```
burn in:
**GROW** @depth 0: [0,0.424242], n=(43,57)
**GROW** @depth 1: [0,0.252525], n=(26,19)
**GROW** @depth 2: [0,0.131313], n=(14,13)
r=1000 d=[0] [0] [0] [0]; n=(11,19,17,53)
r=2000 d=[0] [0] [0] [0]; n=(11,17,19,53)

Sampling @ nn=99 pred locs:
r=1000 d=[0] [0] [0] [0]; mh=3 n=(15,14,18,53)
r=2000 d=[0] [0] [0] [0]; mh=4 n=(14,14,19,53)
r=3000 d=[0] [0] [0] [0]; mh=4 n=(12,16,19,53)
r=4000 d=[0] [0] [0] [0]; mh=4 n=(13,16,18,53)
r=5000 d=[0] [0] [0] [0]; mh=4 n=(13,15,19,53)
Grow: 0.8403%, Prune: 0%, Change: 36.3%, Swap: 84%
```

MCMC progress indicators show successful *grow* and *prune* operations as they happen, and region sizes n every 1,000 rounds. Specifying `verb=3`, or higher will show echo more successful tree operations, i.e., *change*, *swap*, and *rotate*.

Figure 7 shows the resulting posterior predictive surface (*top*) and trees (*bottom*). The MAP partition ($\hat{\mathcal{T}}$) is also drawn onto the surface plot (*top*) in the form of vertical lines. The treed LM captures the smoothness of the linear region just fine, but comes up short in the sinusoidal region—doing the best it can with piecewise linear models.

The ideal model for this data is the Bayesian treed GP because it can be both smooth and wiggly.

```
> sin.btgp <- btgp(X = X, Z = Z, XX = XX, verb = 0)
```

Figure 8 shows the resulting posterior predictive surface (*top*) and MAP $\hat{\mathcal{T}}$ with height=2.

Finally, speedups can be obtained if the GP is allowed to jump to the LLM [15], since half of the response surface is *very* smooth, or linear. This is not shown here since the results are very similar to those above, replacing `btgp` with `btgpllm`. Each of the models fit in this section is a special case of the treed GP LLM, so a model comparison is facilitated by fitting this more general model. The example in the next subsection offers such a comparison for 2-d data. A followup in Appendix B.1 shows how to use parameter traces to extract the posterior probability of linearity in regions of the input space.

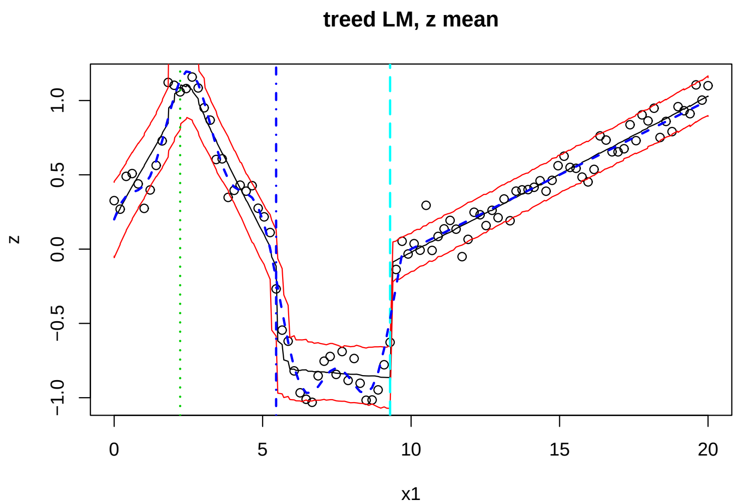
3.3 Synthetic 2-d Exponential Data

The next example involves a two-dimensional input space in $[-2, 6] \times [-2, 6]$. The true response is given by

$$z(\mathbf{x}) = x_1 \exp(-x_1^2 - x_2^2). \tag{17}$$

A small amount of Gaussian noise (with sd = 0.001) is added. Besides its dimensionality, a key difference between this data set and the last one is that

```
> plot(sin.btlm, main = "treed LM,", layout = "surf")
> lines(X, Ztrue, col = 4, lty = 2, lwd = 2)
```




```
> tgp.trees(sin.btlm)
```

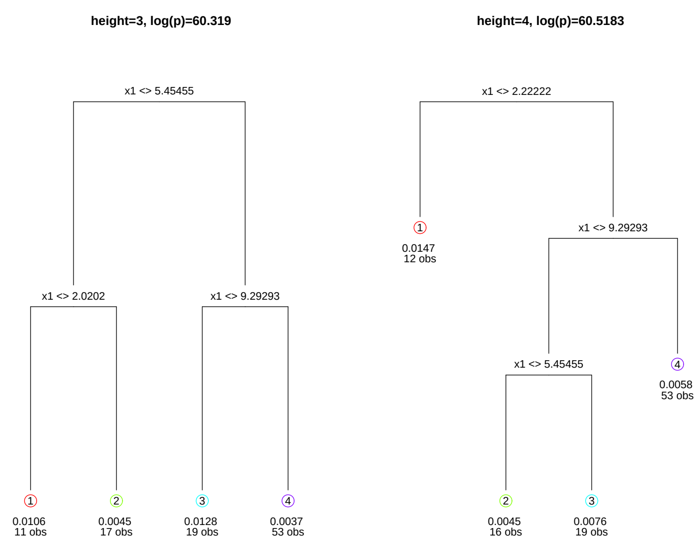


Figure 7: *Top:* Posterior predictive distribution using `btlm` on synthetic sinusoidal data: mean and 90% pointwise credible interval, and MAP partition ($\hat{\mathcal{T}}$). The true mean is overlayed with a dashed line. *Bottom:* MAP trees for each height encountered in the Markov chain showing $\hat{\sigma}^2$ and the number of observation n, at each leaf.

it is not defined using step functions; this smooth function does not have any

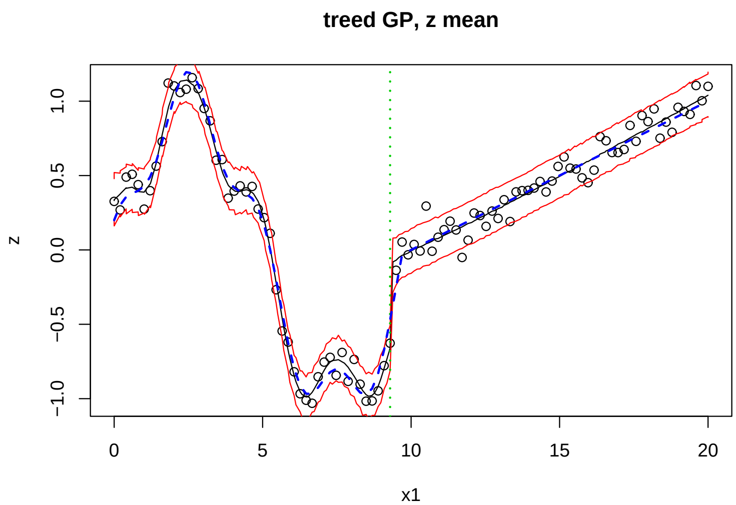


Figure 8: Posterior predictive distribution using `btgp` on synthetic sinusoidal data: mean and 90% pointwise credible interval, and MAP partition $(\hat{\mathcal{T}})$. The true mean is overlayed with a dashed line.

artificial breaks between regions. The `tgp` package provides a function for data subsampled from a grid of inputs and outputs described by (17) which concentrates inputs (`X`) more heavily in the first quadrant where the response is more interesting. Predictive locations (`XX`) are the remaining grid locations.

```
> exp2d.data <- exp2d.rand()
> X <- exp2d.data$X
> Z <- exp2d.data$Z
> XX <- exp2d.data$XX
```

The treed LM is clearly just as inappropriate for this data as it was for the sinusoidal data in the previous section. However, a stationary GP fits this data just fine. After all, the process is quite well behaved. In two dimensions one has a choice between the isotropic and separable correlation functions. Separable is the default in the `tgp` package. For illustrative purposes here, I shall use the isotropic power family.

```
> exp.bgp <- bgp(X = X, Z = Z, XX = XX, corr = "exp",
+     verb = 0)
```

Progress indicators are suppressed. Figure 9 shows the resulting posterior predictive surface under the GP in terms of means (*left*) and variances (*right*) in the default layout. The sampled locations (`X`) are shown as dots on the *right* image plot. Predictive locations (`XX`) are circles. Predictive uncertainty for the stationary GP model is highest where sampling is lowest, despite that the process is very uninteresting there.

A treed GP seems more appropriate for this data. It can separate out the large uninteresting part of the input space from the interesting part. The result

```
> plot(exp.bgp, main = "GP,")
```

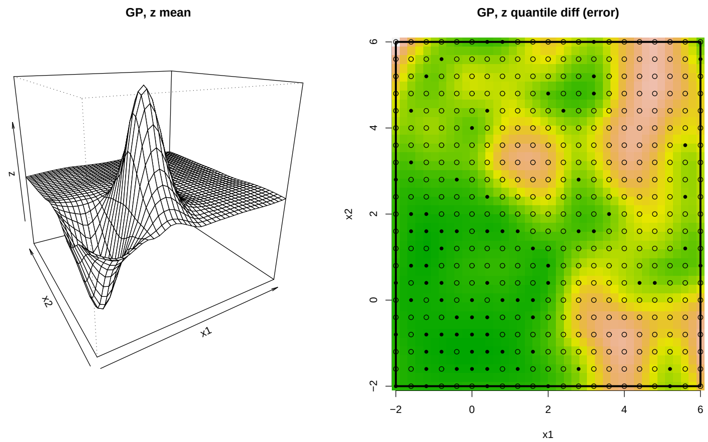


Figure 9: *Left:* posterior predictive mean using `bgp` on synthetic exponential data; *right* image plot of posterior predictive variance with data locations `X` (dots) and predictive locations `XX` (circles).

is speedier inference and region-specific estimates of predictive uncertainty.

```
> exp.btgp <- btgp(X = X, Z = Z, XX = XX, corr = "exp",
+     verb = 0)
```

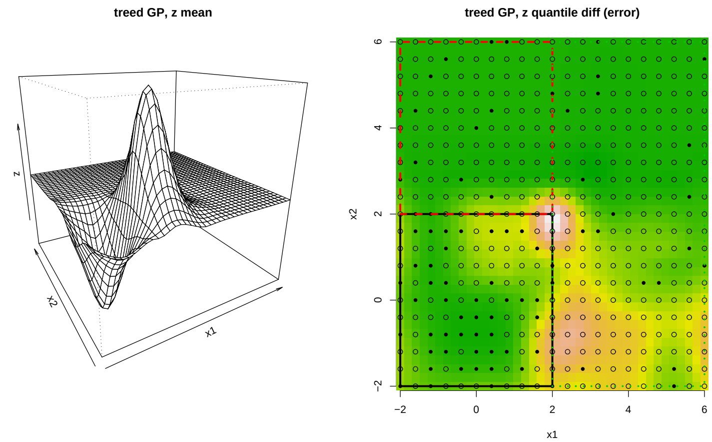
Figure 10 shows the resulting posterior predictive surface (*top*) and trees (*bottom*). Typical runs of the treed GP on this data find two, and if lucky three, partitions. As might be expected, jumping to the LLM for the uninteresting, zero-response, part of the input space can yield even further speedups [15]. Also, Chipman et al. recommend restarting the Markov chain a few times in order to better explore the marginal posterior for $\mathcal{T}$ [5]. This can be important for higher dimensional inputs requiring deeper trees. The `tgp` default is `R = 1`, i.e., one chain with no restarts. Here two chains—one restart—are obtained using `R = 2`.

```
> exp.btgpllm <- btgpllm(X = X, Z = Z, XX = XX, corr = "exp",
+     R = 2)

burn in:
**GROW** @depth 0: [0,0.45], n=(58,22)
**GROW** @depth 1: [1,0.45], n=(48,14)
r=1000 d=0.0218383 0(0.743906) 0(1.25691); n=(50,10,20)
r=2000 d=0.0222512 0.00696964 0(1.01303); n=(48,15,17)

Sampling @ nn=361 pred locs:
r=1000 d=0.0233355 0(0.0162573) 0(1.05867); mh=3 n=(50,10,20)
r=2000 d=0.0201525 0.0323228 0.735674; mh=3 n=(48,15,17)
```

```
> plot(exp.btgp, main = "treed GP,")
```




```
> tgp.trees(exp.btgp)
```

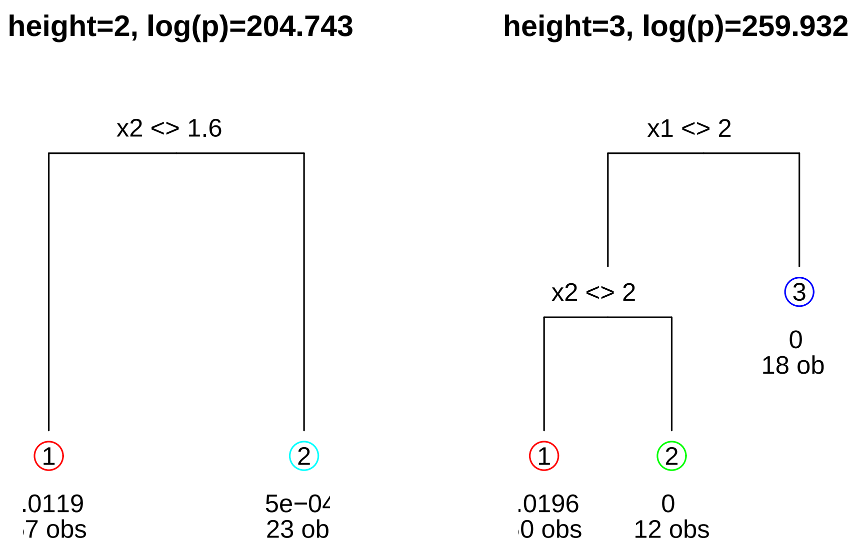


Figure 10: *Top-Left:* posterior predictive mean using `btgp` on synthetic exponential data; *top-right* image plot of posterior predictive variance with data locations `X` (dots) and predictive locations `XX` (circles). `Bottom:` MAP trees of each height encountered in the Markov chain with $\hat{\sigma}^2$ and the number of observations n at the leaves.

```
r=3000 d=0.0212731 0.00712512 0.214455; mh=3 n=(48,14,18)
r=4000 d=0.0235209 0(0.696318) 0(0.0374696); mh=3 n=(50,10,20)
r=5000 d=0.019617 0(0.840593) 0.0798781; mh=3 n=(50,12,18)
Grow: 0.5882%, Prune: 0%, Change: 36.02%, Swap: 44.1%
finished repetition 1 of 2
```

```
removed 3 leaves from the tree

burn in:
**GROW** @depth 0: [1,0.5], n=(60,20)
**GROW** @depth 1: [0,0.45], n=(45,12)
**PRUNE** @depth 1: [0,0.45]
r=1000 d=0.0230657 1.17737; mh=3 n=(57,23)
r=2000 d=0.0171015 1.02635; mh=3 n=(57,23)

Sampling @ nn=361 pred locs:
r=1000 d=0.0194236 1.04473; mh=3 n=(60,20)
**GROW** @depth 1: [0,0.45], n=(45,12)
r=2000 d=0.0212965 0.0533747 0(0.998999); mh=3 n=(50,10,20)
r=3000 d=0.0192226 0(0.0149189) 0(1.15603); mh=3 n=(50,12,18)
r=4000 d=0.0234385 0.0988804 1.21228; mh=3 n=(48,14,18)
r=5000 d=0.0216443 0(1.4039) 0.133204; mh=3 n=(50,12,18)
Grow: 0.7072%, Prune: 0.1488%, Change: 31.02%, Swap: 46.31%
finished repetition 2 of 2
removed 3 leaves from the tree
```

```
> plot(exp.btgpllm, main = "treed GP LLM,")
```

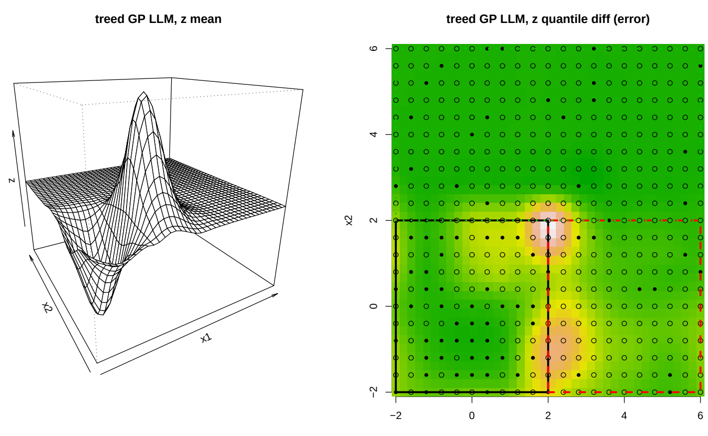


Figure 11: *Left:* posterior predictive mean using `btgpllm` on synthetic exponential data; *right* image plot of posterior predictive variance with data locations `X` (dots) and predictive locations `XX` (circles).

Progress indicators show where the LLM (`corr=0(`d`)`) or the GP is active. Figure 11 shows how similar the resulting posterior predictive surfaces are compared to the treed GP (without LLM). Appendix B.1 shows how parameter traces can be used to calculate the posterior probabilities of regional and location–specific linearity in this example.

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

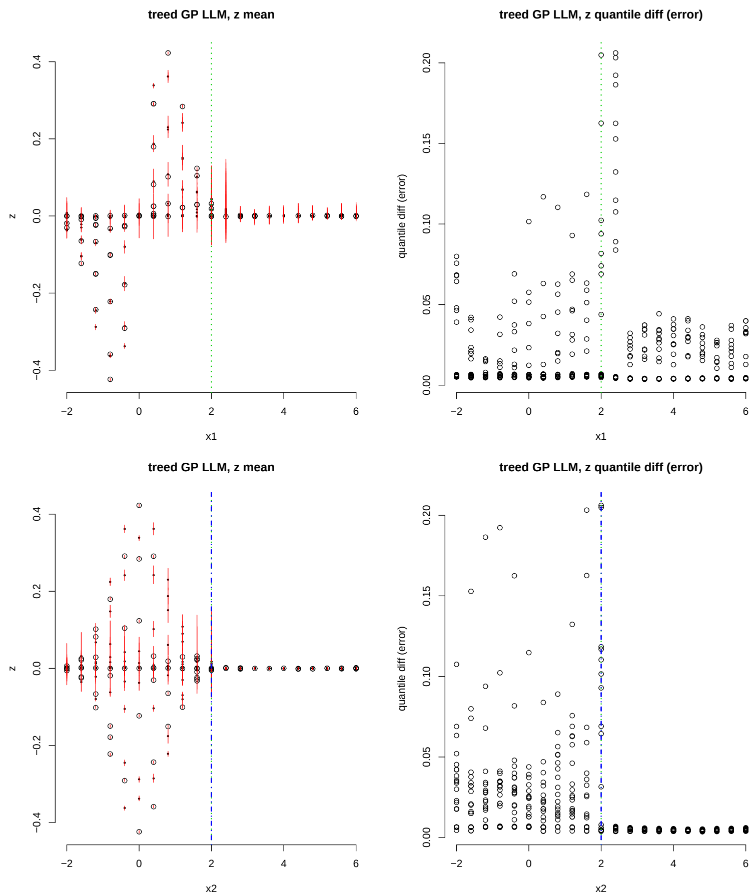


Figure 12: 1-d projections of the posterior predictive surface (*left*) and normed predictive intervals (*right*) of the 1-d tree GP LLM analysis of the synthetic exponential data. The *top* plots show projection onto the first input, and the *bottom* ones show the second.

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