Response to reviewer comments to the manuscript *2019-146: lg: An R package for Local Gaussian Approximations* submitted to *The R Journal* – Round 3

I am grateful for the opportunity to revise the paper for publication in *The R Journal*. Below are my point-by-point response to the comments made by the latest reviewer report.

> *Overview: The article describes how to use the lg package for local gaussian approximations. The proposed methodology provides insights into the correlation structure of data beyond the usual central correlation of Spearman. I think that this methodology is interesting, in particular in the context of financial analysis where tail dependence structures are of importance. I therefore think that the package will be used and the article will be consulted and cited.*

My response: Thank you for your kind remarks. The methodology has indeed been applied to financial analysis of various types.

> *I find the article clear and well written in general.*

My response: Thank you.

> *I would appreciate a few words on the "faithful" dataset which is used in the examples. I didn't know what it contains and had to find that out first somewhere else.*

My response: I have added a brief description of the data, as well as a reference to the description of the data in the R documentation.

> *I would suggest to include a few more words to explain the differences between the bootstrap procedures. In particular, I'm not sure about the difference between the "plain" and the "stationary" implementation.*

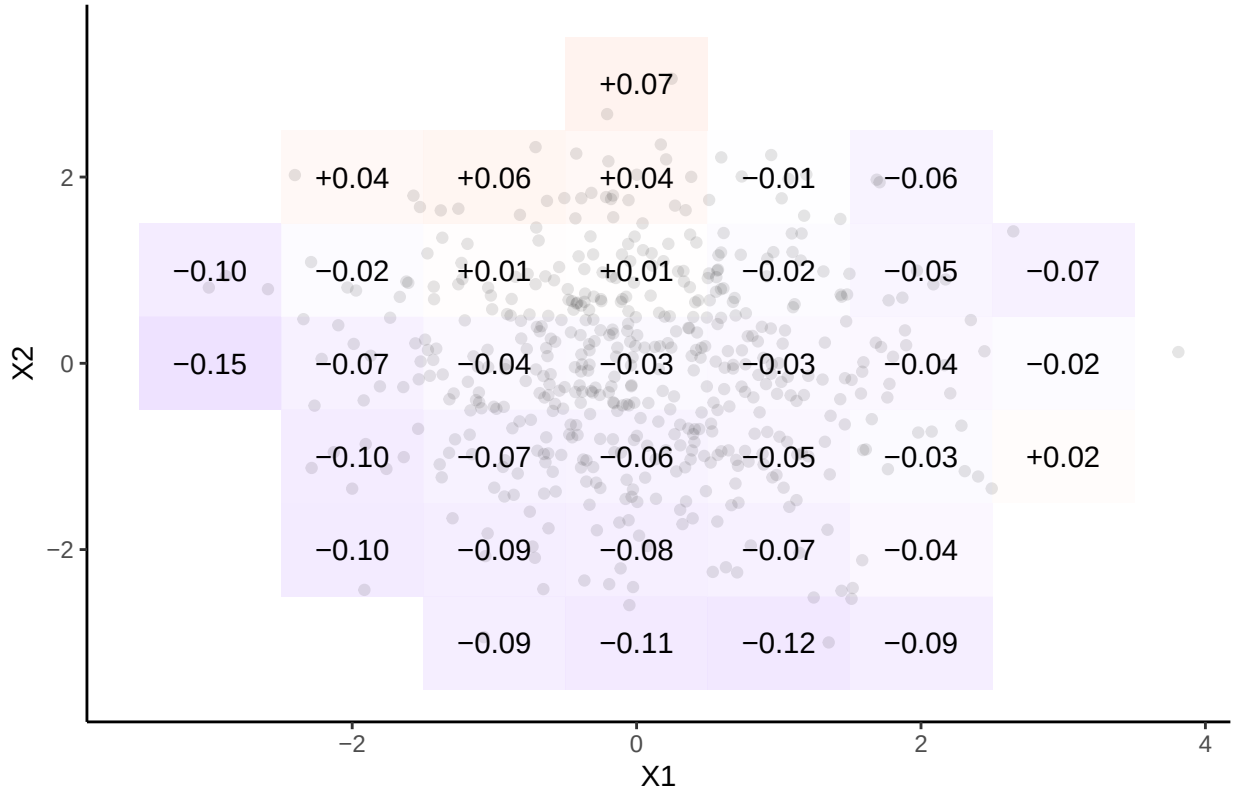
My response: A good suggestion. I have expanded the relevant paragraph so that the `plain`-option is explicitly mentioned and described as standard data resampling with replacement.

> *Some information about the minimum number of observations needed to reliably estimate the tails and boundary regions should be added. I modified the introductory example to x1 <- matrix(rnorm(1000), 500,2) which indicated significant dependence in some parts of the multivariate distribution, in particular at the thin ends. Of course, my data here are independent and this is a small sample issue which should be addressed in the paper as the method suggests dependence which is not there.*

My response: Thank you for raising this important point. I did the same thing, executed the following code, and obtained the figure below:

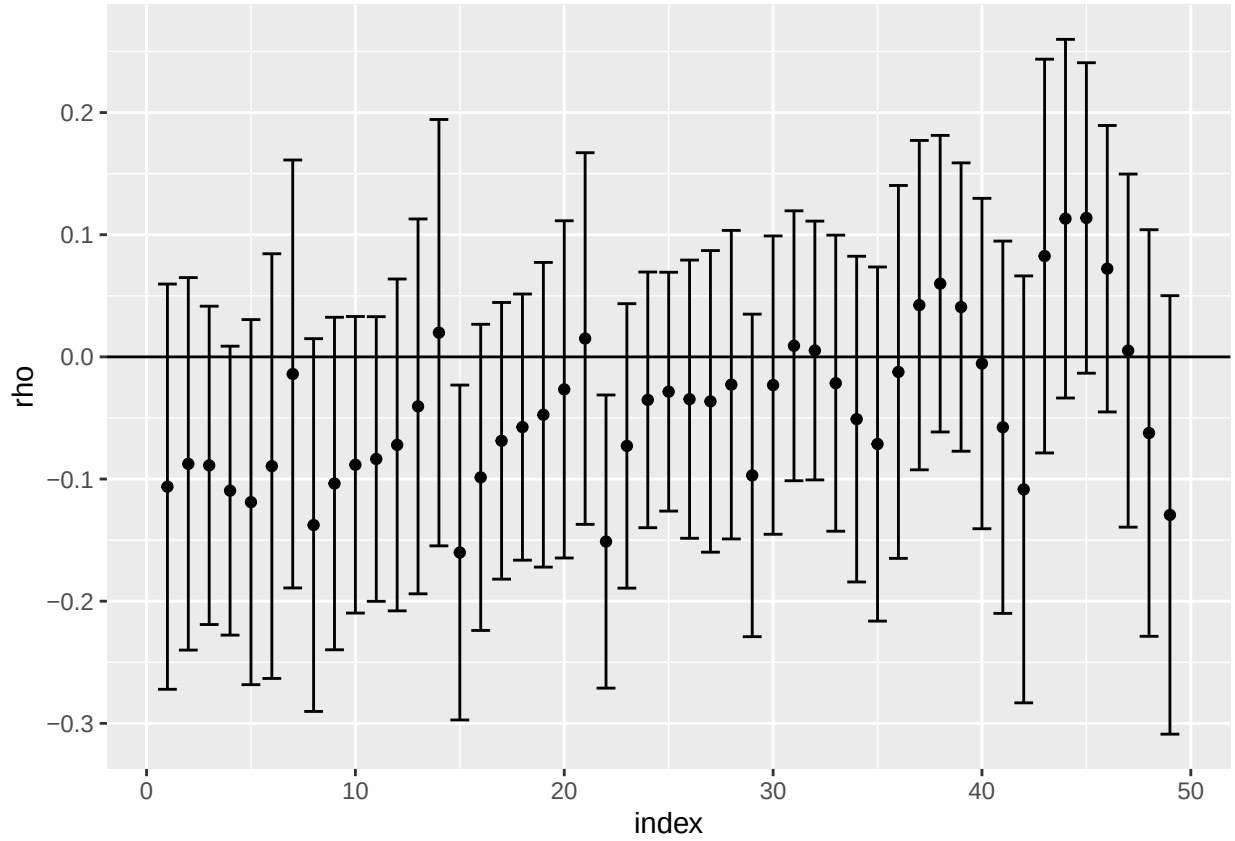
```
library(lg)
library(ggplot2)
library(dplyr)

set.seed(1)

# Simulated Gaussian data
grid1 <- expand.grid(seq(-3, 3, length.out = 7),
                     seq(-3, 3, length.out = 7))

x1 <- matrix(rnorm(1000), 500,2)
lg_object1 <- lg_main(x1,
                      est_method = "5par",
                      transform_to_marginal_normality = FALSE,
                      plugin_constant_joint = 4)
dlg_object1 <- dlg(lg_object1, grid = grid1)
```

```
corplot(dlg_object1, plot_thres = .01,
        xlab = "X1", ylab = "X2",
        main = "",
        plot_obs = TRUE,
        alpha_point = .1,
        plot_legend = FALSE,
        label_size = 4) +
    theme_classic() +
    theme(legend.position = "none")
```



If we have used the same random seed this figure should be identical to what the reviewer has obtained. Whether the observed local correlations represent *significant* dependence (in the statistical sense) can be investigated by looking at the estimated standard deviations, and lower/upper confidence limits which is reported on the 95% confidence level by default, available in the `dlg_object1` in the `$loc_cor_sd`, `$loc_cor_lower` and `$loc_cor_upper` slots respectively. By plotting the estimated correlations with their confidence intervals, as I have done below (the x-axis is just an index of the grid points), we see that zero (represented as a horizontal line) is included in almost all of the intervals.

```
data.frame(index = 1:length(dlg_object1$f_est),
           rho = dlg_object1$loc_cor,
           lower = dlg_object1$loc_cor_lower,
           upper = dlg_object1$loc_cor_upper) %>%
  ggplot(aes(x = index)) +
  geom_point(aes(y = rho)) +
  geom_errorbar(aes(ymin = lower, ymax = upper)) +
  geom_hline(yintercept = 0)
```



The global Pearson correlation coefficient for this data is -0.041. The corresponding independence test does not reject independence for this data. I get a p-value of 0.63 using 100 resamples.

The conclusion is that the dependence map in this case in fact does not indicate dependence that is statistically significant.

The comment made by the reviewer is relevant though, and represents a question that a reader of this paper might be interested in. Several of the main references that analyze the local Gaussian correlation provide asymptotic analyses of convergence rates under various common sets of assumptions, including the iid case as well as time series dependence. It turns out that the local Gaussian correlation estimates converge with the same speed, and hence require approximately the same sample sizes, as other well known non-parametric estimation methods that require a smoothing bandwidth, such as the kernel density estimator. The uncertainty is naturally greater in the tails of the distribution as the reviewer indicates.

I have added a comment that hopefully clears up this point in the discussion of this example. The latter part of the following sentence is new:

"In the first panel, we see that the estimated local correlation coincides with the global correlation, except for the estimation error which is comparable to the uncertainty observed in other non-parametric estimation methods such as the kernel density estimator (see, for instance, Otneim and Tjøstheim (2017) for a formal asymptotic analysis of relevant convergence rates)."

> *I tested the package on an openSuse 10 tumbleweed system (state 20210724) on a thinkpad i7-8650U with 16 GB RAM with R version 4.1.0.*

My response: I appreciate that the reviewer has taken the time to test the code file.

> *line 90:* `lg_object4 <- lg_main(x, est_method = "trivariate")` *cannot be created as x contains 2 columns only*

My response: Yes this is correct. At this point in the script, the data `x` is the bivariate `faithful`data set, for which we cannot fit a trivariate density estimate. That is the reason why the reviewer gets an error. I have commented out this particular line in the script and added an explanation. I have also pointed out in the paper that the function call will result in an error if the data has the wrong dimension.

line 132: `stock_data_d` *is not used, could be dropped*

My response: Correct, has been taken out.

line 239: the independence test (ind_test()) takes quite some time. Is there any possibility to speed that up by using parallel computing? It took 6.805 minutes on my computer, even longer with the return data. I appreciate the warning that it takes time given in the paper.

My response: Indeed, calculating critical values using bootstrapping takes quite a while, even for relatively modest sample sizes. This is noted in the paper, as the reviewer points out. It is possible to speed up the calculations using parallel computing, and this was actually implemented in an earlier version of the package. Unfortunately, this turned out to be a very unstable feature, because the back-ends that have been developed in various packages were too system dependent to work consistently on all platforms. It was therefore decided to take parallelization out of the package in order to keep it robust and future-proof.

It is, however, perfectly possible to wrap the functions in this package in the appropriate parallelized loops in order to speed up the process. As this is somewhat advanced, and would require more than just a brief comment, I have chosen not to explicitly point to this possibility in the paper.

line 269: R gave me a warning on the use of mutate().

My response: I confirm this behavior on my side, and it turns out that this warning appeared after the 4.0 update of R (see `https://github.com/bstewart/stm/issues/222` for a discussion about this in relation to another package), and relates to the use of formulas in the `fGarch`-package. Fixing it requires an update to that package, so unfortunately I can not get rid of it in this example. It has not changed the results.

I could of course calculate the GARCH-residuals locally on my computer and then load them directly into R, but that would make it harder for a reader to apply the method to his or her own data according to the original paper; Støve et. al. (2014).

I have instead added a comment in the script making the user aware of the issue, and clearly stating that the warning does not affects the result of the analysis.

I then obtained a p-value of 0.05 instead of the exact 0 given in the paper on page 13.

My response: Yes, this was a typo in the paper. I re-did the example, but also changed the number of bootstrap samples to 100 in order to make it consistent with the other examples in the paper. The p-value of the test in the example is now 0.01.

line 333: code interrupts with the following error message: Error in logspline::logspline(x[, i]) : Not enough unique values

My response: Yes, thank you for noticing. In order to make the examples as lightweight as possible I have demonstrated the methods on very small samples. The sample was a bit too small in this example, which caused problems in one of the dependencies; the `logspline`-package for estimating the marginal densities. I increased the length of the multivariate time series for this example, which has eliminated the problem.

In the last two points, the reviewer was not able to replicate the results in the paper exactly, which is probably due to a mistake in the original submission. The results should now be exactly replicable, also the parts that involve randomness due to bootstrapping.

I also got the following error (which is admittedly not related to the project but merely on trying to make the graphs look nicer).

```
# Initial example 1 --------
```

```
Error in pdf(file = "gaussian-example.pdf", height = 4, width = 5, family = "CM
Roman")
```

removing the command family = "CM Roman" solved the issue.

My response: Yes, my apologies. My intention was to make the figures exactly replicate as presented in the paper, but this requires a specific font to be installed on the user system. This is not important in practice, so I have commented out the relevant lines for figure export, and added a remark about this on line 10 in the script.

Finally, I have changed the order of lines 182 and 183 in the script in order to take care of a warning message regarding column names, resulting from a recent update in the `tibble`-package.