The Journal of Open Source Software

RBF: An R package to compute a robust backfitting estimator for additive models

Alejandra M. Martínez[1] **and Matias Salibian-Barrera**[2]

1 Departamento de Ciencias Básicas, Universidad Nacional de Luján, Argentina **2** Department of Statistics, University of British Columbia, Canada

DOI: 10.21105/joss.02992

Software

- Review
- Repository
- Archive

Editor: Mikkel Meyer Andersen

Reviewers:

- @mcavs
- @mmrabe

Submitted: 24 December 2020
Published: 07 April 2021

Summary

Although highly flexible, non-parametric regression models typically require large sample sizes to be estimated reliably, particularly when they include many explanatory variables. Additive models provide an alternative that is more flexible than linear models, not affected by the curse of dimensionality, and also allows the exploration of individual covariate effects. Standard algorithms to fit these models can be highly susceptible to the presence of a few atypical or outlying observations in the data. The RBF (Salibian-Barrera & Martínez, 2020) package for R implements the robust estimator for additive models of Boente et al. (2017), which can resist the damaging effect of outliers in the training set.

Statement of Need

The purpose of RBF is to provide a user-friendly implementation of the robust kernel-based estimation procedure for additive models proposed in Boente et al. (2017), which is resistant to the presence of potentially atypical or outlying observations in the training set.

Implementation Goals

RBF implements a user interface similar to that of the R package gam (Hastie, 2019), which computes the standard non-robust kernel-based fit for additive models using the backfitting algorithm. The RBF package also includes several modeling tools, including functions to produce diagnostic plots, obtain fitted values and compute predictions.

Background

Additive models offer a non-parametric generalization of linear models (Hastie & Tibshirani, 1990). They are flexible, interpretable, and avoid the *curse of dimensionality*, which means that as the number of explanatory variables increases, neighbourhoods rapidly become sparse, and many fewer training observations are available to estimate the regression function at any one point.

If Y denotes the response variable, and $\mathbf{X} = (X_1, \ldots, X_d)^\top$ a vector of explanatory variables, then an additive regression model postulates that

$$Y \ = \ \mu + \sum_{j=1}^{d} g_j(X_j) \, + \, \sigma \, \epsilon \, , \tag{1}$$

where the error ϵ is independent of **X** and its distribution is centered at zero, $\sigma > 0$ is an unknown scale parameter, the location parameter $\mu \in \mathbb{R}$, and $g_j : \mathbb{R} \to \mathbb{R}$ are smooth functions. Note that if for all $1 \leq j \leq d$ we have $g_j(X_j) = \beta_j X_j$ for some $\beta_j \in \mathbb{R}$, then Equation 1 reduces to a standard linear regression model.

The backfitting algorithm (Friedman & Stuetzle, 1981) can be used to fit the model in Equation 1 with kernel regression estimators for the smooth components g_j. It is based on the following observation: under Equation 1 the additive components satisfy $g_j(x) = E[Y - \mu - \sum_{\ell \neq j} g_\ell(X_\ell)|X_j = x]$. Thus, each g_j is iteratively computed by smoothing the partial residuals as functions of X_j.

It is well known that these estimators can be seriously affected by a relatively small proportion of atypical observations in the training set. Boente et al. (2017) proposed a robust version of backfitting, which is implemented in the RBF package. Intuitively, the idea is to use the backfitting algorithm with robust smoothers, such as kernel-based M-estimators (Boente & Fraiman, 1989). These robust estimators solve:

$$\min_{\mu, g_1, \ldots, g_d} E\left[\rho\left(\frac{Y - \mu - \sum_{j=1}^{d} g_j(X_j)}{\sigma}\right)\right],$$

where the minimization is computed over $\mu \in \mathbb{R}$, and functions g_j with $E[g_j(X_j)] = 0$ and $E[g_j^2(X_j)] < \infty$. The loss function $\rho : \mathbb{R} \to \mathbb{R}$ is even, non-decreasing and non-negative, and σ is the residual scale parameter. In practice, we replace σ by a preliminary robust estimator $\hat{\sigma}$ (for example, the Median Absolute Deviations (MAD) of the residuals from a local median fit) and the expected value by the average over the training set. Note that different choices of the loss function ρ yield fits with varying robustness properties. Typical choices for ρ are Tukey's bisquare family and Huber's loss (Maronna et al., 2018), and when $\rho(t) = t^2$, this approach reduces to the standard backfitting.

Simulation experiments reported in Boente et al. (2017) show that the robust backfitting algorithm provides more reliable estimators than the classical approach when the training set includes outliers in different proportions and settings. Those experiments also confirm that the robust backfitting estimators are very similar to the standard ones when the data do not contain atypical observations.

In the next section we illustrate the use of the robust backfitting algorithm as implemented in the RBF package by applying it to a real data set. We also compare the results with those obtained with the standard backfitting approach.

Illustration

The airquality data set contains 153 daily air quality measurements in the New York region between May and September, 1973 (Chambers et al., 1983). The interest is in modeling the mean ozone ("Ozone") concentration as a function of three potential explanatory variables: solar radiance in the frequency band 4000-7700 ("Solar.R"), wind speed ("Wind") and temperature ("Temp"). We focus on the 111 complete entries in the data set.

Since the plot in Figure 1 suggests that the relationship between ozone and the other variables is not linear, we propose using an additive regression model of the form

$$\text{Ozone} = \mu + g_1(\text{Solar.R}) + g_2(\text{Wind}) + g_3(\text{Temp}) + \varepsilon\,. \tag{2}$$

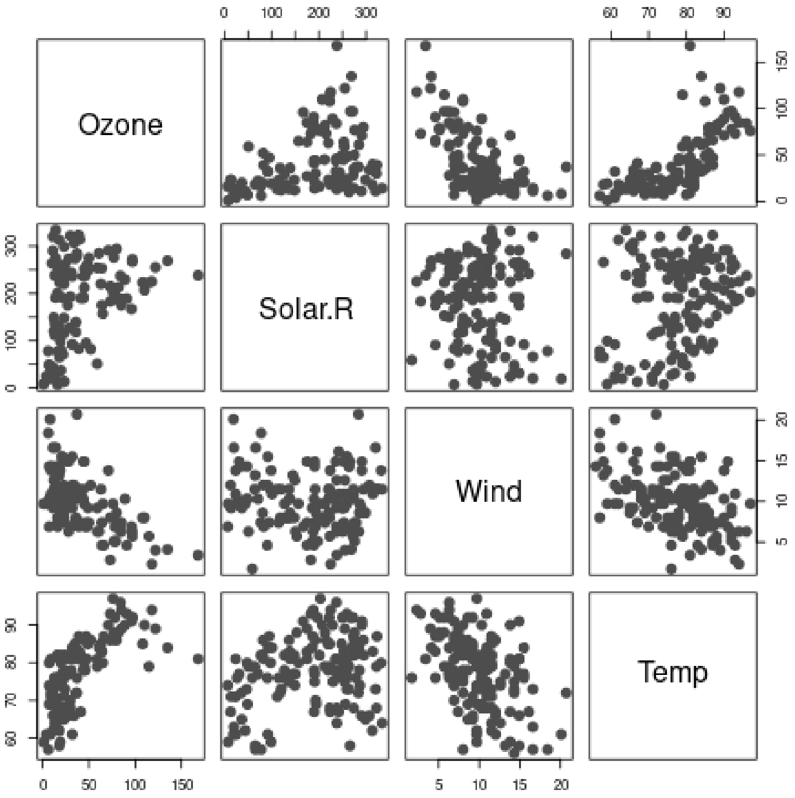

Figure 1: Scatter plot of the `airquality` data. The response variable is Ozone.

To fit the model above we use robust local linear kernel M-estimators with a Tukey's bisquare loss function. These choices are set using the arguments `degree = 1` and `type = 'Tukey '` in the call to the function `backf.rob`. The model is specified with the standard formula notation in R. The argument `windows` is a vector with the bandwidths to be used with each kernel smoother. To estimate optimal values we used a robust leave-one-out cross-validation approach (Boente et al., 2017) which resulted in the following bandwidths:

```
R> bandw <- c(136.7285, 10.67314, 4.764985)
```

The code below computes the corresponding robust backfitting estimator for Equation 2:

```
R> data(airquality)
R> library(RBF)
R> ccs <- complete.cases(airquality)
R> fit.full <- backf.rob(Ozone ~ Solar.R + Wind + Temp, windows=bandw,
                        degree=1, type='Tukey', subset = ccs, data=airquality)
```

A different kernel M-estimator can be used in the robust backfitting algorithm by setting `type = 'Huber'` in the call above. Unlike Tukey's re-descending score function, Huber's function is monotone, and numerical experiments show that the resulting estimator typically has larger bias. However, the corresponding objective function is convex and thus standard algorithms can be used to find the global minimum. Our algorithm takes advantage of this to construct a robust initial value to compute the more robust fit based on Tukey's loss function. For more details we refer the reader to Boente et al. (2017).

The argument `degree` is an integer indicating the desired degree of the local polynomial used in the kernel M-estimator. Its default value is `0` (which corresponds to a local constant fit). Other arguments for `backf.rob` include convergence controls (`epsilon`: the maximum allowed relative difference between consecutive estimates, and `max.it`: the maximum number of iterations), and tuning parameters for the chosen loss function (`k.h` for Huber's loss, and `k.t` for Tukey's). The default values for the latter two are those used to construct robust estimators for linear regression that are 95% efficient compared with the least squares ones.

To compare the robust and classical additive model estimators we use the R package `gam`. Optimal bandwidths were estimated using leave-one-out cross-validation as before.

```
R> library(gam)
R> aircomplete <- airquality[ccs, c('Ozone', 'Solar.R', 'Wind', 'Temp')]
R> fit.gam <- gam(Ozone ~ lo(Solar.R, span=.7) + lo(Wind, span=.7) +
                    lo(Temp, span=.5), data=aircomplete)
```

Figure 2 contains partial residuals plots and both sets of estimated functions: blue solid lines indicate the robust fit and magenta dashed ones the classical one.

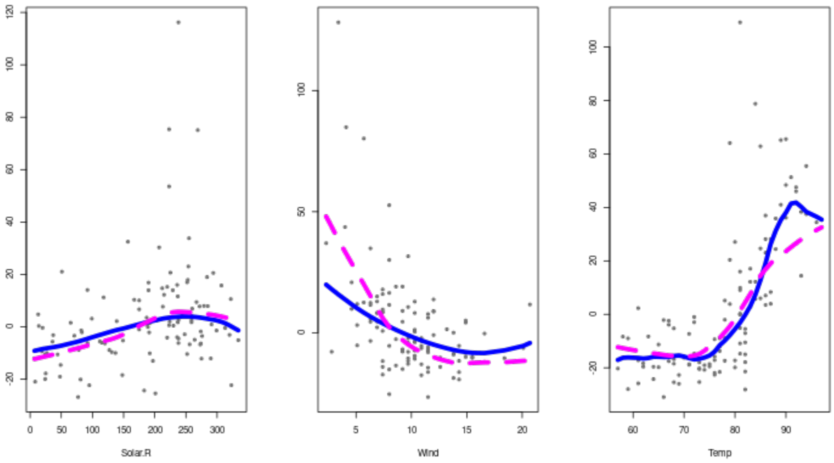

Figure 2: Partial residuals and fits for the `airquality` data. Robust and classical fits are shown with solid blue and dashed magenta lines, respectively.

The two fits differ mainly in the estimated effects of wind speed and temperature. The classical estimate for $g_3(\text{Temp})$ is consistently lower than the robust counterpart for Temp ≥ 85. For wind speed, the non-robust estimate $\hat{g}_2(\text{Wind})$ suggests a higher effect over Ozone concentrations for low wind speeds than the one given by the robust estimate, and the opposite difference for higher speeds.

Residuals from a robust fit can generally be used to detect the presence of atypical observations in the training data. Figure 3 displays a boxplot of these residuals. We note four possible outlying points (indicated with red circles).

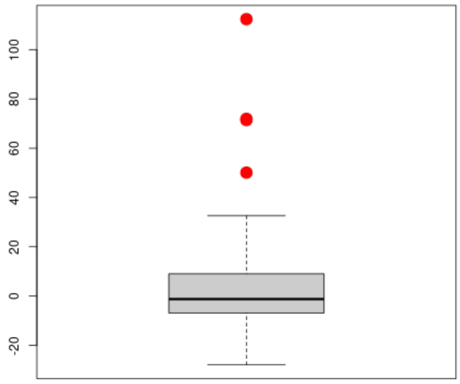

Figure 3: Boxplot of the residuals obtained using the robust fit. Potential outliers are highlighed with solid red circles.

To investigate whether the differences between the robust and non-robust estimators are due to the outliers, we recomputed the classical fit after removing them. Figure 4 shows the estimated curves obtained with the classical estimator using the "clean" data together with the robust ones (computed on the whole data set). Outliers are highlighted in red. Note that both fits are now very close. An intuitive interpretation is that the robust fit has automatically down-weighted potential outliers and produced estimates very similar to the classical ones applied to the "clean" observations.

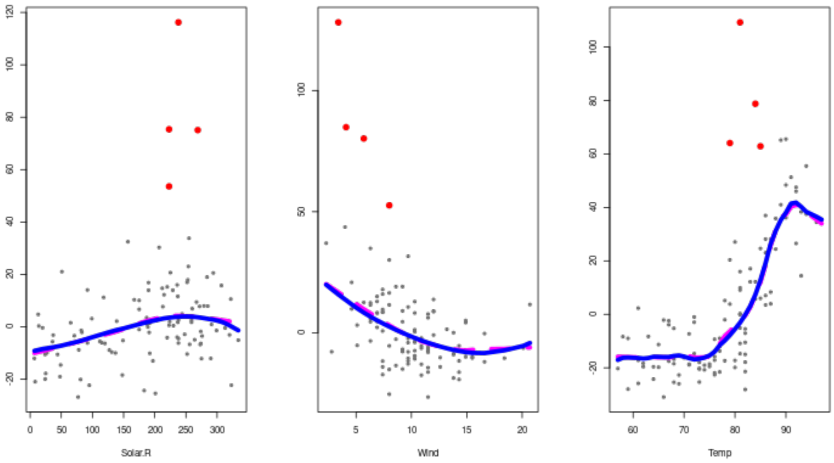

Figure 4: Plots of estimated curves and partial residuals. The solid blue lines indicate the robust fit computed on the whole data set, while the classical estimators computed on the "clean" data are shown with dashed magenta lines. Larger red circles indicate potential outliers.

Availability and Community Guidelines

The software is available at the Comprehensive R Archive Network CRAN and also at the GitHub repository https://github.com/msalibian/RBF. The GitHub repository also contains detailed scripts reproducing the data analysis above, and another example is included in the package vignette.

Contributions to this project can be submitted via pull requests on the GitHub repository. Similarly, GitHub issues are the preferred venue to report suggestions and problems with the current version of the software, and seek support.

Acknowledgements

This research was partially supported by: 20020170100022BA from the Universidad de Buenos Aires; project PICT 2018-00740 from ANPCYT; Internal Projects CD-CBLUJ 301/19 and CD-CBLUJ 204/19 from the Department of Basic Science of the Universidad Nacional de Luján (UNLu); the Researchers in Training Project RESREC-LUJ 224/19 (UNLu); and by the Natural Sciences and Engineering Research Council of Canada (Discovery Grant RGPIN-2016-04288).

References

Boente, G., & Fraiman, R. (1989). Robust nonparametric regression estimation. *Journal of Multivariate Analysis*, *29*(2), 180–198. https://doi.org/10.1016/0047-259X(89)90023-7

Boente, G., Martínez, A., & Salibian-Barrera, M. (2017). Robust estimators for additive models using backfitting. *Journal of Nonparametric Statistics*, *29*(4), 744–767. https://doi.org/10.1080/10485252.2017.1369077

Chambers, J. M., Cleveland, W. S., Kleiner, B., & Tukey, P. A. (1983). *Graphical methods for data analysis* (2nd ed.). Chapman & Hall. https://doi.org/10.1201/9781351072304

Friedman, J. H., & Stuetzle, W. (1981). Projection pursuit regression. *Journal of the American Statistical Association*, *76*(376), 817–823. https://doi.org/10.2307/2287576

Hastie, T. J. (2019). *gam: Generalized additive models*. https://CRAN.R-project.org/package=gam

Hastie, T. J., & Tibshirani, R. J. (Eds.). (1990). *Generalized additive models*. Chapman & Hall. https://doi.org/10.1201/9780203753781-6

Maronna, R. A., Martin, R. D., Yohai, V. J., & Salibian-Barrera, M. (2018). *Robust statistics: Theory and methods (with R)* (2nd ed.). John Wiley & Sons. https://doi.org/10.1002/9781119214656

Salibian-Barrera, M., & Martínez, A. (2020). *RBF: Robust backfitting*. https://CRAN.R-project.org/package=RBF