Journal of Statistical Software

October 2007, Volume 21, Issue 9. *http://www.jstatsoft.org/*

Fitting Single and Mixture of Generalized Lambda Distributions to Data via Discretized and Maximum Likelihood Methods: GLDEX in R

Steve Su
The George Institute for International Health

Abstract

This paper describes the use of **GLDEX** in R to fit distributions to empirical data using the discretized and maximum likelihood methods. The **GLDEX** package also provides diagnostic tests to examine the quality of fit through the resample Kolmogorov-Smirnoff test, quantile plots and comparison of the mean, variance, skewness and kurtosis between the empirical data and the fitted distribution.

Keywords: generalized lambda distributions, mixtures, maximum likelihood estimation, smoothing, R.

1. Introduction

GLDEX (Su 2007b) for R (R Development Core Team 2007) is designed to fit a range of empirical data using both the RS (Ramberg and Schmeiser 1974) and the FMKL (Freimer, Mudholkar, Kollia, and Lin 1988) generalized lambda distributions. For unimodal data, **GLDEX** provides the maximum likelihood estimation (Su 2007a) as well as the discretized approach (Su 2005), which acts as a smoothing device similar to the concept of loess smoothing. For bimodal data, **GLDEX** provides partition likelihood estimation and maximum likelihood estimation using the EM algorithm to find the parameters of the mixture of two generalized lambda distributions (Su 2006). As this package is built from **gld** (King 2007) in R with some modifications, the starship method of fitting FMKL $G\lambda D$ (generalized lambda distribution) to data (King and MacGillivray 1999) is also included. The quality of the distribution fit can be assessed by using histograms, qq pots and resample KS (Kolmogorov-Smirnov) test.

2. Background

2.1. Generalized lambda distributions

The RS $G\lambda D$ is due to the work of Ramberg and Schmeiser (1974) and it is an extension of Tukey's lambda distribution (Hastings, Mosteller, Tukey, and Windsor 1947). It is defined by its inverse distribution function:

$$F^{-1}(u) = \lambda_1 + \frac{u^{\lambda_3} - (1-u)^{\lambda_4}}{\lambda_2} \qquad 0 \leq u \leq 1 \tag{1}$$

From (1) $\lambda_1, \lambda_2, \lambda_3, \lambda_4$ are respectively the location, inverse scale, and shape parameters of generalized lambda distribution $G\lambda D(\lambda_1, \lambda_2, \lambda_3, \lambda_4)$. Karian, Dudewicz, and McDonald (1996) noted that $G\lambda D$ is defined only if $\frac{\lambda_2}{\lambda_3 u^{\lambda_3 - 1} + \lambda_4 (1-u)^{\lambda_4 - 1}} \geq 0$ for $0 \leq u \leq 1$.

Freimer *et al.* (1988) describe another distribution known as FMKL $G\lambda D$. This distribution is slightly different to RS $G\lambda D$. The FMKL $G\lambda D$ can be written as:

$$F^{-1}(u) = \lambda_1 + \frac{\frac{u^{\lambda_3}-1}{\lambda_3} - \frac{(1-u)^{\lambda_4}-1}{\lambda_4}}{\lambda_2} \qquad 0 \leq u \leq 1 \tag{2}$$

Under (2), $\lambda_1, \lambda_2, \lambda_3, \lambda_4$ are respectively the location, scale, and shape parameters of generalized lambda distribution.

The fundamental motivation for the development of FMKL $G\lambda D$ is that the distribution is defined over all λ_3 and λ_4 (Freimer *et al.* 1988). The only restriction on FMKL $G\lambda D$ is $\lambda_2 > 0$.

2.2. Fitting generalized lambda distributions to data

There are two possible general approaches to fitting generalized lambda distributions to data. The first approach is to fit $G\lambda D$ to the empirical data using the discretized method (Su 2005), similar to the concept of loess smoothing or kernel density estimation. This provides a range of different plausible distributions for the same data set, which can be very valuable in sensitivity analysis. The second approach aims to provide a definite fit to the data set such as maximizing the goodness of fit (King and MacGillivray 1999; Lakhany and Massuer 2000) or use the maximum likelihood estimation (Su 2007a, 2006). The current literature is primarily concerned with providing definite fits to a data set using the $G\lambda D$. The maximum likelihood estimation is usually the preferred method (Su 2007a). Maximum likelihood estimation is not only more efficient than the starship method but also tends to produce $G\lambda D$ that has closer first four moments to the data set.

The **GLDEX** package provides distribution fitting methods using both approaches. Specifically, it covers discretized and maximum likelihood approaches (Su 2007a, 2005, 2006). The starship method (King and MacGillivray 1999) is taken directly from **gld** but is included as part of this package to allow comparison of various fitting schemes.

To fit an implicitly defined distribution such as $G\lambda D$ to the data set, it is necessary to find: 1) suitable initial values and 2) optimize the values through an optimization scheme. The initial values and the optimization scheme required for each method (Su 2007a, 2005, 2006) are discussed below.

Discretized and maximum likelihood estimation for single GλD fit

1. Finding initial values.
   The first step is to generate a set of feasible initial values. The initial values for the RS $G\lambda D$ are derived from generating a set of low discrepancy quasi random values for λ_3 and λ_4 ranging from -1.5 to 1.5. These low discrepancy quasi random numbers can either be generated from the Halton or Sobol sequence. Similarly, for the FMKL $G\lambda D$, the initial λ_3 and λ_4 comprise of low discrepancy quasi random numbers ranging from -0.25 to 1.5. These values were chosen by the author as they appear to work well for a wide range of situations and can be modified if necessary.

   Once generated, λ_3 and λ_4 can be used to derive λ_1 and λ_2 using the method of moment for FMKL $G\lambda D$ (Lakhany and Massuer 2000) and method of percentile for RS $G\lambda D$ (Karian and Dudewicz 2000). Each set of $\lambda_1, \lambda_2, \lambda_3, \lambda_4$ is checked to ensure the set is a legal parameterization of the $G\lambda D$ and spans the entire data set.

   From these sets of valid initial values, **GLDEX** will attempt to find the best set of initial values for the subsequent optimization process. For the RS $G\lambda D$, the goal is to find the $G\lambda D$ that matches most closely to the third and fourth percentile of the data in terms of the sum of minimum squared deviations. Similarly, for the FMKL $G\lambda D$, the goal is to find the $G\lambda D$ that matches most closely to the third and fourth moment of the data, again using the minimum squared criterion.

2. Terminology.
   The distributional fitting methods developed in **GLDEX** begin by using the percentile approach for RS and method of moment for FMKL $G\lambda D$ to find initial values. Accordingly, the discretized approach for the RS and FMKL $G\lambda D$ is known as the revised percentile RS (RPRS) method and revised method of moment FMKL (RMFMKL) method respectively. The maximum likelihood approach adds a suffix .ML to the name, so the methods are labeled as RPRS.ML and RMFMKL.ML throughout this article. These abbreviations are used frequently in the **GLDEX** package and form a part of the graphical outputs to allow distinction between different fitting methods.

3. Discretized approach.
   In the discretized approach, the sample data is sorted in ascending order and divided into evenly spaced classes with bin edges that span the data set. Then the proportion of the sample in each class is calculated. An example is shown in Table 1.

   Table 1 shows four classes, with the proportion of the data set belonging to each class shown in the second row. For i = 1, 2, 3 ... k classes, the proportion of data in each class is defined as d_i and the proportion of data from the $G\lambda D$ is t_i. The quantity to be minimized under discretized approach is indicated in Equation (3) or (4).

Classes	1.5-2	2-2.5	2.5-3	3-3.5	Sum
Proportion of Data	0.1	0.6	0.2	0.1	1

Table 1: Proportion of data in each class for a sample data

$$\sum_{i=1}^{k} d_i(d_i - t_i)^2 \tag{3}$$

Equation (3) is the weighted squared deviation of the theoretical proportions with the empirical proportions. This weighting scheme forces data with higher proportions to be given priority in the minimization scheme and this tends to accentuate the peak and suppress the tails of the empirical data. The weighting factor d_i can be removed, resulting in Equation (4).

$$\sum_{i=1}^{k} (d_i - t_i)^2 \tag{4}$$

The number of classes, k, can be solely determined by the user, or determined by finding the number of classes that best matches the mean and variance of the actual data set in terms of minimum squared error. Using the mid point between the boundaries in each class, the mean and variance of data shown in Table 3 are 2.4 and 0.1525 respectively. Alternatively, it is possible to choose a different number of classes as described by Scott (1979) or Freedman and Diaconis (1981) for plotting histograms. The smoothing effect of the discretized method comes from the fact that different choices of k can give different $G\lambda D$ fits, providing a range of different distributions for the same data set.

4. Maximum likelihood estimation.
To find $G\lambda D$ using the maximum likelihood estimation, it is necessary to obtain quantiles u_i under the RS or FMKL $G\lambda D$ for every observation x_i, for i = 1, 2, 3, ... n observations under a set of initial values. This requires solving Equation (1) or (2) numerically. This can be done via **gld** in R which uses Newton-Raphson method.

   Once the u_i's are obtained, they are substituted into the appropriate numerical log likelihood equation as shown in (5) and (6).

$$ML_{RS} = \sum_{i=1}^{n} \log \left[\frac{\lambda_2}{\lambda_3 u_i^{\lambda_3 - 1} + \lambda_4 (1 - u_i)^{\lambda_4 - 1}} \right] \tag{5}$$

$$ML_{FMKL} = \sum_{i=1}^{n} \log \left[\frac{\lambda_2}{u_i^{\lambda_3 - 1} + (1 - u_i)^{\lambda_4 - 1}} \right] \tag{6}$$

   The key here is to maximize the likelihood in (5) and (6) and this can be done using Nelder-Simplex algorithm. To check the numerical optimization, it is always desirable to use a different set of initial values to see if similar results can be obtained in the optimization process.

Fitting mixture of two $G\lambda D$s: Partition maximum likelihood and EM algorithm

The mixture of two $G\lambda D$s is an extension of the maximum likelihood estimation for the single distribution fit case. The automated procedure begins by dividing the data into two parts using either `clara` or `fanny` from **cluster** (Maechler, Rousseeuw, Struyf, Hubert, and

Hornik 2007) in R. The `clara` clustering method appears to work well for a wide variety of empirical data and so the **GLDEX** uses this as the default. These classification procedures are described in Kaufman and Rousseeuw (1990). Any clustering method can be used, thus it is not necessary to use `clara` or `fanny` classification scheme. From the classification process, it is possible to obtain an estimate for p in the mixture distribution equation $pf_1 + (1-p)f_2$, with the $G\lambda D$s being represented by f_1 and f_2 respectively.

Under the partition maximum likelihood estimation, the above classification scheme is sufficient without the need for any further modification. In the case of maximizing the log likelihood using EM algorithm, **GLDEX** will force each partition of the data set to contain the maximum and minimum values of the entire data set as well as 1–2% of randomly selected data from the other group. For example, if data set 1 has 1000 observations and data set 2 has 500, data set 1 is modified to have 1011 observations, with 10 observations randomly selected from data set 2, plus 1 maximum value from data set 2, assuming data set 1 already contains the minimum value of the original data set. A similar procedure is also applied to data set 2. This ensures the partitioned data span the entire range of the data; a necessary step since maximizing the log likelihood for mixture data requires the distribution for each part of the mixture to span the entire data set.

Once the bimodal data set has been split into two, the sub data sets are fitted separately using the maximum likelihood estimation or the starship method. This step gives the necessary initial values for finding the parameters of the mixture $G\lambda D$.

For partition maximum likelihood estimation, the formulae to optimize is (7). This is also the complete log likelihood.

$$\sum_{i=1}^{n} (1-z)\{\log(f_0(x,\theta)) + \log(p)\} + z\{\log(f_1(x,\theta)) + \log(1-p)\} \tag{7}$$

For maximum likelihood estimation via EM algorithm, the conditional expectation of (7) given x is given in (8).

$$\sum_{i=1}^{n} T_i\{\log(f_0(x,\theta)) + \log(p)\} + S_i\{\log(f_1(x,\theta)) + \log(1-p)\} \tag{8}$$

$$P(Z_i|X_i = x_i) = \frac{f_1(x_i,\theta)(1-p)}{f_1(x_i,\theta)(1-p) + f_0(x_i,\theta)(p)} = S_i, 1 - S_i = T_i \tag{9}$$

In the above formulae, (8) and (9), X and Z are the complete data, with X $\sim f_0(x,\theta)$ if $z = 0$ and X $\sim f_1(x,\theta)$ if $z = 1$. The f_0 and f_1 are the $G\lambda D$ fits for each partition of the data set with θ representing the parameters associated with these distributions. In the case of two RS $G\lambda D$ partition maximum likelihood mixture distribution fits, the final equation to maximize is given in (10).

$$\left(\sum_{i=1}^{n_1} \log(p) + \log\left[\frac{\lambda_2}{\lambda_3 u_i^{\lambda_3 - 1} + \lambda_4(1-u_i)^{\lambda_4 - 1}}\right]\right) + \left(\sum_{j=1}^{n_2} \log(1-p) + \log\left[\frac{\delta_2}{\delta_3 v_i^{\delta_3 - 1} + \delta_4(1-v_i)^{\delta_4 - 1}}\right]\right) \tag{10}$$

In (10), $n_1 + n_2 = n$, n_1 and n_2 represent the number of observations in each partition of the data set and δ_k for k = 1, 2, 3, 4 represents the parameters of the second $G\lambda D$ fit. Additionally, u_i and v_i represent the quantiles for each partition of the data set for the i-th observation.

All the other combinations of different RS and FMKL $G\lambda D$ fits for complete data log likelihood and maximum likelihood via EM algorithm can be easily found by substituting the relevant distribution into (7) or (8) and hence will not be written in full here. In **GLDEX**, this final maximization step is done numerically via the Nelder-Mead Simplex algorithm and only solutions which span the entire original data set are accepted.

2.3. Assessing the quality of fit

Once a distribution has been fitted to the data set, it is possible to assess the quality of the distribution fit by using three methods in **GLDEX**:

1. Graphical outputs.
   The most obvious diagnostic check on the resulting distribution fit is to superimpose the resulting distribution fit on to the histogram. While simple and effective, it has shortcomings as it can be difficult to assess the adequacy of the distributional fit on the tails and it remains a subjective matter as to what constitutes a good or bad fit. Different classes or number of bins in the histogram can also give different distributional shape of the data set. It may not be easy to determine whether the resulting fit is adequate if it appears to capture the shape of the data very well under a histogram with 10 bins but not so well with 50. For this reason, quantile plots are also provided so that the user can see more objectively which part of the data the $G\lambda D$ distribution appears to give an adequate fit.

2. Comparing the mean, variance, skewness and kurtosis of the fitted distribution with the empirical data.
   This method provides a more objective way of choosing between alternative distributional fits. The derivation of the four moments of the RS and FMKL $G\lambda D$ (Karian and Dudewicz 2000; Lakhany and Massuer 2000) involves the use of beta function. It is imperative to appreciate, however, a $G\lambda D$ that has very similar mean, variance, skewness and kurtosis to the actual data may still be a bad fit (Karian and Dudewicz 2000; Lakhany and Massuer 2000). In some cases, it may be desirable to choose a good distributional fit with the closest mean, variance, skewness and kurtosis to the data set so that the fitted distribution can be used for simulation studies to model the population of interest.

3. KS resample test.
   Resample KS test assesses the similarity between fitted distribution and actual data by sampling a proportion (for example 90%) of the data and fitted distribution and calculating the KS test p-value. This process is then repeated many times, and the number of times the p-value is not significant is recorded and reported. For example, if 950 times out of 1000 times the p-value does not reject the null hypothesis, then it is possible to state that it is quite likely that the resulting fit is quite adequate for the given data set.

In R 2.5.0, the `ks.test` does not handle ties. While it is true that a real continuous distribution cannot have ties, real life data are often round figures which results in ties. In **GLDEX**, ties in `ks.gof` are handled by jittering, that is, if a tie appears, the tie will be added with a very small uniform random number generated from the minimum value of the data set divided by 10^8 to the minimum value divided by 10^7.

3. Using GLDEX in R

3.1. Installation

The package **GLDEX** is available from the Comprehensive R Archive Network at http://CRAN.R-project.org/. The following command will load the package and allow the user to browse a summary of the important functions of this package.

```
R> library("GLDEX")
R> ?GLDEX
```

3.2. Examples using GLDEX

Single distribution fit

The following examples illustrate distributional fitting methods for 300 randomly generated Weibull variates with shape = 3 and scale = 2. The generation of these random numbers is done as below.

```
R> set.seed(1000)
R> junk <- rweibull(300, 3, 2)
```

As an example, assume that the user wants to use the discretized weighted approach with the default number of classes to fit the `junk` data set. The following command will store the resulting RS and FMKL $G\lambda D$ fits in `obj.fit1.hs`.

```
R> obj.fit1.hs <- fun.data.fit.hs(junk)
```

To check the resulting fit, it is possible to plot a histogram as in Figure 1a.

```
R> fun.plot.fit(obj.fit1.hs, junk, nclass = 50, param = c("rs", "fmkl"),
+ xlab = "x")
```

The result seems adequate but should be verified by further testing. It is possible to assess the goodness of fit through use of the resample KS test. The KS resample test demonstrates the fit is inadequate and indicates that there is no difference between the fitted and simulated distribution at 5% significance level in just over half of the tests.

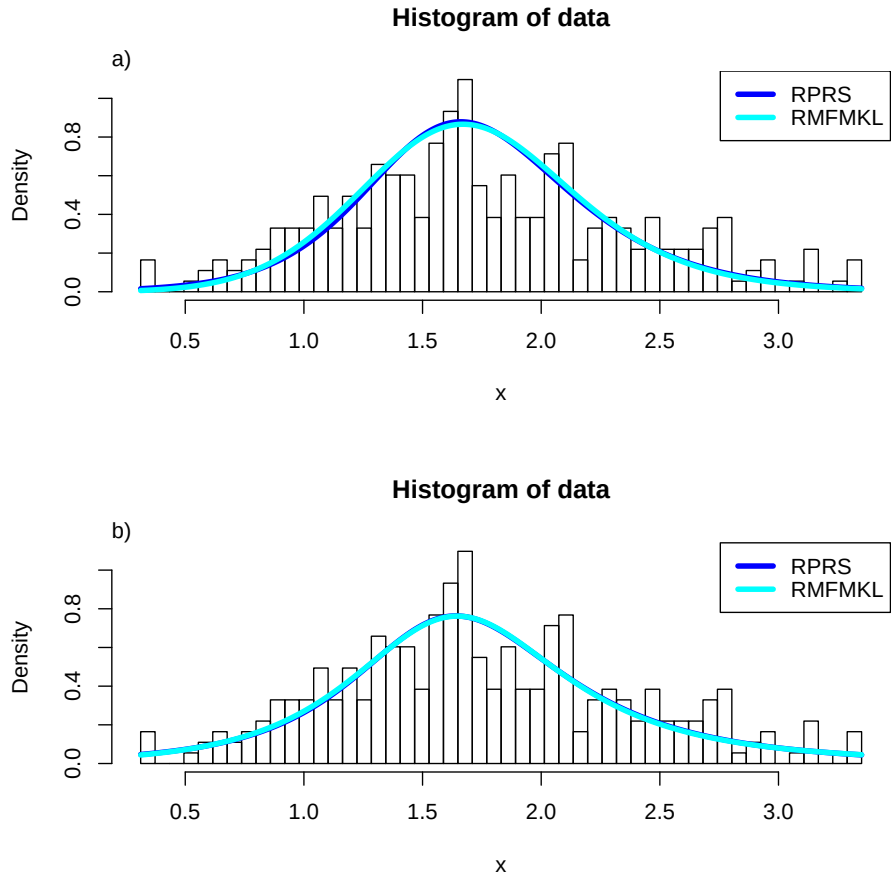

Figure 1: Examples of $G\lambda D$ fits using the weighted discretized method: a) `obj.fit1.hs` and b) `obj.fit2.hs`

```
R> fun.diag.ks.g(obj.fit1.hs[,1], junk, param = "rs")

[1] 646

R> fun.diag.ks.g(obj.fit1.hs[,2], junk, param = "fmkl")

[1] 591
```

Lastly, the following functions can be used to compare the theoretical mean, variance, skewness and kurtosis of the fitted distribution with the simulated `junk` data. There are significant deviations particularly in the skewness and kurtosis statistics.

```
R> fun.theo.mv.gld(obj.fit1.hs[1, 1], obj.fit1.hs[2, 1],
+ obj.fit1.hs[3, 1], obj.fit1.hs[4, 1], param = "rs")

        mean     variance      skewness      kurtosis
1.740486e+00 2.722239e-01 -7.440031e+02 -7.006149e+08
```

```
R> fun.theo.mv.gld(obj.fit1.hs[1, 2], obj.fit1.hs[2, 2],
+ obj.fit1.hs[3, 2], obj.fit1.hs[4, 2], param = "fmkl")

     mean  variance  skewness  kurtosis
1.7376465 0.2507075 0.4792900 4.0042590

R> unlist(fun.moments(junk))

       a1        a2        a3        a4
1.7583304 0.3800741 0.2766219 2.7230887
```

The advantage of the discretized approach to distributional fitting is that it is possible to change the number of classes to improve the fit. Alternatively, the unweighted discretized version using `fun.data.fit.hs.nw` can be used to avoid accentuating the peak of the data and suppressing the tails of the distribution. As a further example, assume the `junk` data is refitted using the weighted discretized method with the number of classes = 15.

```
R> obj.fit2.hs <- fun.data.fit.hs(junk, rs.default = "N",
+ fmkl.default = "N", no.c.rs =  15, no.c.fmkl = 15)
```

Using exactly the same code, replacing `obj.fit1.hs` with `obj.fit2.hs`, the graphical output is shown in Figure 1b.

The theoretical moments of the fitted distributions are again evaluated and these are quite different to the empirical moments and there are some undefined moments.

```
R> fun.theo.mv.gld(obj.fit2.hs[1, 1], obj.fit2.hs[2, 1],
+ obj.fit2.hs[3, 1], obj.fit2.hs[4, 1], param = "rs")

     mean  variance  skewness kurtosis
1.7679902 0.7307534 2.8705474       NA

Warning message: NaNs produced in: beta(a, b)

R> fun.theo.mv.gld(obj.fit2.hs[1, 2], obj.fit2.hs[2, 2],
+ obj.fit2.hs[3, 2], obj.fit2.hs[4, 2], param = "fmkl")

     mean  variance skewness kurtosis
1.8021270 0.9314562       NA       NA

Warning messages:
1: NaNs produced in: beta(a, b)
2: NaNs produced in: beta(a, b)
```

The KS resample test however suggests that the resulting fit is better than the previous fit. More than 90% of the time, the KS tests indicate there is no difference between the fitted distribution and the empirical data.

```
R> fun.diag.ks.g(obj.fit2.hs[,1], junk, param = "rs")

[1] 904

R> fun.diag.ks.g(obj.fit2.hs[,2], junk, param = "fmkl")

[1] 916
```

While the discretized methods act like smoothers with different degrees of smoothing applied under different number of classes, they do not provide a definite fit to the data set. The maximum likelihood estimation and the starship method are useful when it is preferable to find a definite fit to the empirical data, under the assumption that the data represents the underlying population with sufficient accuracy.

To fit the data using maximum likelihood estimation and starship method, the following function is used:

```
R> obj.fit1.ml <- fun.data.fit.ml(junk)
```

As in the previous example, it is possible to plot the resulting distribution fits as shown in Figure 2.

```
R> fun.plot.fit(obj.fit1.ml, junk, nclass = 50,
+ param = c("rs", "fmkl", "fmkl"), xlab = "x")
```

It is also possible to examine the quantiles using `qqplot.gld` as shown below. The quantile plots in Figure 3 suggest both starship and maximum likelihood estimation give very good fits.

```
R> par(mfrow = c(2, 3))
R> qqplot.gld(junk, obj.fit1.ml[,1], "rs", name = "RPRS.ML")
R> qqplot.gld(junk, obj.fit1.ml[,2], "fmkl", name = "RMFMKL.ML")
R> qqplot.gld(junk, obj.fit1.ml[,3], "fmkl", name = "STAR")
R> qqplot.gld(junk, obj.fit1.ml[,1], "rs", name = "RPRS.ML",
+ type = "str.qqplot")
R> qqplot.gld(junk, obj.fit1.ml[,2], "fmkl", name = "RMFMKL.ML",
+ type = "str.qqplot")
R> qqplot.gld(junk, obj.fit1.ml[,3], "fmkl", name = "STAR",
+ type = "str.qqplot")
```

The mean, variance, skewness and kurtosis of the fitted distribution are then compared with the actual data using `fun.comp.moments.ml`. As can be seen, the resulting fits have very close first four moments to the data set. It is often the case that the maximum likelihood estimation provides closer moments to the empirical data than the starship method. The `$eval.mat` gives the sum of the squared deviations between the theoretical four moments and the empirical four moments. It is designed to act as an alternative, objective way of assessing the closest overall estimation. It should be used with caution since a large deviation in one of the moments can seriously inflate `$eval.mat`.

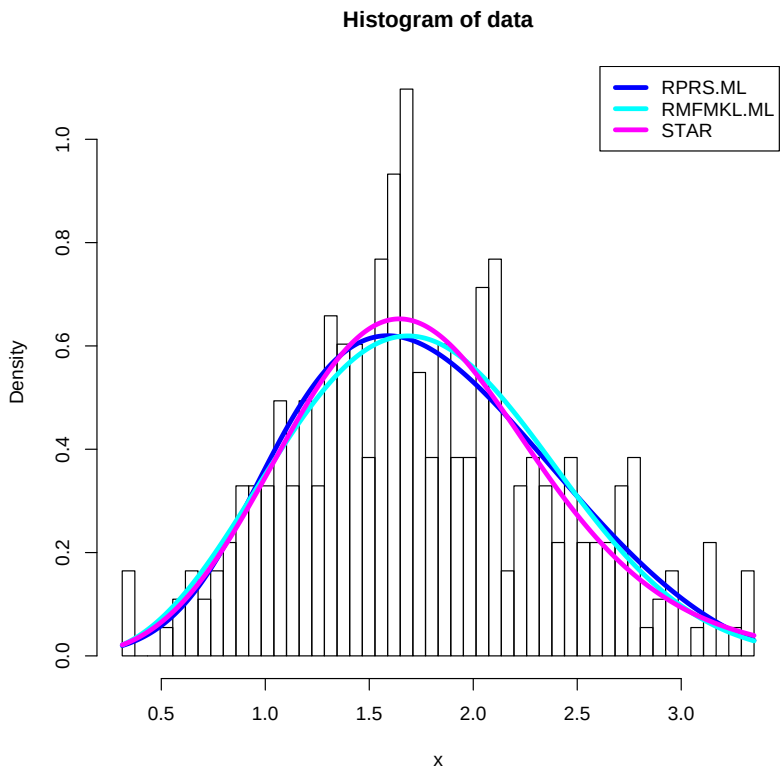

Figure 2: The $G\lambda D$ fits on the `junk` data using maximum likelihood estimation and starship method

```
R> fun.comp.moments.ml(obj.fit1.ml, junk)

$r.mat
           DATA ML   RPRS ML RMFMKL ML   STAR ML
mean     1.7583304 1.7607691 1.7590855 1.7639115
variance 0.3800741 0.3779583 0.3784118 0.4029118
skewness 0.2766219 0.2254039 0.2152840 0.5060725
kurtosis 2.7230887 2.6243629 2.7238032 3.5589265

$eval.mat
   RPRS ML  RMFMKL ML    STAR ML
0.15449838 0.06446981 1.09370711
```

Lastly, the KS resample test on the resulting fit confirms these $G\lambda D$ fits are satisfactory.

```
R> fun.diag2(obj.fit1.ml, junk, 1000)

       rs fmkl star
[1, ] 952  944  963
```

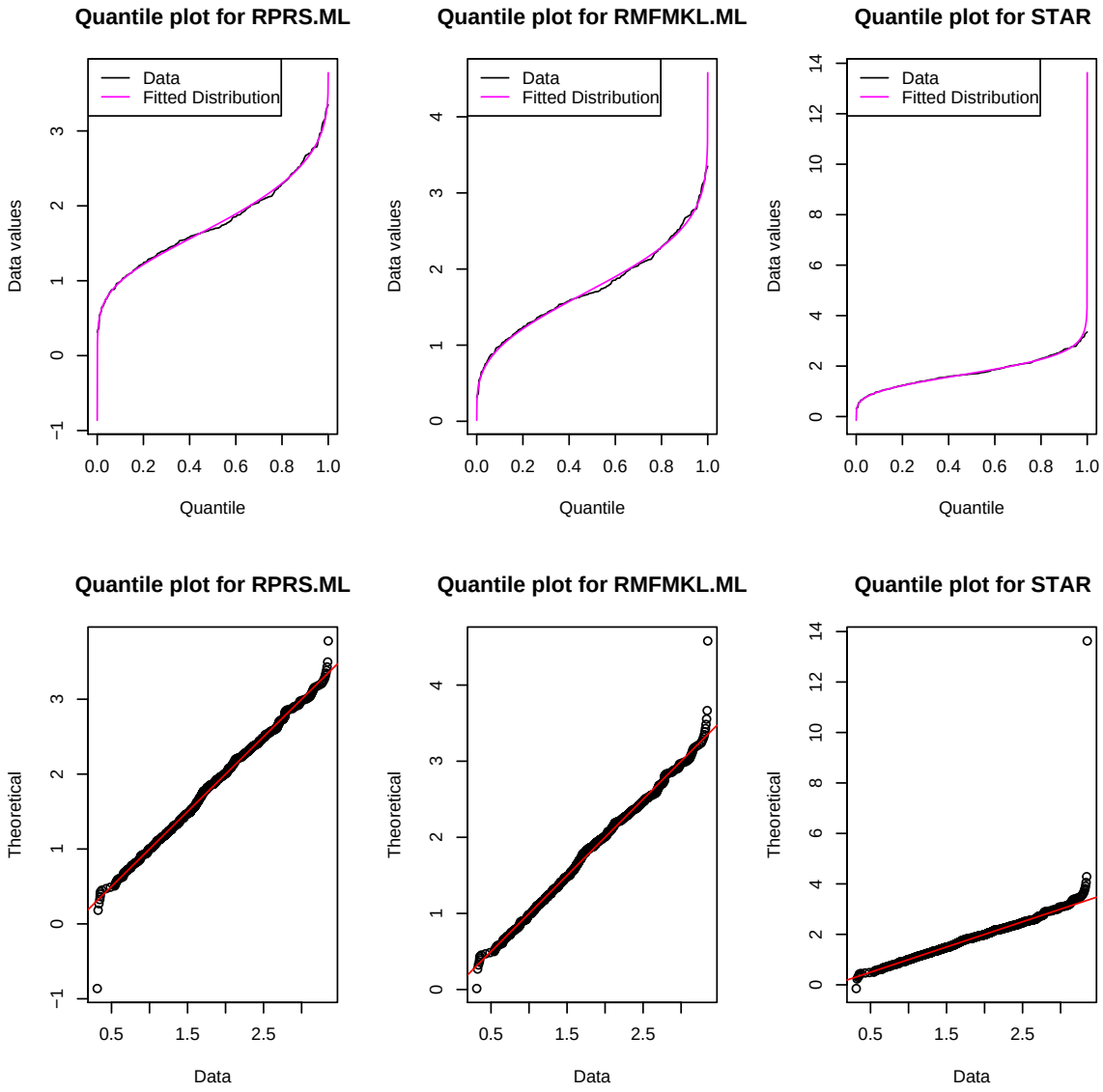

Figure 3: Quantile plots for $G\lambda D$ fits on the `junk` data using maximum likelihood estimation and starship method

The above distributional fits use the sobol sequence, however, it is possible to use other low discrepancy number generator such as `QUnif` or `runif.halton` to see if better solutions can be found. The syntax for running these procedures are shown below.

```
R> fun.data.fit.ml(junk, rs.leap = 409, fmkl.leap = 409, FUN = "QUnif")
R> fun.data.fit.ml(junk, rs.leap = 3, fmkl.leap = 3, FUN = "runif.halton")
```

Fitting mixture distributions

Due to the versatile and rich shapes of the $G\lambda D$s, they are particularly suited for mixture

modeling as they eliminate the need to choose between a wide range of different distributions on the same data set. To illustrate the partition maximum likelihood and the maximum likelihood estimation, the Old faithful data `faithful` is used for illustration.

To fit the first column of the `faithful` data using mixtures of RS and FMKL distributions, the functions `fun.auto.bimodal.ml` and `fun.auto.bimodal.pml` automate the mixture fitting procedure. The resulting fits are shown in Figure 4. The quantile plots are given by the `qqplot.gld.bi` function. In this case, the partition maximum likelihood method has minimum and maximum values beyond the range of the data set, hence the two out of place values from the straight line in the lower bottom qq plot of Figure 4.

```
R> par(mfrow = c(2, 3))
R> junk <- fun.auto.bimodal.ml(faithful[,1], per.of.mix = 0.01,
+ clustering.m = clara, init1.sel = "rprs", init2.sel = "rmfmkl",
+ init1 = c(-1.5, 1.5), init2 = c(-0.25, 1.5), leap1 = 3, leap2 = 3)
R> fun.plot.fit.bm(nclass = 50, fit.obj = junk, data = faithful[,1],
+ name = "Maximum likelihood using", xlab = "faithful1",
+ param.vec = c("rs", "fmkl"))
R> qqplot.gld.bi(faithful[,1], junk$par, param1 = "rs", param2 = "fmkl",
+ name = "\n Maximum likelihood", range = c(0.001, 0.999))
R> qqplot.gld.bi(faithful[,1], junk$par, param1 = "rs", param2 = "fmkl",
+ name = "\n Maximum likelihood", type = "str.qqplot",
+ range = c(0.001, 0.999))
R> junk <- fun.auto.bimodal.pml(faithful[,1], clustering.m = clara,
+ init1.sel = "rprs", init2.sel = "rmfmkl", init1 = c(-1.5, 1.5),
+ init2 = c(-0.25, 1.5), leap1 = 3, leap2 = 3)
R> fun.plot.fit.bm(nclass = 50, fit.obj = junk, data = faithful[,1],
+ name = "Partition maximum likelihood using", xlab = "faithful1",
+ param.vec = c("rs", "fmkl"))
R> qqplot.gld.bi(faithful[,1], junk$par, param1 = "rs", param2 = "fmkl",
+ name = "\n Partition Maximum likelihood")
R> qqplot.gld.bi(faithful[,1], junk$par, param1 = "rs", param2 = "fmkl",
+ name = "\n Partition Maximum likelihood", type = "str.qqplot")
```

Similarly, it is possible to compare the theoretical moments and do the KS resample test. It is also possible to use different low discrepancy quasi random numbers in the initial value search process. The following examples illustrate how these are done in **GLDEX**.

The fit begins by using the sobol sequence generator for the first distribution fit and the halton sequence for the second distribution fit and selects both distributions to be FMKL $G\lambda D$.

```
R> fit1 <- fun.auto.bimodal.ml(faithful[,1], init1.sel = "rmfmkl",
+ init2.sel = "rmfmkl", init1 = c(-0.25, 1.5), init2 = c(-0.25, 1.5),
+ leap1 = 3, leap2 = 3, fun1 = "runif.sobol", fun2 = "runif.halton")
```

After fitting the distribution, a very adequate fit can be observed by running the resample KS test.

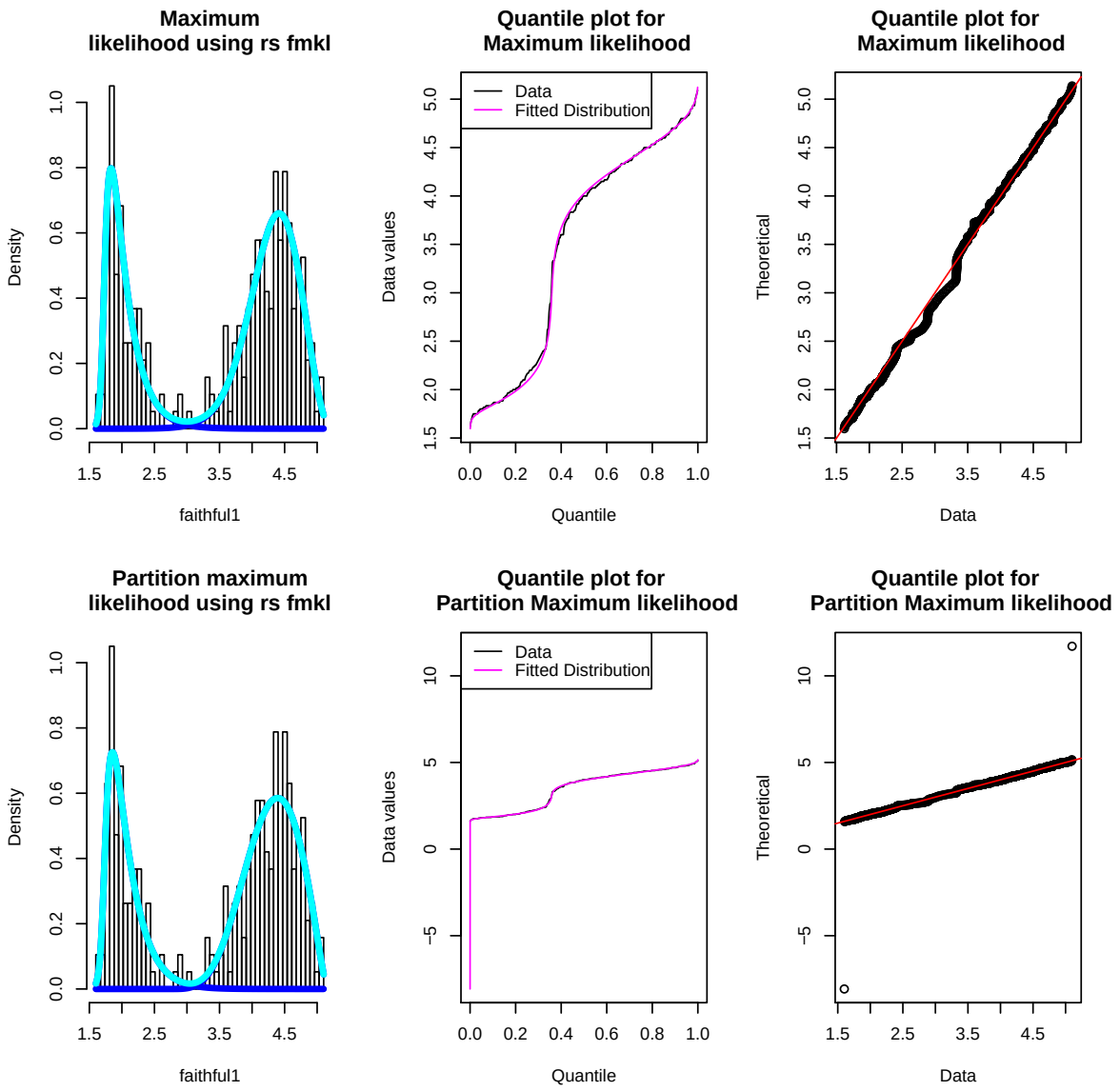

Figure 4: The $G\lambda D$ mixture fits on the `faithful[,1]` data using the maximum likelihood and partition maximum likelihood methods

```
R> fun.diag.ks.g.bimodal(fit1$par[1:4], fit1$par[5:8],
+ prop1 = fit1$par[9], data = faithful[,1], param1 = "fmkl",
+ param2 = "fmkl")

[1] 945
```

Then evaluate the theoretical moments of the distribution fit and compare them with the empirical moments from the data set. The result suggests there is a very good agreement.

```
R> fun.theo.bi.mv.gld(fit1$par[1], fit1$par[2], fit1$par[3],
```

```
+ fit1$par[4], "fmkl", fit1$par[5], fit1$par[6], fit1$par[7],
+ fit1$par[8], "fmkl", fit1$par[9])

     mean  variance   skewness  kurtosis
3.4903990 1.3013733 -0.4281729 1.4899611

R> unlist(fun.moments(faithful[,1]))

       a1        a2         a3        a4
3.4877831 1.2979389 -0.4158410 1.4993996
```

3.3. Limitations

The methods provided in **GLDEX** have been tested against a number of empirical data and the following limitations are acknowledged:

1. Initial value searching method using percentiles with RS $G\lambda D$ fails.
   This error usually arises when the number of observations is low (< 10) and the percentiles do not exist for a data set. This is not a serious error as it rarely happens when the sample size is large. Also, the initial value searching method using the method of moment with FMKL $G\lambda D$ can always be used when the percentile method fails.

2. Different quasi random numbers result in different $G\lambda D$ parameters.
   This problem relates to the convergence of numerical methods (Lakhany and Massuer 2000) and highlights the importance of running the fitting algorithm using different low discrepancy quasi random numbers (`QUnif`, `runif.sobol` and `runif.halton`) before determining the best fit. **GLDEX** provides these alternative ways to generate initial values thus reducing the convergence problem of numerical methods.

3. Slow.
   The fitting algorithms provided in **GLDEX** can be slow if the data set is ≥ 1000. Future research in this area involving narrowing the possible range of parameter values under a given data set may eliminate some of the search algorithm used in **GLDEX** and speed up the fitting algorithms.

4. Conclusion

This paper illustrates that several fitting algorithms can be used to fit $G\lambda D$ to data. When $G\lambda D$ was first introduced in the 20th century it received relatively little attention since computing power was limited. With the increased computing power today, it is now possible to fit these distributions with reasonable ease. With this development, statistical techniques can be tailored to a given data set rather than having to analyze data based on transforming the data into mean and relying on normality. It is hoped that this paper and the software module provided will increase the use of $G\lambda D$ among statisticians and encourage more research on its practical use.

Acknowledgments

The author is grateful to Lillias Nairn and Roma Kewani for their proof reading and encouragement.

References

Freedman D, Diaconis P (1981). "On the Histogram as a Density Estimator: L2 Theory." *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete*, **57**, 453–476.

Freimer M, Mudholkar G, Kollia G, Lin C (1988). "A Study of the Generalised Tukey Lambda Family." *Communications in Statistics – Theory and Methods*, **17**, 3547–3567.

Hastings JC, Mosteller F, Tukey J, Windsor C (1947). "Low Moments for Small Samples: A Comparative Study of Order Statistics." *The Annals of Statistics*, **18**, 413–426.

Karian Z, Dudewicz E (2000). *Fitting Statistical Distributions: The Generalized Lambda Distribution and Generalised Bootstrap Methods*. Chapman and Hall, New York.

Karian Z, Dudewicz E, McDonald P (1996). "The Extended Generalized Lambda Distribution Systems for Fitting Distributions to Data: History, Completion of Theory, Tables, Applications, the "Final Word" on Moment Fits." *Communications in Statistics – Computation and Simulation*, **25**(3), 611–642.

Kaufman L, Rousseeuw PJ (1990). *Finding Groups in Data: An Introduction to Cluster Analysis*. Wiley, New York.

King R (2007). ***gld**: Estimation and Use of the Generalised (Tukey) Lambda Distribution*. R package version 1.8.3, URL http://CRAN.R-project.org/.

King R, MacGillivray H (1999). "A Starship Estimation Method for the Generalised Lambda Distributions." *Australia and New Zealand Journal of Statistics*, **41**(3), 353–374.

Lakhany A, Massuer H (2000). "Estimating the Parameters of the Generalised Lambda Distribution." *Algo Research Quarterly*, pp. 47–58.

Maechler M, Rousseeuw P, Struyf A, Hubert M, Hornik K (2007). "**cluster**: Cluster Analysis Basics and Extensions." R package version 1.11.9, URL http://CRAN.R-project.org/.

Ramberg J, Schmeiser B (1974). "An Approximate Method for Generating Asymmetric Random Variables." *Communications of the Association for Computing Machinery*, **17**, 78–82.

R Development Core Team (2007). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org/.

Scott DW (1979). "On Optimal and Data-based Histograms." *Biometrika*, **66**, 605–610.

Su S (2005). "A Discretized Approach to Flexibly Fit Generalized Lambda Distributions to Data." *Journal of Modern Applied Statistical Methods*, **4**(2), 408–424.

Su S (2006). "Maximum Log Likelihood Estimation Using EM Algorithm and Partition Maximum Log Likelihood Estimation for Mixtures of Generalized Lambda Distributions." *Working paper*.

Su S (2007a). "Numerical Maximum Log Likelihood Estimation for Generalized Lambda Distributions." *Computational Statistics & Data Analysis*, **51**, 3983–3998.

Su S (2007b). ***GLDEX**: Fitting Single and Mixture of Generalized Lambda Distributions (RS and FMKL) Using Discretized and Maximum Likelihood Methods*. R package version 1.0.1, URL http://CRAN.R-project.org/.

Affiliation:

Steve Su
The George Institute for International Health
Affiliated with University of Sydney
PO Box M201, Missenden Road NSW 2050, Australia
E-mail: allegro.su@gmail.com
URL: http://www.georgeinstitute.org/

Journal of Statistical Software http://www.jstatsoft.org/
published by the American Statistical Association http://www.amstat.org/
Volume 21, Issue 9 *Submitted:* 2006-11-07
October 2007 *Accepted:* 2007-09-15