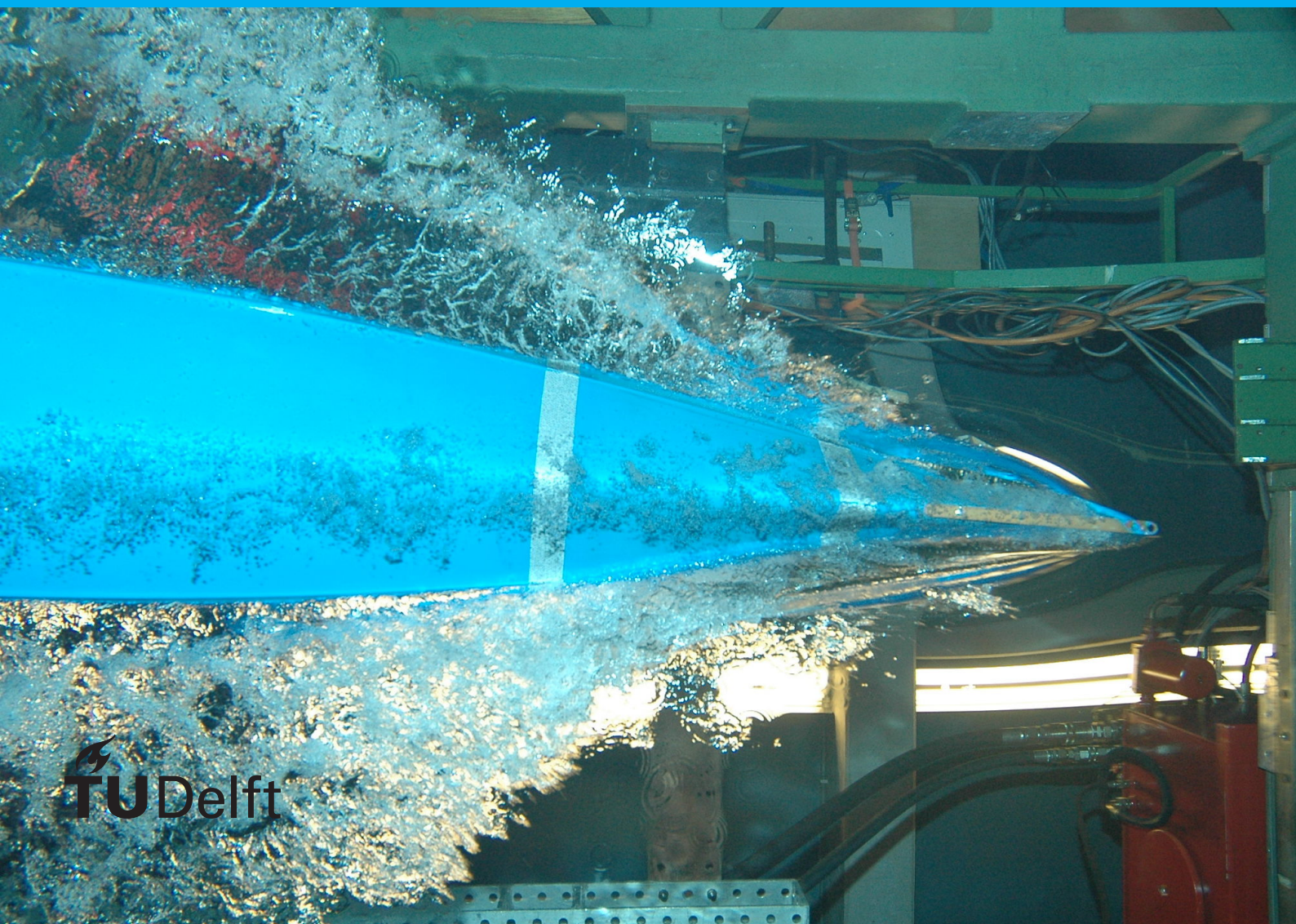
Stitching Gaussian Processes with Multi-fidelity

Optional subtitle

H. Hendriks

Cover Text
possibly
spanning multiple lines

ISBN 000-00-0000-000-0



Stitching Gaussian Processes with Multi-fidelity

Optional subtitle

by

H. Hendriks

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Tuesday January 1, 2013 at 10:00 AM.

Student number:	1234567
Project duration:	February, 2021 – January, 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

Preface

Preface...

H. Hendriks
Delft, January 2013

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 [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 [53]. The model provides uncertainty information regarding its predictions due to its probabilistic nature. This is useful in detecting the extrapolation regime which does not give consistently good results [45]. To overcome this issue the interpolation regime can be extended by adding more observations. However, this results in a higher computational cost of model inference and optimization, and data collection. The latter 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 assumed to be linear and defined using a constant correlation coefficient [17, 21, 45]. These models reduce the number of needed high-fidelity observations [17, 20], and they perform better in the high-fidelity extrapolation regime [17]. Several studies use these methods 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, the prediction of a loading case on a micro-mechanical model is more accurate when a model with linear correlation is trained on that specific loading case instead of multiple because the underlying correlation of each differs [45].

Different non-linear assumptions are studied in the literature, where the correlation is often defined as a function of the observational input and, sometimes, the observations at lower fidelities [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. The thesis investigates two methods that remove them by stitching the local models: adding continuity constraints and taking a weighted sum of the local predictions.

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

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

1.3. Scope

The thesis answers the research questions by investigating these methods on synthetic 1-dimensional data sets. These are created such that the distinguishing factor between the results of the methods is their correlation assumption. This reduces the interference of other factors, such as their ability to capture dimensionality or certain stationary features, resulting in a more deliberate and precise investigation. Hence, this thesis provides a first account of the splitting and stitching of Gaussian process regression models with multi-fidelity in settings with non-linear correlations.

1.4. Outline

The thesis is organized into the following chapters:

- Chapter 2 identifies and discusses the literature regarding the Gaussian process regression models with non-linear correlated multi-fidelity, and gives an account of the current state of the art on splitting and stitching Gaussian process regression models. The current problems and limitations regarding these models are discussed, and how this thesis contributes to the literature.
- 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.
- Chapter 5 presents and discusses the results of the investigations 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., $\mathbf{I}$.

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, and discusses its extensions that can handle non-linearly correlated multi-fidelity datasets, and states what this thesis contributes to the field.

2.1. Local Approximations with GPR

Gaussian process regression (GPR) is a machine learning method based on Bayesian inference that uses Gaussian processes, which are a generalization of Gaussian distributions to the functional space [53]. The method incorporates prior information through the Bayesian formalism, meaning that it expresses a prior belief about the underlying relation of the observations, and it provides an uncertainty measure on its predictions [53]. 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 [53]. 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 $\mathcal{O}(N^3)$ where N is the number of observations. This makes it highly inefficient for large-scale datasets [24, 25].

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. 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 and their predictions are discontinuous at the partition's boundaries [24, 25].

One way to alleviate the discontinuity problem is by constraining equality at the boundaries between the partitions, this is exactly what Park and collaborators aimed to do with their three papers [32, 31, 30]. 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 [31]. This resulted in a more numerically stable method that also allows for continuity to be imposed across the whole boundary. In their third paper [30], they ditched the equivalent optimization approach and opted to put the constraints as pseudo-observations into the standard GPR framework. 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 compared to the previous two [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]. Further development of constraining the local approximation methods might come from using some methods mentioned in the survey on constrained Gaussian process regression [44].

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]. This method produces overconfident predictive variance when increasing the number of local models [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 constraints on the weights [3, 7].

The performance of PoE is improved by imposing a conditional independence assumption, creating a method called the Bayesian committee machine (BCM) [49]. 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 [54, 26], which takes a weighted sum of the probability distributions of the local models instead of using multiplication like the PoE methods [25]. It is used for data that has non-stationary features, but it comes with the cost of having an intractable inference [24]. In the original description, the gating functions are parametric, therefore Tresp extended it to the non-parametric case where the mean, the noise variance, and the weights are modeled by GPs [50]. However, this results in an exceedingly high computational cost which is unattractive for large datasets [25]. Two possible solutions are the localization of the experts, for example, the infinite mixture of GP experts (iMGPRE) [38], and the usage of global approximations for the GPs [25]. Both of them fall under the mixture of implicitly localized experts (MILE) which means that the data is dynamically assigned to the local models which take their performance into account instead of being predetermined. The downside to this approach is that it sometimes results in zero-coefficients that rule out a local model [25]. Another approach is to use the mixture of explicitly localized experts (MELE) where the assignment of data to the local models is predetermined, however, this misses the interaction between the local models [25].

The authors of [29] created a method in a similar fashion to 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]. The authors of [46] created a similar model that uses all local models in its weighing procedure.

2.2. 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 which is proportional to $\mathcal{O}((\sum_i N_i)^3)$. Gratiet [22] decreases the time complexity to $\mathcal{O}(\sum_i N_i^3)$ 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 better handle non-linear correlation between fidelities is the Deep Multi-fidelity GPR model [37]. This is an extension of the AR(1) model by Kennedy et. al. [17] and it generalizes the ManifoldGP model [2] in which a transformation maps the input space to a manifold. The transformed observations are used as input for a standard Gaussian process regression model. The transformation is defined as a multi-layered neural network that is jointly optimized

with the hyperparameters of the 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, specifically computational fluid dynamics for them, 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.

2.3. 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.

The idea of combining local approximation methods with multi-fidelity GPR is not novel. Rumpfkeil et. al. [43, 42] enhanced Gratiet's recursive multi-fidelity GPR formulation [22] with an explicit mixture-of-experts to overcome the limiting inference capabilities of the single global model when presented with non-stationary datasets. This thesis differs in that it specifically investigates the effect of using local approximation methods on the global correlation between low- and high-fidelity.

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 literature review describes a clear gap in the knowledge of inferring non-linear correlation between successive fidelities by local approximation methods. The thesis investigates this by performing experiments with multi-fidelity GPR models where one or more fidelities are modeled with a local approximation method.

The following sections explain the mathematical details of these models. It starts by stating the mathematical background, and the model selection and prediction process of GPR following the explanation and notation in Rasmussen & Williams [53]. 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; the thesis calls this the local modeling approach.

3.1. Gaussian Process Regression

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.

A GP is defined as a collection of random variables, any finite number of which have a joint Gaussian distribution [53]. 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 [53].

When the prior GP is placed on the latent function $f(x)$, then this is written as $f(\mathbf{x}) \sim \mathcal{GP}\left(m(\mathbf{x}), k(\mathbf{x},\mathbf{x}')\right)$. 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 [53]. The interested reader is referred to section 2.7 of Rasmussen & Williams [53] 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 [53]; these are referred to as hyperparameters of the GPR model. The covariance function can be freely defined as long as it is positive semidefinite [53]. For a more detailed overview of covariance functions refer to chapter 4 of Rasmussen & Williams [53] 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 $p\left(\mathbf{f}_*|\mathbf{X},\mathbf{y},\mathbf{X}_*\right)$. 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

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

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

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

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 $\mathrm{cov}\left[\mathbf{f}_*\right]$ [53].

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 and that they deviate more when further away. 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.

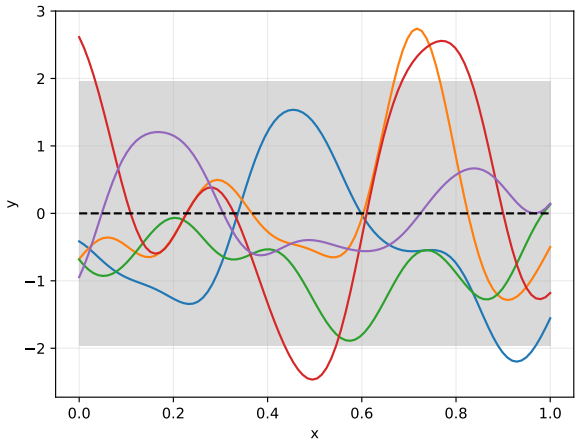


Figure 3.1: 5 Samples from a prior with zero mean

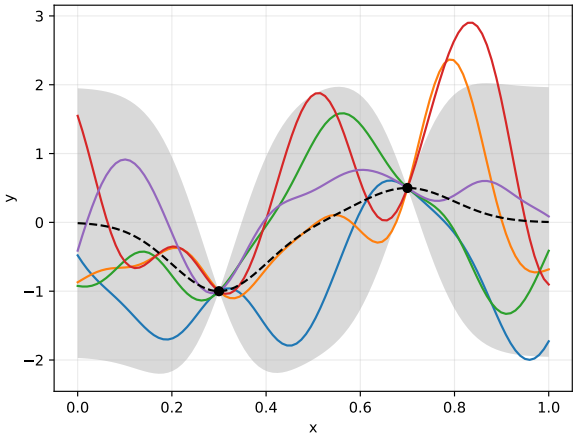


Figure 3.2: 5 Samples from a posterior conditioned on 2 observations

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 [53], 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 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 result

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

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.

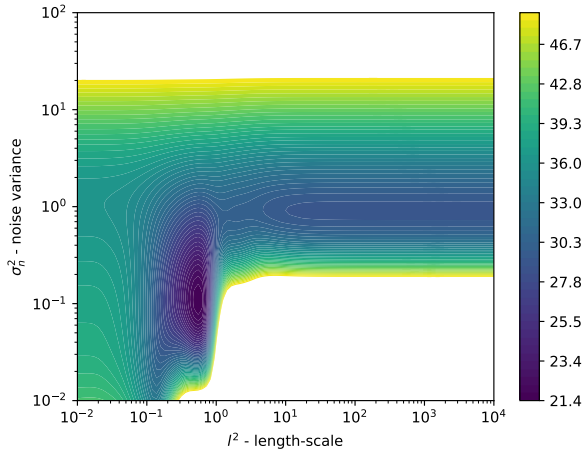


Figure 3.3: GPR Negative Log Marginal Likelihood

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 that in this thesis the respective disjoint regions of the locally disjoint models is given and therefore not optimized.

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 multiple independent GPR models each responsible for its own region where the mean and the variance of the models are constrained to be

equal at their respective boundaries. The constraint is enforced in the optimization of the hyperparameters. Figure 3.4 shows an example of a prediction of a CB-GPR model.

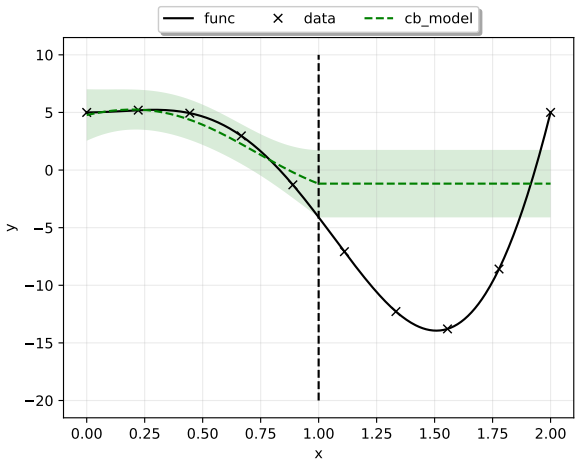


Figure 3.4: Example CB-GPR Method

The model is built on the idea presented in [33] where a GPR model is constrained to be non-negative in the optimization of the hyperparameters. This approach enforces the constraints in a similar manner but changes them to enforce the predictive mean and variance of two neighboring local models to be equal at specific points on the boundary.

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 due to the added constraints. The CB-GPR method constrains the predictive mean and predictive variance of function values $\mathbf{f}_{k_*}$ and $\mathbf{f}_{l_*}$ from partition k and l, respectively, to be equal at prediction inputs $\mathbf{x}_*$ that lie on the boundary between partition k and l, denoted as Γ_{kl}. It is difficult to constrain the predictive function values of two neighboring partitions in two dimensions or higher because their boundary consists of an infinite number of points. Therefore, the constraints are only satisfied at k_{kl} boundary points $\mathbf{x}_{kl_*} \in \Gamma_{kl}$, as this makes the method easier to derive and less computationally expensive. The two constraints with predictive function values $f_{k_*}(\mathbf{x}_{kl_*})$ and $f_{l_*}(\mathbf{x}_{kl_*})$ at boundary point $\mathbf{x}_{kl_*}$ 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

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

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

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 $\epsilon_E = [0.01, 0.1, 1.0]$, 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.

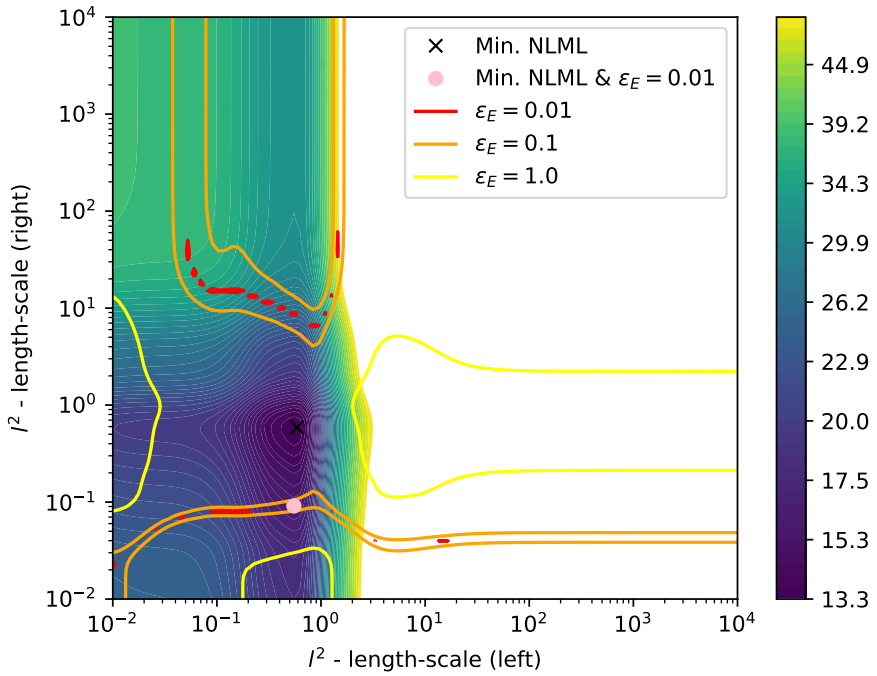


Figure 3.5: CB-GPR - NLML

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. [51] 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

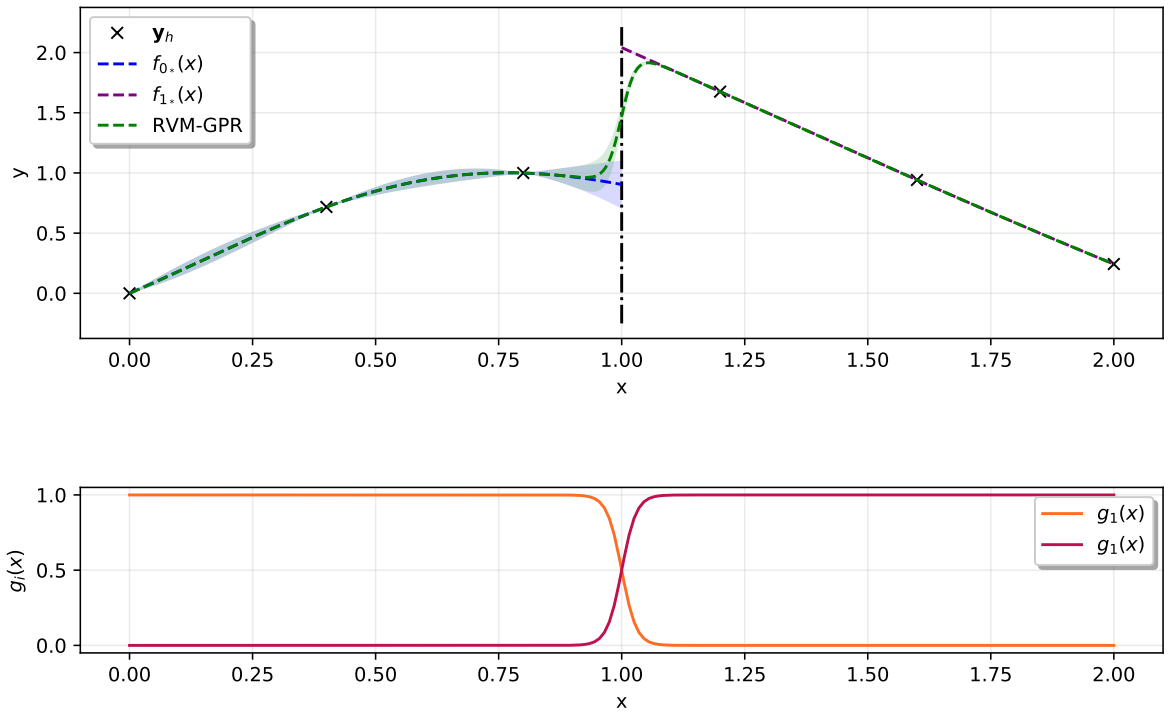


Figure 3.6: A prediction example of an RVM-GPR with two local models, in regions $[0,1]$ and $[1,2]$, (hyperparameters are handpicked) for $f(x) = sin(2x) + \mathcal{H}(x-1)$ and 6 observations linearly spaced across region $[0,2]$.

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

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

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:

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

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

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 [29], the weight functions $g_k(\mathbf{x})$ are defined as the normalized distance measure of distance measures $w_k(\mathbf{x})$

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

The thesis only explores cases with 1-dimensional inputs and two local models, therefore the two distance measures $w_k(\mathbf{x})$ are defined such that $g_k(\mathbf{x})$ are sigmoid functions which means that

$$w_1(\mathbf{x}) = \exp\left(-l(\mathbf{x}-\mathbf{c})\right) \quad \text{and} \tag{3.19}$$
$$w_2(\mathbf{x}) = \exp\left(l(\mathbf{x}-\mathbf{c})\right) \tag{3.20}$$

where l is the length-scale of the weights that determines the size of the transition zone between the two local models and $\mathbf{c}$ 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 5 and one with 50, where the boundary is defined at $\mathbf{c} = 0.5$. This shows that increasing the length-scale sharpens the transition between the weighted functions.

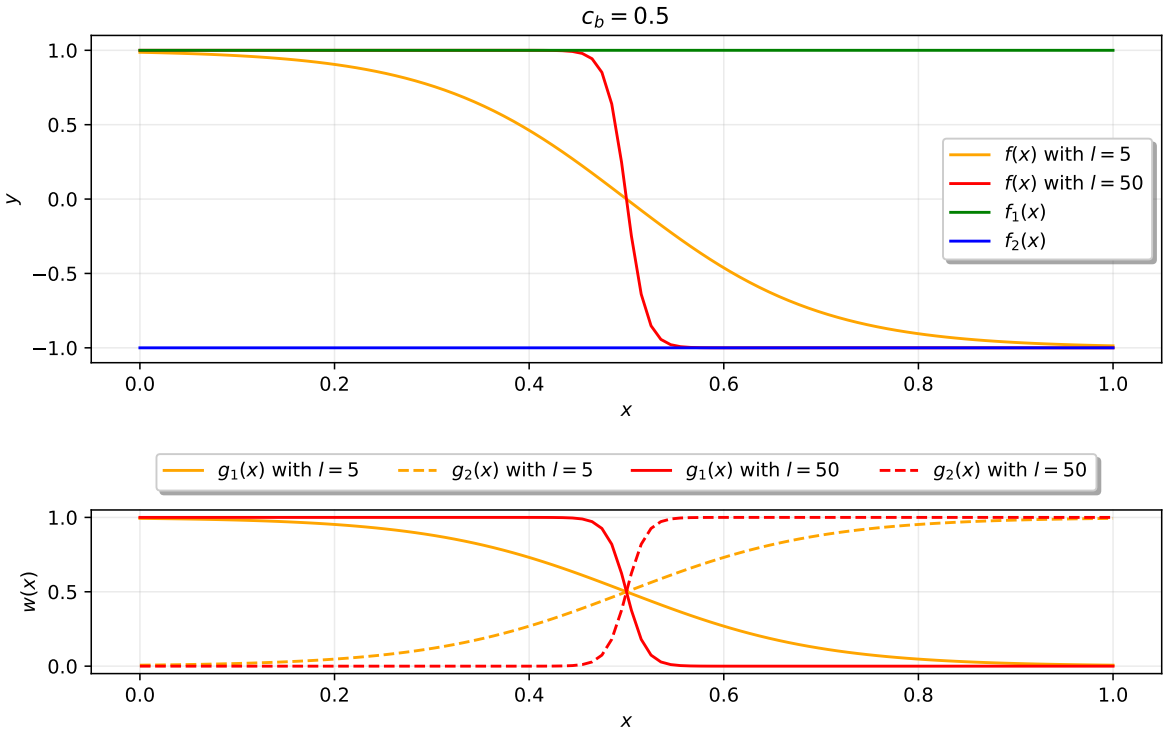


Figure 3.7: RVM-GPR - Length-scale

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]

[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.

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

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. 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

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

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 $\text{cov}\left[f(\mathbf{x}_p), f(\mathbf{x}_q)\right]$ 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

Multi-fidelity methods use multiple fidelities that each represents a different model complexity to create more accurate models. This concept can also be applied to GPR, where multiple GPR models are trained at different fidelities of training data. For example, one accurate and slow model, and one inaccurate and fast model. These GPR models are related to each other and so a model is created that is able to use both types of training data.

The GPR with multi-fidelity is formulated using the recursive approach as detailed in L. Le Gratiet [21]. The thesis uses this approach because it reduces the computational complexity by making it possible to recursively predict prediction inputs and recursively optimize the hyperparameters from the first to the last fidelity.

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 [45] as

$$y_t = f_t(\mathbf{x}_t) + \epsilon_t \quad \text{and} \tag{3.23}$$

$$y_t = \underbrace{\rho_{t-1} f_{t-1}(\mathbf{x}_t) + \delta_t(\mathbf{x}_t)}_{f_t(\mathbf{x}_t)} + \epsilon_t = f_t(\mathbf{x}_t) + \epsilon_t \tag{3.24}$$

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 $\mathcal{GP}\left(m_t(\mathbf{x}_t), k_t(\mathbf{x}_t, \mathbf{x}_t')\right)$. In most cases, the mean function $m_t(\mathbf{x}_t)$ is assumed to be the zero mean function and the covariance functions $k_t(\mathbf{x}_t, \mathbf{x}_t')$ 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 determined 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

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

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

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

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 $\mathrm{cov}\left[\mathbf{f}_{t_*}\right]$.

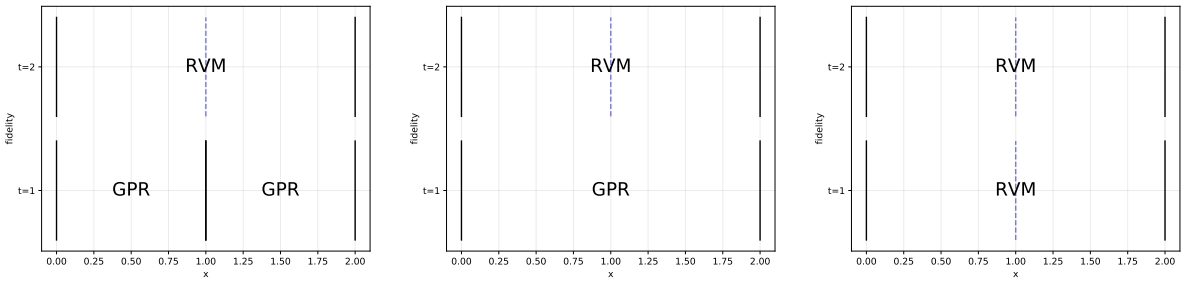
3.4. Stitching with Multi-fidelity

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.

This thesis is interested in the investigation of the correlation between fidelities, and therefore two different extensions are also considered that model each lower fidelity as either one global GPR model 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 two of the three research questions in this thesis: the effect of splitting and stitching of GPR with multi-fidelity on the performance in the non-linear correlated setting in the interpolation and extrapolation regime, and the effect of the number of low- and high-fidelity observations on the performance. The computational cost associated with the splitting and stitching methods is already shown and discussed in the previous chapter.

These two questions are answered by investigating the methods, as described in the previous chapter, on 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.

Different numbers of low- and high-fidelity observations are considered per case and two different sampling strategies for the inputs are explored. This is to answer the second research question on how these influence the prediction accuracy of splitting and stitching.

4.1. Thesis' 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 (from now on the abbreviations are used to denote the methods):

- 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 GPR;
- G-RVM: Global Random Variable Mixture of Experts GPR;
- LS-RVM: Locally Stitched Random Variable Mixture of Experts GPR;
- L-CB: Local Constrained Boundary GPR;
- G-CB: Global Constrained Boundary GPR;
- LS-CB: Locally Stitched Constrained Boundary GPR;

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.

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

4.3. 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 written in table 4.1. 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

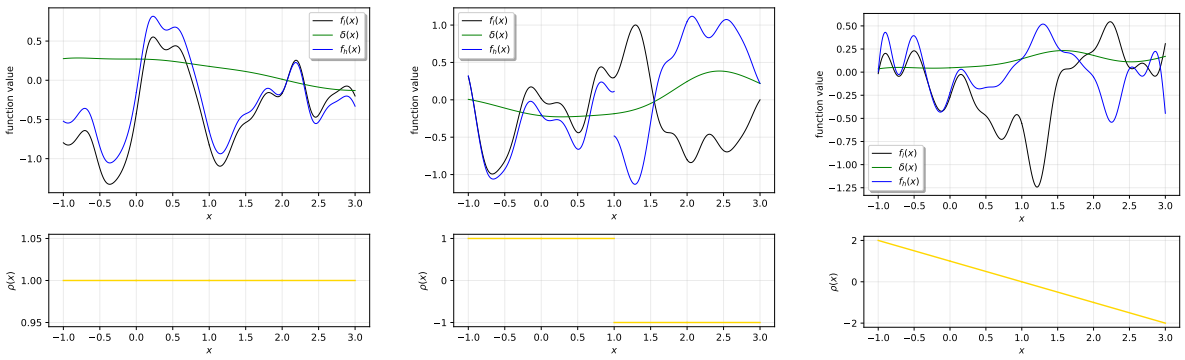
Table 4.1: Hyper Parameter values of sampled MultiGP

The correlation function $\rho(x)$ of each case is defined in table 4.2, where $\mathcal{H}(x)$ 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. For demonstrative purposes, figure 4.1 shows one sampled function of each 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.

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

Table 4.2: Case-specific correlation function



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

Figure 4.1: One sampled function from MultiGP per case.

4.4. 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 uniformly distributed in each region. The different numbers of observations are chosen such that the number of low- and high-fidelity observations have no common divisor. This ensures that they do not have the same inputs as the low-fidelity observations are considered redundant if they overlap with the high-fidelity observations.0 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.

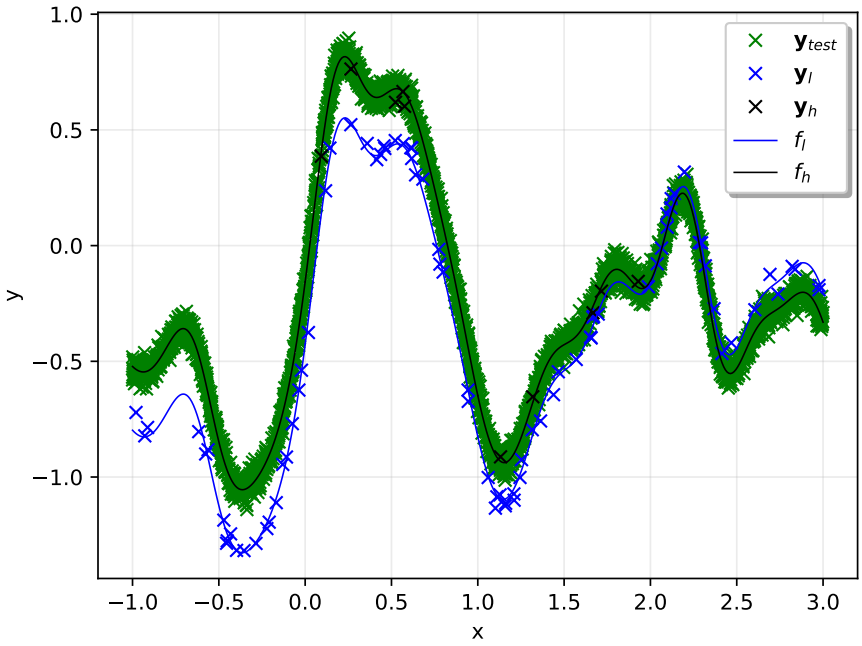


Figure 4.2: Case Constant - sampled function and uniformly distributed training and test data-set.

4.5. Performance

The average performance of a dataset type of a sampled function is defined as the expectation of the test error over its 20 data sets, 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 uniformly distributed (based on the sampling choice of the dataset). In addition, an i.i.d. Gaussian distribution $\epsilon \sim \mathcal{N}(0, 0.001)$ is added to include measurement errors. 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 boxplot similar to the one in Figure 4.3. Each row of this boxplot 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 colors represent the number of high-fidelity observations: purple equals 20, blue equals 10, and green 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.

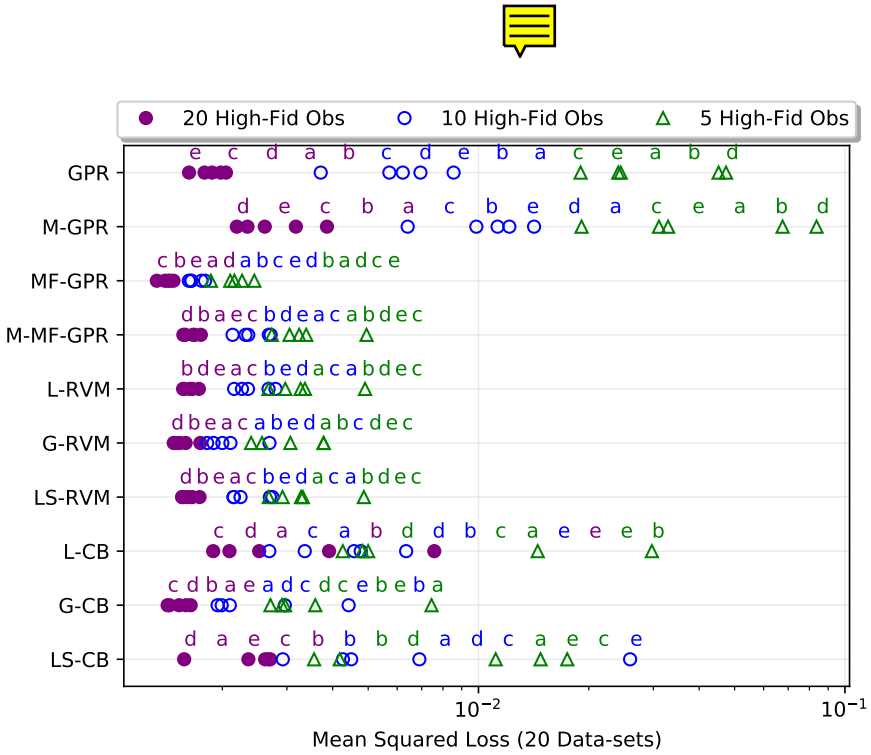


Figure 4.3: Example performance boxplot.

5

Experimental Results

This chapter shows the results of the experiments on the methods and discusses them. First, the difference between single- and multi-fidelity GPR is investigated, thereafter the act of splitting, then the behavior of the RVM-GPR and CB-GPR, and last the stitching methods are compared to MF-GPR and M-MF-GPR.

The case names are renamed in the figures to improve their clarity:

- Constant ρ case → case 0
- Discontinuous ρ case → case 1
- Linearly varying ρ case → case 2

The low- and high-fidelity observations are abbreviated to "low-fids" and "high-fids" in the figures to improve their clarity.

Appendix A shows the results of all experiments.

5.1. Single- or Multi-fidelity

This section compares the GPR and MF-GPR with the three cases. The performance in the interpolation and extrapolation regimes are discussed, as well as the influence of the number of low- and high-fidelity observations. The interpolation and extrapolation regimes are explained with the 20 low-fids and 101 high-fids experiments. The others are discussed in the section on the influence of the number of observations. First, the experiments with uniformly distributed inputs are considered, and after that, a comparison is made against the experiments with linearly spaced inputs.

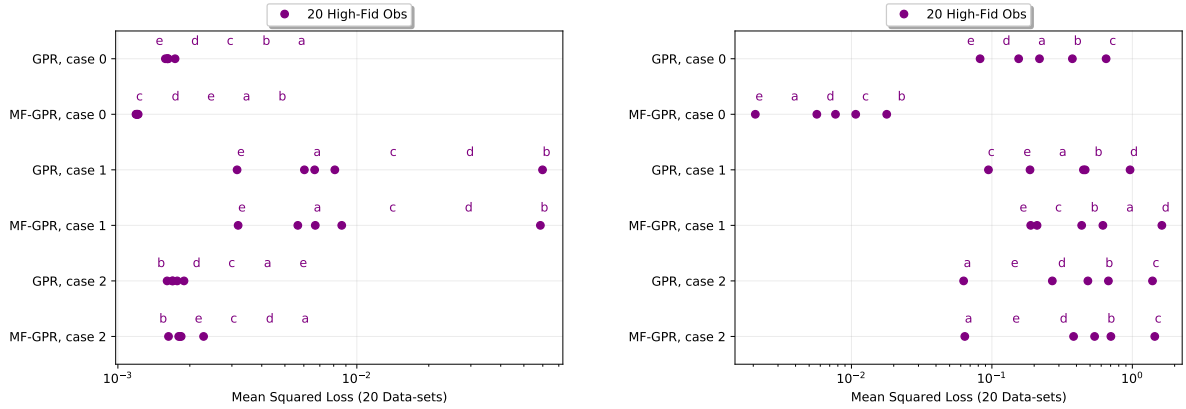
5.1.1. Interpolation Regime

The prediction accuracy of the GPR and MF-GPR is shown in figure 5.1a. The MF-GPR outperforms the GPR in the constant ρ case. This indicates that the low-fidelity observations enhance the MF-GPR's prediction capabilities. However, both methods perform equally well on the discontinuous and linearly varying ρ case when averaged over sampled functions, suggesting that the low-fidelity observations do not add value to non-linear correlated datasets.

The actual predictions across these cases confirm this idea. Figure 5.2 shows one prediction of the GPR and MF-GPR per case. The correlation inference in the constant ρ case is correct. The correlation inference in the discontinuous ρ case is incorrect and is biased towards one side, particularly $[0, 1]$. The correlation inference in the linearly varying ρ case is incorrect as the coefficient is near zero, meaning that the low-fidelity prediction almost plays no role in the inference of the high-fidelity.

Figure 5.3 shows the spread of the correlation coefficient, ρ, of the MF-GPR for each case across the dataset types. The findings of the model plots on the correlation coefficient are reflected back in this

figure: the coefficient is correctly inferred for the constant ρ case, for the discontinuous ρ case a large spread is seen, and for the linearly varying ρ case the coefficient centers around zero. Unexpectedly, the coefficient biases towards the left side of the discontinuity for both the discontinuous and the linearly varying ρ where the bias is larger for the former. These results hold when the number of low- or high-fidelity observations is changed.



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

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

Figure 5.1: performance of GPR and MF-GPR: uniformly distributed, 101 low-fids and 20 high-fids per region.

5.1.2. Extrapolation Regime

The prediction performance of the GPR and MF-GPR in the extrapolation regime across the three cases are shown in figure 5.1b. The MF-GPR outperforms the GPR in the constant ρ case. This shows that capturing the low-fidelity, in the high-fidelity extrapolation regime, enhances the prediction accuracy. The prediction shows that the GPR goes to its zero-mean prior while the MF-GPR follows its low-fidelity model. Therefore, the low-fidelity data enhances the high-fidelity model in regions where its observations are sparse.

However, for the linearly varying ρ case they both perform equally and for the discontinuous ρ case the single-fidelity is better. These trends are explained by looking at the model plots of figure 5.2. The correlation coefficient is zero in the case of the linearly varying ρ case, therefore the low-fidelity adds no value to the high-fidelity prediction which results in almost equal performance. The incorrect inference of the correlation coefficient on one side causes the MF-GPR to be quite incorrect on one side. This causes the error to be bigger than the GPR as it goes to zero in the extrapolation regime.

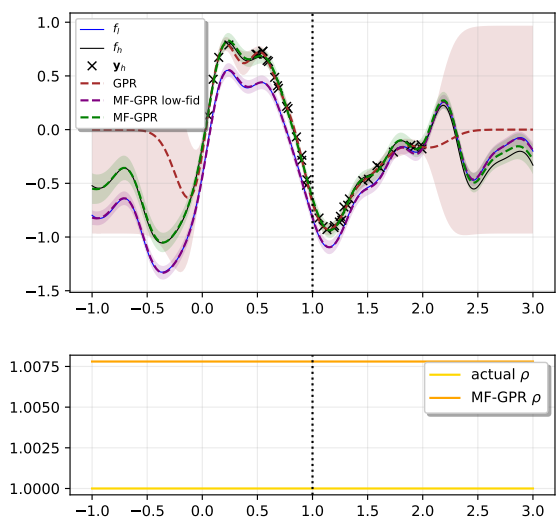
5.1.3. Number of Observations - Interpolation regime

The prediction performance of the GPR and MF-GPR in the interpolation regime across the three cases and with all three high-fidelity options is shown in figures 5.4a and 5.4b. The performance in the interpolation regime increases when the number of high-fidelity observations increases as well. This trend is generally seen in all experiments with all methods. Therefore, they are assumed to be always true unless stated otherwise.

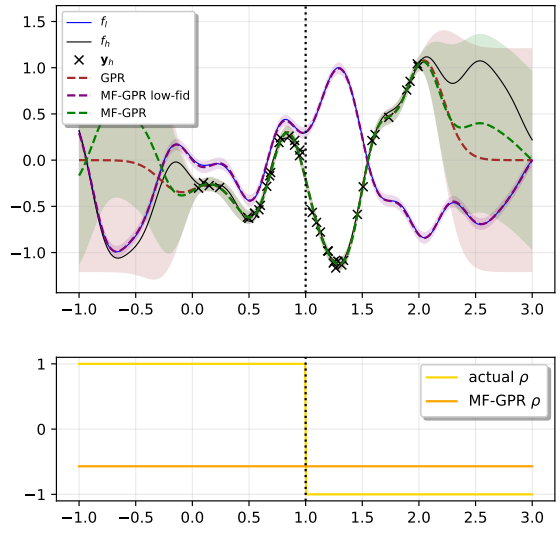
The spread in performance between the GPR and MF-GPR reduces in the constant ρ case when the number of high-fidelity observations increases. This difference is not present in the other cases. This suggests that the underlying high-fidelity function is able to be completely inferred by only the high-fidelity observations, as the results of the GPR converge to the MF-GPRs. Thus, when a large number of high-fidelity observations are present, the addition of low-fidelity observations becomes increasingly unnecessary.

5.1.4. Number of Observations - Extrapolation Regime

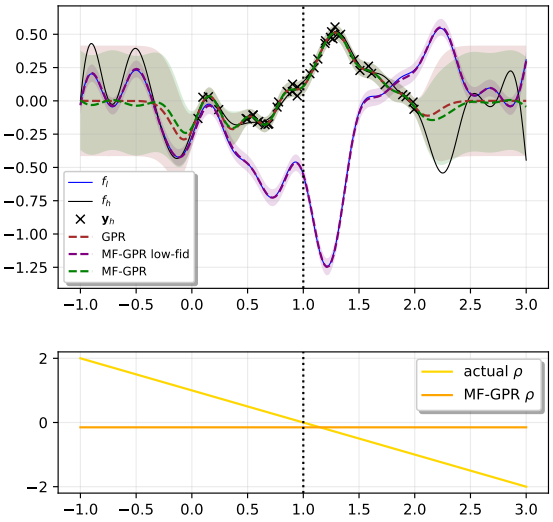
The prediction performance of the GPR and MF-GPR in the extrapolation regime across the three cases and with all three high-fidelity options is shown in figures 5.4c and 5.4d. The prediction performance



(a) constant ρ case.



(b) discontinuous ρ case.



(c) linearly varying ρ case.

Figure 5.2: model predictions with GPR and MF-GPR of function **a** and data-set 0 of each case.

in the extrapolation regime is not highly correlated to the number of low- or high-fidelity observations. There seems to be a slight bias towards more observations but this is almost negligible. The specific sampled function is a better indicator of the performance in the extrapolation regime which is shown in the figure by the grouping of the same sampled function. This trend is generally seen among the experiments for all methods. Therefore, this trend is assumed to be true for all unless stated otherwise.

5.1.5. Input Sampling

The prediction performance of the GPR and MF-GPR in the interpolation and extrapolation regime across the three cases and with all three high-fidelity options for linearly spaced inputs is shown in figure 5.6. The performance of the experiments with the linearly spaced inputs is higher and the spread across the functions is lower compared to the experiments with uniformly distributed inputs. This is due to a more consistent coverage of the input space. Figure 5.7 shows clearly that due to the uniformly distributed inputs three observations ended up close to each other, region $[0.5, 0.6]$, and no observations are in the region $[0.6, 1.1]$. This severely limits the overall prediction accuracy compared to the linearly spaced inputs. This trend is generally seen in all experiments with all methods. Therefore,

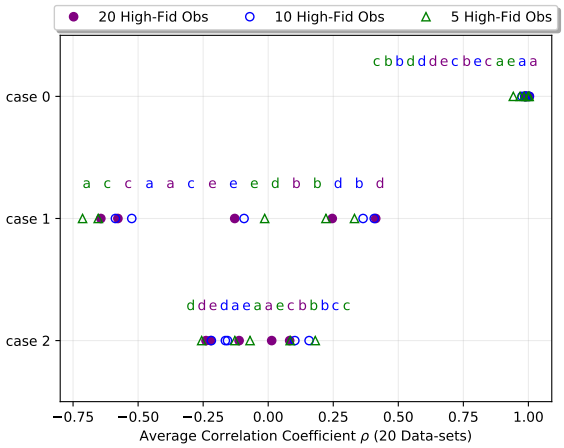
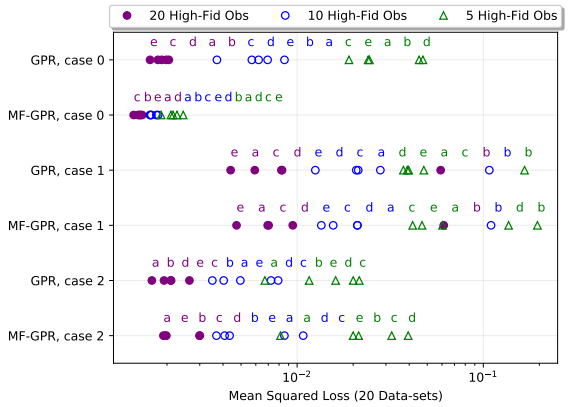


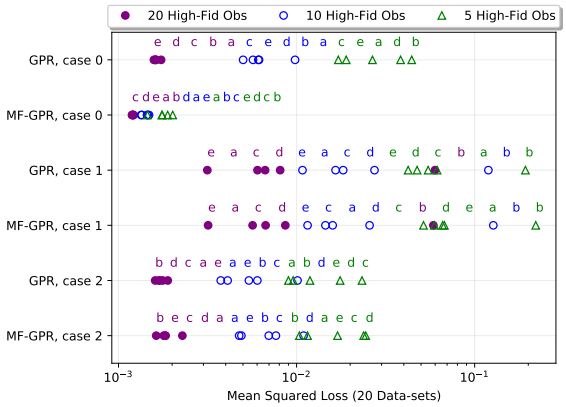
Figure 5.3: Correlation coefficient, ρ, of MF-GPR: uniformly distributed, 101 low-fids and 20 high-fids per region.

they are assumed to be always true unless stated otherwise.

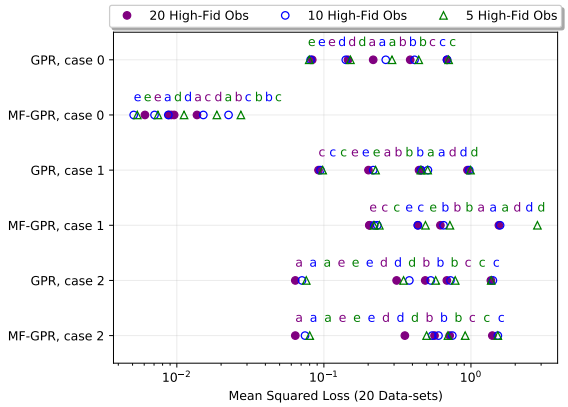
Compared to the uniformly distributed inputs, function **b** of the discontinuous ρ case performs worse than the counterpart with linearly spaced inputs. The predictions of this function with the GPR and MF-GPR show that the predictive variance is large and the predictive mean is almost equal to the prior mean or the low-fidelity, respectively, or rapidly goes towards the values of the observations (see figure 5.8). Across the datasets of function **b**, on average the noise variance is low, the signal variance is high, and the length-scale is low.



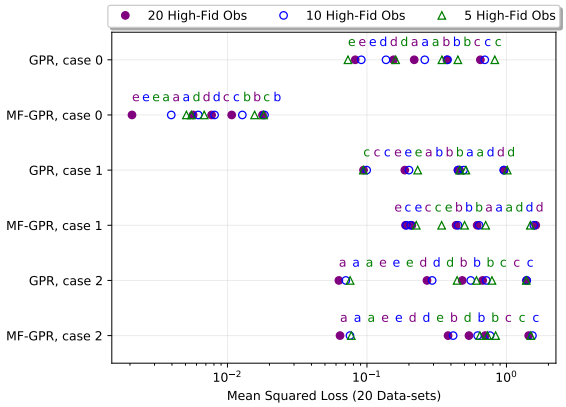
(a) 21 low-fids per region in interpolation regime: $[0,2]$.



(b) 101 low-fids per region in interpolation regime: $[0,2]$.

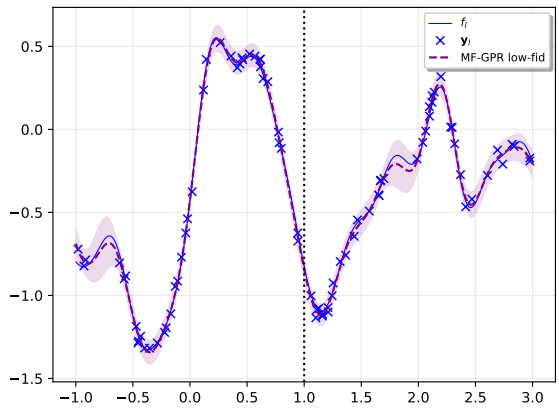


(c) 21 low-fids per region in extrapolation regime: $[-1,0]\cup[2,3]$.

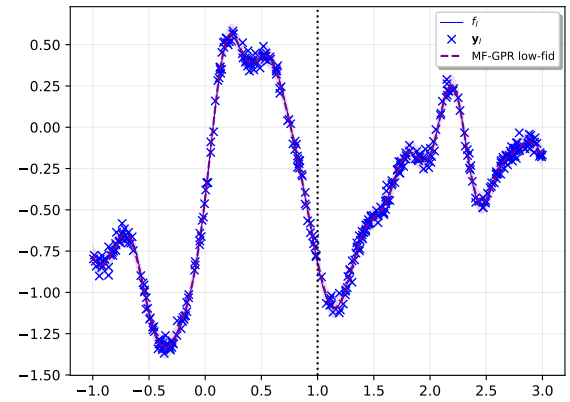


(d) 101 low-fids per region in extrapolation regime: $[-1,0]\cup[2,3]$.

Figure 5.4: performance of GPR and MF-GPR across the three cases with uniformly distributed inputs.

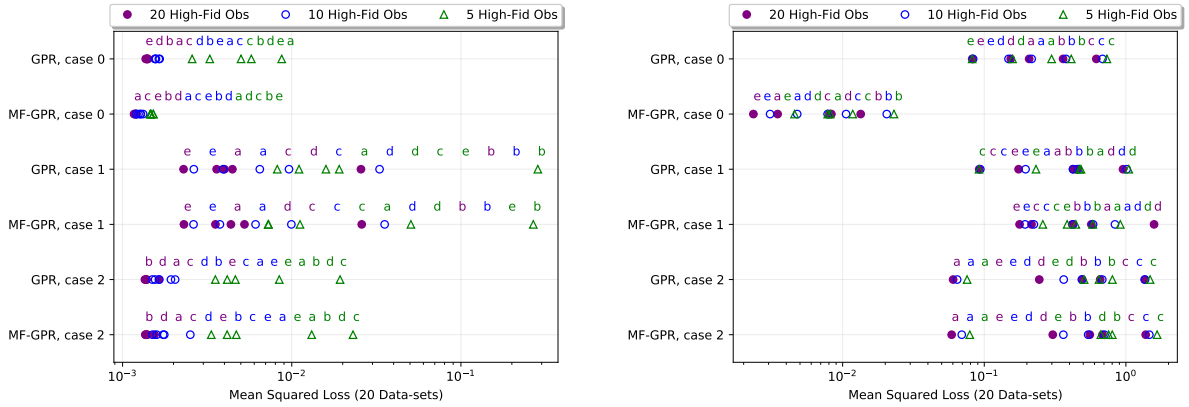


(a) 21 low-fids per region.



(b) 101 low-fids per region.

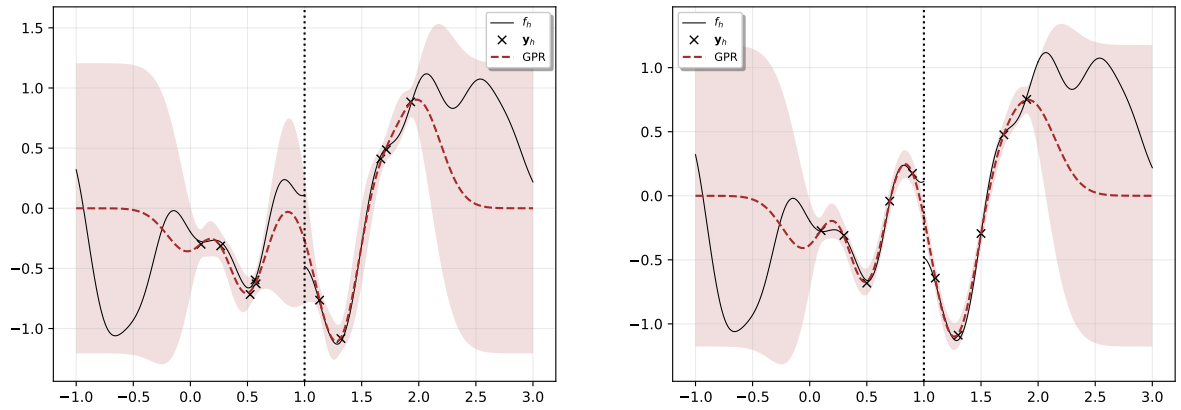
Figure 5.5: low-fidelity model predictions of MF-GPR of function **a** and data-set 0 of the constant ρ case: uniformly distributed, and 5 high-fids per region.



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

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

Figure 5.6: performance of GPR and MF-GPR across the three cases with linearly spaced inputs, 101 low-fids, and 20 high-fids per region.



(a) uniformly distributed.

(b) linearly spaced.

Figure 5.7: model predictions with GPR of function **a** and data-set 0 of the discontinuous ρ case.

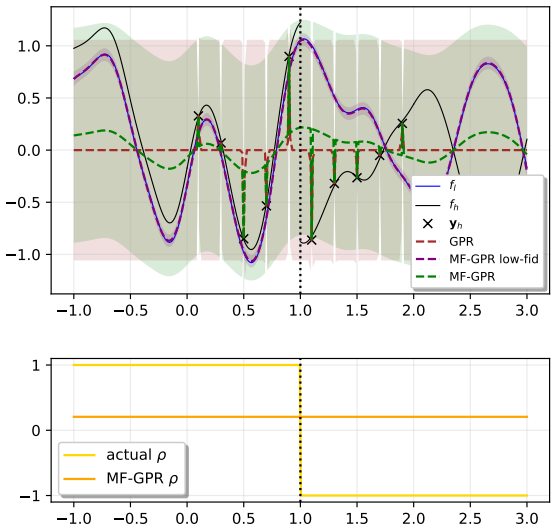


Figure 5.8: model predictions with GPR and MF-GPR of function **b** and data-set 0 of the discontinuous ρ case with 101 low-fids and 5 high-fids.

5.2. Splitting

This section explores the act of splitting a GPR and MF-GPR into two independent models. The performance before and after splitting is compared in both cases for both the interpolation and the extrapolation regime. The number of observations and sampling strategy experiments are not discussed as their trends are similar in nature to that of the previous section.

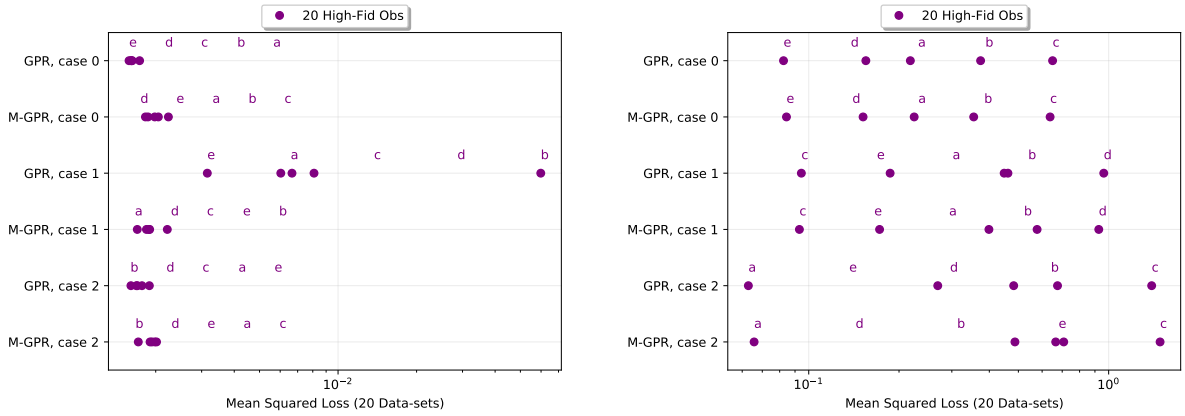
5.2.1. Single-fidelity - Interpolation and Extrapolation Regime

The prediction performance of splitting the GPR in the interpolation and extrapolation regime across the three cases is shown in figure 5.9. In the interpolation regime, the GPR outperforms the M-GPR in the constant ρ case. This is explained by the two independent models of the M-GPR each having fewer observations than the GPR model. This means that the inference is less accurate because the case is completely stationary, meaning that a single GPR is capable of inferring the underlying function completely if enough data is present.

In the interpolation regime, the M-GPR outperforms the GPR in the discontinuous ρ case. The explanation for this is opposite to the constant ρ case. As the case features a discontinuity that divides two distinctive stationary regions $[-1,1]$ and $[1,3]$, the act of splitting is able to capture both while the GPR struggles to capture the discontinuity. This behavior is more explicitly shown in Figure 5.10b which shows a GPR and M-GPR model's prediction of a dataset of the discontinuous ρ case. The M-GPR clearly models the discontinuity while the GPR introduces a large error at the boundary due to it being a continuous model.

The GPR outperforms the M-GPR slightly in the linearly varying ρ case. Although, the splitting might be better at capturing the non-linear rho this is clearly not the case. The two piecewise constant ρ of the M-GPR is still insufficient to capture the complex non-linear correlation between the low- and high-fidelity and the act of splitting, as is also the case for the constant ρ case, decreases the performance due to fewer observations.

The performance between the GPR and the M-GPR in the extrapolation regime is on average equal. This is explained by the fact that the correlation between the left or right extrapolation regime and the opposite region interpolation and extrapolation regime is low, which is caused by the chosen length scale of the cases themselves. This is also shown by the fact that the predictions of both methods go toward the zero mean prior in the extrapolation regime.



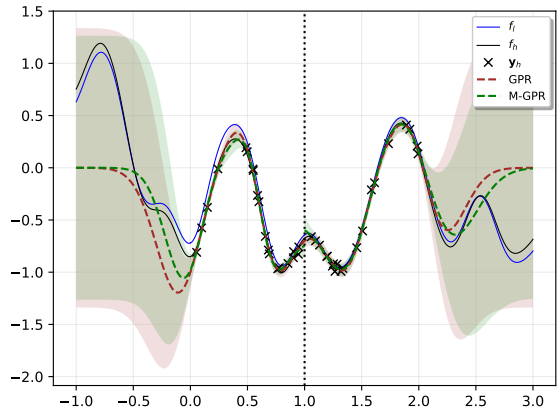
(a) interpolation regime: $[0,2]$.

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

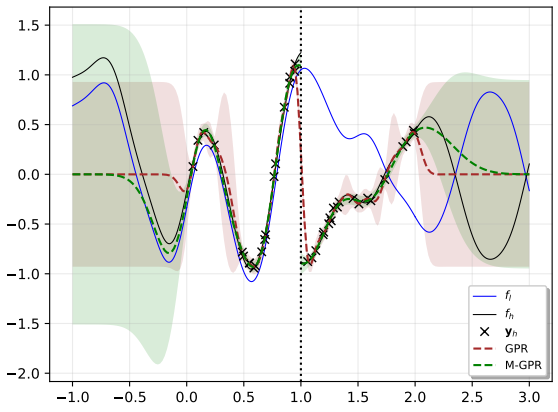
Figure 5.9: performance of GPR and M-GPR: uniformly distributed, 101 low-fids and 20 high-fids per region.

5.2.2. Multi-fidelity - Interpolation and Extrapolation Regime

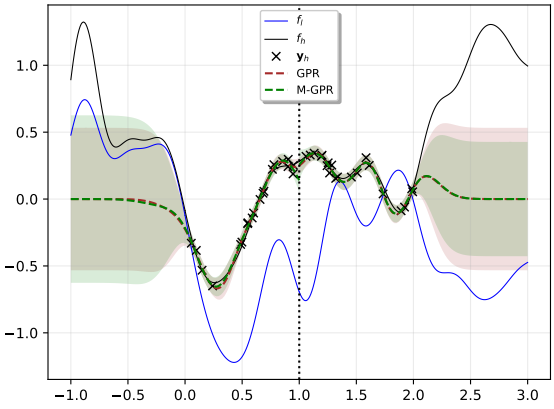
The prediction performance of the splitting of the MF-GPR in the interpolation and extrapolation regime across the three cases is shown in figure 5.11. The trends of the performance between the MF-GPR and M-MF-GPR in the interpolation regime are equivalent to the single-fidelity comparison of splitting. The explanation of these trends follows the same argumentation.



(a) constant ρ case.



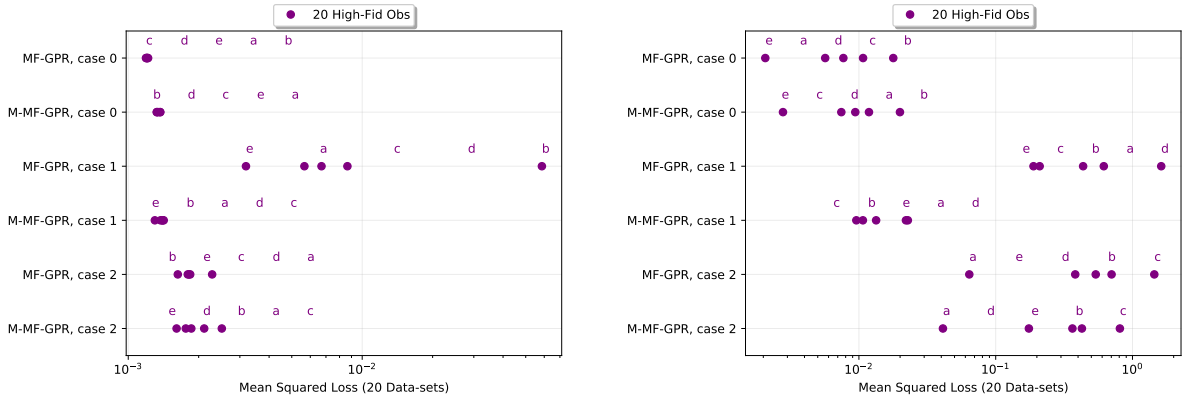
(b) discontinuous ρ case.



(c) linearly varying ρ case.

Figure 5.10: model predictions with GPR and M-GPR of function **b** and data-set 0 of the three cases with uniformly distributed inputs, 101 high-fids and 20 low-fids per region

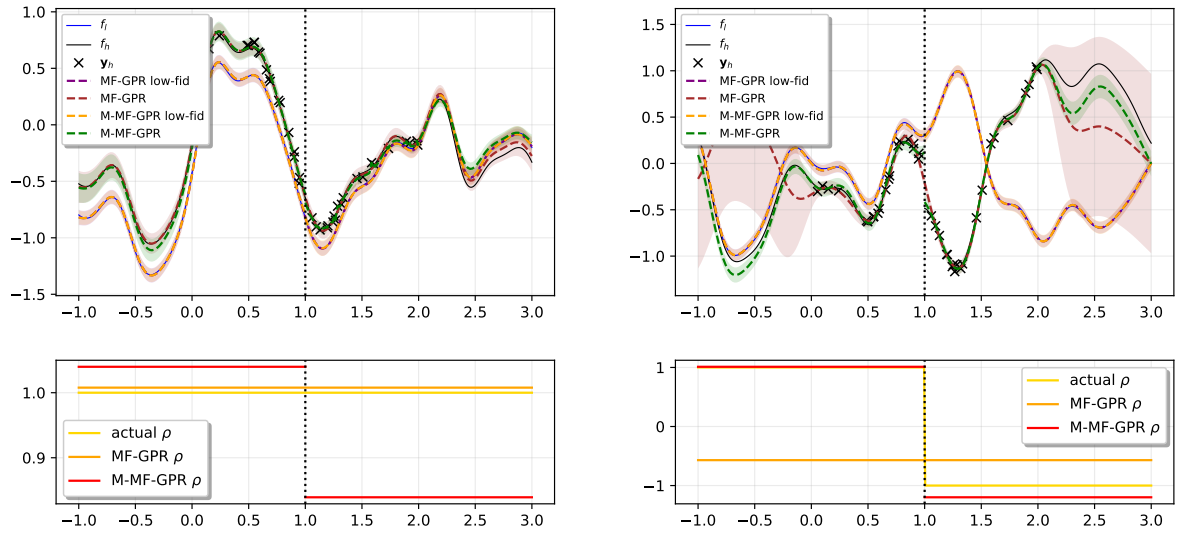
However, the extrapolation regime is not the same for the multi-fidelity methods. Splitting the MF-GPR performs equally well on average in the constant ρ case, but in the linearly varying ρ case and even more in the discontinuous ρ case splitting outperforms not splitting. Figure 5.12 shows the correlation coefficients of the MF-GPR and M-MF-GPR for a particular dataset of each case. In the constant ρ case the actual correlation coefficient on both sides of the boundary is equal, therefore the act of splitting does not improve this inference. This is not true in the other two cases. The discontinuous ρ sees the biggest difference because the actual correlation between the fidelities can be exactly inferred with two independent MF-GPR, which is not possible in the linearly varying ρ case.



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

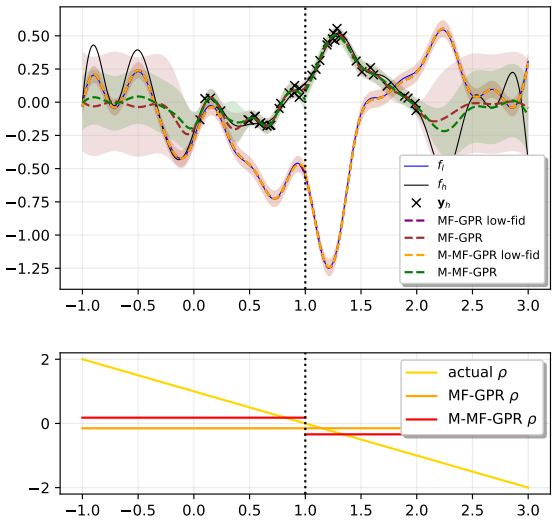
(b) extrapolation regime: $[-1,0] \cup [2,3]$.

Figure 5.11: performance of MF-GPR and M-MF-GPR: uniformly distributed, 101 low-fids and 20 high-fids per region.



(a) constant ρ case.

(b) discontinuous ρ case.



(c) linearly varying ρ case.

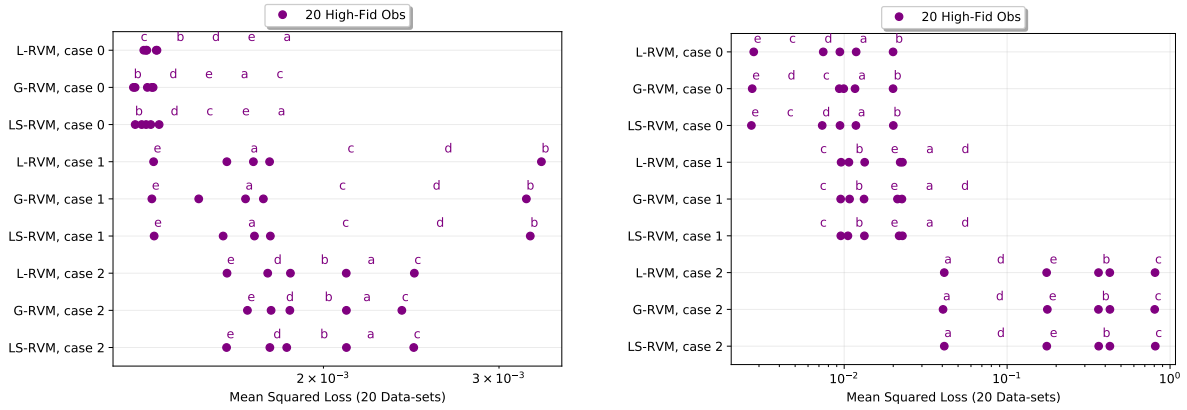
Figure 5.12: model predictions with MF-GPR and M-MF-GPR of function **a** and data-set 0 of the three cases with uniformly distributed inputs, 101 high-fids and 20 low-fids per region

5.3. RVM

Before moving on to comparing the stitching methods to the split and non-split MF-GPR, the peculiarities of the stitching methods are discussed. Because a better understanding of them provides a better base for comparison. First, the multi-fidelity extensions of the RVM-GPR are discussed. The performance of the three different low-fidelity modeling options in the three cases is compared, and thereafter the value of the length-scale parameter in the weights is investigated.

5.3.1. Low-fidelity Modeling - Interpolation and Extrapolation Regime

The prediction performance in the interpolation and extrapolation regime of the three multi-fidelity extensions of the RVM-GPR is shown in figure 5.13. The difference in performance between the three options in both regimes is small. However, the global extension is slightly preferred in the interpolation regime for the constant and discontinuous ρ case. This preference might be explained by the fact that the low-fidelity can be seen as being modeled by either an M-GPR, GPR, or single-fidelity RVM-GPR. For the latter, no previous results are shown, but, from previous results, the GPR model is shown to outperform the M-GPR in the constant ρ case. As the low-fidelity is completely stationary and thus analogous to the constant ρ case, this could explain the slightly increased performance of the global multi-fidelity extension of the RVM-GPR.



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

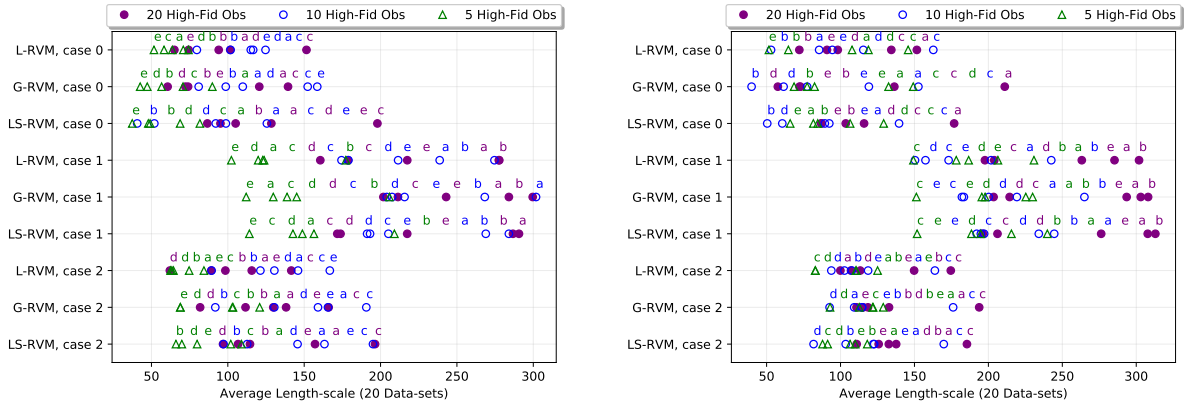
(b) extrapolation regime: $[-1,0] \cup [2,3]$.

Figure 5.13: performance of L-RVM, G-RVM, and LS-RVM: uniformly distributed, 101 low-fids and 20 high-fids per region.

5.3.2. Length-scale

The length scale of the L-RVM, G-RVM, and LS-RVM across the three cases is shown in figure 5.14. It generally holds that the length-scale of the weights is higher when the number of low- or high-fidelity observations increases. It is suspected that this is due to the inputs being closer to the boundary which in turn necessitates a shorter transition zone between the two local models.

The length-scale of the weights is often optimized towards a high value, meaning that the switching between the two local models at the boundary is brief and happens before the nearest observation to the boundary on both sides. In almost all cases a lower length-scale would result in unfitting the observations near the boundary as the local models go towards the zero-mean prior in their respective extrapolation regimes. Thus, a mixture is made between the zero-mean prior and the "correct" local model which obviously decreases the prediction accuracy when low length-scales are applied. One way to improve this method could be to optimize the hyperparameters of the weights and the kernels simultaneously, or by also optimizing the boundary itself. This could lead to other solutions than the length-scale of the weights being high.



(a) 21 low-fidelity observations.

(b) 101 low-fidelity observations.

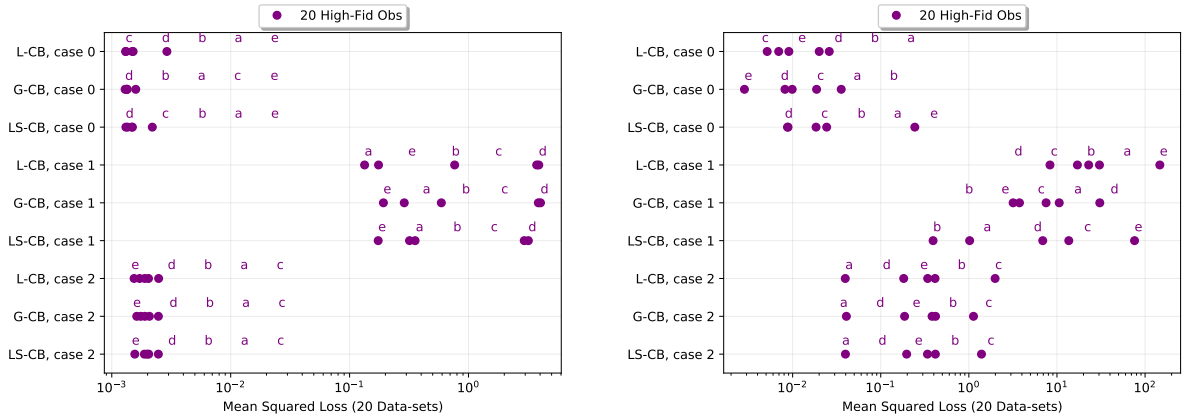
Figure 5.14: Length-scale of L-RVM, G-RVM, and LS-RVM: uniformly distributed, 101 low-fids and 20 high-fids per region.

5.4. CB

The performance of the multi-fidelity extensions of the CB-GPR is discussed in this section.

5.4.1. Low-fidelity Modeling - Interpolation and Extrapolation Regime

The prediction performance in the interpolation and extrapolation regime of the three multi-fidelity extensions of the CB-GPR is shown in figure 5.13. In the interpolation regime, the differences between the extensions are small, no extension is generally preferred over the other. In the extrapolation regime, there seems to be a preference for the global extension in the constant ρ case and for the locally stitched extension in the discontinuous ρ case, but by comparing the performance for each sampled function, it is quickly seen that on average these extensions do not outperform the others.



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

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

Figure 5.15: performance of L-CB, G-CB, and LS-CB: uniformly distributed, 101 low-fids and 20 high-fids per region.

5.5. Stitching

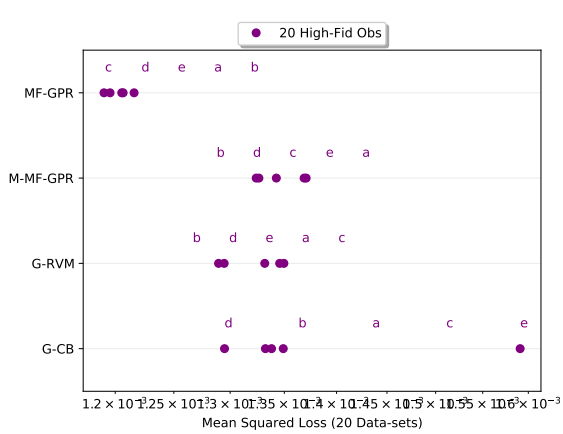
This section explores the act of stitching by comparing the MF-GPR and M-MF-GPR against the global multi-fidelity extension of the RVM-GPR and CB-GPR. The low-fidelity of the RVM-GPR and CB-GPR is modeled as a single GPR hence these methods are denoted with the pre-abbreviation "G-". These four methods are compared across the three cases in the interpolation and extrapolation regimes, and the influence of the number of low- or high-fidelity observations is investigated. The sampling strategy experiments are not discussed as the trends from the results are similar to the trends in the first section of this chapter.

5.5.1. Interpolation Regime

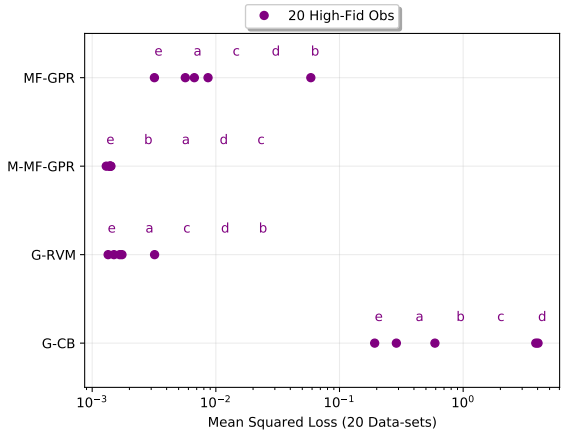
The prediction performance of the MF-GPR, M-MF-GPR, G-RVM and G-CB in the interpolation regime across the three cases is shown in figure 5.16. In the constant ρ case, the stitching methods perform worse than the MF-GPR. However, they do improve slightly over the M-MF-GPR with sampled function **e** of G-CB as an exception. This slight improvement is possibly attributed to an improved prediction accuracy around the boundary. Figure 5.17a strengthens this hypothesis by demonstrating that, for this particular prediction model plot, the act of stitching improves the performance around the boundary.

In the discontinuous ρ case, the stitching methods are outperformed by the M-MF-GPR. Although, the RVM-GPR still outperforms the MF-GPR, the G-CB does not. Clearly, the act of stitching hinders the prediction accuracy at discontinuous boundaries. Figure 5.17b shows that the act of stitching creates a continuous prediction model. However, as the boundary is discontinuous this transition zone between the left and right local model is unnecessary and produces poorer predictions.

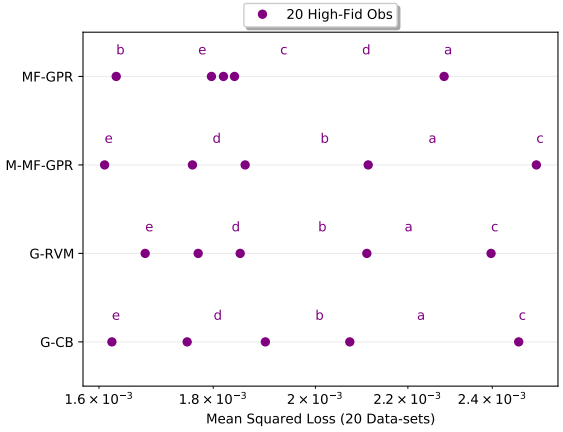
In the linearly varying ρ case, the stitching methods all perform equally well compared to the MF-GPR and M-MF-GPR. As the act of splitting does not improve the predictions, so does the act of stitching. Note, that the mean squared loss is already quite small when comparing it to the other cases, which have similar amplitude ranges in their underlying functions. Thus, the predictions of the non-stitching methods are already quite accurate, which means that there is only a small window for improvement for the stitching methods. As the boundary is continuous, the stitching methods do not have a more favourable position and do thus not produce better results.



(a) constant ρ case.

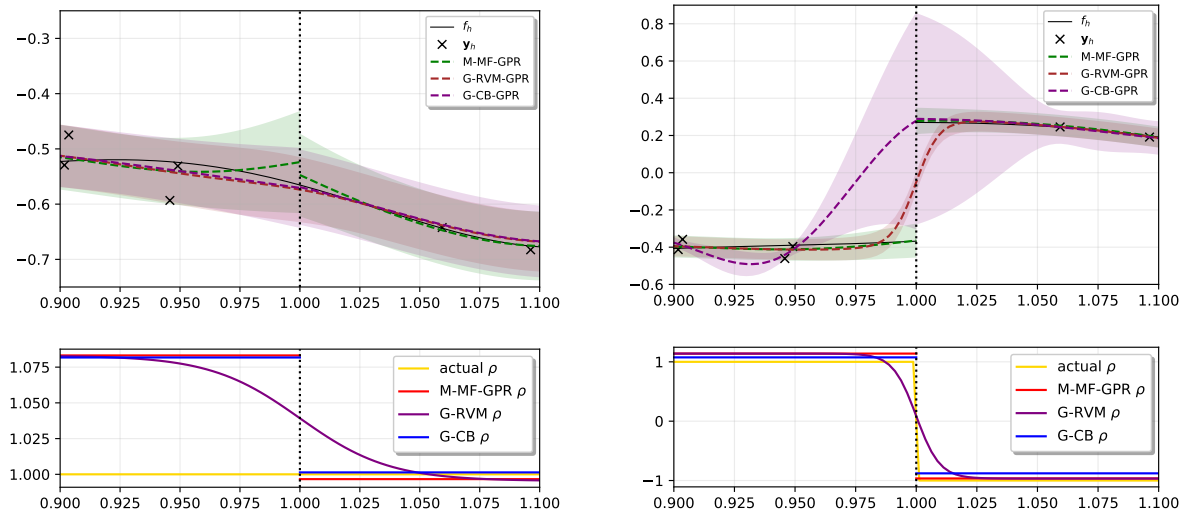


(b) discontinuous ρ case.



(c) linearly varying ρ case.

Figure 5.16: performance of MF-GPR, M-MF-GPR, G-RVM, and G-CB: uniformly distributed, 101 low-fids and 20 high-fids per region in interpolation regime: $[0, 2]$.



(a) constant ρ case.

(b) discontinuous ρ case.

Figure 5.17: model predictions with M-MF-GPR, G-CB and G-RVM of function **d** and data-set 0 of the constant and discontinuous ρ cases with uniformly distributed inputs, 101 high-fids and 20 low-fids per region.

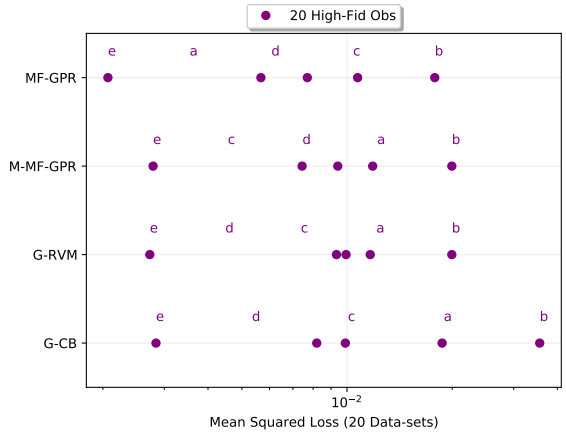
5.5.2. Extrapolation Regime

The prediction performance of the MF-GPR, M-MF-GPR, G-RVM and G-CB in the extrapolation regime across the three cases is shown in figure 5.18. The stitching methods perform almost similarly in the extrapolation regime compared to the M-MF-GPR. It seems that the act of stitching mostly improves the prediction accuracy around the boundary as stated earlier in the interpolation regime section. However, the G-CB performs much worse in the discontinuous ρ case. This shows that enforcing continuity at a discontinuous boundary results in a poor model overall, as for this model in this case both the interpolation and extrapolation regime are compromised.

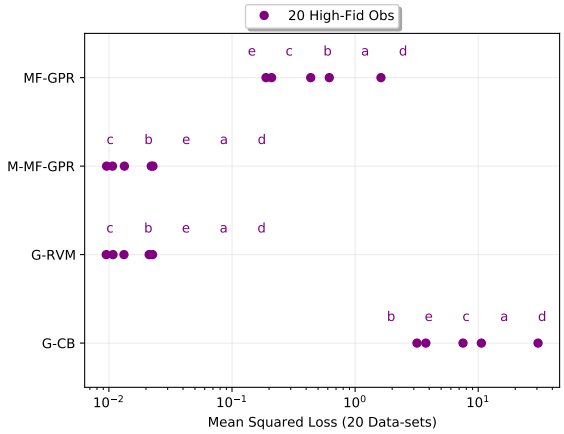
5.5.3. Number of Observations

The prediction performance of the MF-GPR, M-MF-GPR, G-RVM and G-CB in the interpolation regime and extrapolation regime across the constant ρ case with all low- and high-fidelity options is shown in figure 5.19. In the interpolation regime, the performance increases when the number of high-fidelity observations increases with the exception of the G-CB in the 21 low-fids experiments. No such correlation between the number of high-fidelity observations and performance seems to be present in the extrapolation regime; it seems more correlated to its particular sampled function. Although, a slight bias towards more high-fidelity observations is seen. These trends about the performance with respect to the number of high-fidelity observations are also seen in the other two cases, see Figures 5.20 and 5.21.

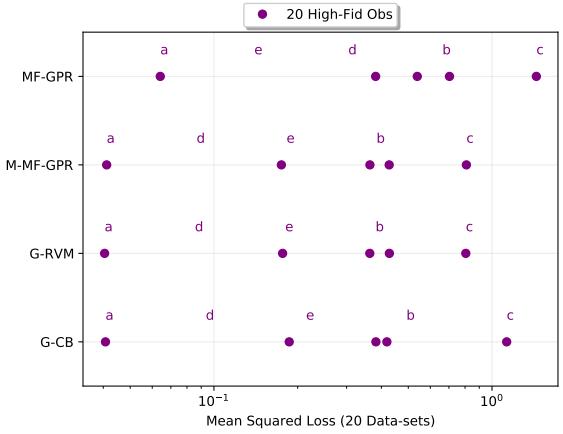
Among these results, the G-CB seems to have more outliers and in some cases increasing the number of high-fidelity observations does not improve the prediction accuracy in the interpolation regime. Most notably, this is the case when only 21 low-fidelity observations are present. Also, the performance correlation to the particular sampled functions in the extrapolation regime, seems to be less present with the G-CB.



(a) constant ρ case.

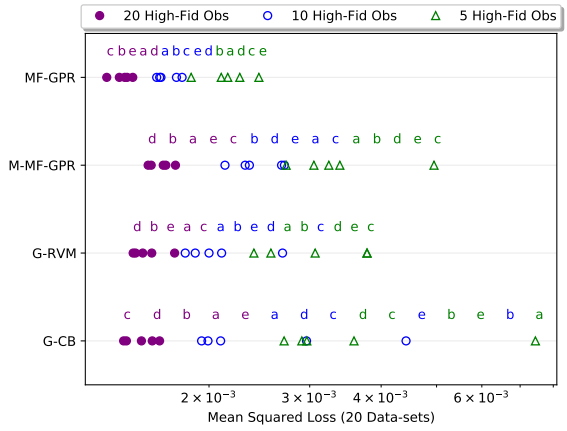


(b) discontinuous ρ case.

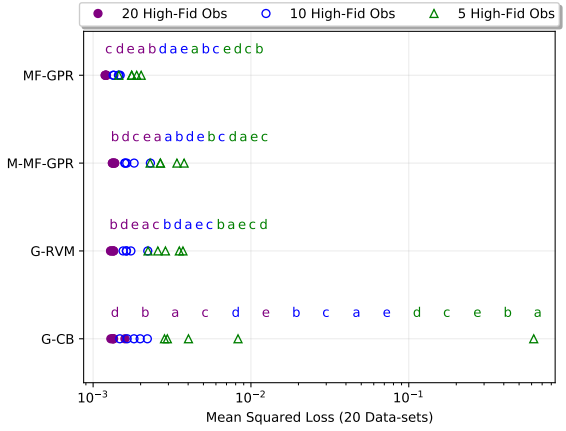


(c) linearly varying ρ case.

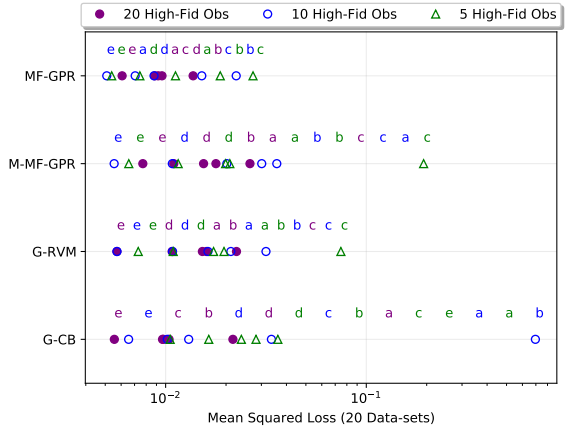
Figure 5.18: performance of MF-GPR, M-MF-GPR, G-RVM, and G-CB: uniformly distributed, 101 low-fids and 20 high-fids per region in extrapolation regime: $[-1,0] \cup [2,3]$.



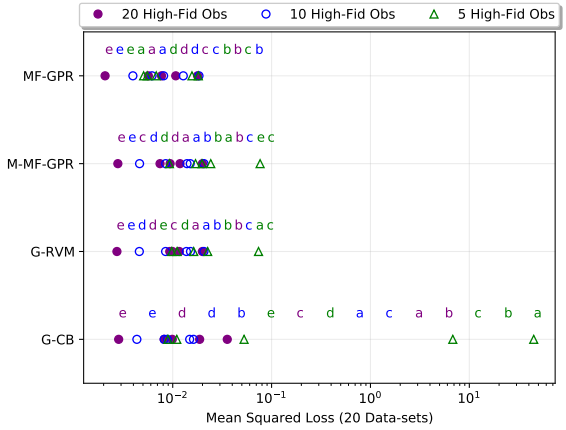
(a) 21 low-fids per region in interpolation regime: $[0,2]$.



(b) 101 low-fids per region in interpolation regime: $[0,2]$.

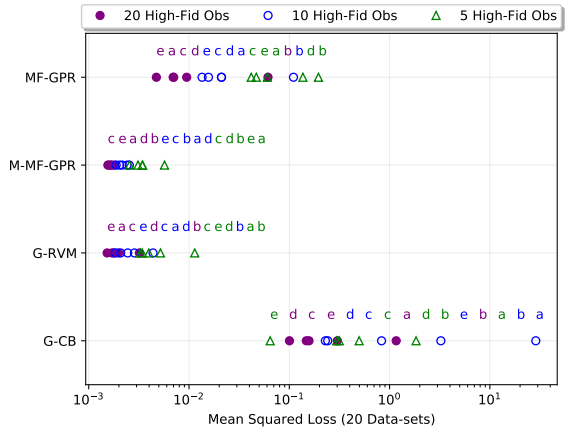


(c) 21 low-fids per region in extrapolation regime: $[-1,0] \cup [2,3]$.

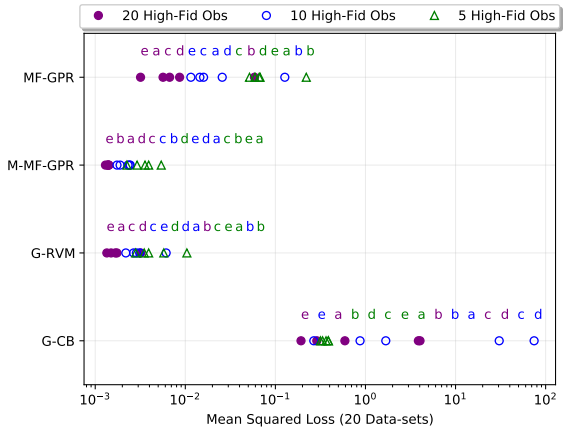


(d) 101 low-fids per region in extrapolation regime: $[-1,0] \cup [2,3]$.

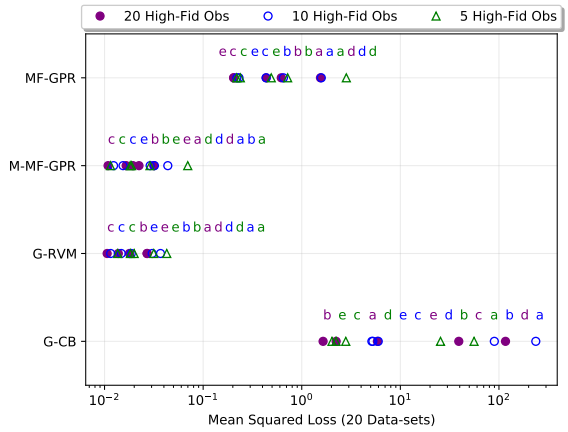
Figure 5.19: performance of MF-GPR, M-MF-GPR, G-RVM, and G-CB across the constant ρ case with uniformly distributed inputs.



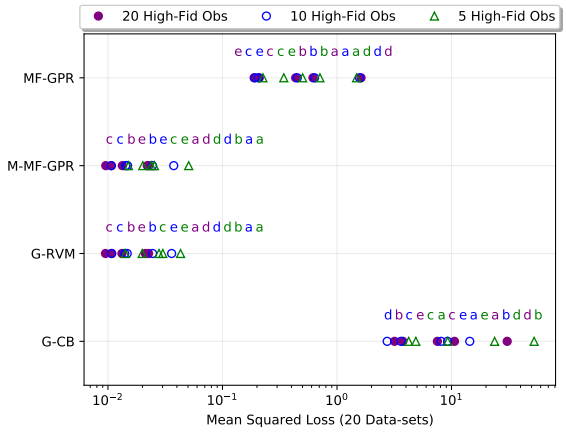
(a) 21 low-fids per region in interpolation regime: $[0,2]$.



(b) 101 low-fids per region in interpolation regime: $[0,2]$.

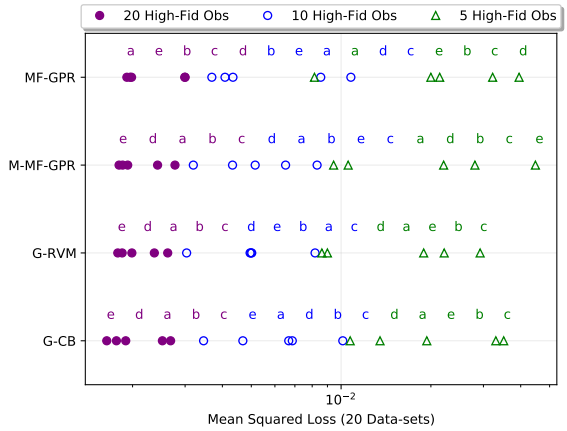


(c) 21 low-fids per region in extrapolation regime: $[-1,0] \cup [2,3]$.

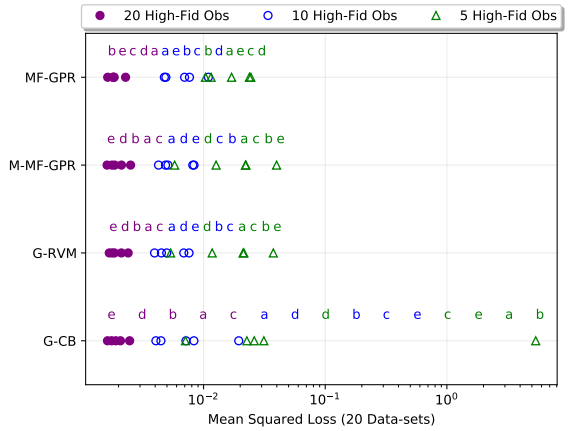


(d) 101 low-fids per region in extrapolation regime: $[-1,0] \cup [2,3]$.

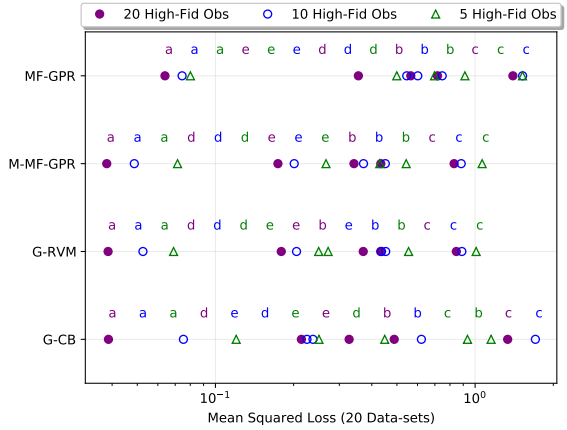
Figure 5.20: performance of MF-GPR, M-MF-GPR, G-RVM, and G-CB across the discontinuous ρ case with uniformly distributed inputs.



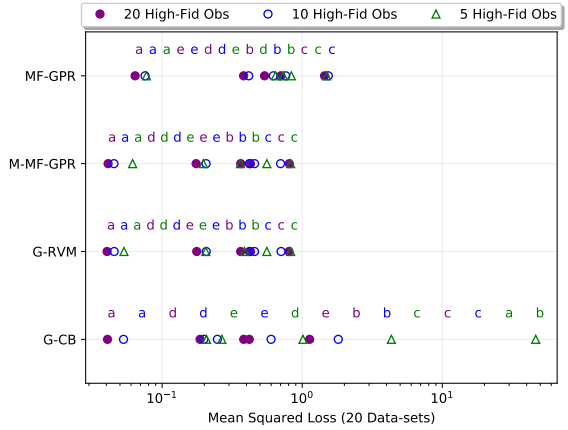
(a) 21 low-fids per region in interpolation regime: $[0,2]$.



(b) 101 low-fids per region in interpolation regime: $[0,2]$.



(c) 21 low-fids per region in extrapolation regime: $[-1,0] \cup [2,3]$.



(d) 101 low-fids per region in extrapolation regime: $[-1,0] \cup [2,3]$.

Figure 5.21: performance of MF-GPR, M-MF-GPR, G-RVM, and G-CB across the linearly varying ρ case with uniformly distributed inputs.

6

Conclusions and Future Work

Conclusions:

- Multi-fidelity GPR models with constant correlation do not improve upon the single-fidelity GPR models when non-linear correlation is present, for example, in the discontinuous and the linearly varying ρ case. This holds true in both the interpolation and extrapolation regimes.
- The performance of all methods generally increases with an increased number of high-fidelity observations.
- In general, the performance is more correlated to the particular sampled function than the number of high-fidelity observations in the extrapolation regime.
- In general, the performance is better when the inputs are linearly spaced compared to uniformly distributed. This is mostly attributed to the denser coverage the linearly spaced inputs have when comparing them to the uniformly distributed inputs with the same number of observations.
- The GPR split methods, M-GPR and M-MF-GPR, perform better than the non-split method, GPR and MF-GPR, in discontinuous cases. Thus, splitting GPR models at discontinuities in the underlying function increases the performance. With non-discontinuous cases, the split GPR methods generally perform slightly worse than their non-split counterparts. This is attributed to the decrease in the number of observations per model in the method.
- The three extensions of the CB-GPR to multi-fidelity perform equally. This almost holds true for the RVM-GPR, but the global extension has a very slight preference.
- In the constant ρ case, the MF-GPR outperforms the splitting and stitching methods. However, the stitching methods improve over the splitting methods. In the discontinuous ρ case, the M-MF-GPR and G-RVM improve over the MF-GPR but the G-CB performs worse. In the linearly varying ρ case, all methods perform equally well. In essence, stitching is only preferred in stationary cases and does not add much improvement in non-linear correlation settings.
- In the current context the RVM method will only perform slightly better or worse than the M-MF-GPR method. This is because the center parameter in the weights is set to the actual boundary and not optimized separately. And, the method is almost equivalent to the M-MF-GPR method except that RVM adds the weighing of the local predictions on top. As the predictions of the local models are almost continuous and continuous in their derivative at the boundary this results in not much difference between the models.

Future Work:

- The case construction method, used in this thesis: sampling MultiGPRs, could serve, with a different perspective, an interesting course of study into the understanding of regression methods. Instead of choosing the hyperparameters of the sampled MultiGPRs, they could be obtained by optimizing them such that they provide the largest difference in performance between the two methods, of course, averaged over samples and dataset realizations. A similar idea is presented by Wilde et. al. [52] in which an evolutionary algorithm generates artificial datasets for which a specific method performs well on a given metric. Note, the process can be generalized to finding a functional distribution in a given space, instead of looking from an optimization of hyperparameters point of view. For the generation of the functional distribution from which the datasets are obtained, the GP sum-product network of Trapp et. al. [48] can be used as it provides a richer structure than a GP. Another approach could be to use, just like Wilde et. al. [52], an evolutionary algorithm to add the kernel constructing methods, as mentioned by Rasmussen & Williams [53] and Duvenaud [8], to the optimization process; a translation of genes, in the evolutionary algorithm, to particularly constructed kernels could be made by creating a sort of grammar (L-system) out of the kernel construction processes. All of this sounds great, but the parameter space in which the optimization process takes place can easily get out of proportion. Luckily, no real-world data is needed, but the experiment is obviously constrained by computational time.

- How much of a method's performance can be attributed to the method itself? Major players in this league are chance, method assumptions, and the cases themselves. As change's influence is reduced by measuring the performance using the expected test error, the question shifts to what extent does the performance correlate to the matching of the assumptions behind the method and the case? As in, if this matching is identical then this would probably lead to the most optimal, in some sense, method and case combination. Although of less interest but not equally unimportant, this begs the reverse of finding a suitable method for a case: finding a suitable case for a method. This reverse ideology could enhance the understanding of particular machine learning methods.

- The cases are sampled with an MF-GPR which has an additive part that is unequal to zero. This introduces variations in the actual correlation between the two fidelities. More experiments can be run in which this additive part is zero, such that the resulting correlation of the models more accurately reflects the chosen correlation in the cases and, thus, the predetermined correlation might be inferred correctly.

- Constrain the CB method in points instead by adding observations with no observational noise. As it can be shown, mathematically, the mean of the predictive distribution is equal to the observation.

- In this thesis only one-dimensional inputs are considered. However, when considering inputs of higher dimensions, the stitching methods must change, because the boundary is not a point anymore. The CB-GPR must be constrained across several points on the boundary between two local models and the RVM-GPR does not have a center parameter with respect to the boundary but a different weight function must be chosen.

A

Experimental Results

The terms "uniformly distributed" and "linearly spaced" are the sampling option for the observational inputs.

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

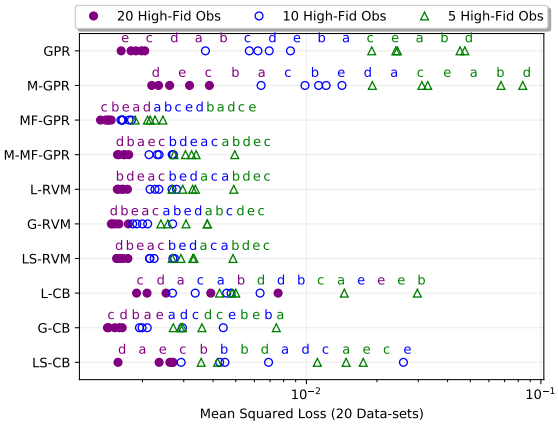
The appendix contents provide an efficient browsing option. Each box plot and prediction plot that is referenced has a reference back.

A.1. Appendix Contents

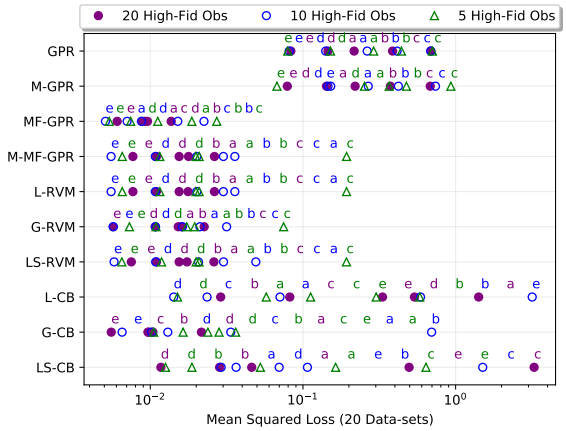
A.2. Uniformly Distributed

A.2.1. Constant ρ case

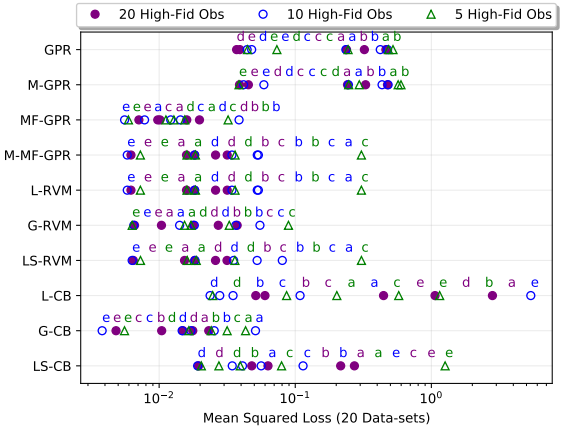
Appendix A: Experimental Results



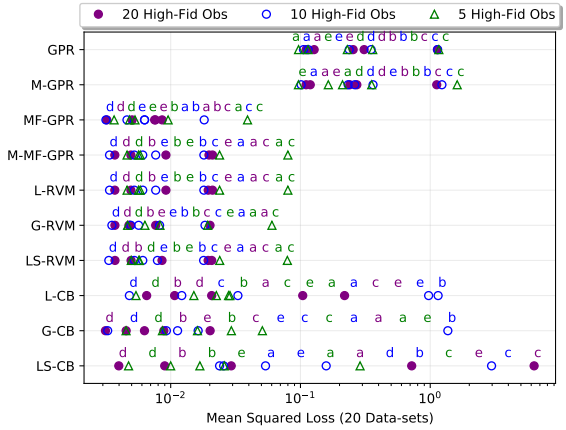
(a) Interpolation regime: $[0,2]$.



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



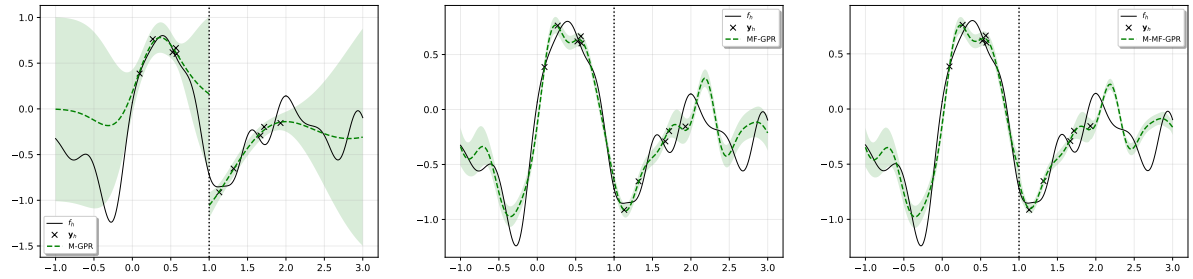
(c) Left extrapolation regime: $[-1,0]$.



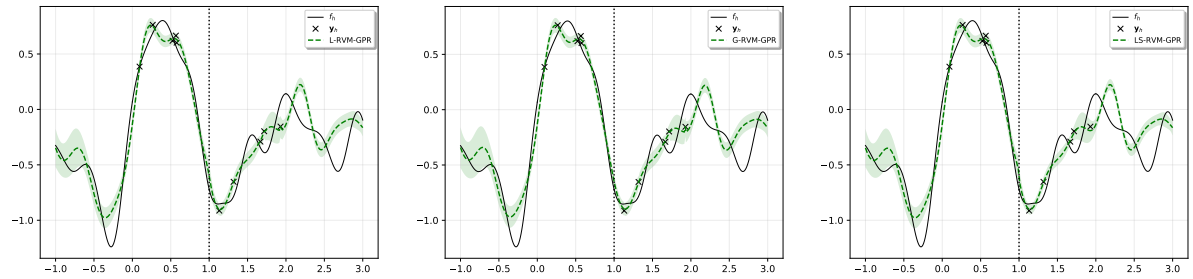
(d) Right extrapolation regime: $[2,3]$.

Figure A.1: constant ρ case, uniformly distributed, and 21 low-fids per region.

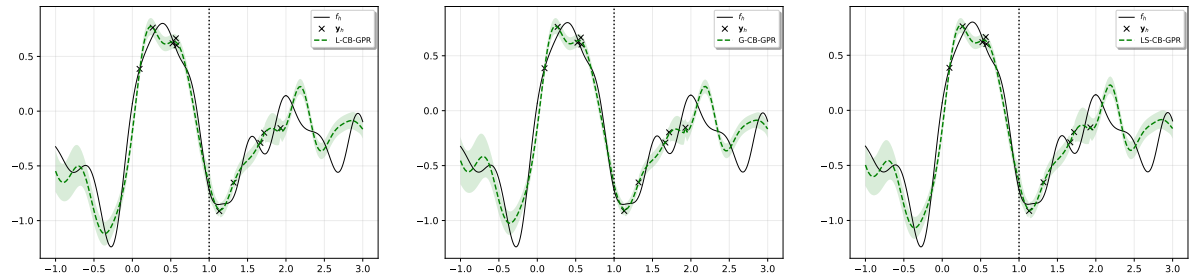
Appendix A: Experimental Results



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



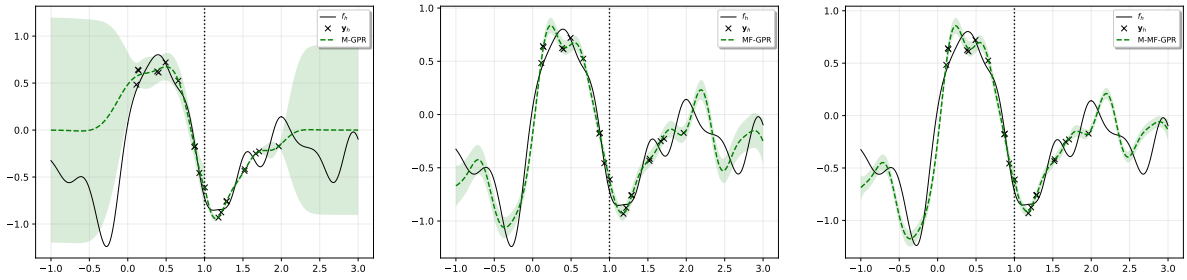
(d) L-RVM (e) G-RVM (f) LS-RVM



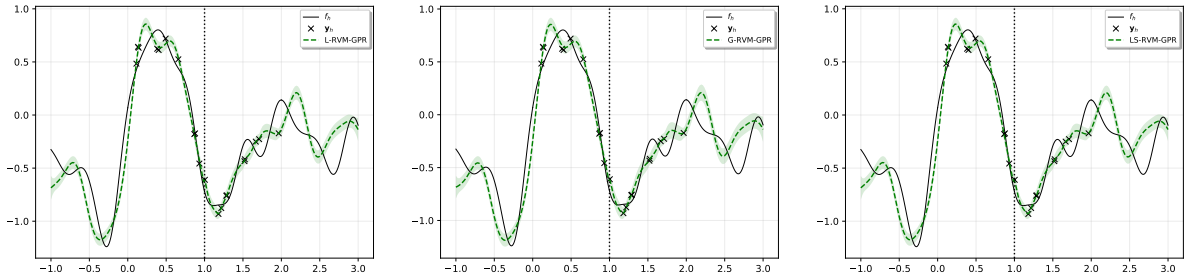
(g) L-CB (h) G-CB (i) LS-CB

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

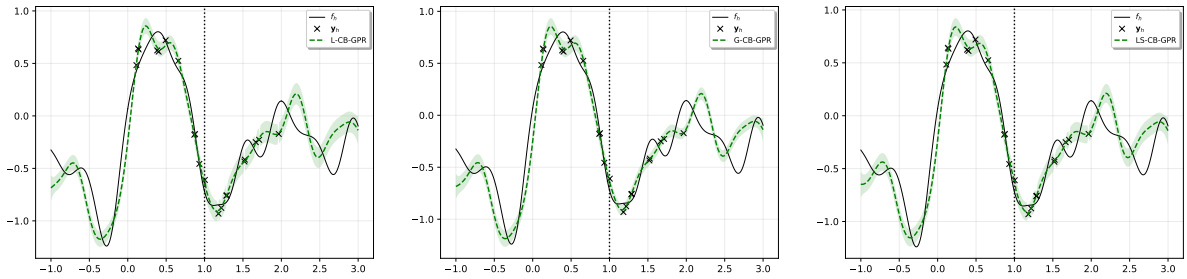
Appendix A: Experimental Results



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



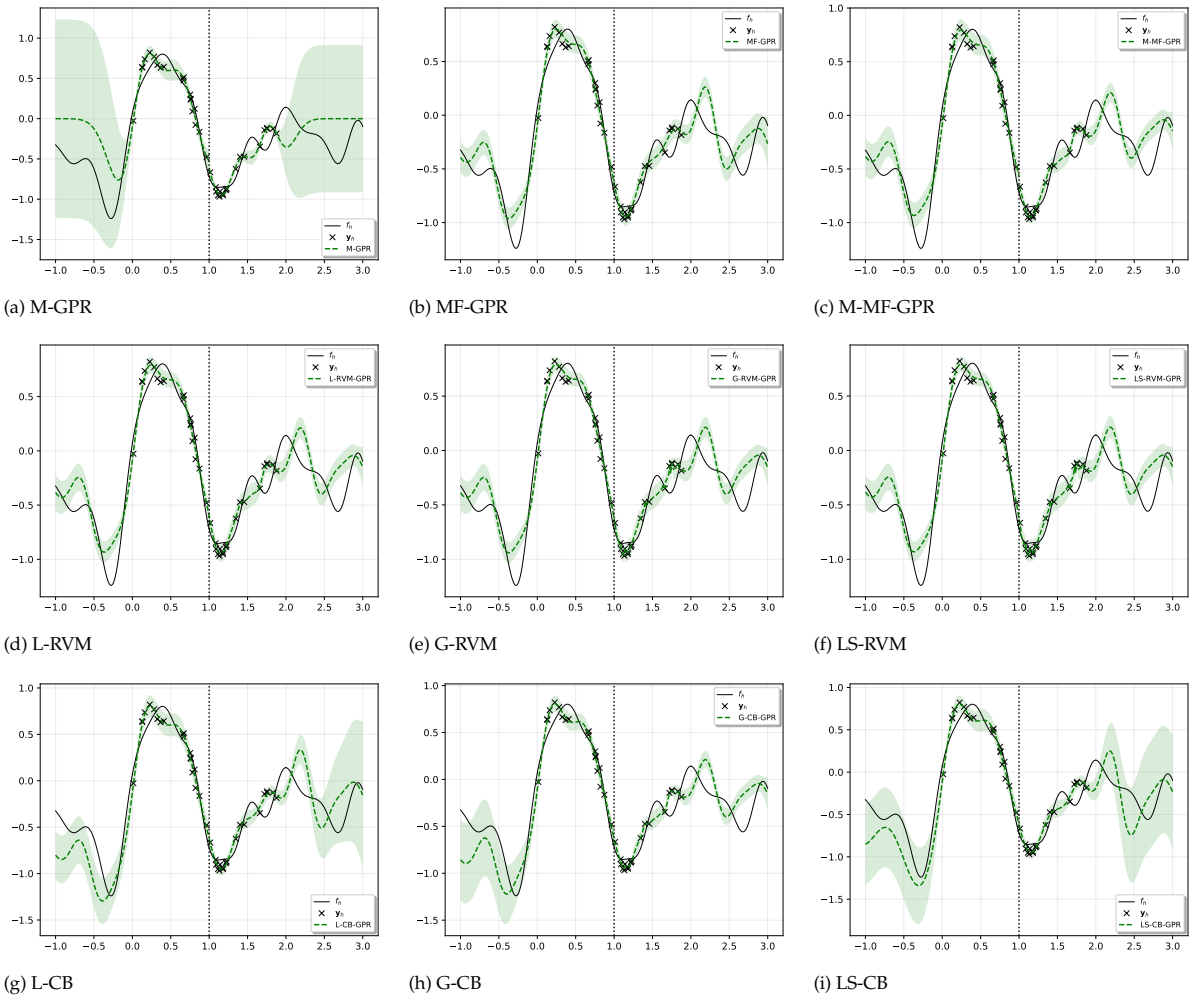
(d) L-RVM (e) G-RVM (f) LS-RVM



(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



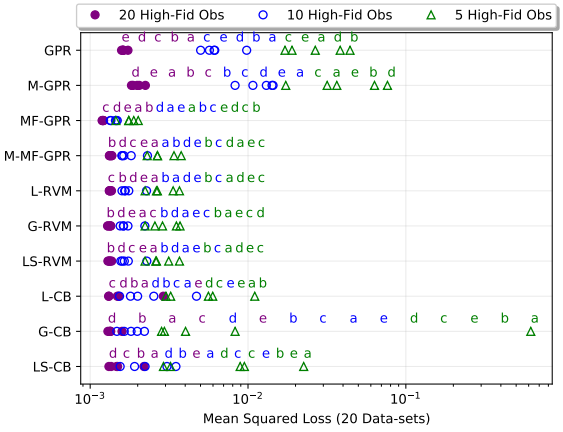
(a) M-GPR (b) MF-GPR (c) M-MF-GPR

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

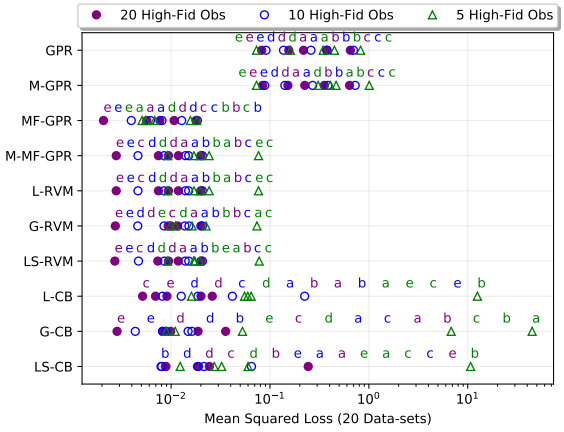
(g) L-CB (h) G-CB (i) LS-CB

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

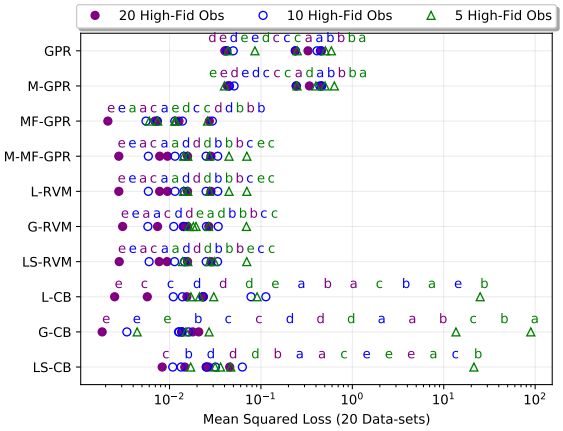
Appendix A: Experimental Results



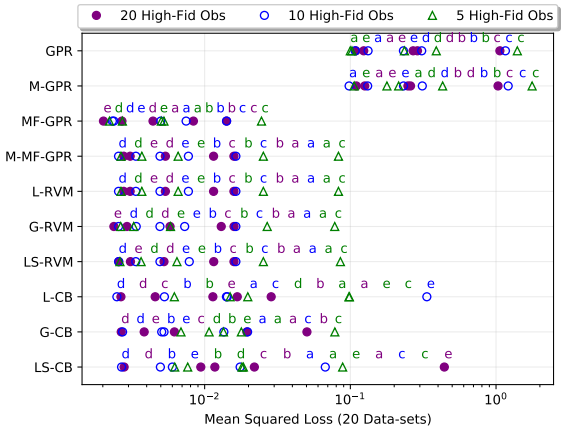
(a) Interpolation regime: $[0,2]$.



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



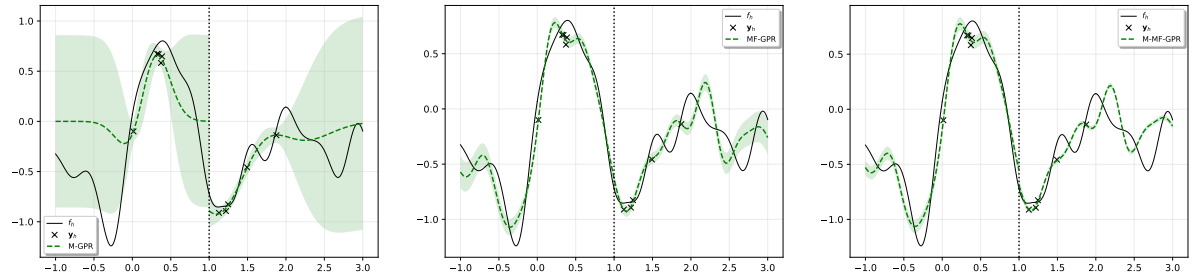
(c) Left extrapolation regime: $[-1,0]$.



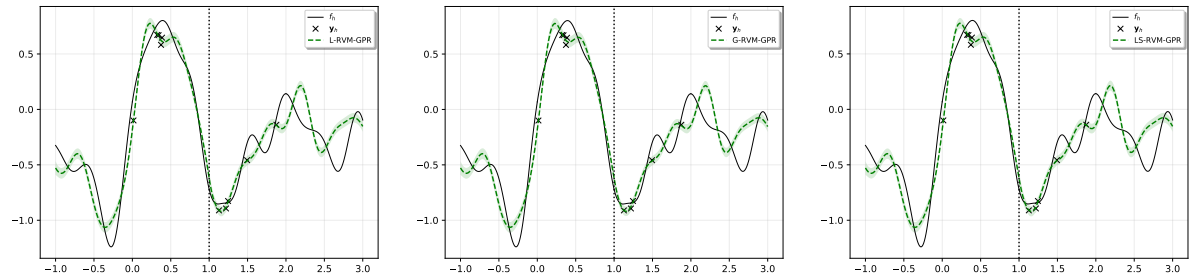
(d) Right extrapolation regime: $[2,3]$.

Figure A.5: constant ρ case, uniformly distributed, and 101 low-fids per region.

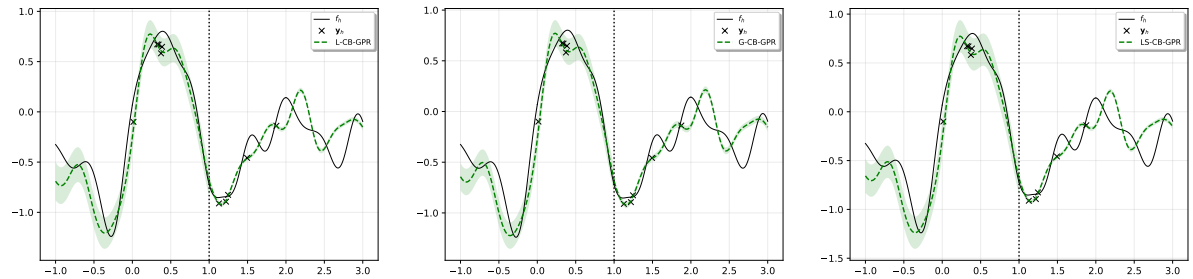
Appendix A: Experimental Results



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



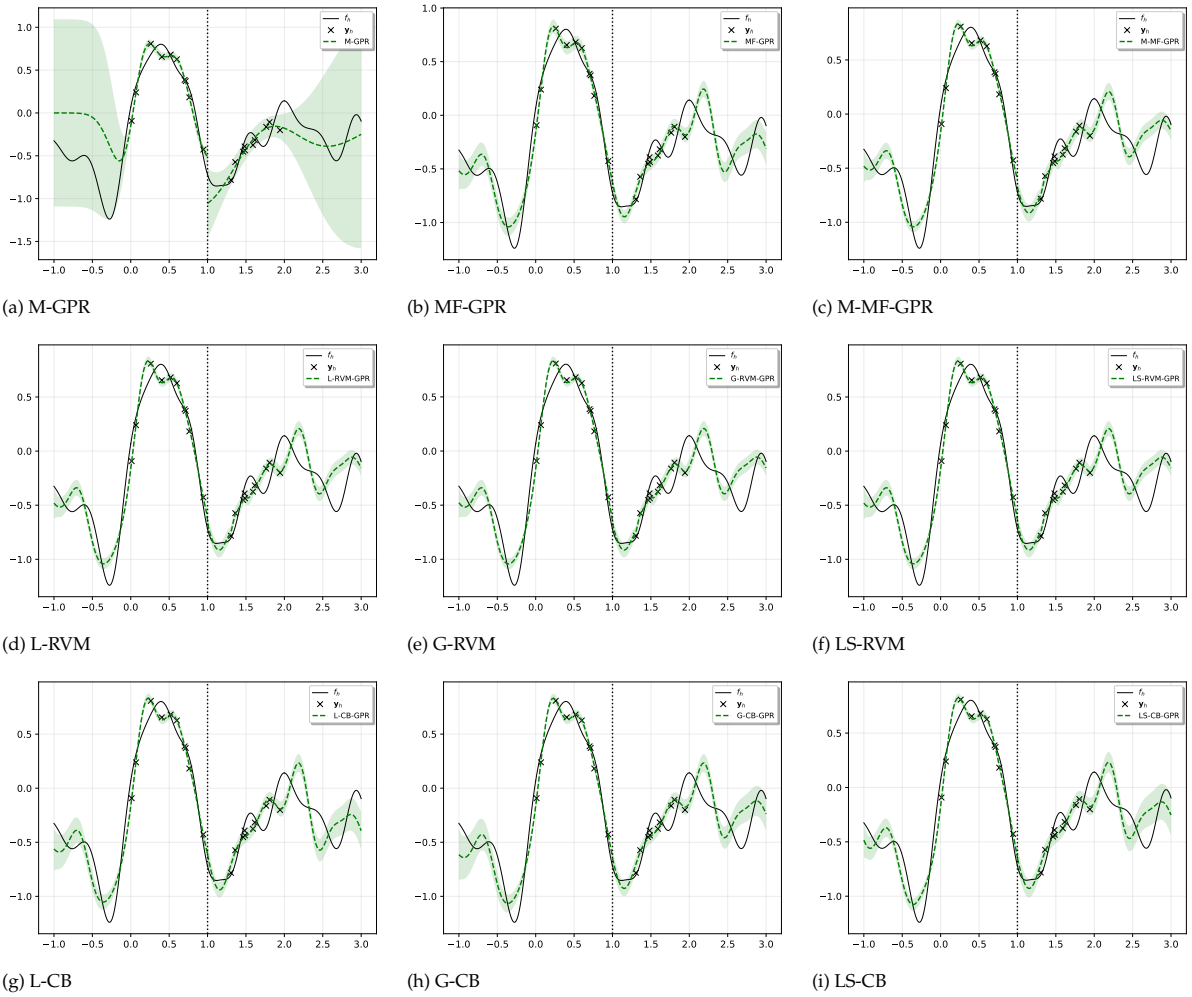
(d) L-RVM (e) G-RVM (f) LS-RVM



(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



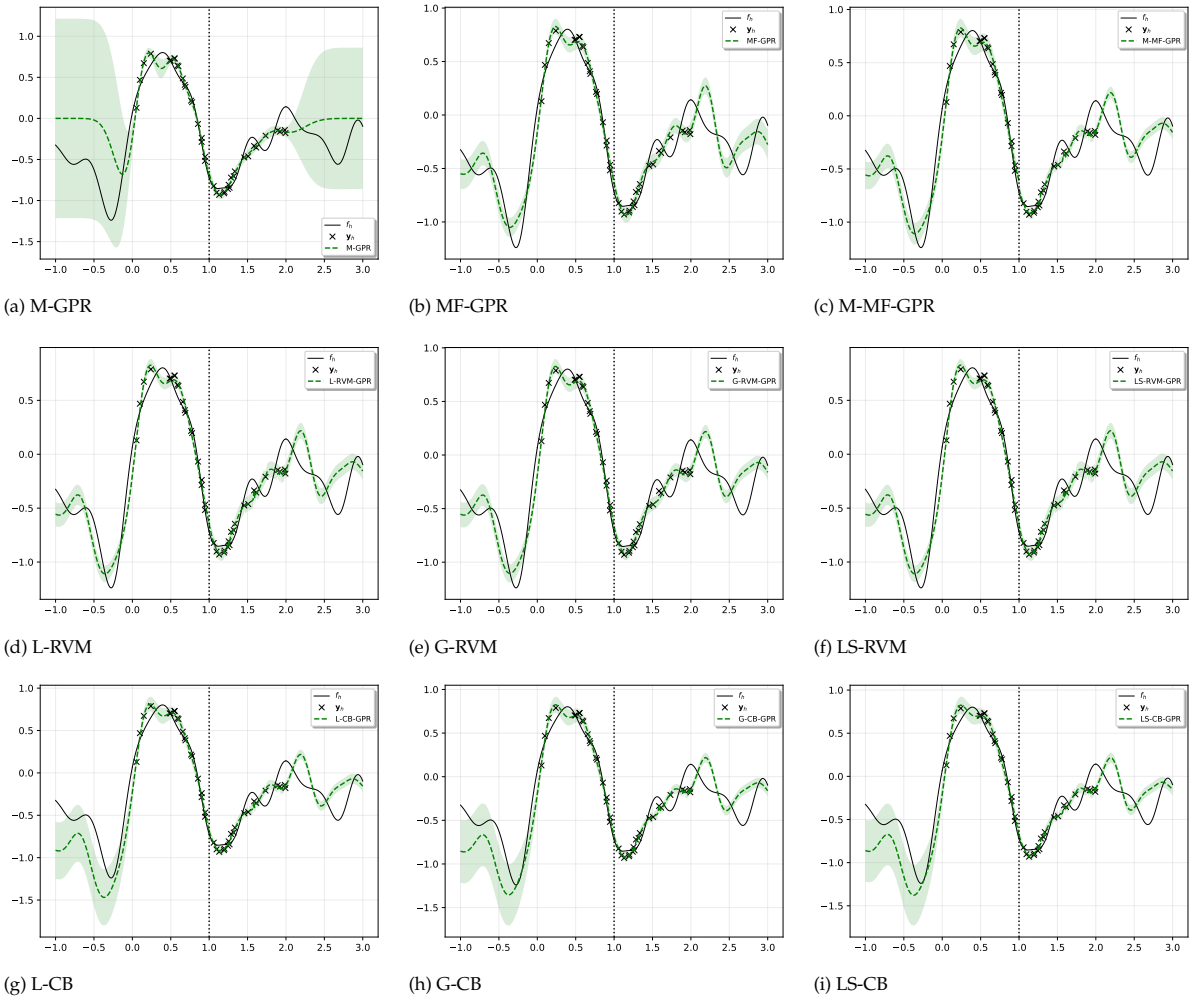
(a) M-GPR (b) MF-GPR (c) M-MF-GPR

(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

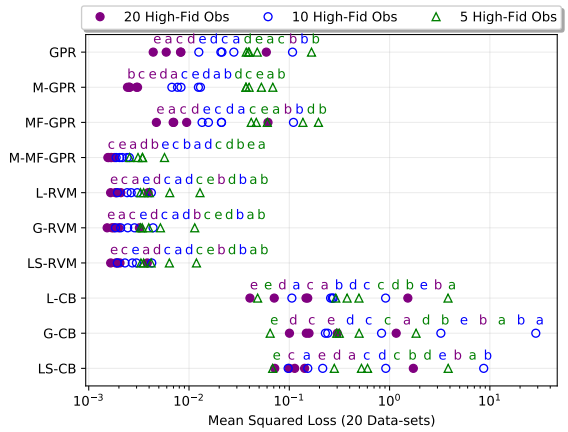
(d) L-RVM (e) G-RVM (f) LS-RVM

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

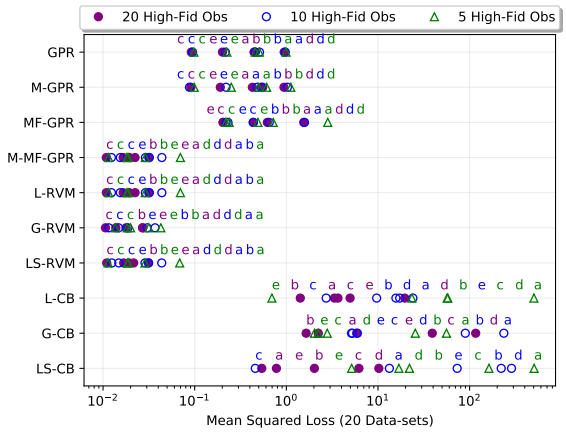
Figure A.8: model predictions with 20 high-fids per region of function **a** and data-set 0 of figure A.5.

A.2.2. Discontinuous ρ case

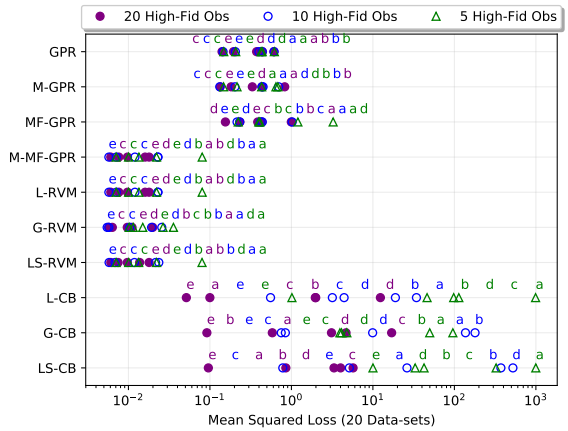
Appendix A: Experimental Results



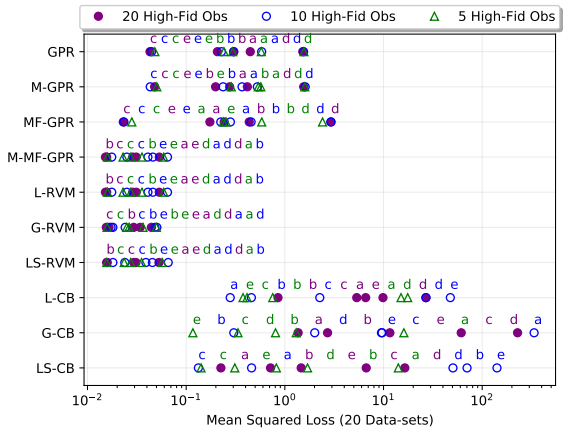
(a) Interpolation regime: $[0,2]$.



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



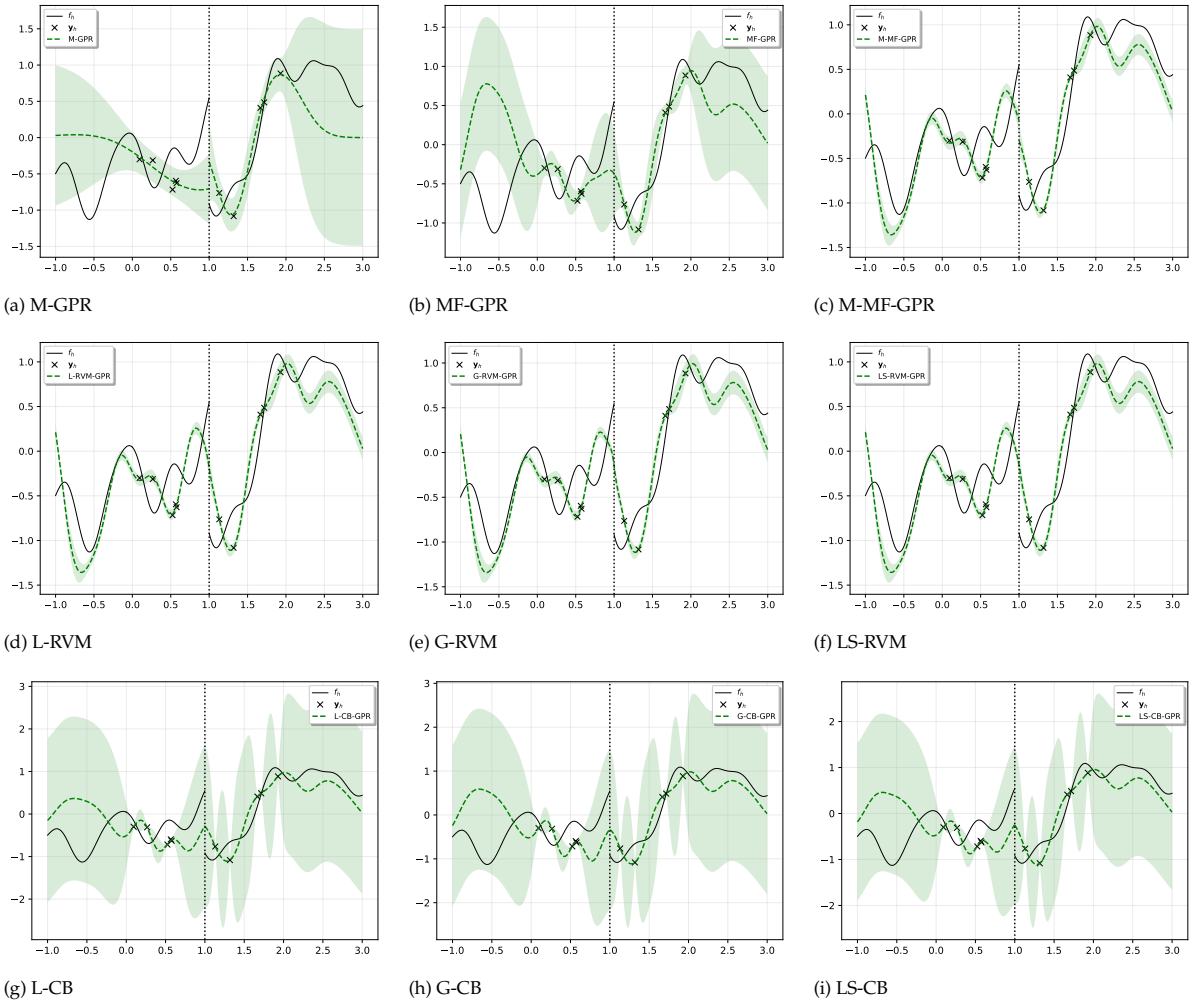
(c) Left extrapolation regime: $[-1,0]$.



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

Figure A.9: discontinuous ρ case, uniformly distributed, and 21 low-fids per region.

Appendix A: Experimental Results



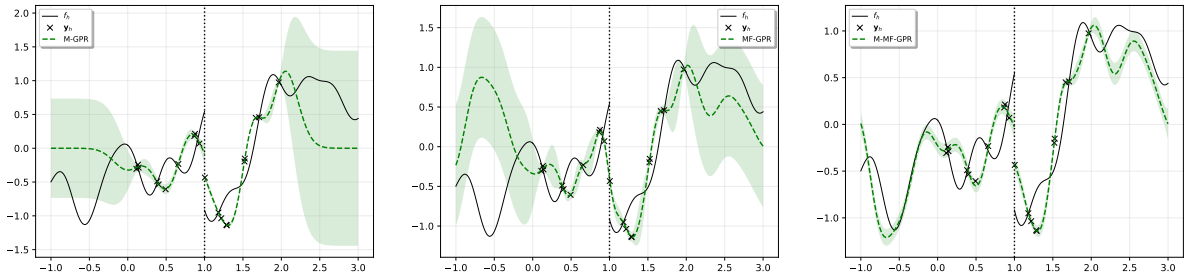
(a) M-GPR (b) MF-GPR (c) M-MF-GPR

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

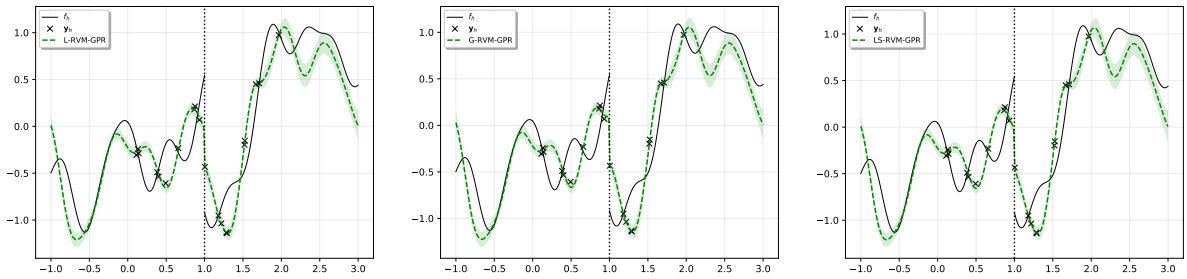
(g) L-CB (h) G-CB (i) LS-CB

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

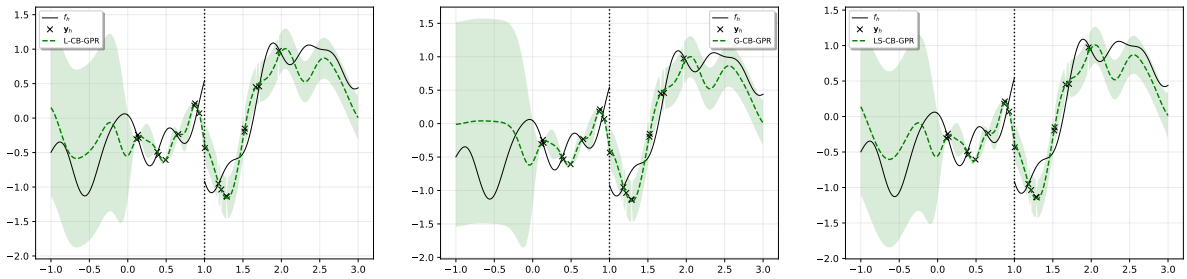
Appendix A: Experimental Results



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



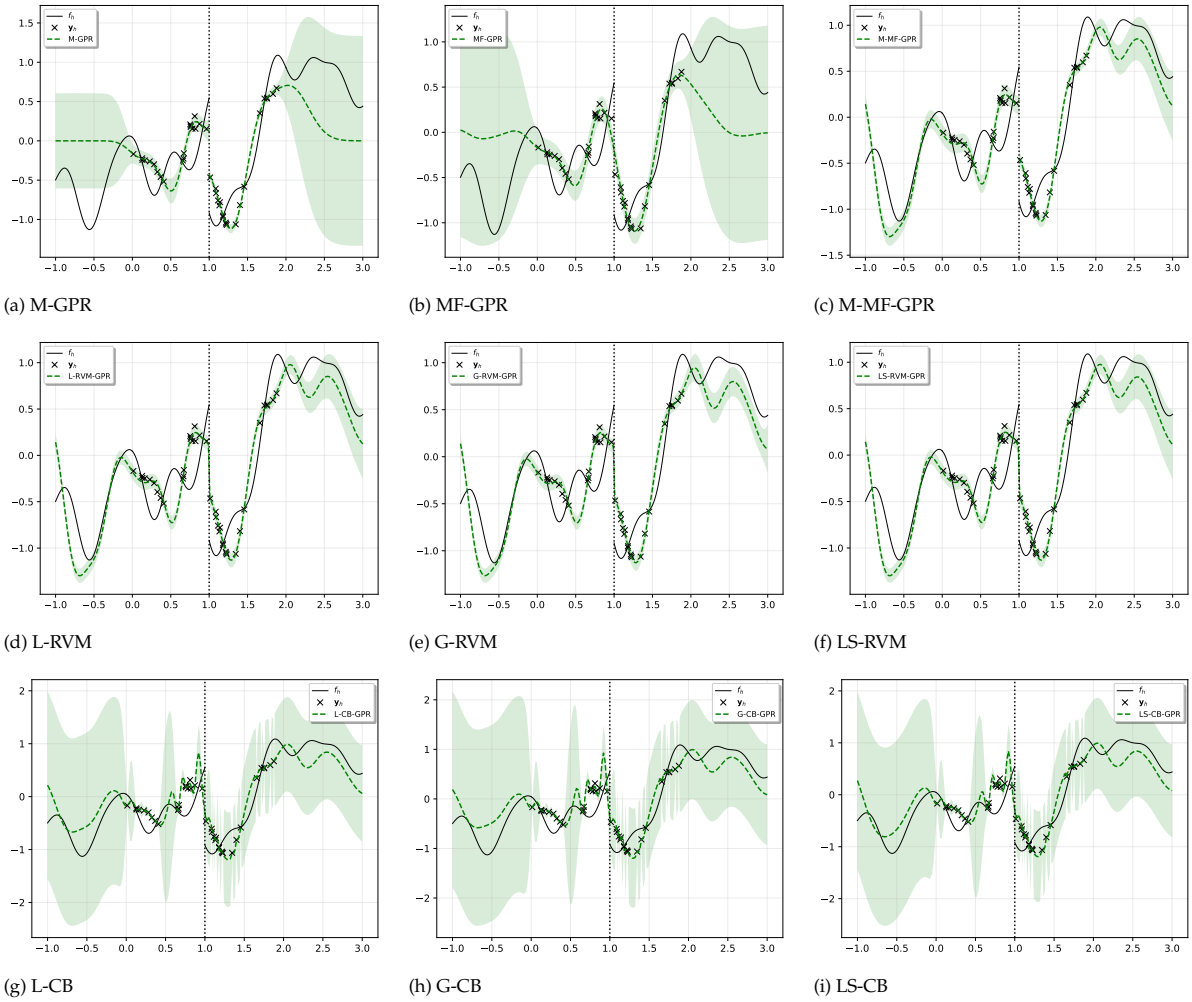
(d) L-RVM (e) G-RVM (f) LS-RVM



(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



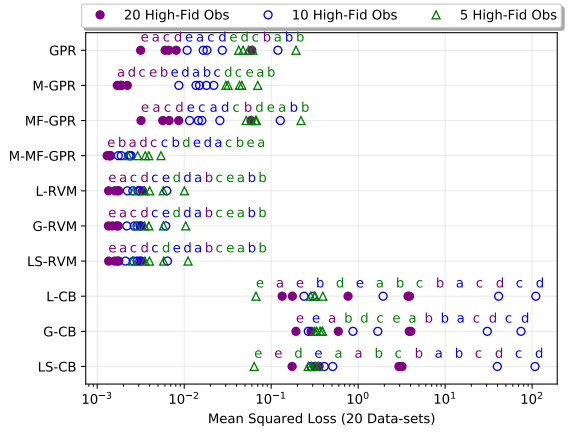
(a) M-GPR (b) MF-GPR (c) M-MF-GPR

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

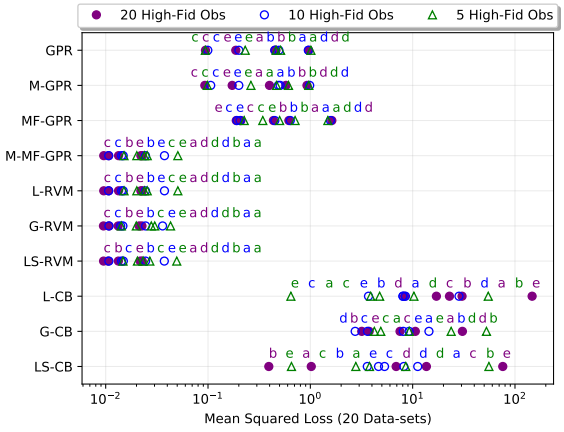
(g) L-CB (h) G-CB (i) LS-CB

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

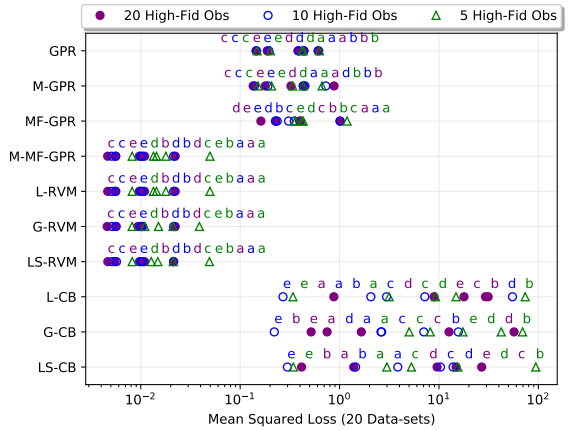
Appendix A: Experimental Results



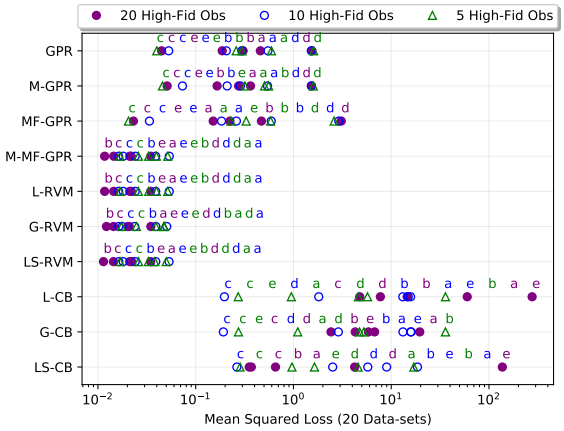
(a) Interpolation regime: $[0,2]$.



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



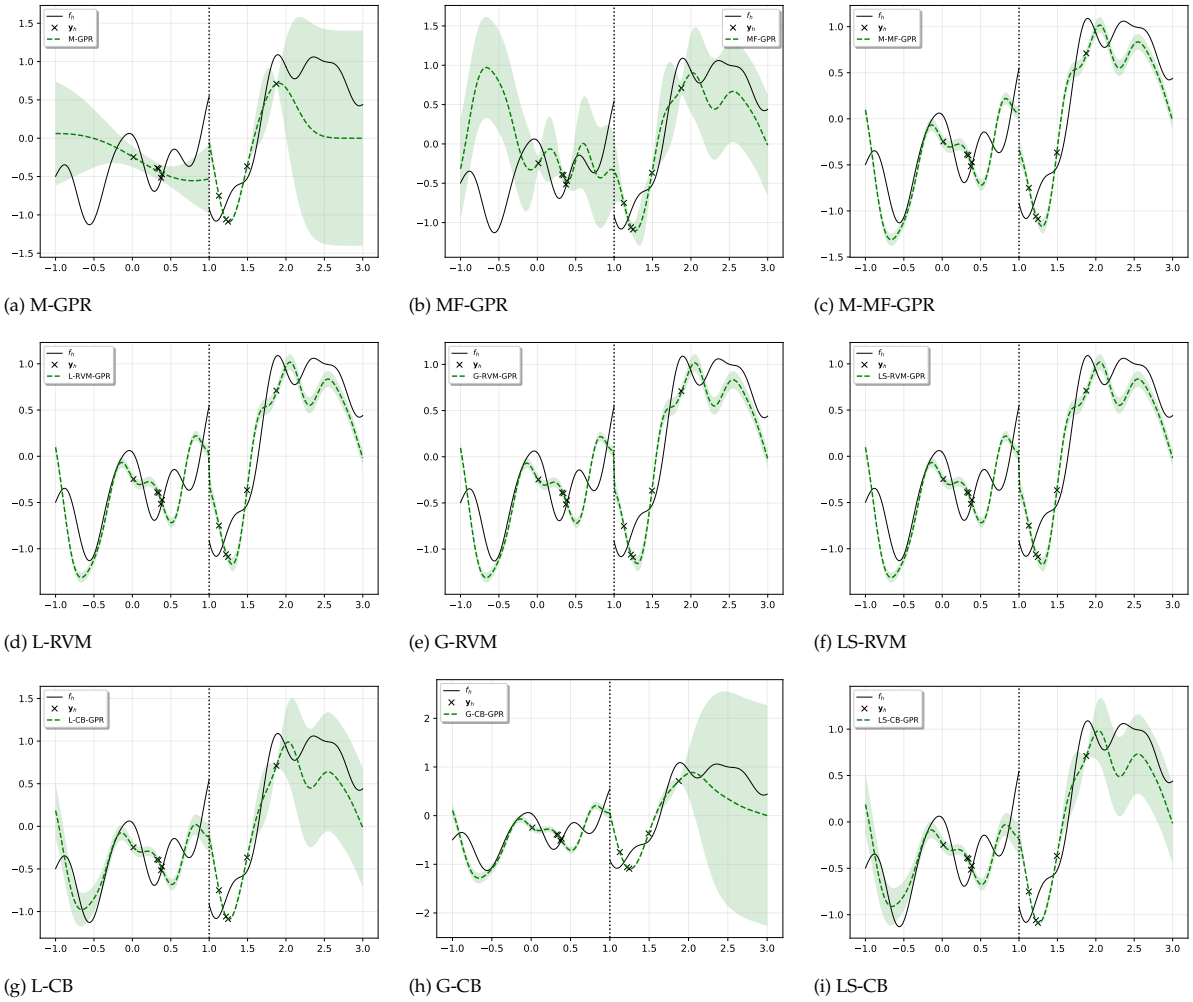
(c) Left extrapolation regime: $[-1,0]$.



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

Figure A.13: discontinuous ρ case, uniformly distributed, and 101 low-fids per region.

Appendix A: Experimental Results



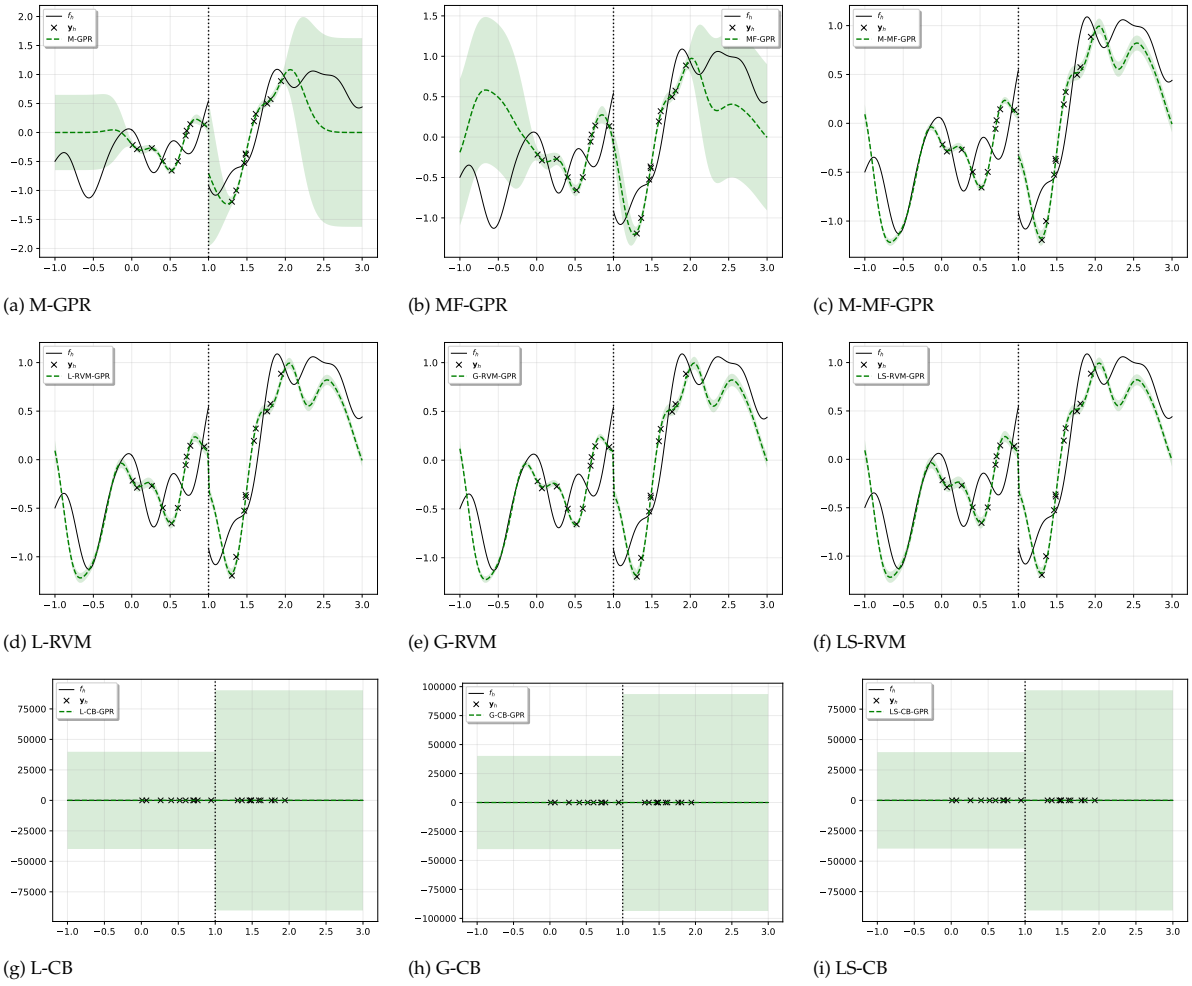
(a) M-GPR (b) MF-GPR (c) M-MF-GPR

(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



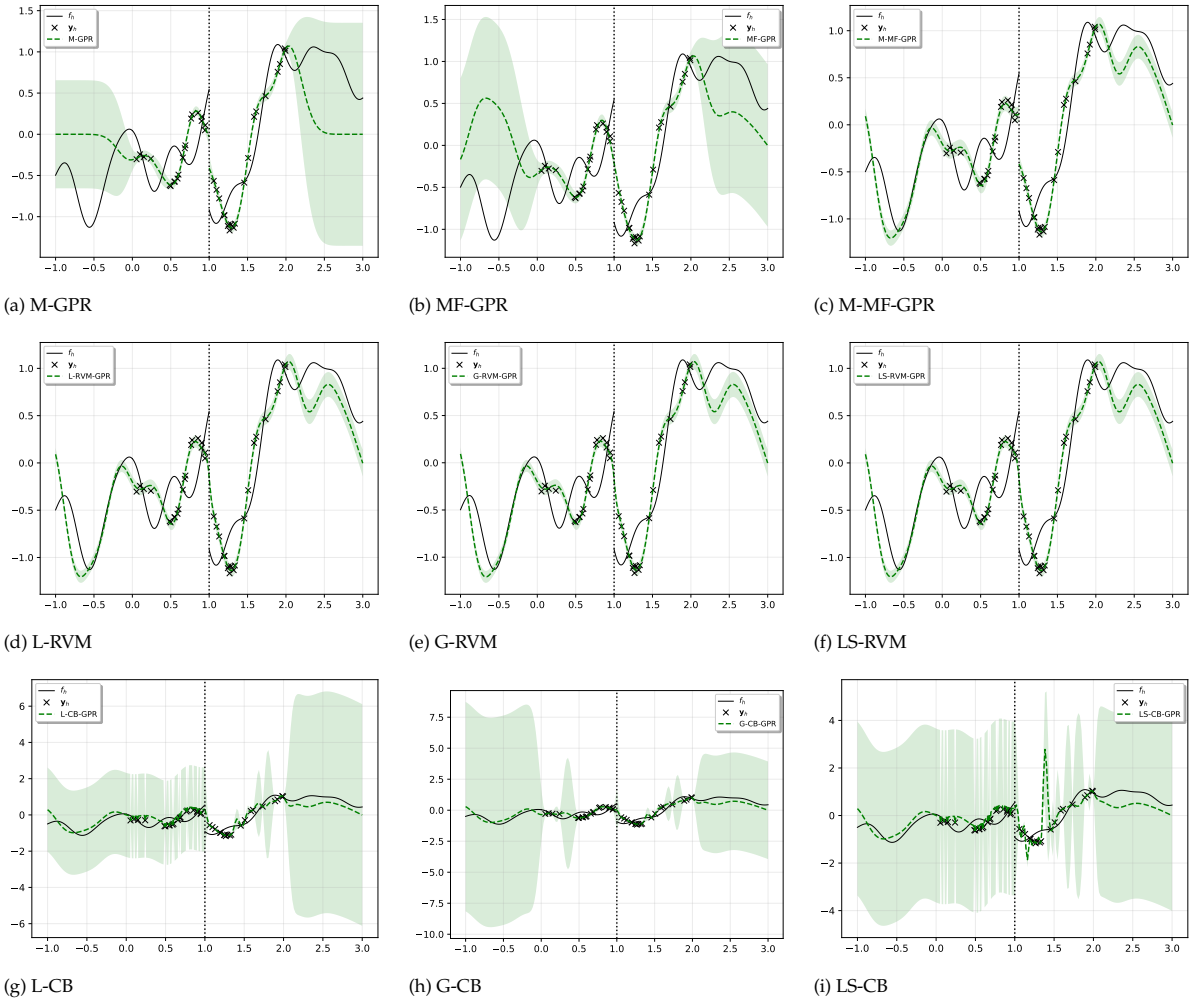
(a) M-GPR (b) MF-GPR (c) M-MF-GPR

(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

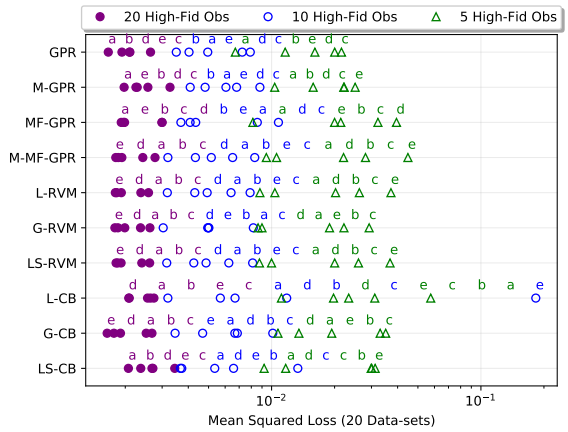
(d) L-RVM (e) G-RVM (f) LS-RVM

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

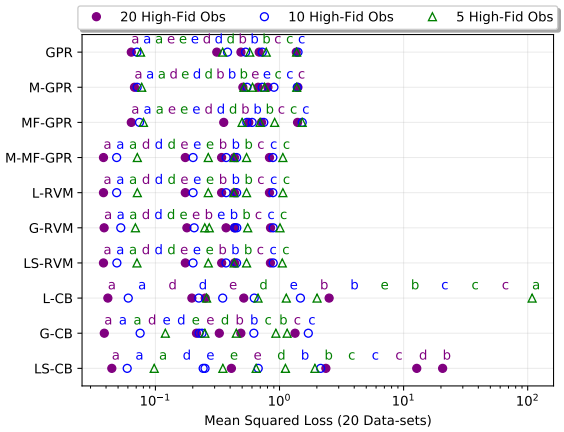
Figure A.16: model predictions with 20 high-fids per region of function **a** and data-set 0 of figure A.13.

A.2.3. Linearly varying ρ case

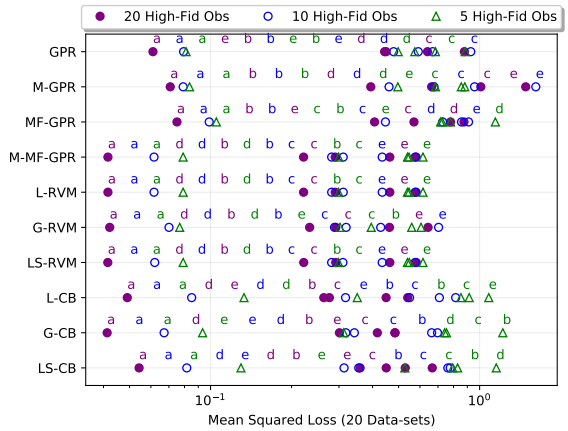
Appendix A: Experimental Results



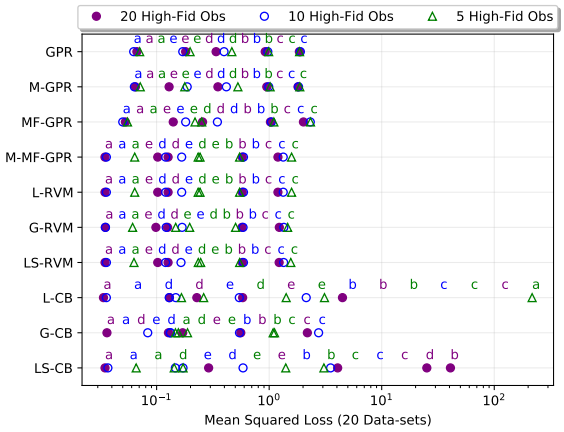
(a) Interpolation regime: $[0,2]$.



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

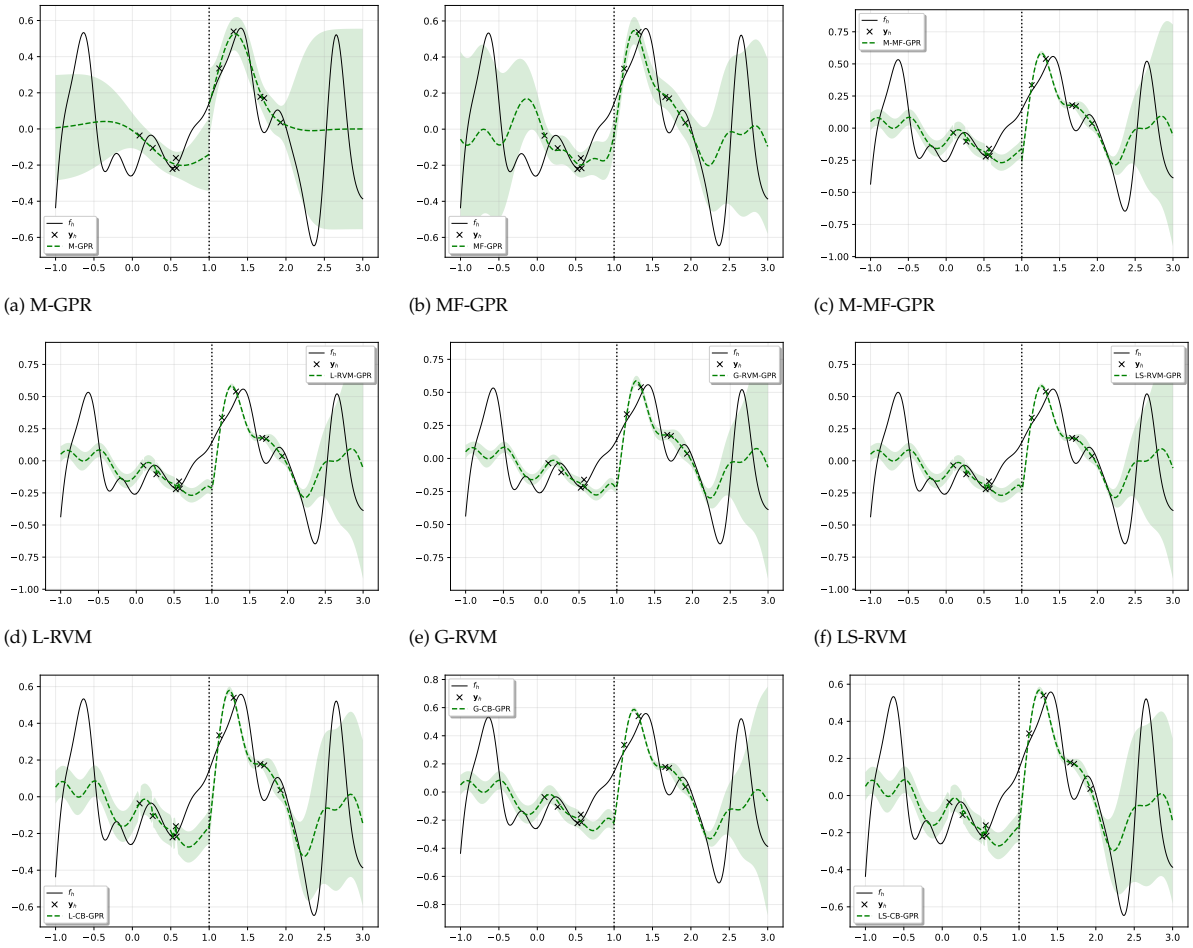


(c) Left extrapolation regime: $[-1,0]$.



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

Figure A.17: linearly varying ρ case, uniformly distributed, and 21 low-fids per region.



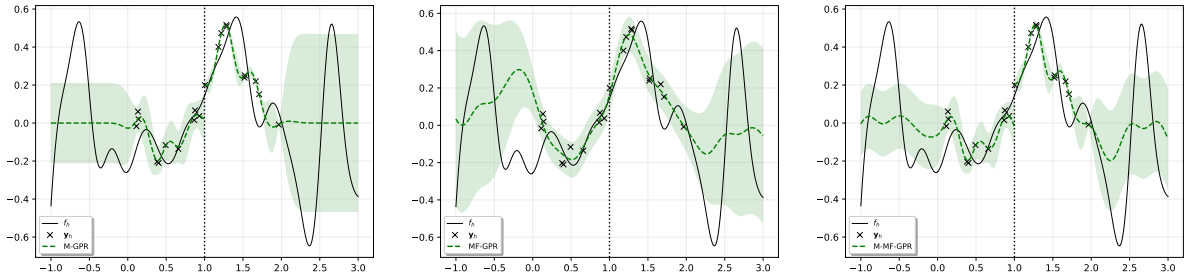
(a) M-GPR (b) MF-GPR (c) M-MF-GPR

(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.

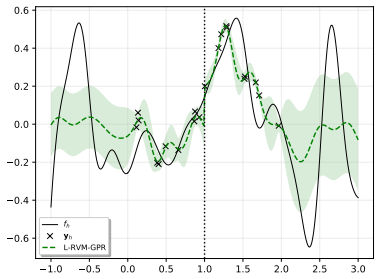
Appendix A: Experimental Results



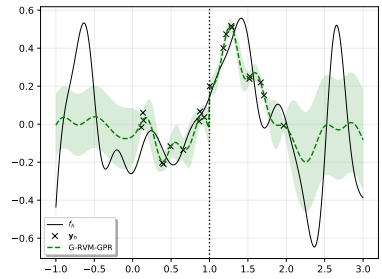
(a) M-GPR

(b) MF-GPR

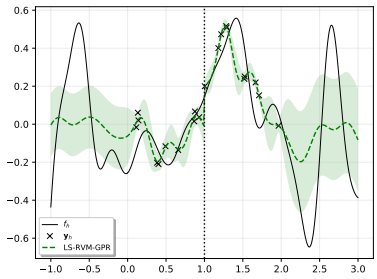
(c) M-MF-GPR



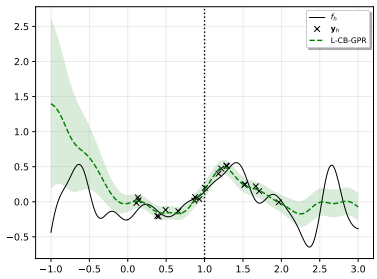
(d) L-RVM



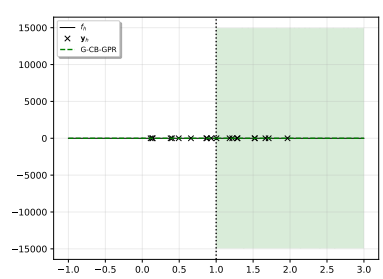
(e) G-RVM



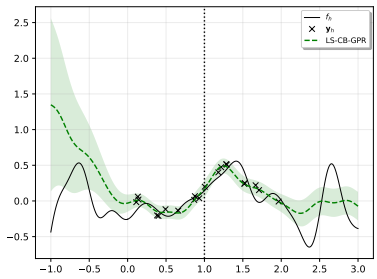
(f) LS-RVM



(g) L-CB

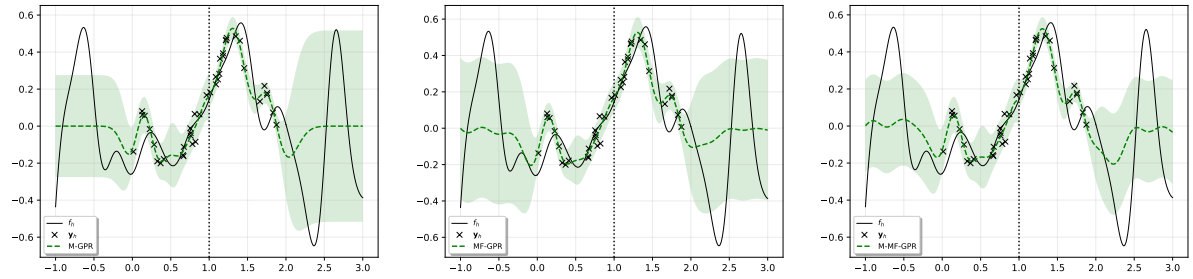


(h) G-CB



(i) LS-CB

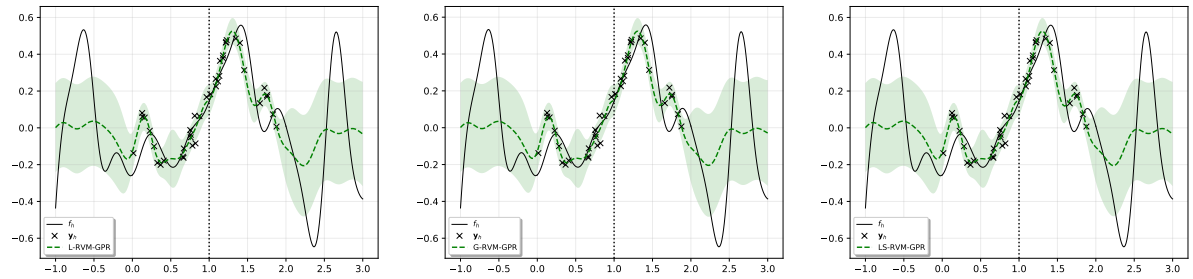
Figure A.19: model predictions with 10 high-fids per region of function **a** and data-set 0 of figure A.17.



(a) M-GPR

(b) MF-GPR

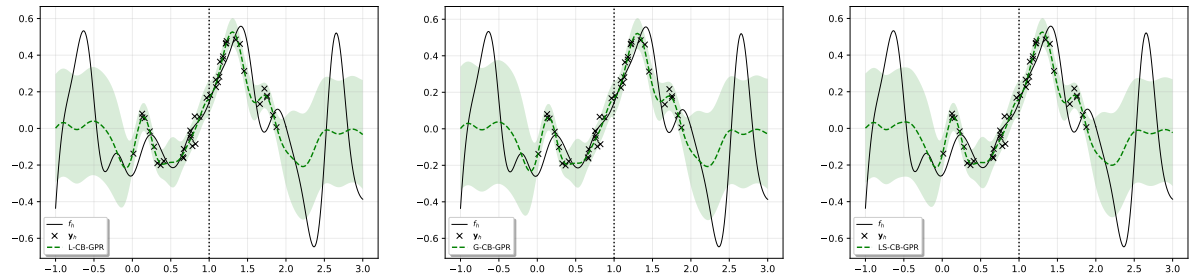
(c) M-MF-GPR



(d) L-RVM

(e) G-RVM

(f) LS-RVM



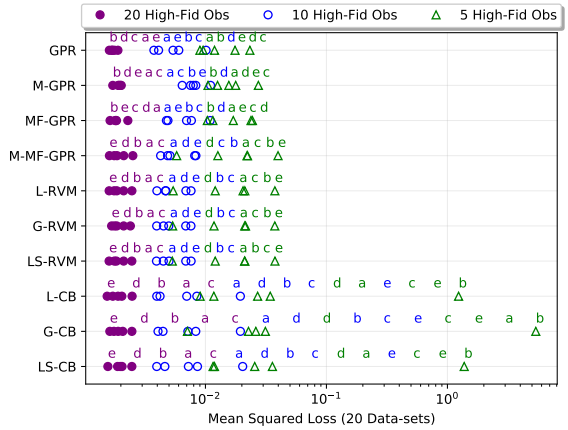
(g) L-CB

(h) G-CB

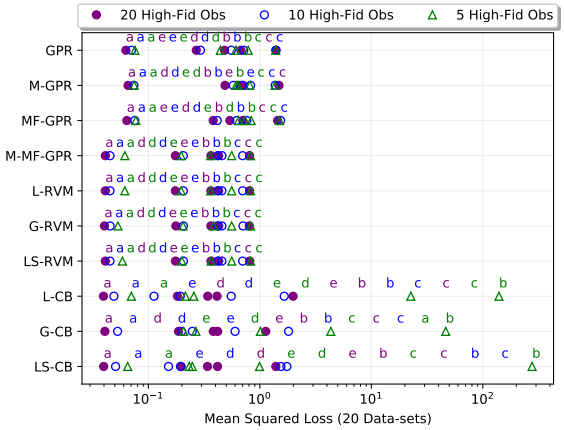
(i) LS-CB

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

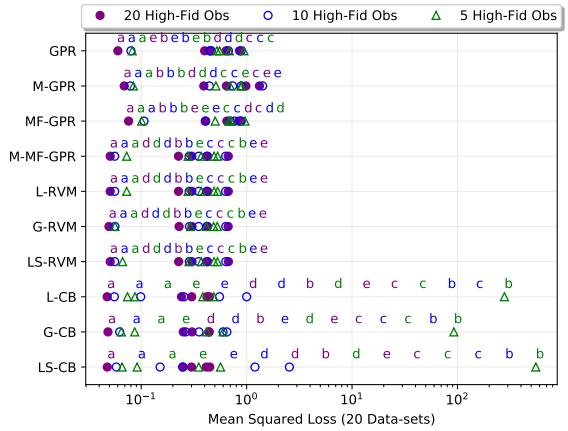
Appendix A: Experimental Results



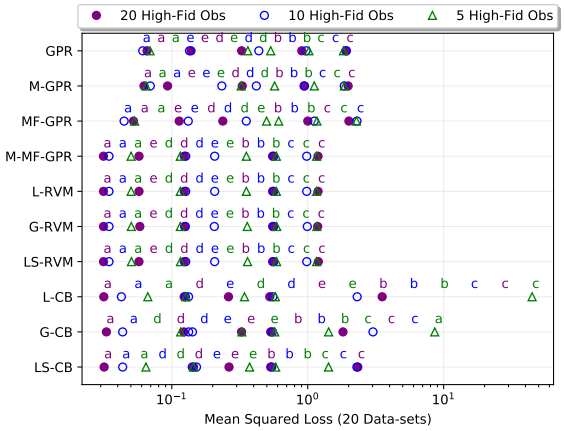
(a) Interpolation regime: $[0,2]$.



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



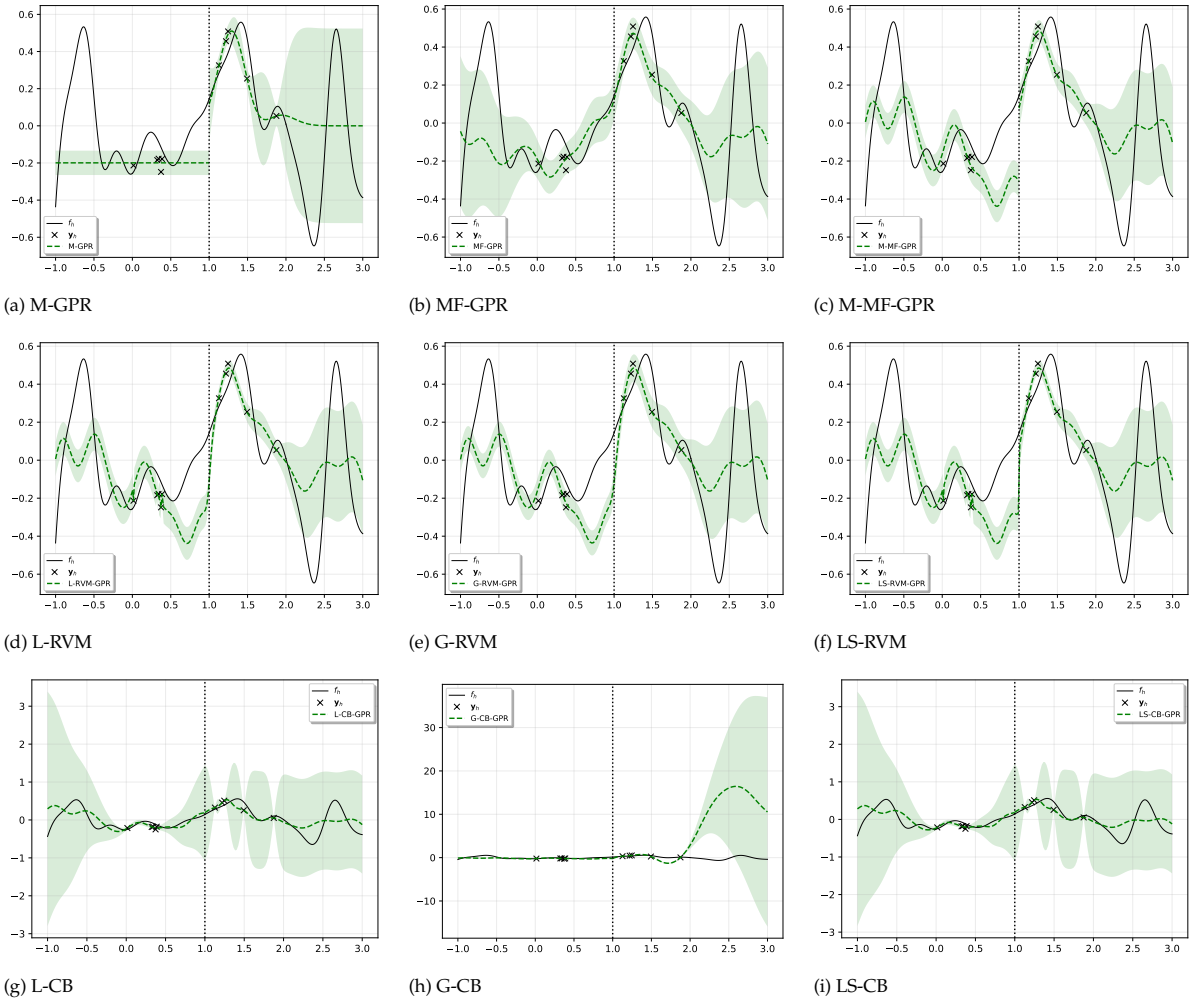
(c) Left extrapolation regime: $[-1,0]$.



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

Figure A.21: linearly varying ρ case, uniformly distributed, and 101 low-fids per region.

Appendix A: Experimental Results



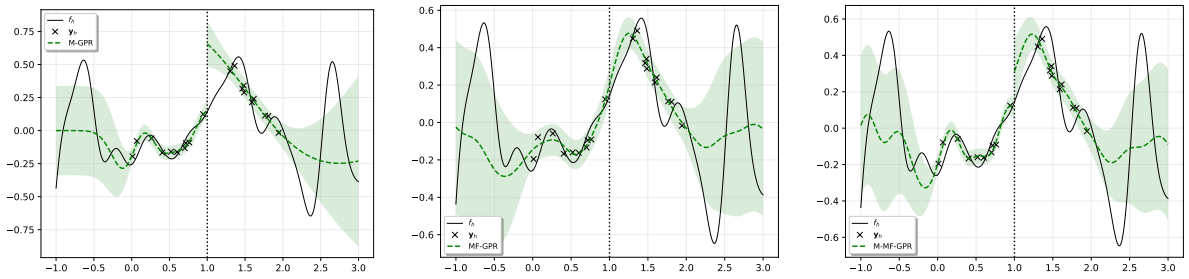
(a) M-GPR (b) MF-GPR (c) M-MF-GPR

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

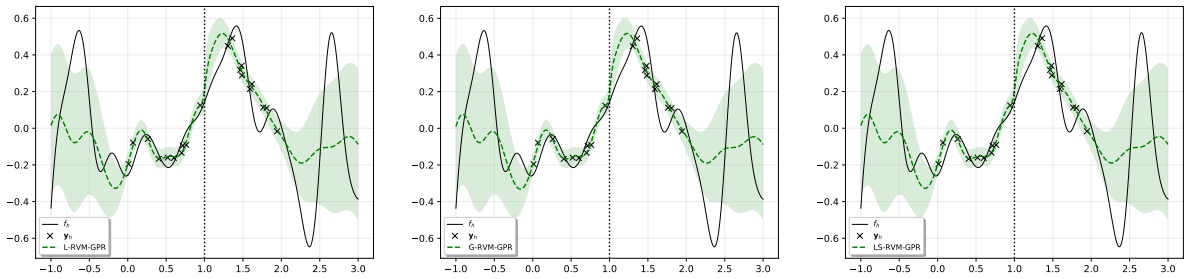
(g) L-CB (h) G-CB (i) LS-CB

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

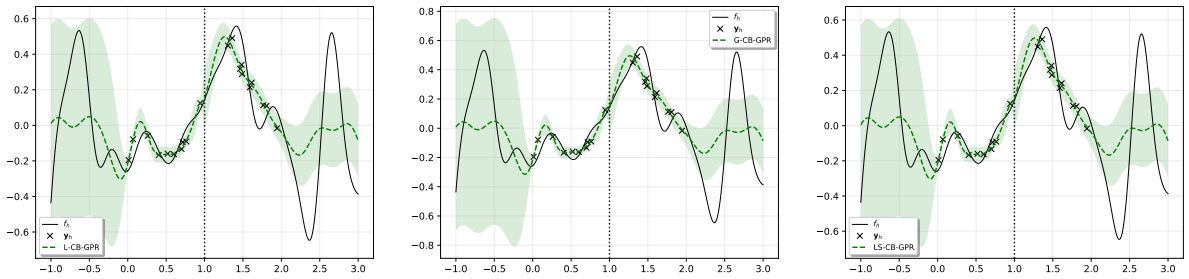
Appendix A: Experimental Results



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

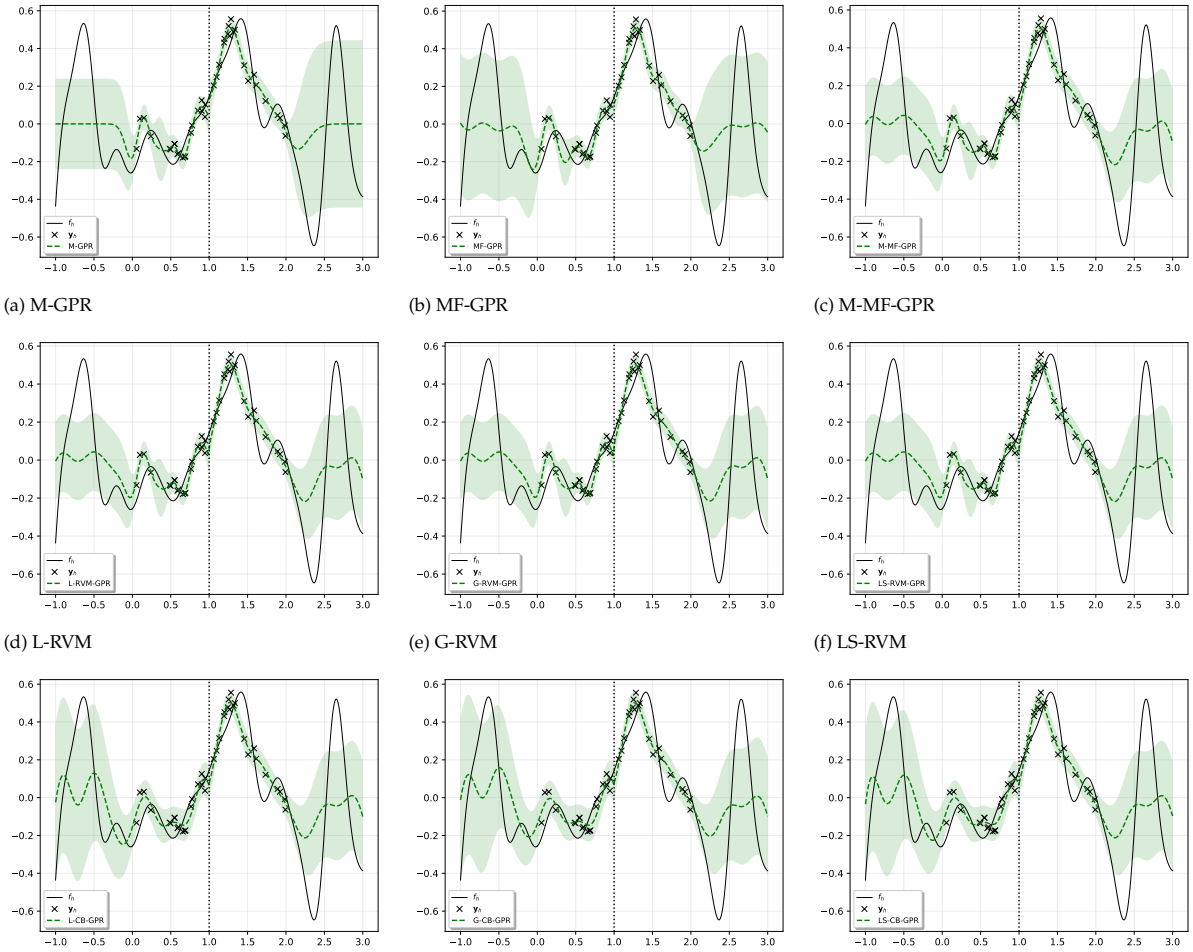


(d) L-RVM (e) G-RVM (f) LS-RVM



(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.



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

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

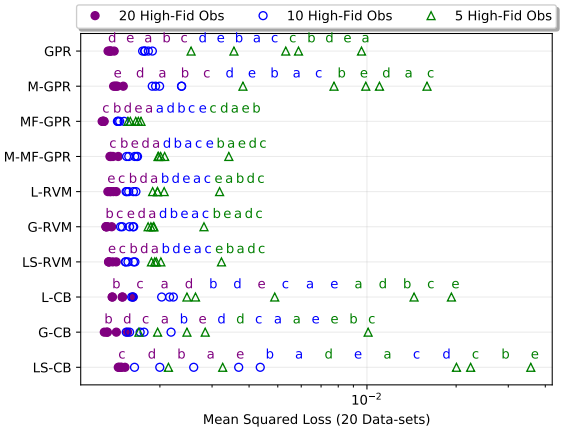
(g) L-CB (h) G-CB (i) LS-CB

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

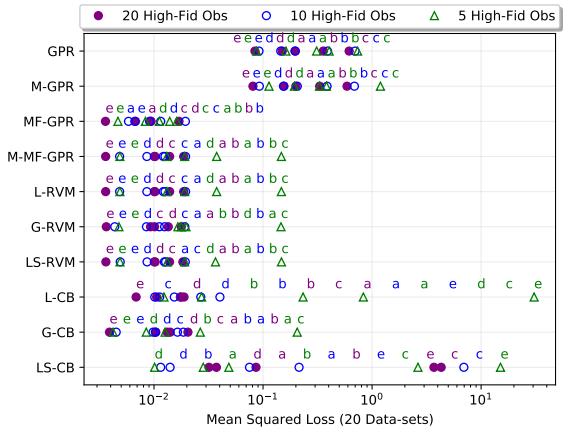
A.3. Linearly Spaced

A.3.1. Constant ρ case

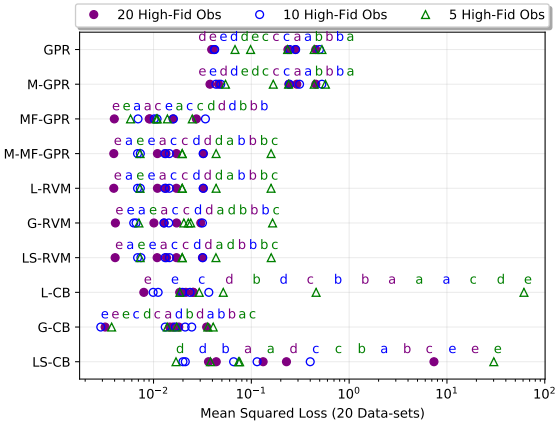
Appendix A: Experimental Results



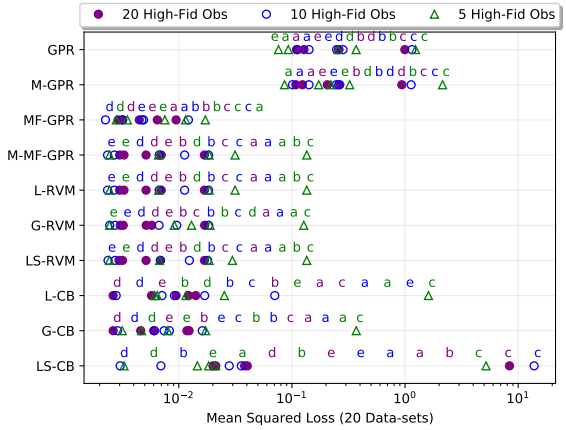
(a) Interpolation regime: $[0,2]$.



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

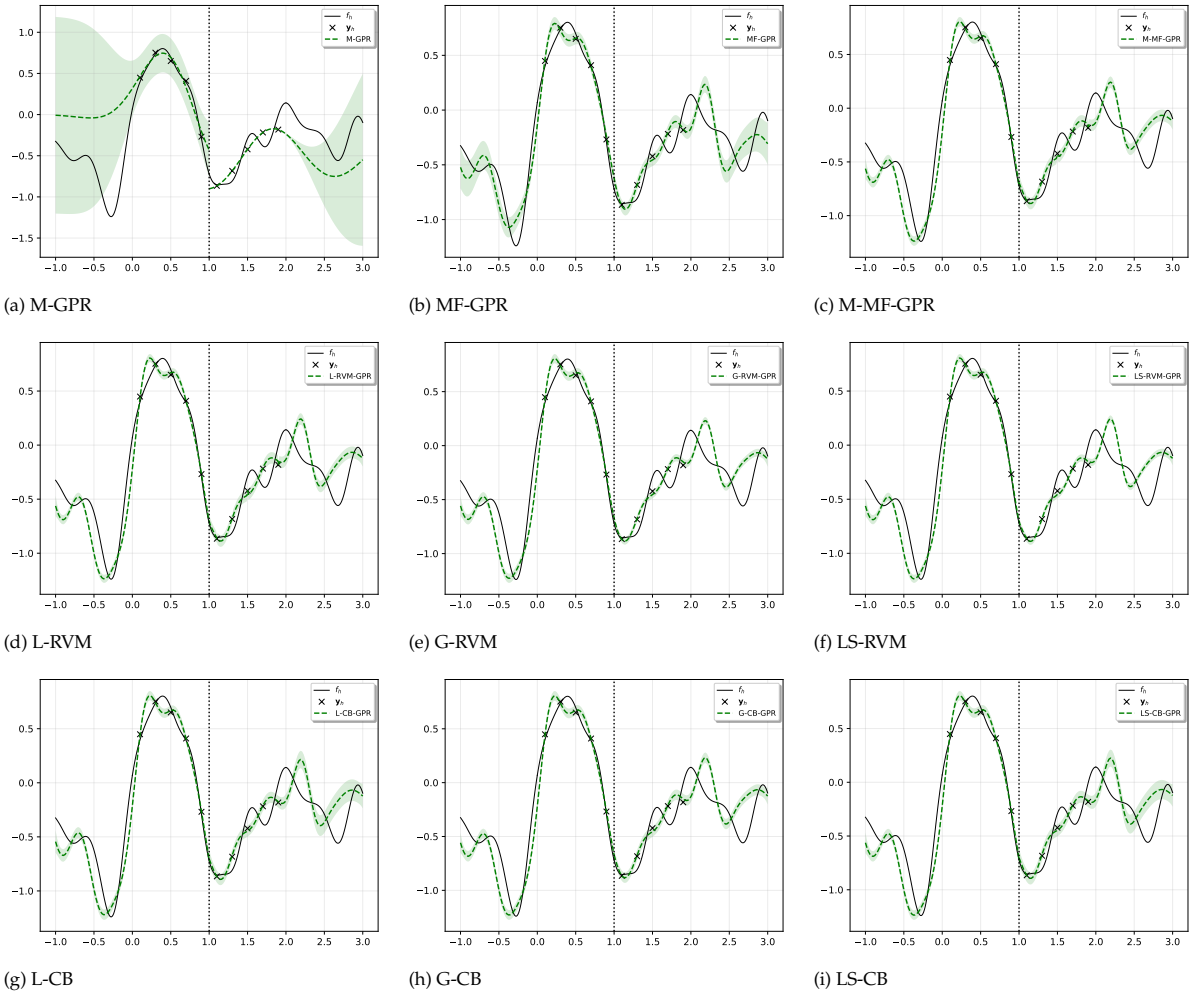


(c) Left extrapolation regime: $[-1,0]$.



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

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



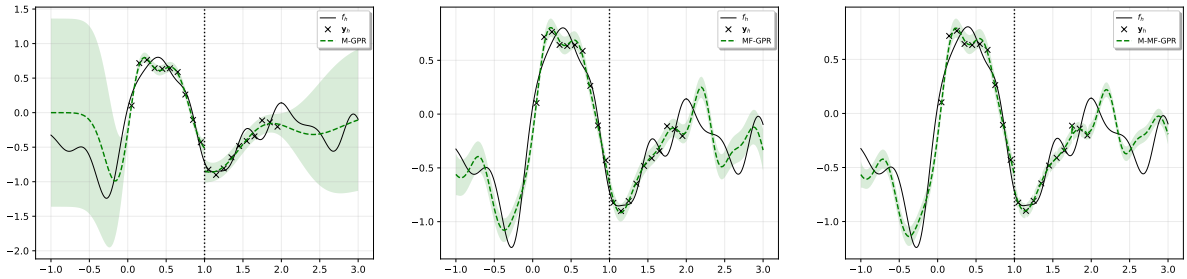
(a) M-GPR (b) MF-GPR (c) M-MF-GPR

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

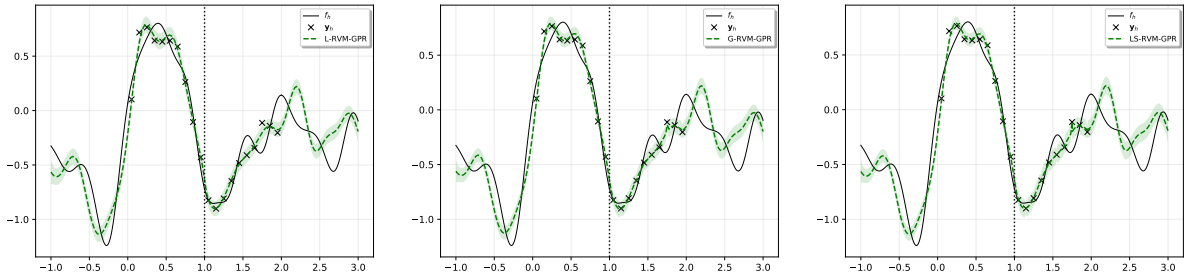
(g) L-CB (h) G-CB (i) LS-CB

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

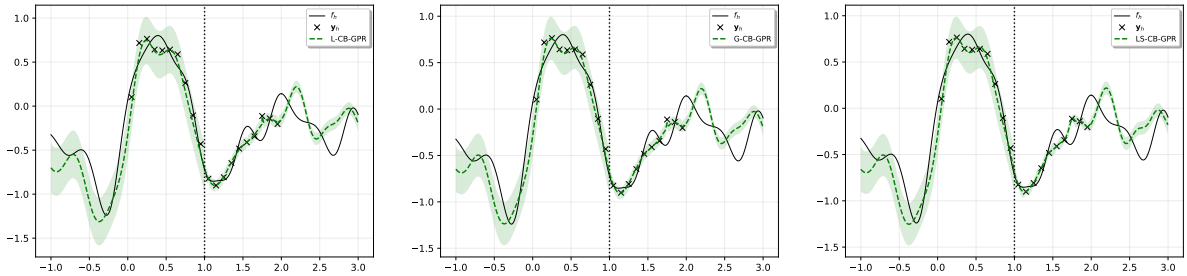
Appendix A: Experimental Results



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



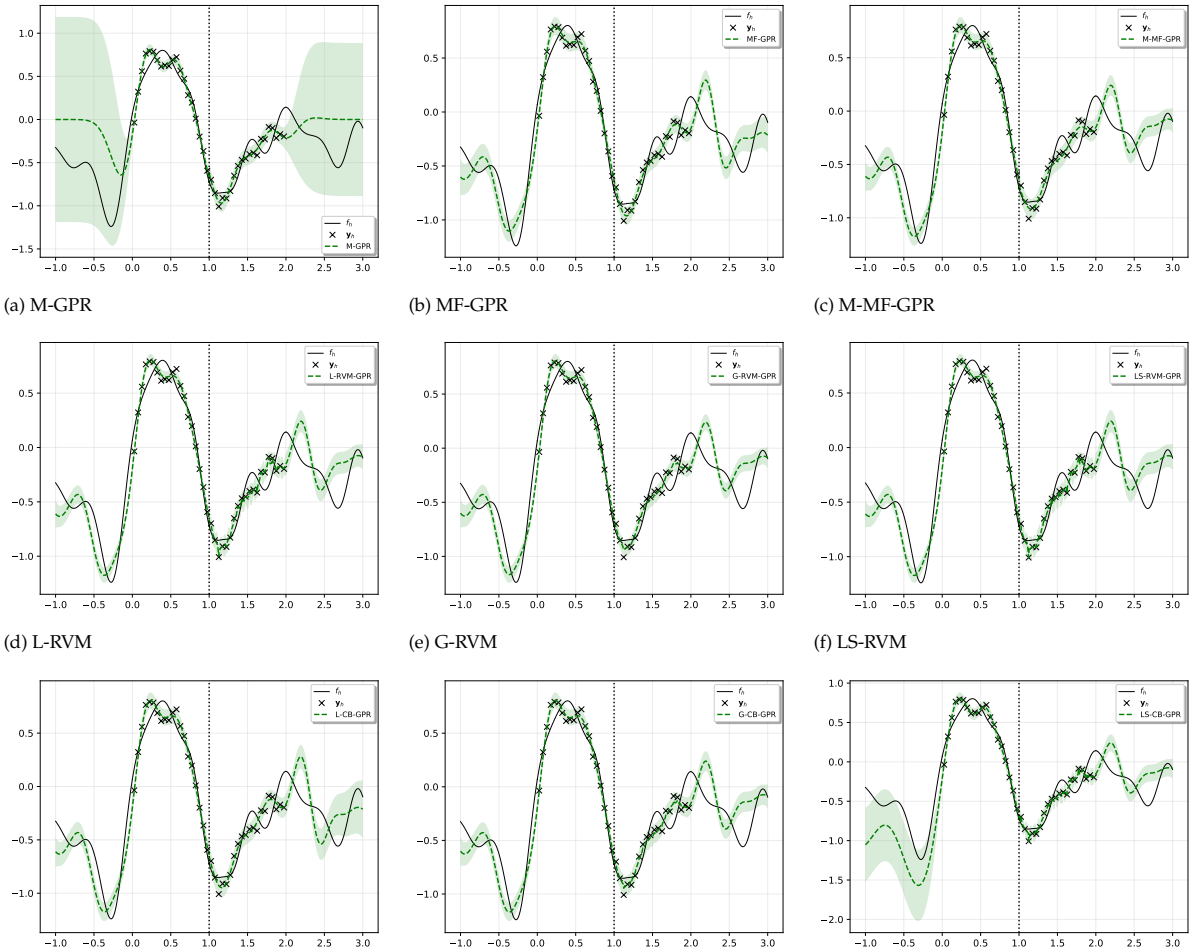
(d) L-RVM (e) G-RVM (f) LS-RVM



(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



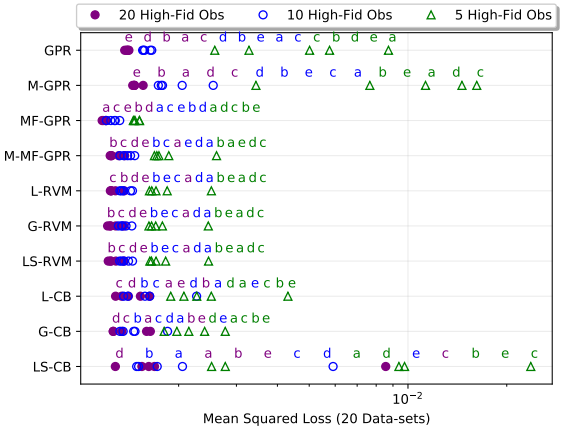
(a) M-GPR (b) MF-GPR (c) M-MF-GPR

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

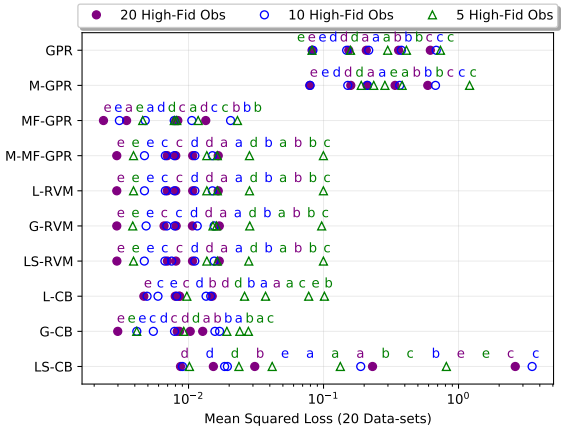
(g) L-CB (h) G-CB (i) LS-CB

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

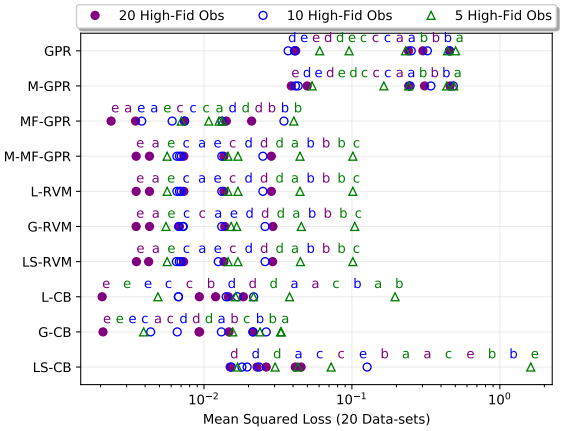
Appendix A: Experimental Results



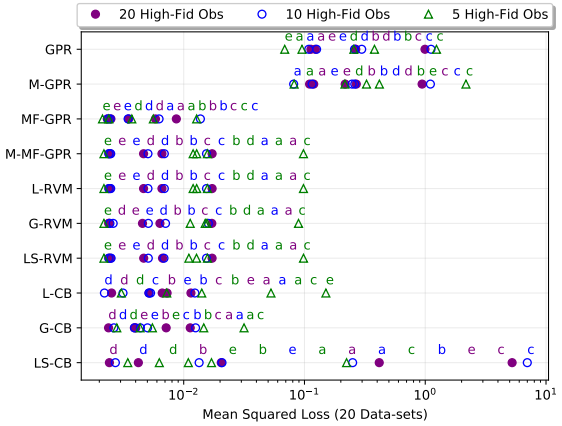
(a) Interpolation regime: $[0,2]$.



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



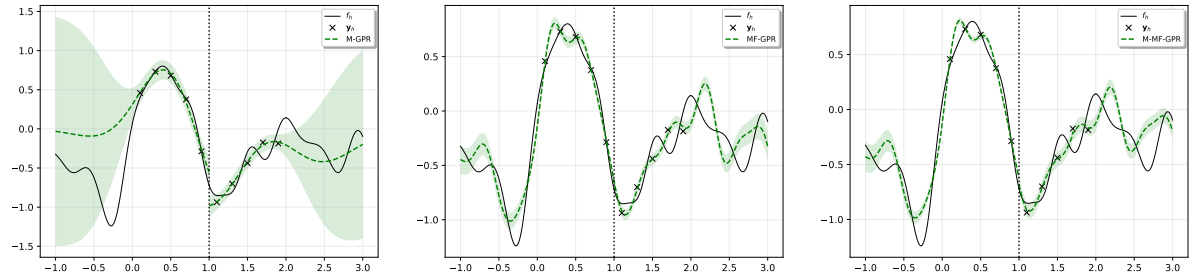
(c) Left extrapolation regime: $[-1,0]$.



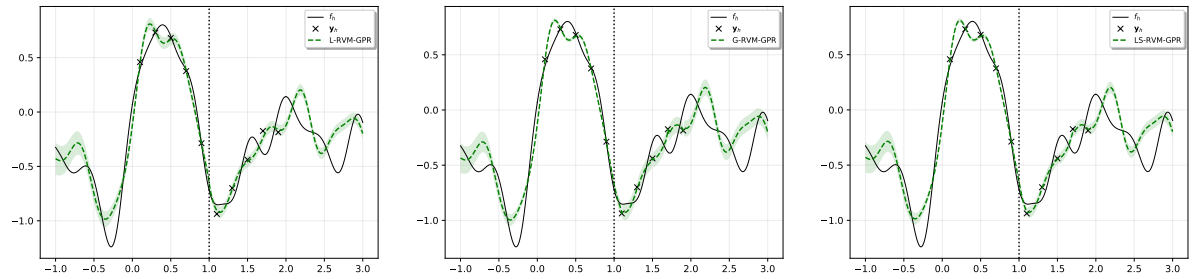
(d) Right extrapolation regime: $[2,3]$.

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

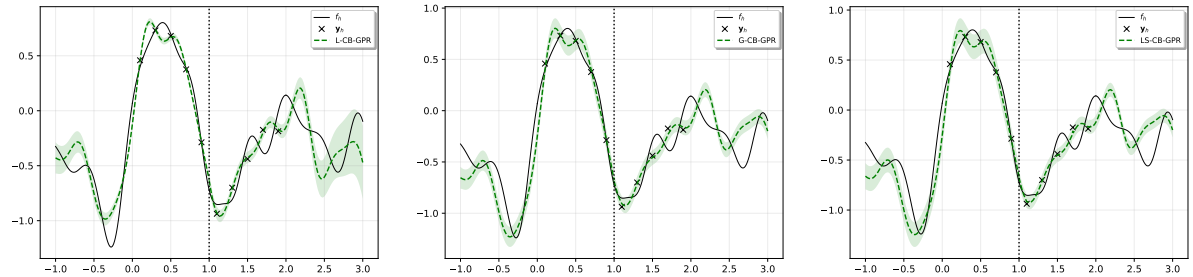
Appendix A: Experimental Results



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



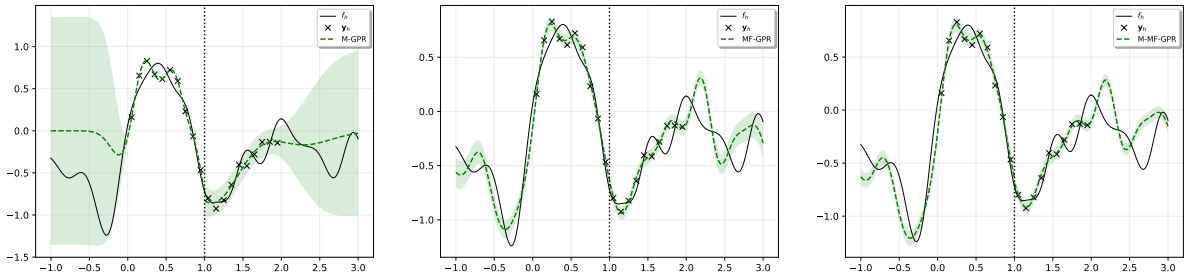
(d) L-RVM (e) G-RVM (f) LS-RVM



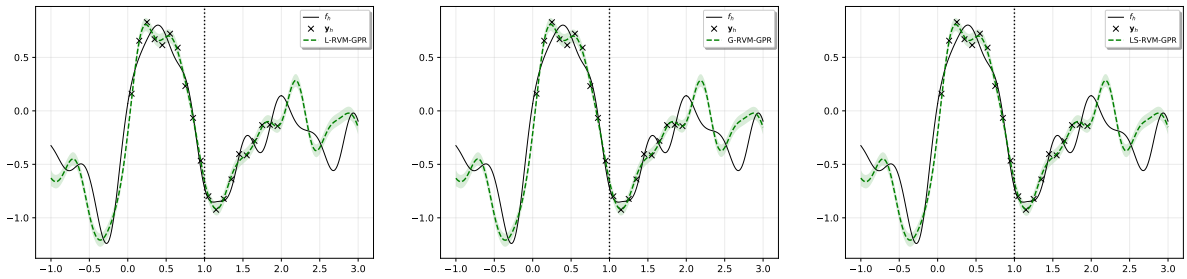
(g) L-CB (h) G-CB (i) LS-CB

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

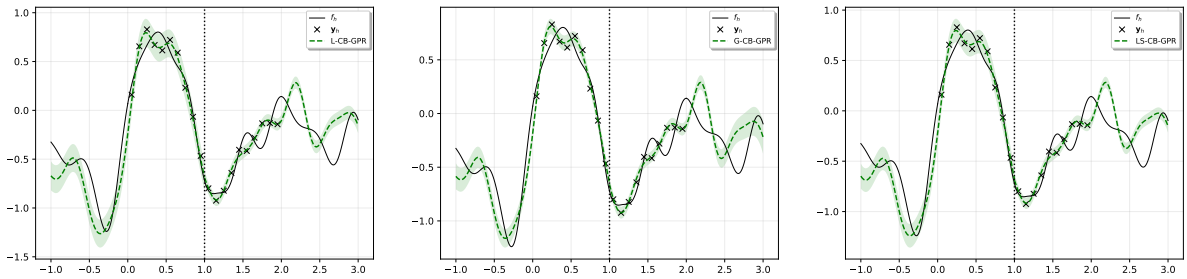
Appendix A: Experimental Results



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



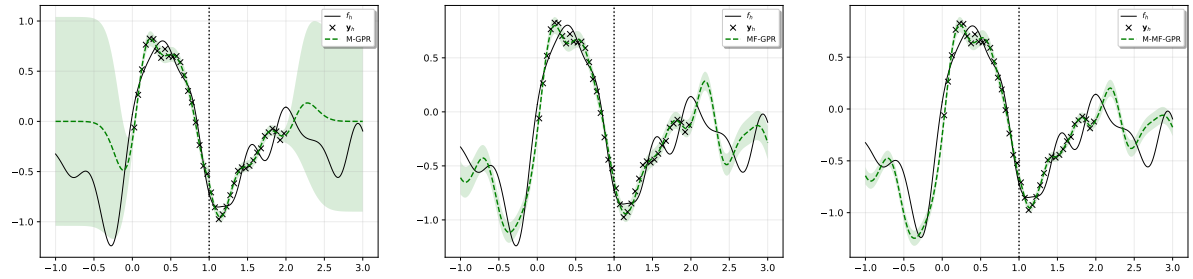
(d) L-RVM (e) G-RVM (f) LS-RVM



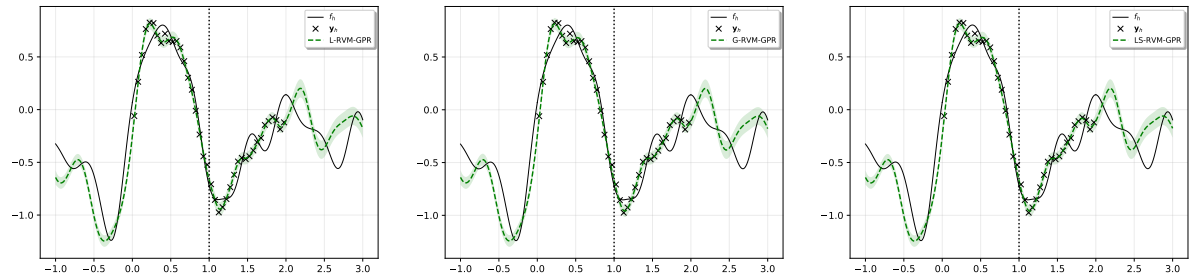
(g) L-CB (h) G-CB (i) LS-CB

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

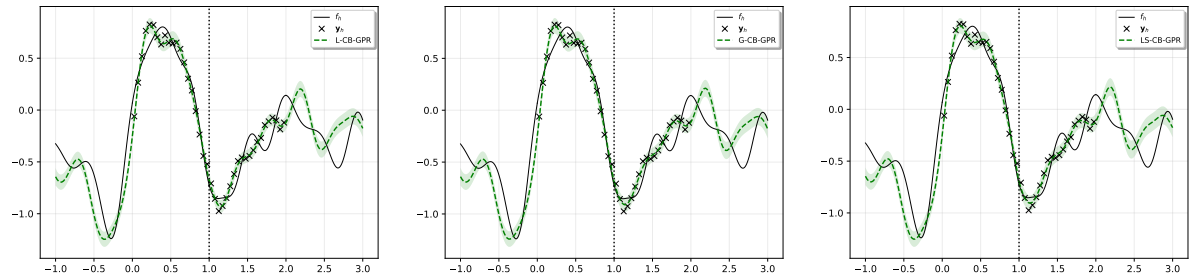
Appendix A: Experimental Results



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



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

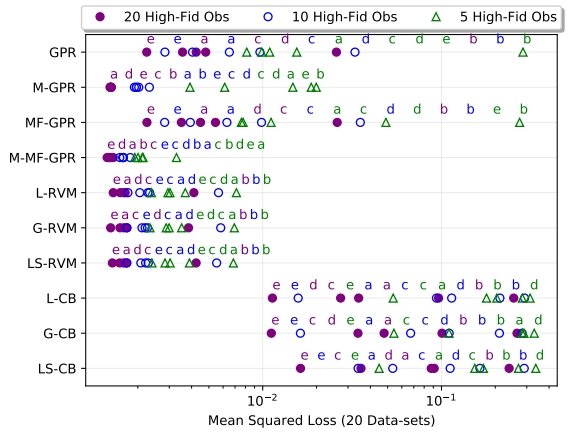


(g) L-CB (h) G-CB (i) LS-CB

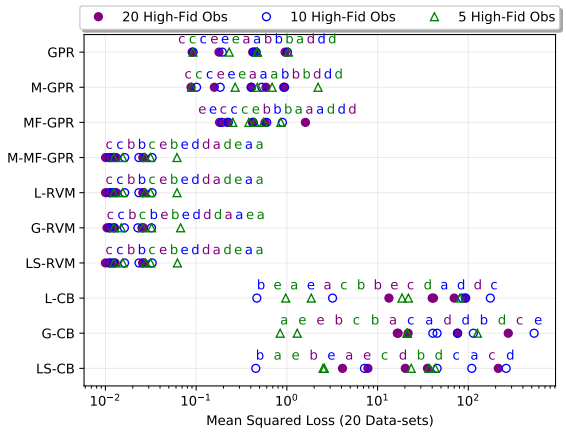
Figure A.32: model predictions with 20 high-fids per region of function **a** and data-set 0 of figure A.29.

A.3.2. Discontinuous ρ case

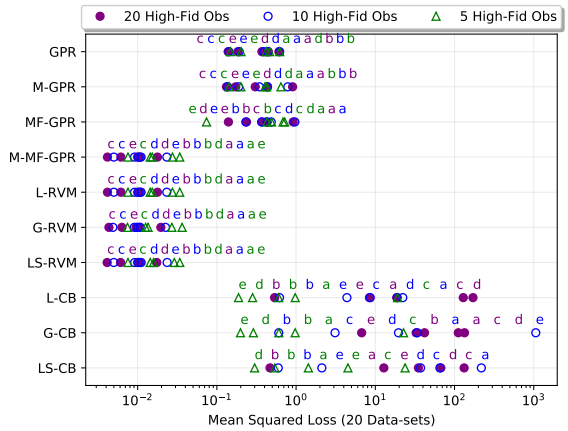
Appendix A: Experimental Results



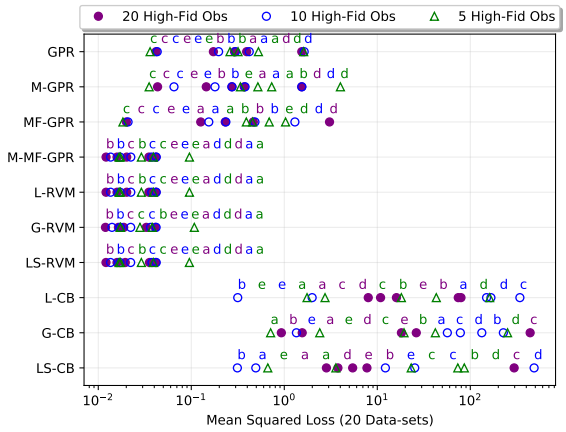
(a) Interpolation regime: $[0,2]$.



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

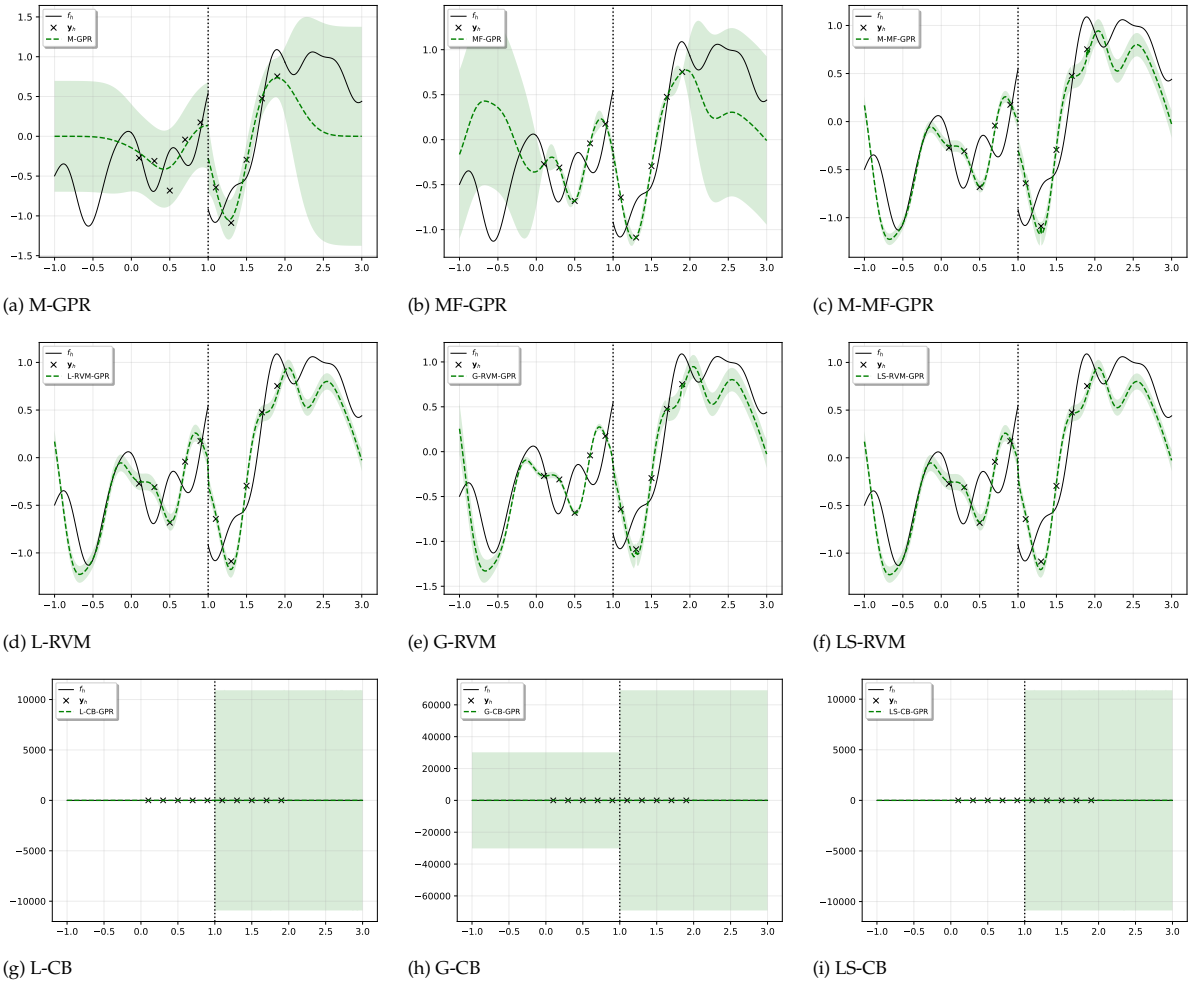


(c) Left extrapolation regime: $[-1,0]$.



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

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



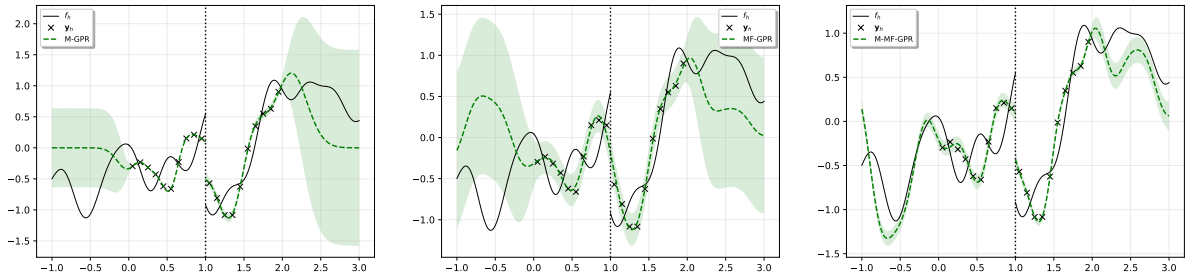
(a) M-GPR (b) MF-GPR (c) M-MF-GPR

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

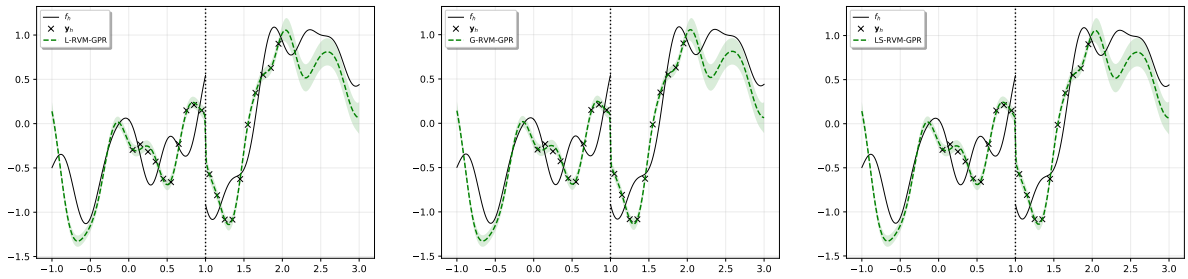
(g) L-CB (h) G-CB (i) LS-CB

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

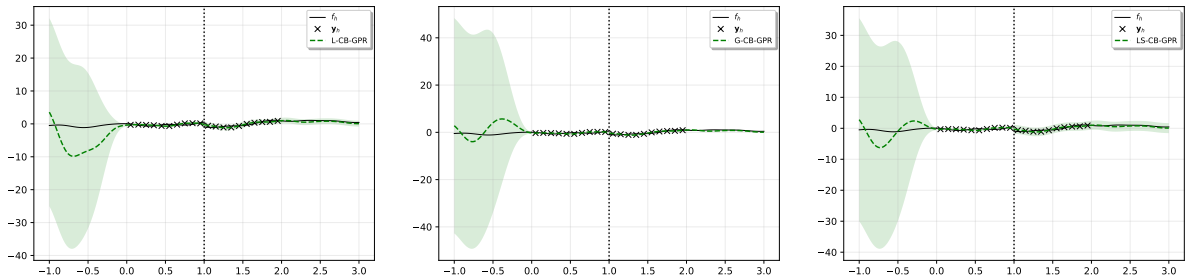
Appendix A: Experimental Results



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



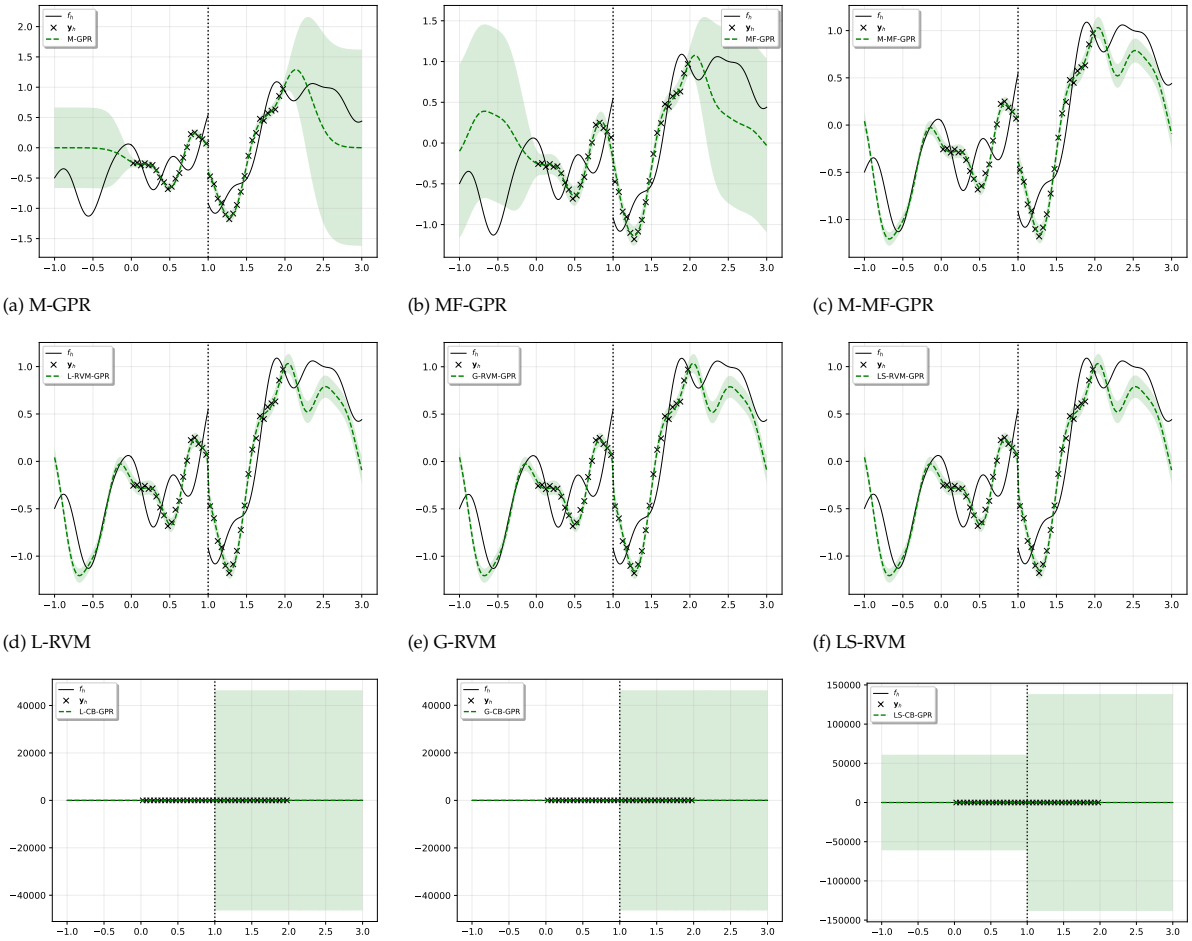
(d) L-RVM (e) G-RVM (f) LS-RVM



(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



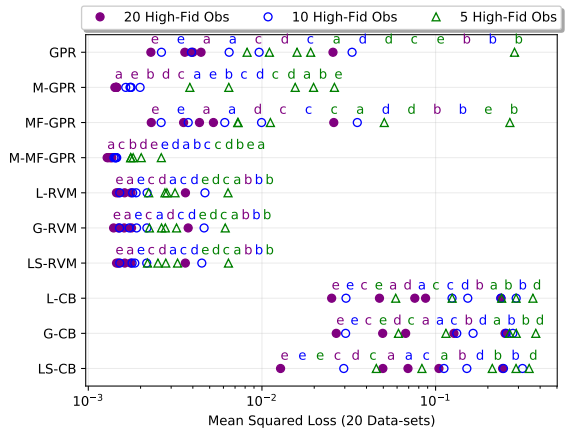
(a) M-GPR (b) MF-GPR (c) M-MF-GPR

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

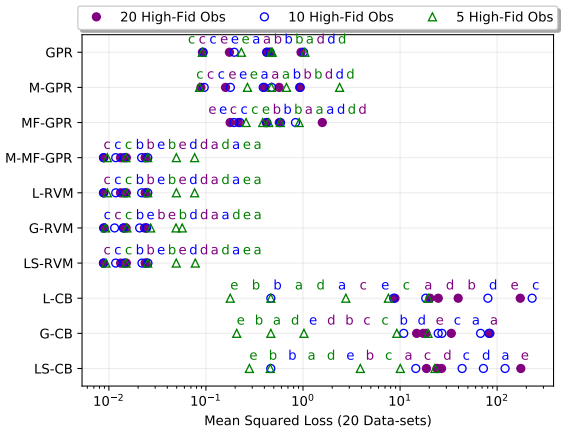
(g) L-CB (h) G-CB (i) LS-CB

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

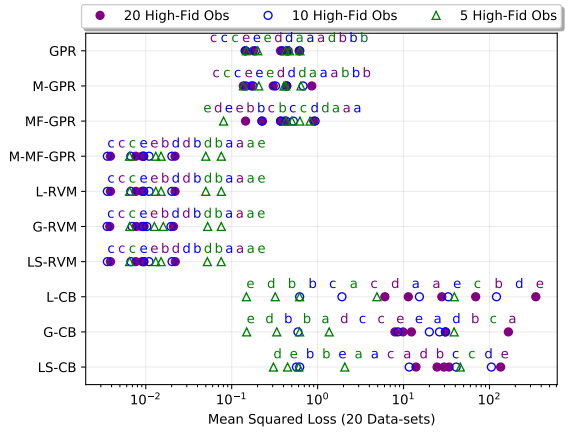
Appendix A: Experimental Results



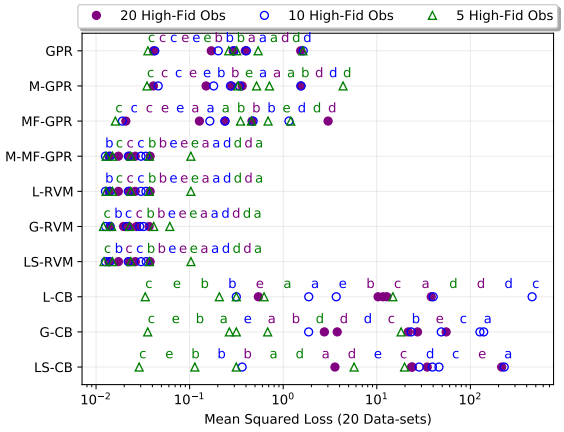
(a) Interpolation regime: $[0,2]$.



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



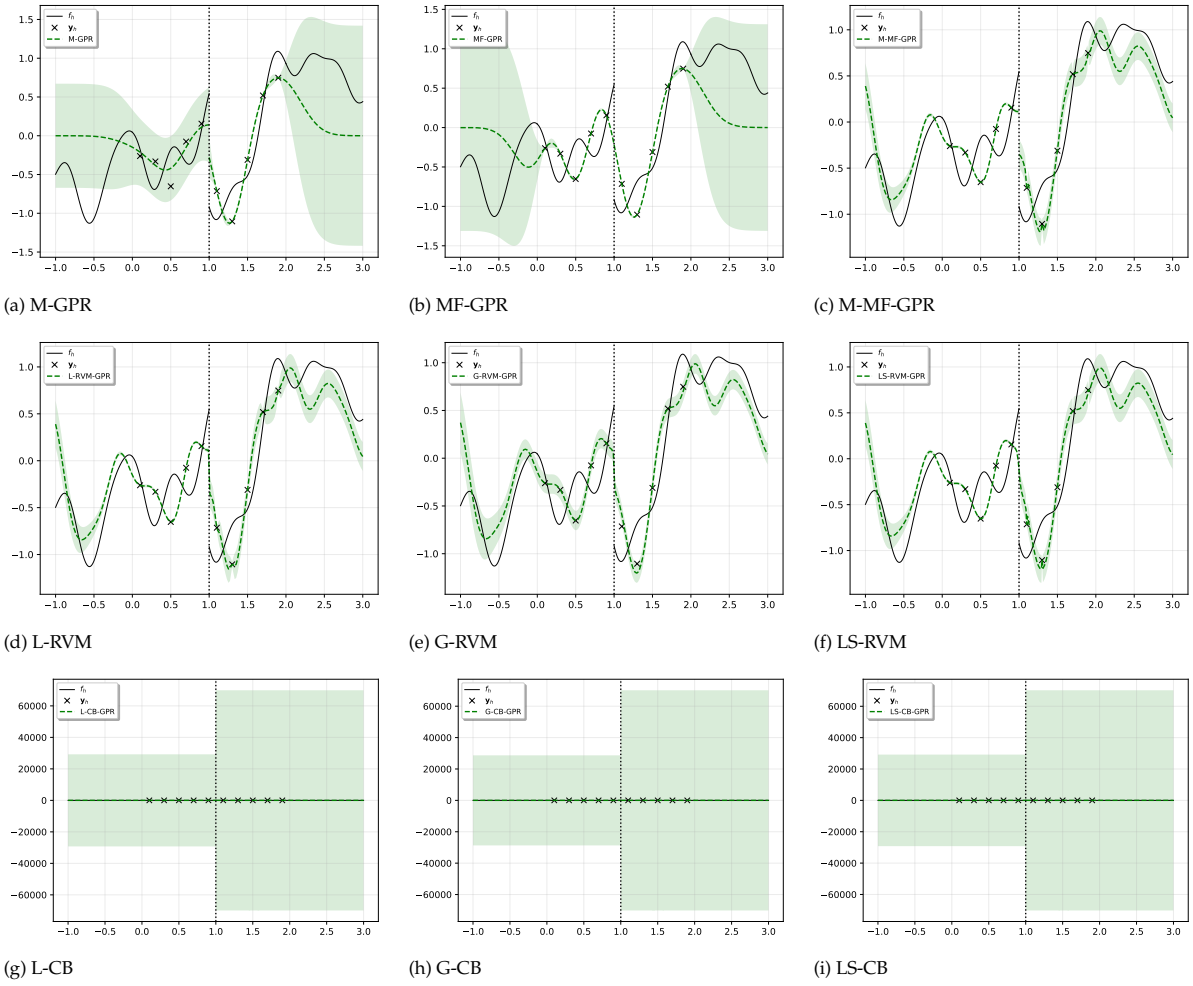
(c) Left extrapolation regime: $[-1,0]$.



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

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

Appendix A: Experimental Results



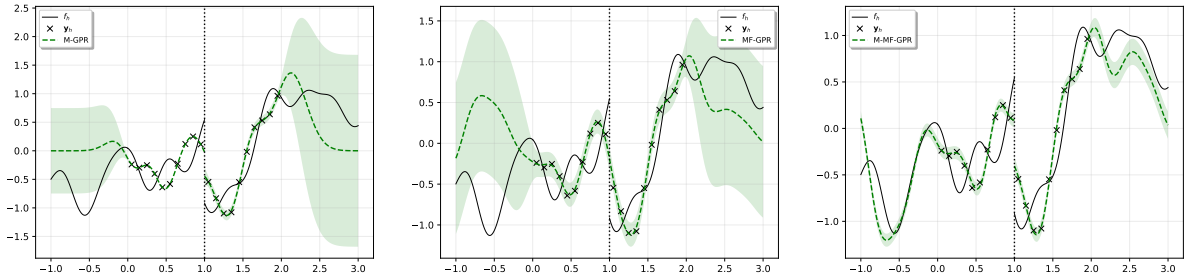
(a) M-GPR (b) MF-GPR (c) M-MF-GPR

(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.

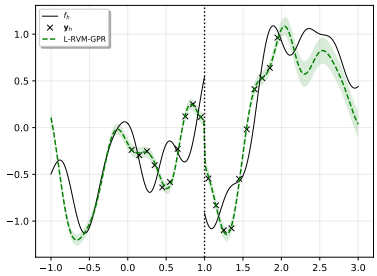
Appendix A: Experimental Results



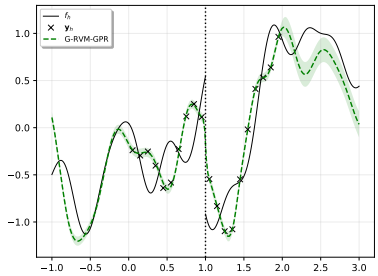
(a) M-GPR

(b) MF-GPR

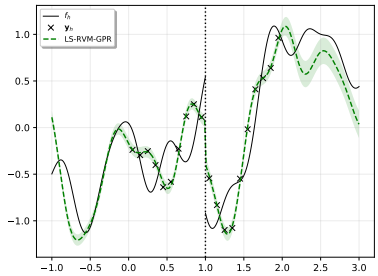
(c) M-MF-GPR



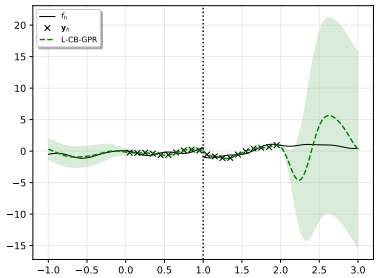
(d) L-RVM



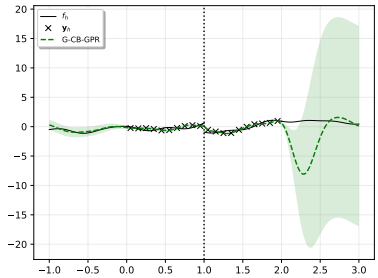
(e) G-RVM



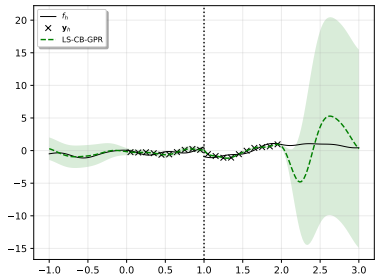
(f) LS-RVM



(g) L-CB



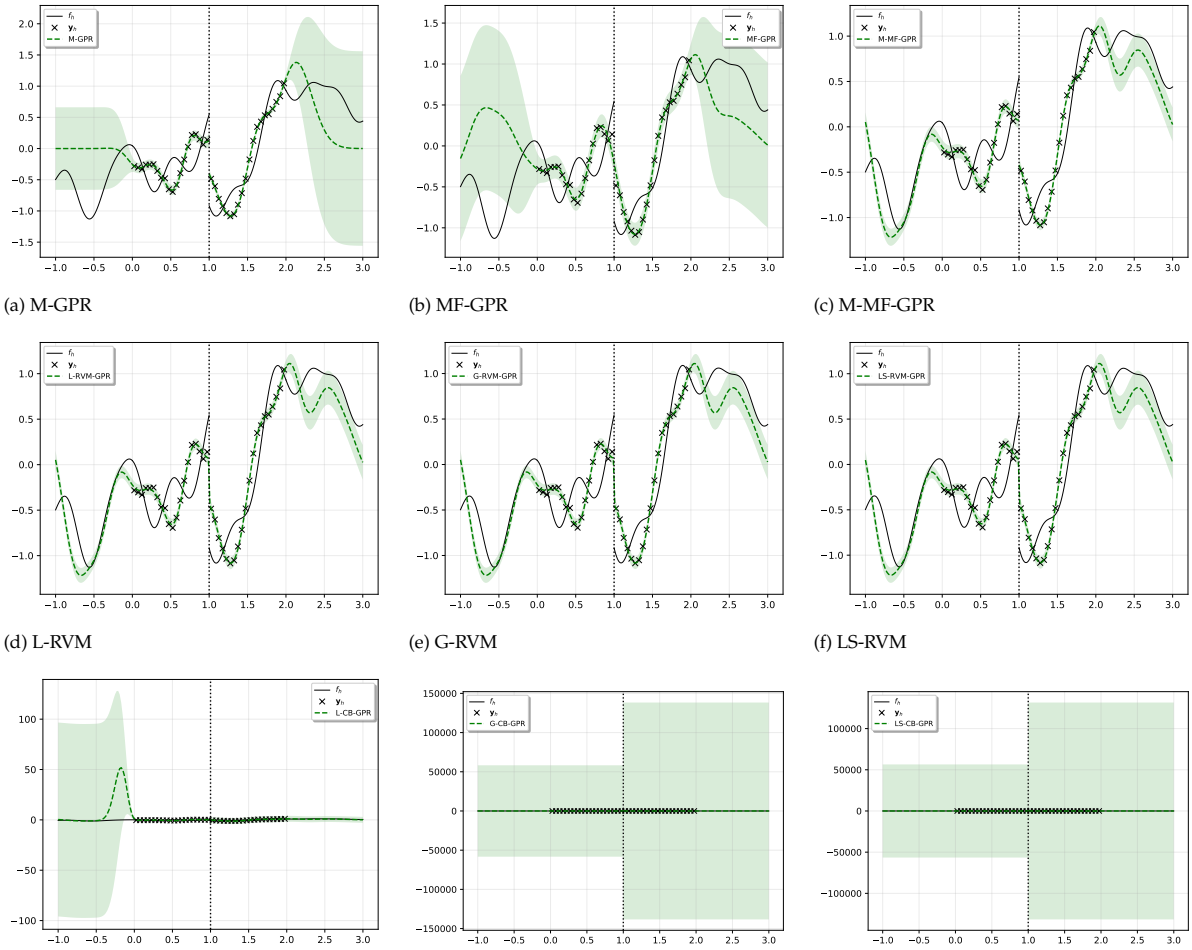
(h) G-CB



(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

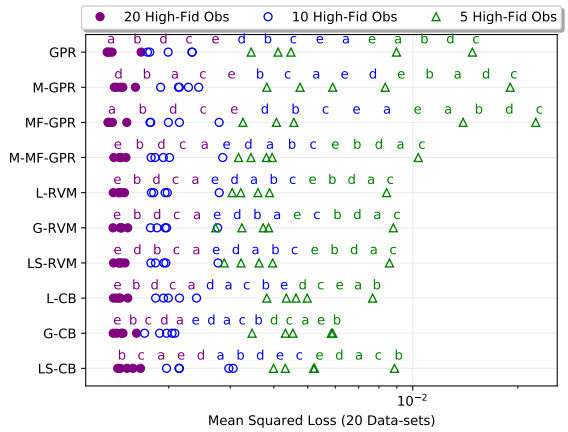
(d) L-RVM (e) G-RVM (f) LS-RVM

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

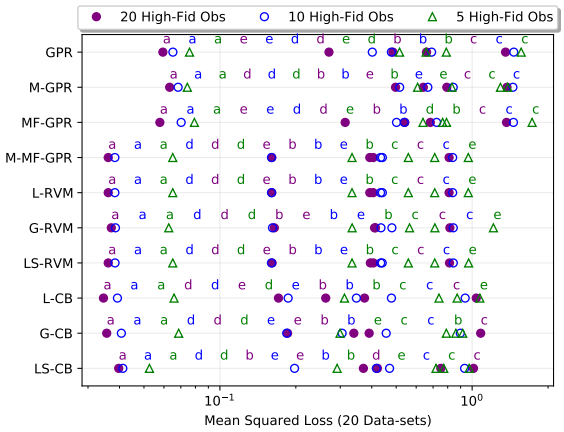
Figure A.40: model predictions with 20 high-fids per region of function **a** and data-set 0 of figure A.37.

A.3.3. Linearly varying ρ case

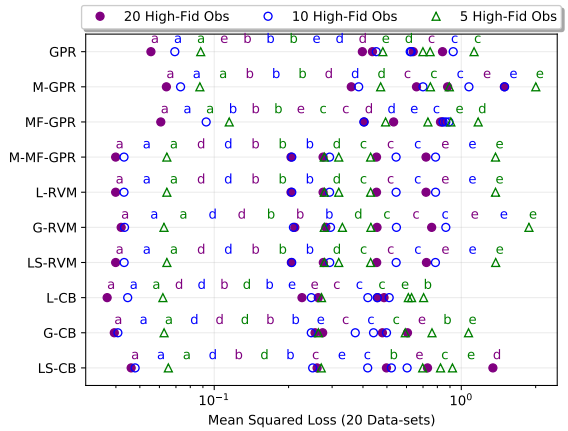
Appendix A: Experimental Results



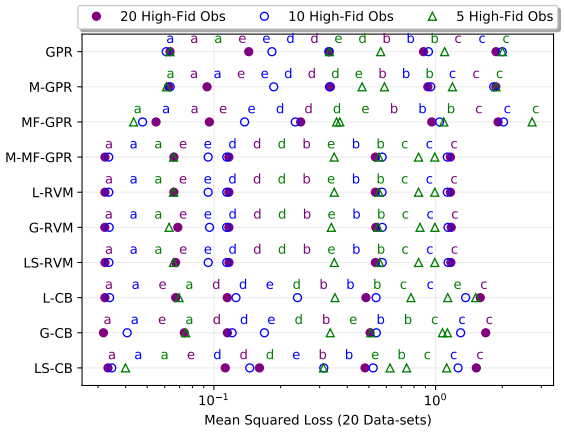
(a) Interpolation regime: $[0,2]$.



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



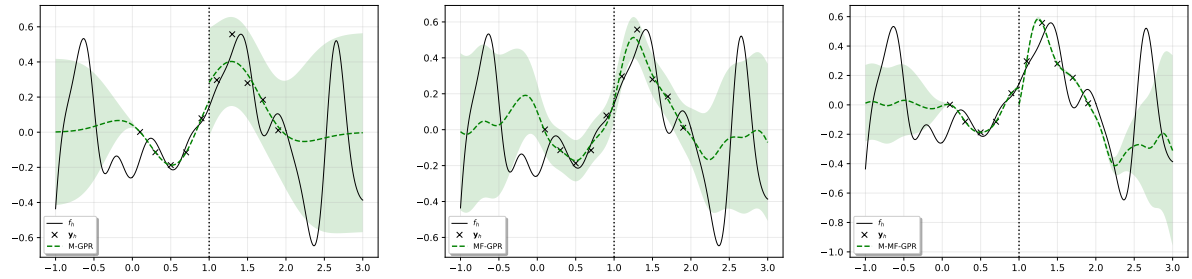
(c) Left extrapolation regime: $[-1,0]$.



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

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

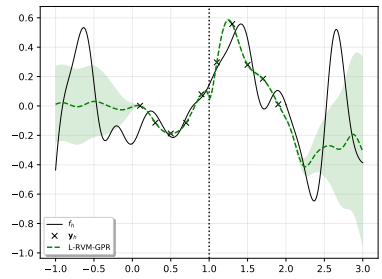
Appendix A: Experimental Results



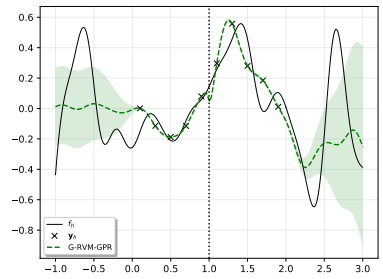
(a) M-GPR

(b) MF-GPR

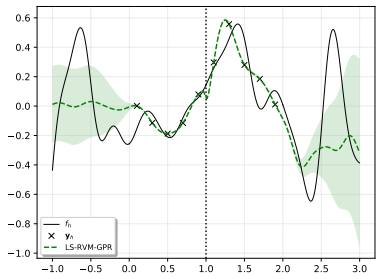
(c) M-MF-GPR



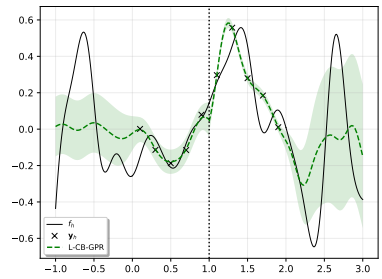
(d) L-RVM



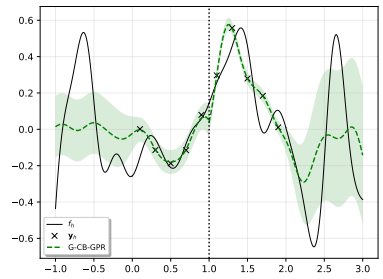
(e) G-RVM



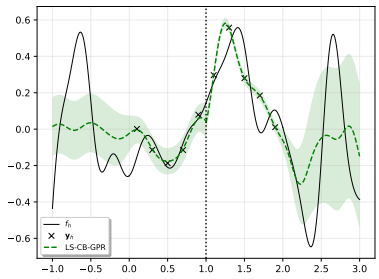
(f) LS-RVM



(g) L-CB



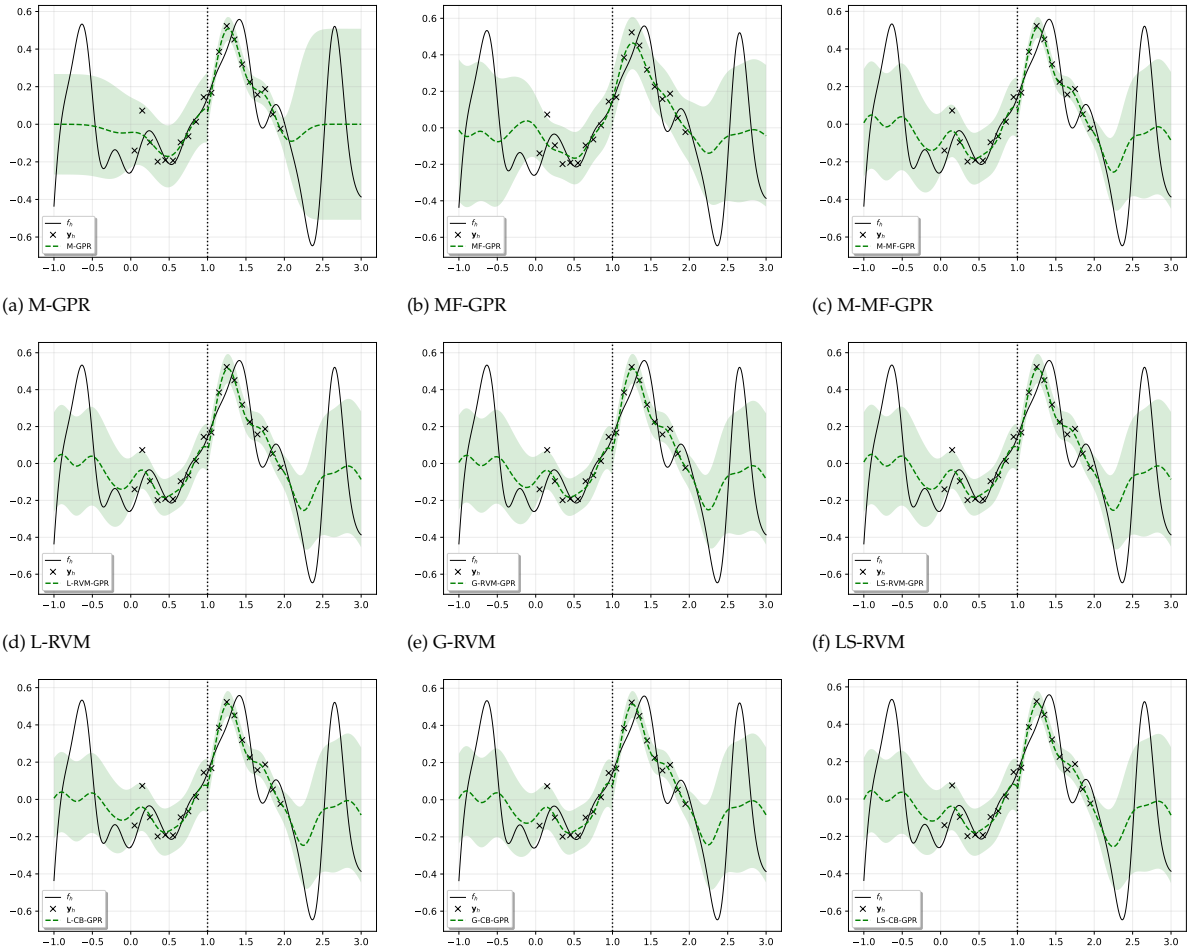
(h) G-CB



(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



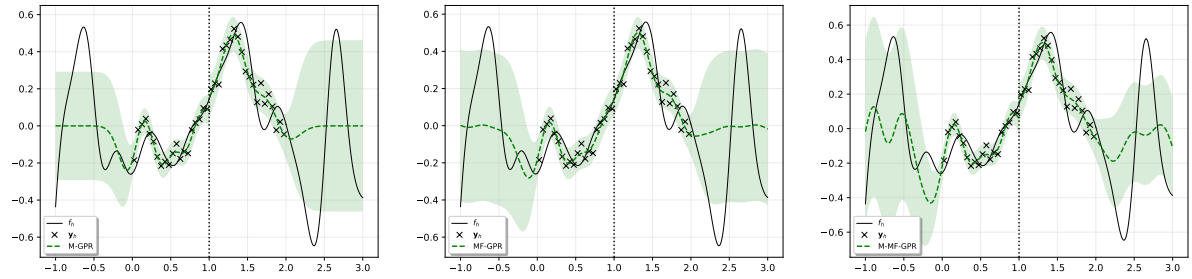
(a) M-GPR (b) MF-GPR (c) M-MF-GPR

(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.

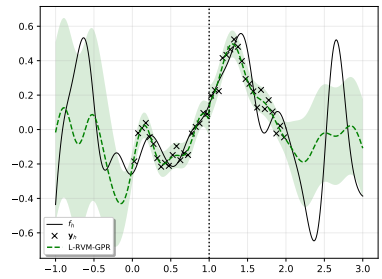
Appendix A: Experimental Results



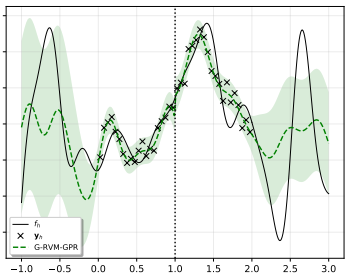
(a) M-GPR

(b) MF-GPR

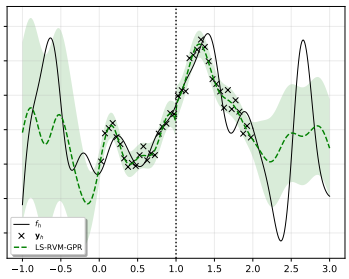
(c) M-MF-GPR



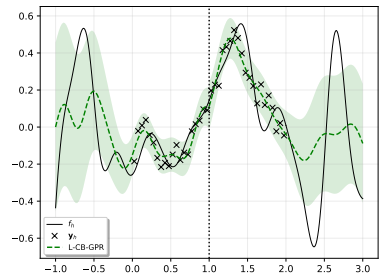
(d) L-RVM



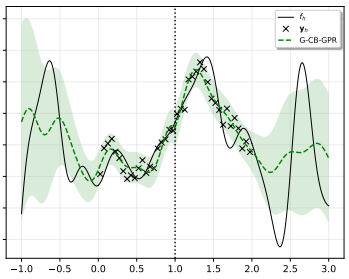
(e) G-RVM



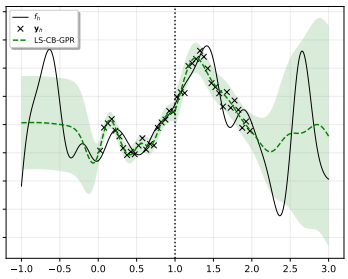
(f) LS-RVM



(g) L-CB



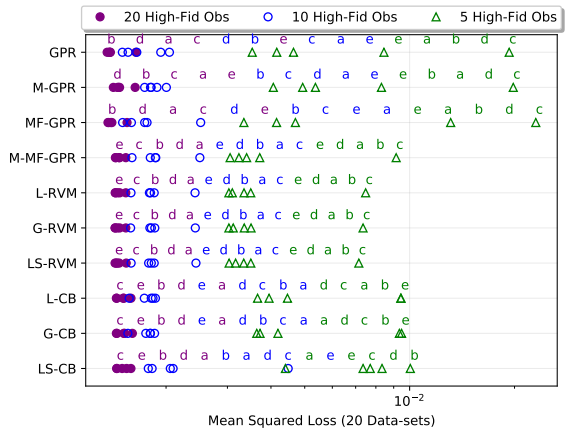
(h) G-CB



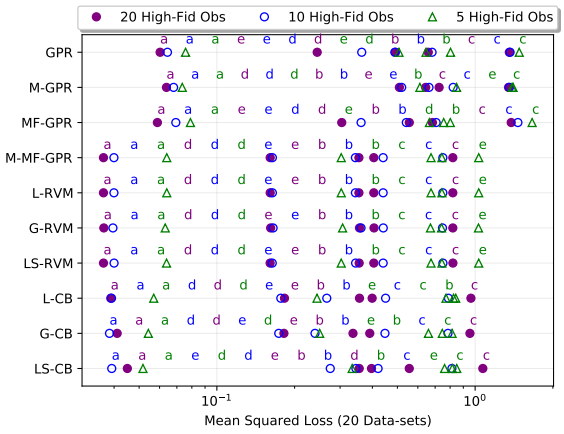
(i) LS-CB

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

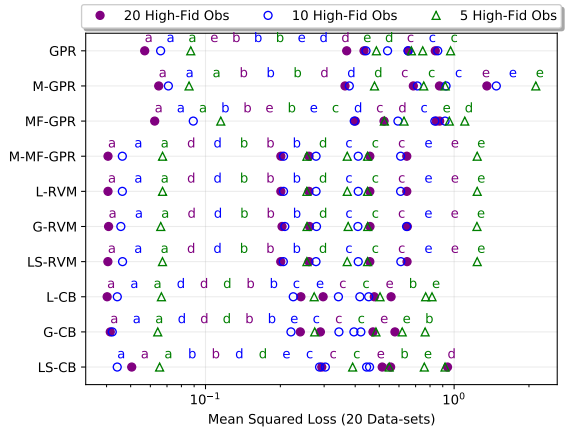
Appendix A: Experimental Results



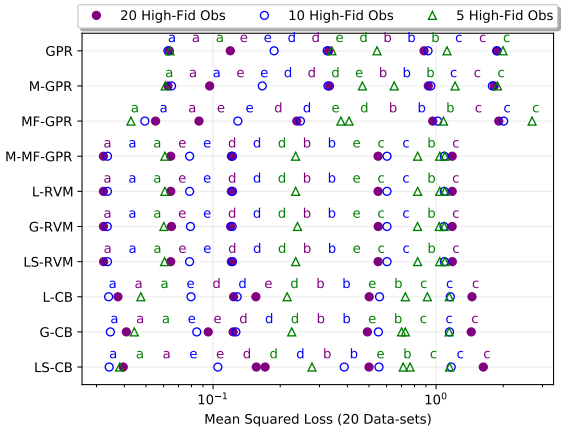
(a) Interpolation regime: $[0,2]$.



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

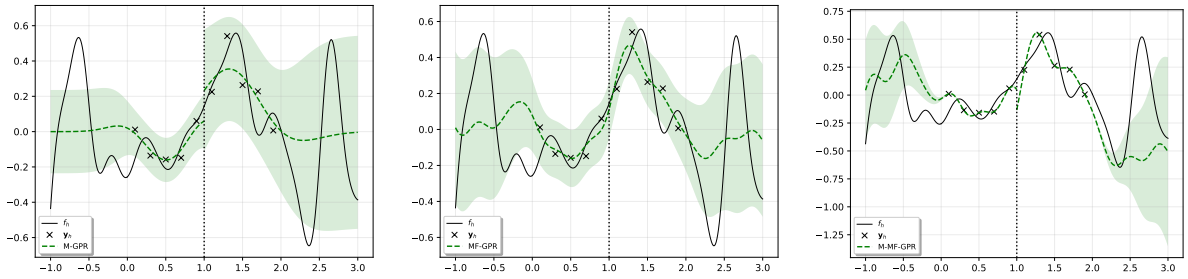


(c) Left extrapolation regime: $[-1,0]$.

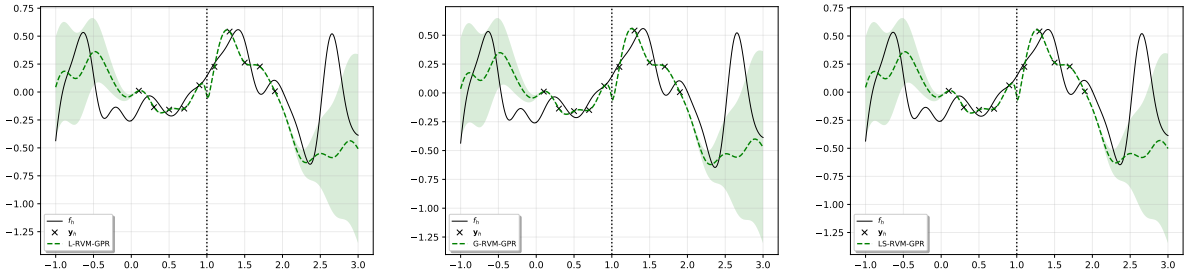


(d) Right extrapolation regime: $[2,3]$.

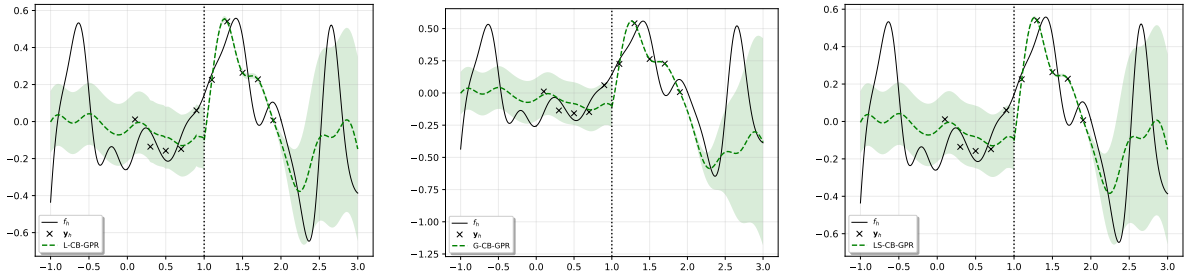
Figure A.45: linearly varying ρ case, linearly spaced, and 101 low-fids per region.



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



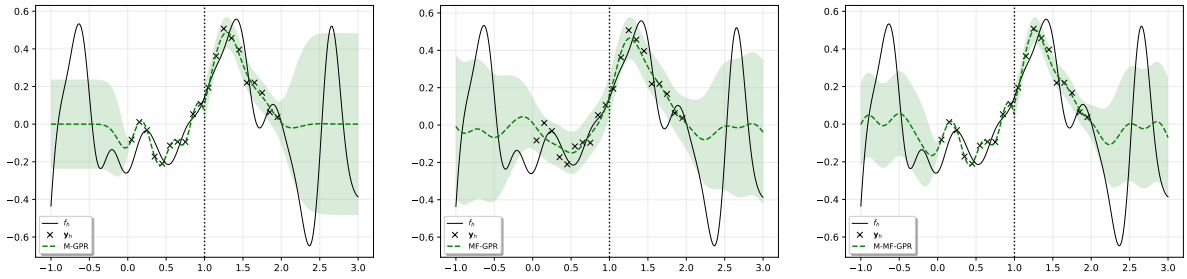
(d) L-RVM (e) G-RVM (f) LS-RVM



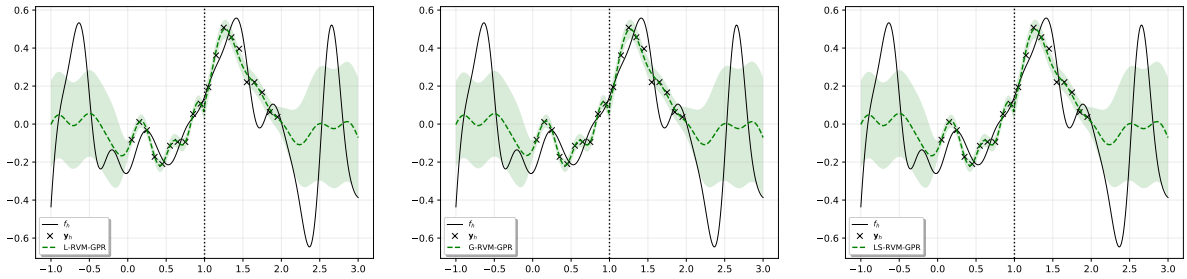
(g) L-CB (h) G-CB (i) LS-CB

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

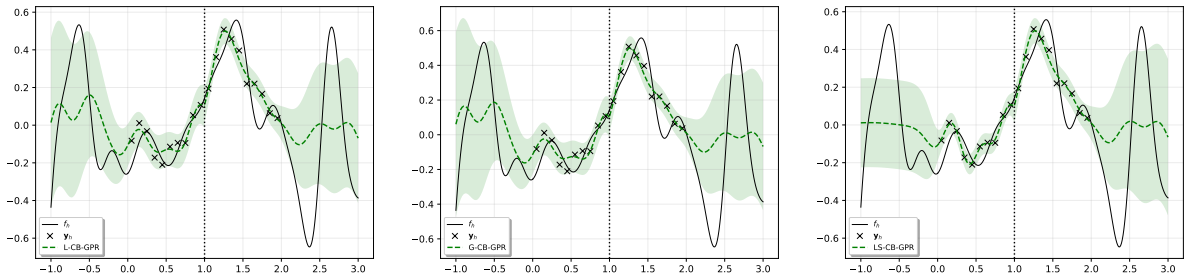
Appendix A: Experimental Results



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

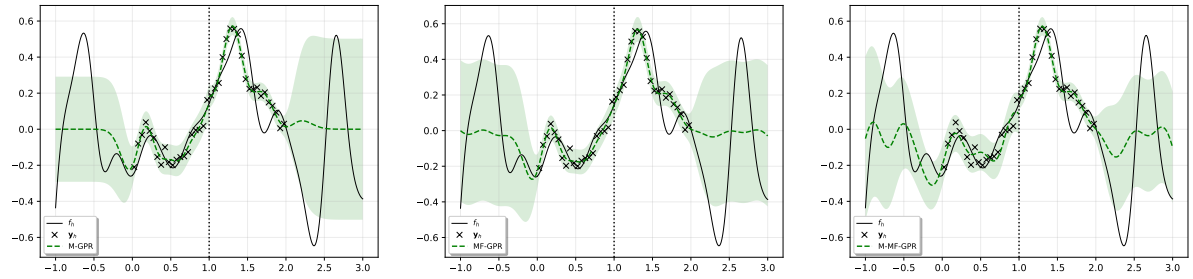


(d) L-RVM (e) G-RVM (f) LS-RVM

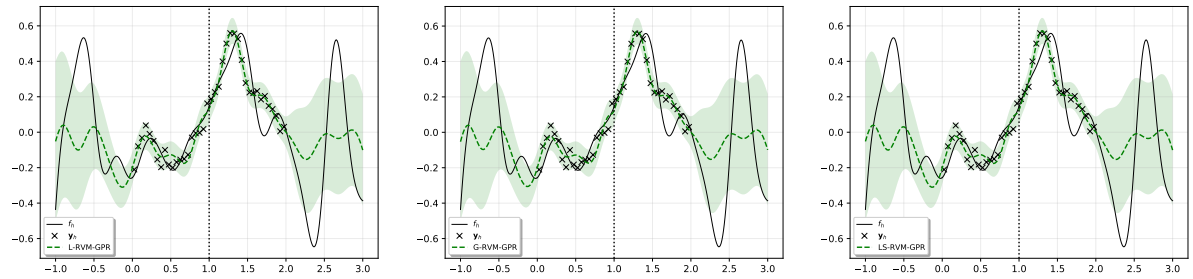


(g) L-CB (h) G-CB (i) LS-CB

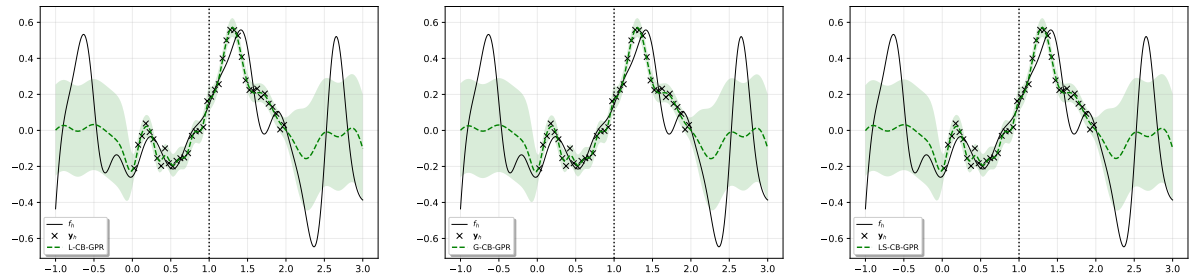
Figure A.47: model predictions with 10 high-fids per region of function **a** and data-set 0 of figure A.45.



(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] Martin Trapp et al. "Deep structured mixtures of gaussian processes". In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2020, pp. 2251–2261.

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

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

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

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

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

[54] 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.