mlegp: an R package for Gaussian process modeling and sensitivity analysis

Garrett Dancik

December 26, 2007

1 *mlegp*: an overview

Gaussian processes (GPs) are commonly used as surrogate statistical models for predicting output of computer experiments (Santner *et al.*, 2003). Generlly, GPs are both interpolators and smoothers of data and are effective predictors when the response surface of interest is a smooth function of the parameter space. The package *mlegp* finds *m*aximum *l*ikelihood *e*stimates of *G*aussian *p*rocesses for univariate and multi-dimensional responses, for Gaussian processes with product exponential correlation structures; constant or linear regression mean functions; no nugget term, constant nugget terms, or a nugget matrix that can be specified up to a multiplicative constant. The latter is an extension of previous Gaussian process models and provides some flexibility for using GPs to model heteroscedastic responses. Diagnostic plotting functions, and the sensitivity analysis tools of Functional Analysis of Variance (FANOVA) decomposition, and plotting of main and two-way factor interaction effects are implemented. Multi-dimensional output can be modelled by fitting independent GPs to each dimension of output, or to the most important principle component weights following singular value decomposition of the output. Plotting of main effects for functional output is also implemented. From within R, a complete list of functions and vignettes can be obtained by calling 'library(help = "mlegp")'.

2 Gaussian process modeling and diagnostics

2.1 Gaussian processes

Let $z_{\text{known}} = \left[z(\theta^{(1)}), \ldots, z(\theta^{(m)})\right]$ be a vector of *observed* responses, where $z(\theta^{(i)})$ is the response observed at the design point $\theta^{(i)}$, the parameter vector $\theta^{(i)} = \left[\theta_1^{(i)}, \ldots, \theta_p^{(i)}\right]$, and we are interested in predicting output $z(\theta^{(\text{new})})$ at the untried parameter setting $\theta^{(\text{new})}$. The correlation between any two responses (observed or unobserved) is assumed to have the (prodcut exponential) form

$$C(\beta)_{i,j} \equiv \text{cor}\left(z(\theta^{(i)}), z(\theta^{(j)})\right) = \exp\left\{\sum_{k=1}^{p}\left(-\beta_k\left(\theta_k^{(i)} - \theta_k^{(j)}\right)^2\right)\right\}. \quad (1)$$

The correlation matrix $\text{C}(\beta) = [\text{C}(\beta)]_{i,j}$, and depends on the correlation parameters $\beta = [\beta_1, \ldots, \beta_p]$

Let $\mu(\cdot)$ be the mean function for the unconditional mean of any observation, and the mean matrix of z_{known} be

$$M \equiv \left[\mu\left(\theta^{(1)}\right), \ldots, \mu\left(\theta^{(m)}\right)\right]. \quad (2)$$

The vector of observed responses, z_{known}, is distributed according to

$$z_{\text{known}} \sim MVN_m(M, \sigma_{GP}^2 C(\beta) + \sigma_e^2 I), \quad (3)$$

where I is a k x k identity matrix, σ^2_{GP} is the unconditional variance of an expected response and σ^2_e, the nugget term, is variance due to the stochasticity of the response (e.g., random noise). For convenience, denote the variance-covariance matrix of z_{known} as

$$V \equiv \sigma^2_{GP} C(\beta) + \sigma^2_e I \tag{4}$$

Also define $r_i = \text{cor}(z(\theta^{(new)}), z(\theta^{(i)}))$, following equation (1), and $r = [r_1, \ldots, r_m]'$. Under the GP assumption, the predictive distribution of $z(\theta^{(new)})$ is normal with mean

$$E[z(\theta^{(new)})|z_{\text{known}}] = \mu(\theta^{(new)}) + \sigma^2_{GP} r' V^{-1}(z_{\text{known}} - M) \tag{5}$$

and variance

$$Var[z(\theta^{(new)})|z_{\text{known}}] = \sigma^2_{\text{GP}} + \sigma^2_e - \sigma^4_{GP} r' V^{-1} r.$$

For more details, see Santner *et al.* (2003).

2.2 Maximum likelihood estimation

We first need some additional notation. Mean functions that are constant or linear in design parameters have the form $\mu(\theta) = x(\theta)F$, where $x(\theta)$ is a row vector of regression parameters, and F is a column vector of regression coefficients. Note that for a constant mean function, $x(\cdot) \equiv 1$ and F is a single value corresponding to the constant mean. The mean matrix M defined in equation (2) has the form $M = XF$, where the i^{th} row of X is equal to $x(\theta^{(i)})$.

Let us also rewrite the variance-covariance matrix V from equation (4) to be

$$V \equiv \sigma^2_{\text{GP}}(C(\beta) + \sigma^2_{e*} I) \equiv \sigma^2_{\text{GP}} W(\beta, \sigma^2_{e*}), \tag{6}$$

where $\sigma^2_{e*} = \sigma^2_e / \sigma^2_{\text{GP}}$, and the matrix W depends on the correlation parameters $\beta = [\beta_1, \ldots, \beta_p]$ and the scaled nugget term σ^2_{e*}.

When the matrix W is fully specified, maximum likelihood estimates of the mean regression parameters and σ^2_{GP} exist in closed form and are

$$\hat{F} = (X^T W^{-1} X)^{-1} X^T W^{-1} z_{\text{known}} \tag{7}$$

and

$$\hat{\sigma}^2_{\text{GP}} = \frac{1}{m}(z_{\text{known}} - \hat{M})^T W^{-1}(z_{\text{known}} - \hat{M}), \tag{8}$$

where $\hat{M} = X\hat{F}$.

The package *mlegp* uses numerical methods in conjunction with equations (7) and (8) to find maximum likelihood estimates of all GP parameters.

2.3 Diagnostics

The cross-validated prediction $z_{-i}(\theta^{(i)})$ is the predicted response obtained using equation (5) after removing all responses at design point $\theta^{(i)}$ from z_{known}. Note that it is possible for multiple $\theta^{(i)}$'s, for various i's, to be identical, in which case all corresponding observations are removed. The cross-validated residual for this observations is

$$\frac{z(\theta^{(i)}) - z_{-i}(\theta^{(i)})}{\text{se}(z_{-i}(\theta^{(i)})}, \tag{9}$$

where $\text{se}(z_{-i}(\theta^{(i)}))$ is the standard error of $z_{-i}(\theta^{(i)})$

2.4 What does *mlegp* do?

The package *mlegp* extends the Gaussian process model of (3) by allowing the user to replace the identity matrix I in equations (3) and (4) with a diagonal matrix N, thereby specifying the *nugget matrix* up to a multiplicative constant. This extension provides some flexibility for modeling heteroscedastic responses. The user also has the option of fitting a GP with a constant mean (i.e., $\mu(\theta) \equiv \mu_0$) or mean functions that are linear regression functions in all elements of θ (plus an intercept term). For multi-dimensional output, the user has the option of fitting independent GPs to each dimension (i.e., each type of observation), or to the most important principle component weights following singular value decomposition. The latter is ideal for data rich situations, such as functional output, and is explained further in Section (5). GP accuracy is analyzed through diagnostic plots of cross-validated predictions and cross-validated residuals, which were described in Section (2.3). Sensitivity analysis tools including FANOVA decomposition, and plotting of main and two-way factor interactions are described in Section (4).

3 Examples: Gaussian process fitting and diagnostics

3.1 A simple example

The function *mlegp* is used to fit Gaussian processes (GPs) to a vector or matrix of responses observed under the same set of design parameters. Data can be input from within R or read from a text file using the command *read.table* (type '?read.table' from within R for more information). The example below shows how to fit multiple Gaussian processes to multiple outputs $z1$ and $z2$ for the design matrix x. Diagnostic plots are obtained using the *plot* function, which graphs observed values vs. cross-validated predicted values for each GP. The plot obtained from the code below appears in Figure (1).

```
> x = -5:5
> z1 = 10 - 5 * x + rnorm(length(x))
> z2 = 7 * sin(x) + rnorm(length(x))
> fitMulti = mlegp(x, cbind(z1, z2))
> plot(fitMulti)
```

After the GPs are fit, simply typing the name of the object (e.g., $fitMulti$) will return basic summary information.

```
> fitMulti

num GPs: 2
Total observations (per GP): 11
Dimensions: 1
```

We can also access individual Gaussian processes by specifying the index. The code below, for examples, displays summary information for the first Gaussian process, including diagnostic statistics of cross-validated root mean squared error (CV RMSE) and cross-validated root max squared error (CV RMaxSE), where squared error corresponds to the squared difference between cross-validated predictions and observed values.

```
> fitMulti[[1]]

Total observations = 11
Dimensions = 1

mu = 10.49854
sig2:        191.4983
nugget:        0
```