```
> plot(exp.btgpllm, main = "treed GP LLM,", proj = c(1))
> plot(exp.btgpllm, main = "treed GP LLM,", proj = c(2))
```

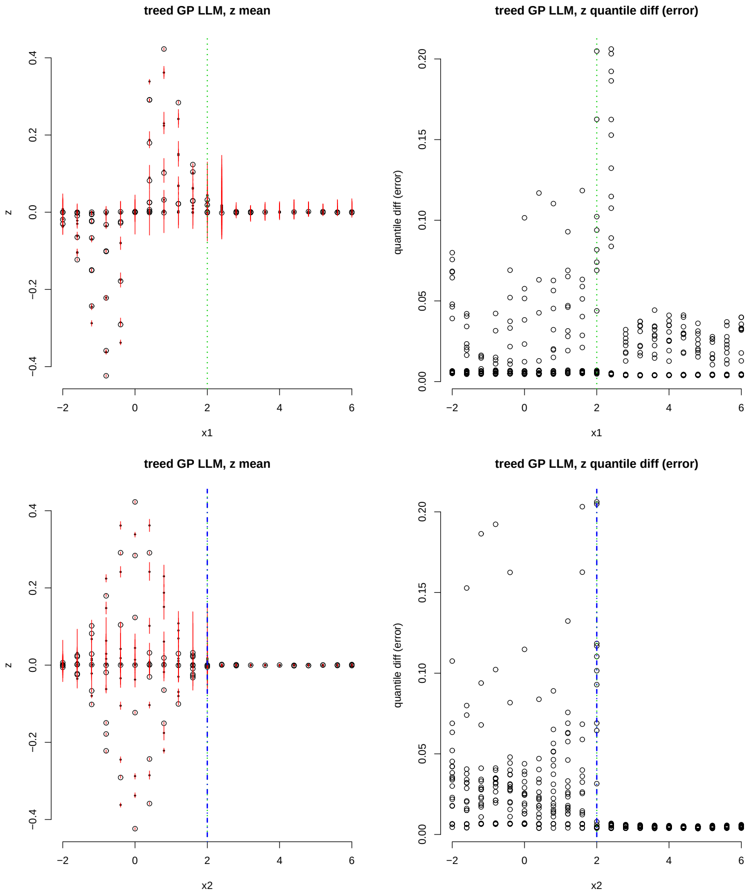


Figure 12: 1-d projections of the posterior predictive surface (*left*) and normed predictive intervals (*right*) of the 1-d tree GP LLM analysis of the synthetic exponential data. The *top* plots show projection onto the first input, and the *bottom* ones show the second.

Finally, viewing 1-d projections of `tgp`-class output is possible by supplying a scalar `proj` argument to the `plot.tgp`. Figure 12 shows the two projections for `exp.btgpllm`. In the *left* surface plots the open circles indicate the mean of posterior predictive distribution. Red lines show the 90% intervals, the norm of which are shown on the *right*.

3.4 Motorcycle Accident Data

The Motorcycle Accident Dataset [28] is a classic nonstationary data set used in recent literature [24] to demonstrate the success of nonstationary models. The data consists of measurements of the acceleration of the head of a motorcycle rider as a function of time in the first moments after an impact. In addition to being nonstationary, the data has input–dependent noise (heteroskedasticity) which makes it useful for illustrating how the treed GP model handles this nuance. There are at least two—perhaps three—three regions where the response exhibits different behavior both in terms of the correlation structure and noise level.

The data is included as part of the MASS library in R.

```
> library(MASS)
> X <- data.frame(times = mcycle[, 1])
> Z <- data.frame(accel = mcycle[, 2])
```

Figure 13 shows how a stationary GP is able to capture the nonlinearity in the response but fails to capture the input dependent noise and increased smoothness (perhaps linearity) in parts of the input space.

```
> moto.bgp <- bgp(X = X, Z = Z, m0r1 = TRUE, verb = 0)
```

Since the responses in this data have a wide range, it helps to translate and rescale them so that they have a mean of zero and a range of one. The m0r1 argument to b* functions automates this procedure. Progress indicators are suppressed.

```
> plot(moto.bgp, main = "GP,", layout = "surf")
```

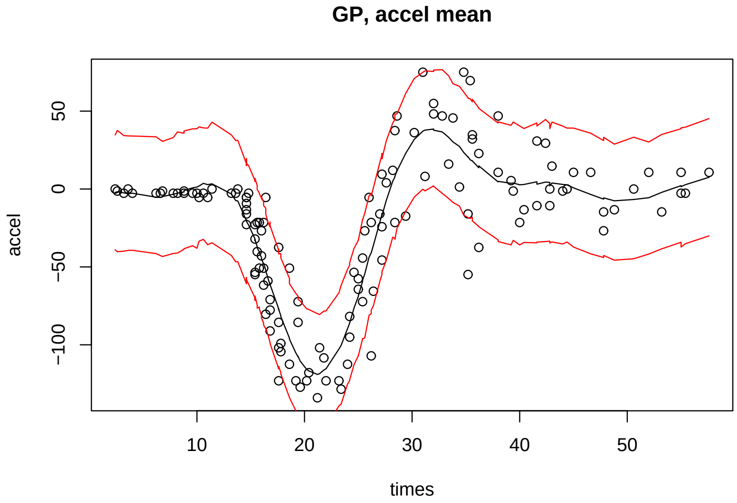


Figure 13: Posterior predictive distribution using bgp on the motorcycle accident data: mean and 90% credible interval

A Bayesian Linear CART model is able to capture the input dependent noise but fails to capture the waviness of the “whiplash”—center— segment of the response.

```
> moto.btlm <- btlm(X = X, Z = Z, m0r1 = TRUE, verb = 0)
```

Figure 14 shows the resulting piecewise linear predictive surface and MAP partition ($\hat{\mathcal{T}}$).

```
> plot(moto.btlm, main = "Bayesian CART,", layout = "surf")
```

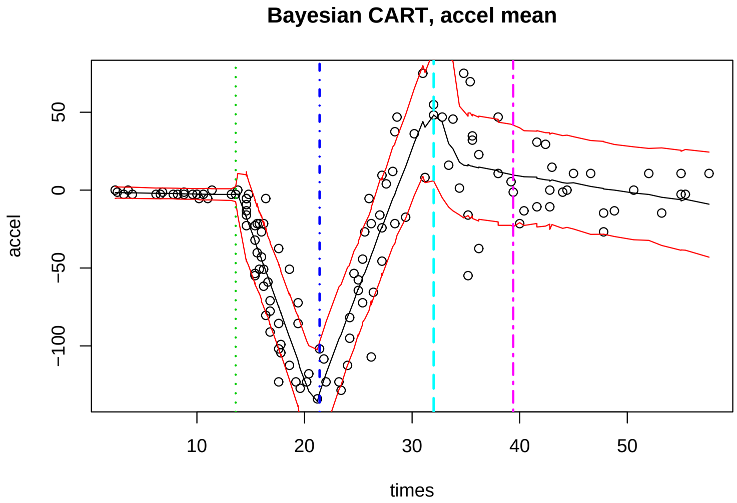


Figure 14: Posterior predictive distribution using `btlm` on the motorcycle accident data: mean and 90% credible interval

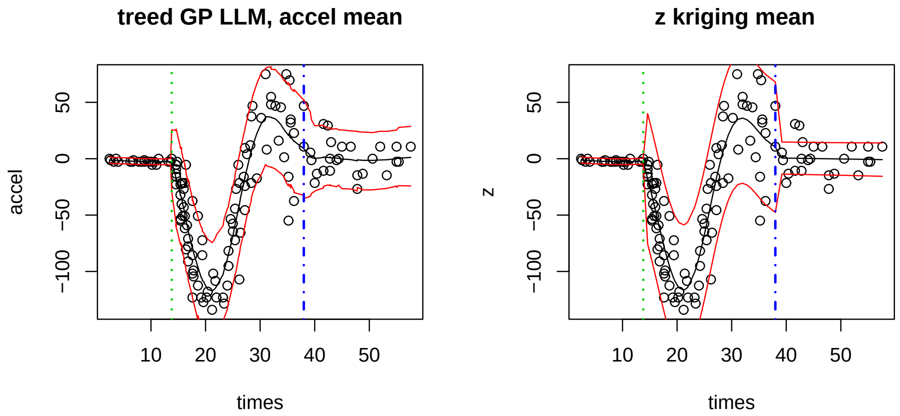
A treed GP model seems appropriate because it can model input dependent smoothness *and* noise. A treed GP LLM is probably most appropriate since the left-hand part of the input space is likely linear. One might further hypothesize that the right–hand region is also linear, perhaps with the same mean as the left–hand region, only with higher noise. The `b*` functions can force an i.i.d. hierarchical linear model by setting `bprior="b0"`.

```
> moto.btgpllm <- btgpllm(X = X, Z = Z, bprior = "b0",
+     m0r1 = TRUE, verb = 0)
> moto.btgpllm.p <- predict(moto.btgpllm)
```

The `predict.tgp` function obtains posterior predictive estimates from the MAP parameterization (a.k.a., *kriging*). The resulting posterior predictive surface is shown in the *top–left* of Figure 15. The *bottom–left* of the figure shows the norm (difference) in predictive quantiles, clearly illustrating the treed GP's ability to capture input-specific noise in the posterior predictive distribution. The *right–*hand side of the figure shows the MAP surfaces obtained from the output of the `predict.tgp` function.

The `tgp`–default `bprior="bflat"` implies an improper prior on the regression coefficients $\boldsymbol{\beta}$. It essentially forces $\mathbf{W} = \infty$, thus eliminating the need to specify priors on $\boldsymbol{\beta}_0$ and $\mathbf{W}^{-1}$ in (1). This was chosen as the default because it works well in many examples, and leads to a simpler overall model and a

```
> par(mfrow = c(1, 2))
> plot(moto.btgpllm, main = "treed GP LLM,", layout = "surf")
> plot(moto.btgpllm.p, center = "km", layout = "surf")
```



```
> par(mfrow = c(1, 2))
> plot(moto.btgpllm, main = "treed GP LLM,", layout = "as")
> plot(moto.btgpllm.p, as = "ks2", layout = "as")
```

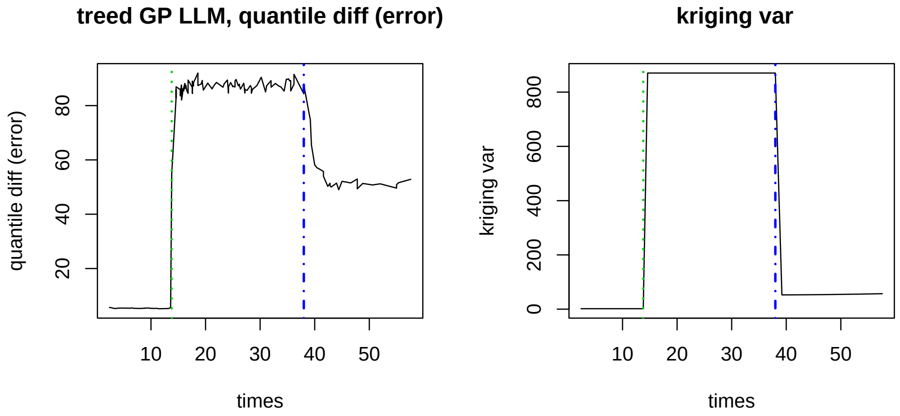


Figure 15: *Top*: Posterior predictive distribution using treed GP LLM on the motorcycle accident data. The *left*–hand panes how mean and 90% credible interval; *bottom*: Quantile-norm (90%-5%) showing input-dependent noise. The *right*–hand panes show similar *kriging* surfaces for the MAP parameterization.

faster implementation. However, the Motorcycle data is an exception. Moreover, when the response data is very noisy (i.e., low signal–to–noise ratio), `tgp` can be expected to partition heavily under the `bprior="bflat"` prior. In such cases, one of the other proper priors like the full hierarchical `bprior="b0"` or `bprior="bmzt"` might be preferred.

An anonymous reviewer pointed out a shortcoming of the treed GP model on this data. The sharp spike in predictive variance near the first regime shift suggests that the symmetric Gaussian noise model may be inappropriate. A log Gaussian process might offer an improvement, at least locally. Running the treed GP MCMC for longer will eventually result in the finding of a partition near time=17, just after the first regime change. The variance is still poorly modeled in this region. Since it is isolated by the tree it could potentially be fit with a different noise model.

3.5 Friedman data

This Friedman data set is the first one of a suite that was used to illustrate MARS (Multivariate Adaptive Regression Splines) [11]. There are 10 covariates in the data ($\mathbf{x} = \{x_1, x_2, \ldots, x_{10}\}$). The function that describes the responses (Z), observed with standard Normal noise, has mean

$$E(Z|\mathbf{x}) = \mu = 10\sin(\pi x_1 x_2) + 20(x_3 - 0.5)^2 + 10x_4 + 5x_5, \tag{18}$$

but depends only on $\{x_1, \ldots, x_5\}$, thus combining nonlinear, linear, and irrelevant effects. Comparisons are made on this data to results provided for several other models in recent literature. Chipman et al. [5] used this data to compare their treed LM algorithm to four other methods of varying parameterization: linear regression, greedy tree, MARS, and neural networks. The statistic they use for comparison is root mean-square error (RMSE)

$$\text{MSE} = \textstyle\sum_{i=1}^{n}(\mu_i - \hat{z}_i)^2/n \qquad\qquad \text{RMSE} = \sqrt{\text{MSE}}$$

where $\hat{z}_i$ is the model–predicted response for input $\mathbf{x}_i$. The $\mathbf{x}$'s are randomly distributed on the unit interval.

Input data, responses, and predictive locations of size $N = 200$ and $N' = 1000$, respectively, can be obtained by a function included in the `tgp` package.

```
> f <- friedman.1.data(200)
> ff <- friedman.1.data(1000)
> X <- f[, 1:10]
> Z <- f$Y
> XX <- ff[, 1:10]
```

This example compares Bayesian treed LMs with Bayesian GP LLM (not treed), following the RMSE experiments of Chipman et al. It helps to scale the responses so that they have a mean of zero and a range of one. First, fit the Bayesian treed LM, and obtain the RMSE.

```
> fr.btlm <- btlm(X = X, Z = Z, XX = XX, tree = c(0.95,
+     2), pred.n = FALSE, m0r1 = TRUE, verb = 0)
> fr.btlm.mse <- sqrt(mean((fr.btlm$ZZ.mean - ff$Ytrue)^2))
> fr.btlm.mse
```

```
[1] 1.939446
```