```
> par(mfrow = c(1, 2))
> plot(fit1, type = 1)
> lines(sin(x), sin(x), col = "blue")
> plot(fit2, type = 1)
> lines(sin(x), sin(x), col = "blue")
```

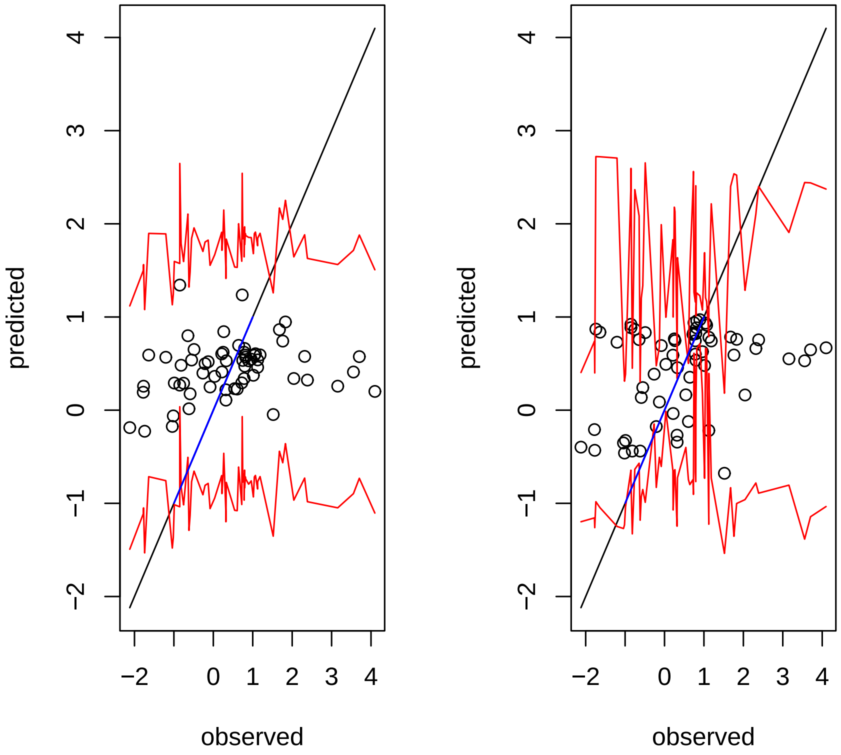

Figure 2: Diagnostic plots for Gaussian processes with constant nugget term (left) and diagonal nugget matrix (right). Open circles, cross-validated predictions; solid black lines, observed response; solid blue line, true (noiseless) response; solid red lines, confidence bands.

4 Sensitivity Analysis

4.1 Background

For a response $y = f(x)$, where x can be multidimensional, sensitivity analysis (SA) is used to (a) quantify the extent in which uncertainty in the response y can be attributed to uncertainty in the design parameters x, and (b) characterize how the response changes as one or more design parameters are varied. General SA methods can be found in Saltelli *et al.* (2000). We briefly describe SA using Gaussian process models, which is described in Schonlau and Welch (2006).

For independent marginal priors on the components of θ, the total variance of the GP predictor can be decomposed into variance contributions from main and higher order interaction effects, a technique known as Functional Analysis of Variance (FANOVA) decomposition. The percentage of the total functional variance accounted for by a particular effect provides a measure of the importance of that effect.

The main effect of parameter θ_k, defined as $E[z(\theta)|z_{\text{known}}, \theta_k]$, predicts output for a fixed value of θ_k, averaged over the remaining parameters according to a prior (or weight function) $\pi(\theta_{\text{-}k})$ on all components of θ except for the k^{th}. The two-way interaction effect for parameters θ_k and θ_l, defined as $E[z(\theta)|z_{\text{known}}, \theta_k, \theta_l]$, predicts output for jointly fixed values of θ_k and θ_l, averaged over the remaining parameters according to a prior $\pi(\theta_{\text{-}k,\text{-}l})$. Main effects plots and contour plots conveniently illustrate main effects and two-factor interactions.

In *mlegp*, we implement FANOVA decomposition and the plotting of main and two-way factor interactions using indepedent uniform priors on all components of θ. By default, the range of each component is taken to be the range of that component in the design matrix, but these ranges can be overwritten via the arguments 'lower'and 'upper'.

4.2 Examples

4.2.1 FANOVA decomposition

The function *FANOVADecomposition* is used to perform FANOVA decomposition on a single Gaussian processes, or on (a subset) of all Gaussian processes in a list. The function returns a table that reports the % contribution of each effect to the total functional variance of the Gaussian process predictor (or each Gaussian process predictor, in the case of a list). The code below demonstrates the use of the *FANOVADecomposition* function on a Gaussian process with two design parameters.

```
> x1 = kronecker(seq(0, 1, by = 0.25), rep(1, 5))
> x2 = rep(seq(0, 1, by = 0.25), 5)
> y = 4 * x1 - 2 * x2 + x1 * x2
> fit = mlegp(cbind(x1, x2), y, param.names = c("x1", "x2"))

> FANOVADecomposition(fit, verbose = F)

  param % contribution
1    x1     90.4937678
2    x2      9.1438403
3 x1:x2      0.3728887
```

4.2.2 Graphical plots for main and interaction effects

The function *plotMainEffects* is a generic function for plotting multiple main effects for a single Gaussian process; comparing a main effect across multiple Gaussian processes; and visualizing main effects of a single parameter on functional output. The first two uses of *plotMainEffects* are demonstrated below; an example of the latter can be found in Section (5).

First, we use *plotMainEffects* to plot the main effects for all input parameters on the Gaussian process created above. By default, all main effects are plotted, but a subset of effects can be specified by either name or number through the argument 'effects'. Setting 'FANOVA = TRUE' will calculate, for each main effect, the percentage contribution of that effect to the total functional variance of the GP predictor, and this will be reported in the legend. The function *plotInteractionEffect* is used to create a contour plot which visualizes interaction effects, and this is also demonstrated below. Output from the code can be seen in Figure (3).

```
> par(mfrow = c(1, 2))
> plotMainEffects(fit, graphStyle = 1, FANOVA = TRUE)
> plotInteractionEffect(fit, effects = c(1, 2))
```

It is also possible to use *plotMainEffects* to compare a main effect of a single parameter across multiple responses. We first create a Gaussian process list object that contains three GPs, each

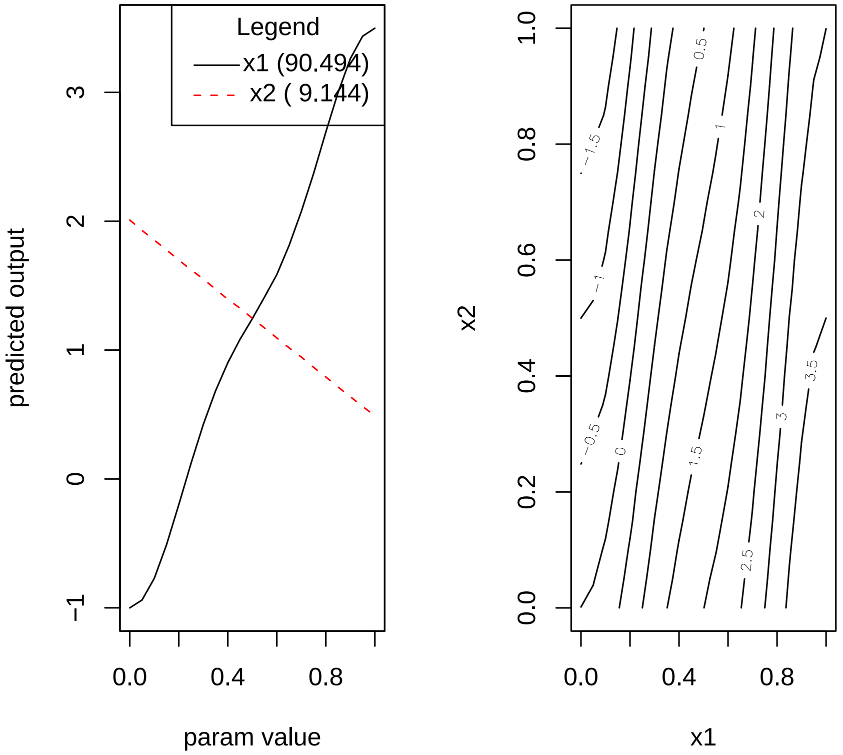

Figure 3: Main and Two-Way Interaction Effect Plots. For each main effect, the percent contribution of that effect to the total functional variance of the Gaussian process is also reported.

with the design parameter x. We then plot the main effect of x on the three GPs. This example also illustrates how the main effect can be referred to by name instead of by number. The main effects plot produced by the code is displayed in Figure (4).

```
> x = -5:5
> z1 = 10 - 5 * x + rnorm(length(x))
> z2 = 4 - 5 * x + rnorm(length(x))
> z3 = 7 * sin(x) + rnorm(length(x))
> fitMulti = mlegp(x, cbind(z1, z2, z3), param.names = "x")


> plotMainEffects(fitMulti, effects = "x", graphStyle = 1)
```

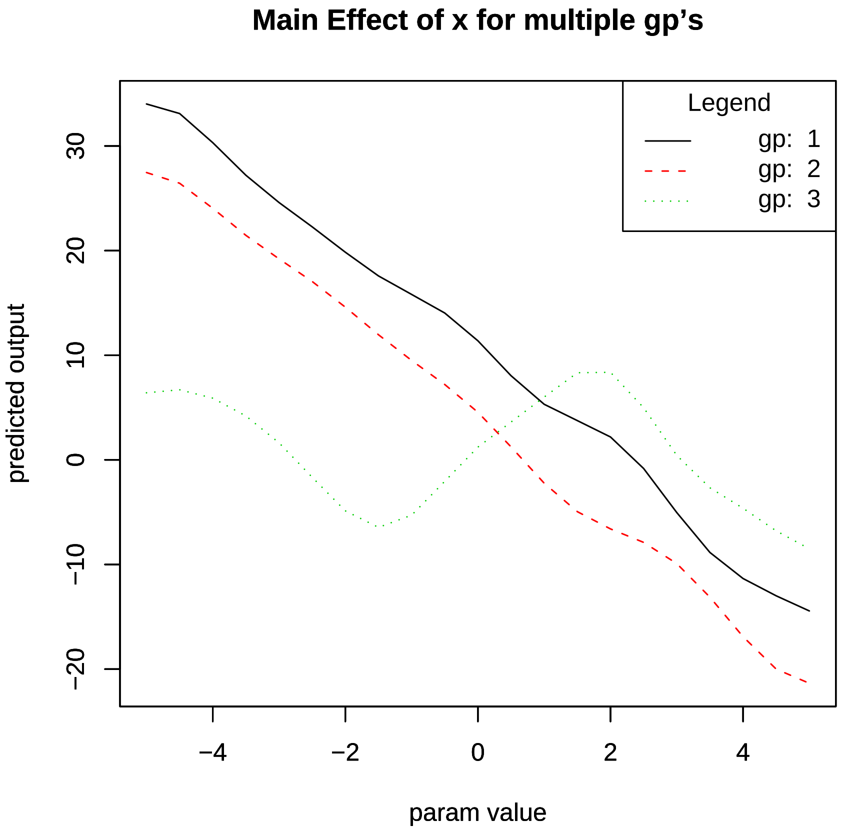

Figure 4: Main effects of the parameter x on a Gaussian process list that models three responses.

5 Multivariate Output and Dimension Reduction

5.1 Background

For multivariate or functional output, singular value decomposition can be used to reduce the dimensionality of the response (Heitmann *et al.*, 2006). Let $[z]_{i,j}$, $i = 1, \ldots, k$, $j = 1, \ldots, m$ be a matrix of m multivariate responses, where column j of the matrix contains the k-dimensional