Journal of Statistical Software

MMMMMM YYYY, Volume VV, Issue II. doi: 10.18637/jss.v000.i00

Graphical Diagnostics for Heavy Tailed Data

John P. Nolan[*]
Department of Mathematics and Statistics
American University

Abstract

Graphical diagnostics based on a transformed empirical cdf are described for general heavy tailed data. This tool allows for a model free assessment of the tails of a univariate data set. The plot is linear in the mid-range and a log-log plot on the tails. In addition, comparisons between a data set and multiple models can be made. The approach gives an easy way to model power law decay on the tails; when combined with the empirical cdf in the middle of the data range, this gives a semi-parametric model for any univariate heavy tailed data set. With such a model, one can compute the cdf, quantiles, and simulate quickly. Some multivariate extensions are described using both a sequence of univariate plots and ordering based on distance from a center. The software is available in the open source R package **ecdfHT**.

Keywords: empirical cdf, heavy tails, R package.

1. Introduction

Graphical diagnostics are a useful tool for selecting a model and assessing how well a model fits data. If the tails of a distribution have heavy tails, then we expect large values in a data set. While these values may be a small proportion of the data, they can be a feature of the data that we wish to model. The term *outlier* is frequently used for recording errors or "fluke" extreme values. In contrast, here we assume that these extreme values are not mistakes in data recording or gathering, but a regular, although perhaps infrequent, part of the problem. We also assume that understanding these extreme values is an important part of the analysis, so removing them from the sample is not appropriate. Indeed, in some cases these extreme values may be the most important part of the data; examples are large losses in a financial market, large claims against an insurance company, maximum wind speed in a hurricane, minimum temperature, etc.

[*] Supported by contract W911NF-12-1-0385 from the Army Research Office.

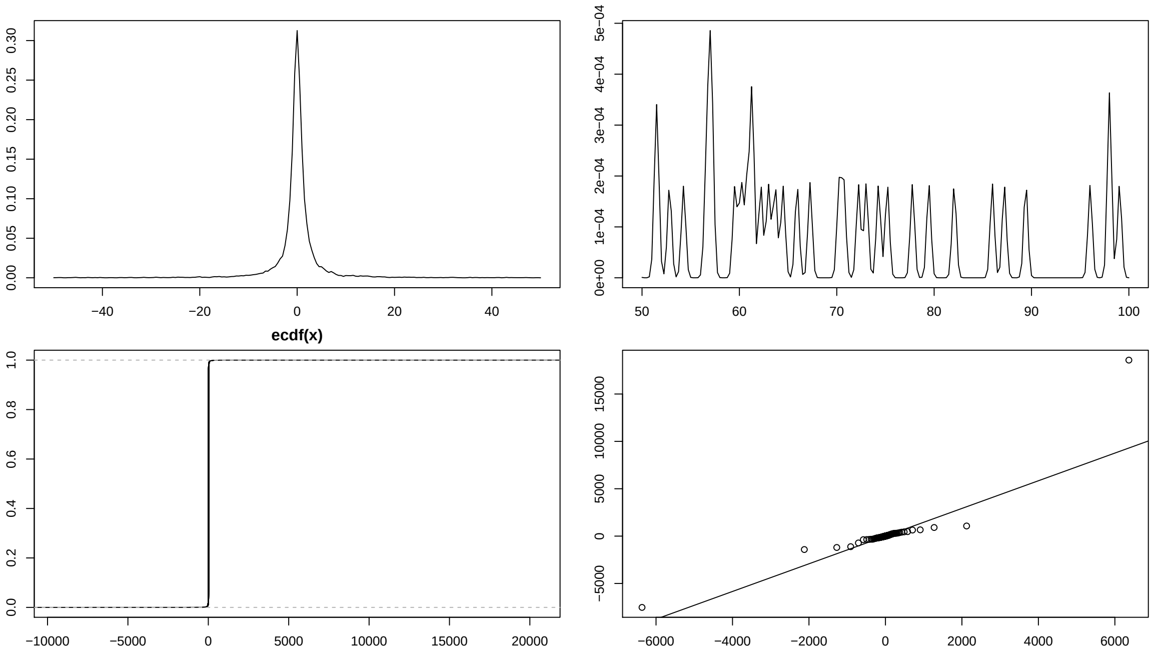

Figure 1: Standard diagnostics with a data set of n=10,000 simulated Cauchy random variates. The range for this data was from -7525.765 to 18578.677. The upper left plot shows a kernel density estimator using default Gaussian kernel and default bandwidth in the R function `density`, clipped to the region (-50,50). The upper right plot shows the same kernel density estimator on the interval (50,100). The lower left shows the ecdf. The lower right plot shows a qqplot of this simulated data vs. the exact equally spaced quantiles of a Cauchy distribution.

The commonly used graphical diagnostics - empirical density, empirical cumulative distribution function (ecdf), and Q-Q plots generally perform poorly with heavy tailed data - see Figure 1. Empirical density plots work well near the center, but behave erratically on the tails where there are few points that are widely scattered, resulting in isolated bumps in a kernel smoothed density. Comparing such an estimator to a model for the data is meaningless on the tails. Standard ecdf plots focus on the center of the distribution, with the tails monotonically approaching 0 on the left, and 1 on the right. For heavy tailed data, a standard ecdf plot looks like a step function, with most of the data visually compressed in a small interval with a steep rise. It is hard to assess tail behavior or compare to a model with such a plot. In the presence of heavy tails, Q-Q plots are visually dominated by the extreme values, with most of the data concentrated in a compressed central region. Furthermore, the inherent variability of the extreme values make for large deviations away from the diagonal line even when the data is being compared to the true model. Unlike the familiar light tailed case, in the heavy tailed case large sample sizes exacerbate these problems, because then there are likely to be more and more extreme values. These issues are compounded in the multivariate case, where heavy tails and directional dependence make it hard to explore multivariate data.

Below we propose a univariate non-parametric graphical diagnostics based on a non-linear transform of the empirical distribution function, where both axes are logarithmically scaled at the extremes. We call this an ecdfHT plot - an acronym for empirical cumulative distribution function (ecdf) for heavy tails (HT). It is assumed that a large data set is available; if not, conclusions about heavy tails are likely to be unreliable. This proposed plot has the advantage that power law behavior on the tails will appear as a straight line, regardless of the value of the tail exponents. The scaling is determined by quantiles of the data, and does not depend on any parametric model for the data. After discussing the univariate case, we describe some multivariate generalizations. The paper ends with some discussion about possible extensions and an outline of the main functions in the package.

2. A univariate plot

Given a data set $x_1, \ldots, x_n$, let $x_{(1)}, \ldots, x_{(n)}$ be the sorted values and let $p_i = (i - 1/2)/n$ be the empirical cdf at $x_{(i)}$. (We subtract the 1/2 in this definition as a simple way to avoid p values of 0 and 1 at the extremes. If there are repeats the package handles them correctly by tallying the number of repeats at a particular x and having a jump of the appropriate size at each repeat.) The standard ecdf plot shows the pairs $(x_{(i)}, p_i)$, $i = 1, \ldots, n$, see D'Agostino and Stephens (1986).

Pick three values $0 \leq q_1 \leq q_2 \leq q_3 \leq 1$ (called "scale quantiles") and define the corresponding data quantiles: $t_i = \widehat{F}^{-1}(q_i)$. Use these values to define the functions (see Figure 2):

$$\begin{aligned}
h_0(x) &= \begin{cases} -1 - \log(-x) & x < -1 \\ x & -1 \leq x \leq 1 \\ 1 + \log(x) & x > 1 \end{cases} \\
h(x) &= h(x|t_1, t_2, t_3) = \begin{cases} h_0\left(\frac{x-t_2}{t_2-t_1}\right) & x < t_2 \\ h_0\left(\frac{x-t_2}{t_3-t_2}\right) & x \geq t_2 \end{cases} \qquad (1) \\
g(p) &= g(p|q_1, q_2, q_3) = \begin{cases} q_1 + q_1 \log \frac{p}{q_1} & p < q_1 \\ p & q_1 \leq p \leq q_3 \\ q_3 - (1-q_3) \log \frac{1-p}{1-q_3} & p > q_3 \end{cases}
\end{aligned}$$

An ecdfHT plot graphs the transformed pairs $(h(x_{(i)}), g(p_i))$, $i = 1, \ldots, n$. Figure 3 shows a basic ecdfHT plot with simulated Cauchy data. The linear behavior on the tails is characteristic of a heavy tailed data set. Note that the coefficients are chosen so that both functions are continuous, monotonically increasing, linear in the middle and with logarithmic scales on the outer intervals. If the data is symmetric and $q_1 = 1 - q_3$, the two line segments in the middle of h have the same slope. The $h(\cdot)$ function pulls in extreme x values and the $g(\cdot)$ function spreads out values near the endpoints $p = 0$ and $p = 1$. This makes it possible to see both the behavior in the middle and the tail behavior on one plot. For symmetric data, it makes sense to use $q_2 = 1/2$ and $q_1 = 1 - q_3$, with for example $q_1 = 1/4$. For one sided data that has a finite left endpoint, it makes sense to use $0 = q_1 = q_2 < q_3$, say $q_3 = 3/4$. This is illustrated in Figure 6. Likewise, use $0 < q_1 < q_2 = q_3 = 1$ for data with a finite right endpoint. Note that in these one sided cases, the functions h and g are not defined on the truncated side, but in these regions the functions are not needed because no data values fall in these regions.

This plot has some features in common with a two-sided complementary cdf plot on a log-log scale, but orients things differently on the lower tail to show the ecdf and it gives a direct view of what happens in the mid-range.

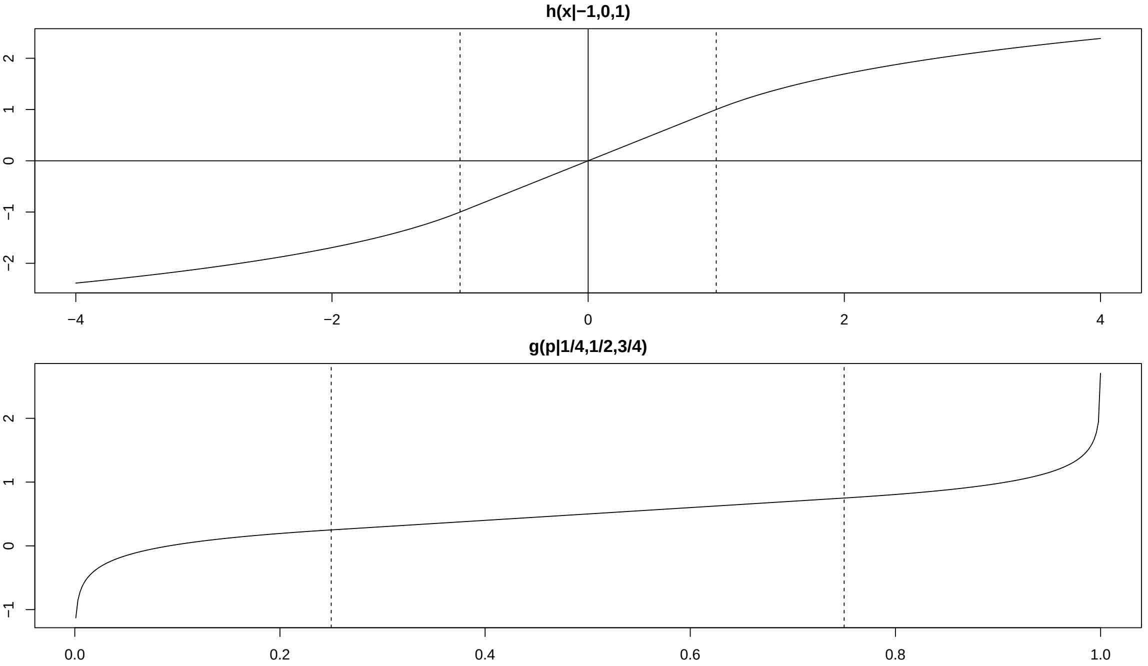

Figure 2: The functions $h(x|t_1 = -1, t_2 = 0, t_3 = 1)$ and $g(p|q_1 = 1/4, q_2 = 1/2, q_3 = 3/4)$. The dashed lines mark the cut points in the function definitions.

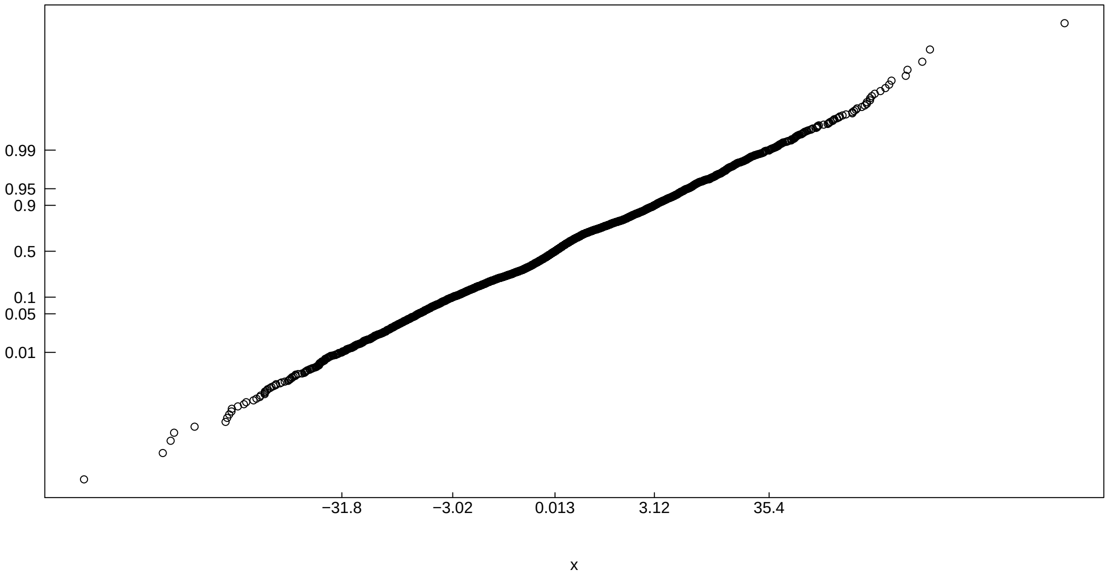

Figure 3: Basic ecdfHT plot with simulated Cauchy data, $n = 10,000$.

2.1. Additions to the plot

In addition to just displaying a data set, it is useful to be able to add more annotations and to compare it to one or more models to select an appropriate model for the data. Since neither axis is linear, it may be useful label specific points on the axes. Figure 4 shows more labels on the vertical axis, manually chosen labels on the horizontal axis, and grid lines at each of the tick marks. One can also compare the data to one or more models by adding the exact quantiles of the model. (This may require estimating parameters for the model from the data.) Figure 4 adds comparison curves of the exact cdf for a Cauchy law and a Gaussian fit. Pointwise confidence bounds can also be drawn for a model; these are shown as dotted lines of the same color around each model in the figure. These bounds are computed by using the standard confidence interval for a binomial parameter and transforming by the function $g(p)$. Note how the plot clearly shows non-Gaussian behavior over virtually the entire range: the data is leptokurtic with more values near the origin and much heavier tails than a Gaussian model, even though the estimated variance for the fitted Gaussian model is inflated by the extreme values in the data.

Figure 5 shows the diagnostic with a data set that consists of 10,000 values of recorded underwater acoustic signals in the Indian Ocean.[1] The top plot shows the original data - note the spikes in the data. These spikes significantly degrade the performance of sonar, and are apparently caused by snapping shrimp. The comparison curves show that a Gaussian model poorly describes the data, but a stable distribution with $\alpha = 1.74$ and $\beta \approx 0$ does a very good job of modeling the data, following the data all the way out to the extremes. (The stable parameters are estimated and cdf values are calculated using the program `stable.exe` available on the author's website.)

The next example uses Danish reinsurance data. These data were collected at Copenhagen Reinsurance and comprise 2167 losses over the period 1980 to 1990, see McNeil (1997).[2] Since this is a reinsurance data set, the claim sizes are truncated from below: only claims greater than 1 million DKK are in the data set. An ecdfHT plot of this one sided data set with scale quantiles $q_1 = q_2 = 0$, $q_3 = 0.75$ yields Figure 6, which shows roughly linear behavior on the upper tail. A simple estimate of the tail behavior will be given below.

These new plots can be useful for model selection as shown in Figure 7. A data set was simulated from a t-distribution with 4 d.f. and the goal is to choose the correct d.f. for a t-distribution to fit the data. In this simulation, visual inspection of the plots leads to correct identification of the d.f. While this may not always work, the plots will indicate where the data differs from a model and it is likely that when model selection is not obvious, that it will generally be hard to distinguish between models.

What model is appropriate depends on what you are using the model for. In general, there may be no single parametric model that closely fits the data over the whole range. The next section describes a semi-parametric model that may be useful for modeling when a parametric model doesn't adequately describe the data.

2.2. A semi-parametric model for heavy tailed data

The way the basic plot was constructed suggests a semi-parametric model for a heavy tailed

[1] This data was kindly provided by Prof. Mandar Chitre at the National University of Singapore.

[2] This data set is available from Alex McNeil's website: `http://www.macs.hw.ac.uk/~mcneil/data.html`

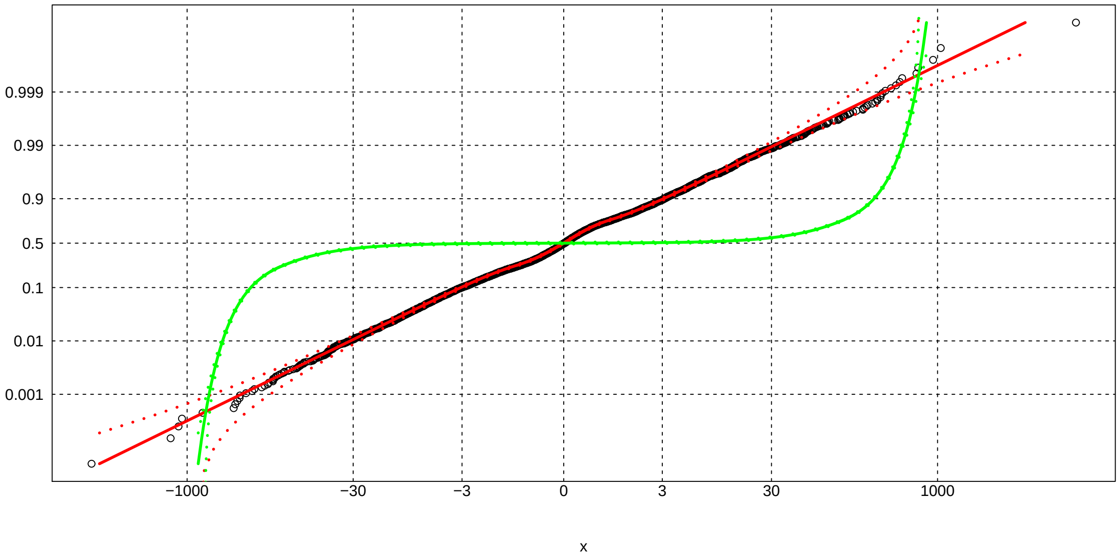

Figure 4: Univariate plot with more annotations and comparison of two models to the simulated Cauchy data, with $n = 10,000$. The solid red curve corresponds to a Cauchy cdf, the green curve to a Gaussian cdf. The dashed lines show pointwise 95% confidence bounds for the two models.

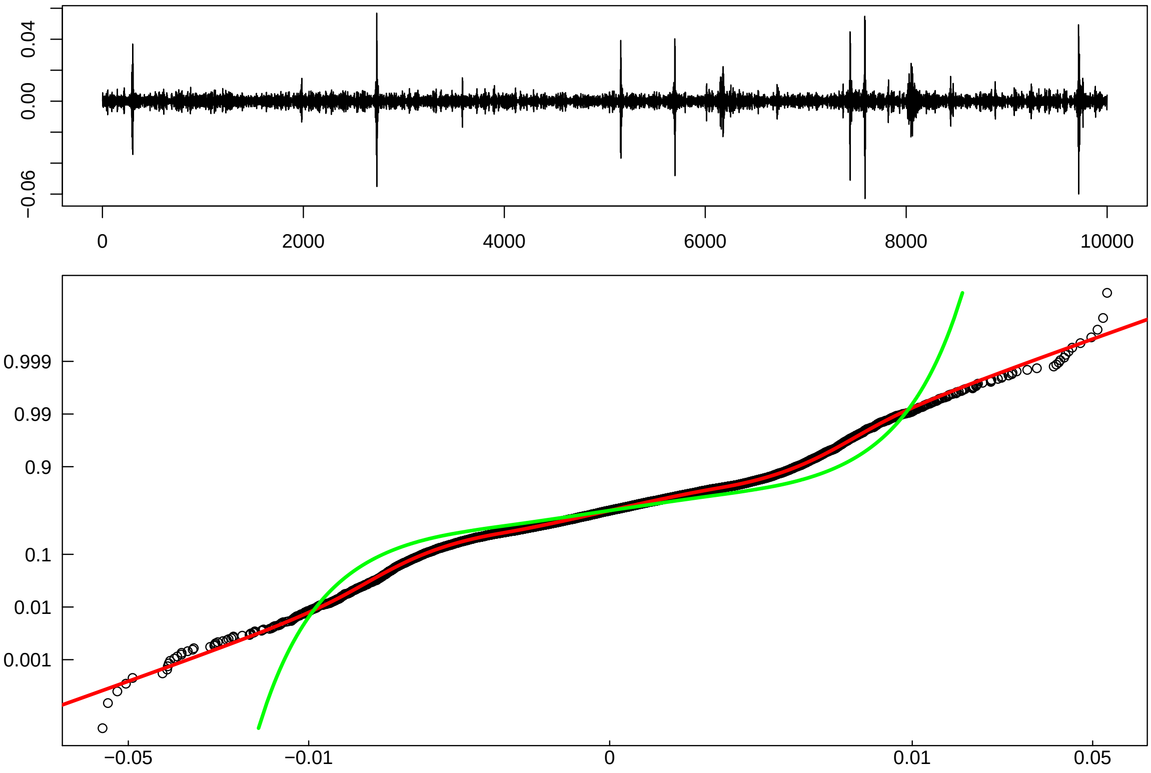

Figure 5: Underwater acoustic data from south Asia. The top plot shows the raw data, the bottom plot shows the ecdfHT plot with scale quantiles=(.25,.50,.75) and comparison to two models. The red curve is a stable distribution with $\alpha = 1.7424$, $\beta = -0.0160$, $\gamma = 0.0018821$, $\delta = 6.45375 \times 10^{-6}$ in the 0-parameterization; the green curve is $\mathrm{N}(\mu = 3.3838 \times 10^{-7}, \sigma^2 = 1.6174 \times 10^{-5})$.

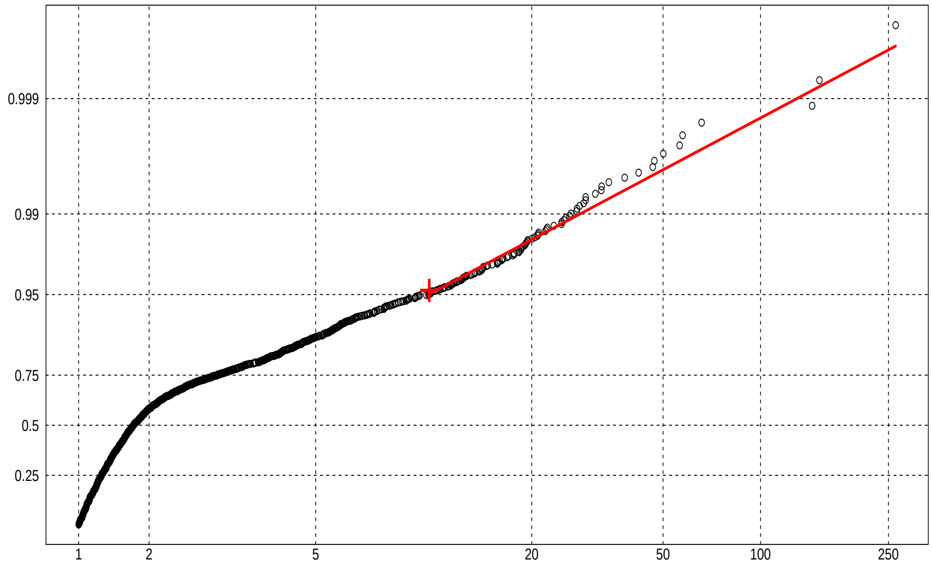

Figure 6: Danish reinsurance data. The red line on the upper tail is described in Section 2.2.

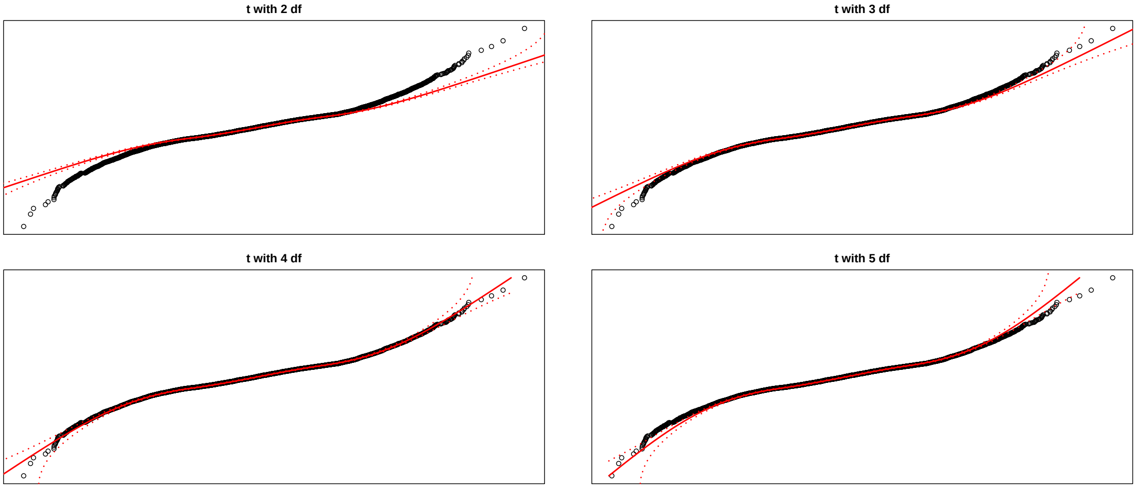

Figure 7: Simulated data from a t-distribution with 4 d.f. and $n = 2,000$ points. The same data is used in all four plots, shown as black circles. The solid lines compare that data to a variety of t-distributions with varying degrees of freedom. The dotted lines are pointwise 95% confidence intervals for each fitted models.

data. The method patches together a power law on the upper and lower tails with a monotonic spline fit to the standard ecdf in the middle range. A simple way to do this for the lower tail is to pick a quantile p_1 and define $z_1 = F^{-1}(p_1)$ and then select the pairs of plotted points corresponding to data values $x \leq z_1$. If the data has a power law decay for $x < z_1$, say $F(x) \sim c_1/|x|^{\alpha_1}$, then the ecdfHT plot will be linear on this region. A weighted linear regression gives estimates of α_1, where the weights are proportional to the variance $p_i(1-p_i)$ of the p-value at $x_{(i)}$. Likewise, for the upper tail pick a p_2 and let z_2 be the corresponding quantile, and do a regression on the points above z_2 to fit $F(x) = 1 - c_2/x^{\alpha_2}$. The constants c_1 and c_2 are picked to guarantee the tails match the empirical cdf at z_1 and z_2. Finally, for the middle range, i.e. for $x \in [z_1, z_2]$, fit a monotonic spline to the ecdf. We choose to use a linear interpolant because it has a simple inverse that is used for the inverse cdf. The result is a model $\widehat{F}(x)$ for the unknown population cdf $F(x)$. One advantage of this construction is that the resulting $\widehat{F}(x)$ is guaranteed to be a cdf. The **ecdfHT** package implements this and the resulting object can be used to compute $\widehat{F}$ at any real value x, an approximation $\widehat{f}(x)$ to the pdf at any x, and quantiles $\widehat{F}^{-1}(p)$ at any $p \in (0,1)$. Using the quantile function, it is easy and fast to simulate via $\widehat{F}^{-1}(U)$, where U is a Uniform(0,1) r.v. This can be useful when there is no convenient parametric model for the data.

Here are some brief comments on this method. First, choosing the cut-off values p_1 and p_2 is not an exact science: different choices will generally lead to different tail fits. Unfortunately, without any further assumptions on the data, there is little guidance that can be given. Second, the above procedure allows different tail exponents α_1 and α_2 for the lower and upper tails. One sided data can be handled by setting $p_1 = 0$ or $p_2 = 1$.

We end this section with an example using the one-sided Danish reinsurance data discussed above. The red plus sign on Figure 6 corresponds to the $p_2 = 0.95$ quantile of the data, and the red line is the regression line for all the points to the right of the corresponding $z_2 = 10.1$ and gives a tail fit with $c_2 = 1.49$ and tail exponent $\alpha_2 = 1.47$. Using this model, one can simulate future claims and roughly capture the heavy tail in the data. Using this tail fit with the function `pecdfHT` allows us to estimate the probability of a claim exceeding 400M DKK is approximately 0.0002197. This fit is heavier tailed than McNeil's generalized Pareto distribution fit, where the tail index based on a cut-off of 10 was $1/\xi = 1/0.497 = 2.01$. This difference is due to two factors: slightly different cut-off values and different models assumed in the fit.

3. Multivariate plots

Here we have data $\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n$, where each $\mathbf{X}_i = (X_{i,1}, \ldots, X_{i,d})$ is d-dimensional. We are interested in analyzing each component separately, and to get some idea of the joint dependence structure. First, one can do a sequence of univariate ecdfHT plots of each marginal. This requires scaling quantiles, so let $Q = [q_{i,j}]$ be a $(3 \times d)$ matrix, with column j being scale quantiles for the $j-$th component of the data. (The default values are to use the quartiles for all columns: $q_{k,j} = k/4$, but if the data is one-sided use $(0, 0, q_{3,j})$ or different scalings can be chosen for different components.) As in Section 2, let $t_{k,j} =$ the $q_{k,j}$ quantile for the $j-$th coordinates of the data and then using these values each component is centered and scaled to produce a sequence of d univariate plots to examine each component for heavy tailed behavior. Perhaps more useful is to superimpose all components on one plot so that the tail behaviors can be compared directly, using the component specific quantiles to have all coordinates be

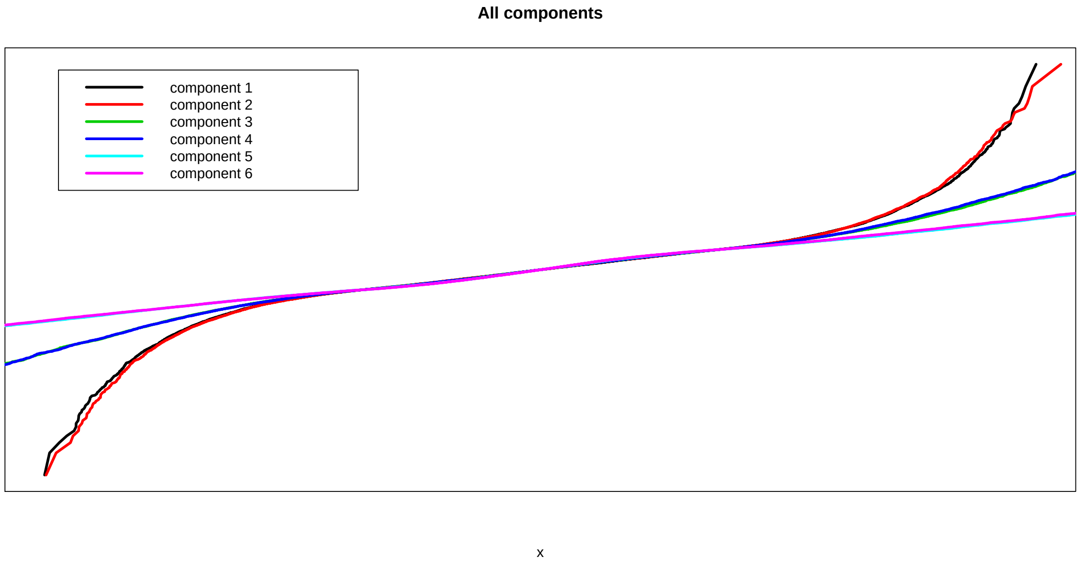

Figure 8: A $d = 6$ dimensional data set with the ecdfHT plot for all components superimposed on one graph.

on roughly the same scale. This is done in Figure 8 with a simulated six dimensional data set with all components independent but different tail behavior: the first two components are from a standard normal distribution, the third and fourth are from a t-distribution with 3 d.f., and the last two components are Cauchy. Note how the plot shows the differing tail behaviors.

Dependence can be hard to analyze with heavy tailed data. Standard scatterplots have the same problems that univariate plots have: extreme values visually dominate the plot and make it hard to see structure, see the left hand plot in Figure 9. In this plot the data is simulated from a bivariate t-distribution with 1 d.f. that is elliptically contoured with shape/dispersion matrix $\Sigma_1 = \begin{pmatrix} 1 & 0.8 \\ 0.8 & 1 \end{pmatrix}$. The elliptical structure is not visible in the standard scatterplot because the extremes dominate.

To construct a transformed scatterplot, first center and scale each coordinate using the above quantiles:

$$\mathbf{X}'_i = \left(\frac{X_{i,1} - t_{2,1}}{t_{3,1} - t_{1,1}}, \frac{X_{i,2} - t_{2,2}}{t_{3,2} - t_{1,2})}, \ldots, \frac{X_{i,d} - t_{2,d}}{t_{3,d} - t_{1,d}} \right).$$

Next, for each data vector, set $R_i = \|\mathbf{X}'_i\|$ = the length of centered $\mathbf{X}_i$. We will use Euclidean length here, but any p-norm can be used. Let r_0 be a measure of what is meant by a "large" R_i, say the q_0-th quantile of $R_1, R_2, \ldots, R_n$.

We next define a transformed vector for each $\mathbf{X}_i$:

$$\mathbf{Y}_i = \left(\frac{h(R_i|0, 0, r_0)}{R_i} \right) \mathbf{X}'_i$$

where $h(\cdot|\cdot,\cdot,\cdot)$ is defined by (1). This centers the data at $(t_{2,1}, t_{2,2}, \ldots, t_{2,d})$ and preserves the direction with respect to that point. The scale factor on the right is one if $R_i \leq r_0$, and logarithmic divided by linear if $R_i > r_0$, which pulls in extremes so that they don't dominate the plots visually.

The right plot of Figure 9 shows a scatterplot of the transformed data $(\mathbf{Y}_1, \ldots, \mathbf{Y}_n)$. Compared to the left plot, this plot clearly shows the elliptical structure of the data. As in the univariate case, the larger the data set, the more likely it is that a few isolated points will visually dominate the raw scatterplot. On the right plot the points inside the circle of radius r_0 are unchanged, but outside that circle the transformation pulls in extremes along the radial direction. While we have not shown it here, a kernel smoothed density estimator for the original data will have the same problems as the univariate density plots: near the center a density estimator will useful, but there will be isolated bumps at each extreme value.

When $d = 3$, the same transform can be used to pull in extremes, as in Figure 10. This is also an elliptical t distribution with shape/dispersion matrix

$$\Sigma_2 = \begin{pmatrix} 1 & 0.8 & 0.5 \\ 0.8 & 1 & 0.8 \\ 0.5 & 0.8 & 1 \end{pmatrix}.$$

When $d > 3$, the package produces two plots, one showing an R `pairs` of the original and the other a `pairs` of the transformed data, as in Figure 11 and 12 respectively. In these plots, the first two columns are t with 1 d.f. and shape matrix Σ_1 and the third and fourth columns are also t with 1 d.f. but a different shape matrix $\Sigma_3 = \begin{pmatrix} 1 & -0.6 \\ -0.6 & 1 \end{pmatrix}$. The first two columns are independent of the last two. This is not obvious in the raw scatterplots, but very obvious in the transformed scatterplots. (When there is independence and heavy tails, extremes cluster along the axes: it is very unlikely to see more than one coordinate large when components are indepedent.)

To evaluate tail behavior, we can do two things. First, we can plot the radii $R_1, \ldots, R_n$ using the univariate ecdfHT plot as in Figure 13. If the components all have the same tail decay, large values of R_i can come from any component; if the components have different decay rates, then the upper extremes of the R_i will come from the components with slowest decay.

This gives a sense of the overall tail behavior, but loses all information about the directional behavior. In the bivariate case we can examine directional behavior by adding probability as a third dimension to the scatterplots: let p_i = ecdf of the radii distribution evaluated at R_i. The first three dimensional plot is just the points $(X_{i,1}, X_{i,2}, p_i)$, it shows original scaling on the $x - y$ plane and a radii-ordered cdf in the vertical direction. As in one dimension, the plot is hard to interpret: most of the data is compressed in a small $x - y$ region with the extreme data values visually dominating. Also the vertical scale is hard to interpret, it is hard to determine the rate at which the p_i approach 1. A second 3d plot is to show $(Y_{i,1}, Y_{i,2}, g(p_i|0, 0, q_0))$, which pulls in extreme values radially in the $x - y$ plane and stretches the vertical axis as p approaches 1. The combination will show a linear cone pattern if the bivariate data has a power law decay. Examples of both of these are shown in Figure 14, where The `rgl` package in R allows one to rotate a 3-dimensional plot. This can be useful to rotate a graph like the ones in Figure 14 and examine behavior in all directions. If d is bigger than two, but not too large, then such plots can be done for all pairs of coordinates.

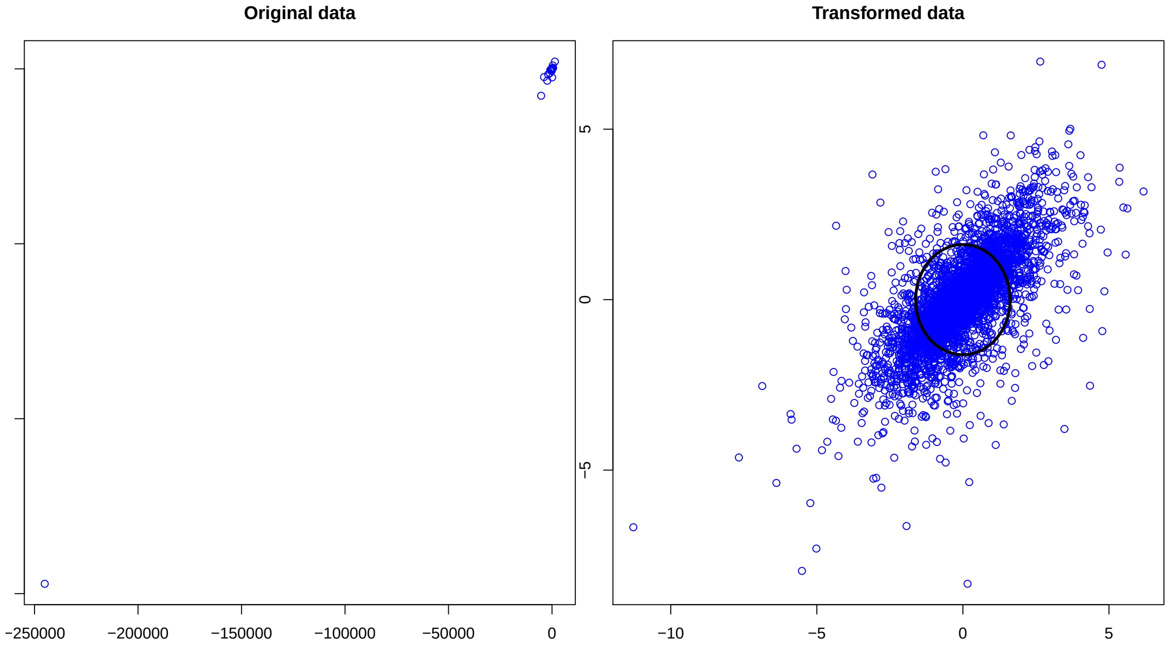

Figure 9: On the left is a scatterplot of the original data set of $n = 10,000$ values generated by simulation from a bivariate elliptical t-distribution with 1 d.f. and shape matrix Σ_1 given in the text. The right plot is the transformed scatterplot.

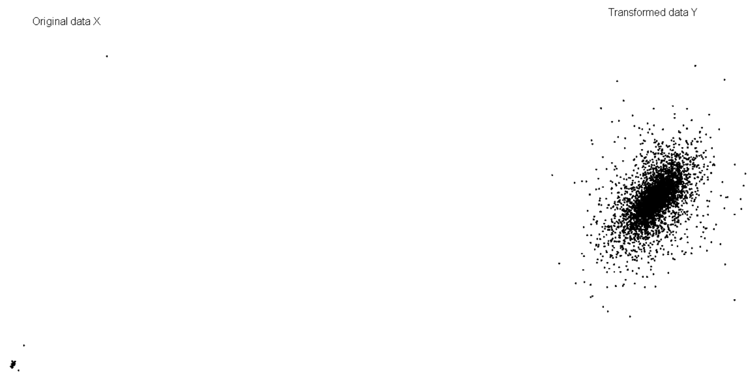

Figure 10: Standard and transformed scatterplots of $n = 10,000$ values generated by a three dimensional elliptical t-distribution with 1 d.f. and shape matrix Σ_2. Note how one extreme value on the left makes it hard to see the elliptical structure.

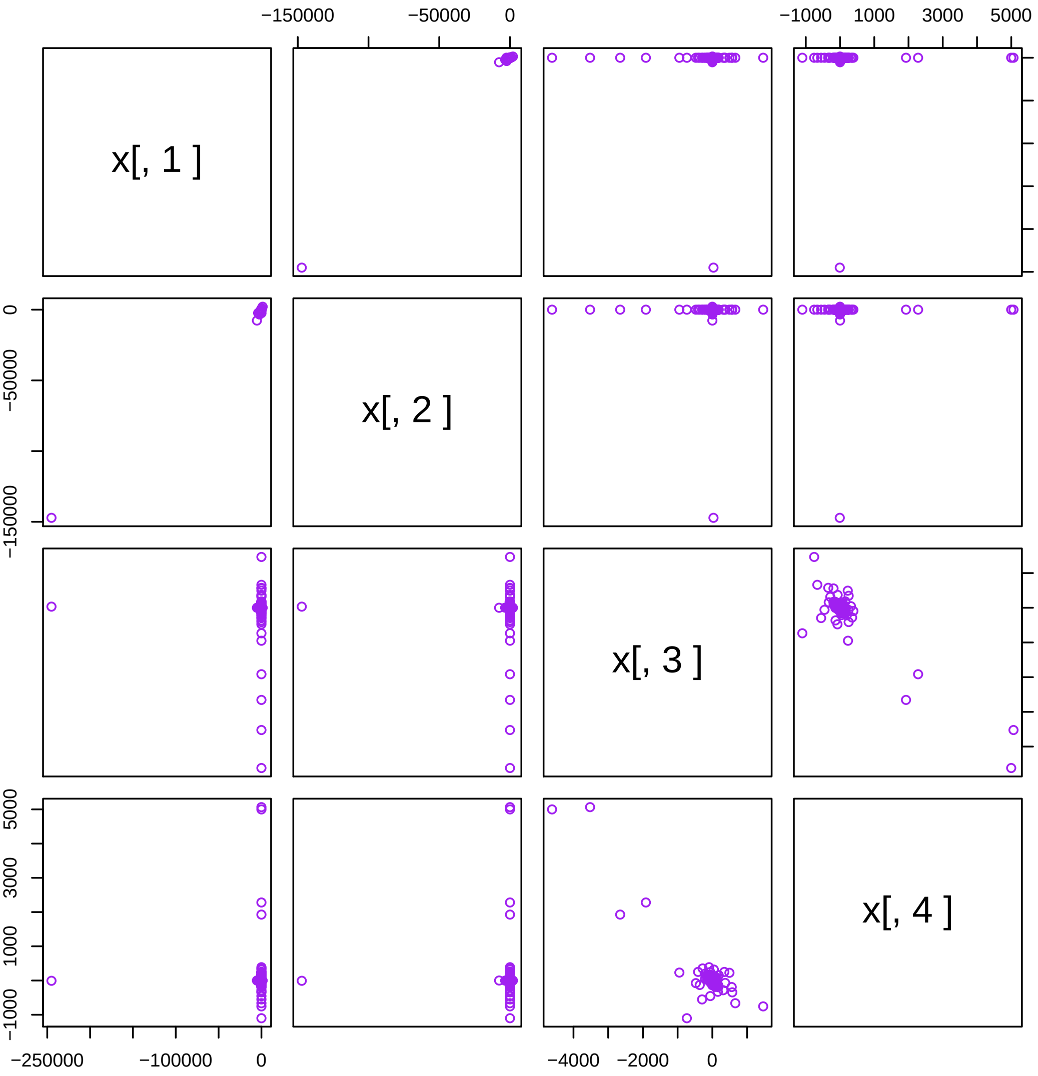

Figure 11: Pairwise scatterplots when $d = 4$.

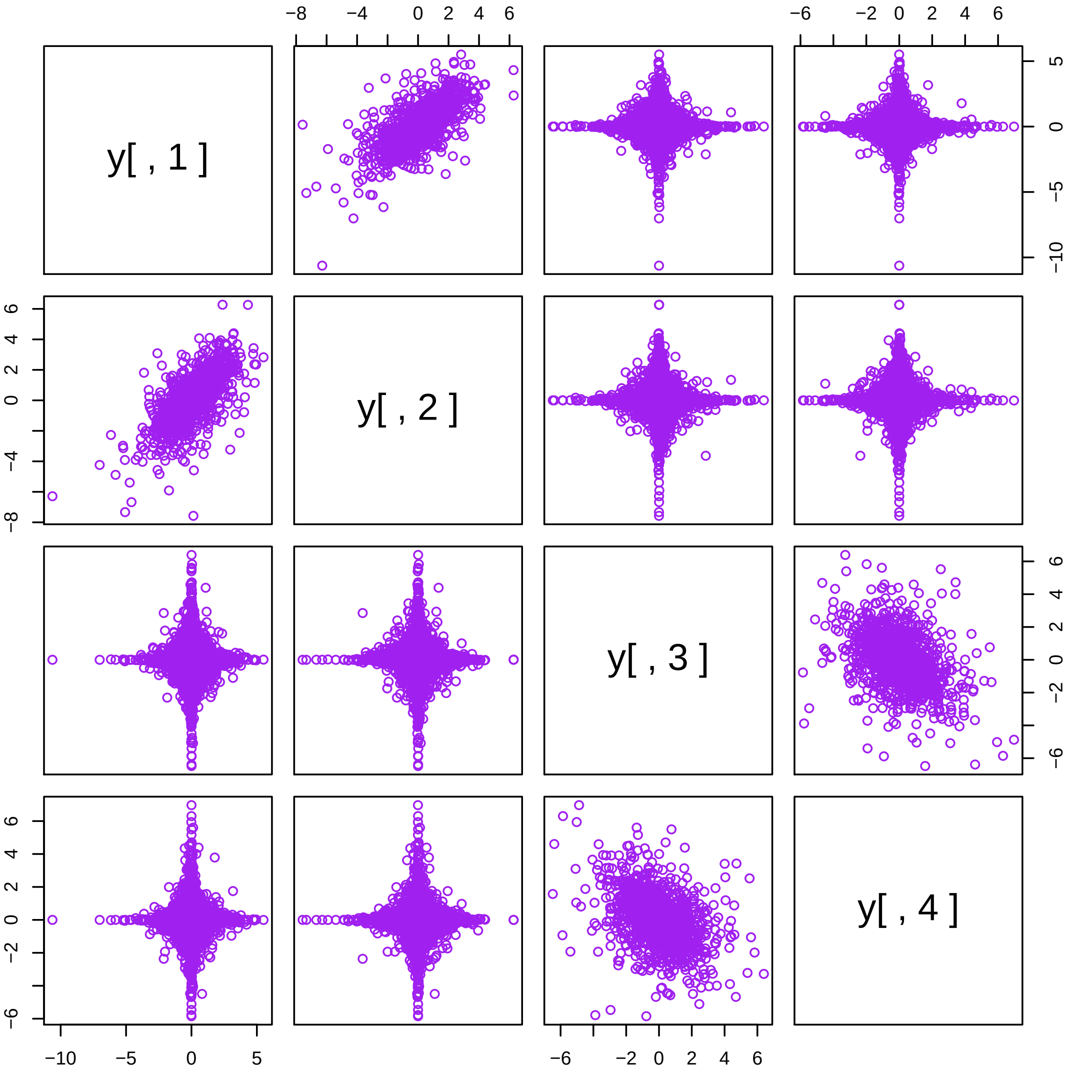

Figure 12: Pairwise transformed scatterplots when $d = 4$.

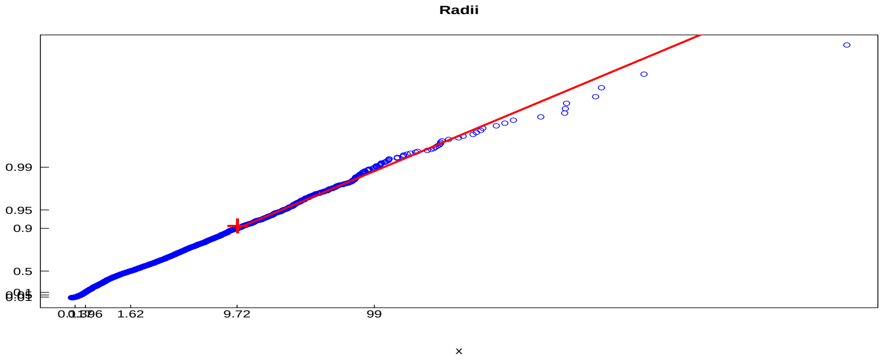

Figure 13: Univariate ecdfHT plot of the radii from the data in Figure 9.

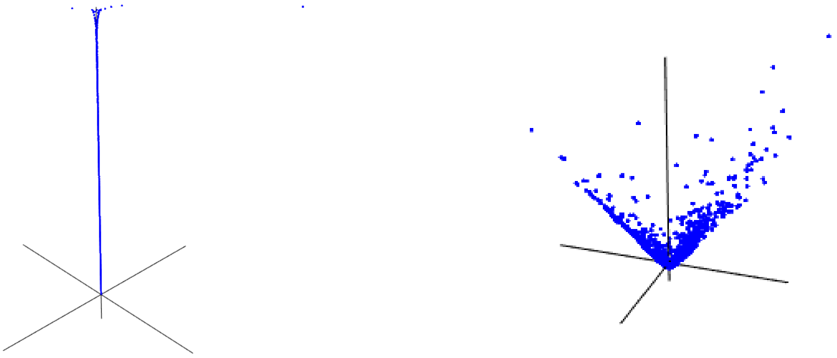

Figure 14: Plots of the bivariate radii-ordered cdf using the data from Figure 9. On the left is the raw data, with points $(X_{i,1}, X_{i,2}, p_i)$; the region has been clipped to exclude the most extreme values so that some detail can be shown. The shape is a very narrow cone: narrow because of the extreme data values on the $x - y$ plane. On the right is the transformed plot of $(Y_{i,1}, Y_{i,2}, g(p_i|0, 0, q_0))$.

4. Comments and Implementation

We end with some comments about further possibilities and outline the R implementation of this package.

If one is looking for a particular type of tail behavior, use different functions for $h(\cdot)$ and $g(\cdot)$. For example, if an exponential tail is suspected, then using $h(x) = x$ and $g(\cdot)$ as above, will show linear behavior on the tail(s) if there is exponential decay. The tail fitting procedure can be adapted to these other situations. Also, we implicitly assumed that the data is continuous when a continuous spline interpolant was used to fit the semi-parametric model; this can be modified to handle discrete data if the support is known, e.g. integer data.

In 2 or 3 dimensions one can restrict to a sector and view behavior in that sector. When $d > 2$, one can also do an animated tour as in Asimov (1985) and Wickham, Cook, Hofmann, and Buja (2011). When $d > 3$ and the components have the same tail decay, then one can get some understanding of the joint dependence by plotting different projections. Linear projections $\mathbf{u}^T\mathbf{X} = u_1X_1 + \cdots + u_dX_d$ are useful in the stable case, and max projections $\bigvee_{i=1}^d u_iX_i = \max(u_1X_1, \ldots, u_dX_d)$ are useful in the extreme value case.

These diagnostic plots are implemented in the R package `ecdfHT`, which is available on the CRAN network. The main user functions are:

- `ecdfHT` - the basic plot function. It takes a data set and scale quantiles, computes the transformation information, calls the second function to plot the data as in Figure 3.
- `ecdfHT.draw` - used to draw the basic plot and add other curves. It uses transformation information from the first function to add to the non-linear axes as in Figure 4.
- `ecdfHT.axes` - used to manually draw axes labels and grid lines.
- `ecdfHT.fit` - fit a semi-parametric model for a data set described in section 2.2.
- `pecdfHT` - compute a continuous interpolation of the cdf using a semi-parametric model.
- `decdfHT` - compute a continuous interpolation of the pdf using a semi-parametric model.
- `qecdfHT` - compute quantiles of a continuous interpolation of the cdf using a semi-parametric model.
- `recdfHT` - simulate from a semi-parametric model.
- ecdfHT.multivar - takes a d-dimensional data set and produces Figures 8-14.

References

Asimov D (1985). “The grand tour: a tool for viewing multidimensional data.” *SIAM J. Sci. Statist. Comput.*, **6**(1), 128–143.

D’Agostino RB, Stephens MA (1986). *Goodness-of-Fit Techniques.* Marcel Dekker.

McNeil AJ (1997). "Estimating the Tails of Loss Severity Distributions Using Extreme Value Theory." *ASTIN Bulletin*, **27**(1), 117–137. DOI: 10.2143/AST.27.1.563210.

Wickham H, Cook D, Hofmann H, Buja A (2011). "tourr: An R Package for Exploring Multivariate Data with Projections." *Journal of Statistical Software*, **40**(2), 1–18.

Affiliation:

John P. Nolan
Department of Mathematics and Statistics
American University
Washington, DC 20016 USA
E-mail: jpnolan@american.edu
URL: http://fs2.american.edu/jpnolan/www/

Journal of Statistical Software http://www.jstatsoft.org/
published by the Foundation for Open Access Statistics http://www.foastat.org/
MMMMMM YYYY, Volume VV, Issue II *Submitted:* 2016-08-31
doi:10.18637/jss.v000.i00 *Accepted:* yyyy-mm-dd