOs, inputs sampled from a uniform distribution, function b, dataset id 0.

To bridge the gap between the specific predictions and the results over multiple realizations, the expectation of the mean squared error, over all sampled functions and datasets of the constant case, in regions of the interpolation regime is shown in a bar plot in figure 5.2. These bar plots provide a localized view of the prediction accuracy and show which regions have relatively larger errors. Subfigure 5.2a compares the M-MF-GPR and the MF-GPR, it shows that splitting increases the error across the regime and that it causes a large error spike at the boundary, see $x = 1$. These results confirm the results found in the literature: splitting a model reduces its ability to capture global stationary features, and the spike is caused by the fact that the boundary lies in the extrapolation regime of the local models, for which GPRs are known to go to their prior which causes the prediction to become worse. Subfigure 5.2b compares the M-MF-GPR and the RVM, it shows that the stitching of the RVM method only affects the predictions near the boundary which causes their error to decrease. This effect is due to the stitching making the M-MF-GPR continuous, as the underlying function is continuous, this improves the prediction accuracy. Note, that there is a slight increase on the right side of the boundary, but as the bar plot is in log-scale the spike at the boundary is substantially larger, and thus the stitching reduces the error overall.



(a) comparison M-MF-GPR and MF-GPR

(b) comparison M-MF-GPR and RVM

Figure 5.2: barplot performance comparison splitting and stitching in interpolation regime $[0,2]$ in constant ρ case:21 LFOs, 5 HFOs, inputs sampled from a uniform distribution.

To conclude this section, the expectation of the error of each sampled function over all datasets over the whole interpolation regime of the four methods is compared in figure 5.3. This provides a measure of how well each method performs in the interpolation regime with which they can be

compared and general conclusions about the constant case can be made. The results in the figure reflect the results in the bar plots, i.e., splitting increases the overall error, RVM slightly decreases the error compared to splitting, and CB (not yet previously discussed) performs worse than the other three methods. The figure not only shows that these are true when averaged over the sampled functions, which the bar plots already showed, but that they are more generally true for each sampled function. Thus, the results in the constant case are comparable to those found in the literature, as discussed in the previous paragraph.

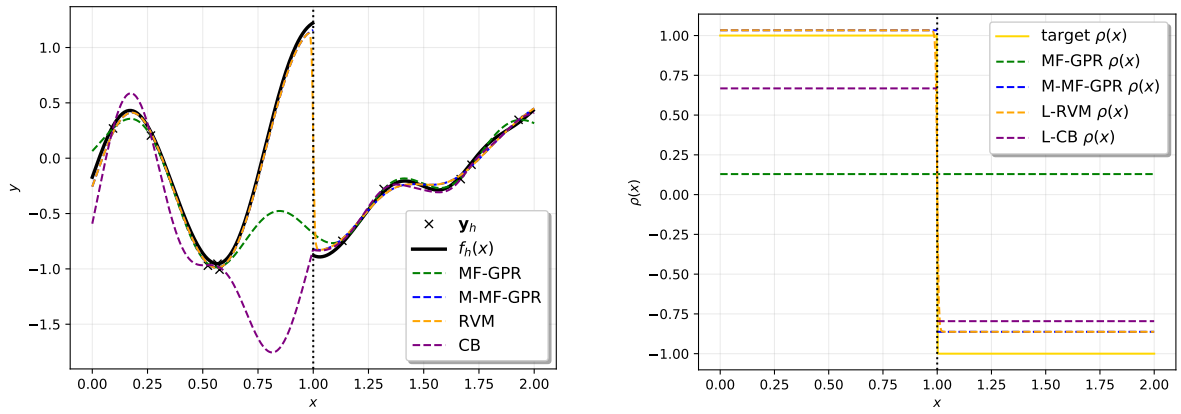
Figure 5.3: splitting and stitching performance in interpolation regime $[0,2]$ in constant ρ case: 21 LFOs, 5 HFOs, inputs sampled from a uniform distribution.

5.1.2. Discontinuous ρ Case

The discontinuous case consists of a discontinuity in the high-fidelity, which is modelled by placing a discontinuity in the correlation between the two fidelities. This case enables the comparison of the four methods in a piecewise constant correlation setting for which it is suspected that splitting and stitching will outperform the MF-GPR.

First, the prediction of each method on a specific sampled function and dataset in the interpolation regime is compared for the discontinuous case. Figure 5.4 shows these predictions and their corresponding correlation coefficients. The MF-GPR follows the underlying function closely except in region $[0.6;1.0]$. In this region there are no high-fidelity observations, meaning that the high-fidelity prediction of the MF-GPR goes to its prior which is the low-fidelity prediction times the correlation coefficient. The correlation coefficient of the MF-GPR is just above zero which is inaccurate in this region, where the actual correlation is equal to 1.0; the MF-GPR makes a compromise of the correlation between the left and right side of the boundary as it is only able to model a constant correlation across the input space. Due to this, the prediction is inaccurate in that region. In contrast, the M-MF-GPR is able to model the underlying function closely even in that region. Due to the splitting, it models the correlation correctly on both the left and right side of the boundary (as shown in figure 5.4b), thus

making the model capable of producing correct predictions at locations with sparse high-fidelity observations in this correlation setting. Compared to the constant case, the RVM makes a worse prediction compared to the M-MF-GPR, as it makes the prediction continuous at the boundary that is discontinuous. However, the transition zone between the left and right local model is small, so the increase in total error is minimal. At last, the CB method performs considerably worse in the beforementioned region $[0.6; 1.0]$.



(a) predictions.

(b) correlation coefficient.

Figure 5.4: prediction comparison of splitting and stitching methods in interpolation regime $[0,2]$ in discontinuous ρ case: 21 LFOs, 5 HFOs, inputs sampled from a uniform distribution, function b, dataset id 0.

As with the constant case, one prediction does not provide accurate information regarding the performance of these methods in a general discontinuous correlation setting. Figure 5.5 provides a general view of the performance by showing the expectation of each sampled function over their datasets. In this figure, the findings from the predictions are seen quantitatively. It shows that splitting and stitching (RVM) reduces the error significantly compared to the MF-GPR, where the stitching has a higher error compared to splitting. As mentioned earlier, this is due to these methods being capable of capturing a two-piecewise constant correlation which the MF-GPR is incapable of, and of course, stitching is unnecessary at a discontinuity. The CB has the highest error compared to the other three. The results confirm the expectations that splitting and stitching (only RVM) outperform the MF-GPR in the interpolation regime.

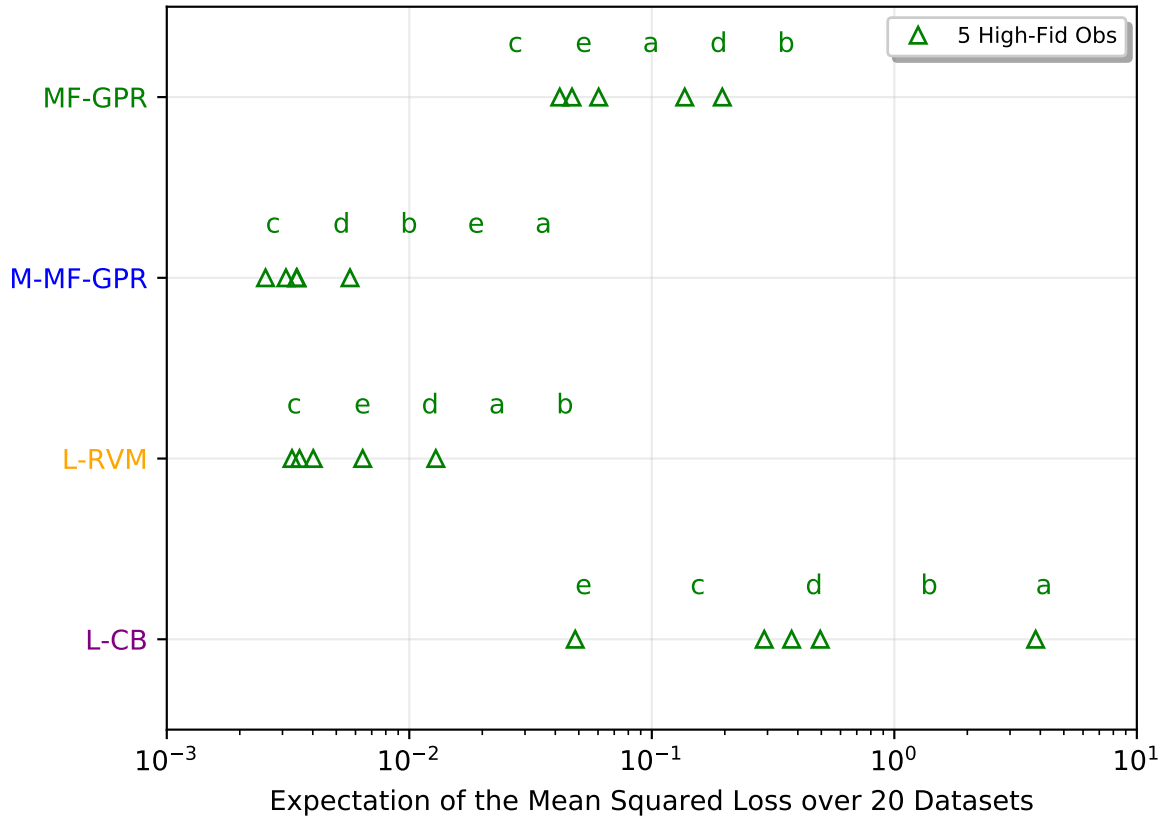


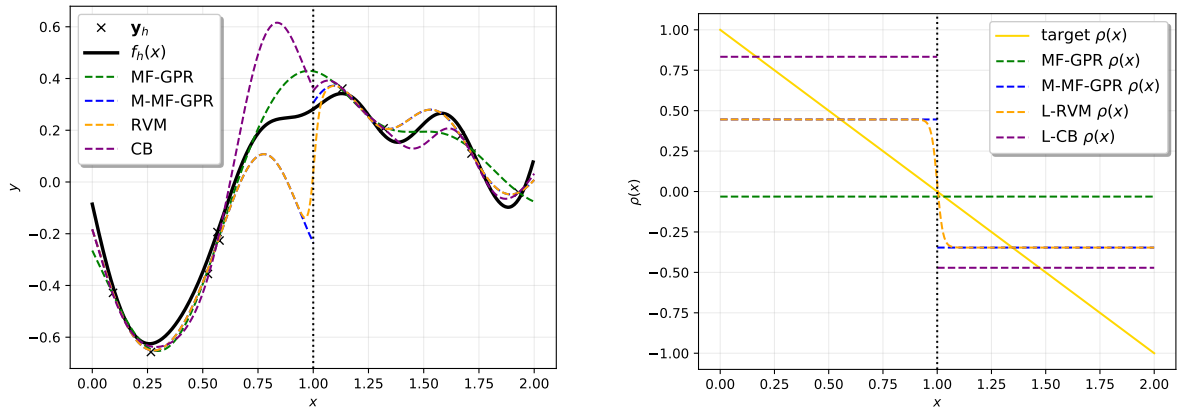
Figure 5.5: splitting and stitching performance in interpolation regime [0,2] in discontinuous ρ case: 21 LFOs, 5 HFOs, inputs sampled from a uniform distribution.

5.1.3. Linearly Varying ρ Case

At last, the interpolation regime of the linearly varying case will be looked at. This case is more nonlinear than the other cases and the correlation can not be captured by the methods exactly, as they can only model constant correlations or a weighing between them, in the case of the RVM method. However, it is expected that the splitting and stitching methods outperform the MF-GPR, as their two piecewise constant correlation matches a linearly varying one more closely than a constant correlation.

First, the prediction of a specific sampled function and dataset is compared with each method. They all show different behavior in regions [0.5;1.2] and [1.3;2.0], therefore it is hard to see which method outperforms the others. These differences correspond to their different correlation coefficients, as seen in subfigure 5.6b. The MF-GPR has a near-zero correlation which indicates that the low-fidelity observations play no role in the high-fidelity prediction. The M-MF-GPR and RVM have a correlation of 0.5 on the left and -0.3 on the right, the correlation on the left is to be expected as it is the average correlation in that region, however, this does not seem to be the case for the left side of the boundary. Two main reasons might be identified: the average location of the inputs, and the additional term added in the MF-GPR that is used to sample the functions for the datasets, see section 4.2. At last, the CB has a correlation of 0.8 and -0.5 on the left and right, respectively. These might be due to the aforementioned reasons with M-MF-GPR and RVM, or due to the constraints that need to be satisfied in the optimization of the hyperparameters. Thus, for all methods it is difficult to see how well each method performs in this dataset, and thus this case, as they all can not fully capture the interaction between the two fidelities in this correlation setting. Therefore, it is even more important for, this case compared to the others, to look at the expected error of each sampled function across their datasets.

Figure 5.7 compares the expected error of each sampled function of each method. Visually, it is hard to see which method performs better than the others. This is similar to the inspection of the prediction, where each method shows different behavior, and all seem to have trouble inferring the



(a) predictions.

(b) correlation coefficient.

Figure 5.6: prediction comparison of splitting and stitching methods in interpolation regime $[0,2]$ in linearly varying ρ case: 21 LFOs, 5 HFOs, inputs sampled from a uniform distribution, function b, dataset id 0.

correlation between the two fidelities. This again is not expected as the splitting and stitching should in theory better capture the correlation. When comparing the actual expectations of each sampled function, the splitting and RVM perform better than MF-GPR with RVM being the best, and the CB methods perform worse than the others. However, these differences are quite small when comparing it to the total error found, i.e., errors are between 10^{-2} and 10^{-1} while the average difference between the expected errors is around $4 * 10^{-3}$. It is suspected that their similar performance, is mainly due to their inability to correctly infer the correlation between the fidelities, and thus the addition of the low-fidelity does not significantly improve the prediction accuracy in the high-fidelity. Thus, from these results, no conclusion can be made on which method outperforms the other regarding the linearly varying case, as they all perform almost similarly when looking at their individual and average performance.

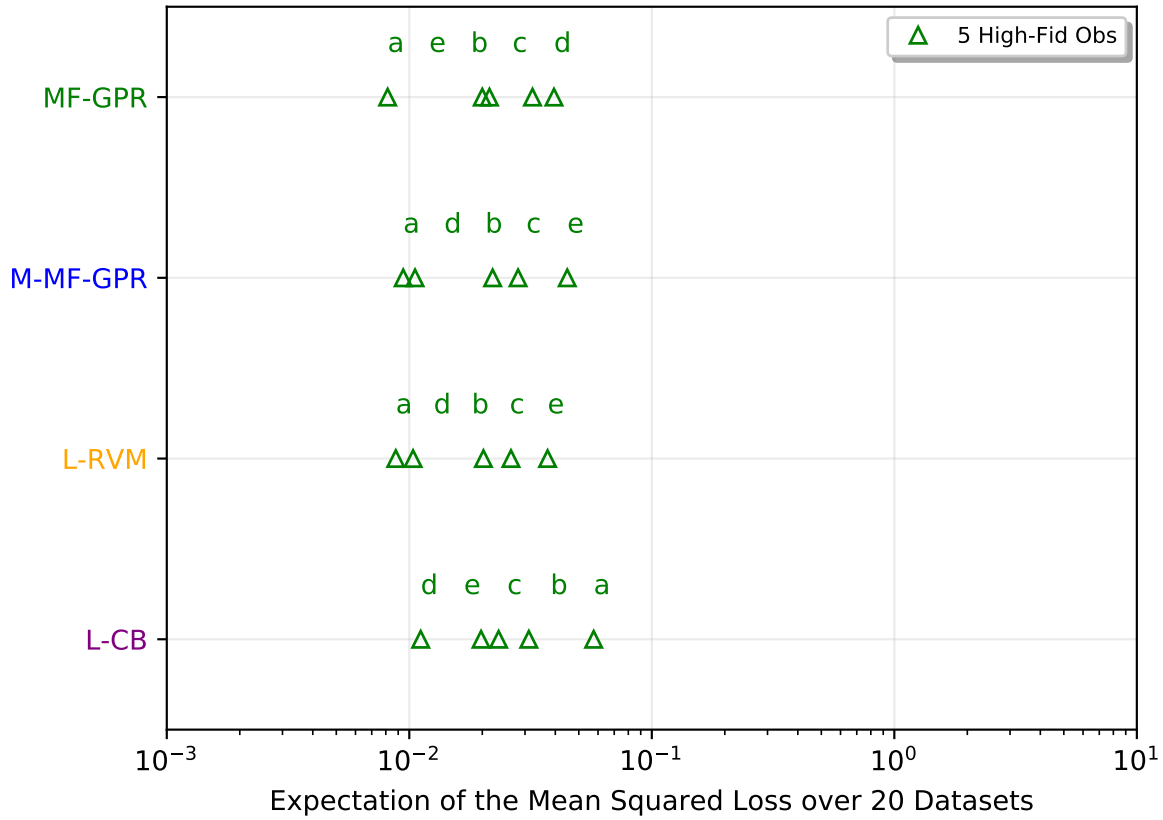


Figure 5.7: splitting and stitching performance in interpolation regime $[0,2]$ in linearly varying ρ case: 21 LFOs, 5 HFOs, inputs sampled from a uniform distribution.

5.2. Extrapolation Regime

This section discusses the results of the experiments across the three cases regarding the extrapolation regime, which is region $[-1,0] \cup [2,3]$. Here, the extrapolation regime means that there are low-fidelity observations but no high-fidelity ones.

5.2.1. Constant ρ Case

Figure 5.8 compares the expected error of each sampled function of each method in the extrapolation regime of the constant case. The trends in this figure follow the trends in the same figure of the interpolation regime: MF-GPR outperforms splitting and RVM, and Cb has a larger error compared to the other three. However, the RVM produces an equal error instead of it being marginally smaller, because the weighing of the local models only affects the prediction near the boundary, as it has a fast transition zone. Thus, the extrapolation regime is unaffected by the weighing of the RVM method. The errors are larger compared to the interpolation regime because the extrapolation regime consists only of low-fidelity observations. So the predictions go towards the prior of the high-fidelity which is equal to the low-fidelity prediction times the correlation coefficient.

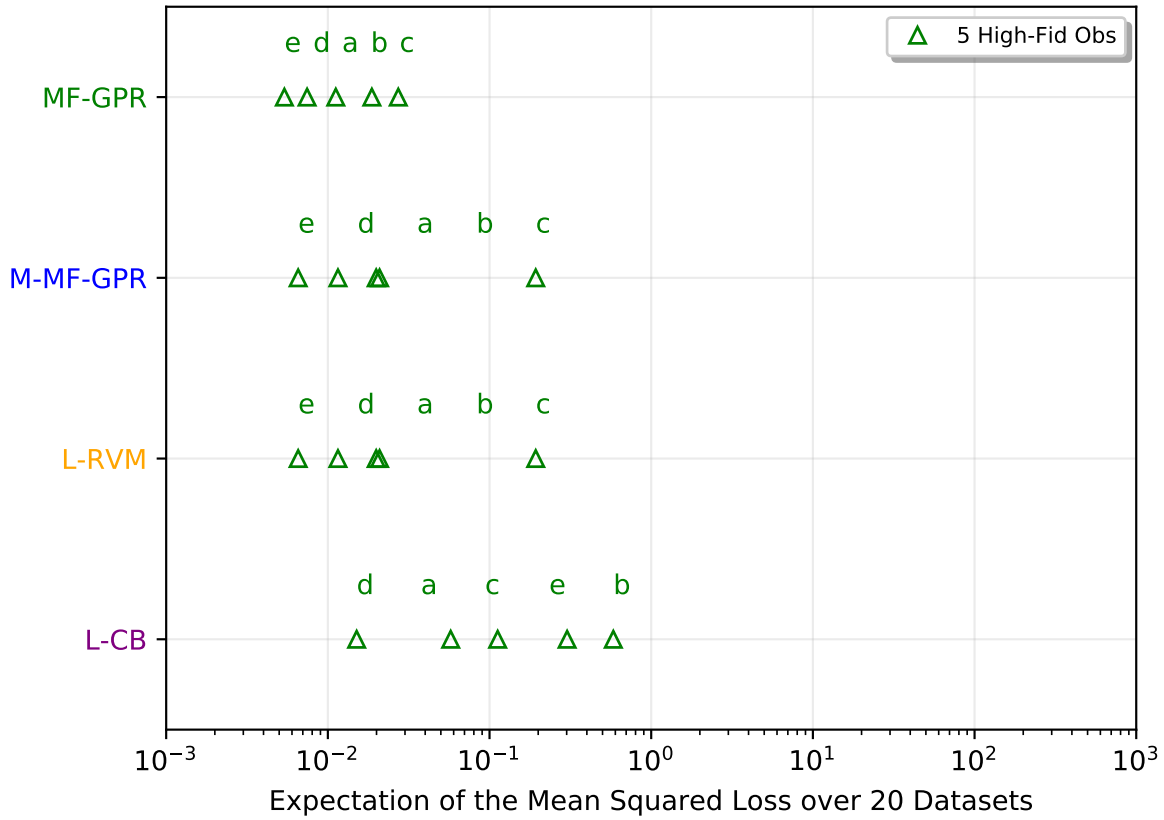


Figure 5.8: splitting and stitching performance in extrapolation regime $[-1,0] \cup [2,3]$ in constant ρ case: 21 LFOs, 5 HFOs, inputs sampled from a uniform distribution.

5.2.2. Discontinuous ρ Case

Figure 5.9 compares the expected error of each sampled function of each method in the extrapolation regime of the discontinuous case. In this case, the differences between the interpolation regime and the extrapolation regime are similar to their differences in the constant case. The trends in the extrapolation regime follow the trends in the interpolation regime: splitting and RVM improve over MF-GPR, while CB performs worse. Where, similarly to the constant case, splitting and RVM have equivalent errors, due to the stitching only affecting the prediction near the boundary. The trends are similar between both regimes because the performance of each method is tied to how well they can capture the target correlation of the dataset, which holds because the predictions in the extrapolation regime go towards the high-fidelity prior.

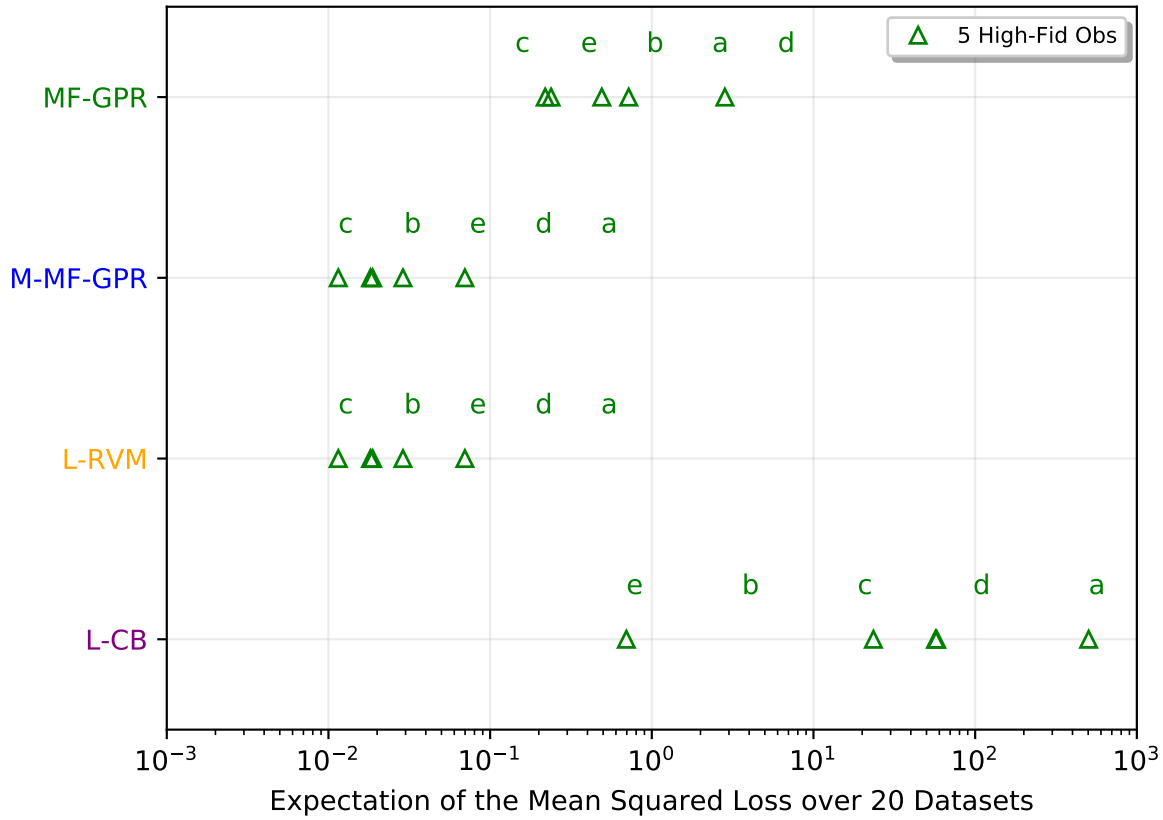


Figure 5.9: splitting and stitching performance in extrapolation regime $[-1,0] \cup [2,3]$ in discontinuous ρ case: 21 LFOs, 5 HFOs, inputs sampled from a uniform distribution.

5.2.3. Linearly Varying ρ Case

Figure 5.10 compares the expected error of each sampled function of each method in the extrapolation regime of the linearly varying case. The error of the splitting and RVM are equivalent but smaller than the MF-GPR, while the CB has a larger error than the other three. The equivalent error between the splitting and RVM is, again, due to the fast transition zone, which is also seen in the constant and discontinuous case. These results are in contrast to the results in the interpolation regime, where all four methods perform almost similarly. In that regime, the trend seen in the extrapolation regime is also present when looking at the actual values of the errors, however, it is argued that in the interpolation regime, these differences are negligible, as they are very small. However, this is not the case in the extrapolation regime. It seems that splitting and RVM are able to more accurately capture the extrapolation regime, due to the increased capability of capturing non-linear correlation. Thus, the mapping from the low-fidelity onto the high-fidelity is playing a more important role in the extrapolation compared to the interpolation regime in this case.

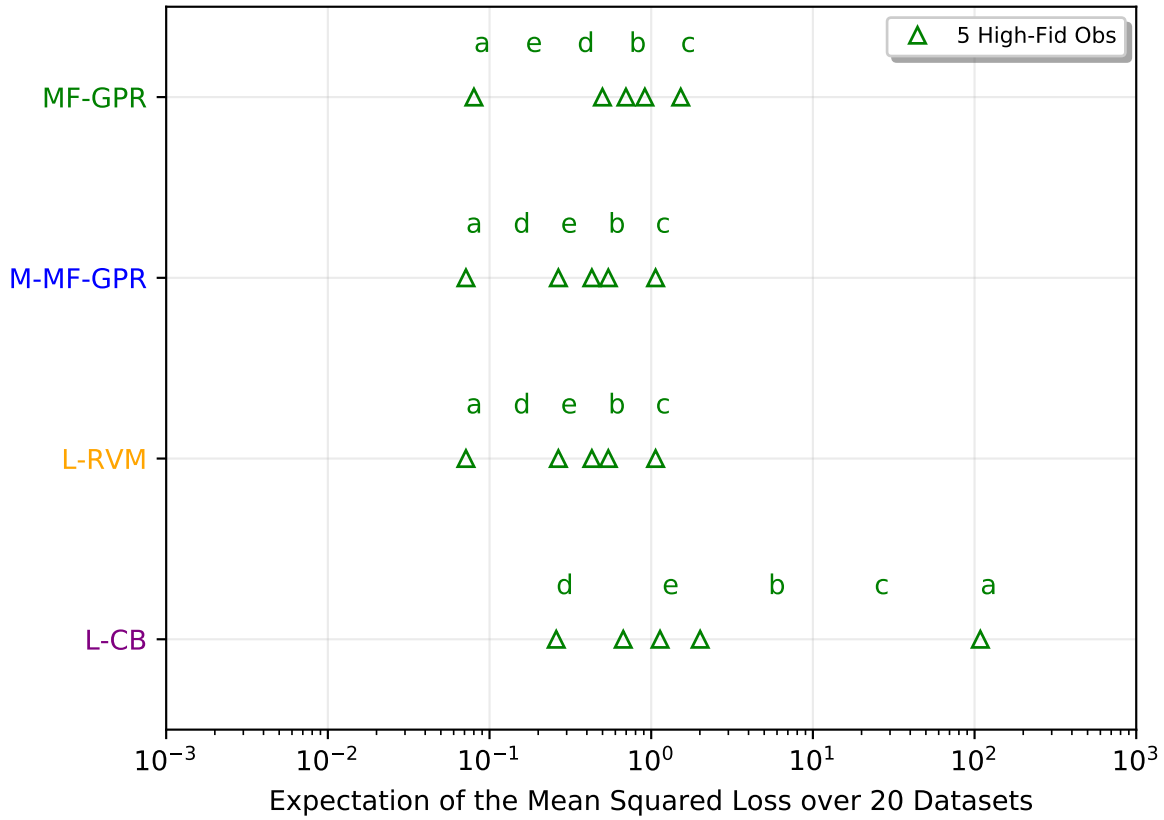


Figure 5.10: splitting and stitching performance in extrapolation regime $[-1,0] \cup [2,3]$ in linearly varying ρ case: 21 LFOs, 5 HFOs, inputs sampled from a uniform distribution.

5.3. Number of Observations

This section shows and discusses the results when increasing the number of low- or high-fidelity observations. Only, the constant case is being looked at. However, the interpolation regime and extrapolation regime are looked at separately as different behavior is seen.

5.3.1. Interpolation Regime

Figure 5.11 shows the expectation of the error of each sampled function over all datasets of each method for each number of low- and high-fidelity observations in the interpolation regime. The three subfigures of figure 5.12 show the same expectations but each subfigure only shows the experiments with a particular number of high-fidelity observations because the trends are more clearly seen in these subfigures compared to figure 5.11. These figures show that increasing the number of low-fidelity observations decreases the error. This is consistent with the results from the literature, as increasing the number of observations, generally, increases the prediction accuracy. It also confirms, that when the low-fidelity is captured more correctly, that this benefits the capturing of the high-fidelity. These figures also show that increasing the number of high-fidelity observations decreases the error. Similarly to the number of low-fidelity observations, this is in line with the expectation. The trends of the performance as seen for each number of low- and high-fidelity observations are similar to that discussed in the section on the interpolation regime of the constant case: splitting and RVM have a higher error compared to the MF-GPR, where RVM is slightly better than splitting, and the CB has a higher error compared to the other three. Thus, all four methods behave as expected under the different numbers of low- and high-fidelity observations.

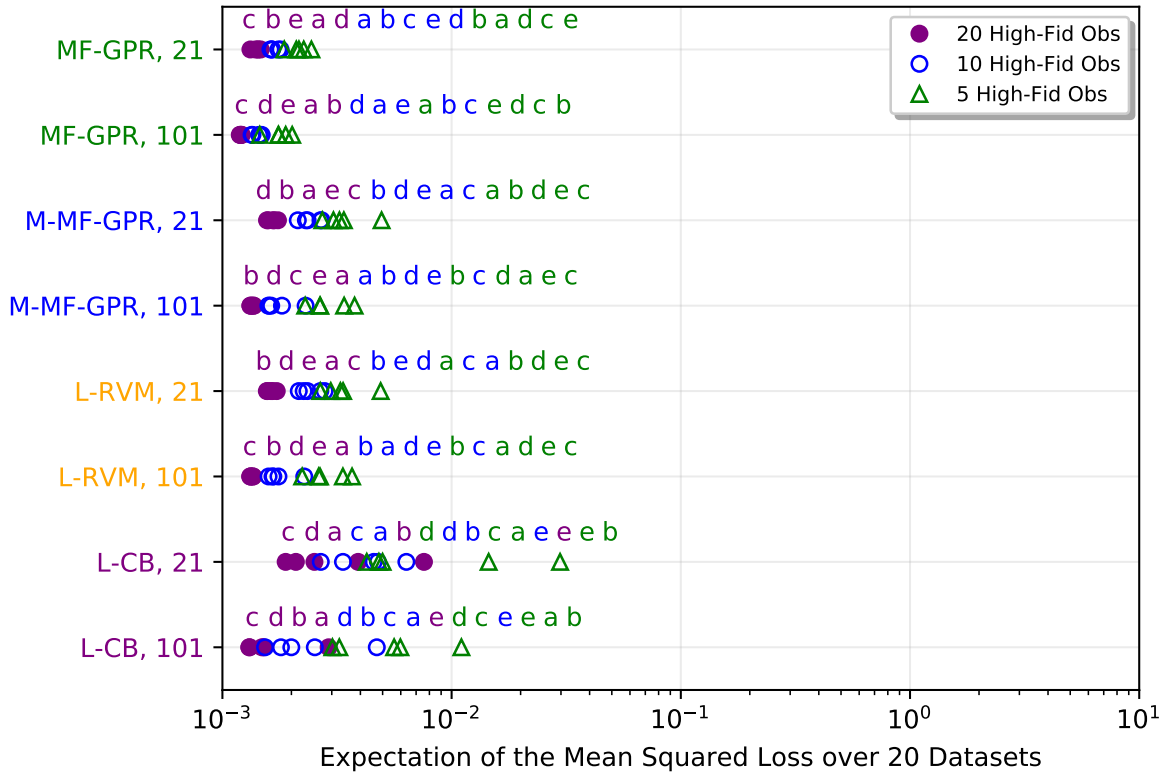
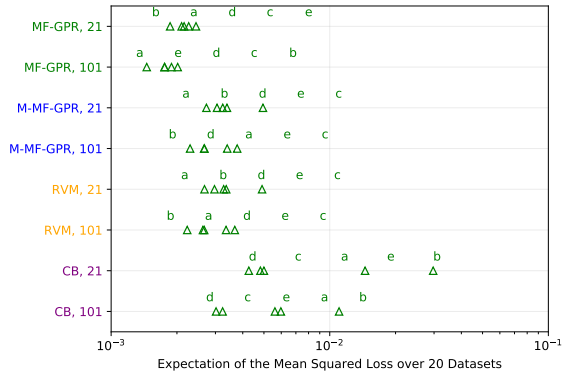
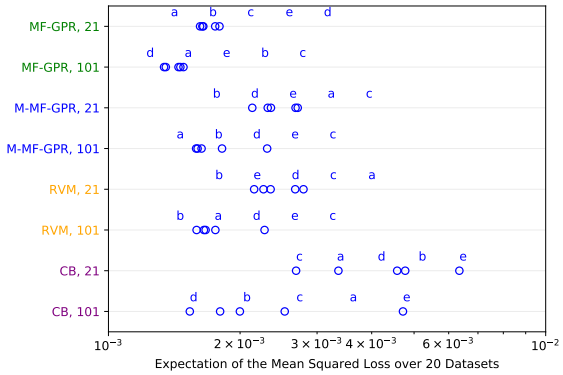


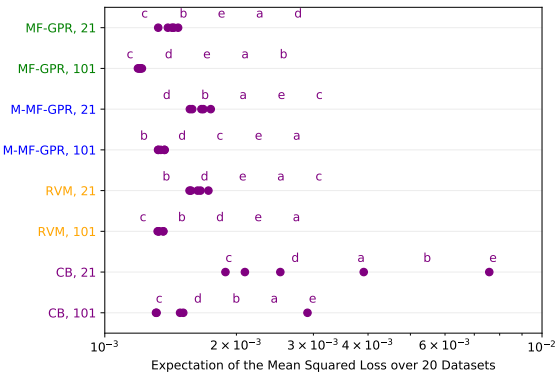
Figure 5.11: performance comparison number of low- and high-fidelity observations in interpolation regime $[0,2]$: constant case, inputs sampled from a uniform distribution.



(a) 5 HFOs.



(b) 10 HFOs.



(c) 20 HFOs.

Figure 5.12: performance comparison number of low-fidelity observations in interpolation regime $[0,2]$: constant case, inputs sampled from a uniform distribution. Results are equivalent to those in figure 5.11 but the different number of HFOs are shown separately.

5.3.2. Extrapolation Regime

Figure 5.13 shows the expectation of the error of each sampled function over all datasets of each method for each number of low- and high-fidelity observations in the interpolation regime. Similarly, figure 5.14 shows the same errors but the experiments with different numbers of high-fidelity observations can be viewed separately, for clarity. In the extrapolation regime, the same behavior is seen across the experiments with different numbers of low- and high-fidelity observations as in the interpolation regime. Increasing both the number of low- and high-fidelity observations decreases the error and with these increases, the trends between the methods do not change (error trend: MF-GPR < RVM & M-MF-GPR < CB). However, the CB method is an exception to this, as the experiments with 21 LFOs and 5 HFOs outperform on average the experiments with 21 LFOs and 10 or 20 HFOs. This exception is mainly attributed to the outlier that is sampled function **b**, as without it, the experiments with 5 HFOs perform worse than the experiments with 10 or 20 HFOs on average. Thus, in general, all four methods behave as expected under different numbers of low- and high-fidelity observations.

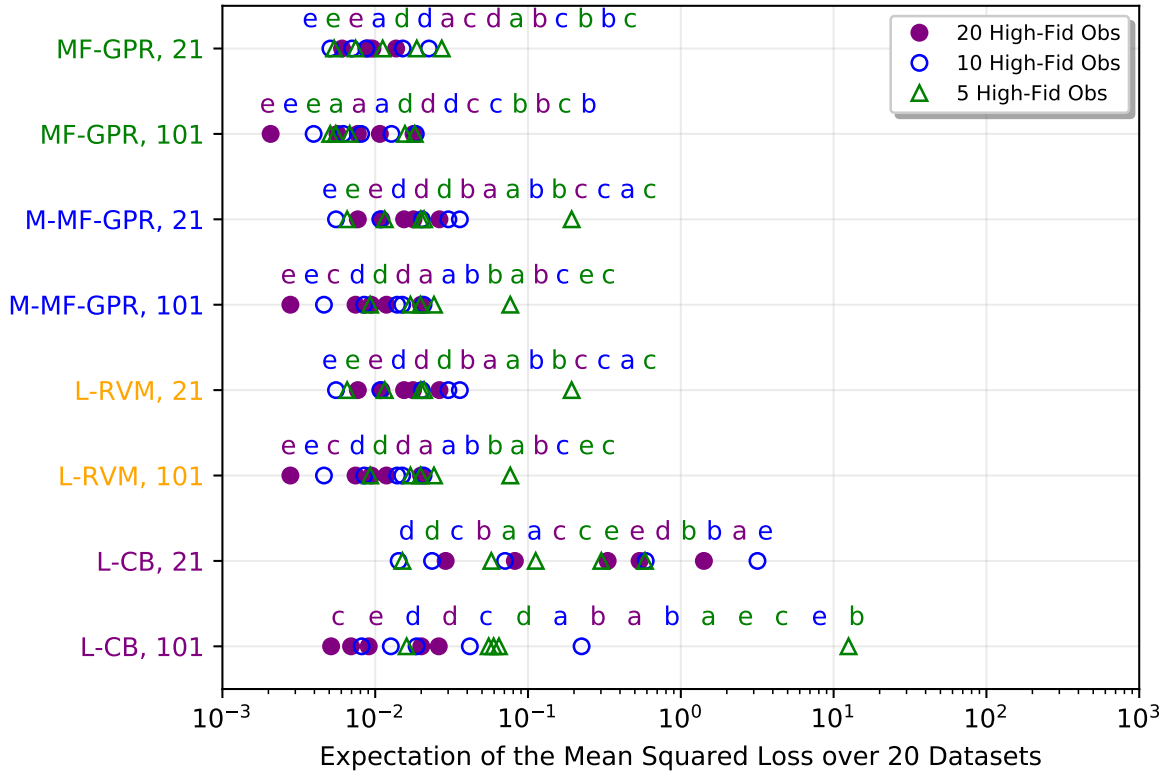
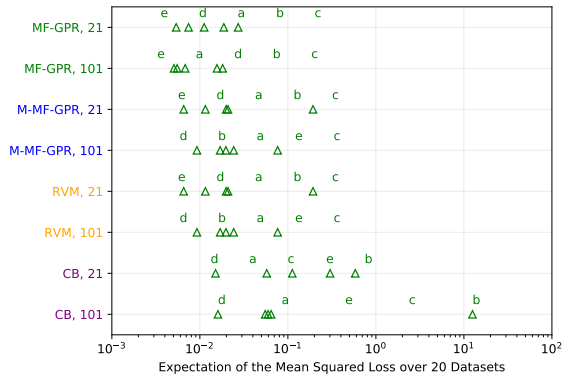
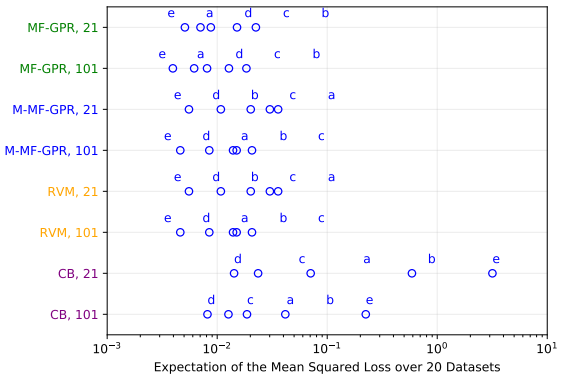


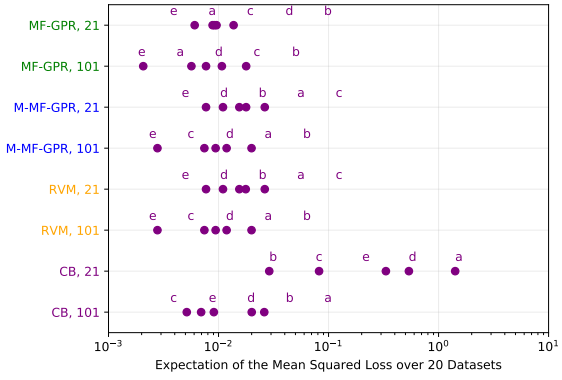
Figure 5.13: performance comparison number of low- and high-fidelity observations in extrapolation regime $[-1,0] \cup [2,3]$: constant case, inputs sampled from a uniform distribution.



(a) 5 HFOs.



(b) 10 HFOs.



(c) 20 HFOs.

Figure 5.14: performance comparison number of low-fidelity observations in extrapolation regime $[-1,0] \cup [2,3]$: constant case, inputs sampled from a uniform distribution. Results are equivalent to those in figure 5.13 but the different number of HFOs are shown separately.

5.4. Sampling Strategy

This section compares the two different sampling strategies of the inputs, linearly spaced and sampled from a uniform distribution. The comparison is made in the interpolation regime of the constant case.

Figure 5.15 shows the expectation of the error of each sampled function over all datasets of each for the two different sampling strategies, linearly spaced and sampled from a uniform distribution, in the interpolation regime, the latter is denoted in the plot as "uniformly distributed". Similarly, to the comparison of the number of observations, figure 5.16 shows the experiments with different numbers of high-fidelity observations separately in three different subfigures. From these figures, it is evident that all experiments with the inputs linearly spaced outperform their counterpart experiments where the inputs are sampled from a uniform distribution. This is attributed to a more consistent coverage of the interpolation regime when the inputs are linearly spaced, because when the inputs are sampled from a uniform distribution, often the gaps between the observations become larger, meaning that some regions have more and some have fewer observations. GPR methods are more inaccurate in regions with sparse observations, and thus the prediction accuracy decreases when the inputs do not consistently cover the interpolation regime. The trends seen in the constant case in the interpolation regime when the inputs are sampled from a uniform distribution are still present when the inputs are linearly spaced (error trend: MF-GPR « RVM < M-MF-GPR « CB). Thus, when the inputs cover the interpolation regime more consistently, the prediction accuracy increases for all methods.

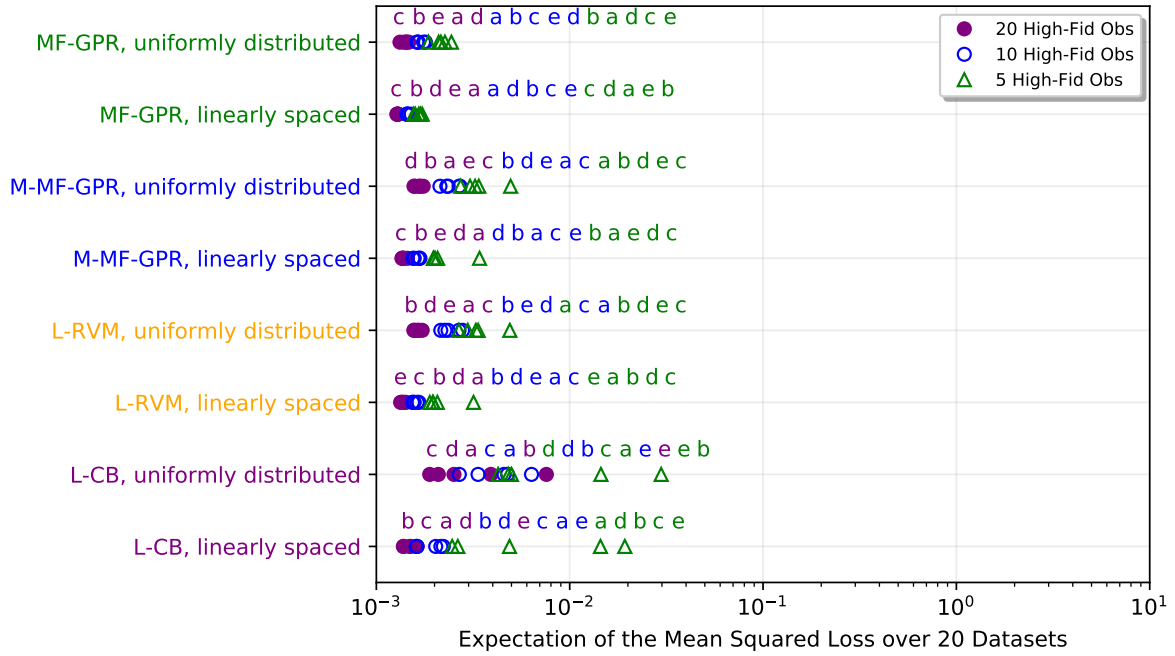
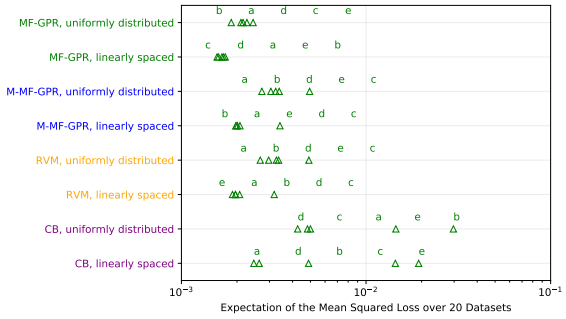
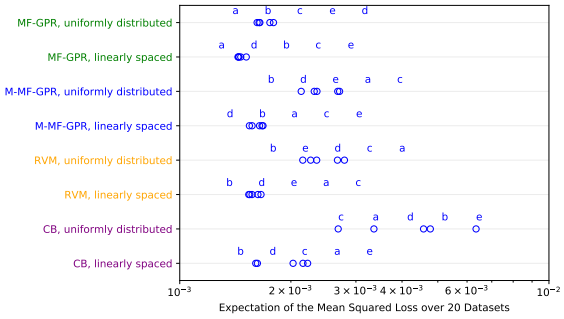


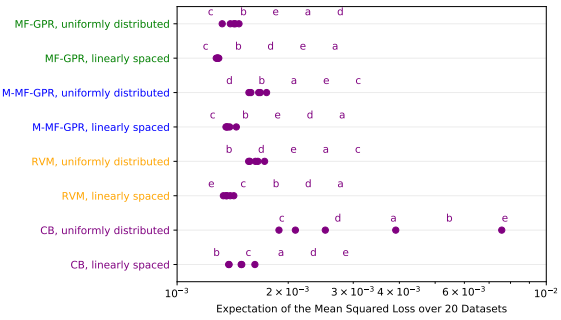
Figure 5.15: performance comparison sampling strategy in interpolation regime $[0,2]$: constant case, 21 LFOs.



(a) 5 HFOs.



(b) 10 HFOs.



(c) 20 HFOs.

Figure 5.16: performance comparison sampling strategy in interpolation regime $[0,2]$: constant case, 21 LFOs. Results are equivalent to those in figure 5.15 but the different number of HFOs are shown separately.

6

Conclusions and Future Work

This chapter draws conclusions for the research questions using the results and discussions of the experiments on the splitting and stitching multi-fidelity GPR methods. With these, a better understanding of splitting and stitching in multi-fidelity settings is made, such that this is a first step in the direction of applying these techniques as surrogates of micro-mechanical models of composites in the FE^2 method. This is necessary as the fidelities of these models have non-linear correlations. In the end, this can further decrease the computational cost of the FE^2 method, by reducing the time spent collecting observations, as fewer are needed, and training the surrogate model.

To bridge the gap between, the methods in this thesis and the actual application of splitting and stitching on surrogates, the future work section discusses possible avenues of research that enhance the understanding of this application and that must be performed before these methods can be applied as a surrogate.

6.1. Conclusions of the Research Questions

RQ1: What is the effect of splitting and stitching on the prediction accuracy of GPR methods with multi-fidelity in a linear and non-linear correlated setting in the interpolation and extrapolation regime?

Note, the conclusions of this research question assume a thoroughly sampled low-fidelity, therefore these are drawn using only the experiments with 21 low-fidelity and 5 high-fidelity observations. Conclusions regarding the experiments with 101 low-fidelity and 10 and 20 high-fidelity observations are considered in the next research question.

In the constant correlation setting in the interpolation regime, the act of splitting and stitching reduces the prediction accuracy. This reduction is attributed to the decrease in the number of observations per model in the method, as with the splitting and stitching methods they must be divided among the local models. This limits the effectiveness of capturing the setting's global stationary features, which mostly manifests as inaccurate predictions at the boundary, thus reducing the effectiveness of these methods. RVM does improve over splitting, however, the difference in prediction accuracy is really insignificant because only the predictions around the boundary are slightly improved. Splitting and RVM have a negligible effect on the extrapolation regime as on average a very slight reduction in the prediction accuracy is seen. The correlation between an interpolation region and the opposite extrapolation region is low, therefore the high-fidelity observations in the interpolation region do not contribute much to the opposite extrapolation region. The CB method performs worse than the others in both the interpolation regime and the extrapolation regime. Thus, in the linear setting for both regimes, the act of splitting and stitching is not preferred.

In the discontinuous correlation setting in both the interpolation and extrapolation regime, the act of splitting and the usage of the RVM stitching method increases the prediction accuracy while the CB method decreases it, even further compared to the non-split MF-GPR. Clearly, the splitting's effectiveness originates from its ability to capture a piecewise constant correlation. The RVM stitching

method performs slightly worse than just splitting because it creates a continuous model while the case is discontinuous. Thus, the decrease is mainly attributed to the misprediction of the boundary. The worse performance of the CB method is due to it satisfying continuity at a discontinuous boundary which is clearly unnecessary. Thus, in a discontinuous non-linear setting it is advantageous to use splitting, but enforcing continuity by means of stitching does not obviously improve the prediction accuracy.

In the linearly varying correlation setting in the interpolation regime, the act of splitting and stitching is on average not changing the prediction accuracy. The correlation coefficient of all models of each method is near zero. When splitting, the correlation of both local models is biased slightly positive and negative. Still, for each method, this means that the low-fidelity has minimal impact on the high-fidelity prediction in the interpolation regime. In the extrapolation regime, splitting and RVM have on average improved prediction accuracy. This is probably attributed to the slightly positive and negative bias that is seen in the correlation coefficient of both local models. CB performs worse than the other methods in the extrapolation regime. Thus, in a linearly varying non-linear correlation setting, splitting is preferred as it reduces the complexity of the model whilst keeping the prediction accuracy. Stitching is only preferred when the model needs to be continuous as it does not improve the prediction accuracy whilst having more complexity.

RQ2: What is the influence of the ratio of the number of low and high-fidelity observations on the prediction accuracy of splitting and stitching methods?

In general, in the interpolation regime of the constant case, the prediction accuracy increases when the number of high-fidelity observations increases. The conclusions drawn regarding the previous research question are still valid when increasing the number of low- and high-fidelity observations in this situation.

In general, in the extrapolation regime of the constant case, the prediction accuracy increases when the number of high-fidelity observations increases. The conclusions drawn regarding the previous research questions are still valid when increasing the number of low- and high-fidelity observations. The CB method has an exception, in that it has an outlier with 101 LFOs and 5 HFOs which cause it to perform better with 5 HFOs compared to 10 and 20.

RQ3: What is the influence of the sampling strategy of the input space on the prediction accuracy of splitting and stitching methods?

In general, in the interpolation regime of the constant case, the performance is better when the inputs are linearly spaced compared to sampled from a uniform distribution. This is mostly attributed to the denser coverage that the linearly spaced inputs have when comparing them to the inputs that are sampled from a uniform distribution with the same number of observations.

6.2. Splitting and Stitching Methods as Surrogates for Micro-mechanical Models

These conclusions provide a direction for future research on how to apply splitting and stitching techniques on GPRs with multi-fidelity as surrogates of micro-mechanical models in the FE^2 method. The conclusions specify the features that the correlation between the fidelities of the micro-mechanical model needs to have when splitting or stitching might be beneficial to use. Although these features are now known, it is unknown how the correlation of the fidelities of a micro-mechanical model behaves. Therefore, it is difficult to recommend the usage of splitting and stitching in general for this application. More research is needed, to characterize the correlations of these fidelities and to test splitting and stitching as such a surrogate. However, before splitting and stitching can be applied as a surrogate a few improvements are necessary to make, namely:

High-dimensional Input Spaces

This thesis only considers a one-dimensional input space. However, most surrogates of micro-mechanical models seek a mapping from the strain tensor to the stress tensor, or vice versa. These tensors have generally more than one element. This means the surrogates must have input spaces of dimensions greater than one. Note, that the target space also has a dimension greater than one. One possible solution for this is to create an independent model for each output component [45].

The splitting and stitching methods require some changes when considering these input spaces. Most importantly, the boundary between the local models changes from a point to an d-dimensional hyperplane. With this change, the CB-GPR can not achieve full continuity across the boundary as continuity can only be satisfied in a finite number of points, and its computational cost scales poorly because the number of constraining points must grow exponentially with the dimension to let the boundary be sufficiently continuous. The increase in dimension also requires the RVM-GPR to change its weight function, as its center parameter can not sufficiently describe the boundary anymore.

Clustering the Local Models

Currently, the determination of the regions of the local models in the splitting and stitching methods is predetermined. However, in real-world applications, the underlying function is not known prior, therefore a clustering algorithm must be applied to determine these regions. This results in a few extra considerations. First, if clustering will take place in the same process as or iteratively with the optimization of the hyperparameters, or if they are separated and a pre-clustering step is applied? Another consideration would be the type of clustering. The splitting and CB-GPR methods may use, for example, uniform grid partitioning or spatial tree partitioning [30]. For the RVM-GPR, it may be embedded in the weight functions themselves, which is currently the case if the center parameter is also optimized.

Number of Local Models

The number of local models in this thesis is always set to two, however, the inference capabilities might highly change when more are used. It is suspected that the ability to capture global stationary features decreases when more local models are used. Therefore, this change is probably unnecessary in the constant and discontinuous correlation settings as one and two, respectively, local models already approximate the actual correlation. However, there are settings in which an increased number of local models approximates the actual correlation more closely, as is the case in the linearly varying setting. Both, the interpolation and extrapolation regime might benefit as an n-piecewise constant correlation better approximates a linear one.

6.3. Future Work

The following two paragraphs discuss possible research options that might be interesting but do not directly impact the applicability of the splitting and stitching methods as surrogates.

CB-GPR with Pseudo-observations

The CB-GPR method performs worse compared to the other methods, therefore it is suggested to look at a different boundary-constraining method for GPR models. It is proposed to constrain the points at the boundary by means of adding observations with zero noise variance. This constrains the predictive mean of both local models to be equal at that point, however, the predictive variance might differ. Note, that this method requires an iterative procedure, as the targets of these observations must be iterated over to find their optimal value.

Another approach would be to follow the work of Park et al. [30], where pseudo-observations are placed at the boundary which states that the difference between the local models is zero: effectively making it continuous at those points. This results in both the predictive mean and variance being equal, but the independency of the local models is lost; it is still less than that of the global GPR method as only the inputs of the pseudo-observations are correlated between the local models.

Sampling MF-GPR without Additive Part

The MF-GPRs, which are used to sample the functions of the cases, use an additive part in the high-fidelity. This is included to better represent real datasets because small variations are introduced to the correlation of the sampled functions. However, this adds more variation to the experimental results, as each sampled function of the same case has a slightly different correlation, which makes it harder to draw general conclusions. Therefore, it is suggested to run the same experiments without the additive part in the sampled MF-GPRs. This would hopefully result in experiments that better identify the differences between the splitting and stitching methods because in each case the correlation of the sampled functions matches the defined correlation.

Optimization of Case Construction

This thesis compares several methods by measuring their performance on datasets in given cases. Another approach would be, to find a case that maximizes the difference in performance between the methods. Such an approach has the advantage to reduce the biases inherent in selecting cases. The case-constructing method of this thesis, sampling an MF-GPR, is naturally suitable for such a practice, as its hyperparameters can be inserted in an optimization algorithm that maximizes the difference in performance between the methods. In our case, consideration must be taken when deciding on a suitable space for the correlation, as it must encapsulate the interested domain of research, i.e., (discontinuous) non-linear correlation. Thus, this new perspective shifts the focus to finding a suitable space of cases in which the methods can differentiate themselves. This approach to method comparison is not entirely new, as Wilde et al. [51] have used a similar idea based on generating an artificial dataset using an evolutionary algorithm for which a specific classification method performs well on a given metric. However, applying the same approach to regression and using MF-GPRs in the case optimization process is novel and could potentially lead to interesting results.

A

Experimental Results

The single-fidelity GPR method is excluded from this appendix for aesthetic reasons: 10 methods do not fit nicely on a page.

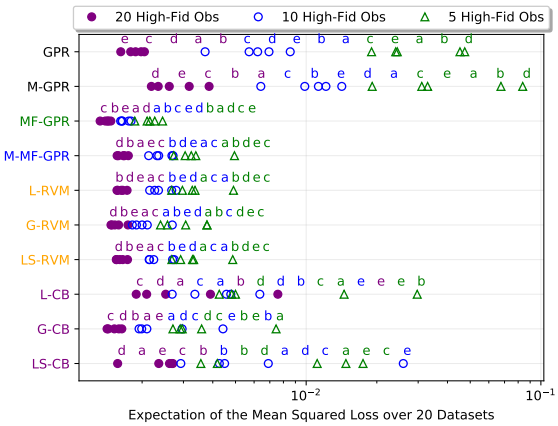
The appendix contents provide an efficient browsing option. Each expected error plot and prediction plot that is referenced has a reference back.

A.1. Appendix Contents

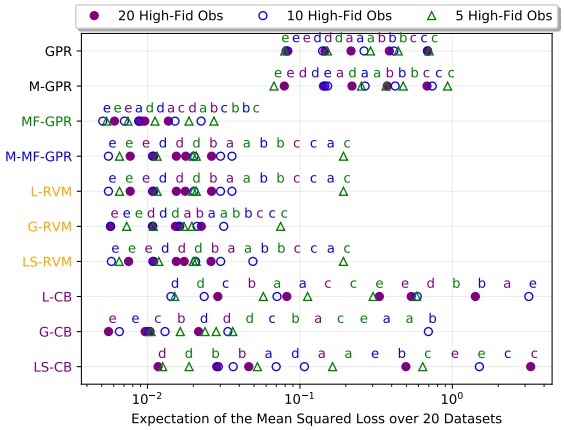
A.2. Inputs sampled from a Uniform Distribution

A.2.1. Constant ρ case

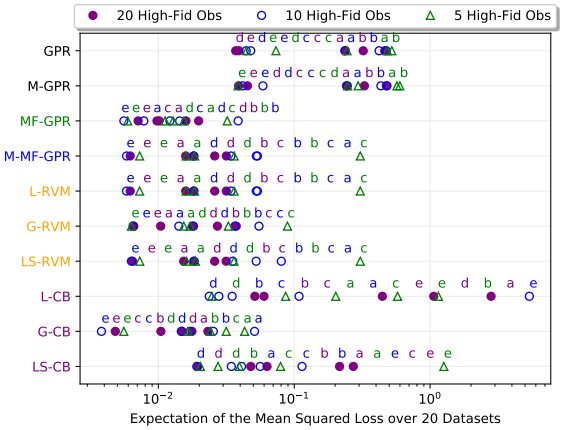
Appendix A: Experimental Results



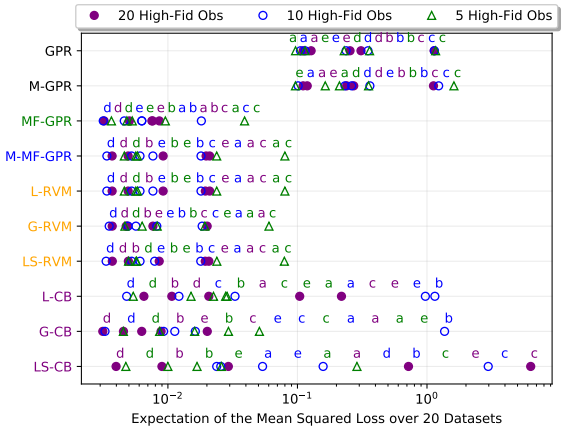
(a) Interpolation regime: $[0,2]$.



(b) Extrapolation regime: $[-1,0] \cup [2,3]$.



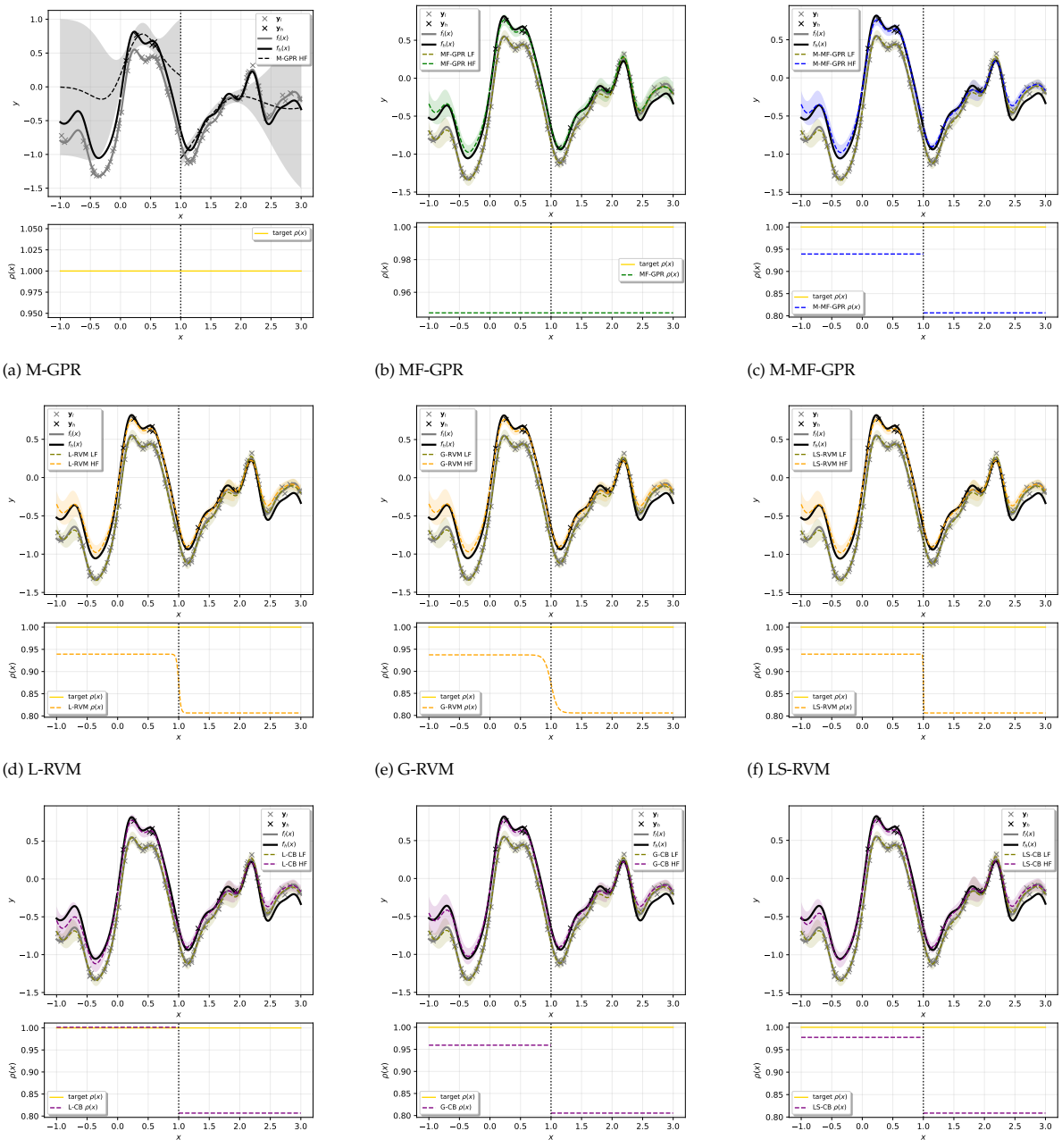
(c) Left extrapolation regime: $[-1,0]$.



(d) Right extrapolation regime: $[2,3]$.

Figure A.1: constant ρ case, inputs sampled from a uniform distribution, and 21 low-fids per region.

Appendix A: Experimental Results



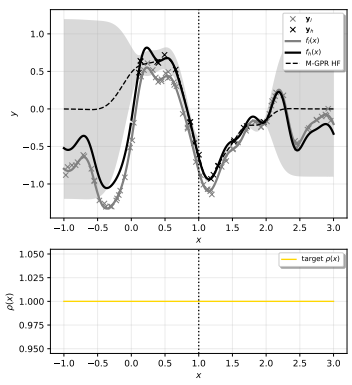
(a) M-GPR (b) MF-GPR (c) M-MF-GPR

(d) L-RVM (e) G-RVM (f) LS-RVM

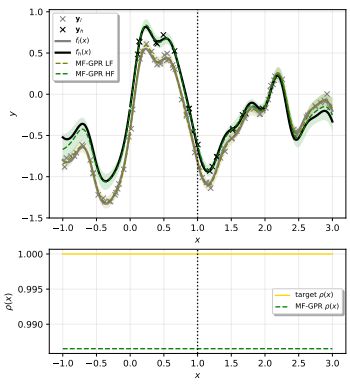
(g) L-CB (h) G-CB (i) LS-CB

Figure A.2: model predictions with 5 high-fids per region of function **a** and data-set 0 of figure A.1.

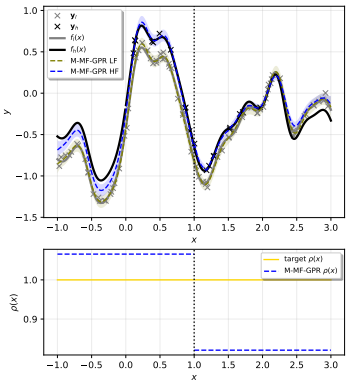
Appendix A: Experimental Results



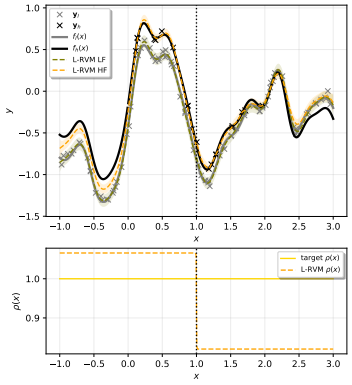
(a) M-GPR



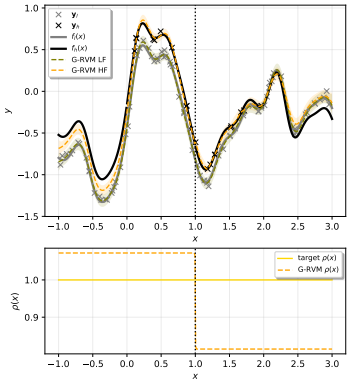
(b) MF-GPR



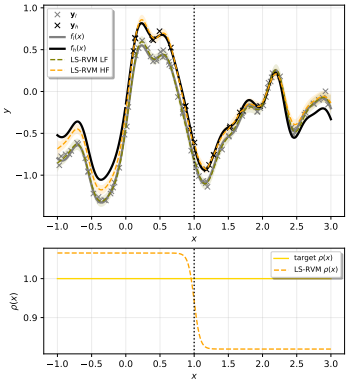
(c) M-MF-GPR



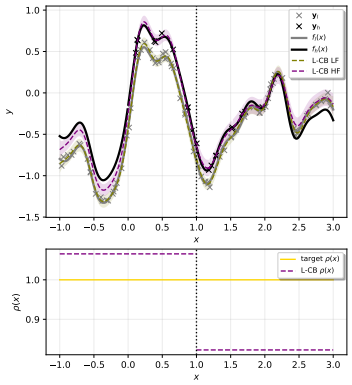
(d) L-RVM



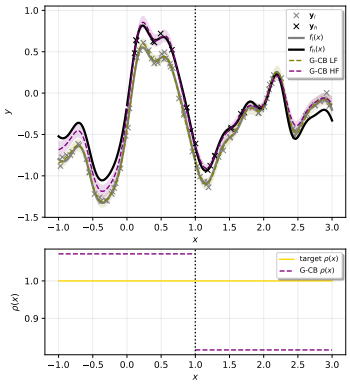
(e) G-RVM



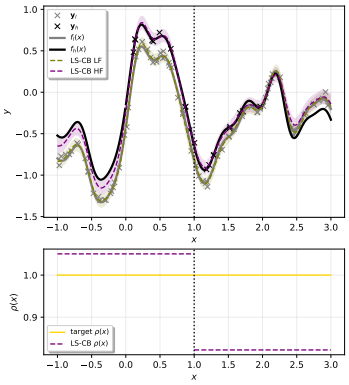
(f) LS-RVM



(g) L-CB



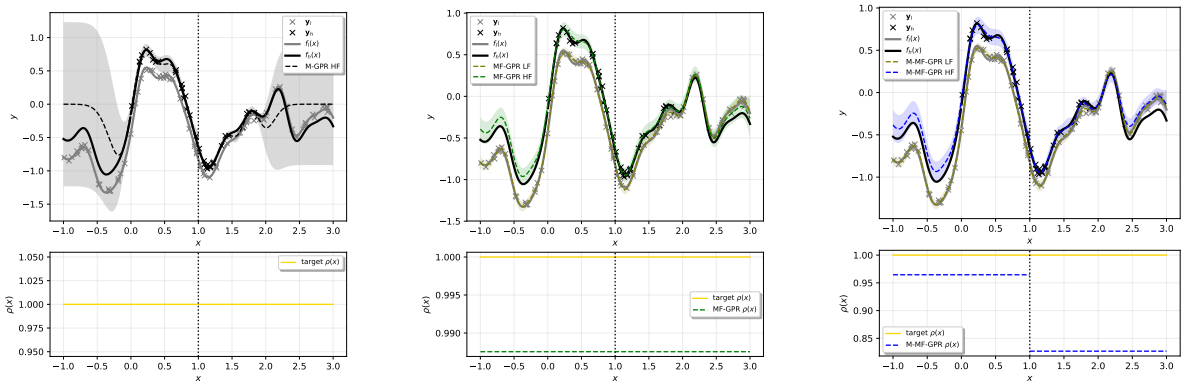
(h) G-CB



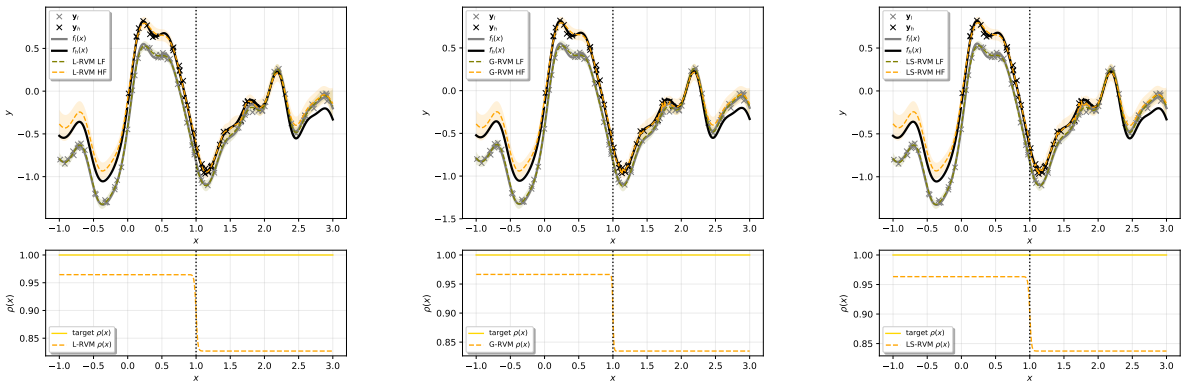
(i) LS-CB

Figure A.3: model predictions with 10 high-fids per region of function **a** and data-set 0 of figure A.1.

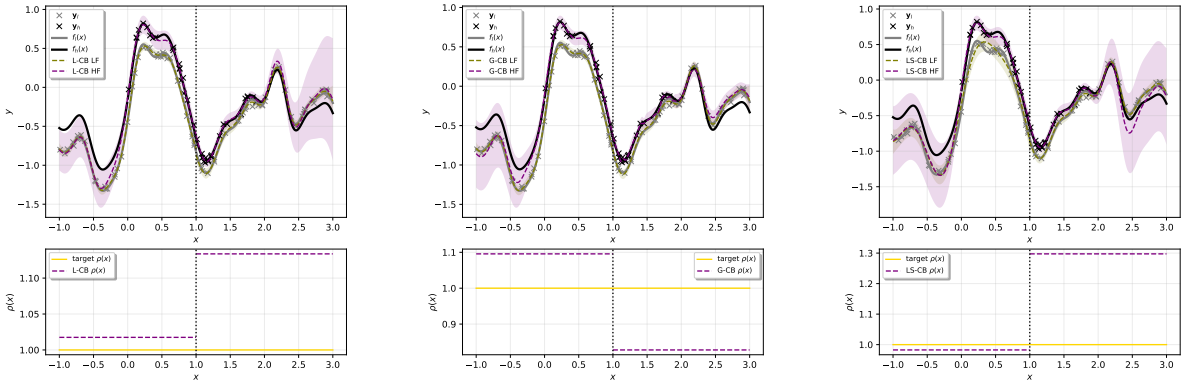
Appendix A: Experimental Results



(a) M-GPR (b) MF-GPR (c) M-MF-GPR



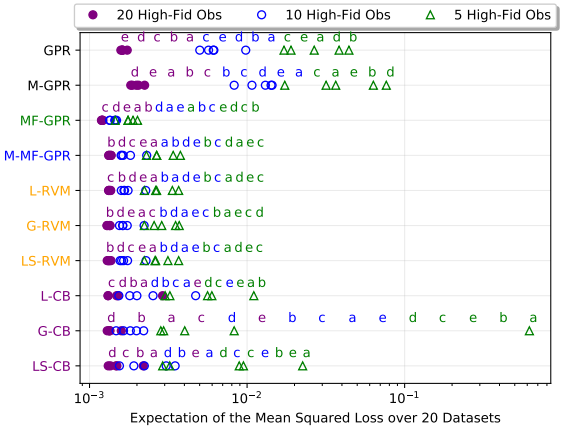
(d) L-RVM (e) G-RVM (f) LS-RVM



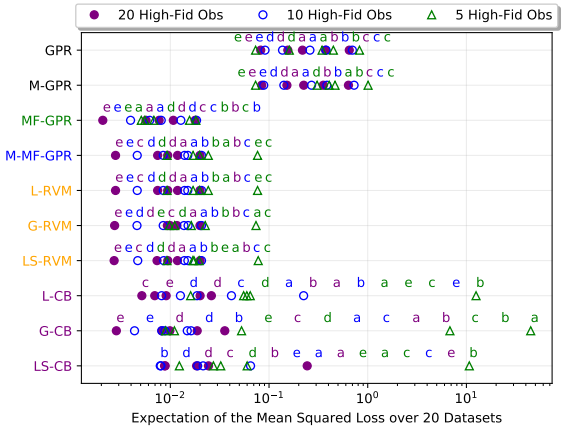
(g) L-CB (h) G-CB (i) LS-CB

Figure A.4: model predictions with 20 high-fids per region of function **a** and data-set 0 of figure A.1.

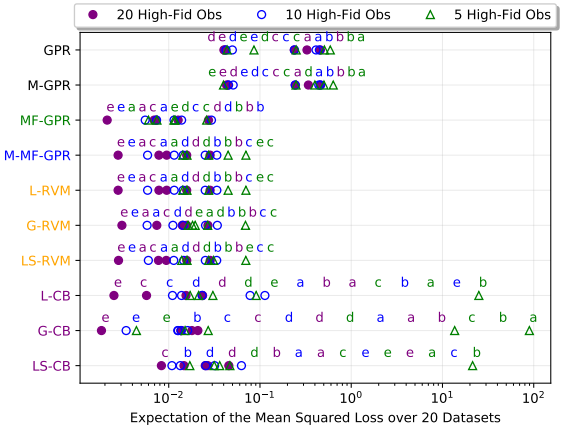
Appendix A: Experimental Results



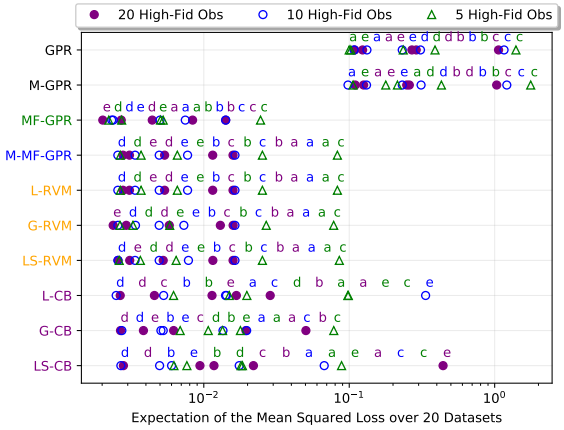
(a) Interpolation regime: $[0,2]$.



(b) Extrapolation regime: $[-1,0] \cup [2,3]$.



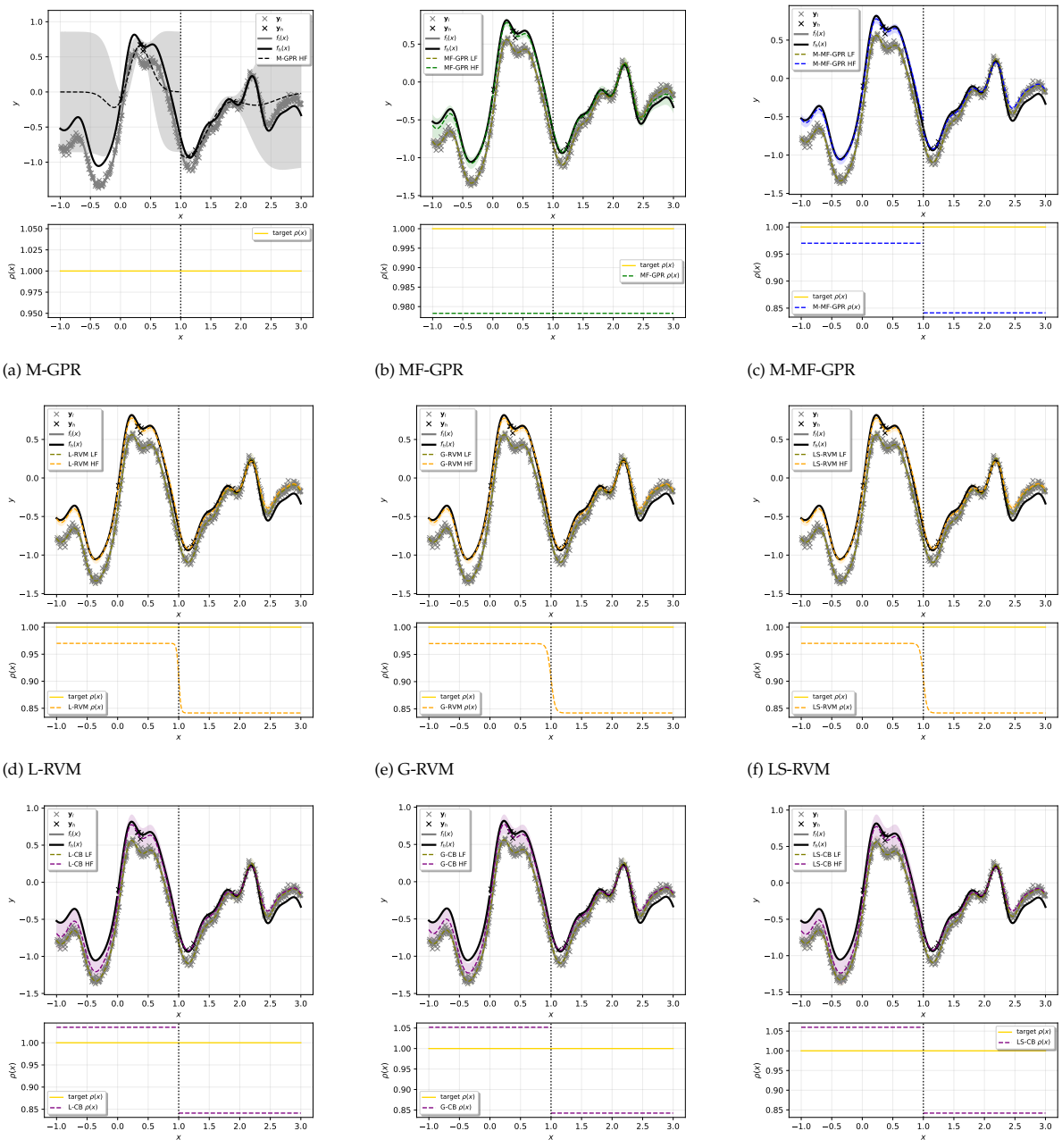
(c) Left extrapolation regime: $[-1,0]$.



(d) Right extrapolation regime: $[2,3]$.

Figure A.5: constant ρ case, inputs sampled from a uniform distribution, and 101 low-fids per region.

Appendix A: Experimental Results



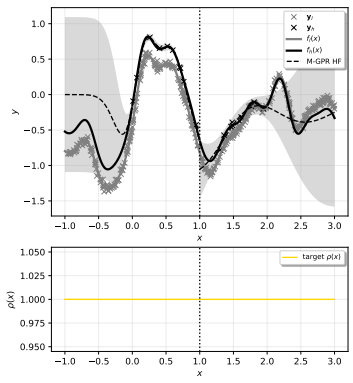
(a) M-GPR (b) MF-GPR (c) M-MF-GPR

(d) L-RVM (e) G-RVM (f) LS-RVM

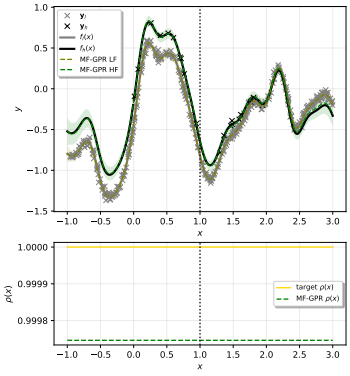
(g) L-CB (h) G-CB (i) LS-CB

Figure A.6: model predictions with 5 high-fids per region of function **a** and data-set 0 of figure A.5.

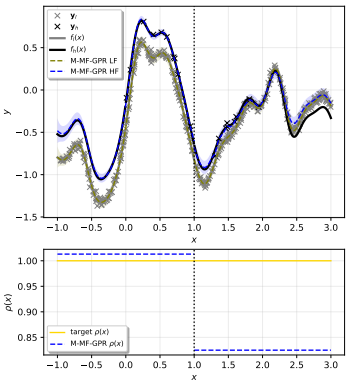
Appendix A: Experimental Results



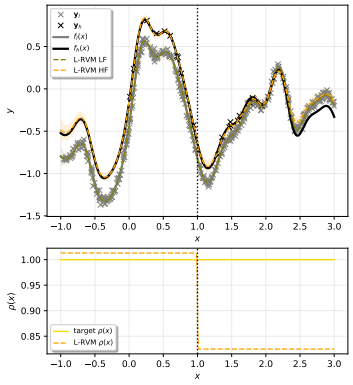
(a) M-GPR



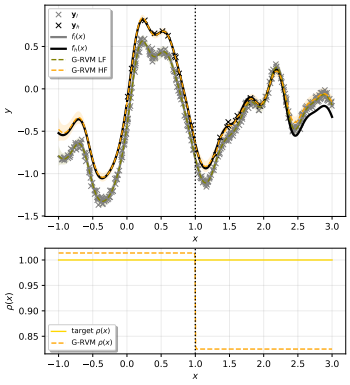
(b) MF-GPR



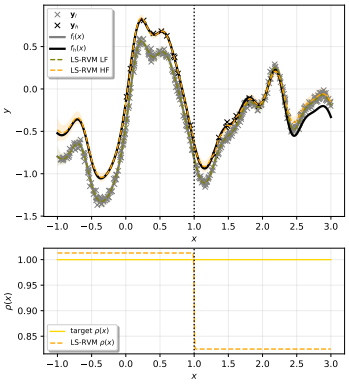
(c) M-MF-GPR



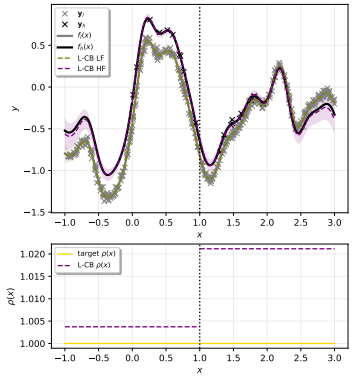
(d) L-RVM



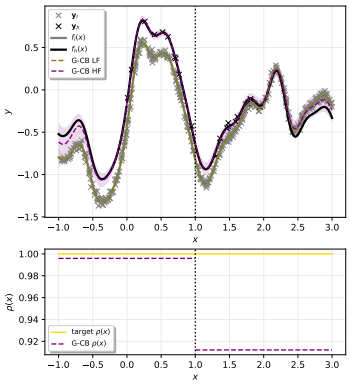
(e) G-RVM



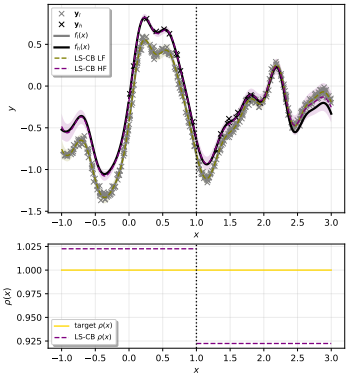
(f) LS-RVM



(g) L-CB



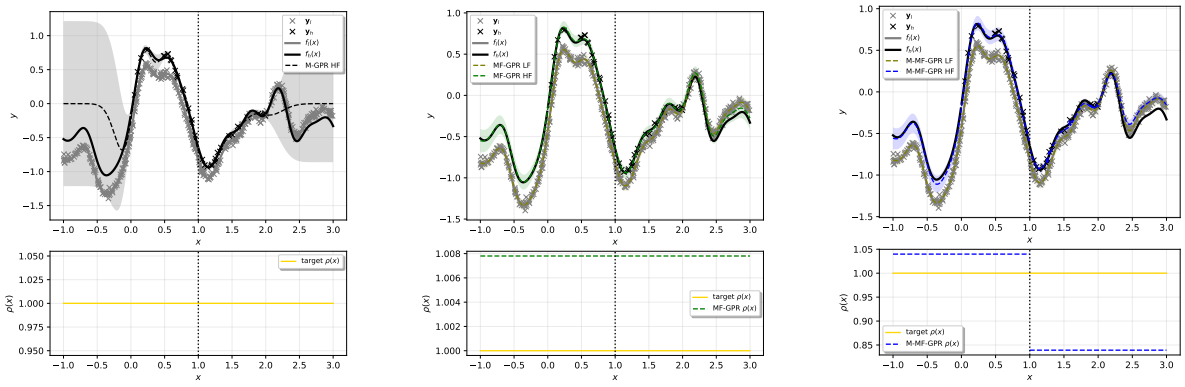
(h) G-CB



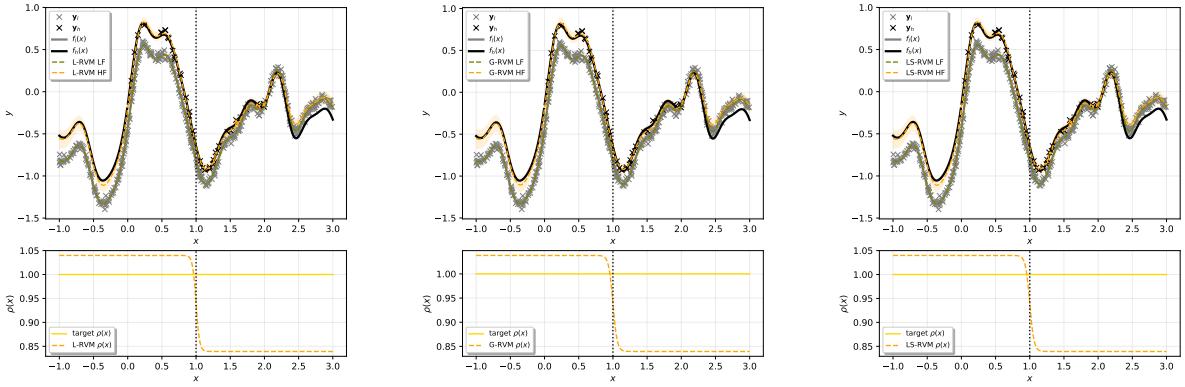
(i) LS-CB

Figure A.7: model predictions with 10 high-fids per region of function **a** and data-set 0 of figure A.5.

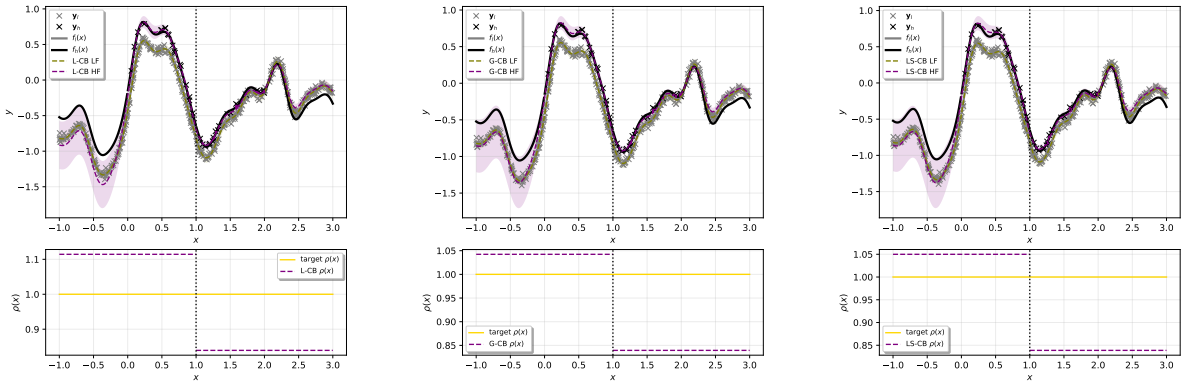
Appendix A: Experimental Results



(a) M-GPR (b) MF-GPR (c) M-MF-GPR



(d) L-RVM (e) G-RVM (f) LS-RVM

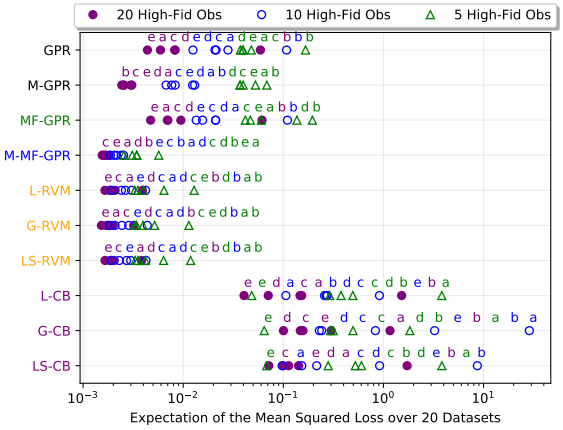


(g) L-CB (h) G-CB (i) LS-CB

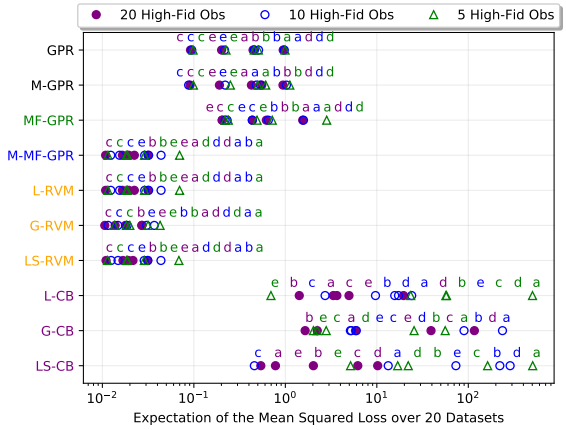
Figure A.8: model predictions with 20 high-fids per region of function **a** and data-set 0 of figure A.5.

A.2.2. Discontinuous ρ case

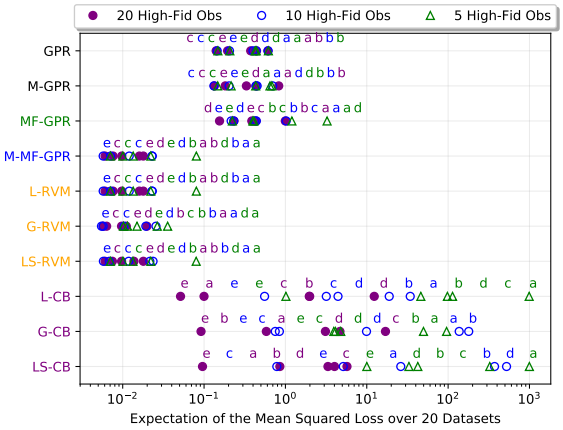
Appendix A: Experimental Results



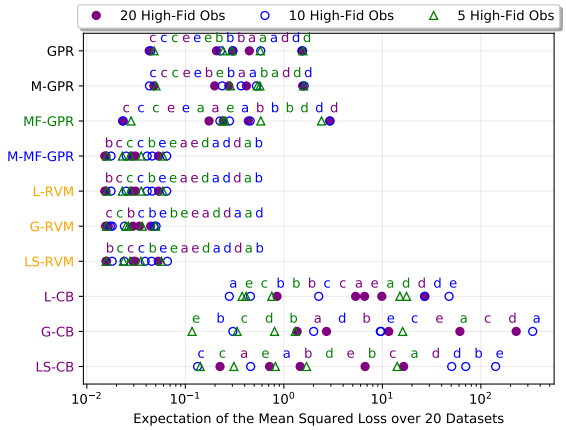
(a) Interpolation regime: $[0,2]$.



(b) Extrapolation regime: $[-1,0] \cup [2,3]$.



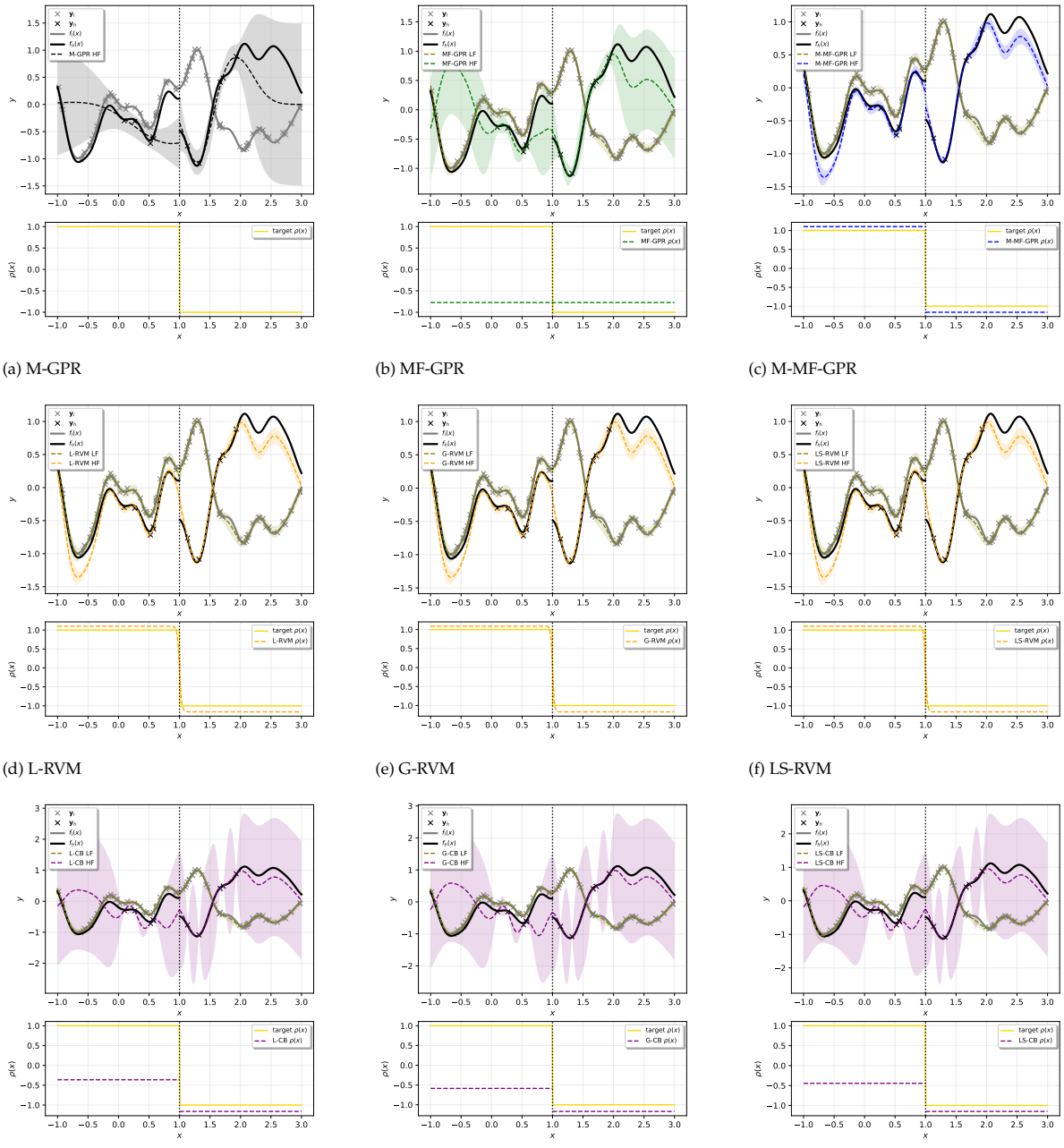
(c) Left extrapolation regime: $[-1,0]$.



(d) Right extrapolation regime: $[2,3]$.

Figure A.9: discontinuous ρ case, inputs sampled from a uniform distribution, and 21 low-fids per region.

Appendix A: Experimental Results



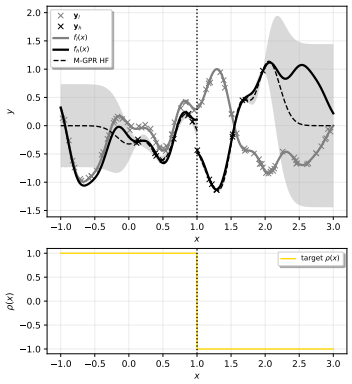
(a) M-GPR (b) MF-GPR (c) M-MF-GPR

(d) L-RVM (e) G-RVM (f) LS-RVM

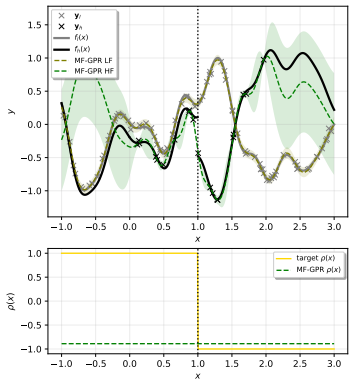
(g) L-CB (h) G-CB (i) LS-CB

Figure A.10: model predictions with 5 high-fids per region of function **a** and data-set 0 of figure A.9.

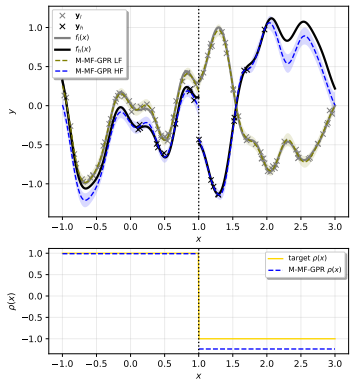
Appendix A: Experimental Results



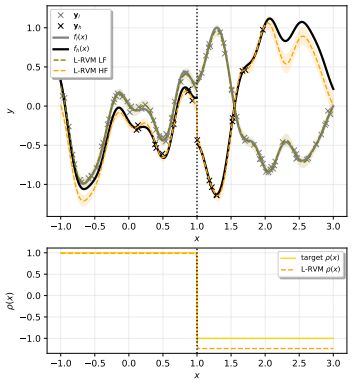
(a) M-GPR



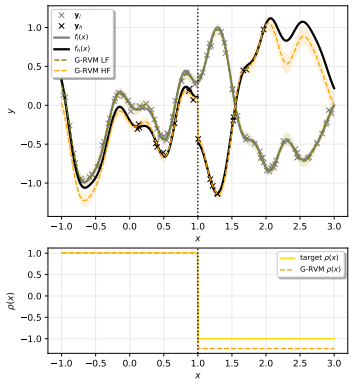
(b) MF-GPR



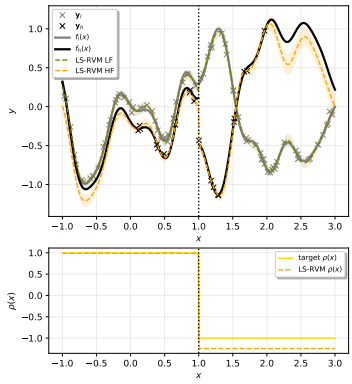
(c) M-MF-GPR



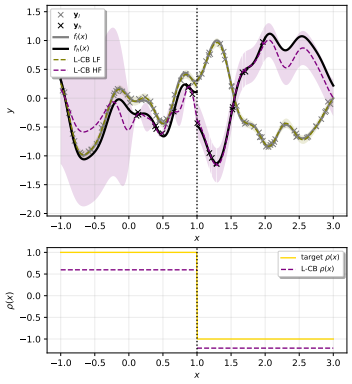
(d) L-RVM



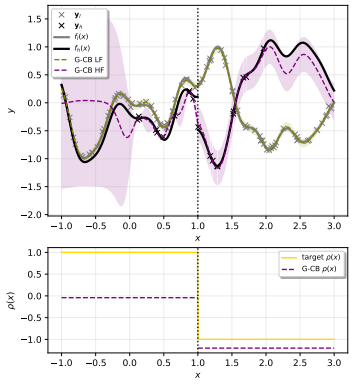
(e) G-RVM



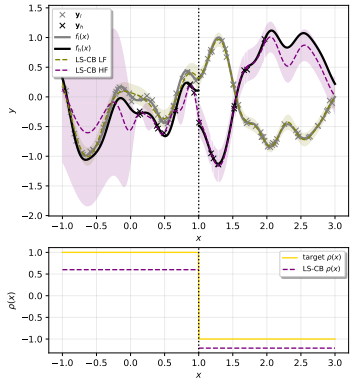
(f) LS-RVM



(g) L-CB



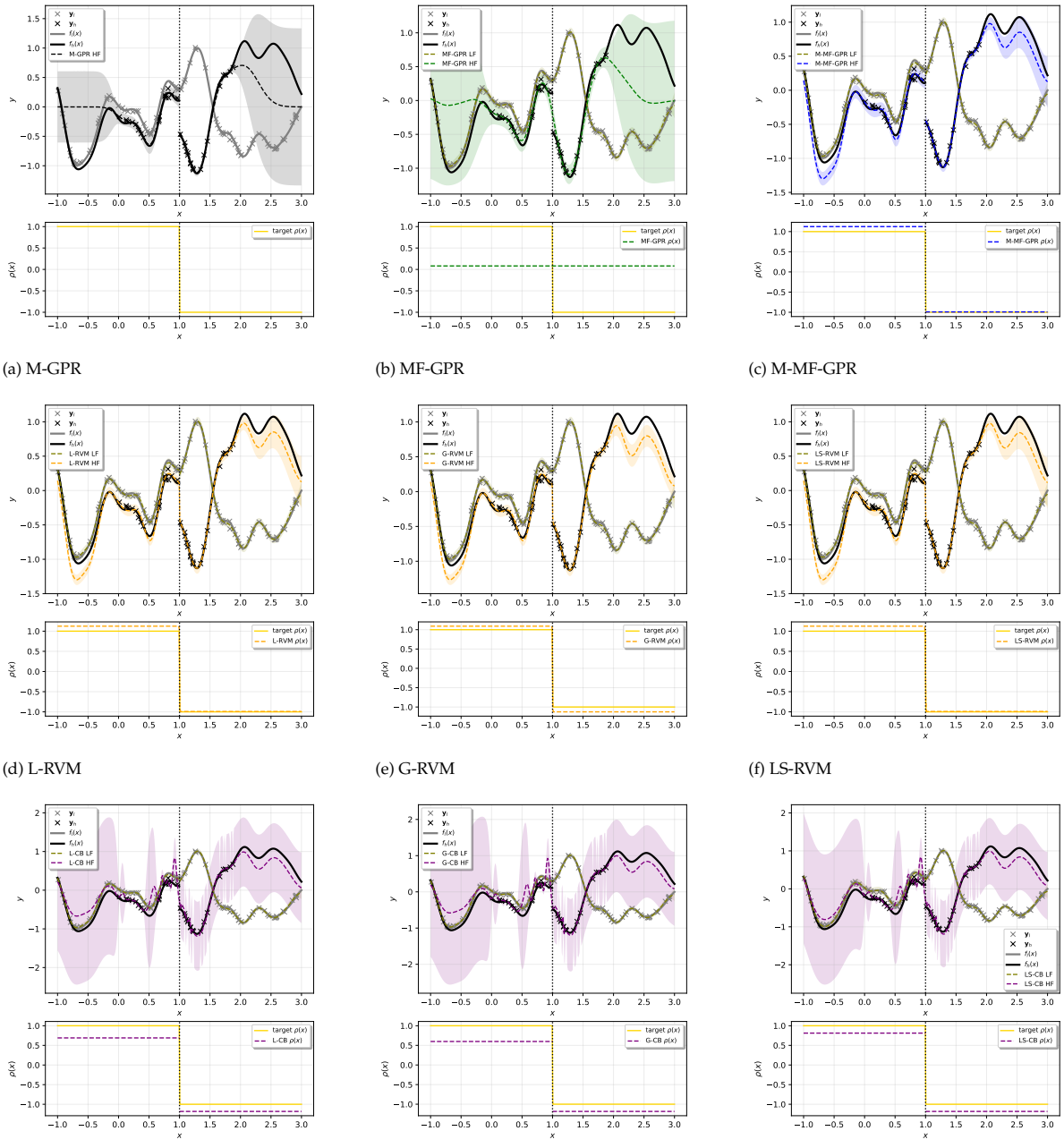
(h) G-CB



(i) LS-CB

Figure A.11: model predictions with 10 high-fids per region of function **a** and data-set 0 of figure A.9.

Appendix A: Experimental Results



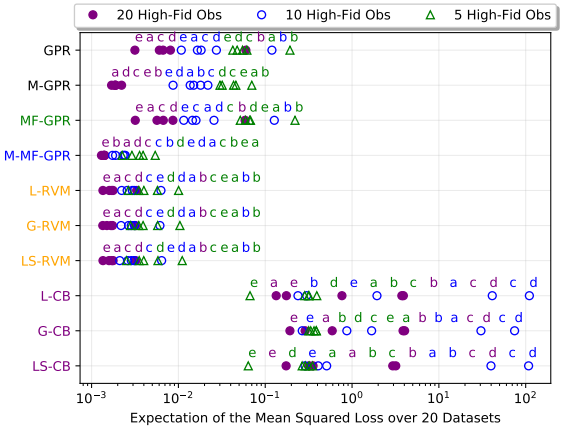
(a) M-GPR (b) MF-GPR (c) M-MF-GPR

(d) L-RVM (e) G-RVM (f) LS-RVM

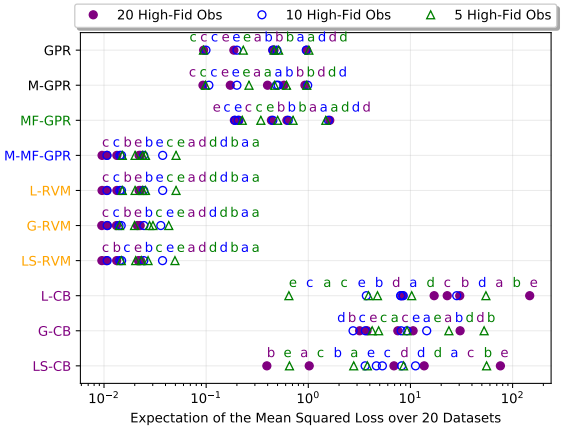
(g) L-CB (h) G-CB (i) LS-CB

Figure A.12: model predictions with 20 high-fids per region of function **a** and data-set 0 of figure A.9.

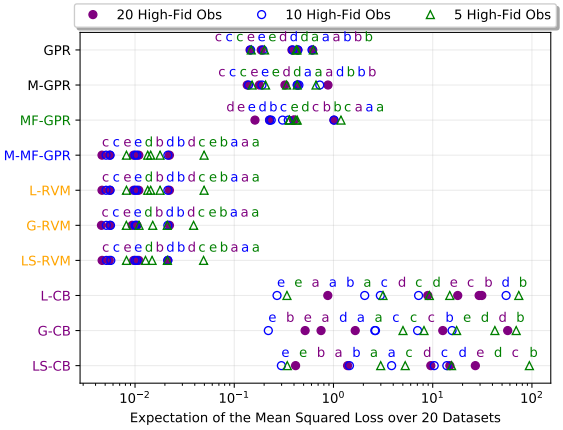
Appendix A: Experimental Results



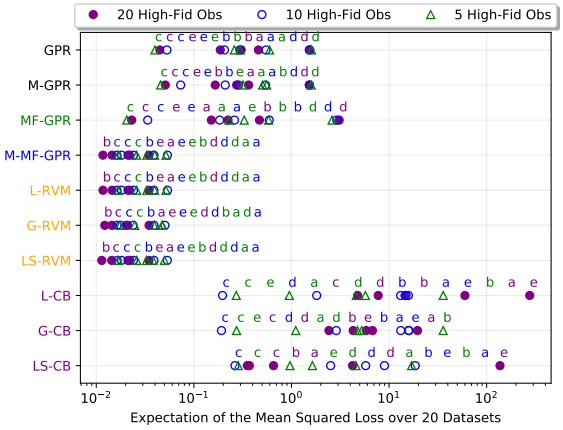
(a) Interpolation regime: $[0,2]$.



(b) Extrapolation regime: $[-1,0] \cup [2,3]$.



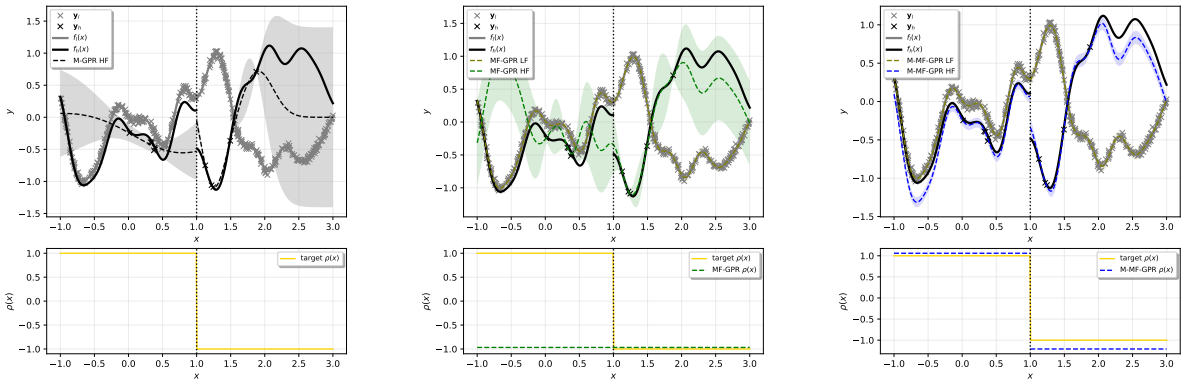
(c) Left extrapolation regime: $[-1,0]$.



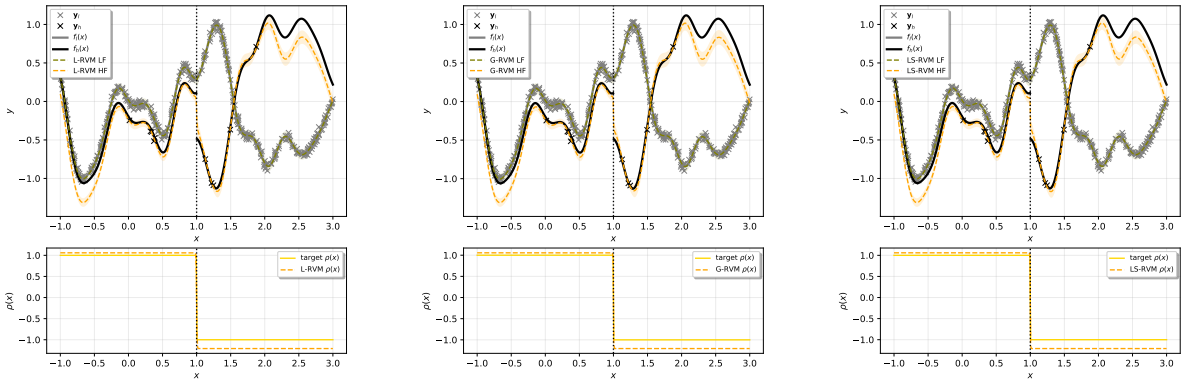
(d) Right extrapolation regime: $[2,3]$.

Figure A.13: discontinuous ρ case, inputs sampled from a uniform distribution, and 101 low-fids per region.

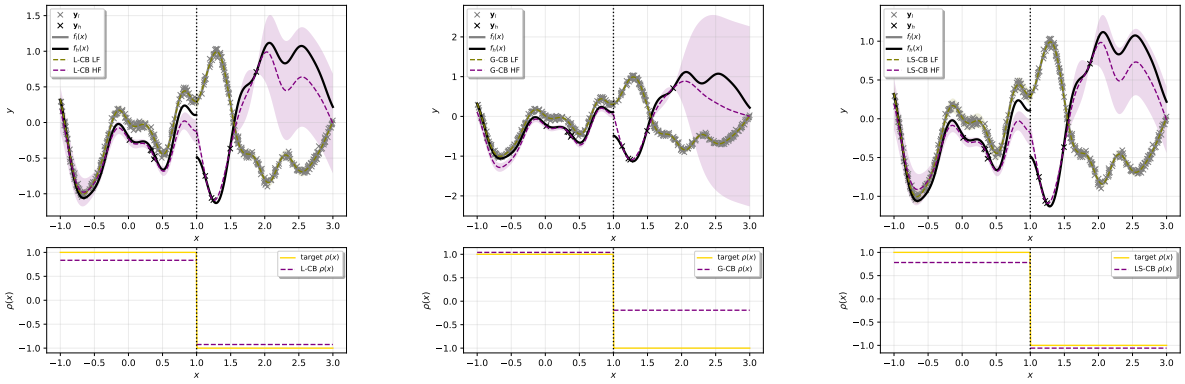
Appendix A: Experimental Results



(a) M-GPR (b) MF-GPR (c) M-MF-GPR



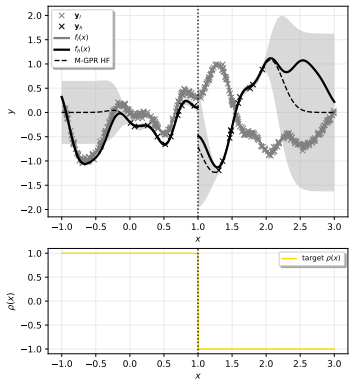
(d) L-RVM (e) G-RVM (f) LS-RVM



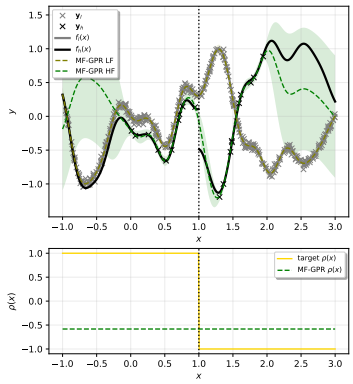
(g) L-CB (h) G-CB (i) LS-CB

Figure A.14: model predictions with 5 high-fids per region of function **a** and data-set 0 of figure A.13.

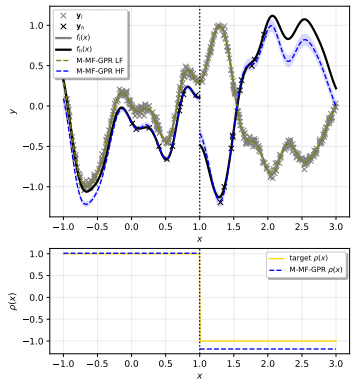
Appendix A: Experimental Results



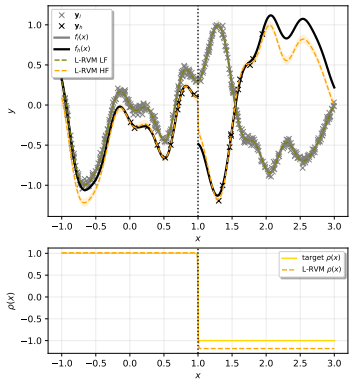
(a) M-GPR



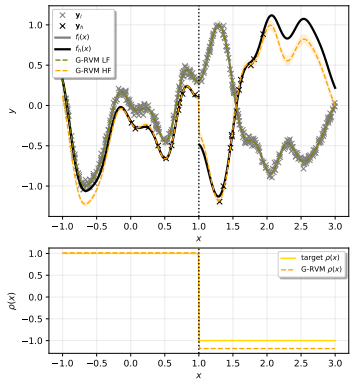
(b) MF-GPR



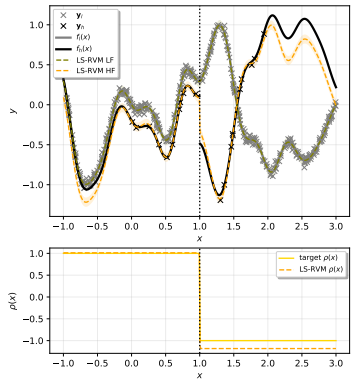
(c) M-MF-GPR



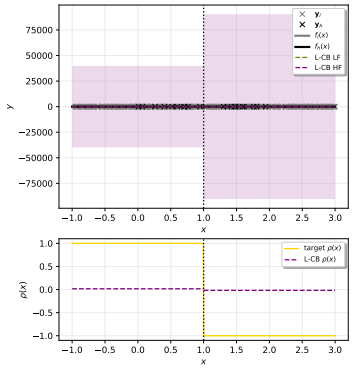
(d) L-RVM



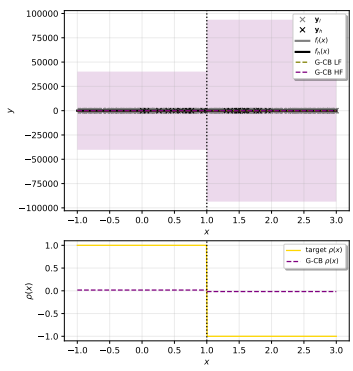
(e) G-RVM



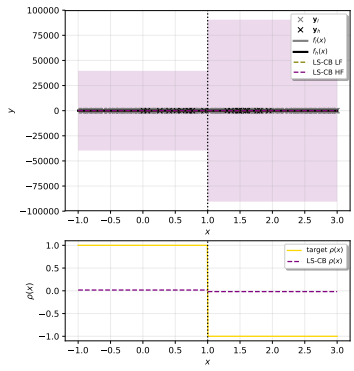
(f) LS-RVM



(g) L-CB



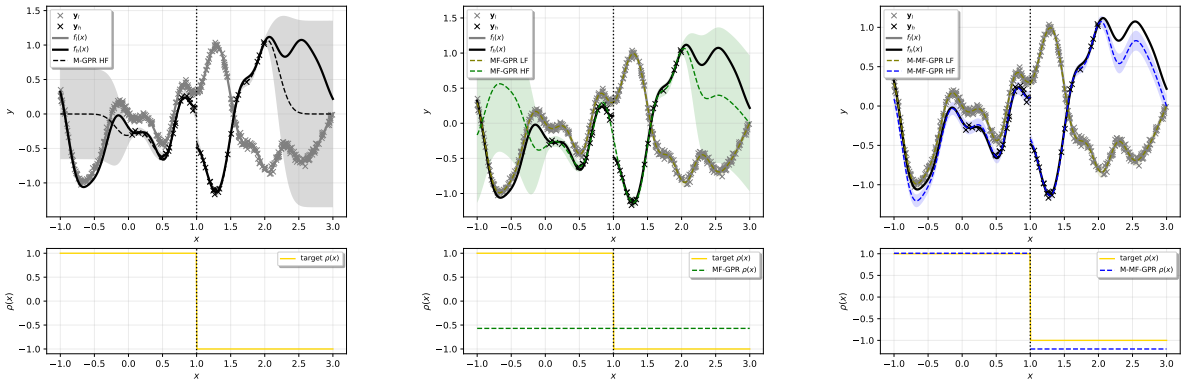
(h) G-CB



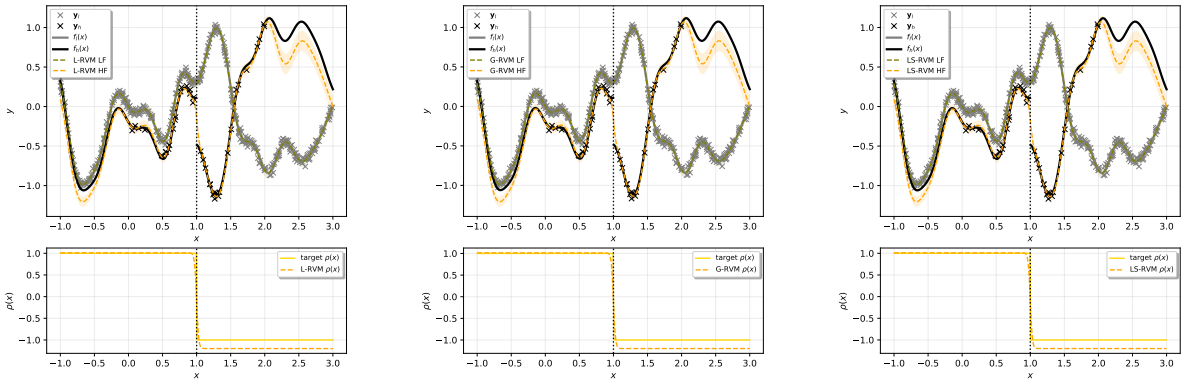
(i) LS-CB

Figure A.15: model predictions with 10 high-fids per region of function **a** and data-set 0 of figure A.13.

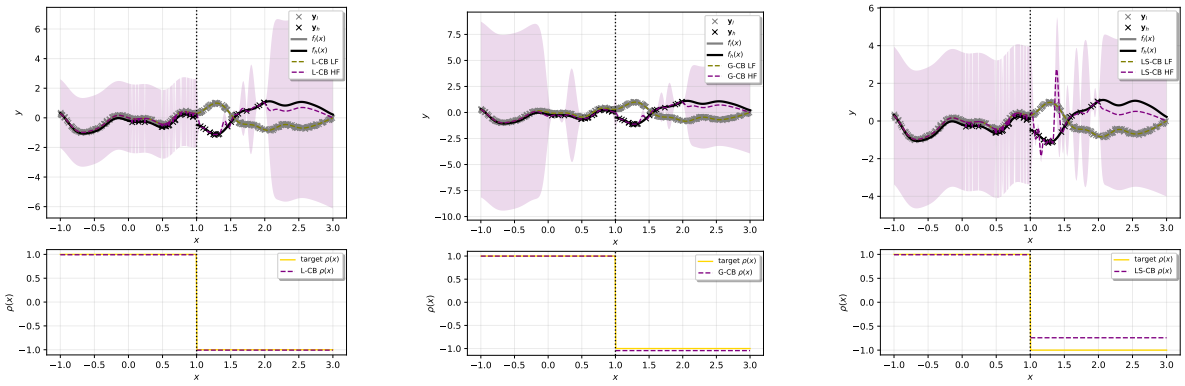
Appendix A: Experimental Results



(a) M-GPR (b) MF-GPR (c) M-MF-GPR



(d) L-RVM (e) G-RVM (f) LS-RVM

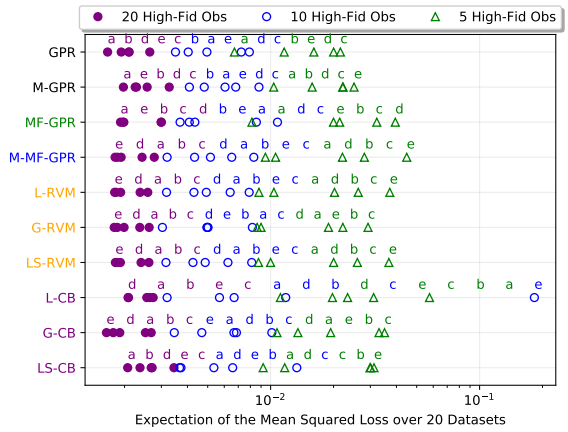


(g) L-CB (h) G-CB (i) LS-CB

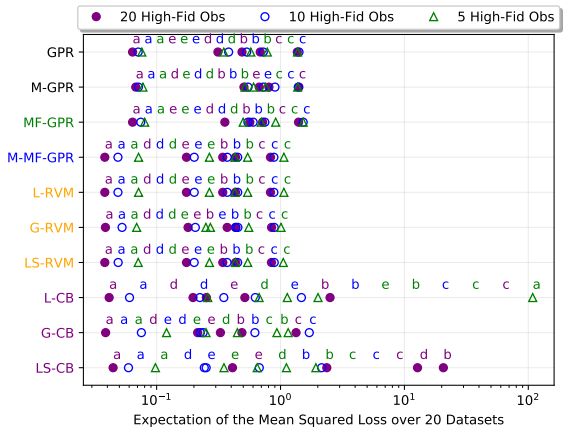
Figure A.16: model predictions with 20 high-fids per region of function **a** and data-set 0 of figure A.13.

A.2.3. Linearly varying ρ case

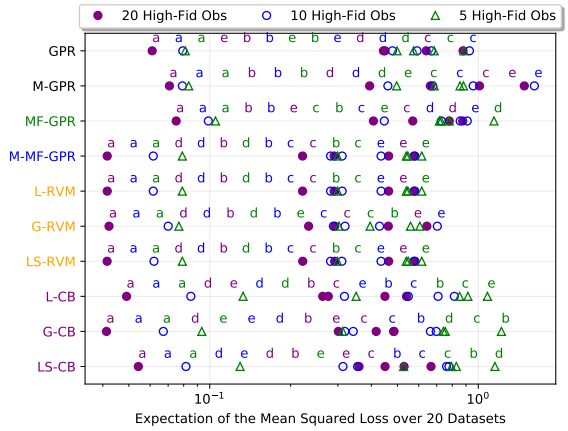
Appendix A: Experimental Results



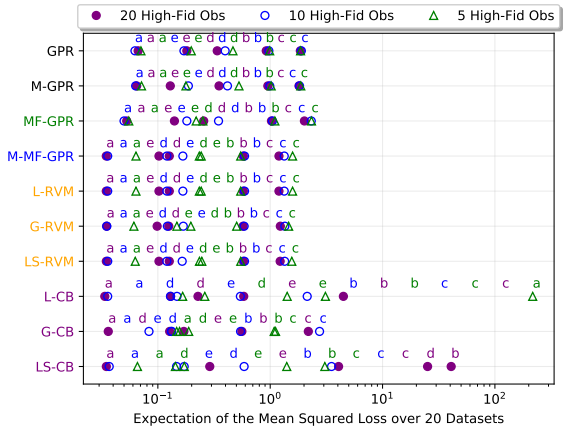
(a) Interpolation regime: $[0,2]$.



(b) Extrapolation regime: $[-1,0] \cup [2,3]$.



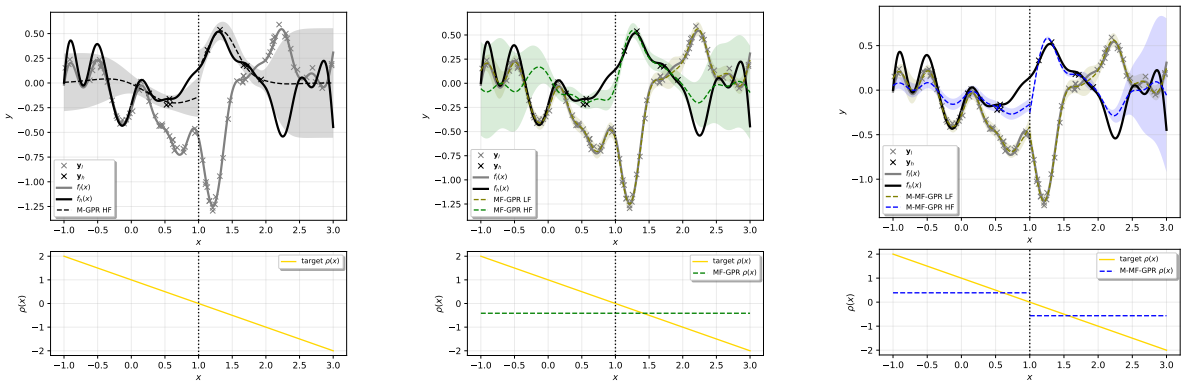
(c) Left extrapolation regime: $[-1,0]$.



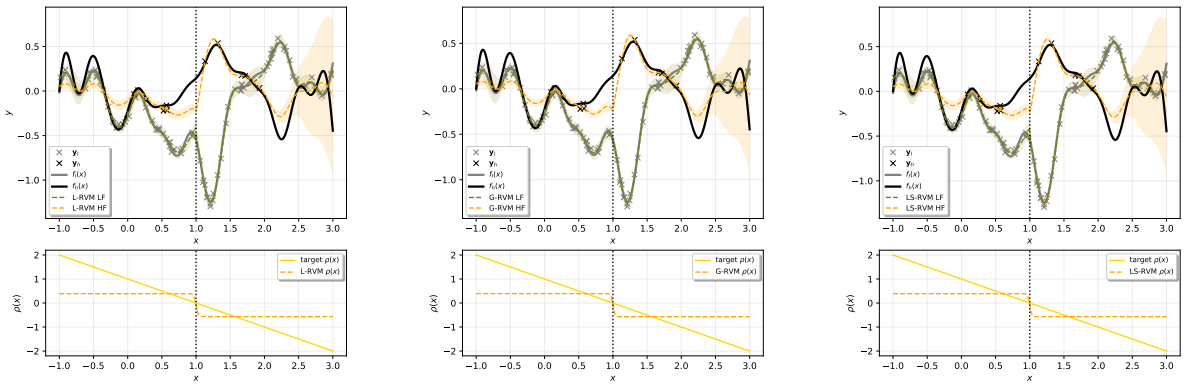
(d) Right extrapolation regime: $[2,3]$.

Figure A.17: linearly varying ρ case, inputs sampled from a uniform distribution, and 21 low-fids per region.

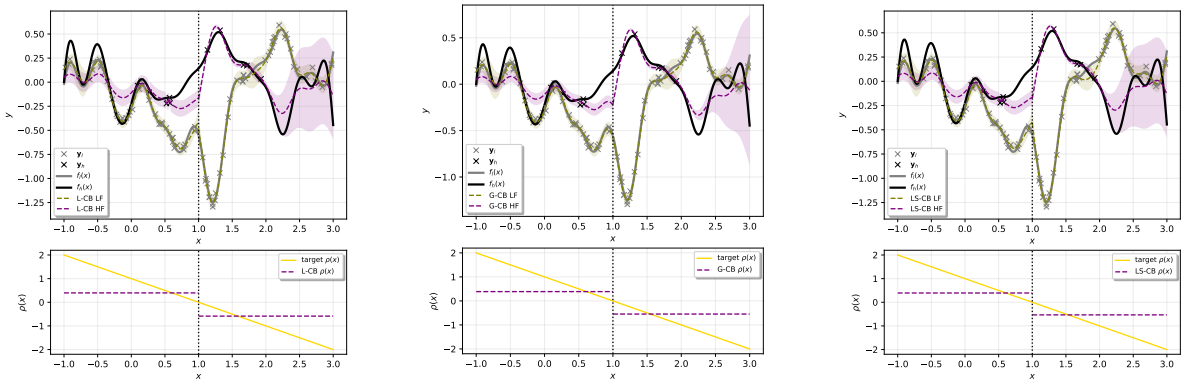
Appendix A: Experimental Results



(a) M-GPR (b) MF-GPR (c) M-MF-GPR



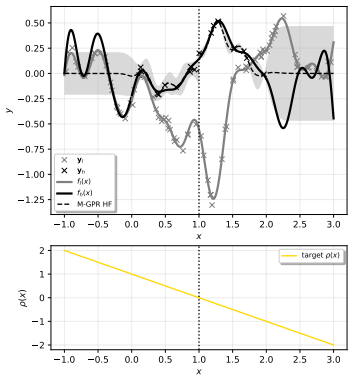
(d) L-RVM (e) G-RVM (f) LS-RVM



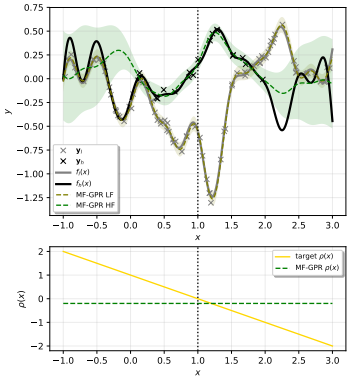
(g) L-CB (h) G-CB (i) LS-CB

Figure A.18: model predictions with 5 high-fids per region of function **a** and data-set 0 of figure A.17.

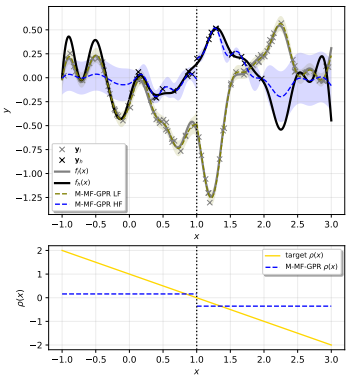
Appendix A: Experimental Results



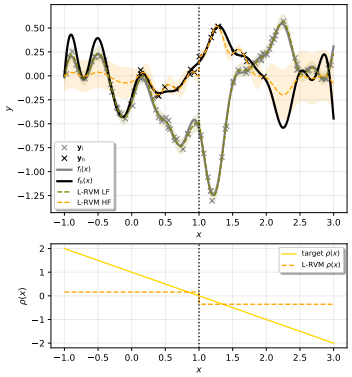
(a) M-GPR



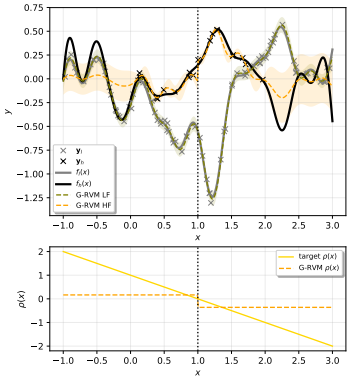
(b) MF-GPR



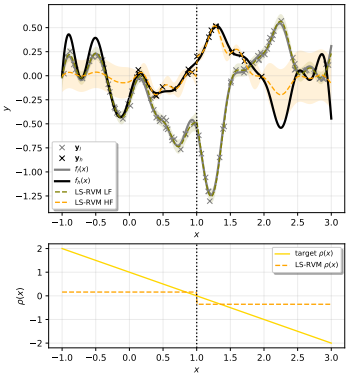
(c) M-MF-GPR



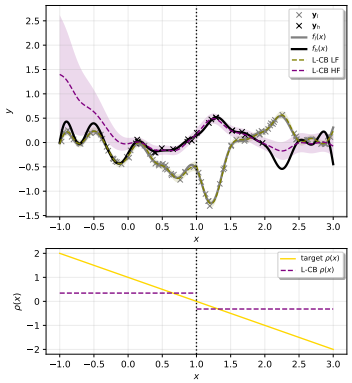
(d) L-RVM



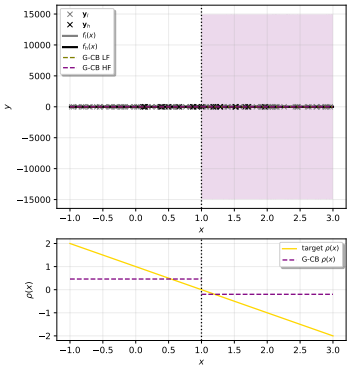
(e) G-RVM



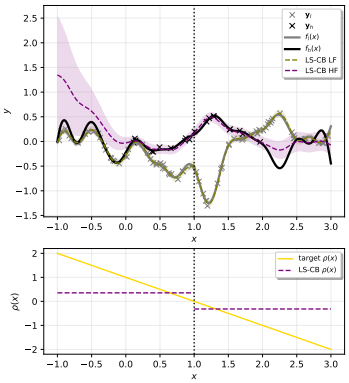
(f) LS-RVM



(g) L-CB



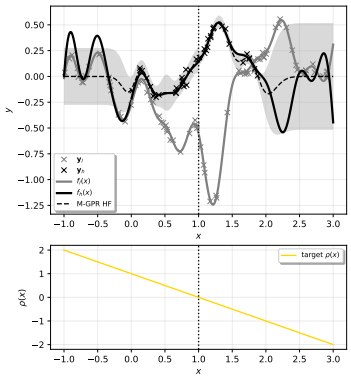
(h) G-CB



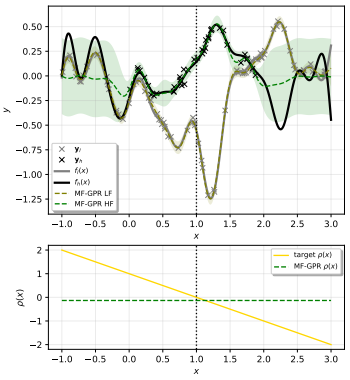
(i) LS-CB

Figure A.19: model predictions with 10 high-fids per region of function **a** and data-set 0 of figure A.17.

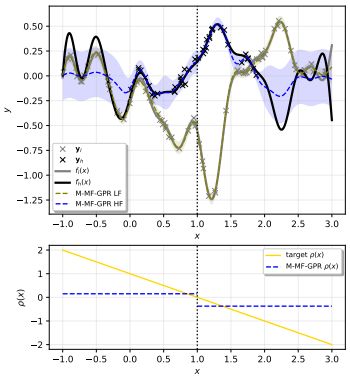
Appendix A: Experimental Results



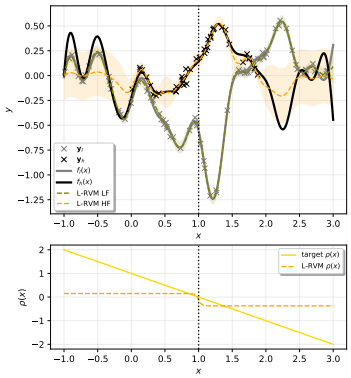
(a) M-GPR



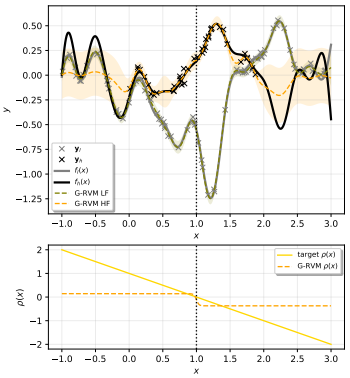
(b) MF-GPR



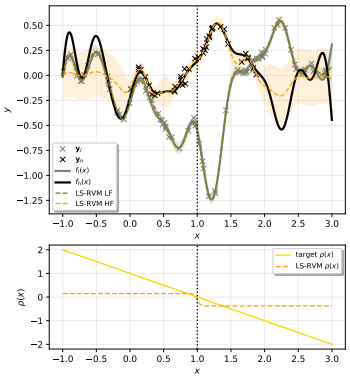
(c) M-MF-GPR



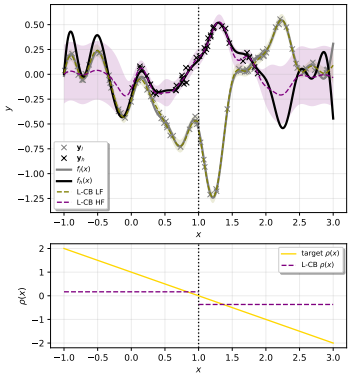
(d) L-RVM



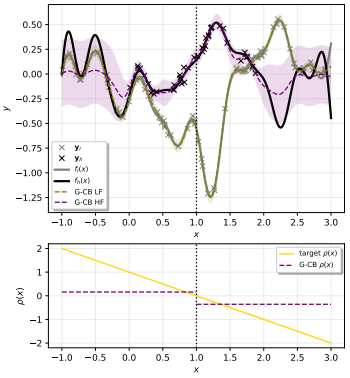
(e) G-RVM



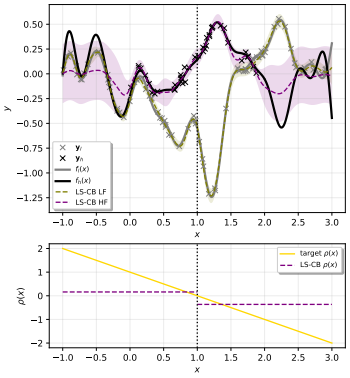
(f) LS-RVM



(g) L-CB



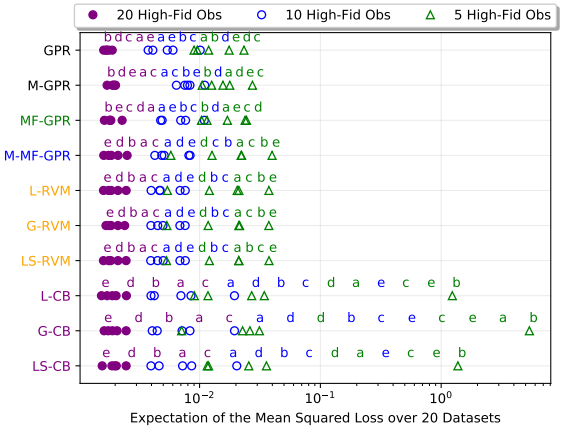
(h) G-CB



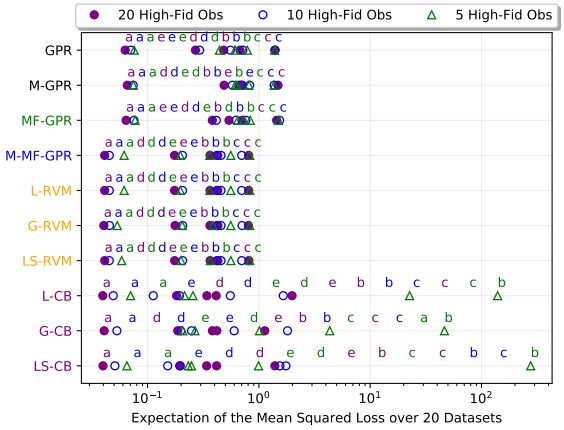
(i) LS-CB

Figure A.20: model predictions with 20 high-fids per region of function **a** and data-set 0 of figure A.17.

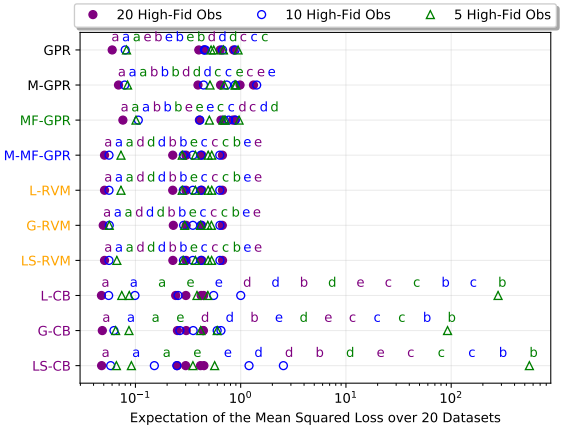
Appendix A: Experimental Results



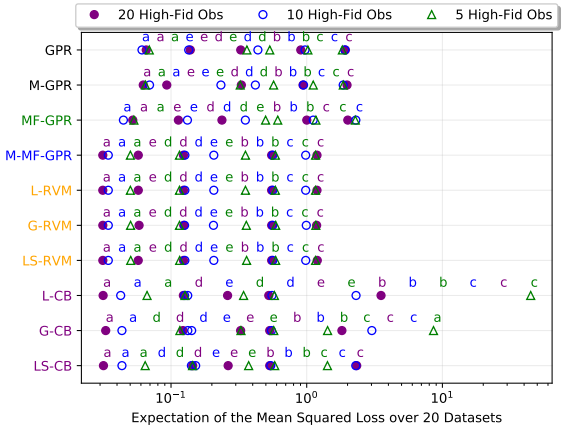
(a) Interpolation regime: $[0,2]$.



(b) Extrapolation regime: $[-1,0] \cup [2,3]$.



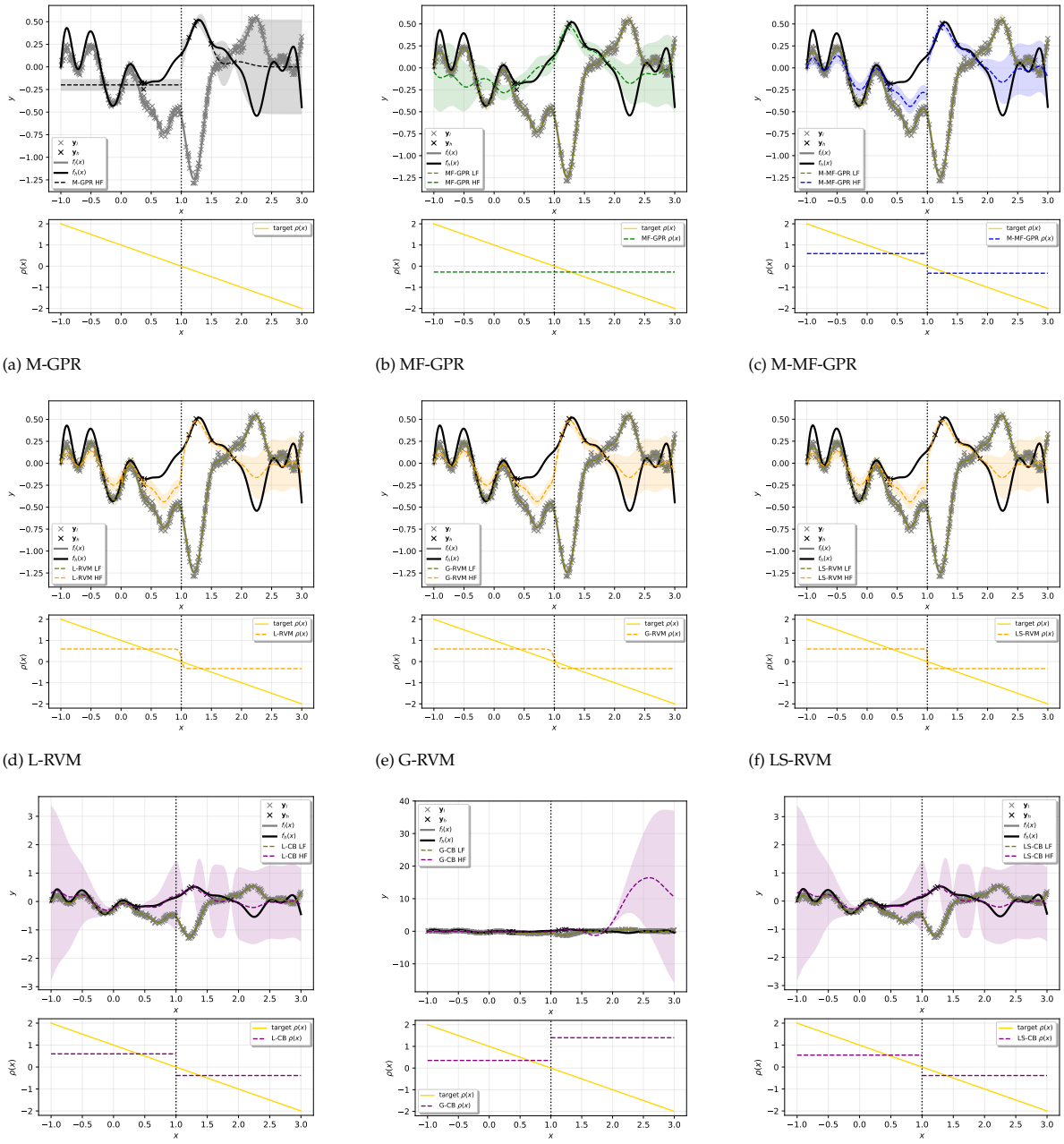
(c) Left extrapolation regime: $[-1,0]$.



(d) Right extrapolation regime: $[2,3]$.

Figure A.21: linearly varying ρ case, inputs sampled from a uniform distribution, and 101 low-fids per region.

Appendix A: Experimental Results



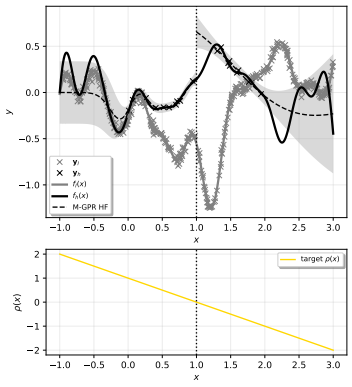
(a) M-GPR (b) MF-GPR (c) M-MF-GPR

(d) L-RVM (e) G-RVM (f) LS-RVM

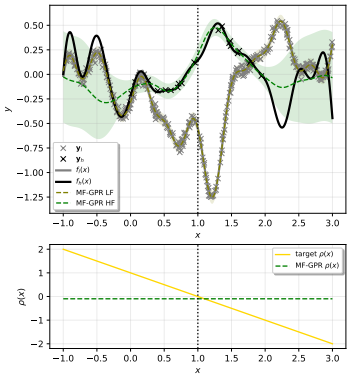
(g) L-CB (h) G-CB (i) LS-CB

Figure A.22: model predictions with 5 high-fids per region of function **a** and data-set 0 of figure A.21.

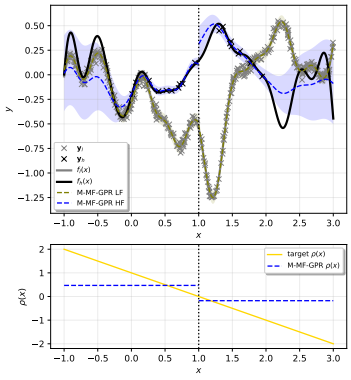
Appendix A: Experimental Results



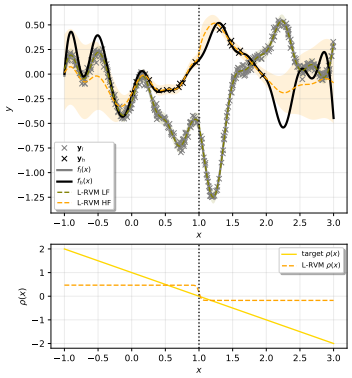
(a) M-GPR



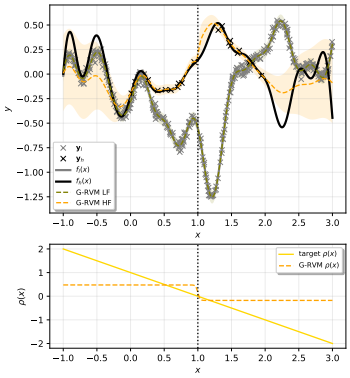
(b) MF-GPR



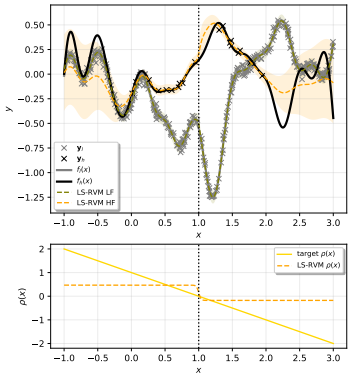
(c) M-MF-GPR



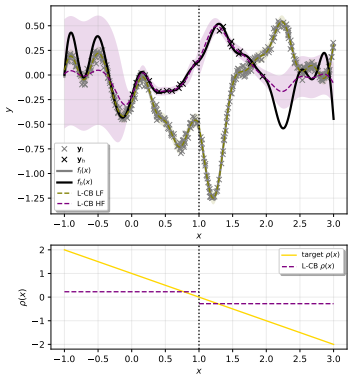
(d) L-RVM



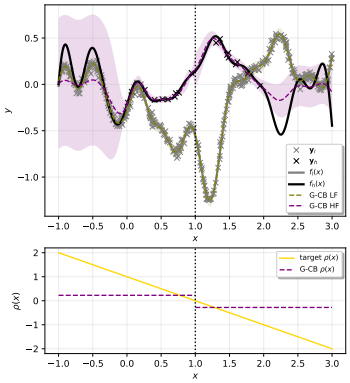
(e) G-RVM



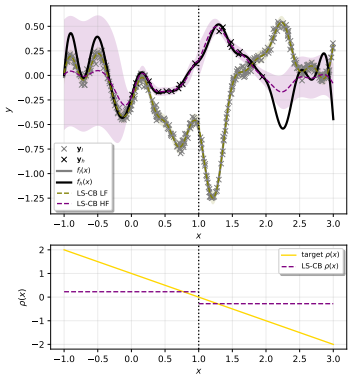
(f) LS-RVM



(g) L-CB



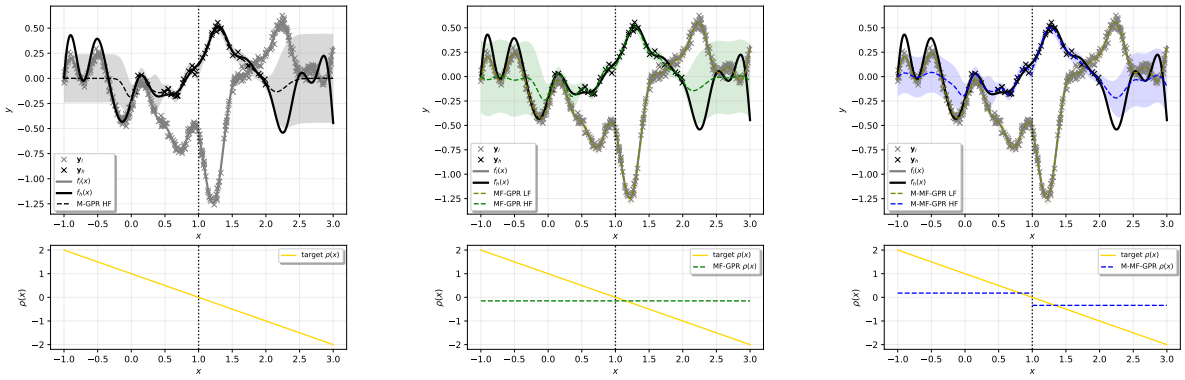
(h) G-CB



(i) LS-CB

Figure A.23: model predictions with 10 high-fids per region of function **a** and data-set 0 of figure A.21.

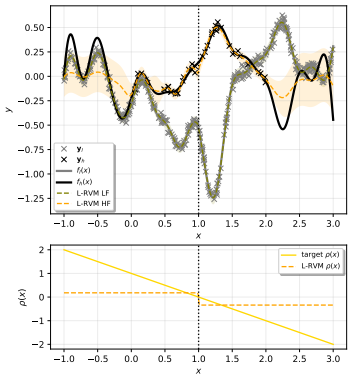
Appendix A: Experimental Results



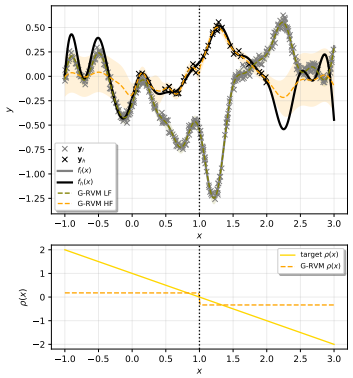
(a) M-GPR

(b) MF-GPR

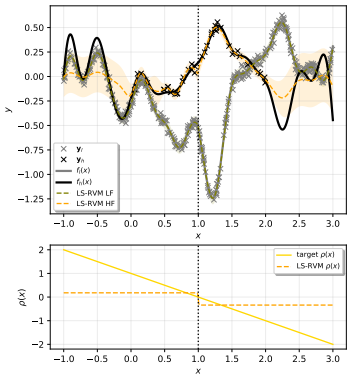
(c) M-MF-GPR



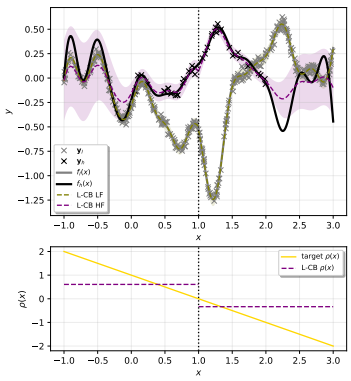
(d) L-RVM



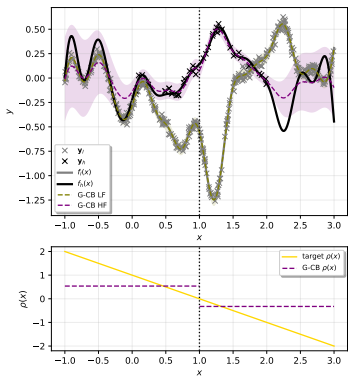
(e) G-RVM



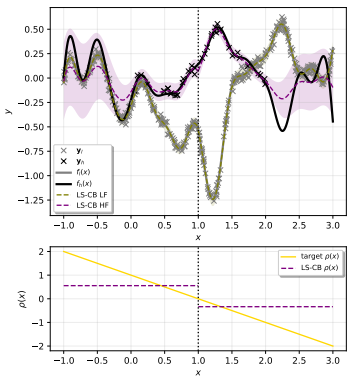
(f) LS-RVM



(g) L-CB



(h) G-CB



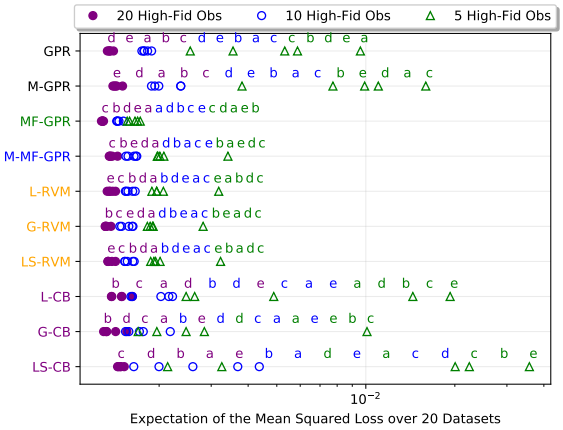
(i) LS-CB

Figure A.24: model predictions with 20 high-fids per region of function **a** and data-set 0 of figure A.21.

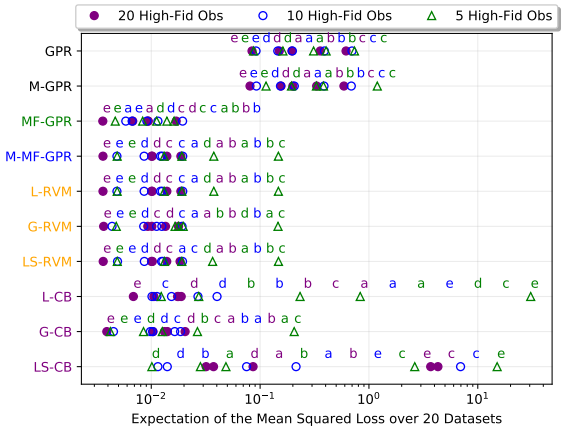
A.3. Inputs Linearly Spaced

A.3.1. Constant ρ case

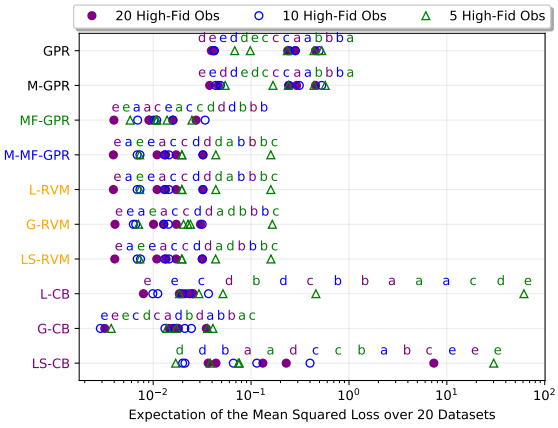
Appendix A: Experimental Results



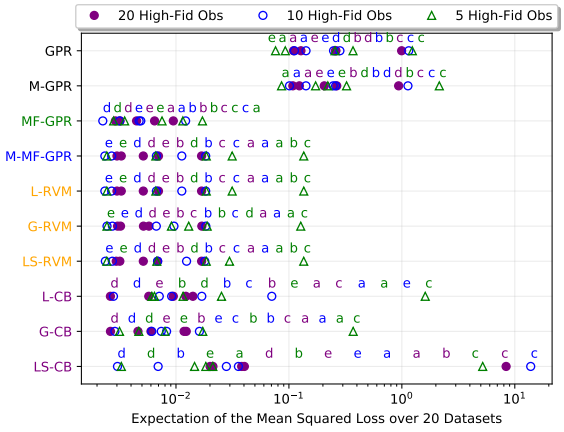
(a) Interpolation regime: $[0,2]$.



(b) Extrapolation regime: $[-1,0] \cup [2,3]$.



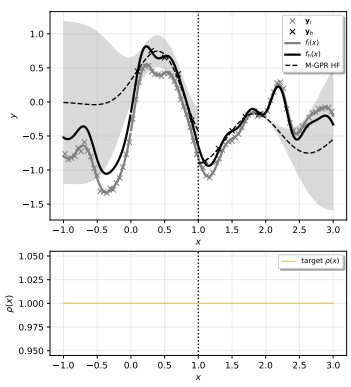
(c) Left extrapolation regime: $[-1,0]$.



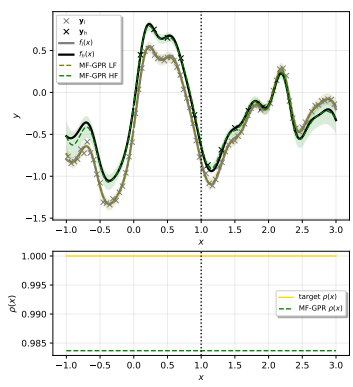
(d) Right extrapolation regime: $[2,3]$.

Figure A.25: constant ρ case, inputs linearly spaced, and 21 low-fids per region.

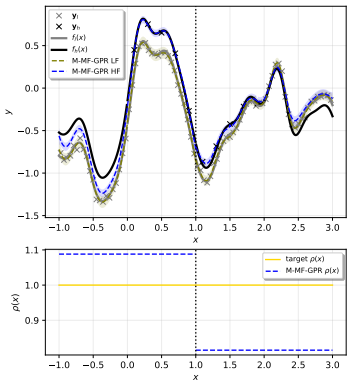
Appendix A: Experimental Results



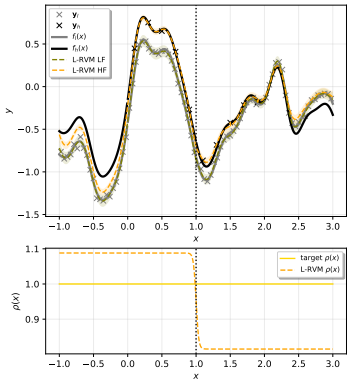
(a) M-GPR



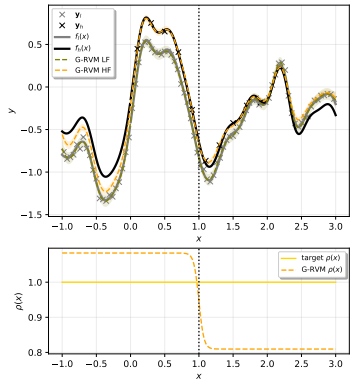
(b) MF-GPR



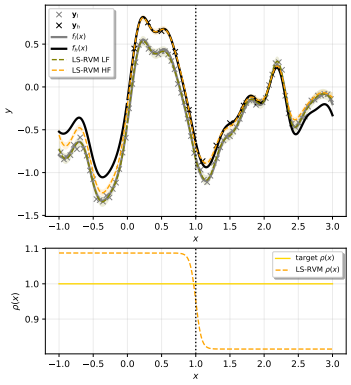
(c) M-MF-GPR



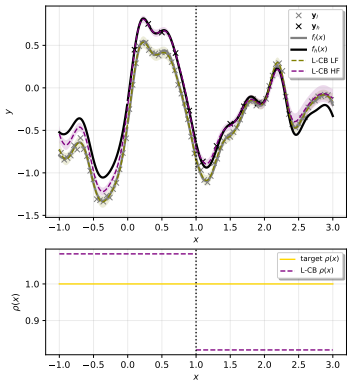
(d) L-RVM



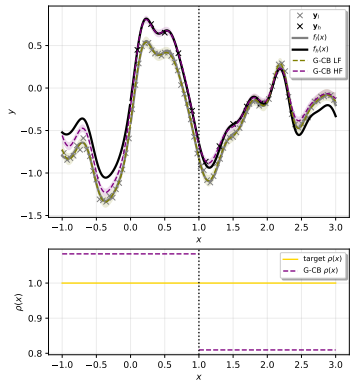
(e) G-RVM



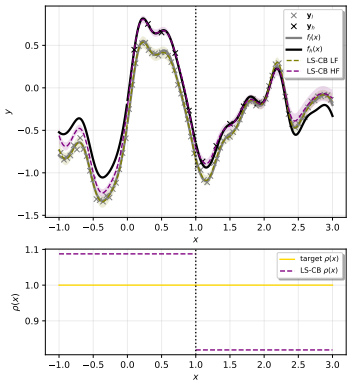
(f) LS-RVM



(g) L-CB



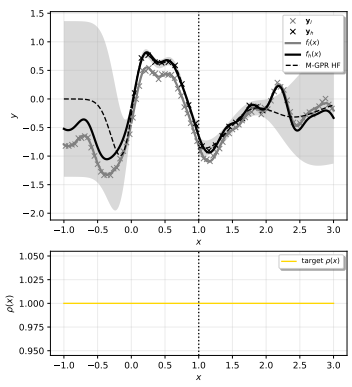
(h) G-CB



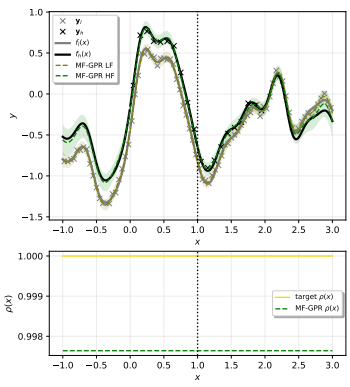
(i) LS-CB

Figure A.26: model predictions with 5 high-fids per region of function **a** and data-set 0 of figure A.25.

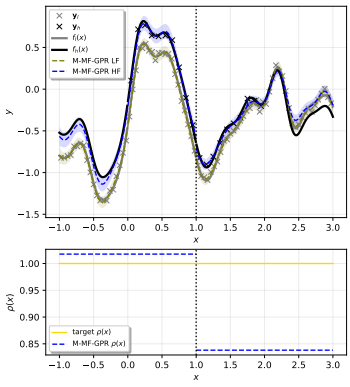
Appendix A: Experimental Results



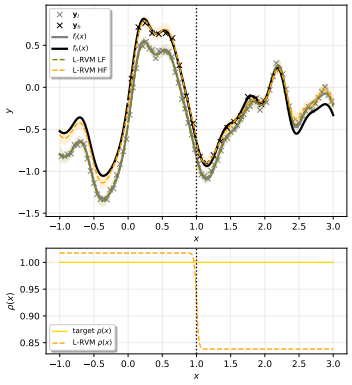
(a) M-GPR



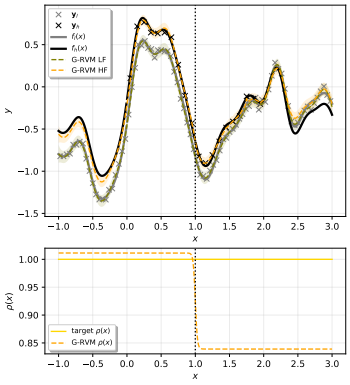
(b) MF-GPR



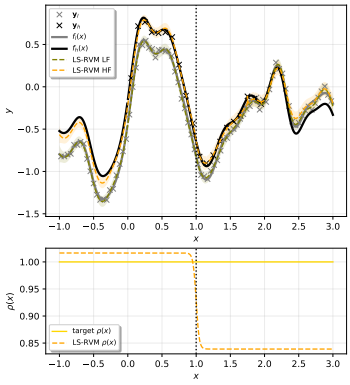
(c) M-MF-GPR



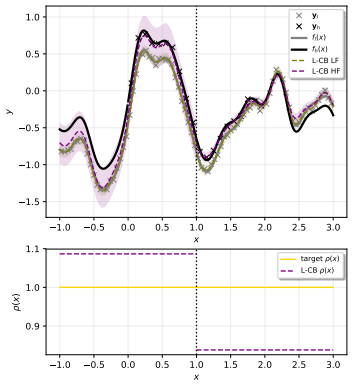
(d) L-RVM



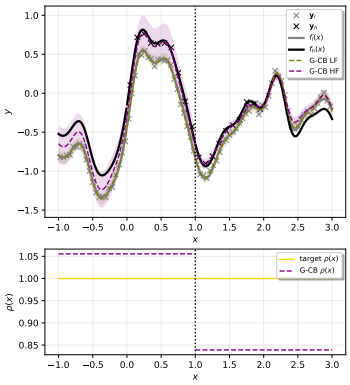
(e) G-RVM



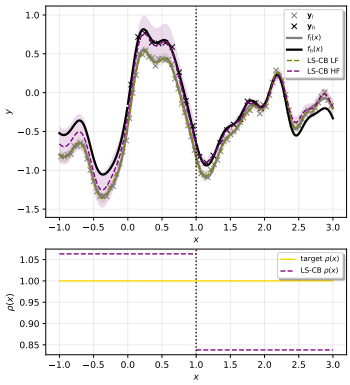
(f) LS-RVM



(g) L-CB



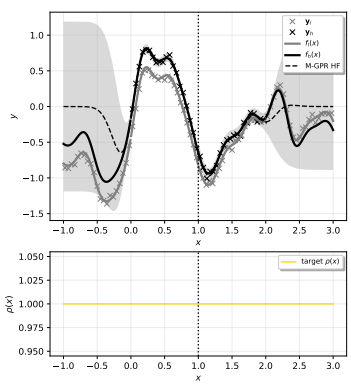
(h) G-CB



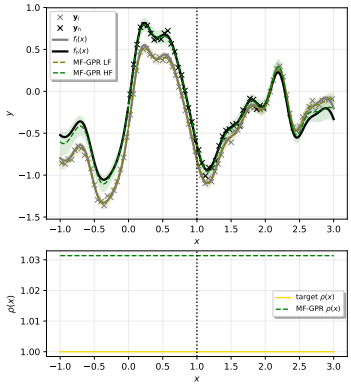
(i) LS-CB

Figure A.27: model predictions with 10 high-fids per region of function **a** and data-set 0 of figure A.25.

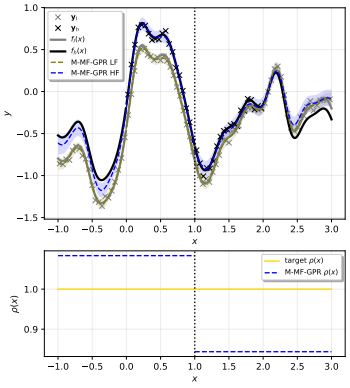
Appendix A: Experimental Results



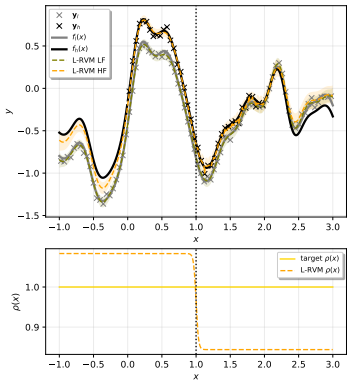
(a) M-GPR



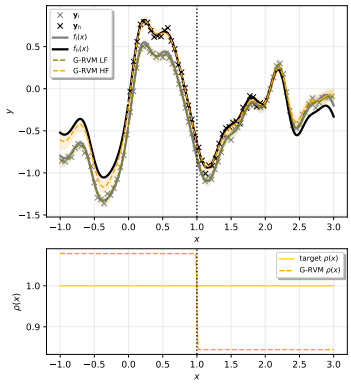
(b) MF-GPR



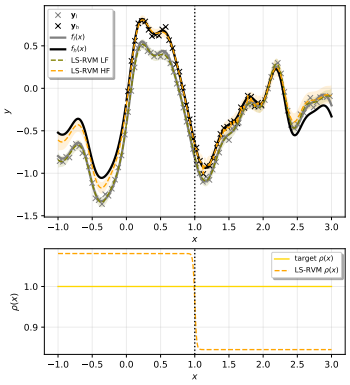
(c) M-MF-GPR



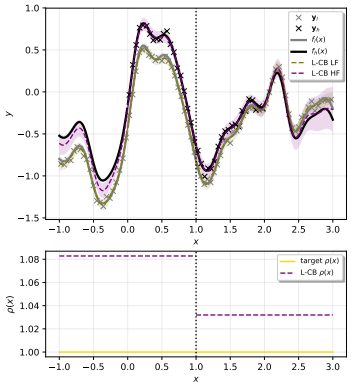
(d) L-RVM



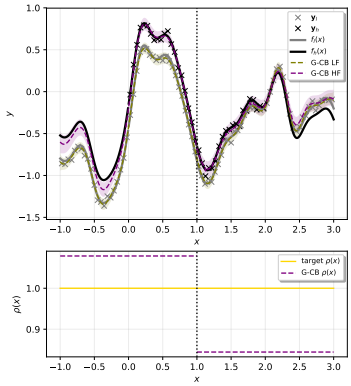
(e) G-RVM



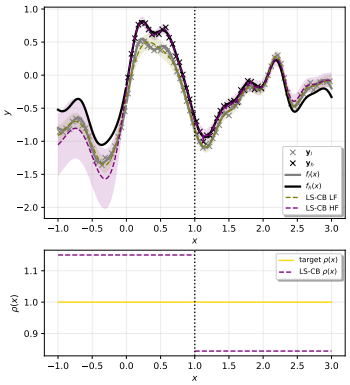
(f) LS-RVM



(g) L-CB



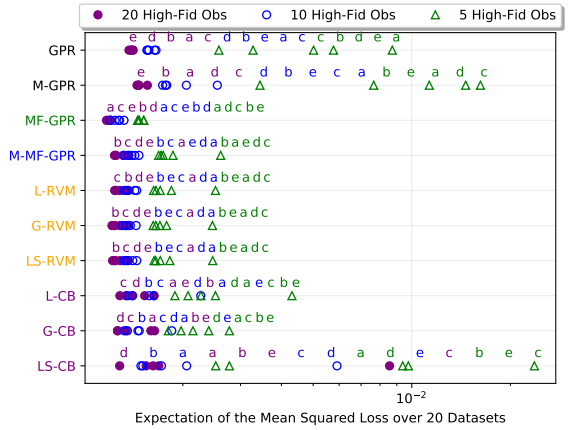
(h) G-CB



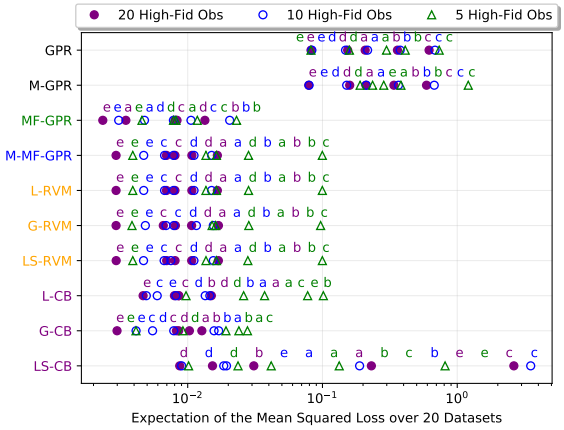
(i) LS-CB

Figure A.28: model predictions with 20 high-fids per region of function **a** and data-set 0 of figure A.25.

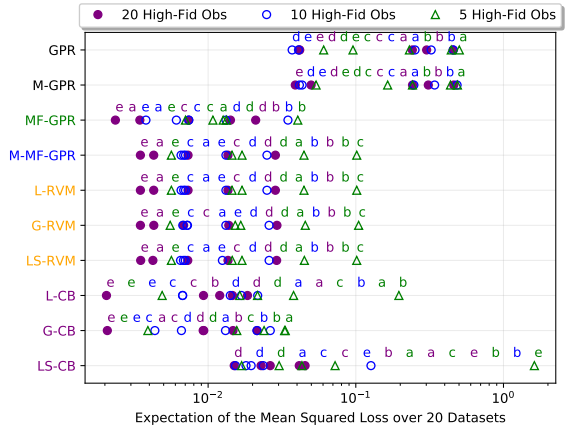
Appendix A: Experimental Results



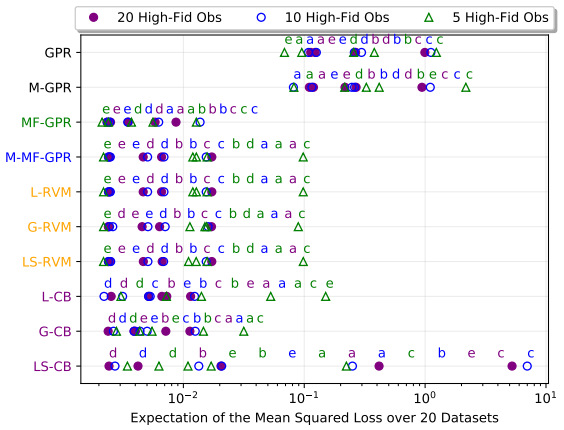
(a) Interpolation regime: $[0,2]$.



(b) Extrapolation regime: $[-1,0] \cup [2,3]$.



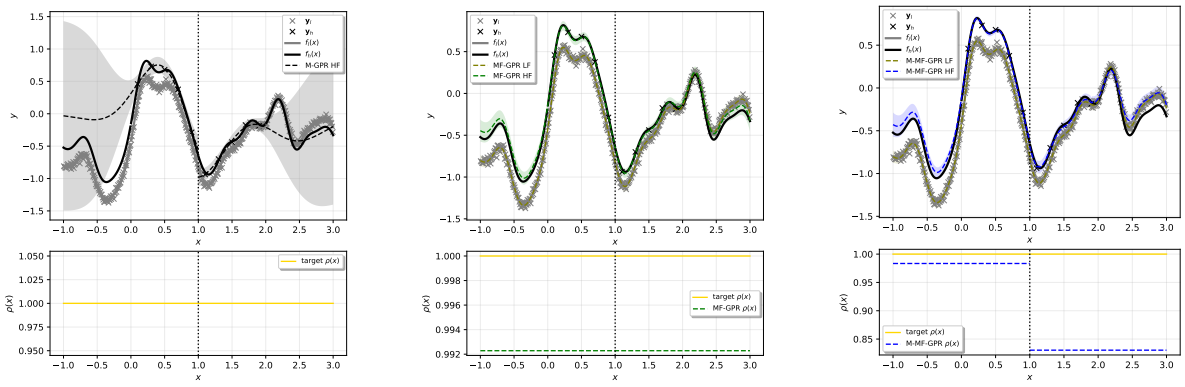
(c) Left extrapolation regime: $[-1,0]$.



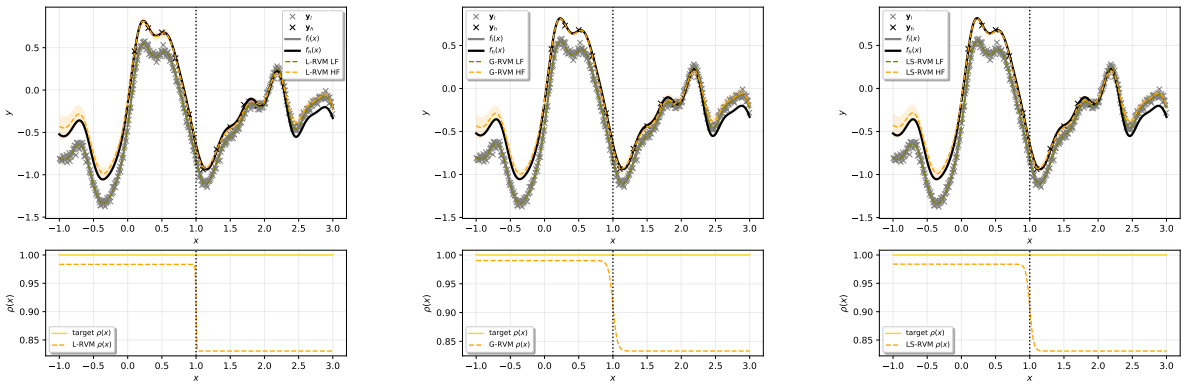
(d) Right extrapolation regime: $[2,3]$.

Figure A.29: constant ρ case, inputs linearly spaced, and 101 low-fids per region.

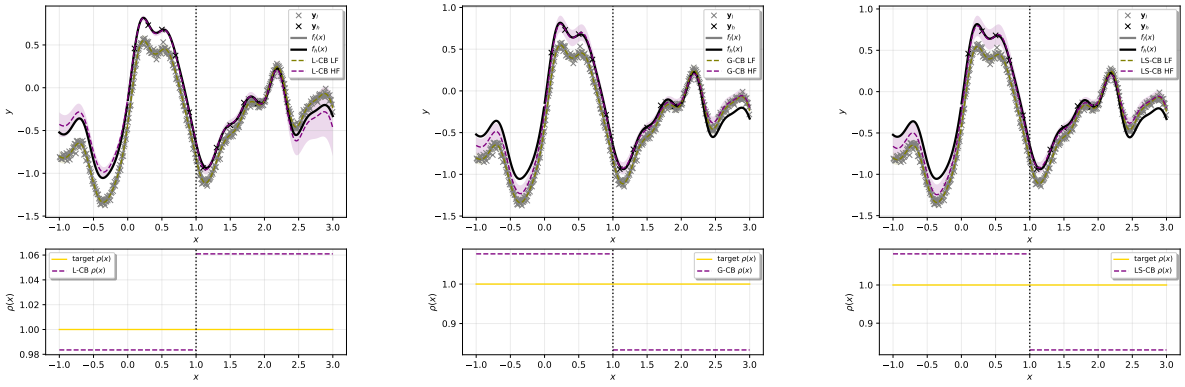
Appendix A: Experimental Results



(a) M-GPR (b) MF-GPR (c) M-MF-GPR



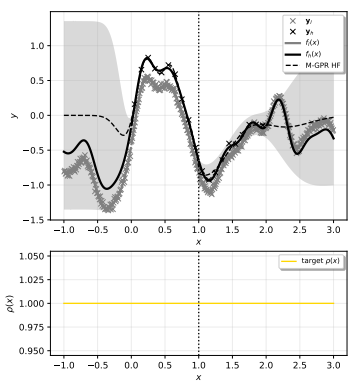
(d) L-RVM (e) G-RVM (f) LS-RVM



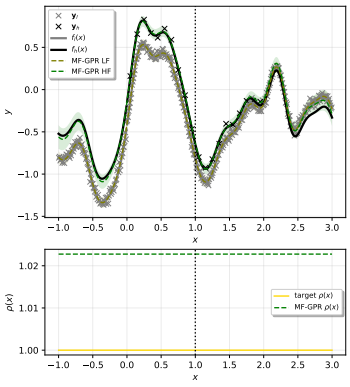
(g) L-CB (h) G-CB (i) LS-CB

Figure A.30: model predictions with 5 high-fids per region of function **a** and data-set 0 of figure A.29.

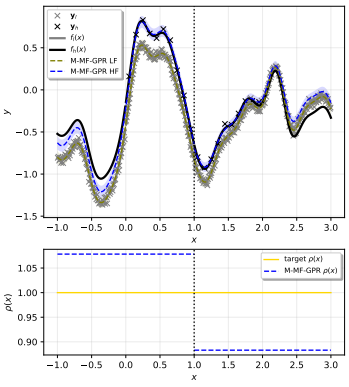
Appendix A: Experimental Results



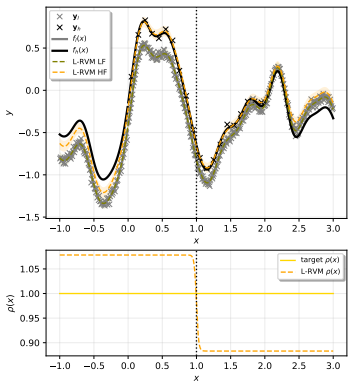
(a) M-GPR



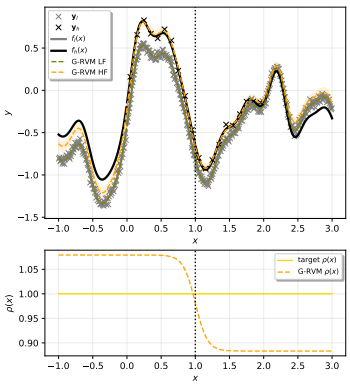
(b) MF-GPR



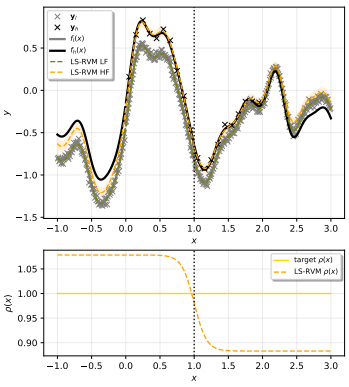
(c) M-MF-GPR



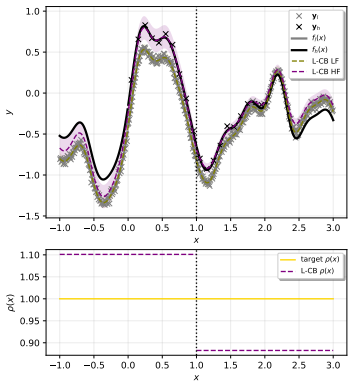
(d) L-RVM



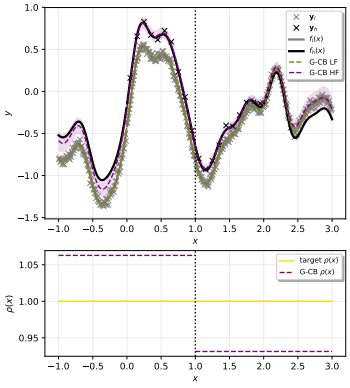
(e) G-RVM



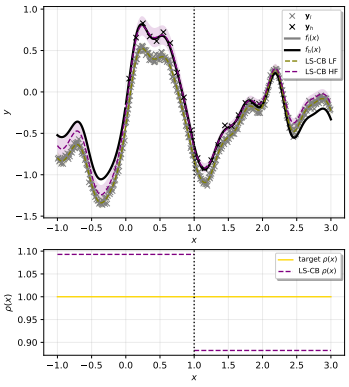
(f) LS-RVM



(g) L-CB



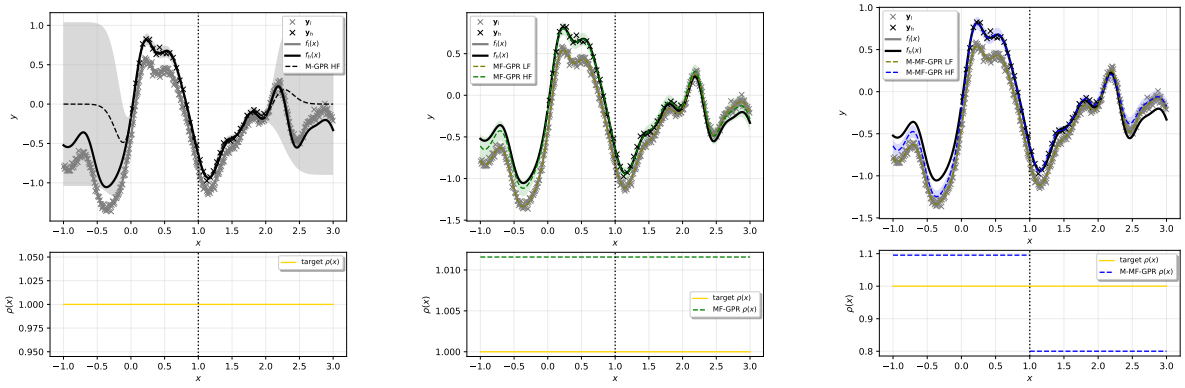
(h) G-CB



(i) LS-CB

Figure A.31: model predictions with 10 high-fids per region of function **a** and data-set 0 of figure A.29.

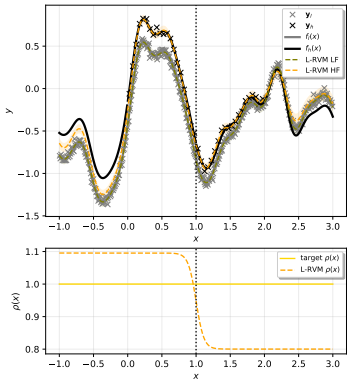
Appendix A: Experimental Results



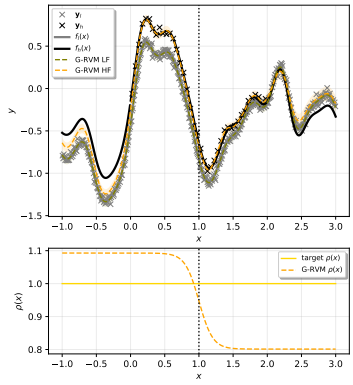
(a) M-GPR

(b) MF-GPR

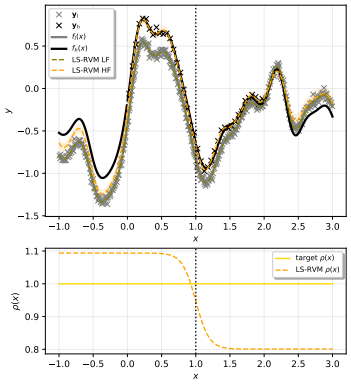
(c) M-MF-GPR



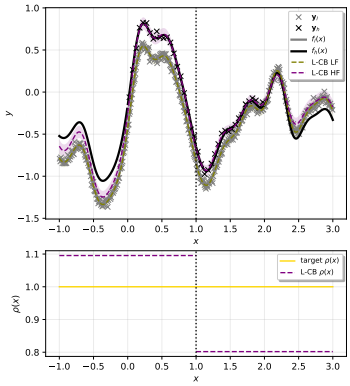
(d) L-RVM



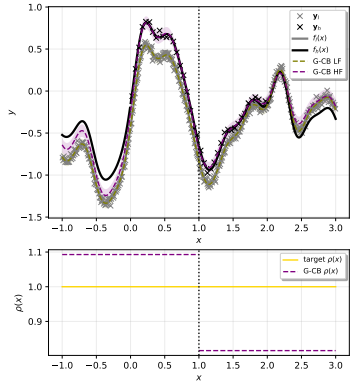
(e) G-RVM



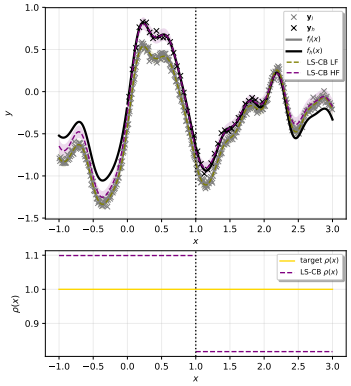
(f) LS-RVM



(g) L-CB



(h) G-CB

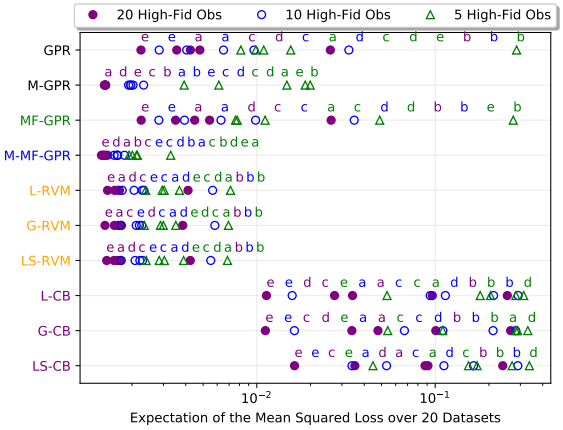


(i) LS-CB

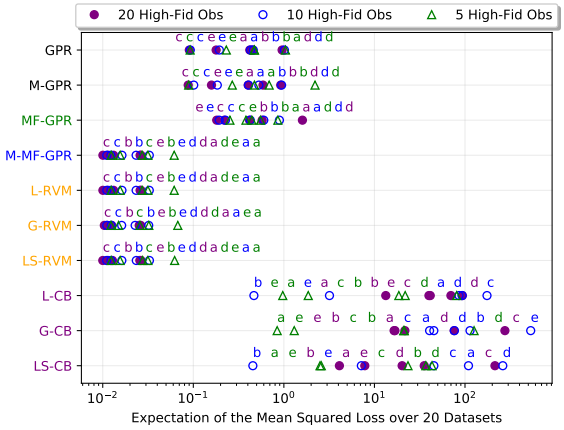
Figure A.32: model predictions with 20 high-fids per region of function **a** and data-set 0 of figure A.29.

A.3.2. Discontinuous ρ case

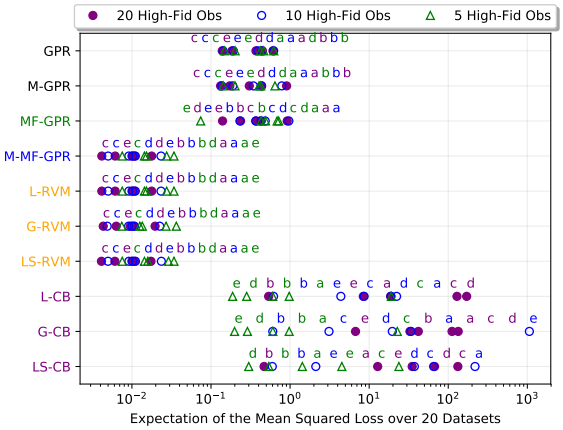
Appendix A: Experimental Results



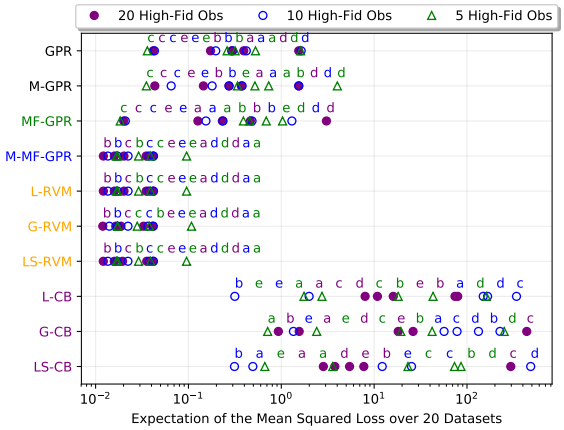
(a) Interpolation regime: $[0,2]$.



(b) Extrapolation regime: $[-1,0] \cup [2,3]$.



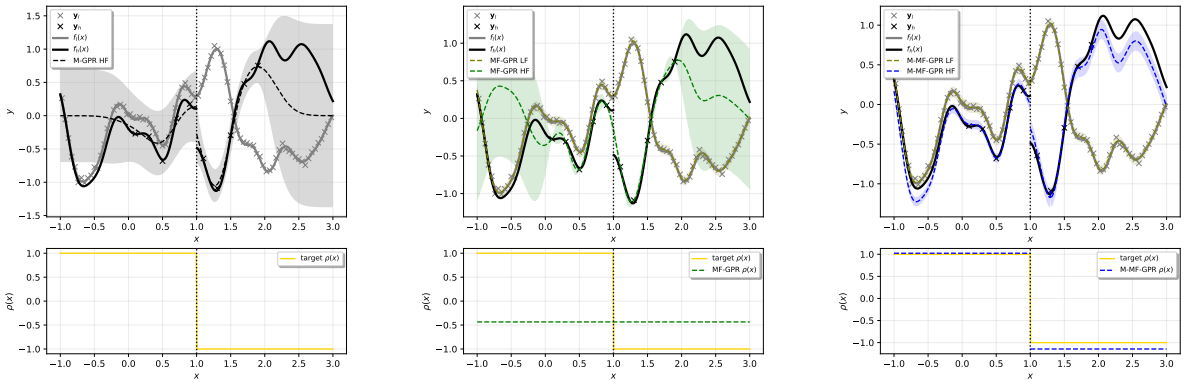
(c) Left extrapolation regime: $[-1,0]$.



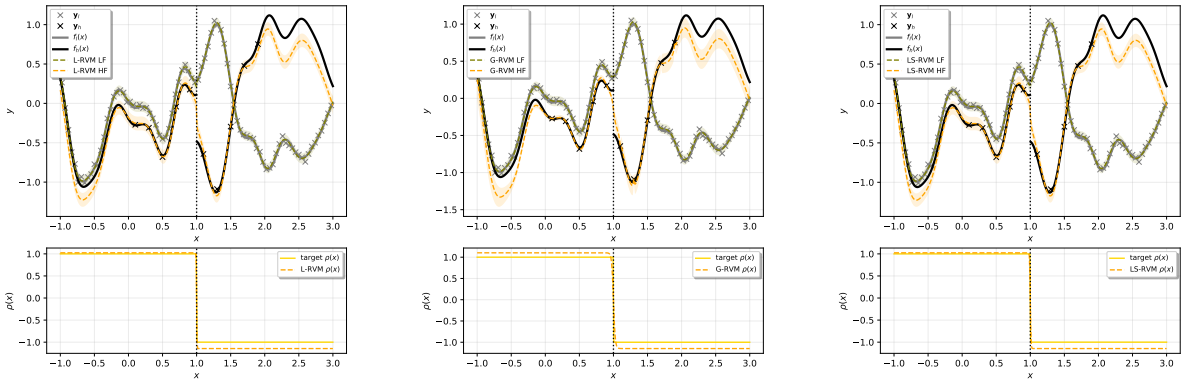
(d) Right extrapolation regime: $[2,3]$.

Figure A.33: discontinuous ρ case, inputs linearly spaced, and 21 low-fids per region.

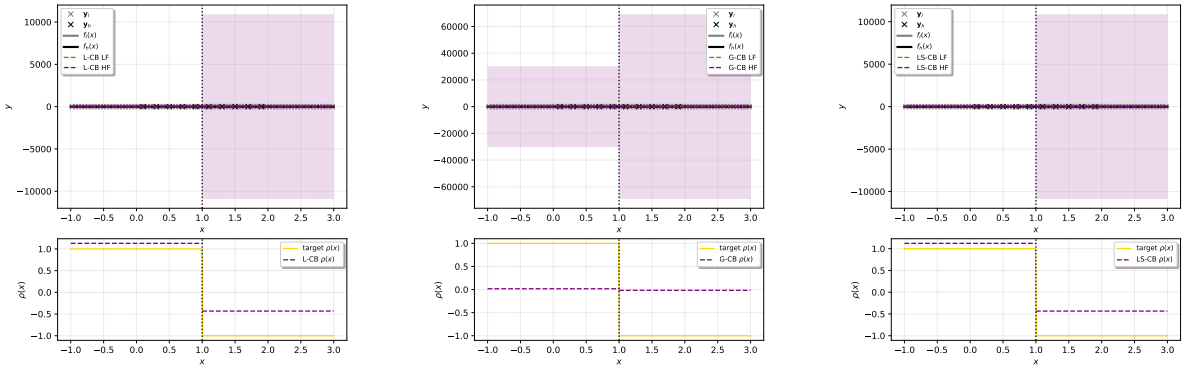
Appendix A: Experimental Results



(a) M-GPR (b) MF-GPR (c) M-MF-GPR



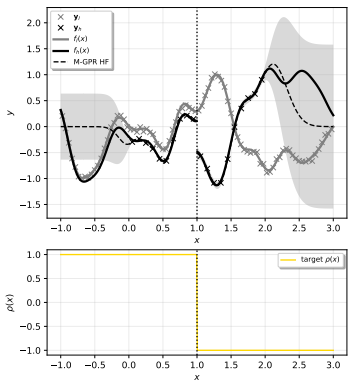
(d) L-RVM (e) G-RVM (f) LS-RVM



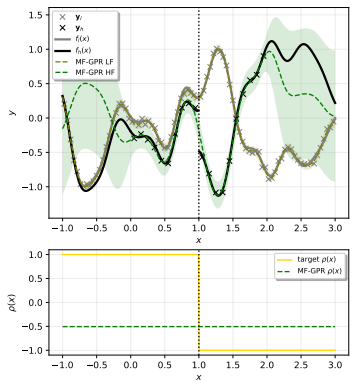
(g) L-CB (h) G-CB (i) LS-CB

Figure A.34: model predictions with 5 high-fids per region of function **a** and data-set 0 of figure A.33.

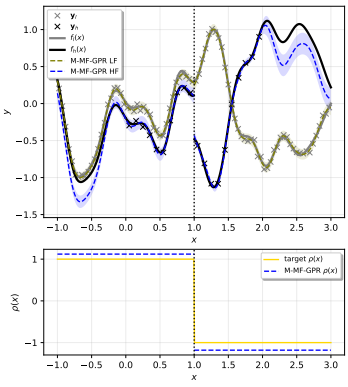
Appendix A: Experimental Results



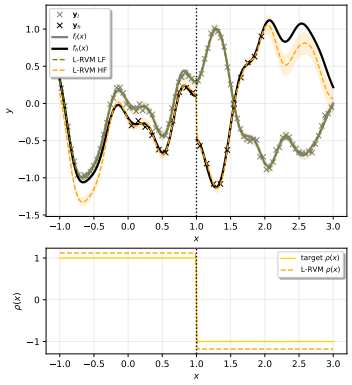
(a) M-GPR



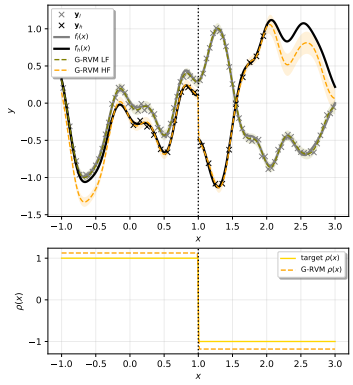
(b) MF-GPR



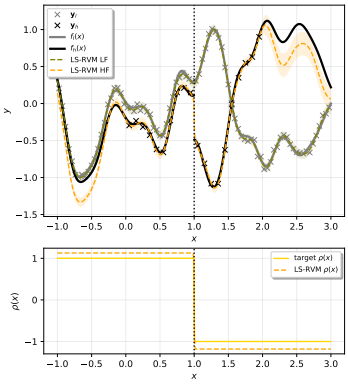
(c) M-MF-GPR



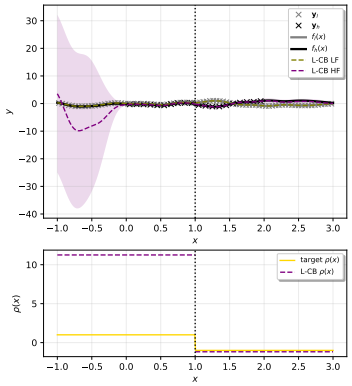
(d) L-RVM



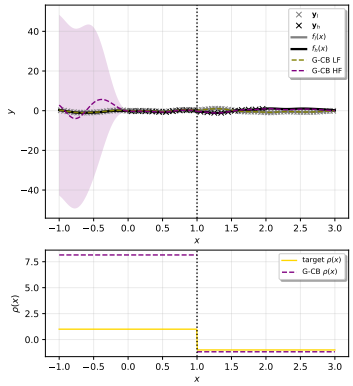
(e) G-RVM



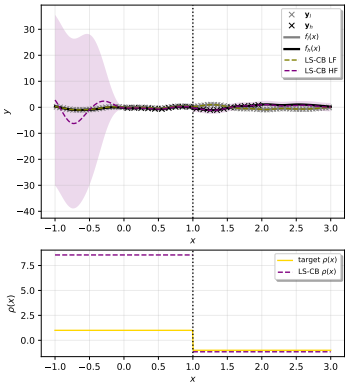
(f) LS-RVM



(g) L-CB



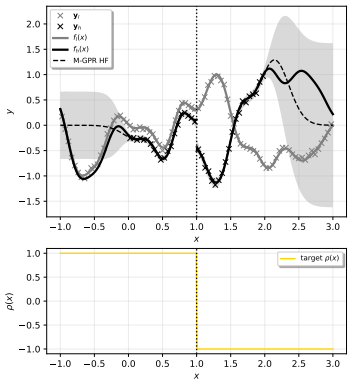
(h) G-CB



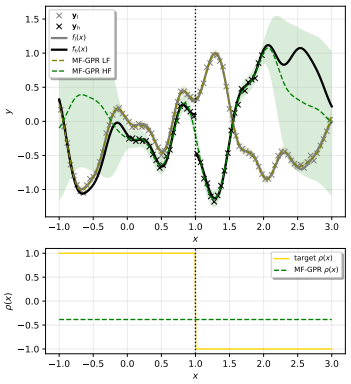
(i) LS-CB

Figure A.35: model predictions with 10 high-fids per region of function **a** and data-set 0 of figure A.33.

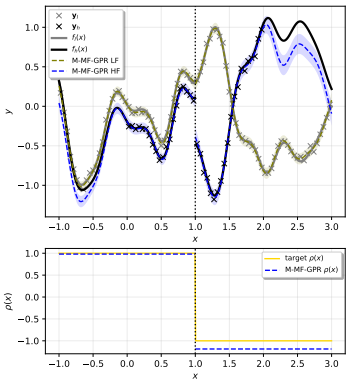
Appendix A: Experimental Results



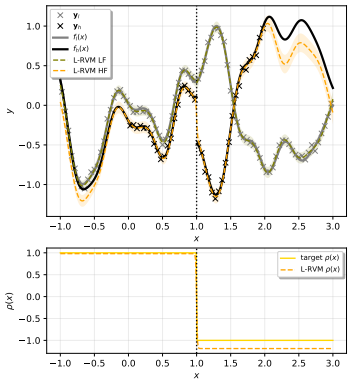
(a) M-GPR



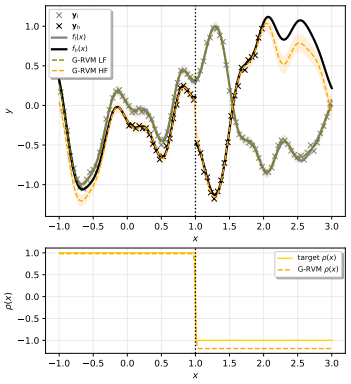
(b) MF-GPR



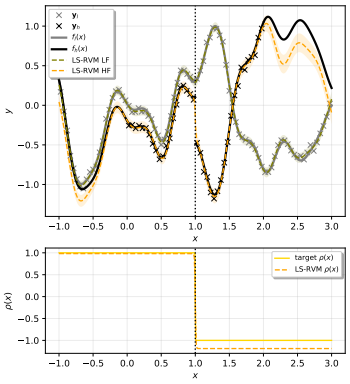
(c) M-MF-GPR



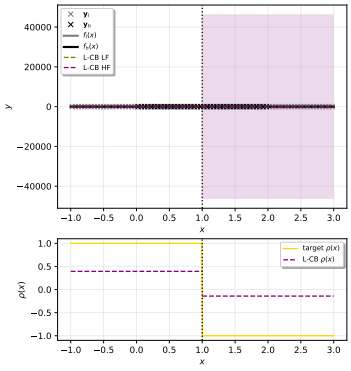
(d) L-RVM



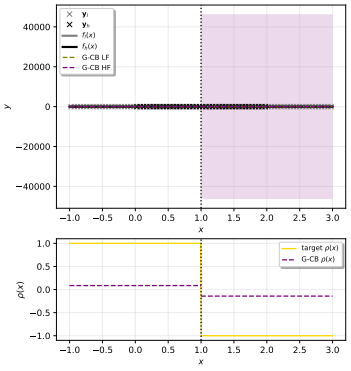
(e) G-RVM



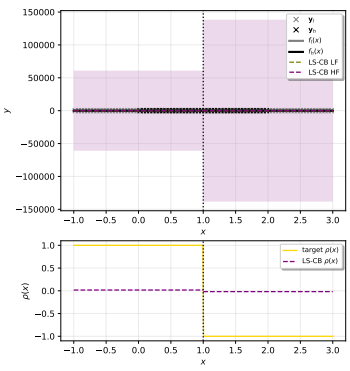
(f) LS-RVM



(g) L-CB



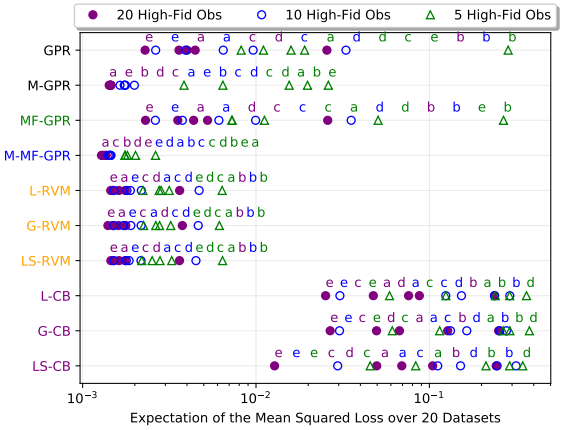
(h) G-CB



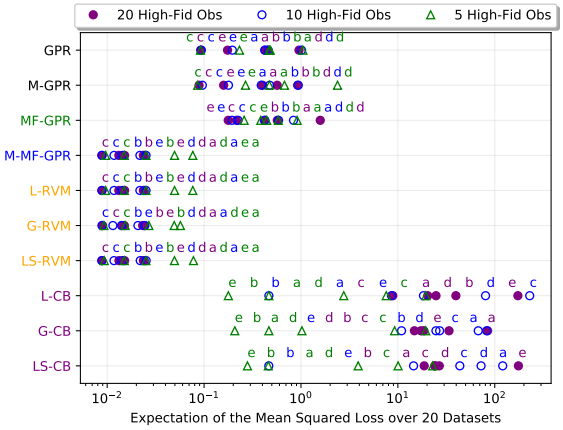
(i) LS-CB

Figure A.36: model predictions with 20 high-fids per region of function **a** and data-set 0 of figure A.33.

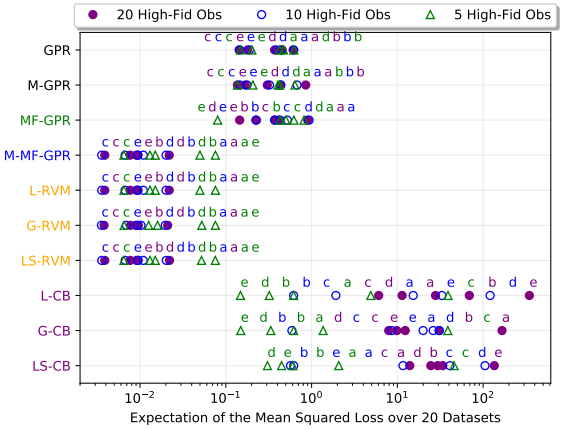
Appendix A: Experimental Results



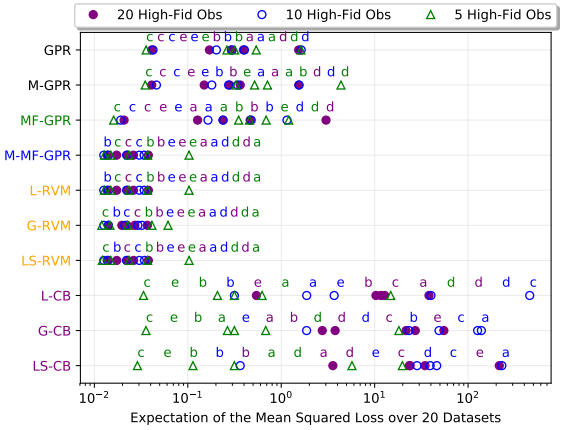
(a) Interpolation regime: $[0,2]$.



(b) Extrapolation regime: $[-1,0] \cup [2,3]$.



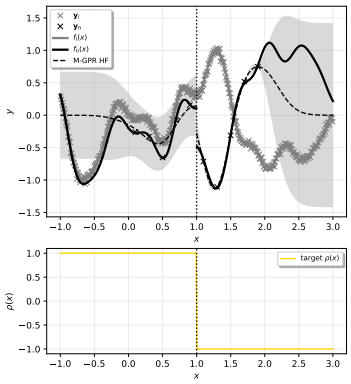
(c) Left extrapolation regime: $[-1,0]$.



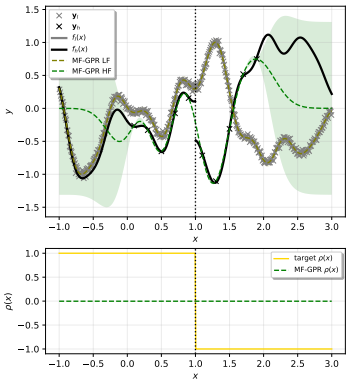
(d) Right extrapolation regime: $[2,3]$.

Figure A.37: discontinuous ρ case, inputs linearly spaced, and 101 low-fids per region.

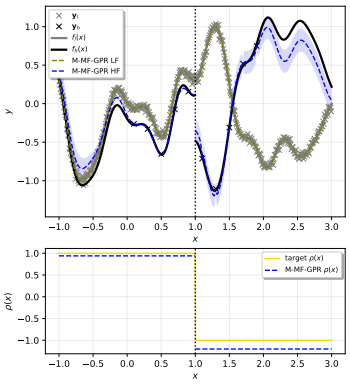
Appendix A: Experimental Results



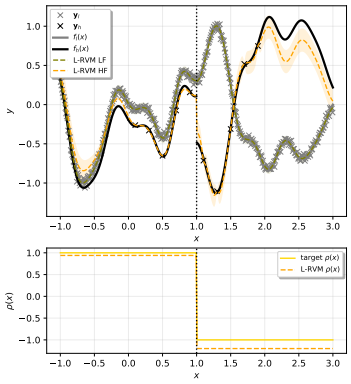
(a) M-GPR



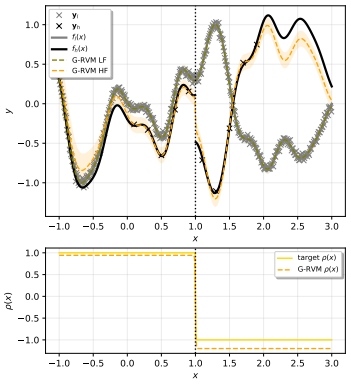
(b) MF-GPR



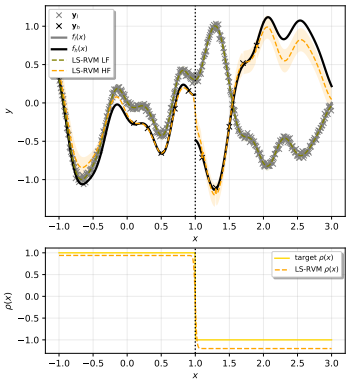
(c) M-MF-GPR



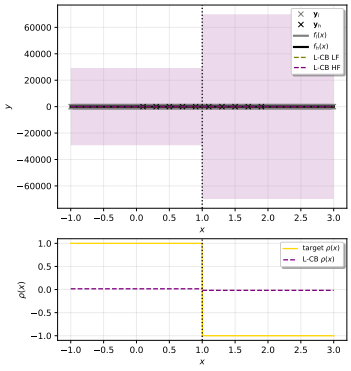
(d) L-RVM



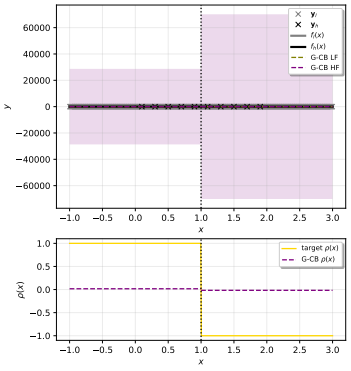
(e) G-RVM



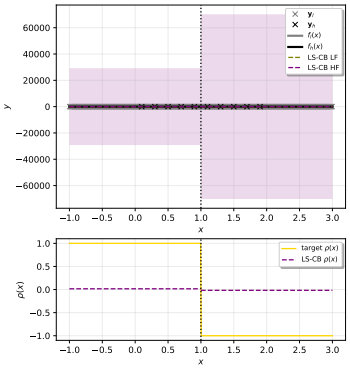
(f) LS-RVM



(g) L-CB



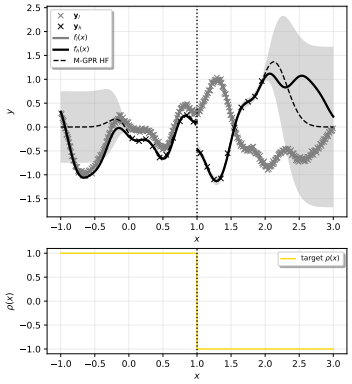
(h) G-CB



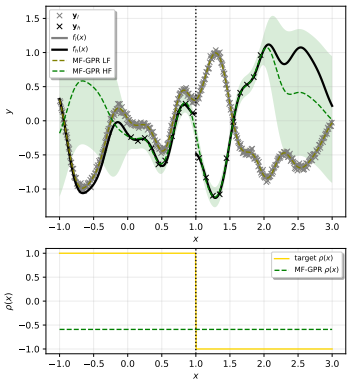
(i) LS-CB

Figure A.38: model predictions with 5 high-fids per region of function **a** and data-set 0 of figure A.37.

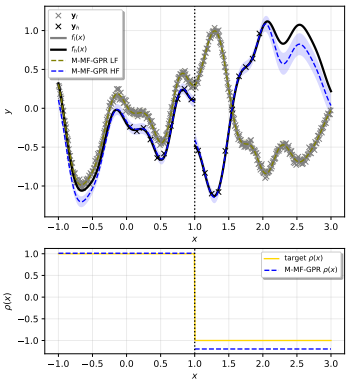
Appendix A: Experimental Results



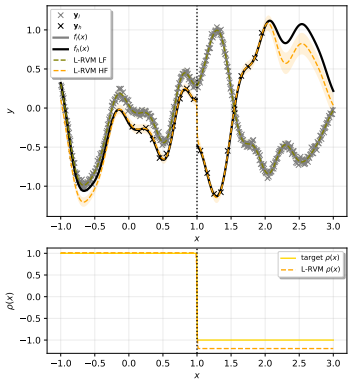
(a) M-GPR



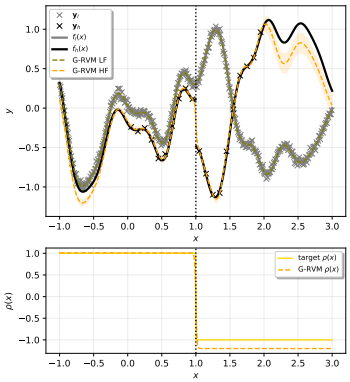
(b) MF-GPR



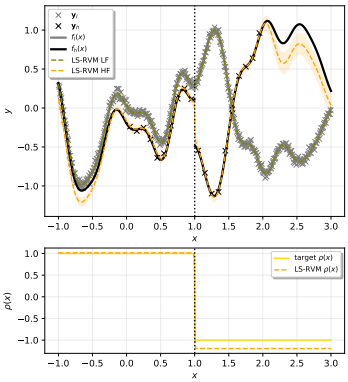
(c) M-MF-GPR



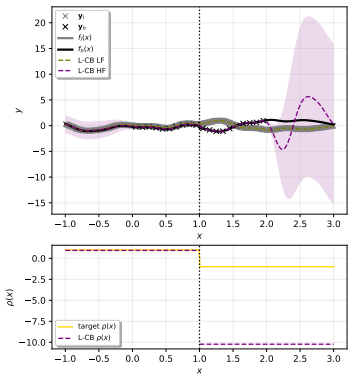
(d) L-RVM



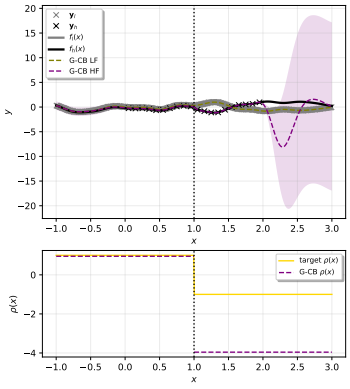
(e) G-RVM



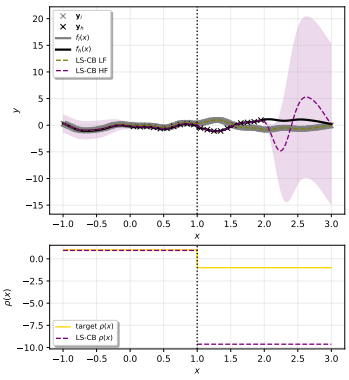
(f) LS-RVM



(g) L-CB



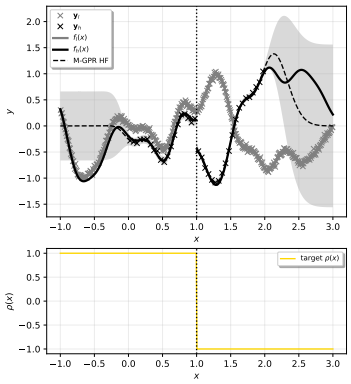
(h) G-CB



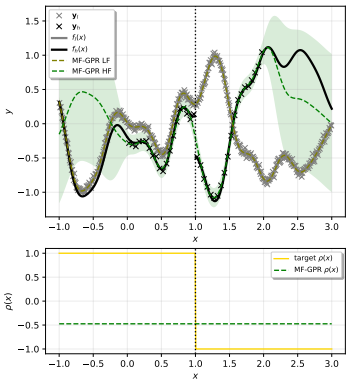
(i) LS-CB

Figure A.39: model predictions with 10 high-fids per region of function **a** and data-set 0 of figure A.37.

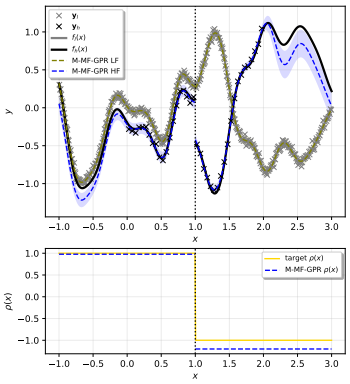
Appendix A: Experimental Results



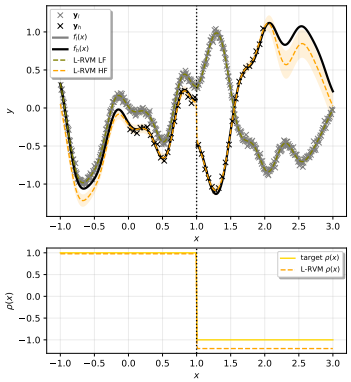
(a) M-GPR



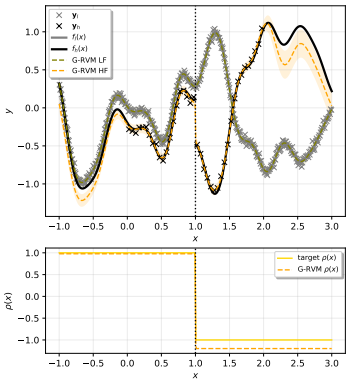
(b) MF-GPR



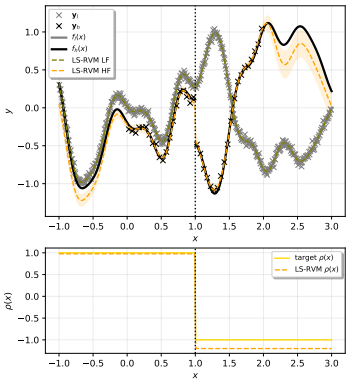
(c) M-MF-GPR



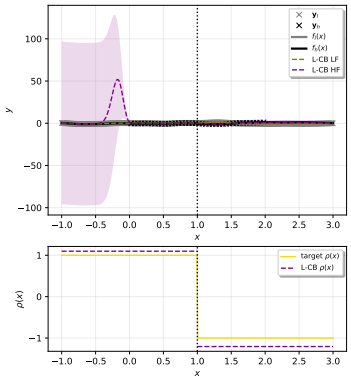
(d) L-RVM



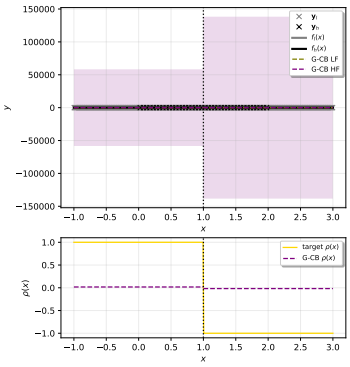
(e) G-RVM



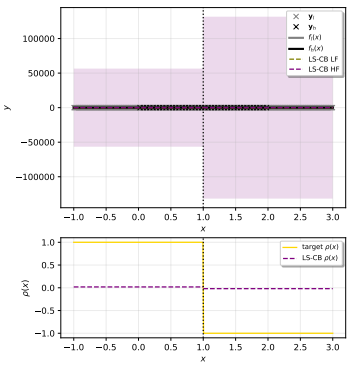
(f) LS-RVM



(g) L-CB



(h) G-CB

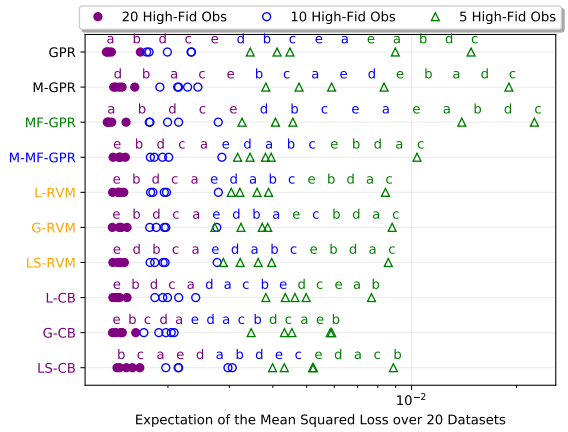


(i) LS-CB

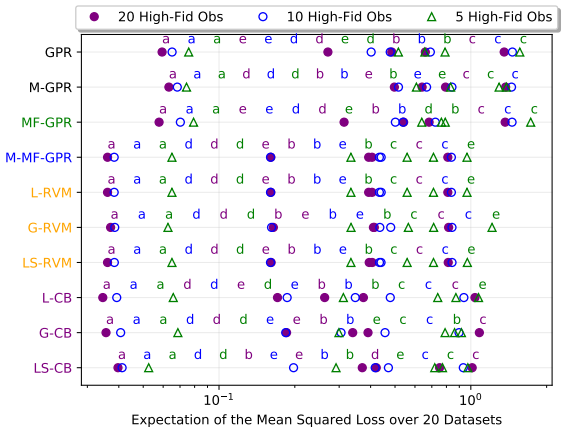
Figure A.40: model predictions with 20 high-fids per region of function **a** and data-set 0 of figure A.37.

A.3.3. Linearly varying ρ case

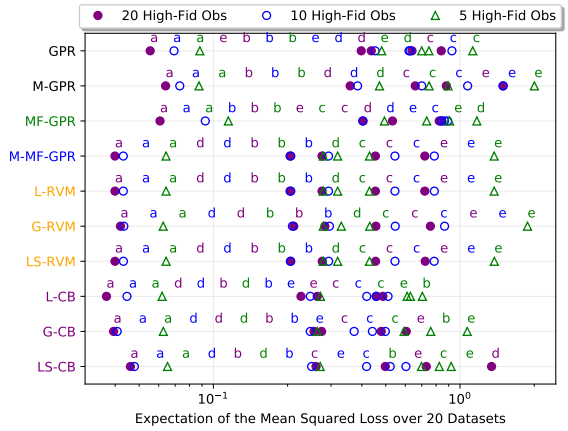
Appendix A: Experimental Results



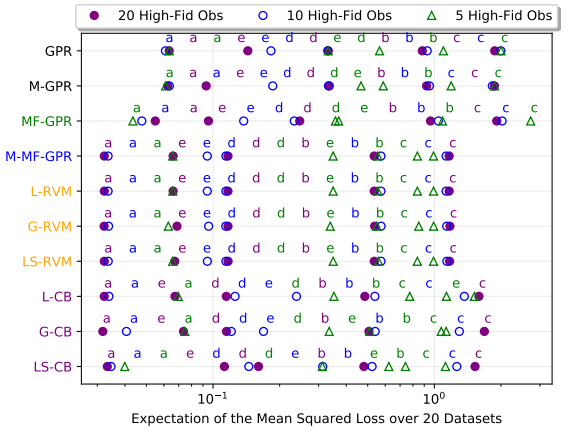
(a) Interpolation regime: $[0,2]$.



(b) Extrapolation regime: $[-1,0] \cup [2,3]$.



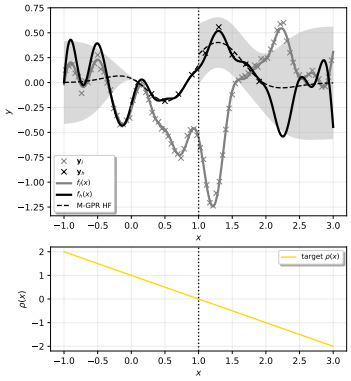
(c) Left extrapolation regime: $[-1,0]$.



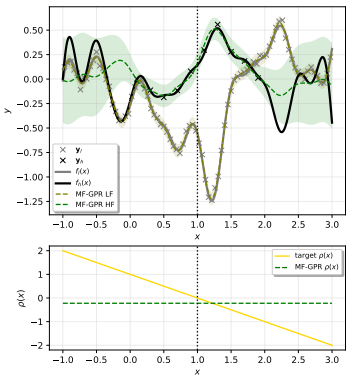
(d) Right extrapolation regime: $[2,3]$.

Figure A.41: linearly varying ρ case, inputs linearly spaced, and 21 low-fids per region.

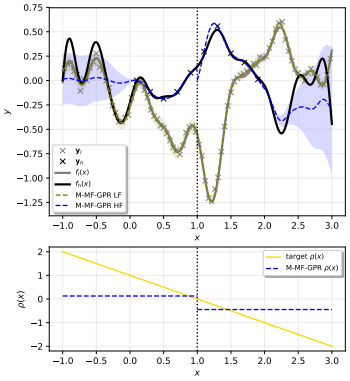
Appendix A: Experimental Results



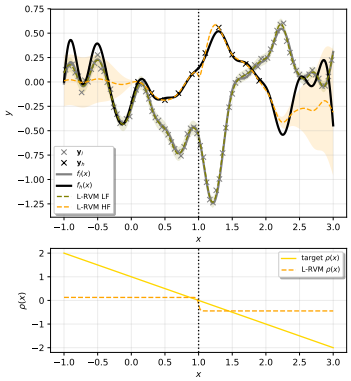
(a) M-GPR



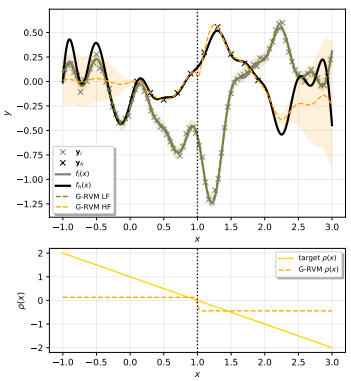
(b) MF-GPR



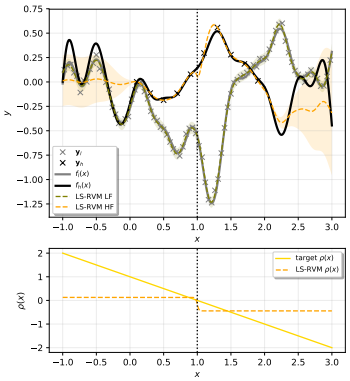
(c) M-MF-GPR



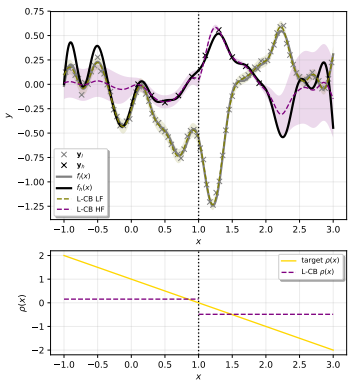
(d) L-RVM



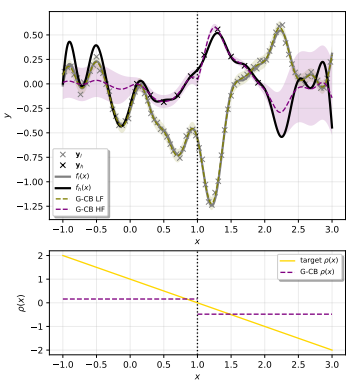
(e) G-RVM



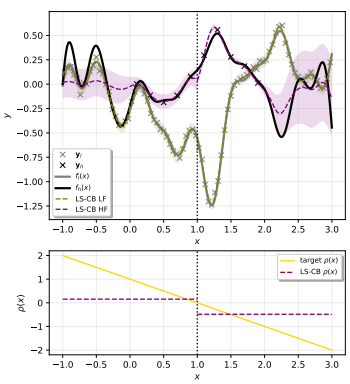
(f) LS-RVM



(g) L-CB



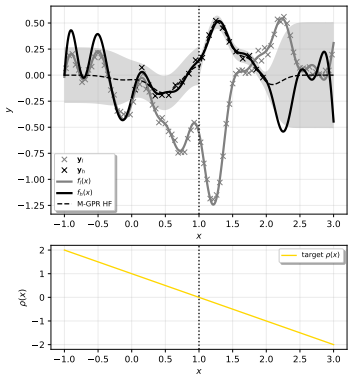
(h) G-CB



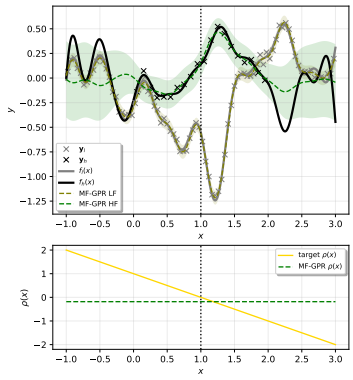
(i) LS-CB

Figure A.42: model predictions with 5 high-fids per region of function **a** and data-set 0 of figure A.41.

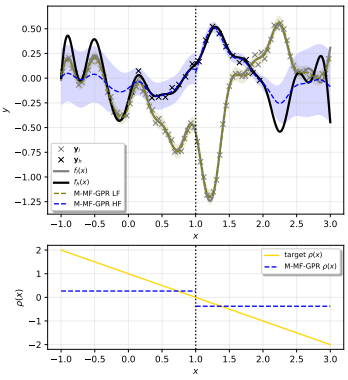
Appendix A: Experimental Results



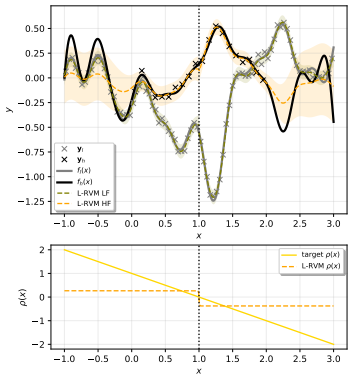
(a) M-GPR



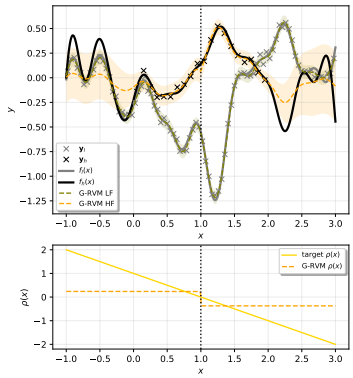
(b) MF-GPR



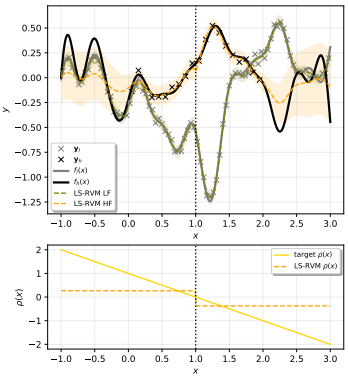
(c) M-MF-GPR



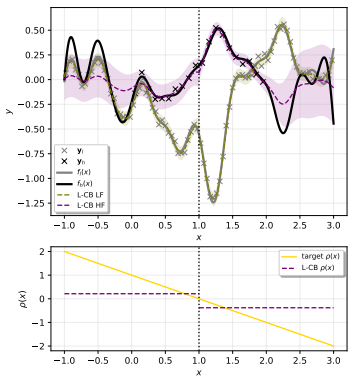
(d) L-RVM



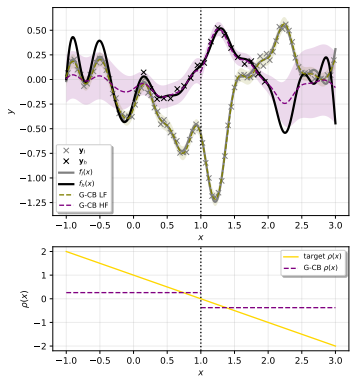
(e) G-RVM



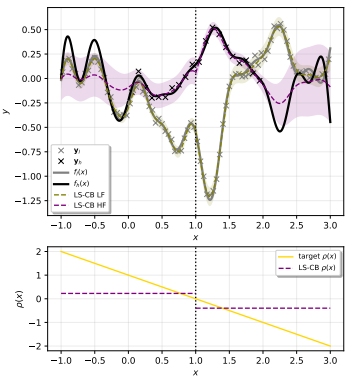
(f) LS-RVM



(g) L-CB



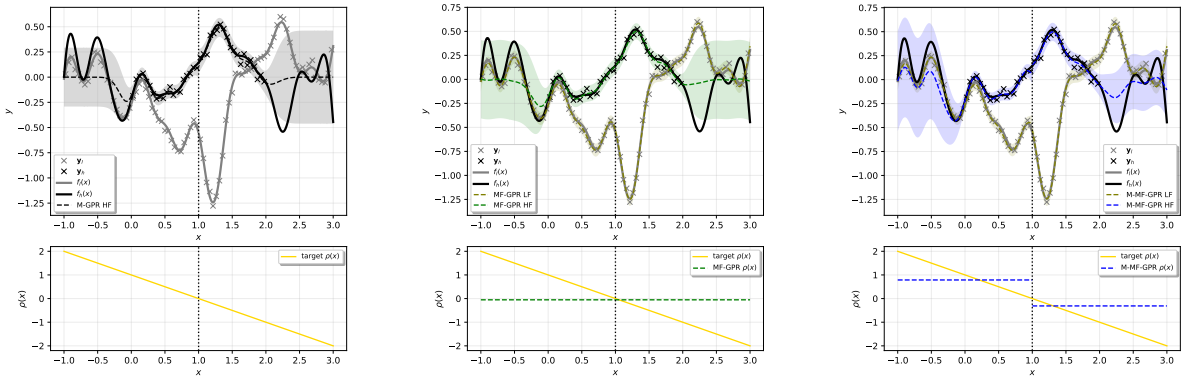
(h) G-CB



(i) LS-CB

Figure A.43: model predictions with 10 high-fids per region of function **a** and data-set 0 of figure A.41.

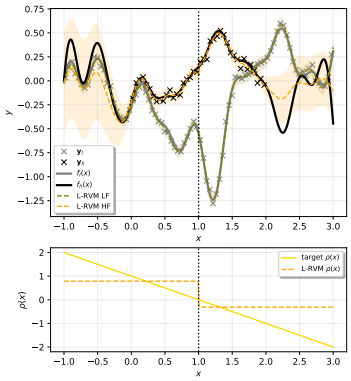
Appendix A: Experimental Results



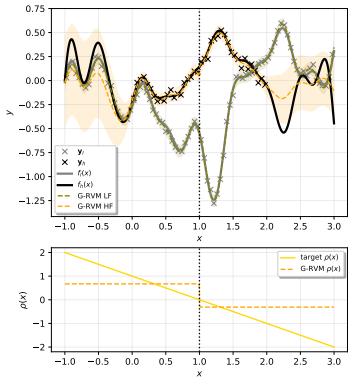
(a) M-GPR

(b) MF-GPR

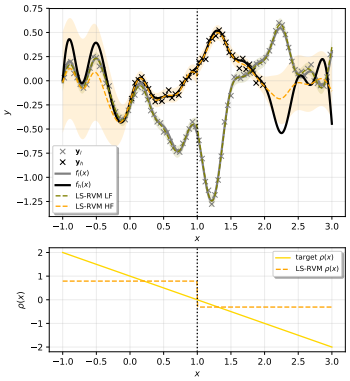
(c) M-MF-GPR



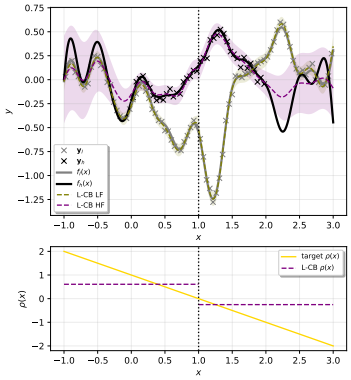
(d) L-RVM



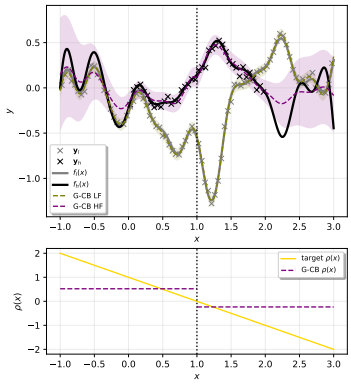
(e) G-RVM



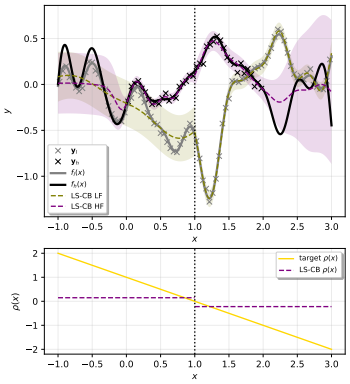
(f) LS-RVM



(g) L-CB



(h) G-CB

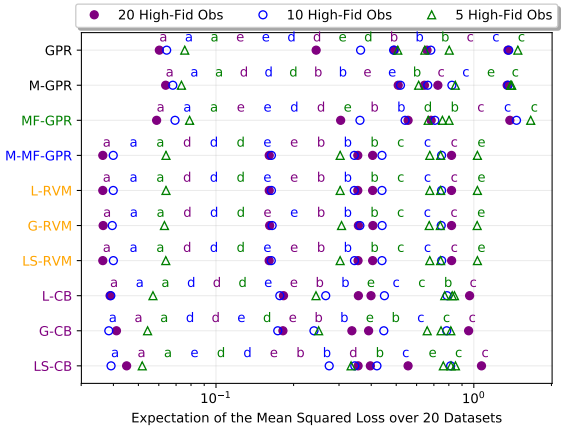


(i) LS-CB

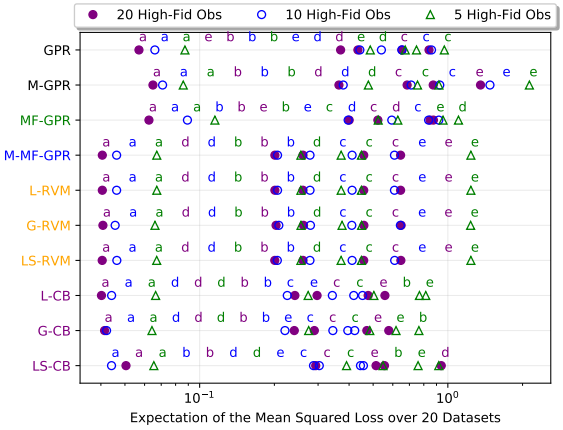
Figure A.44: model predictions with 20 high-fids per region of function **a** and data-set 0 of figure A.41.

Appendix A: Experimental Results

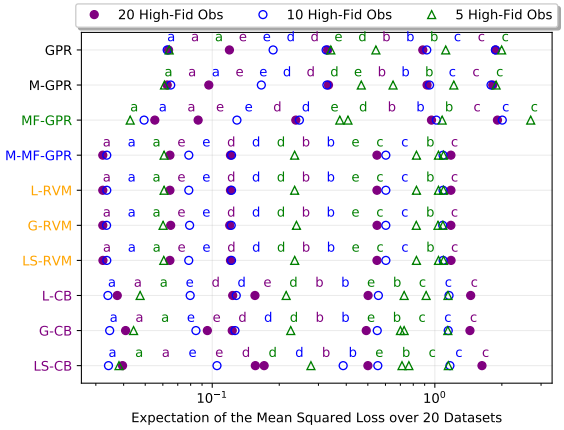
(a) Interpolation regime: $[0,2]$.



(b) Extrapolation regime: $[-1,0] \cup [2,3]$.



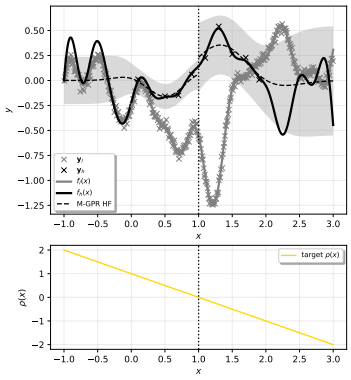
(c) Left extrapolation regime: $[-1,0]$.



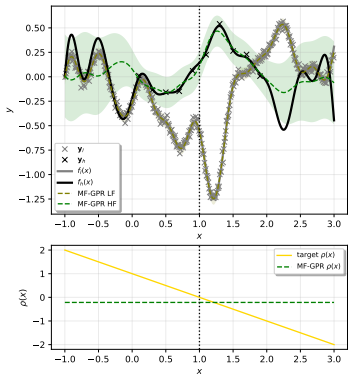
(d) Right extrapolation regime: $[2,3]$.

Figure A.45: linearly varying ρ case, inputs linearly spaced, and 101 low-fids per region.

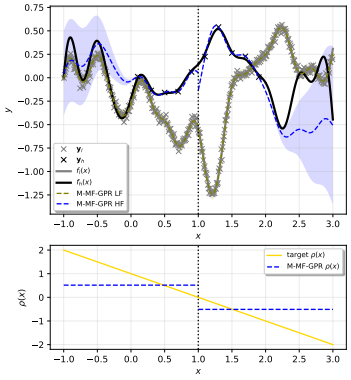
Appendix A: Experimental Results



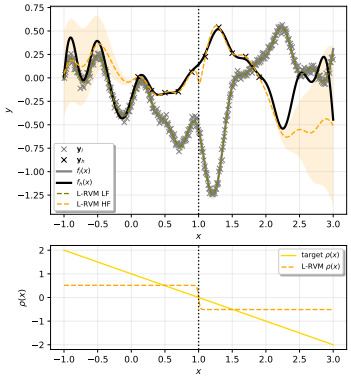
(a) M-GPR



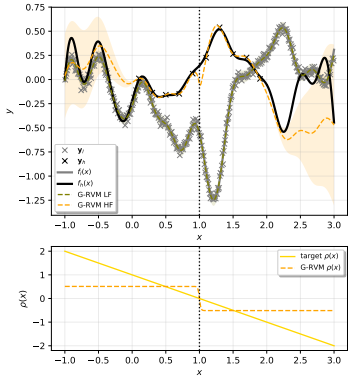
(b) MF-GPR



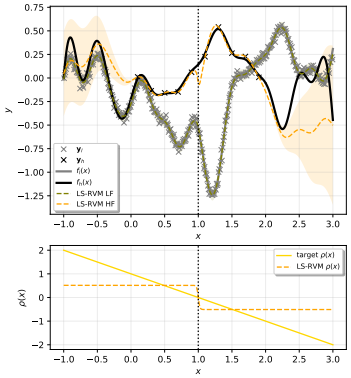
(c) M-MF-GPR



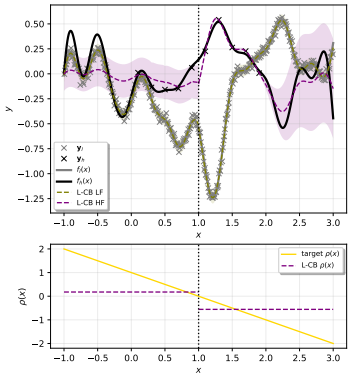
(d) L-RVM



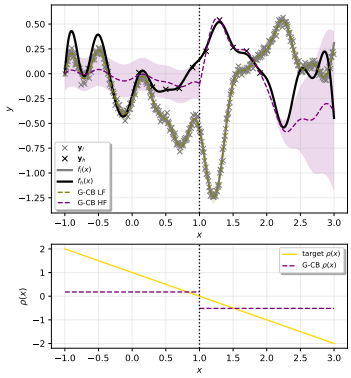
(e) G-RVM



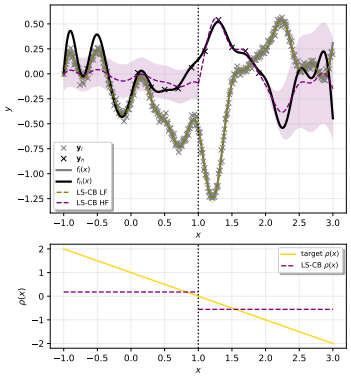
(f) LS-RVM



(g) L-CB



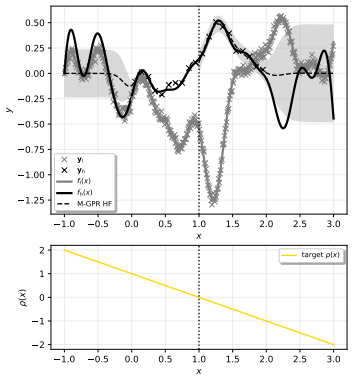
(h) G-CB



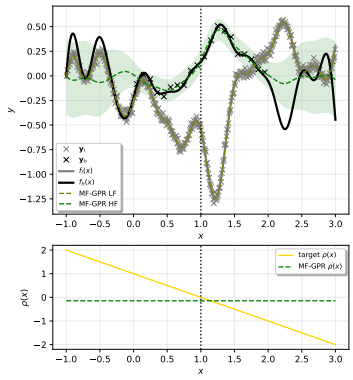
(i) LS-CB

Figure A.46: model predictions with 5 high-fids per region of function **a** and data-set 0 of figure A.45.

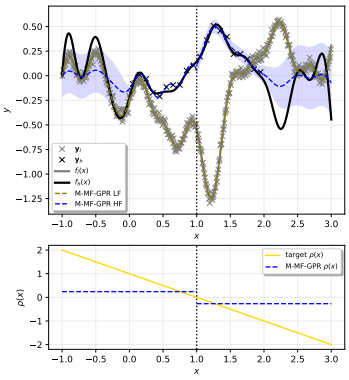
Appendix A: Experimental Results



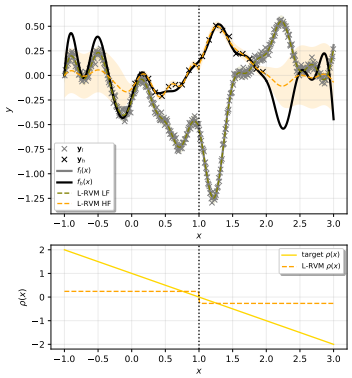
(a) M-GPR



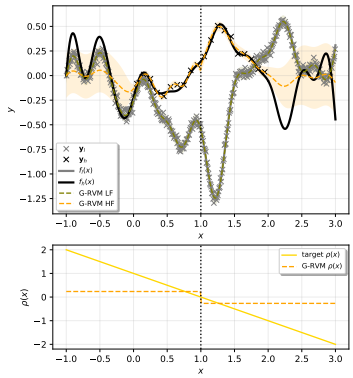
(b) MF-GPR



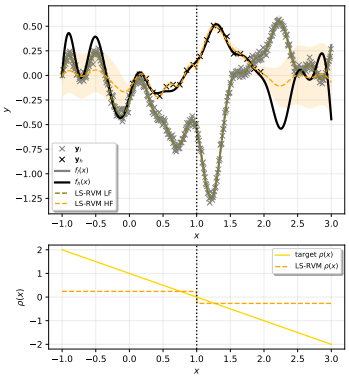
(c) M-MF-GPR



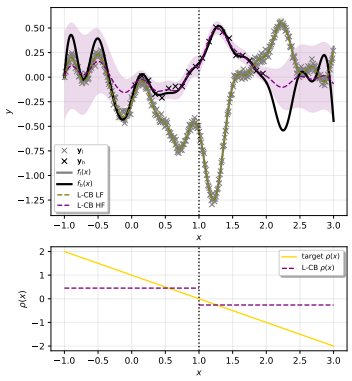
(d) L-RVM



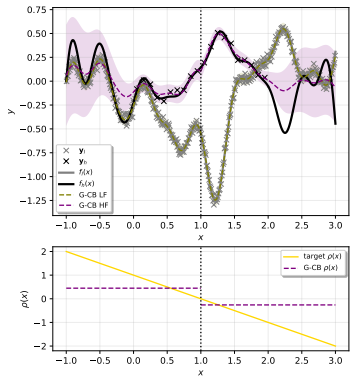
(e) G-RVM



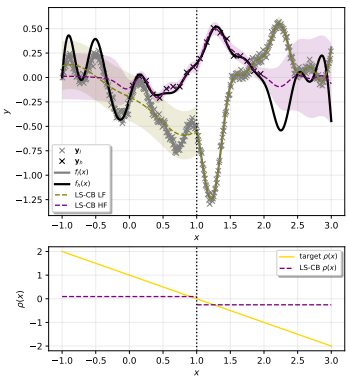
(f) LS-RVM



(g) L-CB



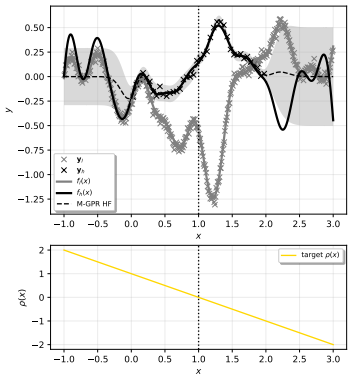
(h) G-CB



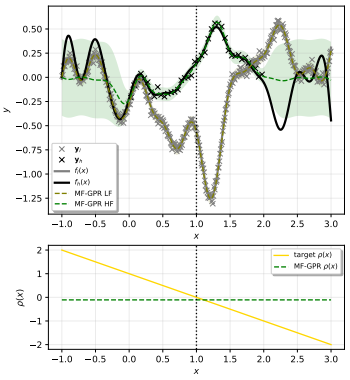
(i) LS-CB

Figure A.47: model predictions with 10 high-fids per region of function **a** and data-set 0 of figure A.45.

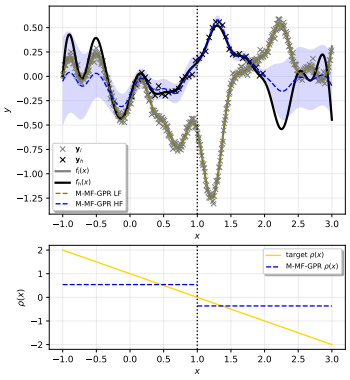
Appendix A: Experimental Results



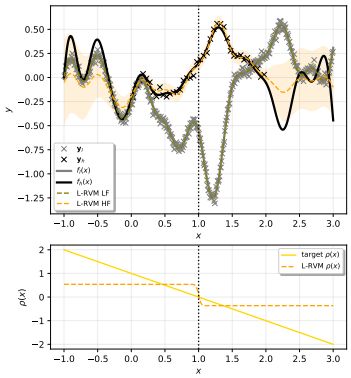
(a) M-GPR



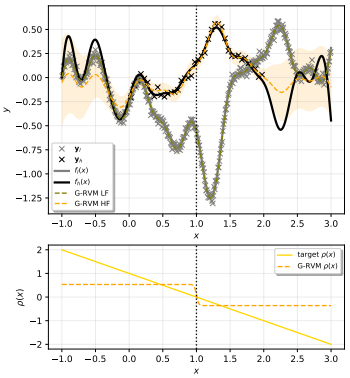
(b) MF-GPR



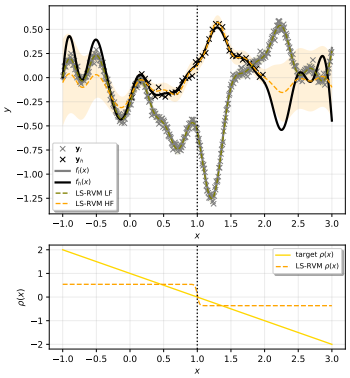
(c) M-MF-GPR



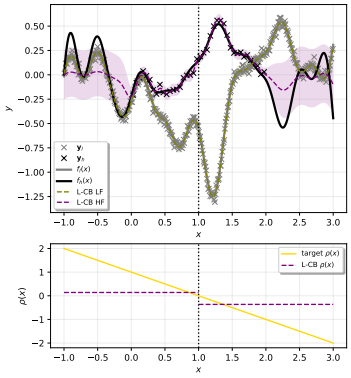
(d) L-RVM



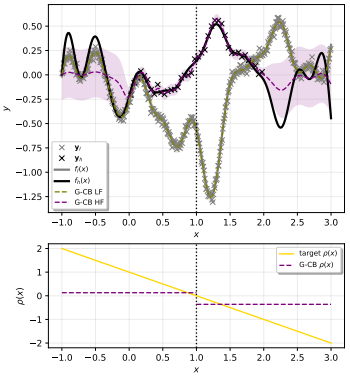
(e) G-RVM



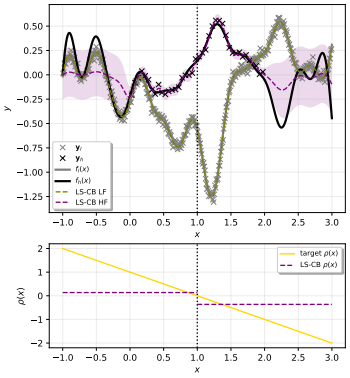
(f) LS-RVM



(g) L-CB



(h) G-CB



(i) LS-CB

Figure A.48: model predictions with 20 high-fids per region of function **a** and data-set 0 of figure A.45.

Bibliography

[1] Loïc Brevault, Mathieu Balesdent, and Ali Hebbal. "Overview of Gaussian process based multi-fidelity techniques with variable relationship between fidelities, application to aerospace systems". In: *Aerospace Science and Technology* 107 (2020), p. 106339.

[2] Roberto Calandra et al. "Manifold Gaussian processes for regression". In: *2016 International Joint Conference on Neural Networks (IJCNN)*. IEEE. 2016, pp. 3338–3345.

[3] Yanshuai Cao and David J Fleet. "Generalized product of experts for automatic and principled fusion of Gaussian process predictions". In: *arXiv preprint arXiv:1410.7827* (2014).

[4] Kurt Cutajar et al. "Deep gaussian processes for multi-fidelity modeling". In: *arXiv preprint arXiv:1903.07320* (2019).

[5] Roger Daley. *Atmospheric data analysis*. 2. Cambridge university press, 1993.

[6] Andreas Damianou and Neil D Lawrence. "Deep gaussian processes". In: *Artificial intelligence and statistics*. PMLR. 2013, pp. 207–215.

[7] Marc Deisenroth and Jun Wei Ng. "Distributed gaussian processes". In: *International Conference on Machine Learning*. PMLR. 2015, pp. 1481–1490.

[8] David Duvenaud. "Automatic model construction with Gaussian processes". PhD thesis. University of Cambridge, 2014.

[9] Alexander IJ Forrester, András Sóbester, and Andy J Keane. "Multi-fidelity optimization via surrogate modelling". In: *Proceedings of the royal society a: mathematical, physical and engineering sciences* 463.2088 (2007), pp. 3251–3269.

[10] Marc GD Geers, Varvara G Kouznetsova, and WAM1402 Brekelmans. "Multi-scale computational homogenization: Trends and challenges". In: *Journal of computational and applied mathematics* 234.7 (2010), pp. 2175–2182.

[11] Robert B Gramacy. *Bayesian treed Gaussian process models*. University of California, Santa Cruz, 2005.

[12] Robert B Gramacy and Herbert K H Lee. "Bayesian treed Gaussian process models with an application to computer modeling". In: *Journal of the American Statistical Association* 103.483 (2008), pp. 1119–1130.

[13] Mengwu Guo et al. "Multi-fidelity regression using artificial neural networks: efficient approximation of parameter-dependent output quantities". In: *Computer methods in applied mechanics and engineering* 389 (2022), p. 114378.

[14] Trevor Hastie et al. *The elements of statistical learning: data mining, inference, and prediction*. Vol. 2. Springer, 2009.

[15] Geoffrey E Hinton. "Training products of experts by minimizing contrastive divergence". In: *Neural computation* 14.8 (2002), pp. 1771–1800.

[16] Andre G Journel and Charles J Huijbregts. "Mining geostatistics". In: (1976).

[17] Marc C Kennedy and Anthony O'Hagan. "Predicting the output from a complex computer code when fast approximations are available". In: *Biometrika* 87.1 (2000), pp. 1–13.

[18] Hyoung-Moon Kim, Bani K Mallick, and Chris C Holmes. "Analyzing nonstationary spatial data using piecewise Gaussian processes". In: *Journal of the American Statistical Association* 100.470 (2005), pp. 653–668.

[19] Varvara Kouznetsova, WAM Brekelmans, and FPT1005 Baaijens. "An approach to micro-macro modeling of heterogeneous materials". In: *Computational mechanics* 27.1 (2001), pp. 37–48.

[20] Yuichi Kuya et al. “Multifidelity surrogate modeling of experimental and computational aerodynamic data sets”. In: *AIAA journal* 49.2 (2011), pp. 289–298.

[21] Loic Le Gratiet. “Multi-fidelity Gaussian process regression for computer experiments”. PhD thesis. Université Paris-Diderot-Paris VII, 2013.

[22] Loic Le Gratiet and Josselin Garnier. “Recursive co-kriging model for design of computer experiments with multiple levels of fidelity”. In: *International Journal for Uncertainty Quantification* 4.5 (2014).

[23] Haitao Liu et al. “Generalized robust Bayesian committee machine for large-scale Gaussian process regression”. In: *International Conference on Machine Learning*. PMLR. 2018, pp. 3131–3140.

[24] Haitao Liu et al. “Understanding and comparing scalable Gaussian process regression for big data”. In: *Knowledge-Based Systems* 164 (2019), pp. 324–335.

[25] Haitao Liu et al. “When Gaussian process meets big data: A review of scalable GPs”. In: *IEEE transactions on neural networks and learning systems* 31.11 (2020), pp. 4405–4423.

[26] Saeed Masoudnia and Reza Ebrahimpour. “Mixture of experts: a literature survey”. In: *Artificial Intelligence Review* 42.2 (2014), pp. 275–293.

[27] Georges Matheron. “The intrinsic random functions and their applications”. In: *Advances in applied probability* 5.3 (1973), pp. 439–468.

[28] Christian Miehe, Jan Schotte, and Jörg Schröder. “Computational micro–macro transitions and overall moduli in the analysis of polycrystals at large strains”. In: *Computational Materials Science* 16.1-4 (1999), pp. 372–382.

[29] Duy Nguyen-Tuong, Jan Peters, and Matthias Seeger. “Local gaussian process regression for real time online model learning and control”. In: *Proceedings of the 21st International Conference on Neural Information Processing Systems*. 2008, pp. 1193–1200.

[30] Chiwoo Park and Daniel Apley. “Patchwork kriging for large-scale gaussian process regression”. In: *The Journal of Machine Learning Research* 19.1 (2018), pp. 269–311.

[31] Chiwoo Park and Jianhua Huang. “Efficient computation of Gaussian process regression for large spatial data sets by patching local Gaussian processes”. In: *Journal of Machine Learning Research* 17 (Oct. 2016), pp. 1–29.

[32] Chiwoo Park, Jianhua Z Huang, and Yu Ding. “Domain decomposition approach for fast Gaussian process regression of large spatial data sets”. In: *1foldr Import 2019-10-08 Batch 12* (2011).

[33] Andrew Pensoneault, Xiu Yang, and Xueyu Zhu. “Nonnegativity-enforced Gaussian process regression”. In: *Theoretical and Applied Mechanics Letters* 10.3 (2020), pp. 182–187.

[34] Paris Perdikaris and George Em Karniadakis. “Model inversion via multi-fidelity Bayesian optimization: a new paradigm for parameter estimation in haemodynamics, and beyond”. In: *Journal of The Royal Society Interface* 13.118 (2016), p. 20151107.

[35] Paris Perdikaris et al. “Nonlinear information fusion algorithms for data-efficient multi-fidelity modelling”. In: *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 473.2198 (2017), p. 20160751.

[36] Sachhidan Prashanth et al. “Fiber reinforced composites-a review”. In: *J. Mater. Sci. Eng* 6.03 (2017), pp. 2–6.

[37] Maziar Raissi and George Karniadakis. “Deep multi-fidelity Gaussian processes”. In: *arXiv preprint arXiv:1604.07484* (2016).

[38] Carl Rasmussen and Zoubin Ghahramani. “Infinite mixtures of Gaussian process experts”. In: *Advances in neural information processing systems* 14 (2001).

[39] Junuthula Narasimha Reddy. *Mechanics of laminated composite plates and shells: theory and analysis*. CRC press, 2003.

[40] IBCM Rocha. “Numerical and Experimental Investigation of Hygrothermal Aging in Laminated Composites”. In: (2019).

[41] IBCM Rocha, Pierre Kerfriden, and FP van der Meer. "On-the-fly construction of surrogate constitutive models for concurrent multiscale mechanical analysis through probabilistic machine learning". In: *Journal of Computational Physics: X* 9 (2021), p. 100083.

[42] Markus P Rumpfkeil, Kyohei Hanazaki, and Philip S Beran. "Construction of Multi-Fidelity Locally Optimized Surrogate Models for Uncertainty Quantification". In: *19th AIAA Non-Deterministic Approaches Conference*. 2017, p. 1948.

[43] Markus Peer Rumpfkeil and Philip Beran. "Construction of multi-fidelity surrogate models for aerodynamic databases". In: *Proceedings of the Ninth International Conference on Computational Fluid Dynamics, ICCFD9, Istanbul, Turkey*. 2016.

[44] Laura P Swiler et al. "A survey of constrained Gaussian process regression: Approaches and implementation challenges". In: *Journal of Machine Learning for Modeling and Computing* 1.2 (2020).

[45] O.T. Taylan. "Surrogate Constitutive Models with Multi-fidelity Gaussian Processes for Composite Micromodels". MA thesis. Delft University of Technology, Aug. 2020.

[46] Nick Terry and Youngjun Choe. "Splitting Gaussian Process Regression for Streaming Data". In: *arXiv preprint arXiv:2010.02424* (2020).

[47] Philip Duncan Thompson. "Optimum Smoothing of Two-Dimensional Fields 1". In: *Tellus* 8.3 (1956), pp. 384–393.

[48] Volker Tresp. "A Bayesian committee machine". In: *Neural computation* 12.11 (2000), pp. 2719–2741.

[49] Volker Tresp. "Mixtures of Gaussian processes". In: *Advances in neural information processing systems* (2001), pp. 654–660.

[50] Sethu Vijayakumar, Aaron D'souza, and Stefan Schaal. "Incremental online learning in high dimensions". In: *Neural computation* 17.12 (2005), pp. 2602–2634.

[51] Henry Wilde, Vincent Knight, and Jonathan Gillard. "Evolutionary dataset optimisation: learning algorithm quality through evolution". In: *Applied Intelligence* 50.4 (2020), pp. 1172–1191.

[52] Christopher K Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*. Vol. 2. 3. MIT press Cambridge, MA, 2006.

[53] Seniha Esen Yuksel, Joseph N Wilson, and Paul D Gader. "Twenty years of mixture of experts". In: *IEEE transactions on neural networks and learning systems* 23.8 (2012), pp. 1177–1193.