A Joint Kriging Model with Application to Constrained Classification

Didier Rullière, Marc Grossouvre

A Joint Kriging Model with Application to Constrained Classification

Didier Rullière*[1] and Marc Grossouvre†[1,2]

[1]Mines Saint-Etienne, Univ Clermont Auvergne, INP Clermont Auvergne, CNRS, UMR 6158 LIMOS, F - 42023 Saint-Etienne France.
[2]URBS.

September 19, 2024

Abstract

Interpolating or predicting data is of utmost importance in machine learning, and Gaussian Process Regression is one of the numerous techniques that are often used in practice. In this paper, we consider the case of multi-input and multi-output data. A simple *Joint* Kriging model is proposed, where common combination weights are applied to all output variables at the same time. This drastically reduces the number of hyperparameters to be optimised while keeping nice interpolating properties. An original constraint on predicted values is also introduced, useful for considering external information or adverse scenarios. Finally, it is shown that, when applied to membership degrees, the model is especially helpful for constrained fuzzy classification problems. In particular, the model allows for prescribed average percentages of each class in predictions. Numerical illustrations are provided for both simulated and real data and show the importance of the constraint on predicted values. The method also competes with the 69 other models of an open real-world benchmark.

Keywords—Multi-output Kriging, Cokriging, Constrained classification, Spatial Prediction, multi-task Gaussian Process regression.

1 Introduction

Interpolating data is widely used in many fields of computer experiments. It is especially useful to predict the values of one or several variables of interest in the context of time-consuming or costly experiments. One considers here a Kriging interpolation problem on several output variables, with specific constraints on predicted values, so that applications to constrained classification are possible. Let us detail the need to deal with such a problem.

Kriging on several outputs. Kriging, or Gaussian Process Regression, is a method of interpolation, especially suited when there are only a few observations that have to be interpolated. It is widely used in many fields of Machine Learning, originally for geostatistical studies and spatial interpolation, but also for computer experiments in many domains (finance, industry, environment, etc.). As an example, the review *Fifty years of Kriging*

*didier.rulliere@emse.fr
†marcgrossouvre@urbs.fr

Chilès and Desassis, 2018 gives various examples from different domains: design of aircrafts (Chung & Alonso, 2002), mechanical properties of nanomaterials (Yan et al., 2012), supply chain networks (Dixit et al., 2016), financial terms structure (Cousin et al., 2016), modeling of social systems (De Oliveira et al., 2013). Recent uses of Kriging includes for example contamination estimation (de Fouquet et al., 2023), hydrodynamic forces prediction (Qi et al., 2024), engineering design optimization (Toal, 2023). Regarding the combined use of Kriging and Machine Learning, one can also cite among recent studies the estimation of soil moisture constants (Tunçay et al., 2023), the study of strength and permeability of permeable base (Wang et al., 2024).

The most basic Kriging theory aims at predicting a single real-valued quantity of interest, the *output* (for instance, gold concentration in the ground), depending on some explanatory variables that are referred to as *input values* or *locations* (for instance, latitude, longitude, and depth). From a statistical point of view, the Kriging method is based on the best linear unbiased combination of observed outputs, with the assumption that observations are random variables whose correlation depends on locations. From a Gaussian random field point of view, in a Gaussian setting, the interpolation is the mean of a conditional Gaussian random field, with confidence bands derived from the variance of the conditional random field. An in-depth review of Gaussian Processes can be found in C. E. Rasmussen and Williams, 2006.

The method has several advantages. First, it is interpretable: the prediction is a weighted average of observations, with quite a logical behaviour of the weights. Second, the method fully interpolates the data, that is, predicts exactly an observed output if one uses the same input values. And third, it not only gives a prediction but also confidence intervals for this prediction. Among limitations of the method and proposed extensions in the literature, one can cite the difficulty to handle numerous observations, (see e.g. Cressie & Johannesson, 2008; Banerjee et al., 2013; Rullière et al., 2018, and references therein), the difficulty to specify the covariance model and to estimate its hyperparameters (Bachoc, 2013), the difficulty to treat multivalued outputs (Furrer & Genton, 2011).

In this paper, we mainly consider this multivalued output problem, which is clearly of practical interest. One originality of this work is that this kind of multivalued interpolation is also applied to membership degrees in a classification setting. Moreover, a proposed simplification of the model is especially useful since it keeps the property of membership degrees summing to one. At last, another novelty is to consider a specific constraint on predicted values. As detailed below, it will allow for proportion constraints in a classification setting.

Constraints on predicted values. There is a well known joke on actuaries: *How much is two plus two? An actuary will ask "What do you want it to equal?".* At first glance, it seems dishonest to require constraints on predicted values, especially if these constraints are very precise. But such constraints can be useful when having external information, for adverse modelling, or for homogenisation needs, as illustrated below. The constraint we consider, for the model presented in this paper, focuses on the average of predicted values.

It can be very helpful to prescribe a specific value for the average of predicted values. Let us instantiate some examples: Due to an industrial accident, one wishes to measure the pollution in the soil for different chemical products. Measures are done at some spatial places, but the number of measures is limited. One would like to infer the quantity of all chemical products everywhere in the soil. Knowing stockpiles of products before the accident, the total quantity of lost chemicals may be known for every chemical product. While Gaussian Process Regression is especially suited to predicting one product dosage

in the soil, it has difficulty handling jointly a lot of products as it needs to model many cross-covariances. Furthermore, it cannot handle any constraint at all, like prescribing the sum of predicted values to be equal to the known quantity of spilled product. Another example is the case where one needs to build a prediction under an adverse scenario: even if the total quantity of lost chemicals is unknown, it can be useful to get an idea of the distribution of pollutants in an adverse case of massive loss.

In other investigations, there may be external knowledge to consider. For example, a regional study might want to be in line with some given national statistics if there is no reason that the regional statistics differ on average. One can observe data due to an exceptional situation (e.g., COVID), and one may want to use it knowing that the situation has returned to normal. Or one might want predictions over different years or over different regions to coincide, at least on average. For instance, one may want that some disease incidence prediction does not differ, on average, over different medical centres. Another situation is the following: imagine that one knows, under an arbitrage-free setting, that some predicted stock returns must be zero on average, or imagine that the regulator wants to force a prediction under specific shock scenarios. Fairness constraints can also be introduced to limit unfairness in algorithmic decision-making (Zafar et al., 2019). Therefore, prescribing the average value of predictions is useful in multiple contexts, be it external information (known quantity of chemical, national statistic, etc.), adverse modelling (regulation, simulation under specific scenarios, etc.), or the need to homogenise results (over different regions, observed years, fairness constraints, etc.).

Constrained Kriging and fuzzy classification. Fuzzy classification is useful when an individual may simultaneously belong to multiple classes of a categorical variable, or when one is trying to predict a distribution of the probabilities for an individual to belong to each class of a categorical variable. In both cases, one usually builds a model to predict a quantitative variable associated with each class. The larger the quantitative variable, the more likely an individual is to belong to the associated class. These quantitative variables are called membership degrees. If those membership degrees are positive and sum to 1, they can be assumed to be probabilities.

Applying multi-output Kriging on membership degrees has several advantages for fuzzy classification. One advantage is that the interpolation property can be preserved, which is not necessarily the case for other classification or clustering techniques like KNN: even at a location very close to a given observation, KNN can predict another class than the observed one. Another advantage of using a multi-output Kriging model on membership degrees is to get an estimation of the uncertainty of the prediction. For instance, at a specific location, one may predict 10 % of the chance that the class is one, but one can also give a confidence interval for this quantity.

Applied to classification, constraints on predicted values are useful for above-mentioned reasons: taking into account external information, adverse modeling, homogeneisation needs. In particular, a constraint on average predicted membership degrres forces predictions with prescribed class percentages: say one wants to predict 5% of A, 70% of B, 25% of C in average over all predictions. Such a constraint may occur in practical situations: e.g. in an epidemic study, one might want an adverse scenario with more disease predictions, or one may know true classes percentages at a larger scale. Such a constraint is also helpful for rarer classes: as an example, in the case where all labels excepts one are rare, an unconstrained classifier would surely always predict the most common label, hence ensuring a good accuracy. But studies may rely on the other rare classes, it can be essential to get some predictions of these rare classes.

Literature. The proposal here is to use multi-output Kriging with classification, under specific constraints on predicted values.

Regarding Multi-output Kriging, there is a huge amount of literature available. Reference books can be found on the topic, such as Wackernagel, 2003 and Chiles and Delfiner, 2012. The modelling of cross-covariance functions is detailed in several papers, as in Alvarez et al., 2012; Genton and Kleiber, 2015. Recent papers are dealing with inference and prediction using multitask Gaussian Processes (Leroy et al., 2022, 2023). Co-Kriging techniques are built to treat several outputs, but there is usually one main output, and others are used to improve the prediction of the main considered output. Furthermore, all cross-covariances between outputs at different locations have to be modelled, which creates $O(p^2)$ covariance models, where p is the number of outputs (Goovaerts, 1998; Ver Hoef & Cressie, 1993; Furrer & Genton, 2011). While highly parametrised models are useful in many situations, the prediction quality relies on the proper specification of the model and on the estimation of its parameters. A fine model with the wrong parameters can sometimes be less efficient than a simpler model with more control over a few parameters (C. Rasmussen & Ghahramani, 2000). One considers, here, a model where Kriging is applied to multivalued outputs in $\mathbb{R}^p$, but with a specific simplification leading to a single covariance function to tune instead of $O(p^2)$.

Some works can be found in the literature about clustering or classification under constraints. A survey on constrained classification can be found in Gordon, 1996, and references therein. Some research works treat size constraints for clustering (Bradley et al., 2000; Höppner & Klawonn, 2008; Ganganath et al., 2014), while others treat the problem of fuzzy clustering with weights (membership degrees), as in the present work, see for example Benatti et al., 2022. Fairness constraints are also considered in Zafar et al., 2019.

Regarding Kriging and classification, some works on classification using Gaussian settings can be found in a dedicated Chapter 3 in the book C. E. Rasmussen and Williams, 2006. In particular, for binary classification, membership probabilities can be approximated by a sigmoid transformation of some latent Gaussian Process. The approach can be generalised to multi-class problems, and Bayesian inference can be conducted using analytic approximations of integrals, or solutions based on Monte Carlo sampling (Williams & Barber, 1998; C. E. Rasmussen & Williams, 2006, and references therein). Other recent approaches involving Multi-task Gaussian processes, using several latent Gaussian processes, and Bayesian inference with approximations or sampling can be found in Dahl and Bonilla, 2019; Panos et al., 2021.

Among works closer to what is proposed in the present work, Indicator Kriging aims at determining the cumulative distribution function (cdf) of an underlying random field at an unknown location, as a weighted average of indicators. It uses linear combinations of transformed observations too, but relies on a direct link between indicators and the underlying random field using thresholds. Hence, it does not seem to be directly suited to classify non-ordinal data (without any hierarchy between classes). It also requires the observation of the latent process that generates the indicators (Journel, 1983; Meer, 1996; Goovaerts, 2009; Chiang et al., 2013). Extensions like indicator co-Kriging require a large number of cross-covariances (Agarwal et al., 2021). In the present paper, the proposed method can be applied to non-ordinal data, and does not require a specific model or thresholds between indicators of membership and an underlying real random field; furthermore, in a simplified setting, the whole method can also rely on a single covariance function.

Proposal To the best of our knowledge, the use of Kriging on several outputs with application to classification under constraints on predicted values has not been developed yet. We present, in this work, such a model. It involves a reduction of the number of hyperparameters. And it includes the possibility of considering specific constraints. The present approach directly yields closed-form formulas without the need for conditional density approximations or sampling. Such original constraints are not typically addressed by classical multi-output Kriging or Gaussian Process regression.

It seems to us that using multi-output Kriging on classification offers many modelling perspectives as well as practical results and performance. We will see that the proposed model competes with the best available methods on an open data set, among the 69 competitors of an open benchmark.

Structure The paper is structured as follows. In Section 2, we define a simplified Kriging model that is suited for multivalued outputs. The model is detailed in three cases: with no specific constraint, similarly to Simple Kriging; with weights summing to 1, similarly to Ordinary Kriging; with constraints on weights summing to 1 and on average predicted values. In each case, we derive optimal weights together with the prediction mean and variance. An extension using an affine prediction is also developed. In Section 3, the proposed interpolation technique is applied to membership degrees, and it is shown that it preserves useful basic properties for the prediction. Section 4 details strategies to fill the required covariance matrices and hyperparameters. In Section 5, numerical applications of the proposed interpolation technique are given. One considers in particular a minimal application on a toy example, an illustration on a multivalued time series on a real data set, and a more detailed real-world application on a classification problem. A conclusion closes the paper.

Appendix . For more readability, all proofs are gathered in Appendix A, page 41. A list of notation and symbols is given in Appendix B. All illustrations are generated with notebooks that are available as online supplementary material[1], in modifiable and executable format `.Rmd` and in already executed directly readable `.html` format (Grossouvre & Rullière, 2023). Hence, the results are fully reproducible, and all specifications for drawing figures are easy to retrieve.

2 Joint Kriging Model

Let us consider a multivalued random field $\mathbf{Y}(x) := (Y_1(x), \ldots, Y_p(x))^\top \in \mathbb{R}^p$, $x \in \chi$ where χ is a metric set of input points, typically $\chi = \mathbb{R}^d$. For the sake of clarity and using analogy with geostatistics, we will refer to x as *locations*, but χ may contain any explanatory variable. The components $Y_1(.), \ldots, Y_p(.)$ will be referred to as the p considered *output variables*. Components of the random field $\mathbf{Y}(x)$ can be dependent. Furthermore, $\mathbf{Y}$ or its components are not necessarily Gaussian. However, one assumes that first- and second-order moments exist. One considers here that $\mathbf{Y}(x) \in \mathbb{R}^p$ and $\chi = \mathbb{R}^d$, but other metric spaces would be possible as soon as expectation and cross-covariances between $\mathbf{Y}(x)$ and $\mathbf{Y}(x')$ can be derived.

Given n observations of $\mathbf{Y}(x_1), \ldots, \mathbf{Y}(x_n)$, we aim at predicting the values of the random field at some unobserved locations $x_1^\star, \ldots, x_q^\star$, i.e., we aim at giving a predictor of $\mathbf{Y}(x_1^\star), \ldots, \mathbf{Y}(x_q^\star)$. At an unobserved location $x^\star$, we define the *Joint Kriging predictor* as

[1] at https://gitlab.com/urbs-imope/rdscripts/jointkrigingsupplementary

a predictor $\mathbf{M}(x) = (M_1(x), \ldots, M_p(x))^\top$ depending linearly on observations, where real coefficients apply jointly to all components of the observations:

$$\mathbf{M}(x^\star) := \sum_{i=1}^{n} \alpha_i(x^\star)\mathbf{Y}(x_i) \text{ where } \forall i \in \{1, \ldots, n\},\ \alpha_i(x^\star) \in \mathbb{R}. \tag{1}$$

These weights $\boldsymbol{\alpha}(x^\star) := (\alpha_1(x^\star), \ldots, \alpha_n(x^\star))^\top$ are optimised in order to minimise some error that we will detail later on, under various possible constraints. Now, defining the $p \times n$ matrix $\mathbb{Y} := [\mathbf{Y}(x_1), \ldots, \mathbf{Y}(x_n)]$, Equation (1) also writes in a compact way:

$$\mathbf{M}(x^\star) = \mathbb{Y}\boldsymbol{\alpha}(x^\star). \tag{2}$$

The main assumption here is that the weights are impacting all components the same way: the first component $M_1(x^\star)$ is a linear combination of the observed first components, namely $Y_1(x_1), \ldots, Y_1(x_n)$; the second component $M_2(x^\star)$ is the same linear combination of the observed second components $Y_2(x_1), \ldots, Y_2(x_n)$, etc. In other words, the weights affect jointly, or simultaneously, all the components of observed $\mathbf{Y}(x_i)$, $i = 1, \ldots, n$, hence the chosen name of *Joint Kriging model*. We will see in Section 3 that this key *simplifying assumption* is especially useful for classification under constraints. It would be technically possible to release this assumption, e.g., by replacing weights $\alpha_i(x)$ by some $p \times p$ matrix for $i = 1, \ldots, n$; one would get closer to some general co-Kriging model with $O(p^2)$ covariance models, but that is not the purpose of the present work.

Let us define the prediction error associated with a vector of weights $\boldsymbol{\alpha}(x^\star)$, at a prediction location $x^\star$. This loss is defined as the scalar value:

$$\Delta(x^\star) := \mathrm{E}\left[\|\mathbf{M}(x^\star) - \mathbf{Y}(x^\star)\|_{\mathbb{W}}^2\right], \tag{3}$$

where $\|\mathbf{v}\|_{\mathbb{W}}^2 := \mathbf{v}^\top\mathbb{W}\mathbf{v}$ is a squared norm with $\mathbb{W}$ a given symmetrical positive-definite matrix of real weights. For instance, if one changes the unit of the first output variable, say multiply it by 100, then it sounds logical that the resulting norm be unchanged. Thus, some weights' matrix may seem reasonable: an inverse covariance matrix as in the Mahalanobis distance, or a diagonal matrix of inverse variances, etc. For simplicity, the reader may imagine that all p output variables are already scaled and that $\mathbb{W}$ is the $p \times p$ identity matrix.

The main difficulty is to derive the optimal weights $\boldsymbol{\alpha}(x^\star)$ under the various constraints one would like to consider. At all prediction locations $x_1^\star, \ldots, x_q^\star$, one thus aims at determining the optimal weights, gathered in a $n \times q$ matrix:

$$\mathbb{A} := \left[\boldsymbol{\alpha}(x_1^\star), \ldots, \boldsymbol{\alpha}(x_q^\star)\right].$$

This is performed in the three following subsections under different constraints.

2.1 Optimal Weights Without Constraints

In this subsection, we define optimal weights that minimise the prediction error without supplementary constraints.

Let us denote $\mathcal{S}_n^+(\mathbb{R})$ the set of real valued symmetric semi-definite positive $n \times n$ matrices and $\mathcal{M}_{n\times q}(\mathbb{R})$ the set of real valued $n \times q$ matrices. The following Proposition expresses the weights such that $\mathbf{M}(x^\star)$ is a BLUP of $\mathbf{Y}(x^\star)$, in the sense of minimising the loss (3). The result looks exactly the same as in the simple Kriging model, but the components in the symmetric positive semidefinite matrix $\mathbb{K}$ and in the vector $\mathbf{h}(x^\star)$ here

aggregate the values of all p observed, mutually dependent, output variables. One retrieves the usual Simple Kriging equations in the case where $p = 1$ and $\mathbb{W}$ is the identity matrix.

Proposition 1 (Simple Joint Kriging weights)**.** *The optimal weights $\boldsymbol{\alpha}(x^\star)$ minimising the loss of Equation* (3) *are given by:*

$$\boldsymbol{\alpha}(x^\star) = \mathbb{K}^{-1}\mathbf{h}(x^\star) \ \in \mathbb{R}^n, \tag{4}$$

or equivalently, using a matrix expression to predict simultaneously over the q locations,

$$\mathbb{A} = \mathbb{K}^{-1}\mathbb{H} \ \in \mathcal{M}_{n\times q}(\mathbb{R}), \tag{5}$$

where

$$\begin{aligned}
\mathbb{K} &:= \mathrm{E}\left[\mathbb{Y}^\top \mathbb{W}\mathbb{Y}\right] \ \in \mathcal{S}_n^+(\mathbb{R}) \text{ is assumed to be invertible,}\\
\mathbf{h}(x^\star) &:= \mathrm{E}\left[\mathbb{Y}^\top \mathbb{W}\mathbf{Y}(x^\star)\right] \ \in \mathbb{R}^n,\\
\mathbb{H} &:= \left[\mathbf{h}(x_1^\star), \ldots, \mathbf{h}(x_q^\star)\right] \ \in \mathcal{M}_{n\times q}(\mathbb{R})
\end{aligned}$$

If furthermore for all output variables $j = 1, \ldots, p$, for all location $x \in \chi$, $\mathrm{E}\left[Y_j(x)\right] = 0$, then $\mathbf{M}(x^\star)$ is unbiased.

Proof. The proof is postponed to Appendix. A.1, page 41. □

Note that the matrix $\mathbb{K}$ is necessarily a covariance matrix since it is symmetric positive semidefinite.

Here, we have weights applied jointly to all components, which leads to a simplified predictor. The prediction accuracy may suffer from this simplifying assumption, compared to heavily parameterised models. However, we operate with general assumptions that are the finite moments of orders 1 and 2, and we do not require stationarity, independence or a Gaussian setup. The assumptions we make are about the covariance function of a tunable weighted sum of components. The predictor can still take into account dependencies between those components and non-stationarities.

We will see that this simplified predictor is required to handle specific constraints, such as "higher-scale constraints" in Section 3.1. More details on covariances in $\mathbb{K}$ and $\mathbb{H}$ will be given in a dedicated Section 4, where links with specific cross-covariance models of the literature are also presented.

2.2 Optimal Weights Summing to One

In this section, one considers an additional constraint. This constraint raises naturally when the random variables $Y_i(x)$ are not centred, and it leads to weights summing to one, as in Ordinary Kriging (Cressie, 1988), namely for all $x^\star$,

$$\boldsymbol{\alpha}^\top(x^\star)\mathbf{1}_n = 1 \tag{6}$$

where $\mathbf{1}_n$ is a $n \times 1$ vector of ones.

The above constraint implies that the prediction is a weighted average of observations. Therefore, in the case where output variables' expectation is constant over the territory χ and equal to $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_p)^\top$, then the expectation of the prediction is also $\boldsymbol{\mu}$:

$$\begin{aligned}
&\text{If (6)}\\
&\text{and } \forall x \in \chi,\ \mathrm{E}\left[\mathbf{Y}(x)\right] = \boldsymbol{\mu}\\
&\text{then } \mathrm{E}\left[\mathbf{M}(x^\star)\right] = \boldsymbol{\mu}\\
&\text{and therefore } \mathrm{E}\left[\mathbf{M}(x^\star)\right] = \mathrm{E}\left[\mathbf{Y}(x^\star)\right].
\end{aligned}$$

Conversely,

$$\begin{aligned} &\text{If } \forall x \in \chi,\ \mathrm{E}\left[\mathbf{Y}(x)\right] = \boldsymbol{\mu} \\ &\text{and } \forall i \in \{1, \ldots, p\},\ \mu_i \neq 0 \\ &\text{and } \mathrm{E}\left[\mathbf{M}(x^\star)\right] = \boldsymbol{\mu} \\ &\text{then (6).} \end{aligned}$$

Hence, (6) is a very natural constraint. It does not imply, however, that $\mathbf{M}(x^\star)$ is a convex combination of all $\mathbf{Y}(x_i)$, because some weights can still be negative.

Under this constraint of weights summing to 1, the following proposition gives the optimal weights. One retrieves similar formulae as in ordinary Kriging, but the involved elements in matrices $\mathbb{K}$ and $\mathbb{H}$ are different: they are computed taking into account all p mutually dependent output variables over all observations.

Proposition 2 (Ordinary Joint Kriging weights)**.** *Under the constraint of Equation* (6)*, the optimal weights* $\boldsymbol{\alpha}(x^\star)$ *minimising the loss of Equation* (2) *are given by:*

$$\boldsymbol{\alpha}(x^\star) = \mathbb{K}^{-1}\left(\mathbf{h}(x^\star) + \lambda(x^\star)\mathbf{1}_n\right) \in \mathbb{R}^n.$$

Equivalently, using matrix expressions, one gets

$$\mathbb{A} = \mathbb{K}^{-1}\left(\mathbb{H} + \mathbf{1}_n\boldsymbol{\lambda}^\top\right) \in \mathcal{M}_{n\times q}(\mathbb{R})$$

where

$$\begin{aligned} \delta &:= {\mathbf{1}_n}^\top\mathbb{K}^{-1}\mathbf{1}_n \in \mathbb{R}, \\ \lambda(x^\star) &:= \frac{1}{\delta}\left(1 - {\mathbf{1}_n}^\top\mathbb{K}^{-1}\mathbf{h}(x^\star)\right) \in \mathbb{R}, \qquad & \mathbb{K} &:= \mathrm{E}\left[\mathbb{Y}^\top\mathbb{W}\mathbb{Y}\right] \in \mathcal{S}_n^+(\mathbb{R}), \\ \boldsymbol{\lambda} &:= \left(\lambda(x_1^\star), \ldots, \lambda(x_q^\star)\right)^\top \in \mathbb{R}^q, \qquad & \mathbf{h}(x^\star) &:= \mathrm{E}\left[\mathbb{Y}^\top\mathbb{W}\mathbf{Y}(x^\star)\right] \in \mathbb{R}^n, \\ \boldsymbol{\lambda}^\top &= \frac{1}{\delta}\left({\mathbf{1}_q}^\top - {\mathbf{1}_n}^\top\mathbb{K}^{-1}\mathbb{H}\right), \qquad & \mathbb{H} &:= \left[\mathbf{h}(x_1^\star), \ldots, \mathbf{h}(x_q^\star)\right] \in \mathcal{M}_{n\times q}(\mathbb{R}). \end{aligned}$$

If furthermore, for all output variables $i = 1, \ldots, p$*, for all locations* $x \in \chi$*,* $\mathrm{E}\left[Y_i(x)\right] = \mu_i$*, then* $\mathbf{M}(x^\star)$ *is unbiased.*

Proof. The proof is postponed to Appendix. Subsection A.2, page 41. □

In some cases, matrices can be expressed indifferently with compact expressions, using $\mathbb{K}$ and $\mathbf{h}(x^\star)$, or with more classical covariances, using $\widetilde{\mathbb{K}}$ and $\widetilde{\mathbf{h}}(x^\star)$, as stated in the following remark.

Remark 1 (Covariance matrices)**.** *Let us define the true unknown values of* $\mathbf{Y}$ *at all prediction points by* $\mathbb{Y}^\star := \left[\mathbf{Y}(x_1^\star), \ldots, \mathbf{Y}(x_q^\star)\right]$*. Assume* $\mathrm{E}\left[\mathbf{Y}(x)\right] = \boldsymbol{\mu}$ *for all* $x \in \chi$*. Furthermore, assume that either weights sum to one, that is,* $\boldsymbol{\alpha}(x^\star)^\top\mathbf{1}_n = 1$*, or* $\boldsymbol{\mu} = \mathbf{0}_p$*.*

Then the matrices $\mathbb{K}$*,* $\mathbb{H}$*, and the vector* $\mathbf{h}(x^\star)$ *can be replaced by*

$$\begin{aligned} \widetilde{\mathbb{K}} &:= \mathrm{E}\left[\mathbb{Y}^\top\mathbb{W}\mathbb{Y}\right] - \mathrm{E}\left[\mathbb{Y}^\top\right]\mathbb{W}\,\mathrm{E}\left[\mathbb{Y}\right] \\ \widetilde{\mathbb{H}} &:= \mathrm{E}\left[\mathbb{Y}^\top\mathbb{W}\mathbb{Y}^\star\right] - \mathrm{E}\left[\mathbb{Y}^\top\right]\mathbb{W}\,\mathrm{E}\left[\mathbb{Y}^\star\right] \\ \widetilde{\mathbf{h}}(x^\star) &:= \mathrm{E}\left[\mathbb{Y}^\top\mathbb{W}\mathbf{Y}(x^\star)\right] - \mathrm{E}\left[\mathbb{Y}^\top\right]\mathbb{W}\,\mathrm{E}\left[\mathbf{Y}(x^\star)\right] \end{aligned}$$

everywhere in Proposition 2, without changing the optimal weights $\boldsymbol{\alpha}(x^\star)$*.*

Proof. The proof is postponed to Appendix. Subsection A.3, page 42. □

2.3 Optimal Weights With Constraint on Predictions

The constraint we consider here is more original than the previous one: we would like that, given observations $\mathbb{Y}$, the average of the predicted values has some prescribed value. Formally, we introduce $X^\star$, a random variable taking values in prediction locations $\{x_1^\star, \ldots, x_q^\star\}$, and we introduce the constraint:

$$\mathrm{E}\left[\mathbf{M}(X^\star) \,|\, \mathbb{Y}\right] = \mathbf{m} \text{ for some } \mathbf{m} \in \mathbb{R}^p. \tag{7}$$

This constraint relies on predicted values for a given set of observations. The idea is to force the optimal weights to take into account this *a posteriori* constraint. The interpretation of this constraint is that there is a secondary source of information that gives knowledge of the output expectation over the points to predict. A typical case would be one where the observations and the points to predict both form a representative sample of the territory.

Notice the importance of conditioning by $\mathbb{Y}$. Otherwise, if all $Y_i(x)$ are centred, then the constraint would not be possible to satisfy in general since all $\mathbf{M}(x^\star)$ would be centred. We will see that this kind of constraint is particularly useful for fuzzy classification when one wishes to force the proportions of classes, whatever the observed values.

Gathering all predictors in a single matrix $\mathbb{M}$, we have:

$$\begin{aligned}
\mathbb{M} &:= \left[\mathbf{M}(x_1^\star), \ldots, \mathbf{M}(x_q^\star)\right] \\
\text{i.e. } \mathbb{M} &= \mathbb{Y}\mathbb{A}, \\
\text{and denoting } \pi_{x^\star} &:= \mathrm{P}\left[X^\star = x^\star\right], \\
\boldsymbol{\pi} &:= \left(\pi_{x_1^\star}, \ldots, \pi_{x_q^\star}\right)^\top, \\
\text{we have } \mathbb{Y}\mathbb{A}\boldsymbol{\pi} &= \mathbf{m}.
\end{aligned} \tag{8}$$

One specificity is that the resulting weights in the matrix $\mathbb{A}$ will have to be solved all at once for all q prediction locations. This is different from usual Kriging settings, where prediction locations can be treated separately if desired.

The constraint (7) is cumulated with the above constraint (6), so that the new system of constraints is:

$$\begin{cases} \mathbb{A}^\top \mathbf{1}_n = \mathbf{1}_q \\ \mathbb{Y}\mathbb{A}\boldsymbol{\pi} = \mathbf{m} \end{cases} \tag{9}$$

In general, those constraints are linearly independent. The necessary and sufficient condition for those constraints to be linearly dependent is:

$$\exists \boldsymbol{\omega} \in \mathbb{R}^p \setminus \{\mathbf{0}_p\}, \exists \omega_0 \in \mathbb{R}, \text{ such that } \mathbb{Y}^\top \boldsymbol{\omega} = \omega_0 \mathbf{1}_n$$

It is the case, in particular, if $\mathbb{Y}$ is a matrix of membership degrees in a fuzzy classification context, where $\mathbb{Y}^\top \mathbf{1}_p = \mathbf{1}_n$.

In the following proposition, we give the matrix of optimal weights $\mathbb{A}$ when both constraints are considered at the same time: the constraint (7) on predicted values and the constraint (6) on weights summing to one. We treat both cases: when the system of Equations (9) is of full rank $q + p$ and when its rank is $q + p - 1$.

Proposition 3 (Joint Kriging weights under a predicted values constraint)**.** *The Joint Kriging weights minimising the loss of Equation* (3) *under the constraint of weights summing to one of Equation* (6)*, and prescribed average predicted values of Equation* (7) *write:*

$$\mathbb{A} = \mathbb{K}^{-1}\left(\mathbb{H} + \mathbf{1}_n\boldsymbol{\lambda}^\top + \mathbb{Y}^\top\boldsymbol{\lambda}'\boldsymbol{\pi}^\top\right) \tag{10}$$

- *If the system of Equations* (9) *is of full rank $q+p$, Lagrange multipliers are:*

$$\begin{aligned}
\boldsymbol{\lambda}' &= \frac{1}{\gamma}\left(\frac{1}{\delta}\mathbf{u}\mathbf{u}^\top - \mathbb{Y}\mathbb{K}^{-1}\mathbb{Y}^\top\right)^{-1}\left(\mathbb{Y}\mathbb{K}^{-1}\mathbb{H}\boldsymbol{\pi} + \frac{1}{\delta}\mathbf{u}\left(1 - \mathbf{1}_n{}^\top\mathbb{K}^{-1}\mathbb{H}\boldsymbol{\pi}\right) - \mathbf{m}\right) && \in \mathbb{R}^p \\
\boldsymbol{\lambda} &= \frac{1}{\delta}\left(\mathbf{1}_q - \mathbb{H}^\top\mathbb{K}^{-1}\mathbf{1}_n - \boldsymbol{\pi}\boldsymbol{\lambda}'{}^\top\mathbf{u}\right) && \in \mathbb{R}^q
\end{aligned}$$

- *If the system of Equations* (9) *is of rank $n+p-1$, we remove arbitrarily the first constraint of the first equation and Lagrange multipliers become:*

$$\begin{aligned}
\boldsymbol{\lambda}' &= \left(\frac{\gamma_1}{\delta}\mathbf{u}\mathbf{u}^\top - \gamma\mathbb{Y}\mathbb{K}^{-1}\mathbb{Y}^\top\right)^{-1} \\
&\qquad\qquad \left(\mathbb{Y}\mathbb{K}^{-1}\mathbb{H}\boldsymbol{\pi} + \frac{1-\pi_1}{\delta}\mathbf{u} - \frac{1}{\delta}\mathbf{u}\mathbf{1}_n{}^\top\mathbb{K}^{-1}\mathbb{H}_1\boldsymbol{\pi}_1 - \mathbf{m}\right) && \in \mathbb{R}^p \\
\boldsymbol{\lambda}_1 &= \frac{1}{\delta}\left(\mathbf{1}_{q-1} - \mathbb{H}_1{}^\top\mathbb{K}^{-1}\mathbf{1}_n - \boldsymbol{\pi}_1\boldsymbol{\lambda}'{}^\top\mathbf{u}\right) && \in \mathbb{R}^{q-1} \\
\boldsymbol{\lambda} &= \begin{pmatrix} 0 \\ \boldsymbol{\lambda}_1 \end{pmatrix} && \in \mathbb{R}^q
\end{aligned}$$

where $\pi_1 := \pi_{x_1^\star} \in \mathbb{R}$, $\boldsymbol{\pi}_1 := \left(\pi_{x^\star 2}, ..., \pi_{x_q^\star}\right)^\top \in \mathbb{R}_+^{q-1}$, $\mathbf{u} := \mathbb{Y}\mathbb{K}^{-1}\mathbf{1}_n \in \mathbb{R}^p$, $\gamma := \boldsymbol{\pi}^\top\boldsymbol{\pi} \in \mathbb{R}$, $\gamma_1 := \boldsymbol{\pi}_1{}^\top\boldsymbol{\pi}_1 \in \mathbb{R}$, $\delta := \mathbf{1}_n{}^\top\mathbb{K}^{-1}\mathbf{1}_n \in \mathbb{R}$, $\mathbb{H} := \left[\mathbf{h}(x_1^\star), ..., \mathbf{h}(x_q^\star)\right] \in \mathcal{M}_{n\times q}(\mathbb{R})$, $\mathbb{H}_1 := \left[\mathbf{h}(x_2^\star), ..., \mathbf{h}(x_q^\star)\right] \in \mathcal{M}_{n\times(q-1)}(\mathbb{R})$.

Proof. The proof is postponed to Appendix. A.4, page 42. □

Note that it is also possible to compute a model with the only constraint $\mathbb{Y}\mathbb{A}\boldsymbol{\pi} = \mathbf{m}$, but without requiring weights summing to one. In view of further classification applications, we do not develop it here and keep both constraints.

Again, an originality is that the previous result can be expressed using compact expressions for $\mathbb{K}$ and $\mathbb{H}$ or more classical covariances, as stated in the following remark. The covariance functions that can be used to fill those matrices are detailed in Section 4.

Remark 2 (Covariance matrices with two constraints)**.** *Under the assumptions of Remark 1 and using the same notations, the matrices $\mathbb{K}$ and $\mathbb{H}$ can be replaced by $\widetilde{\mathbb{K}}$ and $\widetilde{\mathbb{H}}$ everywhere in Proposition 3, without changing the optimal weights $\mathbb{A}$.*

Proof. The proof is postponed to Appendix. A.5, page 46. □

Notice that the constraint on predicted values depends on prediction locations, which is the innovative aspect of this work. And the constraints become obviously too strong with a single predicted location; the prediction would be entirely prescribed. In practice, such a constraint is typically applied either on a given static grid of locations or on problems where the prediction locations are known (e.g., when the observed locations constitute a subset of some given finite set). Remark also that the optimal weights correspond to a global

minimum of some quadratic loss. Under the considered constraints, the optimal weights still corresponds to a global loss minimum under constraints, but of course the loss itself is increased, as an unconstraint optimisation can necessarily do as well or better than a constraint one. However, even if the observed loss is increased, the predictive performance under constraint can increase, if the constraints contain useful information: that will likely not be the case when considering adverse scenarii, but it can be the case when knowing some statistics at a upper scale, or external trustable information.

2.4 Optimal Weights With Affine Extension

A well-known characteristic of Simple Kriging is that the Kriging weights and the Kriging mean both tend to zero far from observed locations. In our setting, predicted values should be $\mathbf{m}$ on average. Hence, one may desire that predictions return to $\mathbf{m}$ far from the observed locations. This behaviour is similar to what one may expect from a Simple Kriging model applied on $\mathbf{Y}-\mathbf{m}$, where predictions' weights far from the observations tend to 0. However, it is important to keep in mind that since we want the sum of a prediction's weights to be equal to 1, it is incompatible with Simple Kriging with a null limit. We present in this section an affine extension of the Joint Kriging model, which is useful when one needs, at the same time, weights summing to 1 and a tunable behaviour far from the observations.

Up to this point, one has only considered linear predictors, where a predictor is a linear combination of observed responses $\mathbf{Y}(x_1), \ldots, \mathbf{Y}(x_n)$, under various constraints. We now consider the case where the prediction involves one additional term.

The constraint on predicted values in the Joint Kriging model suggests that there is an external source of information giving a hint on the prediction. In addition to the observations, one knows that predicted values should be $\mathbf{m}$ on average. This information may come, for instance, from some known overall statistics on the territory, some expert knowledge, or from an expectancy estimator. Let us denote $\mathbf{Z}$ the $p \times 1$ random vector containing this external source of information.

With this in mind, we define an affine prediction:

$$\mathbf{M}^{+}(x^\star) := \alpha_0(x^\star)\mathbf{Z} + \sum_{i=1}^{n} \alpha_i(x^\star)\mathbf{Y}(x_i), \tag{11}$$

given $\mathbf{Z} = \mathbf{m}$, a constant term is included in the sum, hence the name "affine prediction".

The sum of weights constraint on the new vector $\boldsymbol{\alpha}^{+} = (\alpha_0(x^\star), \ldots, \alpha_n(x^\star))$ can be written:

$$\mathbf{1}_{n+1}{}^{\top} \boldsymbol{\alpha}^{+}(x^\star) = 1.$$

This way, if the p components of $\mathbf{m}$ and $\mathbf{Y}(x_i)$, $i = 1, \ldots, n$ are probabilities summing to one, then the p components of the predictor $\mathbf{M}(x^\star)$ will also sum to one.

For the second constraint on average predicted values, previously detailed in Equation (7), there is an implicit conditioning by $\mathbf{Z} = \mathbf{m}$. This constraint may write, with $X^\star$ a r.v. defined on $\{x_1^\star, \ldots, x_q^\star\}$:

$$\mathrm{E}\left[\mathbf{M}^{+}(X^\star) \,\middle|\, \mathbf{Z} = \mathbf{m}, \mathbb{Y}\right] = \mathbf{m}. \tag{12}$$

Finally, provided covariances between $\mathbf{Z}$ and $\mathbf{Y}(x)$ are given for all $x \in \chi$, then the setting is the same as in previous Propositions 1, 2, and 3, except that one observation $\mathbf{Z} = \mathbf{m}$ is added in the vectors of observations $\mathbf{Y}(x_1), \ldots, \mathbf{Y}(x_n)$. The covariance matrices are also updated. This is detailed in the following Proposition.

Proposition 4 (Affine version of predictors)**.** *Assume that the following covariance vectors are given:*

$$\begin{aligned}\mathbf{P}^\top &:= \mathrm{E}\left[\mathbf{Z}^\top \mathbb{W}\mathbb{Y}\right] - \mathrm{E}\left[\mathbf{Z}^\top\right]\mathbb{W}\,\mathrm{E}\left[\mathbb{Y}\right],\\ \mathbf{Q}^\top &:= \mathrm{E}\left[\mathbf{Z}^\top \mathbb{W}\mathbb{Y}^\star\right] - \mathrm{E}\left[\mathbf{Z}^\top\right]\mathbb{W}\,\mathrm{E}\left[\mathbb{Y}^\star\right],\\ \sigma_Z^2 &:= \mathrm{E}\left[\mathbf{Z}^\top \mathbb{W}\mathbf{Z}\right] - \mathrm{E}\left[\mathbf{Z}^\top\right]\mathbb{W}\,\mathrm{E}\left[\mathbf{Z}\right].\end{aligned}$$

Then, affine predictors corresponding to the simple unconstrained case, to the ordinary case with one constraint, and to the case with two constraints can be obtained by replacing $\mathbb{Y}$*,* $\mathbb{K}$*, and* $\mathbb{H}$ *by*

$$\mathbb{Y}^+ = \begin{pmatrix}\mathbf{m} & \mathbb{Y}\end{pmatrix}, \quad \mathbb{K}^+ = \begin{pmatrix}\sigma_Z^2 & \mathbf{P}^\top\\ \mathbf{P} & \mathbb{K}\end{pmatrix}, \quad \mathbb{H}^+ = \begin{pmatrix}\mathbf{Q}^\top\\ \mathbb{H}\end{pmatrix},$$

in Propositions 1, 2, and 3 respectively.

Proof. The proof is straightforward, hence not appearing in the Appendix. □

Notice that the previous Proposition 4 can be easily extended to several sources of information: $\mathbf{Z}_1, \mathbf{Z}_2, \ldots$. For the sake of simplicity, this is not developed here.

As detailed in Section 4, the matrices $\mathbb{K}, \mathbb{H}$ can be derived from simple correlation functions. Now it remains to derive one expression for $\mathbf{P}$ and $\mathbf{Q}$.

Remark 3 (Extra covariances for affine prediction)**.** *Let* $\mathbf{P} = (P_1, \ldots, P_n)$, $\mathbf{Q} = (Q_1, \ldots, Q_q)$, *and* $\mathbf{Z} = (Z_1, \ldots, Z_p)$.

Let us assume that $\mathbf{P}$ *and* $\mathbf{Q}$ *do not depend on* x_i *nor on* $x_j^\star$*, which means that the general source of information informs about the whole process, not about a particular location. then one can propose*

$$\begin{aligned}P_i &= \rho\sigma\sigma_Z, \quad i = 1, \ldots n\\ Q_j &= \rho\sigma\sigma_Z, \quad j = 1, \ldots q\end{aligned}$$

This happens, for instance, when $Y_k(x) = \rho\frac{\sigma}{\sigma_Z}Z_k + G_k(x)$*,* $k = 1, \ldots, p$*, where all* $G_k(x)$ *are independent of all* Z_k*.*

The parameter $\rho \in [-1, 1]$ *measures how redundant the information provided by* $\mathbf{Z}$ *is, and can even be set to* 0 *if one considers that the external information source is completely independent of observations. The parameter* σ_Z *measures how certain the external information is: when* σ_Z *is high, the added information cannot be trusted, and one retrieves the linear predictor; when* σ_Z *is low, the added information is trustable, so that far from observed locations,* $\mathbf{M}(x)$ *gets nearer to* $\mathbf{m}$*. In practice, one can set* $0 < \sigma_Z \ll \sigma$ *to see the maximal difference with the linear predictor. One can even optimise this parameter* σ_Z *to smoothly switch from a linear to an affine model.*

Other assumptions can be made, leading to different vectors $\mathbf{P}$ *and* $\mathbf{Q}$*.*

Notice that the affine method can be applied with or without constraints, it is just a supplementary term that allows to tune the behaviour of the predictor far from observations. Far from observations, all the weights in Equation (11) tend to predict the external source of information $\mathbf{Z}$. By choosing specific values of $\mathbf{Z}$, the default behaviour of the output variables is tunable. The affine predictor hence satisfies both weights summing to one and the default limit of the output variables, which is chosen here to be $\mathbf{Z} = \mathbf{m}$.

2.5 Joint Kriging Mean and Variance

In this subsection, we derive the mean predictor and the prediction error, assuming the optimal weights have been calculated with chosen constraints, as detailed in previous subsections.

Consider $\mathbf{M}(x^\star)$ and $\boldsymbol{\alpha}(x^\star)$ a Joint Kriging predictor and the associated weights with or without constraints. In the following, we call *Joint Kriging mean* the value of the predictor $\mathbf{M}(x^\star)$ and *Joint Kriging variance* the value of the quadratic error $\Delta(x^\star)$. Let us recall that:

$$\begin{aligned}\mathbf{M}(x^\star) &:= \mathbb{Y}\boldsymbol{\alpha}(x^\star),\\ \Delta(x^\star) &:= \mathrm{E}\left[\|\mathbf{M}(x^\star) - \mathbf{Y}(x^\star)\|^2_{\mathbb{W}}\right].\end{aligned}$$

where $\mathbb{Y} = [\mathbf{Y}(x_1), \ldots, \mathbf{Y}(x_p)]$ is the $p \times n$ matrix of observations. If $p = 1$ and if $\mathbb{W}$ is the identity matrix, Joint Kriging mean and Joint Kriging variance are exactly the Kriging mean and the Kriging variance usually known in Kriging.

The following Proposition gives a closed formula to compute the Joint Kriging variance.

Proposition 5 (Joint Kriging variance with arbitrary weights)**.** *Let $\boldsymbol{\alpha}(x^\star)$ be any vector of weights, possibly satisfying supplementary constraints. The associated Joint Kriging variance writes:*

$$\Delta(x^\star) = \boldsymbol{\alpha}(x^\star)^\top \mathbb{K}\boldsymbol{\alpha}(x^\star) - 2\boldsymbol{\alpha}(x^\star)^\top \mathbf{h}(x^\star) + v(x^\star), \tag{13}$$

or using a matrix expression, denoting $\boldsymbol{\Delta} := (\Delta(x_1^\star), \ldots, \Delta(x_1^\star))^\top$, we get

$$\boldsymbol{\Delta} = \mathrm{diag}\left[\mathbb{A}^\top \mathbb{K}\mathbb{A}\right] - 2\mathrm{diag}\left[\mathbb{A}^\top \mathbb{H}\right] + \mathrm{diag}[\mathbb{K}^\star],$$

where

$$\begin{aligned}\mathbb{K} &:= \mathrm{E}\left[\mathbb{Y}^\top \mathbb{W}\mathbb{Y}\right] && \in \mathcal{S}_n^+(\mathbb{R}),\\ \mathbb{K}^\star &:= \mathrm{E}\left[\mathbb{Y}^{\star\top} \mathbb{W}\mathbb{Y}^\star\right] && \in \mathcal{S}_q^+(\mathbb{R}),\\ \mathbf{h}(x^\star) &:= \mathrm{E}\left[\mathbb{Y}^\top \mathbb{W}\mathbf{Y}(x^\star)\right] && \in \mathbb{R}^n,\\ v(x^\star) &:= \mathrm{E}\left[\mathbf{Y}(x^\star)^\top \mathbb{W}\mathbf{Y}(x^\star)\right] && \in \mathbb{R}\end{aligned}$$

are assumed to be known. $\mathrm{diag}[.]$ *is the vector whose entries are the diagonal of the considered matrix.*

Proof. The proof is postponed to Appendix. A.6, page 47. □

Note that the above Proposition 5 can be directly adapted to the affine case of Proposition 4 by replacing $\mathbb{Y}, \mathbb{K}, \mathbb{H}$ by $\mathbb{Y}^+, \mathbb{K}^+, \mathbb{H}^+$, v being unchanged: one can interpret the predictor to be a linear predictor with one more observation, with correct covariances.

As previously stated in Remarks 1 and 2, and using the same notation, one can replace $\mathbb{K}, \mathbb{H}, \mathbf{h}$ with $\widetilde{\mathbb{K}}, \widetilde{\mathbb{H}}, \widetilde{\mathbf{h}}$, provided that the following new quantities are defined:

$$\begin{aligned}\widetilde{\mathbb{K}}^\star &= \mathrm{E}\left[\mathbb{Y}^{\star\top} \mathbb{W}\mathbb{Y}^\star\right] - \mathrm{E}\left[\mathbb{Y}^{\star\top}\right] \mathbb{W}\, \mathrm{E}\left[\mathbb{Y}^\star\right]\\ \widetilde{v}(x^\star) &= \mathrm{E}\left[\mathbf{Y}(x^\star)^\top \mathbb{W}\mathbf{Y}(x^\star)\right] - \mathrm{E}\left[\mathbf{Y}(x^\star)^\top\right] \mathbb{W}\, \mathrm{E}\left[\mathbf{Y}(x^\star)\right].\end{aligned}$$

The result is stated in Remark 4 below. Hence, in practice, all these covariances can be filled using a given covariance function $k(x, x')$, under suitable assumptions, as detailed in Section 4.

Remark 4 (Covariance matrices in Joint Kriging mean and variance)**.** *Under the assumptions of Remark 1 and using the same notation, the matrices* $\mathbb{K}$, $\mathbb{H}$, $\mathbb{K}^\star$, *the vector* $\mathbf{h}(x^\star)$ *and the scalar* $v(x^\star)$ *can be replaced by* $\widetilde{\mathbb{K}}$, $\widetilde{\mathbb{H}}$, $\widetilde{\mathbb{K}}$, $\widetilde{\mathbf{h}}(x^\star)$ *and* $\widetilde{v}(x^\star)$ *everywhere in Proposition 5, without changing the Joint Kriging mean and variance.*

Proof. The proof is postponed to Appendix. A.7, page 47. □

Now, remark that Proposition 5 gives only an overall error:

$$\Delta(x^\star) := \mathrm{E}\left[\|\mathbf{M}(x^\star) - \mathbf{Y}(x^\star)\|_{\mathbb{W}}^2\right]$$

which is a weighted sum of errors over all components of $\mathbf{M}(x^\star)$.

This is a strength of the method since the quantity to optimise is real-valued, which allows using standard covariance functions as detailed in Example 3. This is also an important limitation, because in practice, one surely needs prediction errors for each component of $\mathbf{M}(x)$:

$$\delta_i(x^\star) := \mathrm{E}\left[\|M_i(x^\star) - Y_i(x^\star)\|^2\right], \quad i = 1, \ldots, p.$$

The following Proposition 6 shows that one can get this error $\delta_i(x^\star)$ for each component $i = 1, \ldots, p$. It relies on a supplementary assumption on the matrix $\mathbb{W}$, but this assumption is only useful for determining the confidence bands for each component of the predictor $\mathbf{M}(x)$, not for computing $\mathbf{M}(x)$ itself.

Proposition 6 (Variance sharing)**.** *Assume that transformed observations* $\widetilde{\mathbf{Y}}(x) := \mathbb{W}^{1/2}\mathbf{Y}(x)$ *are such that components of* $\widetilde{\mathbf{Y}}$ *are uncorrelated and bear the same share of the covariance function* k*, that is to say:*

$$\mathrm{Cov}\left[\widetilde{Y}_i(x), \widetilde{Y}_j(x')\right] = \frac{1}{p}k(x, x')\mathbb{1}_{\{i=j\}}, \quad i, j \in \{1, \ldots, p\},\ x, x' \in \chi,$$

Assume also that $\mathrm{E}[\mathbf{Y}(x)] = \boldsymbol{\mu}$ *for all* $x \in \chi$*. Furthermore, assume that either the weights sum to one or* $\boldsymbol{\mu} = \mathbf{0}_p$*. Then, the local errors write:*

$$\delta_i(x^\star) = \frac{\sigma_i^2}{\sigma^2}\Delta(x^\star), \quad i = 1, \ldots, p. \tag{14}$$

where $\sigma_i^2 := \mathrm{Var}[Y_i(x)]$ *is the variance of the component* $Y_i(x)$*, assumed to be constant over* x*.*

Proof. The proof is postponed to Appendix. A.8, page 47. □

The result of Proposition 6 states that for a well-chosen matrix $\mathbb{W}$, the error $\delta_i(x^\star)$ is proportional to the unit global error $\sigma^{-2}\Delta(x^\star)$: one has to apply the variance σ_i^2 of the component instead of the variance σ^2 of the aggregated weighted components.

3 Constrained Classification

In this section, we now apply multi-output prediction to membership degrees for fuzzy classification. We show that the Joint Kriging predictor, together with constraints on weights and predicted values, is especially suited to this task, and the above constraints make sense in a classification setting.

3.1 Prescribed Constraints

We aim here at proposing a fuzzy classification with a prescribed average of predicted membership degrees. Either because one requires that predicted values are overall distributed like the observed ones or because an external source of information gives the expected label percentages on a higher scale. It can be the case for a regional study, knowing some statistics at a national level. It can also be used for modelling adverse scenarios, as discussed in the Introduction of this paper.

Consider a classification problem with p possible labels. Labels depend on some explanatory variables $x \in \chi$, so that one may observe labels $\ell(x_1), \ldots, \ell(x_n)$ taking values in $\{1, \ldots, p\}$. Assume that, at a prediction point $x^\star$, a fuzzy classification method provides membership degrees of the p classes, gathered in a vector $\mathbf{M}(x^\star)$. Consider q prediction points $x_1^\star, \ldots, x_q^\star$, and bind all predicted membership degrees in a $p \times q$ matrix:

$$\mathbb{M} := \left[\mathbf{M}(x_1^\star), \ldots, \mathbf{M}(x_q^\star)\right].$$

Components of $\mathbf{M}(x^\star)$ should be positive and sum to one at each prediction point $x^\star$, and the weighted average of predictions $\mathbf{M}(x_1^\star), \ldots, \mathbf{M}(x_q^\star)$ is prescribed. As a result, we must satisfy positivity and the following system of constraints:

$$\begin{cases} {\mathbf{1}_p}^\top \mathbb{M} = {\mathbf{1}_q}^\top & \text{(probabilistic constraint)} \\ \mathbb{M}\boldsymbol{\pi} = \mathbf{m} & \text{(higher-scale constraint)} \end{cases} \tag{15}$$

where $\boldsymbol{\pi}$ and $\mathbf{m}$ are two vectors of positive weights summing to one (i.e., two distributions). Hence, the set of predictions is subject to both constraints on the prescribed sum of rows and the prescribed sum of columns.

In Table 1, we show an example of a confusion matrix, deriving from the previous constraints: the distribution of predicted classes is chosen to be identical to the one of actual classes, assumed to be given (or estimated). This is especially useful in situations where a class is dominant: all models tend to predict this dominant class, ensuring good accuracy, but the study of other classes thus becomes very difficult. The constraint forces the model to predict the right class probabilities, providing a way to study rarer classes.

Actual Classes \ Predicted Classes	A	B	C	D	Sum
A	52.3	37	20.5	10.2	120
B	23.6	65.4	44.9	16.1	150
C	37	38.4	72.9	21.7	170
D	7.1	9.2	31.7	42	90
Sum	120	150	170	90	530

Table 1: **A Constrained Confusion Matrix:** the sum of predicted classes, i.e., the sum of predicted membership degrees, is, here, equal to the sum of actual classes. For example, knowing that one must predict 120 labels A (higher-scale constraint), the sum of predicted membership degrees for the actual class A is forced to be exactly 120.

3.2 Application of the Joint Kriging Model

We have considered specific constraints, such as higher-scale constraints. We show here that other predictors are usually not suited to satisfy such constraints, but that the Joint Kriging model naturally satisfies them.

Predictors of the literature may be unsuited to deal with considered constraints: at an unobserved location $x^\star$, for a predictor $L(x^\star) \in \{1, \dots, p\}$ of a label $\ell(x^\star)$, the reader may convince himself that, for given probabilities p_j, $j \in \{1, \dots, p\}$, constraints on predicted classification such as

$$\mathrm{P}\left[L(X^\star) = j \,|\, \text{observed labels}\right] = p_j,$$

are not so easy to handle, even if $X^\star$ is a uniformly distributed random variable over prediction points $x_1^\star, \dots, x_q^\star$. This is because such constraints usually act in a non-linear way on the predictor $L(X^\star)$, and the predictor $L(.)$ itself can be some complicated function of observed labels. Existing predictors, such as Indicator Kriging, may be unable to deal with such constraints. Furthermore, they may not be appropriate in cases where the considered labels do not correspond to ordinal classes.

Now, let us adapt the classification problem to the Joint Kriging model. In a classification problem, each label $\ell \in \{1, \dots, p\}$ can be converted into a $p \times 1$ vector of indicator functions, namely

$$\mathbf{Y} := \left(\mathbb{1}_{\{j=\ell\}}\right)_{j=1,\dots,p}.$$

This transformation is well known in the machine learning community as *label binarisation* (see also dummy variables or one-hot encoding), and is implemented in many languages. It also appears in some contexts of multiple outputs (Alvarez et al., 2012, Section 3.1). With this representation, the equality ${\mathbf{1}_p}^\top \mathbf{Y} = 1$ is verified.

In practice, it is common to observe true label values depending on some explanatory variables $x \in \chi$. But it may also happen that one observes uncertain labels: multiple and distinct observed labels for the same $x \in \chi$, uncertainty in the value of x, etc. To handle this problem, one generalises slightly the previous label binarisation: one assumes here that observations consist in a distribution of possible labels, so that one observes n vectors $\mathbf{Y}(x_1), \dots, \mathbf{Y}(x_n)$, such that the components of each vector are summing to one: ${\mathbf{1}_p}^\top \mathbf{Y}(x_i) = 1$, $i = 1, \dots, n$. In other words, the p components of $\mathbf{Y}(x_i)$ represent the membership degrees of the p possible classes at an observed location x_i, $i = 1, \dots, n$. Using the previous notation, recall that $\mathbb{Y} := [\mathbf{Y}(x_1), \dots, \mathbf{Y}(x_n)]$, so that observed membership degrees satisfy

$${\mathbf{1}_p}^\top \mathbb{Y} = {\mathbf{1}_n}^\top. \tag{16}$$

Finally, using a Joint Kriging model, one can infer the membership degree of an unobserved location $x^\star$ using the predictor of Equation (1):

$$\mathbf{M}(x^\star) := \sum_{i=1}^{n} \alpha_i(x^\star)\mathbf{Y}(x_i) \tag{17}$$

The next remark details the impact of both constraints, weights summing to one (probabilistic constraint) and prescribed average predicted values (higher-scale constraint), in the particular setting of membership degrees that are summing to one. It shows that the Joint Kriging model naturally satisfies the considered system of constraints in the fuzzy classification setting.

Remark 5 (Constraints' impact)**.** *Consider the membership degree assumption given in Equation* (16)*,* $\mathbf{1}_p{}^\top \mathbb{Y} = \mathbf{1}_q{}^\top$*. Consider also the two previous constraints on weights and predicted values, namely the constraints of Equation* (6) *and Equation* (7)*. Then the Joint Kriging model implies that:*

- *Predicted membership degrees are summing to one:*

$$\mathbf{1}_p{}^\top \mathbf{M}(x^\star) = 1,$$

  for all prediction points $x^\star \in \chi$*. In particular,* $\mathbf{1}_p{}^\top \mathbb{M} = \mathbf{1}_q{}^\top$*.*

- *The average membership degree over prediction points can be chosen:*

$$\mathrm{E}\left[\mathbf{M}(X^\star) \,|\, \mathbb{Y}\right] = \mathbf{m},$$

  where $\mathbf{m}$ *is a prescribed average of predicted membership degrees of each class, with* $\mathbf{1}_p{}^\top \mathbf{m} = 1$*, and* $X^\star$ *a random variable over all prediction points.*

Proof. The proof is postponed to Appendix. A.9, page 49. □

3.3 Positivity Requirement

As noticed before, although predicted membership degrees are summing to one, there is no guarantee of positivity for the predicted values yet. We discuss here how to deal with this problem.

For hard clustering, it should be noted that, even without a positive weights requirement, predictions can still be used, as they can be interpreted as more general membership scores. It is quite natural to predict a class by selecting the one with the highest membership score. As an example, in a two-class problem, this leads to choosing one class or another if the associated prediction is greater than 0.5 or not, and even more so when the prediction is below 0 or greater than 1.

For fuzzy clustering, it is, of course, highly desirable to force the positivity of the Kriging weights. This way, when summing to one, the weights belong to a simplex, ensuring that predictions can be seen as probabilistic membership degrees. In practice, as is the case with numerous machine learning methods, a post-treatment of results may be required in specific cases involving negative membership degrees (see, e.g., the use of the softmax function with neural networks). With such a positivity requirement, the prediction falls within the framework of compositional data analysis, often treated by transformations of the observations and the predictions (Chiles & Delfiner, 2012, Section 5.7.4). Recent papers on this topic are Clarotto et al., 2022; Martínez-Minaya and Rue, 2023. Some usual

transformations require the strict positivity of values and are thus unsuited to the one-hot encoding that is used here. Hence, the set of usable transformations is restricted to the very recent literature on the topic.

A more prominent difficulty in our setting is that applying a transformation on weights in order to keep them in the $[0, 1]$ interval would alter the prescription of the predictions' expectation, so that it is not so straightforward to prescribe $\mathbf{m}$ and to ensure the weights to be positive and summing to one, as higher-scale constraints of Section 3.1. Furthermore, the distance to be minimised would also be altered, and the predictor would lose some minimal variance properties. Another approach would be to do an optimisation with such a positivity constraint, leading to quadratic programming (Chiles & Delfiner, 2012, Section 3.9.1), but losing the closed-form expressions that are presented here.

In the numerical illustrations presented in Section 5, we have chosen another approach: we found empirically that adding a small nugget effect (i.e., adding a constant to the diagonal of the covariance matrix $\mathbb{K}$) was sufficient to ensure the positivity of weights when needed. Indeed, this empirical finding is supported by the following Proposition 7. Furthermore, in our investigations of the numerical Section 5, the added nugget effects were small enough, so that prediction accuracies were not significantly altered.

Proposition 7 (Nugget ensuring positive weights)**.** *Assume that $\mathbb{K}$ is replaced by $\mathbb{K}_{nug} := \mathbb{K} + \eta\mathbb{I}_n$ in Proposition 2 and Proposition 3, where $\eta > 0$ is a nugget parameter and $\mathbb{I}_n$ is the identity matrix of size n. Assume furthermore that the prescribed vector $\mathbf{m} = \frac{1}{n}\mathbb{Y}\mathbf{1}_n$ in the latter Proposition 3, so that $\mathbf{m}$ contains the proportion of each label in the observations. Then, for the predictors given by both Propositions 2 and 3, there exists a nugget η large enough ensuring that all weights in $\mathbb{A}$ are positive and summing to one.*

Proof. The proof is postponed to Appendix. A.10, page 49. □

In practice for the classification problem, it means that adding a sufficient nugget effect ensures the positivity of weights so that the predictions can be considered as membership degrees, summing to one and positive.
This approach was sufficient for the tested numerical illustrations. Otherwise, optimisation under the positivity constraint would still be possible, but would require either the use of quadratic programming or the extension of recent methods to higher-scale constraints.

4 Filling Cross-Covariances

In the previous sections, we have seen the Joint Kriging model and its applications to constrained classification. The assumptions on the underlying random fields were very general: dependent components of $\mathbf{Y}(.)$, with the existence of the first two cross-moments, without any Gaussian assumption. The obtained results were derived from specific covariances, in particular in matrices $\mathbb{K}$ and $\mathbb{H}$. In the present section, we discuss practical strategies that can be used to fill the needed covariance matrices.

The main result of previous sections is the expression of optimal weights, with weights summing to one and a constraint on average predicted values, with an affine extension. However, despite a rather general model, the predictor is simplified, with weights applying jointly to all components in Equation (1). The application to constrained classification is justifying this simplifying assumption: indeed, applying different weights to components of $\mathbf{Y}(.)$ would make it far more difficult to preserve classification higher-scale constraints, as presented in Section 3.1 and Table 1. To the authors knowledge, existing more general multi-output models are not conceived to handle such higher-scale constraints.

It is worth mentioning that, once the Joint Kriging simplifying assumption is accepted, there is no obstacle to using the cross-covariance function of any multi-output process $\mathbf{Y}(.)$, even a non-stationary one. In particular, the reader may refer to the books Wackernagel, 2003, Chapter 20 and Chiles and Delfiner, 2012, Chapter 5 for general considerations about cross-covariances for multivariate models. Important articles on the topic are Alvarez et al., 2012 and Genton and Kleiber, 2015, where the estimation is also discussed.

In order to fill the matrices $\widetilde{\mathbb{K}}$ and $\widetilde{\mathbb{H}}$, for a given positive definite matrix $\mathbb{W}$, we need a function $k(.,.)$ that gives:

$$k(x,x') := \mathrm{E}\left[\mathbf{Y}(x)^\top \mathbb{W}\mathbf{Y}(x')\right] - \mathrm{E}\left[\mathbf{Y}(x)^\top\right]\mathbb{W}\,\mathrm{E}\left[\mathbf{Y}(x')\right]. \tag{18}$$

Then the elements of the covariance matrices $\widetilde{K}$ and $\widetilde{H}$ can be derived from the covariance function $k : \chi \times \chi \to \mathbb{R}$ by setting:

$$\widetilde{\mathbb{K}}_{ij} = k(x_i, x_j) \tag{19}$$

$$\widetilde{\mathbb{H}}_{ik} = k(x_i, x_k^\star) \tag{20}$$

Denoting

$$c_{i,j}(x,x') := \mathrm{Cov}\left[Y_i(x), Y_j(x')\right],$$

one can derive:

$$k(x,x') = \sum_{i=1}^{p}\sum_{j=1}^{p} c_{i,j}(x,x')\mathbb{W}_{i,j}.$$

By gathering covariances in a matrix, we denote:

$$\mathbb{C}(x,x') = [c_{i,j}(x,x')]_{\substack{i\in\{1,\dots,p\}\\ i\in\{1,\dots,p\}}} \in \mathcal{S}_p^+(\mathbb{R}).$$

One can also check that the covariance function $k(x,x')$ is equal to the trace of the transformed process $\mathbb{W}^{1/2}Y(.)$ cross-covariances:

$$k(x,x') = \mathrm{Tr}\left[\mathbb{W}^{1/2}\mathbb{C}(x,x')\mathbb{W}^{1/2\top}\right].$$

If all $c_{i,j}(.,.)$ are known, then the latter quantity can be used to fill all needed covariance matrices. It can however rely on many parameters, as each $c_{i,j}$ can come with its own hyperparameters, namely $\mathcal{O}\left(p^2\right)$ hyperparameters.

Through the following examples, we investigate the link with several classical cross-covariance models.

Example 1 (Separable cross-covariances)**.** *Let us consider the simplifying assumption of multi-output separable kernel functions (Alvarez et al., 2012, Section 4), for which there exists a covariance function $c(x,x')$ and a $p \times p$ real matrix $\mathbb{S} = [S_{i,j}]_{\substack{i\in\{1,\dots,p\}\\ j\in\{1,\dots,p\}}}$, such that:*

$$c_{i,j}(x,x') = S_{i,j}c(x,x'), \quad (i,j) \in \{1,\dots,p\}^2.$$

Then

$$k(x,x') = c(x,x')\sum_{i=1}^{p}\sum_{j=1}^{p} S_{i,j}\mathbb{W}_{i,j}.$$

One sees that the role played by the matrix $\mathbb{W}$ is quite similar to the one played by the separability matrix $\mathbb{S}$, and that, in this simplified setting, $k(x,x')$ is also proportional to the driving covariance function $c(x,x')$.

Example 2 (Linear model of coregionalisation)**.** *The Linear Model of Coregionalisation (LMC) is a classical approach for combining several univariate covariances, see Chiles and Delfiner, 2012, Section 5.6.4 and Genton and Kleiber, 2015, Sections 2.1 and 4. Let* $\mathbf{R}(x, x') = \text{diag}[\rho_1(x, x'), \ldots, \rho_r(x, x')]$ *be a diagonal matrix of* r *univariate correlation functions. Assuming that the output variables are generated by a linear combination of* r *independent univariate random fields with correlation functions* ρ_i, $i \in \{1, \ldots, r\}$, *the LMC combines the univariate covariances in* $\mathbf{R}$ *by setting:*

$$\mathbb{C}(x, x') = \mathbb{B}\mathbf{R}(x, x')\mathbb{B}^\top, \tag{21}$$

where $\mathbb{B}$ *is a* $p \times r$ *full rank matrix. As a consequence, we get:*

$$k(x, x') = \text{Tr}\left[\mathbb{W}^{1/2}\mathbb{B}\mathbf{R}(x, x')\mathbb{B}^\top{\mathbb{W}^{1/2}}^\top\right].$$

One sees that the role played by the matrix $\mathbb{W}^{1/2}$ *is quite similar to the one played by the LMC matrix* $\mathbb{B}$*, especially in the case where* $r = p$*.*

Whatever the underlying model of cross-covariances $c_{i,j}(x, x')$, the model uses a single covariance function $k(x, x')$, which can be seen as the covariance function of a weighted sum of all output variables. This is due to the simplifying assumption and the optimisation of a single scalar error. The next example shows that in some cases, the latter covariance function depends on fewer parameters than the whole set of cross-covariance functions $\{c_{i,j}(x, x') : 1 \leq i, j \leq p\}$.

Example 3 (Isotropic k)**.** *Let us recall Equation* (18)*:*

$$k(x, x') := \text{E}\left[\mathbf{Y}(x)^\top\mathbb{W}\mathbf{Y}(x')\right] - \text{E}\left[\mathbf{Y}(x)^\top\right]\mathbb{W}\,\text{E}\left[\mathbf{Y}(x')\right].$$

Let us assume that there exists a positive definite matrix $\mathbb{W}$ *such that the covariance function* $k(.,.)$ *is isotropic. That is to say,* $k(x, x')$ *depends only on some distance between* x *and* x'*. Then* k *can be written in a simplified form, so one does not need to estimate* $\mathbb{W}$*.*

In this case, with one variance hyperparameter $\sigma^2 > 0$ *and* d *positive hyperparameters* $\theta_1, \ldots, \theta_d$*, usually referred to as characteristic length scales (see C. E. Rasmussen & Williams, 2006, page 14), one can set, for example:*

$$k(x, x') = \sigma^2 r_0\left(\|x - x'\|_{\boldsymbol{\theta}}\right),$$

where r_0 *is a correlation function and* $\|x - x'\|^2_{\boldsymbol{\theta}} = \sum_{i=1}^d \left(\frac{x_i - x'_i}{\theta_i}\right)^2$ *is a rescaled Euclidean norm. Notice that this expression does not depend on* $\mathbb{W}$ *any more, so that when using the above assumption, we do not have to estimate* $\mathbb{W}$*.*

As a consequence of the previous Example 3, for a given matrix $\mathbb{W}$, a noticeable advantage of the Joint Kriging method is the possibility to use a limited number of hyperparameters that need to be optimised. In that case, despite the multivariate output of the $p \times 1$ response vector $\mathbf{Y}(x)$, $x \in \chi$, there are only a few hyperparameters required for defining the covariances: for instance, σ^2, $\boldsymbol{\theta}$, and the covariance family. This is quite different from co-Kriging techniques where all cross-covariances between components $Y_i(x)$ and $Y_j(x')$ should be defined for all $i, j \in \{1, \ldots, p\}$, and $x, x' \in \chi$, which ends up in an order of $O(p^2)$ covariance functions and many associated hyperparameters. Furthermore, the method satisfies all prescribed constraints.

The relaxation of the simplifying assumption of Equation (1) would surely exploit more precisely the cross-covariance structure of output variables and increase the generality and accuracy of the model. However, the preservation of higher-scale constraints makes the use of other existing models in the literature difficult.

5 Numerical Illustrations

In this section, one considers different numerical illustrations for both prediction and classification. The first illustration focuses on the impact of constraints with one output, and the second one on the behaviour of the predictor with multiple outputs. The third illustration gives an application to classification and a benchmark with numerous competitors. All the illustrations are created in `R markdown` notebooks, one per subsection, available as online supplementary material at https://gitlab.com/urbs-imope/rdscripts/jointkrigingsupplementary (Grossouvre & Rullière, 2023). Notebooks are given in both an executable format and an executed `html` format. The presented figures are directly extracted from the notebooks, and the results are fully reproducible.

5.1 A Simplified Toy Example

One considers here the very simple case where there is a single output variable: the output $\mathbf{Y}(x)$ is belonging to $\mathbb{R}^p$, with $p = 1$. The interest of testing the Joint Kriging with one single output variable is to discuss the impact of the constraint on predicted values and the impact of the affine prediction. For $p = 1$, Simple Joint Kriging and Ordinary Joint Kriging are identical to common Simple Kriging and Ordinary Kriging, but the constraint on predicted values leads to a new original predictor. We keep here the vector bold font for vectors $\mathbf{Y}(x) \in \mathbb{R}^p$ and $\mathbf{m} \in \mathbb{R}^p$, even though $p = 1$, in order to keep the very same notation as in the rest of the paper.

Let us consider that the process $\mathbf{Y}(x)$ aims at approximating a hidden function, with say $a = 1$ and $b = 4$,

$$f(x) := a + \sin(x/b).$$

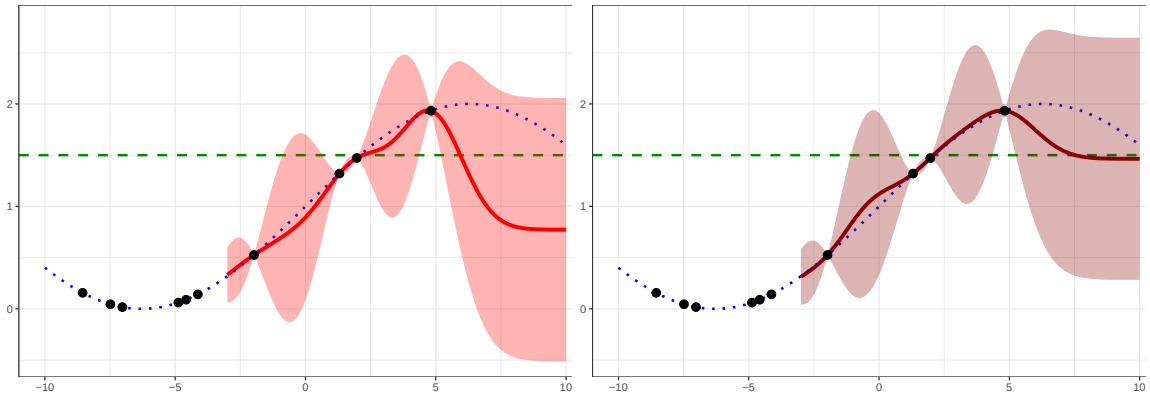
The observed locations $x_1, \ldots, x_n$ are randomly chosen with a uniform distribution over the interval $[-10, 5]$, and q prediction locations $x_1^\star, \ldots, x_q^\star$ are chosen regularly spaced over the interval $[-3, 10]$. Both intervals are purposely shifted so that some prediction points are far from observations, and vice versa. It also seeks to illustrate how the constraint on average predictions is affected by the prediction sites. Observed responses in $\mathbb{R}^p$, with $p = 1$, are $\mathbf{Y}(x_i) = f(x_i)$, $i = 1, \ldots, n$, with $n = 10$. Prediction is made over a set of $q = 100$ points. One defines $X^\star$ as a discrete and uniform random variable over all prediction points. The purpose here is not to interpolate as precisely as possible the hidden function f given a few observations, but only to illustrate the differences between various possible interpolators and the impact of requiring a prescribed average values for predicted values.

The prescribed value for $\mathbf{m} \in \mathbb{R}^p$, with $p = 1$, is the scalar $\mathbf{m} = 1.5$. The covariances between $\mathbf{Y}(.)$ are modelled as prescribed in Example 3, from a single covariance function, using a squared exponential kernel. One could also pick a kernel that reflects f periodicity. However, the purpose is not to make the best possible prediction but, rather, to understand the impact of various constraints.

$$\mathrm{Cov}\left[\mathbf{Y}(x), \mathbf{Y}(x')\right] = k(x, x') = \sigma^2 \exp\left(-\frac{(x - x')^2}{2\theta^2}\right).$$

We set $\sigma^2 = 0.6$ mainly for the visibility of the confidence band in the presented figures, and $\theta = 1.2$.

In Figure 1, one exclusively considers the constraint of sum of weights, which is assumed to be one: ${\mathbf{1}_n}^\top \boldsymbol{\alpha} = 1$. The predictor $\mathbf{M}(x)$ appears in a thick red line, together with confidence intervals built from the variance $\boldsymbol{\Delta}(x)$.



(a) One constraint, linear

(b) One constraint, affine

Figure 1: Prediction with one constraint: weights summing to one. Left: linear predictor, and right: affine predictor. In both cases, the average of the predicted values is distinct from the prescribed value $\mathbf{m} = 1.5$ (horizontal dashed line). The observations are the black dots. The thin, dotted, blue line is the underlying function. In the right panel, one applies the assumption in Remark 3 with $\rho = 0$ and $\sigma_Z = \sigma/10$.

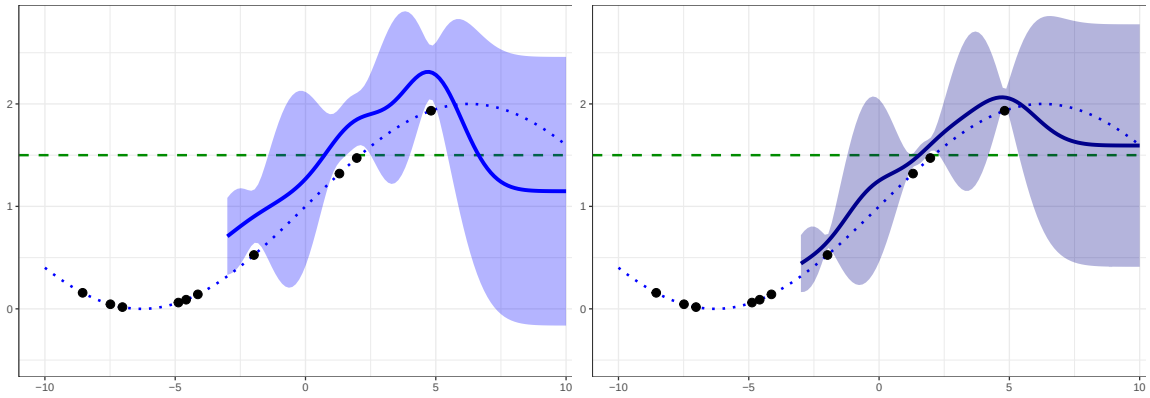
Figure 1a presents the result of ordinary Kriging exposed in Proposition 2. As is well known, when the location x is large (and far from observed locations), the ordinary Kriging mean tends to return to the estimated mean of the observations. The average value of the Kriging mean $\mathrm{E}\left[\mathbf{M}(X^\star) \,|\, \mathbb{Y}\right] \simeq 1.12$ is different from the value $\mathbf{m} = 1.5$ (horizontal dashed line), which is natural as this constraint has not been taken into account yet.

Figure 1b uses the Proposition 4 to add a supplementary affine term to the linear combination, while preserving the sum of weights equal to one. The affine term is derived from a random variable $\mathbf{Z}$, and we choose $\sigma_Z = \sigma/10$ so that this external information is assumed to be trustworthy (small variance). Given $\mathbf{Z} = \mathbf{m}$, the consequence is that, far from observed locations, the prediction tends to put all weight on this external source of information, so that the prediction gets closer to $\mathbf{m}$, as one can see at the extreme right of Figure 1b. This also makes the average $\mathrm{E}\left[\mathbf{M}(X^\star) \,|\, \mathbb{Y}\right] \simeq 1.37$ closer to $\mathbf{m} = 1.5$, but the values of those two quantities remain distinct. Another consequence of the affine term is the reduction of the confidence band width, as a new source of information has been added.

In Figure 2, one considers both the constraint of sum of weights, which is assumed to be one: ${\mathbf{1}_n}^\top \boldsymbol{\alpha} = 1$, together with the prescribed average of predicted values $\mathrm{E}\left[\mathbf{M}(X^\star) \,|\, \mathbb{Y}\right] = \mathbf{m}$. The predictor $\mathbf{M}(x)$ appears in a thick blue line, together with confidence intervals built from the variance $\Delta(x)$.

Figure 2a presents the result of ordinary Kriging exposed in Proposition 3. The average value of the Kriging mean $\mathrm{E}\left[\mathbf{M}(X^\star) \,|\, \mathbb{Y}\right] = 1.5$ is exactly the prescribed one $\mathbf{m} = 1.5$ (horizontal dashed line), which is natural as this constraint has been taken into account during the joint optimisation of all $\boldsymbol{\alpha}(x_j^\star)$, $j = 1, \ldots, q$. However, the predictor is no longer interpolating. This is logical: if $q = 1$, one has one only prediction point $x_1^\star$, and the constraint $\mathrm{E}\left[\mathbf{M}(X^\star) \,|\, \mathbb{Y}\right] = \mathbf{m}$ becomes $\mathbf{M}(x_1^\star) = \mathbf{m}$, which is distinct from an observation $\mathbf{Y}(x_n)$, even if $x_1^\star$ gets closer to x_n. Another example: if on the one hand observation points and prediction points are the same, if on the other hand $\mathbf{m}$ is not the average value of observations, then at least one prediction must be different from the associated observation to satisfy the constraint.

Figure 2b uses Proposition 4 to add a supplementary affine term to the previous linear



(a) Two constraints, linear (b) Two constraints, affine

Figure 2: Prediction with two constraints: weights summing to one and the average of predicted values set to $m = 1.5$ (horizontal dashed line). The average of predictions is equal to this value $m = 1.5$ in both cases. The observations are the black dots. The thin, dotted blue line is the underlying function. In the right panel, one applies the assumption in Remark 3 with $\rho = 0$ and $\sigma_Z = \sigma/10$.

predictor of Figure 2a while preserving the sum of weights being equal to one. The affine term is derived from a random variable $\mathbf{Z}$, and we choose, as previously, $\sigma_Z = \sigma/10$, so that this external information is assumed to be trustworthy. As above, the average of predicted values is exactly the prescribed one, by construction. Again, given $\mathbf{Z} = \mathbf{m}$, the consequence is that, far from observed locations, the prediction tends to put all weights on this external source of information, so that the prediction gets closer to $\mathbf{m}$, as one can see at the extreme right of Figure 2b. Another consequence of the affine term is the reduction of the confidence band width, as a new source of information has been added. With the prescribed average of the predicted value, the predictor is not interpolating, but adding the affine term helps the prediction get closer to observations.

Notice that the constraint $\mathbf{m}$ is purposely set to an arbitrary value, so that adding this constraint does not necessarily improve the prediction in this toy example: it is not the aim of such a constraint. The reader may imagine the case of an adverse scenario, which can worsen the prediction, or the case of external useful information, which can improve it.

In this simple toy example, one can check numerically that each prediction satisfies the constraints that it should. One can also clearly visualise the impact of the specific constraint on average predicted values and the behaviour of the predictor when adding an affine term.

The illustrations that have been presented in this subsection are available in the notebook `Application1D` of the online supplementary material.

5.2 A Multi-Output Time Series Example

In the previous example, we illustrated the impact of constraints on the prediction of a one-dimensional output. Hence, the *joint* aspect of the estimation was not discussed. In the present example, one considers multi-output data so as to illustrate the specificity of the single hyperparameter estimation with multiple outputs. We choose one-dimensional inputs in $\mathbb{R}$ to facilitate the interpolation representation, but considering more general inputs in $\mathbb{R}^d$, $d > 1$, would be easy. It would only change the number of hyperparameters

to estimate, d instead of 1.

Imagine the following situation: a city wants to infer the history of the concentration of some pollutants at a particular crossroad based on a small series of measurements. This simple problem requires a model that takes time as input and multiple concentrations as output. Obviously, the end purpose would be to have a model with space and time as input, but this is outside this illustration's framework.

Using the data *air quality* (Vito, 2016), one tries to infer the concentration of several pollutants from only a few values. The studied pollutants in the data were chosen arbitrarily: CO, C6H6, NOx and NO2. The time range of learning data has been selected so that visually there is not too much missing data in the period (sensor stuck to an inferior bound or missing). It corresponds to hourly measurements from 23/04/2004 18.00.00 to 28/04/2004 17.00.00. Missing values are tagged with the value -200 in this data. They were all filtered before the study, as if they were not informative at all. The challenge is to predict all hourly measurements in the selected period from only $n = 10$ values.

The purpose here is not to give specific conclusions about the measured pollution but only to illustrate the capacity of the Joint Kriging model to handle complex multivalued data with very few hyperparameters to optimise. The idea is to create a *joint* model that is as simple as possible. Many refinements of the model could be suggested, but this is not the purpose of this example.

Let us model the covariances between components of $\mathbf{Y}(.)$ using Example 3. The proposed method does not require the definition of each cross-correlation between a pollutant concentration at one location and a different pollutant concentration at a different location. It just takes one covariance function $k(x, x')$ between an implicitly weighted sum of all output variables. We use the multiplication of two covariance kernels (hence it is positive semi-definite): a periodic kernel with a period of one day and a kernel of the Matérn 3/2 family (see C. E. Rasmussen & Williams, 2006, Chapter 4 and Equation (4.31)):

$$k(x, x') = \sigma^2 \exp\left(-\sin^2(\pi|x - x'|)\right)\left(1 + \frac{|x - x'|}{\theta}\right)\exp\left(-\frac{|x - x'|}{\theta}\right). \tag{22}$$

The parametrisation has been simplified, e.g. factors $\sqrt{3}$ in Matérn covariance expressions are not used here because they have the same effect as a rescaling of the characteristic length-scale θ. Notice that despite the p dimensional output where $p = 4$ is the number of studied pollutants, the kernel $k(x, x')$ in Equation (22) depends only on two hyperparameters θ and σ^2. Since σ^2 impacts the uncertainty in the prediction but not the prediction itself, it is set to $\sigma^2 = 1$.

Let us first consider one single constraint: the sum of weights should be one. It corresponds to the Joint Ordinary Kriging predictor.

Figure 3 shows the optimisation of the *single* length-scale hyperparameter θ. As this study does not aim at comparing the prediction accuracy with other methods, we did not use a separate test sample but only a validation sample, keeping in mind that it may lead to overfitting. The validation data used for this single hyperparameter estimation is set to all hourly measurements in the selected period. For the hyperparameter optimisation, a specific error has been chosen, where one optimises the worst standardised mean absolute error over all $p = 4$ series: The errors have been standardised in order to make them unitless and scale-invariant. The best estimation is $\hat{\theta} \simeq 1.4$. It is kept for all other illustrations in the subsection.

The optimisation here depends quite heavily on the chosen observation locations, so that in practice, an averaged error on several training and validation datasets would probably be more stable. In many situations on real data, the error function is monotonic, either

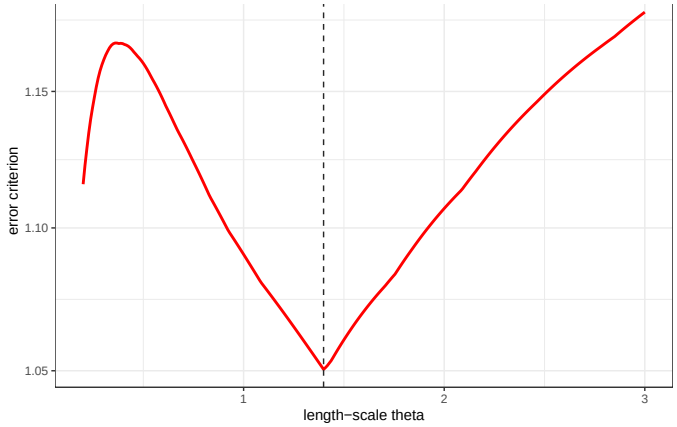


Figure 3: optimisation of the single correlation hyperparameter θ for the four selected pollutants, data extracted from Air quality data set.

increasing and leading to extremely small optimised hyperparameter θ (the prediction then tends to return quickly to an average value), or either decreasing, leading to a very large value of θ (the prediction then tends to smooth data a lot). Classical co-Kriging strategies that define a large number of cross-covariance hyperparameters would probably worsen the situation, highlighting the utility of a small number of hyperparameters.

Figure 4 presents the simultaneous predictions of the four pollutant concentrations with Joint Kriging, the only constraint being that weights sum to 1. The confidence band associated with a given pollutant is proportional to the standard deviation of this pollutant's concentration, as detailed in Proposition 6. Pollutant concentrations have very different orders of magnitude, but when applying Proposition 6, the obtained confidence bands look quite comparable between series, as desired.

With very few hyperparameters and a rough covariance model, the result has a lot of room for improvement. Nevertheless, despite the single model hyperparameter θ and, considering the limited number of observations $n = 10$, the predictions of the $p = 4$ concentrations seem quite reasonable. By construction, each prediction is a combination of observed values of the considered pollutant, with weights summing to one. In Figure 4, no other constraint is added, so that the average of predictions does not correspond at all to a specific prescribed value.

Figure 5 presents the simultaneous predictions of the four pollutant concentrations with Joint Kriging, on which both constraints on the weights and on the predicted values are imposed using the affine model of Remark 4. The left panels show an adverse scenario where the average of predictions is set to 130 % of the true average. The right panels show a normal scenario where the average of predictions is set to 100 % of the true average. Using this setting, the interpolation property is lost, as seen in the previous example of Section 5.1, but the $n = 10$ observations still have a large influence, and the global shape of the prediction is preserved. By construction, the average of predicted values (thick solid line) is exactly the prescribed one (horizontal thick dashed dark green line).

Considering this single hyperparameter model with a basic covariance model, the results also seem reasonable when using two constraints. In the left panels, the average of predicted values is exactly set to 30 % more than the observed average of pollutant concentration, which is a lot. However, the visual differences between true and predicted sequences look surprisingly moderate, even in this adverse scenario. Despite satisfying all constraints, the model still offers a good fit with observations.

The goal of this numerical experiment is to demonstrate the Joint Kriging model's

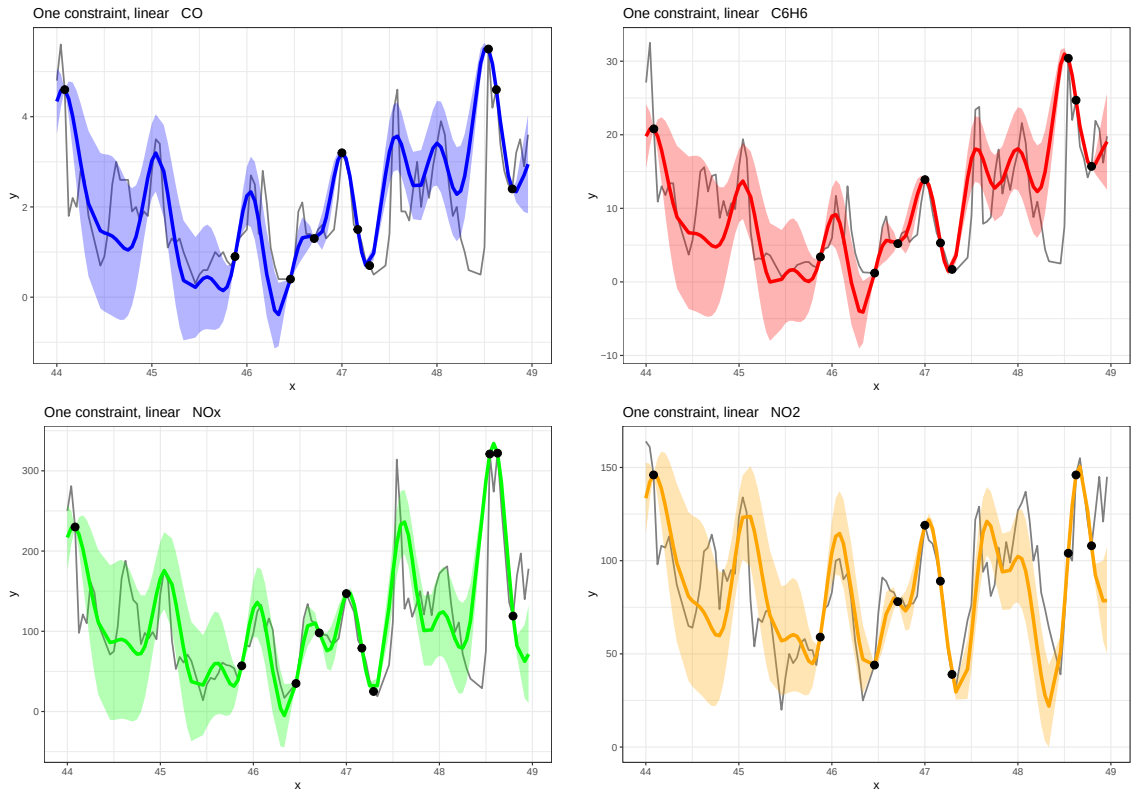


Figure 4: Joint Kriging interpolation: using the joint ordinary model with weights summing to one, with very few data points (black dots) and a single optimised length-scale hyperparameter obtained in Figure 3. Upper left: CO, upper right: C6H6, lower left: NOx, lower right: NO2. Predictions are in thick solid lines, and true values are in thin black solid lines.

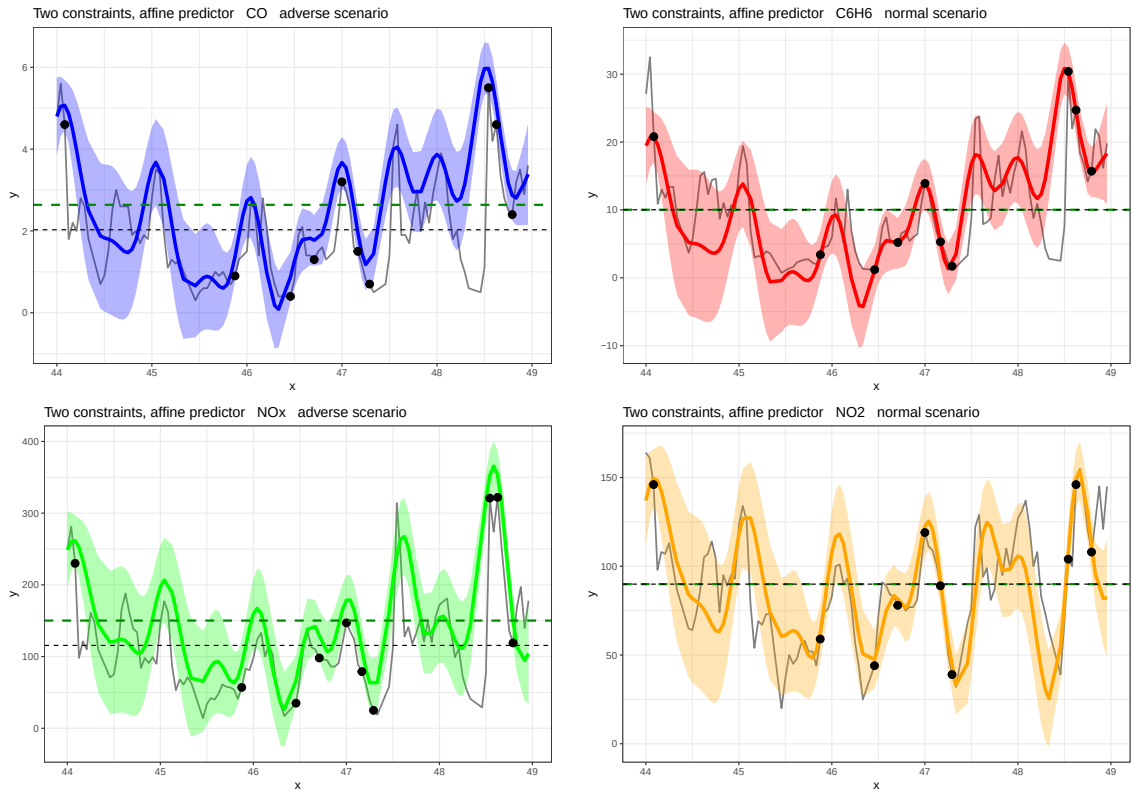


Figure 5: Adverse scenarios: interpolation using the joint affine model with two constraints (weights summing to one, prescribed average predictions), with very few data points (black dots), and a single optimised length-scale hyperparameter obtained in Figure 3. Upper left: CO, upper right: C6H6, lower left: NOx, lower right: NO2. Predictions are in thick, solid lines, and true values are in thin, black, solid lines. Left panels are adverse scenarios where the average of predictions (thick dark green horizontal dashed line) is set to 130 % of the true average (thin black horizontal dashed line). Right panels are scenarios where the average of predictions is set to 100 % of the true average.

ability to handle complex multivalued data. It also illustrates the advantage of having a limited number of hyperparameters. One sees here that with a quite simple model, in a difficult problem (predicting four quite erratic time series from 10 observations), the model performs reasonably well. Furthermore, it allows for introducing some constraints, like setting an adverse scenario of a 30 % increase in the pollutant concentration.

The illustrations that have been presented in this subsection are available in the notebook `ApplicationAirQuality` of the online supplementary material.

5.3 A Constrained Classification Example

We present in this subsection the specific case of multi-dimensional outputs derived from a classification problem. As presented in Section 3, Joint Kriging can be implemented for fuzzy classification. Different modalities of a classification variable are regarded as multiple output variables with values in $[0, 1]$.

Imagine the case of an event with measurable intensity that may occur at a given location in a territory. We are interested in classifying the intensity of this event, if it occurs, into multiple classes, depending on some thresholds. In the following, this event is an earthquake, and its intensity is its Richter magnitude.

The Quake data set given in Simonoff, 1996, visualised in Figure 6, describes 2 178 earthquakes with their latitude, longitude, focal depth, and magnitude. A given location x has three coordinates: latitude, longitude, and focal depth. For a single observation at a location x, the target $\mathbf{Y}(x) = (Y_1(x), Y_2(x))^\top$ is equal to $(1, 0)^\top$ if an earthquake is occurring here with a magnitude above the data set average magnitude, or $(0, 1)^\top$ otherwise. If a location x is observed repeatedly, the membership degrees at x are averaged out over observations. It makes sense to impose that membership degrees are summing to one, so that ${\mathbf{1}_p}^\top \mathbf{Y}(x) = 1$. Extensions with more thresholds are easy to conduct, as in Figure 11. We keep here $p = 2$ for comparison to existing benchmarks. The binarised data is available at www.openml.org/search?type=data&id=772, on the openML website Bischl et al., 2021.

The purpose here is to compare the performance of Joint Kriging with a set of 69 other models' performances. The study available at www.openml.org/search?type=task&id=4516 (data retrieved on the 28th of June 2024) compares models, called flows in openML, performing 10 times a 10-fold cross-validation and computing the predictive accuracy as a performance indicator (see tab `Analysis`, measure `predictive_accuracy`).

Remember from Example 3 that although we are constructing a bivariate model, we need a single covariance kernel. The latter should be periodic with respect to latitude and longitude, not with respect to focal depth. A simple way to define an admissible kernel is to multiply the three kernels associated with the three dimensions (see C. E. Rasmussen & Williams, 2006):

$$k(x, x') = \sigma^2 \exp\left(-2\frac{\sin^2((x_1 - x'_1)/2)}{\theta_1^2} - 2\frac{\sin^2((x_2 - x'_2)/2)}{\theta_2^2} - 2\frac{(x_3 - x'_3)^2}{\theta_3^2}\right)$$

The hyperparameters estimation has been treated separately on other train/test splits to avoid overfitting the data. The resulting values for $\boldsymbol{\theta}$ are 2.3 for latitude, 0.9 for longitude, and 196.8 for focal depth.

In order to visualise the algorithm's behaviour, we predict on a grid of latitude, longitude, and focal depth values. In addition to imposing the sum of membership degrees to be 1, we set the output mean expectation to be the same as in the data set. Predictions on a grid of latitude, longitude, and focal depth are presented in Figure 7. One can observe

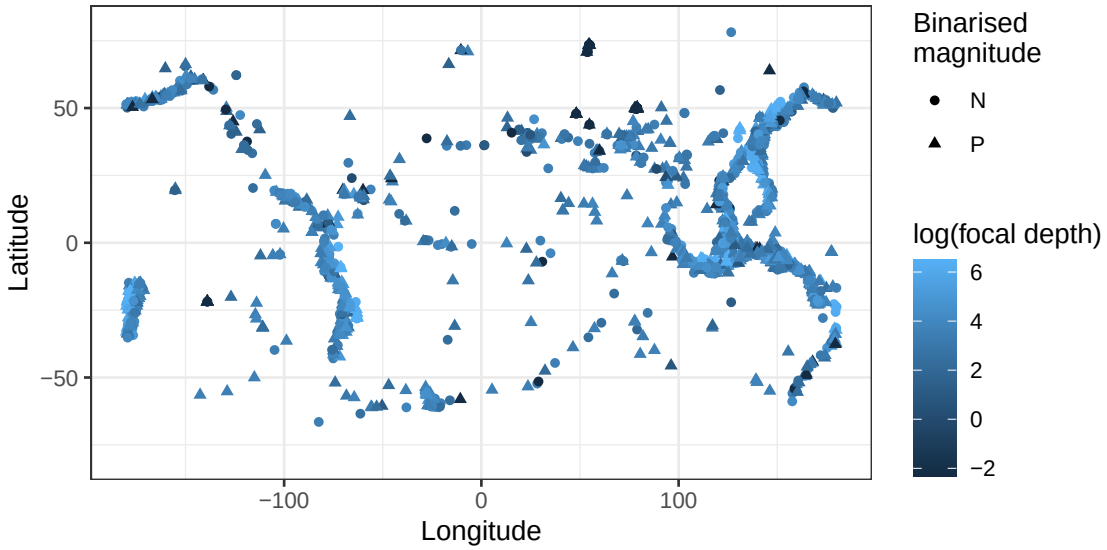


Figure 6: Earthquakes observations. An earthquake is a point with latitude, longitude, and focal depth (given by the colour) as its coordinates. Triangles represent earthquakes whose magnitude is above average. Circles represent earthquakes whose magnitude is below average.

that maps representing membership degrees (first two rows) can be deduced from each other by $y = 1 - x$. The third row shows a segmentation of the plane into areas where the membership degree for "P: magnitude is greater than average" is greater than 0.5, and areas where the converse is true. This segmentation depends on the focal depth: a small focal depth on the top row (21 km, first quartile) and a greater one on the bottom row (68 km, third quartile). For instance, looking at the bottom left corner of the map, which is around the Fiji archipelago, one can predict that earthquakes with small focal depths are more likely to be of large magnitude than deep earthquakes. However, the converse is true in the South Atlantic area (bottom-centre part of the map). Moreover, the predictor achieves circular coherence along longitude due to the periodicity of covariance. Periodicity along latitude is more difficult to observe because it covers only 180°.

Performances are evaluated using Predictive Accuracy which is the percentage of instances that are classified correctly. It is measured on binarised predicted membership degrees, on a ten times 10-fold cross-validation, as in the OpenML benchmark, in order to get comparable results. Prior to that, the hyperparameters optimisation has been treated separately on other train/test splits in order not to overfit the data. Figure 9 presents, from top to bottom, two results found in openML, i.e., the best recorded model, which is the Kernel Logistic Regression with Radial Basis Function Kernel and Random Forest for reference. Below are the results of the Joint Kriging models: the simple model without constraint, the model with weights summing to 1, the model with constraint on the prediction and weights summing to 1, the affine model with weights summing to 1, and the affine model with constrained output.

For the ten runs, each diagram shows the Predictive Accuracy of each run (coloured points), the minimum, first quartile, median, 3rd quartile, and maximum, as well as the mean value materialised by a cross. Although the runs' performances stay in the range of those observed for Random Forest and Kernel Logistic Regression, the average values obtained with Joint Kriging are greater: the average is 0.558 ± 0.002 for the best model

in the OpenML benchmark and 0.5660 ± 0.0038 for the best Joint Kriging model. The latter was even slightly greater, 0.5669, during hyperparameter optimisation, due to a slight overfit that has been reduced when using different train/test splits. Benchmark being based on this average value, it means that Joint Kriging has a better performance than the 69 models tested in the OpenML benchmark.

One can expect the multiplication of constraints to have an adverse effect on performance, as a constrained optimisation has less degree of freedom than an unconstrained one. On the other hand, injecting useful information through constraints may improve the performance. Figure 9 shows that overall, the performance is improved, especially when adding the constraint on the output. Figure 10 shows the distribution of the mean predictive accuracies for the 69 models tested in the OpenML website. None of them is above 0.56, while all Joint Kriging models are.

In Figure 8, one uses the affine version of Joint Kriging with two constraints: weights summing to one and prescribed average prediction. In the left panels, an adverse scenario forces the average predicted membership degrees of the first class (large magnitude events) to be equal to 65 %. In the right panels, this percentage is set to the observed percentage of large-magnitude events, 55 %. This illustrates the usefulness of the constraint for adverse modelling.

In order to compare the results with existing benchmarks, we studied above the $p = 2$ binary classification problem. But the method can handle more classes as well. As an example, in Figure 11, we give a prediction for $p = 4$ classes. Observations have been converted into four classes using three Richter magnitude thresholds: 5.85, 5.95 and 6.15. Specific thresholds have been chosen for this illustration in order to get enough observations in each class (at least 17 % observations), but a seismology study might focus on other thresholds. Once again, the predictor achieves a circular coherence along longitude, and one can observe complex patterns that would be difficult to catch with classification trees. The presented classification task was constructed from indicators deriving from an underlying real value, the Richter magnitude, and from thresholds, thus creating ordinal classes. But the prediction can also be derived from observations of non ordinal class labels without any underlying process or thresholds.

The use of the constraint on predicted values, as can be seen in Figure 9, is likely to increase the prediction accuracy, if the knowledge of classes distribution is useful and trustable. However, the recommendation to use or not such a constraint mostly depends on the available information (do we know the desired classes distribution) and on the objective of the study (do we need adverse scenarii, do predictions need to be coherent with observed class percentages, or do we need to study the rare classes). One can imagine cases where the prediction accuracy is decreased with the constraint, but where we need the constraint to study rare classes. As an example, if one class is 99% of the observations, a unconstrained model is likely to predict only this class; it would has an excellent accuracy but would not allow to study other classes.

A comparison with co-Kriging would be desirable, but co-Kriging would not lead to predictions summing to one, hence it is not directly applicable to classification. In terms of complexity for p output variables, the $O(p^2)$ parameters to optimize in co-Kriging would lead, on a grid search of hyperparameters, to p times more iterations than the $O(p)$ parameters of Joint Kriging.

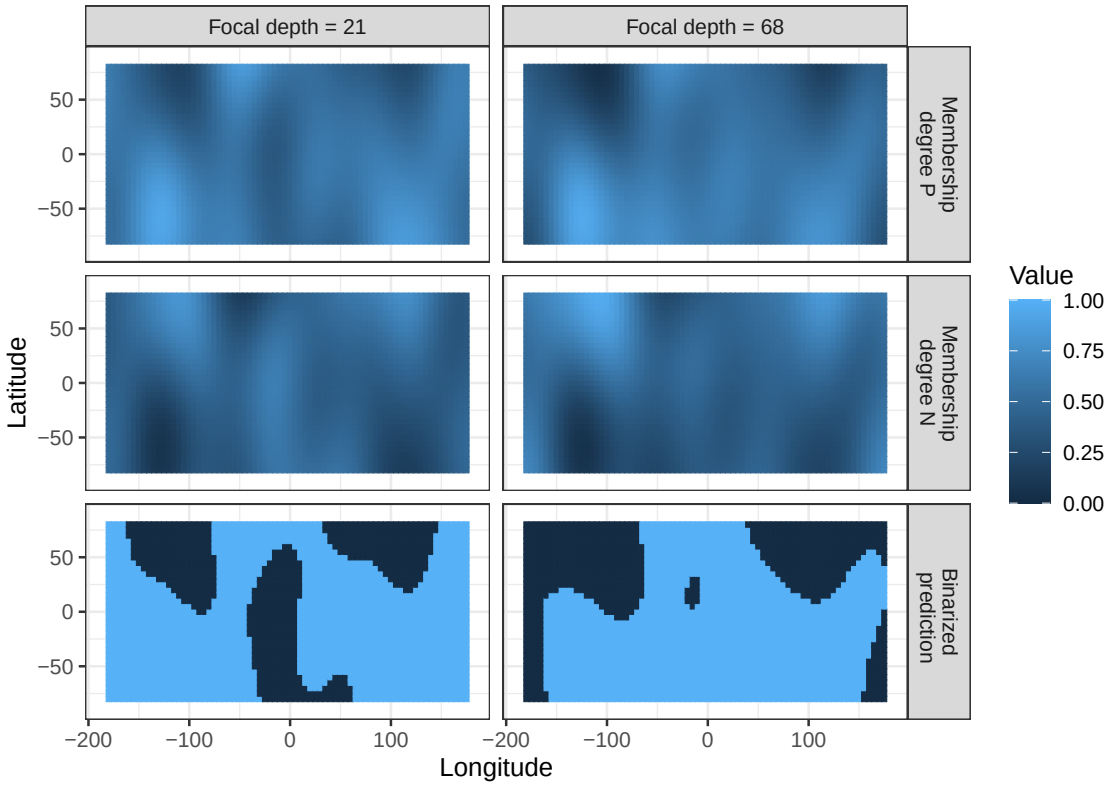


Figure 7: Joint Kriging with 2 constraints, earthquakes' magnitude prediction into 2 classes. From top to bottom: membership degree of "P: magnitude is above average", membership degree of "N: magnitude is below average", binarised prediction (1 if membership degree of P is greater than 0.5). Left: focal depth of 21 km. Right: focal depth of 68 km.

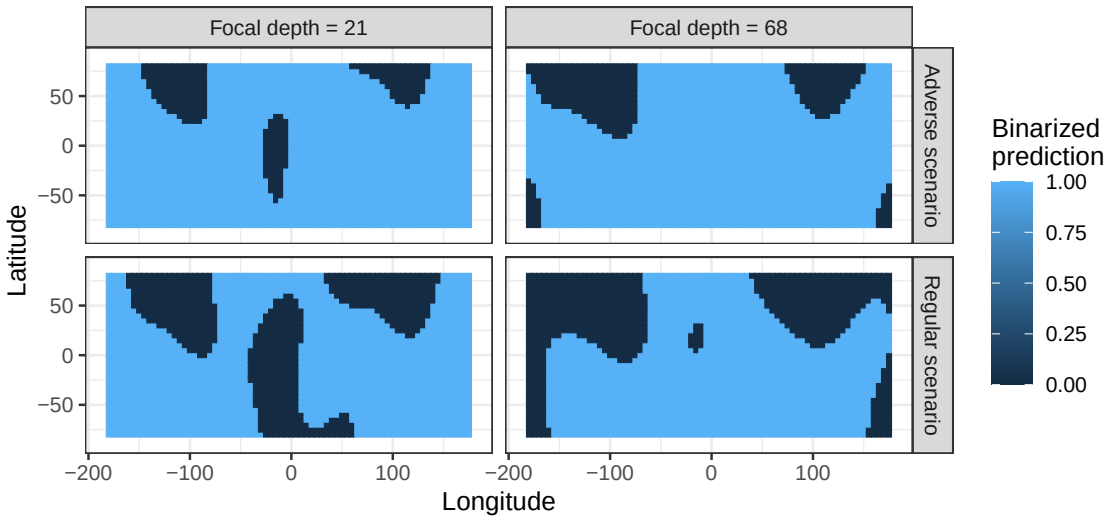


Figure 8: Adverse scenario: predicted membership degrees of earthquakes' magnitude using Joint Kriging with two constraints. Top panels: adverse scenario, first-class output average constrained to be 65 %. Bottom panels: regular scenario, output average constrained to 55.5 %. Left: focal depth of 21 km. Right: focal depth of 68 km.

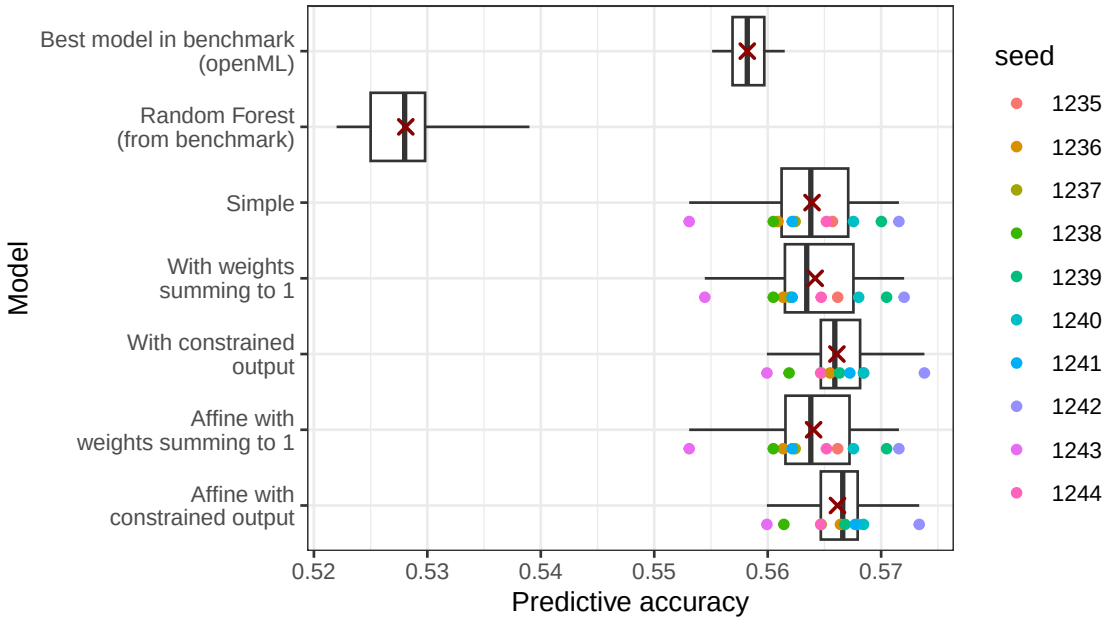


Figure 9: Distribution of performances for 10 runs of two OpenML models (the best OpenML model among 69 models, and Random Forest), and the 5 different types of Joint Kriging model. The whisker plots give the minimum, first quartile, median, third quartile, and maximum. The dark red cross indicates the average predictive accuracy; the higher, the better. The average is 0.5660 ± 0.0038 for the best Joint Kriging model and 0.558 ± 0.002 for the best model in the OpenML benchmark. Other 67 models of the Benchmark are omitted here. Data were extracted on June 28th, 2024.

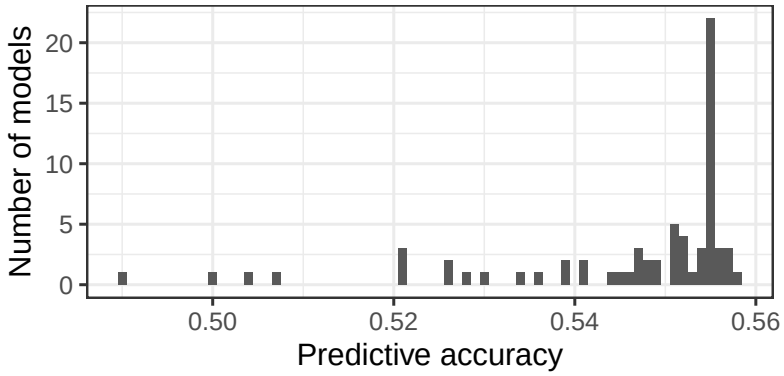


Figure 10: Distribution of the mean predictive accuracies for the 69 models tested on the Quake dataset, in the OpenML framework. Note that some models have been run multiple times, in which case we select only the best run. The graphic is a bar plot of the predictive accuracies rounded to the nearest third digit. Data were extracted on June 28th, 2024.

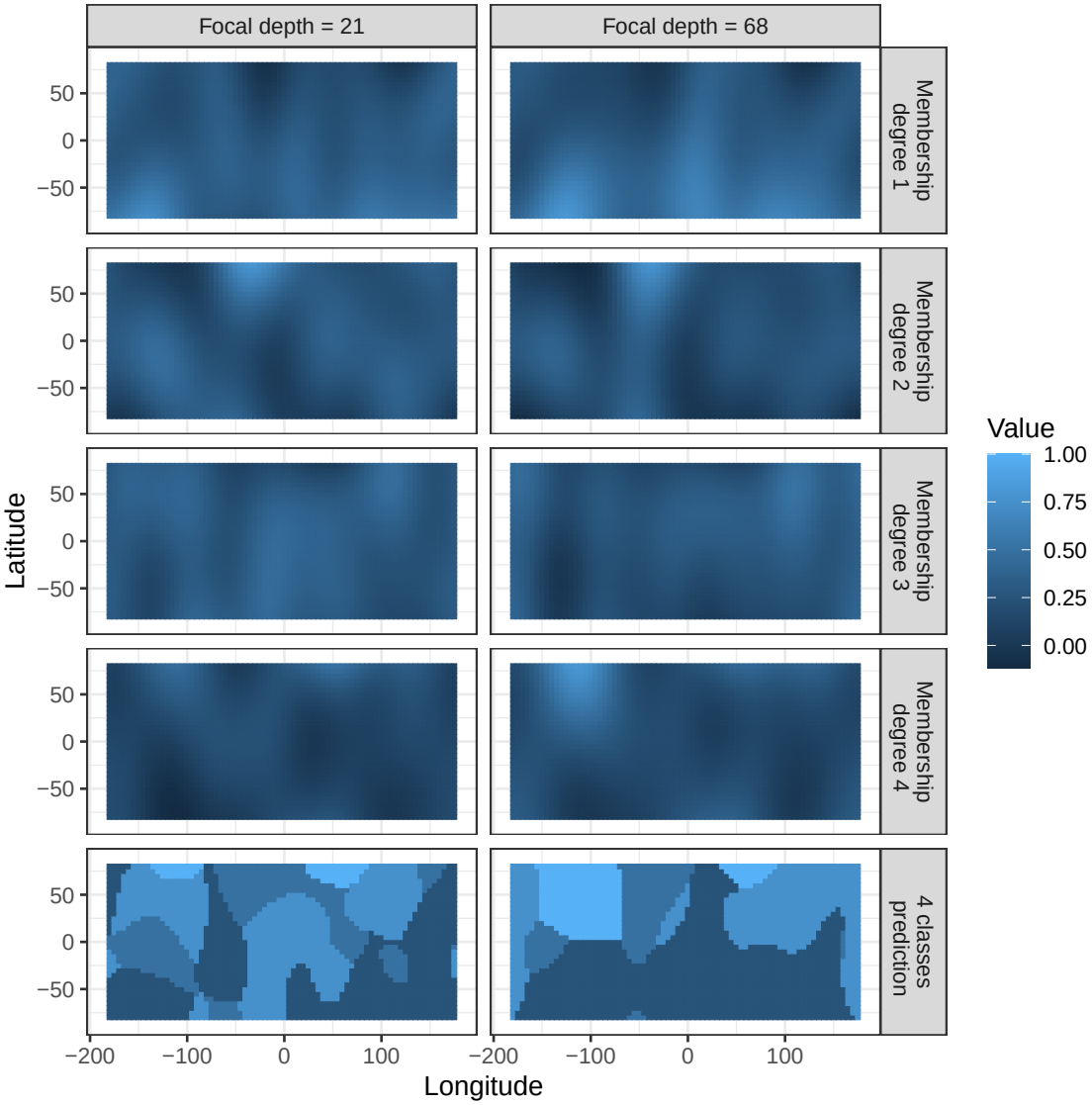


Figure 11: Affine Joint Kriging with 2 constraints. Earthquakes' magnitude is divided into 4 classes. From top to bottom: membership degrees of "1: magnitude is smaller than 5.85", "2: magnitude is between 5.85 and 5.95", "3: magnitude is between 5.95 and 6.15", "4: magnitude is greater than 6.15" and class of greatest membership degree in the 5^{th} row coloured by increasing magnitude from dark to light blue. From left to right: focal depth of 21 km, focal depth of 68 km.

In this classification example, we used a direct implementation of the model with a single covariance family. Nevertheless, the average performance is the best one among the whole OpenML benchmark. Fine tuning of the model would surely lead to performance improvements. The illustration above aims at demonstrating that, with very basic assumptions, the method is competitive with an open benchmark that has numerous competitors, as shown in Figure 9. It also aims at showing that it can model adverse scenarios, as in Figure 8, or multiple classes, as in Figure 11.

The illustrations that have been presented in this subsection are available in the notebook `ApplicationQKmain` of the online supplementary material.

5.4 Performance on multiple datasets

In this subsection, we perform a standard test to compare Joint Kriging with three other models that are Penalized Multinomial Regression, Weighted k-Nearest Neighbours and Random Forest, using several datasets and several performance metrics.

As highlighted in Demšar, 2006, in order to do a comparative study of several classification methods, it is essential to build a statistical analysis of the performance over several datasets. The subsection aims at giving a better idea of the practical performance of the Joint Kriging model, using several datasets. Furthermore, it also aims at comparing the performances using different metrics. A detailed review of different practical metrics for multi-class classification, together with their advantages and disadvantages is available in Grandini et al., 2020.

The present study did not use a very large number of datasets, of methods, of tried hyperparameters. More significant results would require a far more extensive benchmark, that is let to further studies. We reproduce the methodology of the previous subsection by performing a cross-validation prediction 10 times on each dataset of a sample of 9 datasets. Prior to the experiment, we pick the datasets from `caret` and `mlbench` packages available in the Comprehensive R Archive Network (CRAN). The chosen datasets are described in the Table 2. The considered models are Joint Kriging (with Affine term and prescribed average values), Penalized Multinomial Regression, Weighted k-Nearest Neighbours Classification, Parallel Random Forest. The detailed methodology of this experiment is given in Algorithm 1.

The results are gathered in Tables 3 and 4. The Table 3 gives the performance of the considered methods using several indicators, see Grandini et al., 2020: Balanced Accuracy, Macro Average Precision, Micro Average Precision (also equal to Micro Average Recall), Macro Average Recall. A summary based on ranks is available in Table 4. One can see that, on the few tested datasets, the Joint Kriging method performs reasonably well, and exhibits the best average ranks for several indicators. However, due to the limited size of the benchmark, the statistical tests presented in Demšar, 2006 were not able to prove the superiority of one classifier over the others. A larger benchmark with much more datasets, finer hyperparameters optimization, and more concurrent methods would be necessary to perfectly assess the performance of all classifiers. Understanding situations where one classifier is more suited than the others is also challenging. These developments are let for further studies.

This short subsection shows that the performance of the Joint Kriging method is clearly

Algorithm 1 Detailed methodology of the multiple datasets benchmark

for each model m and dataset dt **do**
  for each repetition in $1, ..., 10$ **do**
    Draw a random sample of maximum 555 observations in dt and split it into 10 subsets $S_1, ..., S_{10}$.
    for each $i \in 1, ..., 10$ **do**
      Take $\cup_{j \neq i} S_j$ and split it into a training set T_i (80% of the observations) and a validation set V_i (20% of the observations).
      Set $bestAccuracy := -1$; $currentAccuracy := 0$.
      Set $testedVars := all\ variables$; $selectedVars := \emptyset$.
      while $currentAccuracy > bestAccuracy$ **do**
        for each $Var \in testedVars$ **do**
          Run the model m by testing 5 different hyperparameters settings with training on T_i and prediction on V_i and variables $selectedVars \cup Var$.
          Compute the accuracy.
        end for
        Keep the best variable $bestVar$ and the best accuracy $currentAccuracy$.
        if $currentAccuracy > bestAccuracy$ **then**
          Update $testVars$ withdrawing $bestVar$ and $selectedVars$ adding $bestVar$.
          $bestAccuracy := currentAccuracy$
        end if
      end while
      End up with a set of selected variables and best parameters.
      Train the model on $\cup_{j \neq i} S_j$ and predict on S_i.
      Compute all performance indicators.
    end for
    Compute the mean value of each performance indicator over the 10 folds.
  end for
  Compute the mean value of each performance indicator over the 10 repetitions.
end for

	Package	Dataset	Classes	Observations	Features	Folds
oil	caret	oil	7	96	7	2
scat	caret	scat	3	110	19	10
segmentationData	caret	segmentationData	2	2019	61	10
BreastCancer	mlbench	BreastCancer	2	699	11	10
Glass	mlbench	Glass	7	214	10	10
LetterRecognition	mlbench	LetterRecognition	26	20000	17	10
Satellite	mlbench	Satellite	6	6435	4	10
Vehicle	mlbench	Vehicle	4	846	19	10
Vowel	mlbench	Vowel	11	990	10	10

Table 2: List of datasets used for comparing Joint Kriging with other models. Folds indicate the number of folds used for cross-validation.

promising, even if more efforts and a much more extensive benchmark would be required to demonstrate a possible superiority in specific contexts. Let us also recall that, even when ignoring the performance of the methods, the need to prescribe the average percentages of predicted classes may also justify the use of the Joint Kriging method.

The illustrations that have been presented in this subsection are available in the notebook `ApplicationMultipleDatasets` of the online supplementary material.

6 Conclusion

A Joint Kriging model on multiple outputs has been presented, where at each prediction location, the same weights apply to all outputs. This simplification was necessary to handle all the considered constraints. It also allows for easy covariance modelling with very few hyperparameters even though the number of outputs p is large. Still, the model benefits from Kriging advantages: interpretability, ability to interpolate data, prediction of the uncertainty in each prediction, and specific covariance modelling. As with any simplification, the model can surely be improved and may have some limitations compared to heavily parametrised models. For instance, co-Kriging with many cross-covariance functions might be more flexible for dealing with time series with different regularities, or models with parametrised distortions of locations might be more convenient for dealing with non-stationarities. However, the limited number of hyperparameters and the simplicity of their estimation are an asset of the model, while allowing specific model characteristics such as periodicity. Furthermore, the model is not limited to Gaussian Processes, as it only relies on the existence of moments of order one and two.

An original constraint on predicted values was also introduced. It appears to be useful for using external information, for adverse modelling, for homogenising results, or for considering fairness constraints. To handle this constraint, all weights of predicted points need to be computed at the same time, unlike usual Kriging techniques. But the resulting predictor itself is quite simple to derive since it is given by a closed formula. Some extensions using an affine term were also proposed to account for external information and provide more control over the behaviour of the predictor far from observations.

Ultimately, an application to classification was developed. Applying a multi-output Kriging model on classification is feasible through the prediction of membership degrees. Even without constraints, it is in itself interesting: it allows for interpretability, modelling uncertainty's estimation, and interpolating data. Using Joint Kriging with the proposed constraints easily ensures that membership degrees sum to one and allows for prescribed

Dataset	Model	Accuracy	BA	MacroAP	MicroAP	MacroAR
BreastCancer	multinom	0.948	0.940	0.944	0.948	0.940
BreastCancer	kknn	0.943	0.938	0.937	0.943	0.938
BreastCancer	parRF	0.943	0.937	0.938	0.943	0.937
BreastCancer	JointKriging	0.941	0.936	0.934	0.941	0.936
Glass	parRF	0.725	0.792	0.685	0.725	0.647
Glass	JointKriging	0.711	0.770	0.680	0.711	0.608
Glass	kknn	0.685	0.760	0.608	0.685	0.592
Glass	multinom	0.557	0.656	0.477	0.556	0.421
LetterRecognition	JointKriging	0.725	0.856	0.732	0.725	0.724
LetterRecognition	kknn	0.705	0.846	0.713	0.705	0.705
LetterRecognition	parRF	0.692	0.839	0.701	0.692	0.691
LetterRecognition	multinom	0.571	0.776	0.569	0.571	0.570
Satellite	JointKriging	0.845	0.887	0.822	0.845	0.806
Satellite	parRF	0.837	0.882	0.816	0.837	0.798
Satellite	kknn	0.828	0.877	0.808	0.828	0.789
Satellite	multinom	0.808	0.852	0.765	0.808	0.744
Vehicle	JointKriging	0.700	0.801	0.689	0.700	0.703
Vehicle	parRF	0.692	0.796	0.681	0.692	0.695
Vehicle	kknn	0.655	0.771	0.645	0.655	0.658
Vehicle	multinom	0.648	0.767	0.634	0.648	0.651
Vowel	kknn	0.909	0.950	0.911	0.909	0.909
Vowel	parRF	0.866	0.926	0.868	0.866	0.866
Vowel	JointKriging	0.786	0.882	0.788	0.786	0.786
Vowel	multinom	0.622	0.792	0.622	0.622	0.622
oil	parRF	0.848	0.860	0.780	0.842	0.749
oil	multinom	0.847	0.838	0.809	0.875	0.705
oil	kknn	0.816	0.828	0.760	0.826	0.691
oil	JointKriging	0.782	0.787	0.621	0.759	0.614
scat	multinom	0.671	0.675	0.647	0.671	0.560
scat	JointKriging	0.579	0.620	0.521	0.579	0.494
scat	kknn	0.568	0.610	0.509	0.568	0.479
scat	parRF	0.559	0.612	0.496	0.559	0.482
segmentationData	multinom	0.755	0.712	0.736	0.755	0.712
segmentationData	JointKriging	0.751	0.721	0.729	0.751	0.721
segmentationData	parRF	0.748	0.722	0.725	0.748	0.722
segmentationData	kknn	0.736	0.709	0.712	0.736	0.709

Table 3: Average performances of each model with each dataset. The average is computed over 10 runs.
multinom: Penalized Multinomial Regression; kknn: Weighted k-Nearest Neighbours Classification; parRF: Parallel Random Forest.
BA: Balanced Accuracy; MacroAP: Macro Average Precision; MicroAP: Micro Average Precision (also equal to Micro Average Recall); MacroAR: Macro Average Recall.

Indicator	Joint Kriging	kknn	parRF	multinom
Accuracy	**2.22**	2.72	2.28	2.78
Balanced Accuracy	2.22	2.78	**2.00**	3.00
Macro Average Precision	**2.22**	2.78	2.33	2.67
Micro Average Precision	**2.22**	2.72	2.39	2.67
Macro Average Recall	2.22	2.78	**2.00**	3.00

Table 4: Average rank of each model over the 9 datasets. The lower the better (in bold font).

percentages of each predicted class. The simplified covariance model greatly eases the hyperparameters' estimation. At the same time, with Joint Kriging, classification tasks benefit from the diversity of covariance kernels, including periodicity. The resulting classification performs especially well in the investigated practical case: in the earthquake numerical example, the model competes with the best-provided approaches on an open data set with numerous competitors.

Multiple extensions to the model can be imagined. For instance, the model with constrained predicted values does not guarantee continuous interpolation, so that further work may fix this problem. A specific estimation procedure for the underlying joint covariance structure could also be of interest. Moreover, once applied to classification, membership degrees summing to one do not imply the combinations to be convex. Some weights can still be negative or greater than 1 so that an adjustment of the nugget effect may be required. Ensuring the combinations to be convex without any nugget effect adjustment could also be an improvement. Eventually, one may be interested in searching for a way to relax the simplifying assumption while keeping the constraints.

Bibliography

Agarwal, G., Sun, Y., & Wang, H. J. (2021). Copula-based multiple indicator kriging for non-gaussian random fields. *Spatial Statistics*, *44*, 100524. https://doi.org/10/gtn36g

Alvarez, M. A., Rosasco, L., Lawrence, N. D., et al. (2012). Kernels for vector-valued functions: A review. *Foundations and Trends® in Machine Learning*, *4*(3), 195–266. https://doi.org/10/gmmw42

Bachoc, F. (2013). Cross validation and maximum likelihood estimations of hyper-parameters of gaussian processes with model misspecification. *Computational Statistics & Data Analysis*, *66*, 55–69. https://doi.org/10/gmt4z5

Banerjee, A., Dunson, D. B., & Tokdar, S. T. (2013). Efficient gaussian process regression for large datasets. *Biometrika*, *100*(1), 75–89. https://doi.org/10/f4q3jt

Benatti, K. A., Pedroso, L. G., & Ribeiro, A. A. (2022). Theoretical analysis of classic and capacity constrained fuzzy clustering. *Information Sciences*, *616*, 127–140. https://doi.org/10/gtn36f

Bischl, B., Casalicchio, G., Feurer, M., Gijsbers, P., Hutter, F., Lang, M., Mantovani, R. G., Rijn, J. N. v., & Vanschoren, J. (2021). OpenML benchmarking suites. *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*. https://openreview.net/forum?id=OCrD8ycKjG

Bradley, P. S., Bennett, K. P., & Demiriz, A. (2000). Constrained k-means clustering. *Microsoft Research, Redmond*, *20*.

Chiang, J.-L., Liou, J.-J., Wei, C., & Cheng, K.-S. (2013). A feature-space indicator kriging approach for remote sensing image classification. *IEEE transactions on geoscience and remote sensing*, *52*(7), 4046–4055. https://doi.org/10/gdp8zr

Chiles, J.-P., & Delfiner, P. (2012). *Geostatistics: Modeling spatial uncertainty* (Vol. 713). John Wiley & Sons.

Chilès, J.-P., & Desassis, N. (2018). Fifty years of kriging. *Handbook of mathematical geosciences: Fifty years of IAMG*, 589–612.

Chung, H. S., & Alonso, J. (2002). Design of a low-boom supersonic business jet using cokriging approximation models. *9th AIAA/ISSMO symposium on multidisciplinary analysis and optimization*, 5598.

Clarotto, L., Allard, D., & Menafoglio, A. (2022). A new class of alpha-transformations for the spatial analysis of compositional data. *Spatial Statistics*, *47*, 100570. https://doi.org/10/gtn36m

Cousin, A., Maatouk, H., & Rullière, D. (2016). Kriging of financial term-structures. *European Journal of Operational Research*, *255*(2), 631–648.

Cressie, N. (1988). Spatial prediction and ordinary kriging. *Mathematical geology*, *20*, 405–421. https://doi.org/10/d8hfzg

Cressie, N., & Johannesson, G. (2008). Fixed rank kriging for very large spatial data sets. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, *70*(1), 209–226. https://doi.org/10/fq9f68

Dahl, A., & Bonilla, E. V. (2019). Grouped gaussian processes for solar power prediction. *Machine Learning*, *108*(8), 1287–1306. https://doi.org/10/gjhjht

De Oliveira, M. A., Possamai, O., Dalla Valentina, L. V., & Flesch, C. A. (2013). Modeling the leadership–project performance relation: Radial basis function, gaussian and kriging methods as alternatives to linear regression. *Expert Systems with Applications*, *40*(1), 272–280.

de Fouquet, C., Le Coz, M., Freulon, X., & Pannecoucke, L. (2023). Making kriging consistent with flow equations: Application of kriging with numerical covariances for estimating a contamination plume. *Hydrogeology Journal*, *31*(6), 1491–1503.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine learning research*, *7*, 1–30.

Dixit, V., Seshadrinath, N., & Tiwari, M. (2016). Performance measures based optimization of supply chain network resilience: A nsga-ii+ co-kriging approach. *Computers & Industrial Engineering*, *93*, 205–214.

Furrer, R., & Genton, M. G. (2011). Aggregation-cokriging for highly multivariate spatial data. *Biometrika*, *98*(3), 615–631. https://doi.org/10/djxb6d

Ganganath, N., Cheng, C.-T., & Tse, C. K. (2014). Data clustering with cluster size constraints using a modified k-means algorithm. *2014 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery*, 158–161. https://doi.org/10/gtn36d

Genton, M. G., & Kleiber, W. (2015). Cross-covariance functions for multivariate geostatistics. *Statistical Science*, *30*(2). https://doi.org/10/gh4j8k

Goovaerts, P. (2009). AUTO-IK: A 2d indicator kriging program for the automated non-parametric modeling of local uncertainty in earth sciences. *Computers & Geosciences*, *35*(6), 1255–1270. https://doi.org/10/cktgxr

Goovaerts, P. (1998). Ordinary cokriging revisited. *Mathematical Geology*, *30*, 21–42.

Gordon, A. (1996). A survey of constrained classification. *Computational Statistics & Data Analysis*, *21*(1), 17–29. https://doi.org/10/d4zv5x

Grandini, M., Bagli, E., & Visani, G. (2020). Metrics for multi-class classification: An overview. *arXiv preprint arXiv:2008.05756*.

Grossouvre, M., & Rullière, D. (2023). Supplementary material to: A joint kriging model with application to constrained classification [GitHub repository]. https://gitlab.com/urbs-imope/rdscripts/jointkrigingsupplementary

Höppner, F., & Klawonn, F. (2008). Clustering with size constraints. In L. C. Jain, M. Sato-Ilic, M. Virvou, G. A. Tsihrintzis, V. E. Balas, & C. Abeynayake (Eds.), *Computational intelligence paradigms: Innovative applications* (pp. 167–180). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-79474-5_8

Journel, A. G. (1983). Nonparametric estimation of spatial distributions. *Journal of the International Association for Mathematical Geology*, *15*, 445–468. https://doi.org/10/dn3nf3

Leroy, A., Latouche, P., Guedj, B., & Gey, S. (2022). MAGMA: Inference and prediction using multi-task gaussian processes with common mean. *Machine Learning*, *111*(5), 1821–1849. https://doi.org/10/gtn36k

Leroy, A., Latouche, P., Guedj, B., & Gey, S. (2023). Cluster-specific predictions with multi-task gaussian processes. *Journal of Machine Learning Research*, *24*(5), 1–49.

Martínez-Minaya, J., & Rue, H. (2023). A flexible bayesian tool for CoDa mixed models: Logistic-normal distribution with dirichlet covariance. *arXiv preprint arXiv:2308.13928*.

Meer, F. V. D. (1996). Classification of remotely-sensed imagery using an indicator kriging approach: Application to the problem of calcite-dolomite mineral mapping. *International Journal of Remote Sensing*, *17*(6), 1233–1249. https://doi.org/10/ckz384

Panos, A., Dellaportas, P., & Titsias, M. K. (2021). Large scale multi-label learning using gaussian processes. *Machine Learning*, *110*, 965–987. https://doi.org/10/gtn36h

Qi, B., Cheng, X., & Han, K. (2024). Research on hydrodynamic forces prediction of underwater vehicle based on co-kriging model. *Journal of Marine Science and Technology*, 1–12.

Rasmussen, C., & Ghahramani, Z. (2000). Occam's razor. *Advances in neural information processing systems*, *13*.

Rasmussen, C. E., & Williams, C. K. I. (2006). *Gaussian processes for machine learning*. MIT Press. https://gaussianprocess.org/gpml/chapters/RW.pdf

Rullière, D., Durrande, N., Bachoc, F., & Chevalier, C. (2018). Nested kriging predictions for datasets with a large number of observations. *Statistics and Computing*, *28*, 849–867. https://doi.org/10/gmt4z6

Simonoff, J. S. (1996). *Smoothing methods in statistics*. Springer-Verlag.

Toal, D. J. (2023). Applications of multi-fidelity multi-output kriging to engineering design optimization. *Structural and Multidisciplinary Optimization*, *66*(6), 125.

Tunçay, T., Alaboz, P., Dengiz, O., & Başkan, O. (2023). Application of regression kriging and machine learning methods to estimate soil moisture constants in a semi-arid terrestrial area. *Computers and Electronics in Agriculture*, *212*, 108118.

Ver Hoef, J. M., & Cressie, N. (1993). Multivariable spatial prediction. *Mathematical Geology*, *25*, 219–240. https://doi.org/10/bxj7qj

Vito, S. (2016). Air quality [UCI Machine Learning Repository].

Wackernagel, H. (2003). *Multivariate geostatistics: An introduction with applications*. Springer Science & Business Media.

Wang, X., Xiao, Y., Li, W., Wang, M., Zhou, Y., Chen, Y., & Li, Z. (2024). Kriging-based surrogate data-enriching artificial neural network prediction of strength and permeability of permeable cement-stabilized base. *Nature Communications*, *15*(1), 4891.

Williams, C. K., & Barber, D. (1998). Bayesian classification with gaussian processes. *IEEE Transactions on pattern analysis and machine intelligence*, *20*(12), 1342–1351. https://doi.org/10/dt7m5q

Yan, J., Liew, K. M., & He, L. (2012). A mesh-free computational framework for predicting buckling behaviors of single-walled carbon nanocones under axial compression based on the moving kriging interpolation. *Computer methods in applied mechanics and engineering*, *247*, 103–112.

Zafar, M. B., Valera, I., Gomez-Rodriguez, M., & Gummadi, K. P. (2019). Fairness constraints: A flexible approach for fair classification. *The Journal of Machine Learning Research*, *20*(1), 2737–2778.

A Proofs

A.1 Proof of Proposition 1 Simple Joint Kriging weights

Proof of Proposition 1. The proof is very similar to the geo-statistical proof of Simple Kriging model. It does not rely on any Gaussian assumption, but just on existing moments of order two. Recall that $\mathbb{W}$ is a symmetrical positive definite matrix, so that $\mathbb{W} = \mathbb{W}^\top$. Let us calculate the gradient of $\Delta(x^\star)$ with respect to $\boldsymbol{\alpha}(x^\star)$:

$$\begin{aligned}
&\nabla_{\boldsymbol{\alpha}(x^\star)} \mathrm{E}\left[\|\mathbf{M}(x^\star) - \mathbf{Y}(x^\star)\|^2_{\mathbb{W}}\right] \\
&= \nabla_{\boldsymbol{\alpha}(x^\star)} \mathrm{E}\left[(\mathbf{M}(x^\star) - \mathbf{Y}(x^\star))^\top \mathbb{W} (\mathbf{M}(x^\star) - \mathbf{Y}(x^\star))\right] \\
&= \nabla_{\boldsymbol{\alpha}(x^\star)} \mathrm{E}\left[(\mathbb{Y}\boldsymbol{\alpha}(x^\star) - \mathbf{Y}(x^\star))^\top \mathbb{W} (\mathbb{Y}\boldsymbol{\alpha}(x^\star) - \mathbf{Y}(x^\star))\right] \\
&= \nabla_{\boldsymbol{\alpha}(x^\star)} \mathrm{E}\left[\boldsymbol{\alpha}(x^\star)^\top \mathbb{Y}^\top \mathbb{W}\mathbb{Y}\boldsymbol{\alpha}(x^\star) - 2\boldsymbol{\alpha}(x^\star)^\top \mathbb{Y}^\top \mathbb{W}\mathbf{Y}(x^\star) + \mathbf{Y}(x^\star)^\top \mathbb{W}\mathbf{Y}(x^\star)\right] \\
&= 2\,\mathrm{E}\left[\mathbb{Y}^\top \mathbb{W}\mathbb{Y}\right] \boldsymbol{\alpha}(x^\star) - 2\,\mathrm{E}\left[\mathbb{Y}^\top \mathbb{W}\mathbf{Y}(x^\star)\right].
\end{aligned}$$

Thus,

$$\nabla_{\boldsymbol{\alpha}(x^\star)} \Delta(x^*) = 2\mathbb{K}\boldsymbol{\alpha}(x^\star) - 2\mathbf{h}(x^\star). \tag{23}$$

Where $\mathbb{K} := \mathrm{E}\left[\mathbb{Y}^\top \mathbb{W}\mathbb{Y}\right]$ is a $n \times n$ matrix and $\mathbf{h}(x^\star) := \mathrm{E}\left[\mathbb{Y}^\top \mathbb{W}\mathbf{Y}(x^\star)\right]$ is a $n \times 1$ vector, thus leading to a $n \times 1$ gradient. Hence $\boldsymbol{\alpha}(x^\star) = \mathbb{K}^{-1}\mathbf{h}(x^\star)$ in Equation (4) when the gradient is zero. The matrix expression $\mathbb{A} = \mathbb{K}^{-1}\mathbb{H}$ of Equation (5) is obtained by binding column vectors of Equation (4), for all prediction locations. Remark that, under assumption that $\mathrm{E}[\mathbf{Y}(x)] = \mathbf{0}_p$ for all $x \in \chi$, it is clear that $\mathrm{E}[\mathbf{M}(x^\star)] = \mathrm{E}[\mathbf{Y}(x^\star)] = 0$, so that the predictor is unbiased.

In that case, the (i, j) component of the matrix $\mathbb{Y}^\top\mathbb{Y}$ is

$$\left(\mathrm{E}\left[\mathbb{Y}^\top\mathbb{Y}\right]\right)_{ij} = \sum_{k=1}^{n} \mathrm{E}\left[Y_i(x_k)Y_j(x_k)\right] = \sum_{k=1}^{n} \mathrm{Cov}\left[Y_i(x_k), Y_j(x_k)\right].$$

Hence $\mathbb{Y}^\top\mathbb{Y}$ is a symmetric positive semi-definite matrix. The same holds for $\mathbb{K}$: writing $\mathbb{K} = \left(\mathbb{W}^{1/2}\mathbb{Y}\right)^\top \left(\mathbb{W}^{1/2}\mathbb{Y}\right)$, it is clear that for any vector $\mathbf{v}$, $\mathbf{v}^\top\mathbb{K}\mathbf{v} = \widetilde{\mathbf{v}}^\top\widetilde{\mathbf{v}} \geq 0$, where the vector $\widetilde{\mathbf{v}} := \mathbb{W}^{1/2}\mathbb{Y}\mathbf{v}$. Thus $\mathbb{K}$ is a symmetric semi-definite positive matrix, i.e. a covariance matrix. □

A.2 Proof of Proposition 2 Ordinary Joint Kriging weights

Proof of Proposition 2. Under the constraint (6), and using a Lagrange multiplier $\lambda \in \mathbb{R}$, the loss to minimise is

$$\Delta_1(x^\star) := \Delta(x^\star) - 2\lambda(x^\star)\left(\boldsymbol{\alpha}(x^\star)^\top \mathbf{1}_n - 1\right)$$

Using Equation (23), the gradient of $\Delta_1(x^\star)$ with respect to $\boldsymbol{\alpha}(x^\star)$ is

$$\nabla_{\boldsymbol{\alpha}(x^\star)} \Delta_1(x^\star) = 2\,\mathrm{E}\left[\mathbb{Y}^\top \mathbb{W}\mathbb{Y}\right] \boldsymbol{\alpha}(x^\star) - 2\,\mathrm{E}\left[\mathbb{Y}^\top \mathbb{W}\mathbf{Y}(x^\star)\right] - 2\lambda(x^\star)\mathbf{1}_n \tag{24}$$

Setting this $\nabla_{\boldsymbol{\alpha}(x^\star)} \Delta_1(x^\star)$ to be zero for all of its p components, we get

$$\mathrm{E}\left[\mathbb{Y}^\top \mathbb{W}\mathbb{Y}\right] \boldsymbol{\alpha}(x^\star) = \mathrm{E}\left[\mathbb{Y}^\top \mathbb{W}\mathbf{Y}(x^\star)\right] + \lambda(x^\star)\mathbf{1}_n.$$

and finally,

$$\mathbb{K}\boldsymbol{\alpha}(x^\star) = \mathbf{h}(x^\star) + \lambda(x^\star)\mathbf{1}_n.$$

Once $\boldsymbol{\alpha}(x^\star)$ is written as a function of $\lambda(x^\star)$, one easily gets the value of $\lambda(x^\star)$ by setting ${\mathbf{1}_n}^\top \boldsymbol{\alpha}(x^\star) = 1$. Hence the result. Matrix expressions are obtained by binding column vectors for all $x^\star$ in $\{x_1^\star, \ldots, x_q^\star\}$ □

A.3 Proof of Remark 1 Covariance matrices

Proof of Remark 1. Recall that $\mathbb{K} = \mathrm{E}\left[\mathbb{Y}^\top \mathbb{W} \mathbb{Y}\right]$ and $\mathbf{h}(x^\star) = \mathrm{E}\left[\mathbb{Y}^\top \mathbb{W} \mathbf{Y}(x^\star)\right]$. Under the chosen mean assumption, both $\mathrm{E}\left[\mathbf{Y}(x^\star)\right] = \boldsymbol{\mu}$ and $\mathrm{E}\left[\mathbb{Y}\right] = \boldsymbol{\mu}{\mathbf{1}_n}^\top$. Thus, under the given constraint ${\boldsymbol{\alpha}(x^\star)}^\top \mathbf{1}_n = 1$, or when $\boldsymbol{\mu} = \mathbf{0}_p$,

$$\mathrm{E}\left[\mathbb{Y}^\top\right] \mathbb{W}\, \mathrm{E}\left[\mathbb{Y}\right] \boldsymbol{\alpha}(x^\star) = \mathrm{E}\left[\mathbb{Y}^\top\right] \mathbb{W}\, \mathrm{E}\left[\mathbf{Y}(x^\star)\right] = \mathbf{1}_n \boldsymbol{\mu}^\top \mathbb{W} \boldsymbol{\mu}.$$

Hence the gradient of $\Delta(x^\star)$ in Equation (23) also writes

$$\nabla_{\boldsymbol{\alpha}(x^\star)} \Delta(x^*) = 2\mathbb{K}\boldsymbol{\alpha}(x^\star) - 2\mathbf{h}(x^\star) = 2\widetilde{\mathbb{K}}\boldsymbol{\alpha}(x^\star) - 2\widetilde{\mathbf{h}}(x^\star).$$

As a consequence, the gradient of $\Delta_1(x^\star)$ in Equation (24) is unchanged when replacing both ($\mathbb{K}$, $\mathbf{h}$) by ($\widetilde{\mathbb{K}}$, $\widetilde{\mathbf{h}}$). Thus, one can freely replace both ($\mathbb{K}$, $\mathbf{h}$) by ($\widetilde{\mathbb{K}}$, $\widetilde{\mathbf{h}}$) in the rest of the proof of Proposition 2, without changing the result. □

A.4 Proof of Proposition 3 Joint Kriging weights under a predicted values constraint

Let us first study the rank of the system of constraints (9):

$$\begin{cases} {\mathbb{A}}^\top \mathbf{1}_n = \mathbf{1}_q \\ \mathbb{Y}\mathbb{A}\boldsymbol{\pi} = \mathbf{m} \end{cases} \tag{25}$$

We denote $a_{ij} := \alpha_i(x_j^\star)$, $y_{ij} := Y_j(x_i)$, $\pi_i := \pi_{x_i^\star}$, $m_i$the i-th component of $\mathbf{m}$. The system of constraints rewrites:

$$\begin{pmatrix} 1 & \ldots & 1 & 0 & \ldots & 0 & \ldots & 0 & \ldots & 0 \\ 0 & \ldots & 0 & 1 & \ldots & 1 & \ldots & 0 & \ldots & 0 \\ \vdots & & \vdots & \vdots & & \vdots & & \vdots & & \\ 0 & \ldots & 0 & 0 & \ldots & 0 & \ldots & 1 & \ldots & 1 \\ \pi_1 y_{11} & \ldots \pi_1 y_{1n} & \pi_2 y_{11} & \ldots & \pi_2 y_{1n} & \ldots & \pi_q y_{11} & \ldots & \pi_q y_{1n} & \\ \vdots & & \vdots & \vdots & & \vdots & & \vdots & & \\ \pi_1 y_{p1} & \ldots \pi_1 y_{pn} & \pi_2 y_{p1} & \ldots & \pi_2 y_{pn} & \ldots & \pi_q y_{p1} & \ldots & \pi_q y_{pn} & \end{pmatrix} \begin{pmatrix} a_{11} \\ \vdots \\ a_{n1} \\ a_{12} \\ \vdots \\ a_{n2} \\ \vdots \\ a_{1q} \\ \vdots \\ a_{nq} \end{pmatrix} = \begin{pmatrix} 1 \\ \vdots \\ \vdots \\ 1 \\ m_1 \\ \vdots \\ m_p \end{pmatrix}$$

Now we have to study the matrix of constraints. The reader can recognise that the last p rows are an aggregation of q times the matrix $\mathbb{Y}$, each time multiplied by a different factor.

- It is clear that the first q rows are linearly independent.

- If the p last rows are linearly dependent then, taking into account that $\boldsymbol{\pi}$ is strictly positive, it means that there exists at least one vector $\boldsymbol{\omega} \in \mathbb{R}^p \setminus \{\mathbf{0}_p\}$ such that:

$$\mathbb{Y}^\top \boldsymbol{\omega} = \mathbf{0}_n = 0\mathbf{1}_n$$

- If the p last rows are linearly dependent with the q first one, it means that there exists at least one vector $\boldsymbol{\omega} \in \mathbb{R}^p$ and one scalar ω_0 such that:

$$\mathbb{Y}^\top \boldsymbol{\omega} = \omega_0 \mathbf{1}_n$$

Therefore, the system is not of full rank if and only if there exists $\boldsymbol{\omega}$ and ω_0 such that:

$$\mathbb{Y}^\top \boldsymbol{\omega} = \omega_0 \mathbf{1}_n$$

Depending on the situation, the matrix of constraints can be of rank ranging from $q+1$ up to $q+p$. In the following, we are interested in two important cases, when the matrix of constraints is of full rank $q+p$ and when it is of rank $q+p-1$. The second case is useful for fuzzy classification. Other cases are of no interest for our study although they could also be treated removing a sufficient number of constraints in the system. Note that a system of rank lower than $q+p-1$ corresponds to the cases where $\mathbb{Y}$ carries little information and a variable selection should be implemented.

Proof of Proposition 3 when the system of Equations (9) *is of full rank* $q+p$.

$$\Delta_2(x^\star) := \Delta(x^\star) - 2\lambda(x^\star)\left(\boldsymbol{\alpha}(x^\star)^\top \mathbf{1}_n - 1\right) - 2\boldsymbol{\lambda}'^\top \left(\mathbb{Y}\mathbb{A}\boldsymbol{\pi} - \mathbf{m}\right),$$

where $\boldsymbol{\lambda}'$ is a $p \times 1$ vector of Lagrange multipliers. The gradient of the last term, with respect to $\boldsymbol{\alpha}(x^\star)$ is

$$\begin{aligned}
&\nabla_{\boldsymbol{\alpha}(x^\star)} 2\boldsymbol{\lambda}'^\top \left(\mathrm{E}\left[\mathbf{M}(X^\star) \,|\, \mathbb{Y}\right] - \mathbf{m}\right) \\
&= \nabla_{\boldsymbol{\alpha}(x^\star)} 2\boldsymbol{\lambda}'^\top \left(\mathrm{P}\left[X^\star = x^\star\right] \mathrm{E}\left[\mathbf{M}(x^\star) \,|\, \mathbb{Y}\right] + \mathrm{P}\left[X^\star \neq x^\star\right] \mathrm{E}\left[\mathbf{M}(X^\star) \,|\, X^\star \neq x^\star, \mathbb{Y}\right] - \mathbf{m}\right) \\
&= \nabla_{\boldsymbol{\alpha}(x^\star)} 2\boldsymbol{\lambda}'^\top \left(\mathrm{P}\left[X^\star = x^\star\right] \mathrm{E}\left[\mathbf{M}(x^\star) \,|\, \mathbb{Y}\right] - \mathbf{m}\right) + 0 \\
&= \nabla_{\boldsymbol{\alpha}(x^\star)} 2\boldsymbol{\lambda}'^\top \mathrm{P}\left[X^\star = x^\star\right] \mathrm{E}\left[\mathbb{Y}\boldsymbol{\alpha}(x^\star) - \mathbf{m} \,|\, \mathbb{Y}\right] \\
&= \nabla_{\boldsymbol{\alpha}(x^\star)} 2\boldsymbol{\lambda}'^\top \left(\pi_{x^\star} \mathbb{Y}\boldsymbol{\alpha}(x^\star) - \mathbf{m}\right) \\
&= 2\pi_{x^\star} \mathbb{Y}^\top \boldsymbol{\lambda}'
\end{aligned}$$

Hence using the gradient of $\Delta(x^\star)$ in Equation (23), one gets

$$\nabla_{\boldsymbol{\alpha}(x^\star)} \Delta_2(x^\star) = 2\mathbb{K}\boldsymbol{\alpha}(x^\star) - 2\mathbf{h}(x^\star) - 2\lambda(x^\star)\mathbf{1}_n - 2\pi_{x^\star}\mathbb{Y}^\top \boldsymbol{\lambda}' \tag{26}$$

Setting the gradient to be equal to a $n \times 1$ vector of zeros, we get for all prediction locations $x^\star \in \{x_1^\star, \ldots, x_q^\star\}$

$$\begin{cases}
\mathbb{K}\boldsymbol{\alpha}(x^\star) = \mathbf{h}(x^\star) + \lambda(x^\star)\mathbf{1}_n + \pi_{x^\star}\mathbb{Y}^\top \boldsymbol{\lambda}' \\
\mathbf{1}_n{}^\top \boldsymbol{\alpha}(x^\star) = 1 \\
\mathbb{Y}\mathbb{A}\boldsymbol{\pi} = \mathbf{m}
\end{cases}$$

As optimal weights are gathered in the $n \times q$ matrix $\mathbb{A} := \big[\boldsymbol{\alpha}(x_1^\star), ..., \boldsymbol{\alpha}(x_q^\star)\big]$, if one defines the $n \times q$ matrix $\mathbb{H} := \big[\mathbf{h}(x_1^\star), ..., \mathbf{h}(x_q^\star)\big]$, then the previous system can be written, by binding columns for all $x^\star \in \{x_1^\star, \dots, x_q^\star\}$:

$$\begin{cases} \mathbb{K}\mathbb{A} = \mathbb{H} + \mathbf{1}_n \boldsymbol{\lambda}^\top + \mathbb{Y}^\top \boldsymbol{\lambda}' \boldsymbol{\pi}^\top \\ {\mathbf{1}_n}^\top \mathbb{A} = {\mathbf{1}_q}^\top \\ \mathbb{Y}\mathbb{A}\boldsymbol{\pi} = \mathbf{m} \end{cases} \tag{27}$$

with $q \times 1$ Lagrange multiplier $\boldsymbol{\lambda}$, and $p \times 1$ Lagrange multiplier $\boldsymbol{\lambda}'$.

If $\mathbb{K}$ is invertible, then the first equation writes

$$\mathbb{A} = \mathbb{K}^{-1}\mathbb{H} + \mathbb{K}^{-1}\mathbf{1}_n \boldsymbol{\lambda}^\top + \mathbb{K}^{-1}\mathbb{Y}^\top \boldsymbol{\lambda}' \boldsymbol{\pi}^\top$$

Injecting this value of $\mathbb{A}$ into the first constraint ${\mathbf{1}_n}^\top \mathbb{A} = {\mathbf{1}_q}^\top$, denoting $\gamma := \boldsymbol{\pi}^\top \boldsymbol{\pi} \in \mathbb{R}$ and $\delta := {\mathbf{1}_n}^\top \mathbb{K}^{-1} \mathbf{1}_n \in \mathbb{R}$ one gets:

$$\begin{aligned} {\mathbf{1}_q}^\top &= {\mathbf{1}_n}^\top \mathbb{K}^{-1}\mathbb{H} + {\mathbf{1}_n}^\top \mathbb{K}^{-1}\mathbf{1}_n \boldsymbol{\lambda}^\top + {\mathbf{1}_n}^\top \mathbb{K}^{-1}\mathbb{Y}^\top \boldsymbol{\lambda}' \boldsymbol{\pi}^\top \\ {\mathbf{1}_q}^\top \boldsymbol{\pi} &= {\mathbf{1}_n}^\top \mathbb{K}^{-1}\mathbb{H}\boldsymbol{\pi} + {\mathbf{1}_n}^\top \mathbb{K}^{-1}\mathbf{1}_n \boldsymbol{\lambda}^\top \boldsymbol{\pi} + {\mathbf{1}_n}^\top \mathbb{K}^{-1}\mathbb{Y}^\top \boldsymbol{\lambda}' \boldsymbol{\pi}^\top \boldsymbol{\pi} \\ 1 &= {\mathbf{1}_n}^\top \mathbb{K}^{-1}\mathbb{H}\boldsymbol{\pi} + \delta \boldsymbol{\lambda}^\top \boldsymbol{\pi} + \gamma {\mathbf{1}_n}^\top \mathbb{K}^{-1}\mathbb{Y}^\top \boldsymbol{\lambda}' \\ \delta \boldsymbol{\lambda}^\top \boldsymbol{\pi} &= 1 - {\mathbf{1}_n}^\top \mathbb{K}^{-1}\mathbb{H}\boldsymbol{\pi} - \gamma {\mathbf{1}_n}^\top \mathbb{K}^{-1}\mathbb{Y}^\top \boldsymbol{\lambda}' \end{aligned}$$

Now injecting the value of $\mathbb{A}$ into the second constraint $\mathbb{Y}\mathbb{A}\boldsymbol{\pi} = \mathbf{m}$, and using the last equation, denoting the $p \times 1$ vector $\mathbf{u} := \mathbb{Y}\mathbb{K}^{-1}\mathbf{1}_n$, one gets

$$\begin{aligned} \mathbf{m} &= \mathbb{Y}\mathbb{K}^{-1}\mathbb{H}\boldsymbol{\pi} + \mathbb{Y}\mathbb{K}^{-1}\mathbf{1}_n \boldsymbol{\lambda}^\top \boldsymbol{\pi} + \mathbb{Y}\mathbb{K}^{-1}\mathbb{Y}^\top \boldsymbol{\lambda}' \boldsymbol{\pi}^\top \boldsymbol{\pi} \\ &= \mathbb{Y}\mathbb{K}^{-1}\mathbb{H}\boldsymbol{\pi} + \mathbb{Y}\mathbb{K}^{-1}\mathbf{1}_n \frac{1}{\delta}\left(1 - {\mathbf{1}_n}^\top \mathbb{K}^{-1}\mathbb{H}\boldsymbol{\pi} - \gamma {\mathbf{1}_n}^\top \mathbb{K}^{-1}\mathbb{Y}^\top \boldsymbol{\lambda}'\right) + \gamma \mathbb{Y}\mathbb{K}^{-1}\mathbb{Y}^\top \boldsymbol{\lambda}' \\ &= \mathbb{Y}\mathbb{K}^{-1}\mathbb{H}\boldsymbol{\pi} + \frac{1}{\delta}\mathbf{u}\left(1 - {\mathbf{1}_n}^\top \mathbb{K}^{-1}\mathbb{H}\boldsymbol{\pi}\right) - \gamma \frac{1}{\delta}\mathbf{u}\mathbf{u}^\top \boldsymbol{\lambda}' + \gamma \mathbb{Y}\mathbb{K}^{-1}\mathbb{Y}^\top \boldsymbol{\lambda}' \end{aligned}$$

and finally, the vector $\boldsymbol{\lambda}'$ must satisfies

$$\gamma\left(\frac{1}{\delta}\mathbf{u}\mathbf{u}^\top - \mathbb{Y}\mathbb{K}^{-1}\mathbb{Y}^\top\right)\boldsymbol{\lambda}' = \mathbb{Y}\mathbb{K}^{-1}\mathbb{H}\boldsymbol{\pi} + \frac{1}{\delta}\mathbf{u}\left(1 - {\mathbf{1}_n}^\top \mathbb{K}^{-1}\mathbb{H}\boldsymbol{\pi}\right) - \mathbf{m}.$$

Hence, provided the matrix factor is invertible,

$$\boldsymbol{\lambda}' = \gamma^{-1}\left(\frac{1}{\delta}\mathbf{u}\mathbf{u}^\top - \mathbb{Y}\mathbb{K}^{-1}\mathbb{Y}^\top\right)^{-1}\left(\mathbb{Y}\mathbb{K}^{-1}\mathbb{H}\boldsymbol{\pi} + \frac{1}{\delta}\mathbf{u}\left(1 - {\mathbf{1}_n}^\top \mathbb{K}^{-1}\mathbb{H}\boldsymbol{\pi}\right) - \mathbf{m}\right)$$

Once $\boldsymbol{\lambda}'$ computed, one gets for $\boldsymbol{\lambda}$

$$\begin{aligned} {\mathbf{1}_q}^\top &= {\mathbf{1}_n}^\top \mathbb{K}^{-1}\mathbb{H} + \delta \boldsymbol{\lambda}^\top + {\mathbf{1}_n}^\top \mathbb{K}^{-1}\mathbb{Y}^\top \boldsymbol{\lambda}' \boldsymbol{\pi}^\top \\ \delta \boldsymbol{\lambda}^\top &= -{\mathbf{1}_n}^\top \mathbb{K}^{-1}\mathbb{H} - {\mathbf{1}_n}^\top \mathbb{K}^{-1}\mathbb{Y}^\top \boldsymbol{\lambda}' \boldsymbol{\pi}^\top + {\mathbf{1}_q}^\top \end{aligned}$$

And finally, using $\mathbf{u} = \mathbb{Y}\mathbb{K}^{-1}\mathbf{1}_n$,

$$\boldsymbol{\lambda} = \delta^{-1}\left(\mathbf{1}_q - \mathbb{H}^\top \mathbb{K}^{-1}\mathbf{1}_n - \boldsymbol{\pi}{\boldsymbol{\lambda}'}^\top \mathbf{u}\right)$$

□

The above proof of Proposition 3 in the case where all constraints are independent relies on the invertibility of the $p \times p$ matrix $\mathbb{S} = \frac{1}{\delta}\mathbf{u}\mathbf{u}^\top - \mathbb{Y}\mathbb{K}^{-1}\mathbb{Y}^\top$. Let us now discuss this condition.

Let us assume take a vector $\boldsymbol{\omega}$, not null, such that $\mathbb{S}\boldsymbol{\omega} = \mathbf{0}_p$. It implies that $\boldsymbol{\omega}^\top\mathbb{S}\boldsymbol{\omega} = 0$. And rewriting $\mathbb{S}$, we get:

$$0 = \boldsymbol{\omega}^\top \left(\frac{1}{\mathbf{1}_n{}^\top\mathbb{K}^{-1}\mathbf{1}_n}\mathbb{Y}\mathbb{K}^{-1}\mathbf{1}_n\mathbf{1}_n{}^\top\mathbb{K}^{-1}\mathbb{Y}^\top - \mathbb{Y}\mathbb{K}^{-1}\mathbb{Y}^\top \right)\boldsymbol{\omega}$$

We denote $\mathbf{w}_n = \mathbb{Y}^\top\boldsymbol{\omega}$ and get:

$$\frac{\left(\mathbf{w}_n{}^\top\mathbb{K}^{-1}\mathbf{1}_n\right)\left(\mathbf{1}_n{}^\top\mathbb{K}^{-1}\mathbf{w}_n\right)}{\mathbf{1}_n{}^\top\mathbb{K}^{-1}\mathbf{1}_n} = \mathbf{w}_n{}^\top\mathbb{K}^{-1}\mathbf{w}_n$$

And since $\mathbb{K}$ is a definite positive symmetric matrix, its inverse too and this inverse can be seen as a scalar product denoted $< ., . >$. We get:

$$< \mathbf{w}_n, \mathbf{1}_n >^2 = < \mathbf{w}_n, \mathbf{w}_n >< \mathbf{1}_n, \mathbf{1}_n >$$

Due to Cauchy-Schwartz inequality, this is possible if and only if we, $\mathbf{w}_n = \omega_0\mathbf{1}_n$ for some scalar ω_0. Which means that $\mathbb{Y}^\top\boldsymbol{\omega} = \omega_0\mathbf{1}_n$. But this case has been excluded because it implies that the system of constraints is not of full rank (see Equation (A.4)). Therefore, the matrix $\mathbb{S}$ is always invertible.

Proof of Proposition 3 when the system of Equations (9) *is of rank* $q + p - 1$. Theory of Lagrangian factors holds only in the case of regular constraints, meaning that constraints' gradients should be independent. In particular, constraints are not regular if the constraints themselves are not regular. Let us show how to solve the optimisation problem, removing one of the conditions on the optimal weights. For the sake of simplicity, we remove the first one. Keeping the above notations, we denote:

$$\begin{aligned}
\boldsymbol{\lambda}_0 &= (0, \lambda_2, \ldots, \lambda_q)^\top \in \mathbb{R}^{q-1} \\
\boldsymbol{\lambda}_1 &= (\lambda_2, \ldots, \lambda_q)^\top \in \mathbb{R}^{q-1} \\
\boldsymbol{\pi}_1 &= \left(\pi_{x_2^\star,\ldots,x_q^\star}\right)^\top \in \mathbb{R}_+^{q-1} \\
\pi_1 &= \pi_{x_1^\star} \\
\mathbb{A}_1 &= \left[\boldsymbol{\alpha}_{x_2^\star}, \ldots, \boldsymbol{\alpha}_{x_q^\star}\right] \in \mathbb{R}^{p\times(q-1)} \\
\gamma_1 &= \boldsymbol{\pi}_1{}^\top\boldsymbol{\pi}_1 = \gamma - \pi_1^2 \in \mathbb{R}_+ \\
\mathbb{H}_1 &= \left[\mathbf{h}(x_2^\star, \ldots, x_q^\star\right]
\end{aligned}$$

The constraints (6) rewrite:

$$\mathbf{1}_n{}^\top\mathbb{A}_1 = \mathbf{1}_q{}^\top$$

And the system of Equations (27) rewrites:

$$\begin{cases}
\mathbb{K}\mathbb{A} = \mathbb{H} + \mathbf{1}_n\boldsymbol{\lambda}_0{}^\top + \mathbb{Y}^\top\boldsymbol{\lambda}'\boldsymbol{\pi}^\top \\
\mathbf{1}_n{}^\top\mathbb{A}_1 = \mathbf{1}_{q-1}{}^\top \\
\mathbb{Y}\mathbb{A}\boldsymbol{\pi} = \mathbf{m}
\end{cases} \tag{28}$$

The first equation implies:

$$\begin{aligned}\mathbb{A} &= \mathbb{K}^{-1}\mathbb{H} + \mathbb{K}^{-1}\mathbf{1}_n\boldsymbol{\lambda}_0{}^\top + \mathbb{K}^{-1}\mathbb{Y}^\top\boldsymbol{\lambda}'\boldsymbol{\pi}^\top \\ \text{therefore } \mathbb{A}_1 &= \mathbb{K}^{-1}\mathbb{H}_1 + \mathbb{K}^{-1}\mathbf{1}_n\boldsymbol{\lambda}_1{}^\top + \mathbb{K}^{-1}\mathbb{Y}^\top\boldsymbol{\lambda}'\boldsymbol{\pi}_1{}^\top\end{aligned}$$

We replace $\mathbb{A}_1$ in the second equation:

$$\begin{aligned}\mathbf{1}_{q-1}{}^\top &= \mathbf{1}_n{}^\top\mathbb{K}^{-1}\mathbb{H}_1 + \mathbf{1}_n{}^\top\mathbb{K}^{-1}\mathbf{1}_n\boldsymbol{\lambda}_1{}^\top + \mathbf{1}_n{}^\top\mathbb{K}^{-1}\mathbb{Y}^\top\boldsymbol{\lambda}'\boldsymbol{\pi}_1{}^\top \\ \mathbf{1}_{q-1}{}^\top\boldsymbol{\pi}_1 &= \mathbf{1}_n{}^\top\mathbb{K}^{-1}\mathbb{H}_1\boldsymbol{\pi}_1 + \mathbf{1}_n{}^\top\mathbb{K}^{-1}\mathbf{1}_n\boldsymbol{\lambda}_1{}^\top\boldsymbol{\pi}_1 + \mathbf{1}_n{}^\top\mathbb{K}^{-1}\mathbb{Y}^\top\boldsymbol{\lambda}'\boldsymbol{\pi}_1{}^\top\boldsymbol{\pi}_1 \\ 1-\pi_1 &= \mathbf{1}_n{}^\top\mathbb{K}^{-1}\mathbb{H}_1\boldsymbol{\pi}_1 + \delta\boldsymbol{\lambda}_1{}^\top\boldsymbol{\pi}_1 + \gamma_1\mathbf{u}^\top\boldsymbol{\lambda}' \\ \delta\boldsymbol{\lambda}_1{}^\top\boldsymbol{\pi}_1 &= 1-\pi_1 - \mathbf{1}_n{}^\top\mathbb{K}^{-1}\mathbb{H}_1\boldsymbol{\pi}_1 - \gamma_1\mathbf{u}^\top\boldsymbol{\lambda}' \\ \boldsymbol{\lambda}_0{}^\top\boldsymbol{\pi} &= \frac{1}{\delta}\left(1-\pi_1 - \mathbf{1}_n{}^\top\mathbb{K}^{-1}\mathbb{H}_1\boldsymbol{\pi}_1 - \gamma_1\mathbf{u}^\top\boldsymbol{\lambda}'\right)\end{aligned}$$

We can also replace $\mathbb{A}$ in the third equation and replace $\boldsymbol{\lambda}_0{}^\top\boldsymbol{\pi}$ with the last result:

$$\begin{aligned}m &= \mathbb{Y}\mathbb{K}^{-1}\mathbb{H}\boldsymbol{\pi} + \mathbb{Y}\mathbb{K}^{-1}\mathbf{1}_n\boldsymbol{\lambda}_0{}^\top\boldsymbol{\pi} + \mathbb{Y}\mathbb{K}^{-1}\mathbb{Y}^\top\boldsymbol{\lambda}'\boldsymbol{\pi}^\top\boldsymbol{\pi} \\ m &= \mathbb{Y}\mathbb{K}^{-1}\mathbb{H}\boldsymbol{\pi} + \mathbf{u}\boldsymbol{\lambda}_0{}^\top\boldsymbol{\pi} + \gamma\mathbb{Y}\mathbb{K}^{-1}\mathbb{Y}^\top\boldsymbol{\lambda}' \\ m &= \mathbb{Y}\mathbb{K}^{-1}\mathbb{H}\boldsymbol{\pi} + \frac{1}{\delta}\mathbf{u}\left(1-\pi_1 - \mathbf{1}_n{}^\top\mathbb{K}^{-1}\mathbb{H}_1\boldsymbol{\pi}_1 - \gamma_1\mathbf{u}^\top\boldsymbol{\lambda}'\right) + \gamma\mathbb{Y}\mathbb{K}^{-1}\mathbb{Y}^\top\boldsymbol{\lambda}' \\ m &= \mathbb{Y}\mathbb{K}^{-1}\mathbb{H}\boldsymbol{\pi} + \frac{1-\pi_1}{\delta}\mathbf{u} - \frac{1}{\delta}\mathbf{u}\mathbf{1}_n{}^\top\mathbb{K}^{-1}\mathbb{H}_1\boldsymbol{\pi}_1 - \frac{\gamma_1}{\delta}\mathbf{u}\mathbf{u}^\top\boldsymbol{\lambda}' + \gamma\mathbb{Y}\mathbb{K}^{-1}\mathbb{Y}^\top\boldsymbol{\lambda}'\end{aligned}$$

Which yields:

$$\left(\frac{\gamma_1}{\delta}\mathbf{u}\mathbf{u}^\top - \gamma\mathbb{Y}\mathbb{K}^{-1}\mathbb{Y}^\top\right)\boldsymbol{\lambda}' = \mathbb{Y}\mathbb{K}^{-1}\mathbb{H}\boldsymbol{\pi} + \frac{1-\pi_1}{\delta}\mathbf{u} - \frac{1}{\delta}\mathbf{u}\mathbf{1}_n{}^\top\mathbb{K}^{-1}\mathbb{H}_1\boldsymbol{\pi}_1 - m$$

Assuming that $\frac{\gamma_1}{\delta}\mathbf{u}\mathbf{u}^\top - \gamma\mathbb{Y}\mathbb{K}^{-1}\mathbb{Y}^\top$ is invertible:

$$\boldsymbol{\lambda}' = \left(\frac{\gamma_1}{\delta}\mathbf{u}\mathbf{u}^\top - \gamma\mathbb{Y}\mathbb{K}^{-1}\mathbb{Y}^\top\right)^{-1}\left(\mathbb{Y}\mathbb{K}^{-1}\mathbb{H}\boldsymbol{\pi} + \frac{1-\pi_1}{\delta}\mathbf{u} - \frac{1}{\delta}\mathbf{u}\mathbf{1}_n{}^\top\mathbb{K}^{-1}\mathbb{H}_1\boldsymbol{\pi}_1 - m\right)$$

But we know that:

$$\begin{aligned}\mathbf{1}_{q-1}{}^\top &= \mathbf{1}_n{}^\top\mathbb{K}^{-1}\mathbb{H}_1 + \delta\boldsymbol{\lambda}_1{}^\top + \mathbf{u}^\top\boldsymbol{\lambda}'\boldsymbol{\pi}_1{}^\top \\ \text{therefore } \boldsymbol{\lambda}_1 &= \frac{1}{\delta}\left(\mathbf{1}_{q-1} - \mathbb{H}_1{}^\top\mathbb{K}^{-1}\mathbf{1}_n - \boldsymbol{\pi}_1\boldsymbol{\lambda}'{}^\top\mathbf{u}\right)\end{aligned}$$

And $\mathbb{A}$ can finally be computed with the equation:

$$\mathbb{A} = \mathbb{K}^{-1}\mathbb{H} + \mathbb{K}^{-1}\mathbf{1}_n\boldsymbol{\lambda}_0{}^\top + \mathbb{K}^{-1}\mathbb{Y}^\top\boldsymbol{\lambda}'\boldsymbol{\pi}^\top$$

□

A.5 Proof of Remark 2 Covariance matrices with two constraints

Proof of Remark 2. - The proof is similar to the one of Remark 1 and uses the fact that, under chosen assumptions and for any prediction point $x^\star$,

$$\mathbb{K}\boldsymbol{\alpha}(x^\star) - \mathbf{h}(x^\star) = \widetilde{\mathbb{K}}\boldsymbol{\alpha}(x^\star) - \widetilde{\mathbf{h}}(x^\star).$$

Hence the gradient of $\Delta_2(x^\star)$ in Equation (26) is unchanged when replacing $\mathbb{K}$ and $\mathbf{h}(x^\star)$ by $\widetilde{\mathbb{K}}$ and $\widetilde{\mathbf{h}}(x^\star)$, and all further expressions follows the same way in the proof of Proposition 3.

□

A.6 Proof of Proposition 5 Joint Kriging variance with arbitrary weights

Proof of Proposition 5. The first equation is a simple vector rewriting of Equation (1). For the prediction error, one simply write, whatever the weights $\boldsymbol{\alpha}(x^\star)$,

$$\begin{aligned}
\Delta(x^\star) &= \mathrm{E}\left[\|\mathbf{M}(x^\star) - \mathbf{Y}(x^\star)\|_{\mathbb{W}}^2\right] \\
&= \mathrm{E}\left[(\mathbf{M}(x^\star) - \mathbf{Y}(x^\star))^\top \mathbb{W} (\mathbf{M}(x^\star) - \mathbf{Y}(x^\star))\right] \\
&= \mathrm{E}\left[(\mathbb{Y}\boldsymbol{\alpha}(x^\star) - \mathbf{Y}(x^\star))^\top \mathbb{W} (\mathbb{Y}\boldsymbol{\alpha}(x^\star) - \mathbf{Y}(x^\star))\right] \\
&= \mathrm{E}\left[\boldsymbol{\alpha}(x^\star)^\top \mathbb{Y}^\top \mathbb{W}\mathbb{Y}\boldsymbol{\alpha}(x^\star) - 2\boldsymbol{\alpha}(x^\star)^\top \mathbb{Y}^\top \mathbb{W}\mathbf{Y}(x^\star) + \mathbf{Y}(x^\star)^\top \mathbb{W}\mathbf{Y}(x^\star)\right].
\end{aligned}$$

Hence the result. □

A.7 Proof of Remark 4 Covariance matrices in Joint Kriging mean and variance

Proof of Remark 4. The case where $\boldsymbol{\mu} = \mathbf{0}_p$ is straightforward, as in that case $\widetilde{\mathbb{K}} = \mathbb{K}$, $\widetilde{\mathbf{h}}(x^\star) = \mathbf{h}(x^\star)$ and $\widetilde{v}(x^\star) = v(x^\star)$, whatever the weights $\boldsymbol{\alpha}(x^\star)$. It remains the case where weights are summing to one. As in previous remarks, under chosen assumptions one gets

$$\mathbb{K}\boldsymbol{\alpha}(x^\star) - \mathbf{h}(x^\star) = \widetilde{\mathbb{K}}\boldsymbol{\alpha}(x^\star) - \widetilde{\mathbf{h}}(x^\star),$$

and moreover one can show that

$$-\alpha(x^\star)^\top \mathbf{h}(x^\star) + v(x^\star) = -\alpha(x^\star)^\top \widetilde{\mathbf{h}}(x^\star) + \widetilde{v}(x^\star).$$

Hence the result. □

A.8 Proof of Proposition 6 Variance sharing

Proof of Proposition 6. The difficulty here is to derive the cross-covariance $k_{ij}(x, x') = \mathrm{Cov}\left[Y_i(x), Y_j(x')\right]$ from the expression of $k(x, x')$ that is detailed in Remark 3

$$k(x, x') := \mathrm{E}\left[\mathbf{Y}(x)^\top \mathbb{W}\mathbf{Y}(x')\right] - \mathrm{E}\left[\mathbf{Y}(x)^\top\right] \mathbb{W}\, \mathrm{E}\left[\mathbf{Y}(x')\right]$$

Denoting $\widetilde{\mathbf{Y}}(x) := \mathbb{W}^{1/2}\mathbf{Y}(x)$, $x \in \chi$, this scalar covariance writes

$$k(x, x') = \mathrm{E}\left[\widetilde{\mathbf{Y}}(x)^\top \widetilde{\mathbf{Y}}(x')\right] - \mathrm{E}\left[\widetilde{\mathbf{Y}}(x)^\top\right] \mathrm{E}\left[\widetilde{\mathbf{Y}}(x')\right] \tag{29}$$

One would like to compute the $p \times p$ cross-covariance matrix between $\mathbf{Y}(x)$ and $\mathbf{Y}(x')$, using $\mathbf{Y}(x) = \mathbb{W}^{-1/2}\widetilde{\mathbf{Y}}(x)$, $x \in \chi$:

$$\begin{aligned}
\mathbb{K}_Y(x, x') &:= \mathrm{E}\left[\mathbf{Y}(x)\mathbf{Y}(x')^\top\right] - \mathrm{E}\left[\mathbf{Y}(x)\right] \mathrm{E}\left[\mathbf{Y}(x')^\top\right] \\
&= \mathbb{W}^{-1/2}\left(\mathrm{E}\left[\widetilde{\mathbf{Y}}(x)\widetilde{\mathbf{Y}}(x')^\top\right] - \mathrm{E}\left[\widetilde{\mathbf{Y}}(x)\right] \mathrm{E}\left[\widetilde{\mathbf{Y}}(x')^\top\right]\right) \mathbb{W}^{-1/2\top} \\
&= \mathbb{W}^{-1/2}\mathbb{K}_{\widetilde{Y}}(x, x')\mathbb{W}^{-1/2\top}
\end{aligned} \tag{30}$$

where one defines $\mathbb{K}_{\widetilde{Y}}(x, x') := \mathrm{E}\left[\widetilde{\mathbf{Y}}(x)\widetilde{\mathbf{Y}}(x')^\top\right] - \mathrm{E}\left[\widetilde{\mathbf{Y}}(x)\right] \mathrm{E}\left[\widetilde{\mathbf{Y}}(x')^\top\right]$.

Now assume that:

$$\mathrm{Cov}\left[\widetilde{Y}_i(x), \widetilde{Y}_j(x')\right] = 0 \quad \text{whenever } i \neq j, x, x' \in \chi\ .$$

This implies that $\mathbb{W}^{1/2}$ is proportional to a whitening transformation, so that all components of $\widetilde{Y}_1(x), \ldots, \widetilde{Y}_p(x)$ are uncorrelated.

Assume furthermore that:

$$\mathrm{Cov}\left[\widetilde{Y}_1(x), \widetilde{Y}_1(x')\right] = \ldots = \mathrm{Cov}\left[\widetilde{Y}_p(x), \widetilde{Y}_p(x')\right], \quad x, x' \in \chi.$$

Then one easily sees from Equation (29) that the scalar $k(x, x')$ satisfies

$$k(x, x') = \sum_{i=1}^{p} \mathrm{Cov}\left[\widetilde{Y}_i(x), \widetilde{Y}_i(x')\right] = p\,\mathrm{Cov}\left[\widetilde{Y}_j(x), \widetilde{Y}_j(x')\right], \quad j = 1, \ldots, p$$

Hence under these assumptions, denoting $\mathbb{I}_p$ the $p \times p$ identity matrix,

$$\mathbb{K}_{\widetilde{Y}}(x, x') = \frac{1}{p} k(x, x') \mathbb{I}_p.$$

As a consequence, from Equation (30),

$$\mathbb{K}_Y(x, x') := \mathrm{E}\left[\mathbf{Y}(x)\mathbf{Y}(x')^\top\right] - \mathrm{E}\left[\mathbf{Y}(x)\right]\mathrm{E}\left[\mathbf{Y}(x')^\top\right] = \frac{1}{p} k(x, x') \mathbb{W}^{-1} \tag{31}$$

$$\mathrm{Cov}\left[Y_i(x), Y_j(x')\right] = \frac{1}{p} k(x, x') (\mathbb{W}^{-1})_{ij} \tag{32}$$

Now from this, one can derive the local cross errors

$$\delta_{ij}(x, x') := \mathrm{E}\left[\left(M_i(x) - Y_i(x)\right)\left(M_j(x') - Y_j(x')\right)\right]$$

Let us denote by $\mathbb{Y}_{i.}$ the ith row vector of the matrix $\mathbb{Y}$. We get

$$\begin{aligned}
\delta_{ij}(x, x') &= \mathrm{E}\left[\left(\mathbb{Y}_{i.}\boldsymbol{\alpha}(x) - Y_i(x)\right)\left(\mathbb{Y}_{j.}\boldsymbol{\alpha}(x') - Y_j(x')\right)\right] \\
&= \boldsymbol{\alpha}(x)^\top \mathrm{E}\left[\mathbb{Y}_{i.}{}^\top \mathbb{Y}_{j.}\right] \boldsymbol{\alpha}(x') - \boldsymbol{\alpha}(x)^\top \mathrm{E}\left[\mathbb{Y}_{i.}{}^\top Y_j(x')\right] \\
&\quad - \mathrm{E}\left[Y_i(x)^\top \mathbb{Y}_{j.}\right] \boldsymbol{\alpha}(x') + \mathrm{E}\left[Y_i(x) Y_j(x')\right]
\end{aligned}$$

Now assume $\mathrm{E}\left[\mathbf{Y}(x)\right] = \boldsymbol{\mu}$ for all $x \in \chi$. Then from Equation (32),

$$\mathrm{E}\left[Y_i(x)^\top Y_j(x')\right] - \mu_i \mu_j = \frac{1}{p} k(x, x') (\mathbb{W}^{-1})_{ij}$$

which implies, using the matrix $\widetilde{\mathbb{K}}$ defined in Equations (19), page 19:

$$\mathrm{E}\left[\mathbb{Y}_{i.}{}^\top \mathbb{Y}_{j.}\right] - \mu_i \mu_j \mathbf{1}_n \mathbf{1}_n{}^\top = \frac{1}{p} (\mathbb{W}^{-1})_{ij} \widetilde{\mathbb{K}} \quad \text{and} \quad \mathrm{E}\left[\mathbb{Y}_{i.}{}^\top Y_j(x')\right] - \mu_i \mu_j \mathbf{1}_n = \frac{1}{p} (\mathbb{W}^{-1})_{ij} \widetilde{\mathbf{h}}(x').$$

Furthermore, assume that either weights sum to one or $\boldsymbol{\mu} = \mathbf{0}_p$, then terms in $\mu_i \mu_j$ vanish and one gets:

$$\delta_{ij}(x, x') = \frac{1}{p} (\mathbb{W}^{-1})_{ij} \left(\boldsymbol{\alpha}(x)^\top \widetilde{\mathbb{K}} \boldsymbol{\alpha}(x') - \boldsymbol{\alpha}(x)^\top \widetilde{\mathbf{h}}(x') - \widetilde{\mathbf{h}}^\top(x) \boldsymbol{\alpha}(x') + k(x, x')\right).$$

In particular from Proposition 5, using Remark 3 and Remark 4,

$$\delta_i(x^\star) = \frac{1}{p} (\mathbb{W}^{-1})_{ii} \Delta(x^\star). \tag{33}$$

From Equation (31), when $k(x,x) = \sigma^2$ for all x, one can write

$$\mathbb{K}_Y(x,x) = \frac{1}{p}\sigma^2 \mathbb{W}^{-1}.$$

Using $\sigma_i^2 := \mathrm{Var}\left[Y_i(x)\right]$, assumed to be constant over x,

$$\frac{1}{p}(\mathbb{W}^{-1})_{ii} = \frac{(\mathbb{K}_Y(x,x))_{ii}}{\sigma^2} = \frac{\sigma_i^2}{\sigma^2}.$$

Hence from Equation (33),

$$\delta_i(x^\star) = \frac{\sigma_i^2}{\sigma^2}\Delta(x^\star).$$

□

A.9 Proof of Remark 5 Constraints' impact

Proof: Nugget for positive weights. [Proof of Remark 5] The result is a very straightforward rewriting and interpretation of constraints (6) and (7). From $\mathbb{Y}\mathbb{A}\boldsymbol{\pi} = \mathbf{m}$ one derives $\mathbf{1}_p{}^\top \mathbf{m} = \mathbf{1}_p{}^\top \mathbb{Y}\mathbb{A}\boldsymbol{\pi} = \mathbf{1}_n{}^\top \mathbb{A}\boldsymbol{\pi} = \mathbf{1}_q{}^\top \boldsymbol{\pi} = 1$, hence the natural constraint on prescribed average membership degrees in $\mathbf{m}$, that must sum to one. □

A.10 Proof of Proposition 7 Nugget ensuring positive weights

Proof of Proposition 7.

$$\begin{aligned}
&\text{We have} && \mathbb{K}_{nug} := \mathbb{K} + \eta\mathbb{I}_n \ .\\
&\text{We denote} && \varepsilon := \frac{1}{\eta} \ .\\
&\text{therefore} && \mathbb{K}_{nug} = \mathbb{K} + \frac{1}{\varepsilon}\mathbb{I}_n \ ,\\
& && \quad = \frac{1}{\varepsilon}\left(\mathbb{I}_n + \varepsilon\mathbb{K}\right)\\
&\text{and} && \mathbb{K}_{nug}^{-1} = \varepsilon(\mathbb{I}_n - \varepsilon\mathbb{K} + \mathrm{o}\left(\varepsilon\right))\\
& && \quad = \varepsilon\mathbb{I}_n + \mathrm{o}\left(\varepsilon\right) \ .
\end{aligned}$$

Following notations of Proposition 2, we have:

$$\begin{aligned}
\delta &= \mathbf{1}_n{}^\top \mathbb{K}_{nug}^{-1}\mathbf{1}_n\\
\delta &= n\varepsilon + \mathrm{o}\left(\varepsilon\right)\\
\delta &= n\varepsilon(1 + \mathrm{o}\left(1\right))\\
\delta^{-1} &= \frac{1}{n\varepsilon}(1 + \mathrm{o}\left(1\right))\\
\boldsymbol{\lambda}^\top &= \frac{1}{n\varepsilon}\left(1 + \mathrm{o}\left(1\right)\right)\left(\mathbf{1}_q{}^\top - \mathbf{1}_n{}^\top\left(\varepsilon\mathbb{I}_n + \mathrm{o}\left(\varepsilon\right)\right)\mathbb{H}\right)\\
\boldsymbol{\lambda}^\top &= \frac{1}{n\varepsilon}\left(\mathbf{1}_q{}^\top + \mathrm{o}\left(1\right)\right)
\end{aligned}$$

And eventually:

$$\begin{aligned}
\mathbb{A} &= (\varepsilon \mathbb{I}_n + \mathrm{o}(\varepsilon)) \left(\mathbb{H} + \frac{1}{n\varepsilon} \mathbf{1}_n \mathbf{1}_q{}^\top + \frac{1}{n\varepsilon} \mathrm{o}(1) \right) \\
\mathbb{A} &= \frac{1}{n} \mathbf{1}_n \mathbf{1}_q{}^\top + \varepsilon \mathbb{H} - \mathrm{o}(1) \\
\lim_{\varepsilon \to 0} \mathbb{A} &= \frac{1}{n} \mathbf{1}_n \mathbf{1}_q{}^\top
\end{aligned}$$

Which is the expected result. □

B Symbols and Notation

locations
$\mathcal{X}$ set of locations (inputs/design points).
n, q number of observed locations, of prediction locations.
x any location. $x_1, \ldots, x_n$ are all observed locations.
$x^\star$ any prediction location. $x_1^\star, \ldots, x_q^\star$ are all prediction locations.
$X^\star$ a random variable over prediction locations.
$\boldsymbol{\pi} = (\pi_{x_1^\star}, \ldots, \pi_{x_q^\star})$ the $q \times 1$ distribution of $X^\star$ over prediction locations.
$\gamma = \boldsymbol{\pi}^\top \boldsymbol{\pi}$ an intermediate real value used in calculations.

output variables
p number of output variables.
$\mathbf{Y}(x)$ the $p \times 1$ vector of output variables at location x.
$\boldsymbol{\mu} = \mathrm{E}\left[\mathbf{Y}(x)\right]$ the $p \times 1$ mean of $\mathbf{Y}(x)$, when constant over x.
$\mathbb{Y} = [\mathbf{Y}(x_1), \ldots, \mathbf{Y}(x_n)]$ all the $p \times n$ values of observed output variables.
$\mathbb{Y}^\star = \left[\mathbf{Y}(x_1^\star), \ldots, \mathbf{Y}(x_q^\star)\right]$ all $p \times q$ unknown output variables at prediction locations.

prediction
$\mathbf{M}(x^\star)$ a $p \times 1$ predictor of $\mathbf{Y}(x)$
$\mathbb{M} = \left[\mathbf{M}(x_1^\star), \ldots, \mathbf{M}(x_q^\star)\right]$ the $p \times q$ matrix of all predictions.
$\boldsymbol{\alpha}(x^\star)$ the $n \times 1$ linear weights for the prediction in $x^\star$.
$\mathbb{A} = \left[\boldsymbol{\alpha}(x_1^\star), \ldots, \boldsymbol{\alpha}(x_q^\star)\right]$ the $n \times q$ matrix of weights for all predictions.
$\mathbf{m}$ a given constant $p \times 1$ vector of prescribed mean predicted values.
$\Delta(x^\star), \Delta_1(x^\star), \Delta_2(x^\star)$ losses to be minimized for finding $\mathbf{M}(x^\star)$.
$\boldsymbol{\lambda}$ a $q \times 1$ vector of Lagrange multipliers (relative to sum of weights)
$\boldsymbol{\lambda}'$ a $p \times 1$ vector of Lagrange multipliers (relative to predicted values)
$\mathbf{u} = \mathbb{Y}\mathbb{K}^{-1}\mathbf{1}_n$ an intermediate $p \times 1$ vector in calculations.
$\mathbf{Z}$ an additional $p \times 1$ factor for affine predictions.

covariances
$\mathbb{W}$ a given symmetric positive definite matrix for computing norms.
$\mathbf{h}(x^\star) = \mathrm{E}\left[\mathbb{Y}^\top \mathbb{W} \mathbf{Y}(x^\star)\right]$ a $n \times 1$ covariance vector.
$\mathbb{H} = \left[\mathbf{h}(x_1^\star), \ldots, \mathbf{h}(x_q^\star)\right]$ a $n \times q$ covariance matrix.
$\mathbb{K} = \mathrm{E}\left[\mathbb{Y}^\top \mathbb{W} \mathbb{Y}\right]$ a $n \times n$ covariance matrix.
$\widetilde{\mathbb{K}}, \widetilde{\mathbf{h}}(x^\star), \widetilde{\mathbb{H}}$ other covariances using centred expressions.
$\delta = {\mathbf{1}_n}^\top \mathbb{K}^{-1} \mathbf{1}_n$ an intermediate real value in calculations.
$\mathbf{P}$ additional $n \times 1$ covariance vector between $\mathbf{Z}$ and $\mathbf{Y}(x_i)$
$\mathbf{Q}$ additional $q \times 1$ covariance vector between $\mathbf{Z}$ and $\mathbf{Y}(x_j^\star)$

miscellaneous
$\mathbf{v}$ a generic vector for defining norm or checking psd characteristic.
$\mathbf{1}_n, \mathbf{1}_p, \mathbf{1}_q$ a vector of ones of size n, p, q respectively.
$\mathbf{0}_n, \mathbf{0}_p, \mathbf{0}_q$ a vector of zeros of size n, p, q respectively.

Contents

1 Introduction **1**

2 Joint Kriging Model **5**
2.1 Optimal Weights Without Constraints . . . 6
2.2 Optimal Weights Summing to One . . . 7
2.3 Optimal Weights With Constraint on Predictions . . . 9
2.4 Optimal Weights With Affine Extension . . . 11
2.5 Joint Kriging Mean and Variance . . . 13

3 Constrained Classification **15**
3.1 Prescribed Constraints . . . 15
3.2 Application of the Joint Kriging Model . . . 16
3.3 Positivity Requirement . . . 17

4 Filling Cross-Covariances **18**

5 Numerical Illustrations **21**
5.1 A Simplified Toy Example . . . 21
5.2 A Multi-Output Time Series Example . . . 23
5.3 A Constrained Classification Example . . . 28
5.4 Performance on multiple datasets . . . 34

6 Conclusion **36**

Bibliography **38**

A Proofs **41**
A.1 Proof of Proposition 1 Simple Joint Kriging weights . . . 41
A.2 Proof of Proposition 2 Ordinary Joint Kriging weights . . . 41
A.3 Proof of Remark 1 Covariance matrices . . . 42
A.4 Proof of Proposition 3 Joint Kriging weights under a predicted values constraint . . . 42
A.5 Proof of Remark 2 Covariance matrices with two constraints . . . 46
A.6 Proof of Proposition 5 Joint Kriging variance with arbitrary weights . . . 47
A.7 Proof of Remark 4 Covariance matrices in Joint Kriging mean and variance 47
A.8 Proof of Proposition 6 Variance sharing . . . 47
A.9 Proof of Remark 5 Constraints' impact . . . 49
A.10 Proof of Proposition 7 Nugget ensuring positive weights . . . 49

B Symbols and Notation **51**