A New Package for the General Error Distribution

The normalp package

Angelo M. Mineo

Introduction

The General Error Distribution, whose first formulation could be ascribed to the Russian mathematician Subbotin (1923), is a general distribution for random errors. To derive this random error distribution, Subbotin extended the two axioms used by Gauss to derive the usual normal (Gaussian) error distribution, by generalizing the first one. Subbotin used the following axioms:

1. The probability of an error ε depends only on the greatness of the error itself and can be expressed by a function $\varphi(\varepsilon)$ with continuous first derivative almost everywhere.

2. The most likely value of a quantity, for which direct measurements x_i are available, must not depend on the adopted unit of measure.

In this way Subbotin obtains the probability distribution with the following density function:

$$\varphi(\varepsilon) = \frac{mh}{2\Gamma(1/m)} \cdot exp[-h^m|\varepsilon|^m]$$

with $-\infty < \varepsilon < +\infty$, $h > 0$ and $m \geq 1$. This distribution is also known as Exponential Power Distribution and it has been used, for example, by Box and Tiao (1992) in Bayesian inference. In the Italian statistical literature, a different parametrization of this distribution has been derived by Lunetta (1963), who followed the procedure introduced by Pearson (1895) to derive new probability distributions, solving this differential equation

$$\frac{d\log f}{dx} = p \cdot \frac{\log f - \log a}{x - c}$$

and obtaining a distribution with the following probability density function

$$f(x) = \frac{1}{2\sigma_p p^{1/p}\Gamma(1+1/p)} \cdot exp\left(-\frac{|x-\mu|^p}{p\sigma_p^p}\right)$$

with $-\infty < x < +\infty$ and $-\infty < \mu < +\infty$, $\sigma_p > 0$ and $p \geq 1$. This distribution is known as the order p normal distribution (Vianelli, 1963). It is easy to see how this distribution is characterized by three parameters: μ is the location parameter, σ_p is the scale parameter and p is the structure parameter. By changing the structure parameter p, we can recognize some known probability distribution: for example, for $p = 1$ we have the Laplace distribution, for $p = 2$ we have the normal (Gaussian) distribution, for $p \to +\infty$ we have the uniform distribution. A graphical description of some normal of order p curves is in figure 1 (this plot has been made with the command `graphnp()` of the package **normalp**).

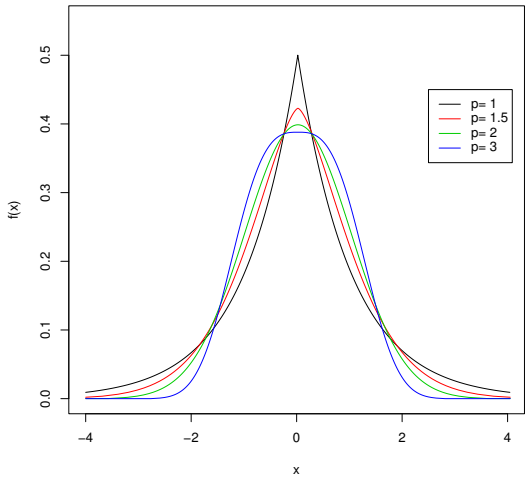

Figure 1: Normal of order p curves.

In this paper we present the main functions of the package **normalp** and some examples of their use.

The normalp functions

The package contains the four classical functions dealing with the computation of the density function, the distribution function, the quantiles and the generation of pseudo-random observations from an order p normal distribution. Some examples related to the use of these commands are the following:

```
> dnormp(3, mu = 0, sigmap = 1, p = 1.5,
+        log = FALSE)
[1] 0.01323032
> pnormp(0.5, mu = 2, sigmap = 3, p = 1.5)
[1] 0.3071983
> qnormp(0.3071983, mu = 2, sigmap = 3,
+        p = 1.5)
[1] 0.5
> rnormp(6, mu = 2, sigmap = 5, p = 2.5)
[1]  3.941597 -1.943872 -2.498598
[4]  1.869880  6.709037 14.873287
```

In case of generation of pseudo-random numbers we have implemented two methods: one, faster, based

on the relationship linking an order p normal distribution and a gamma distribution (see Lunetta, 1963), and one based on the generalization of the Marsaglia (1964) method to generate pseudo-random numbers from a normal distribution. Chiodi (1986) describes how the representation of the order p normal distribution as a generalization of a normal (Gaussian) distribution can be used for simulation of random variates.

Another group of functions concerns the estimation of the order p normal distribution parameters. To estimate the structure parameter p, an estimation method based on an index of kurtosis is used; in particular, the function `estimatep()` formulates an estimate of p based on the index VI given by

$$VI = \frac{\sqrt{\mu_2}}{\mu_1} = \frac{\sqrt{\Gamma(1/p)\Gamma(3/p)}}{\Gamma(2/p)}.$$

by comparing its theoretical value and the empirical value computed on the sample. For a comparison between this estimation method and others based on the likelihood approach see Mineo (2003). With the function `kurtosis()` it is possible to compute the theoretical values of, besides VI, β_2 and β_p given by

$$\beta_2 = \frac{\mu_4}{\mu_2^2} = \frac{\Gamma(1/p)\Gamma(5/p)}{[\Gamma(3/p)]^2}$$

$$\beta_p = \frac{\sqrt{\mu_{2p}}}{\mu_p^2} = p + 1$$

Moreover, it is possible to compute the empirical values of these indexes given by

$$\widehat{VI} = \frac{\sqrt{n\sum_{i=1}^n (x_i - M)^2}}{\sum_{i=1}^n |x_i - M|}$$

$$\hat{\beta}_2 = \frac{n\sum_{i=1}^n (x_i - M)^4}{[\sum_{i=1}^n (x_i - M)^2]^2}$$

$$\hat{\beta}_p = \frac{n\sum_{i=1}^n |x_i - M|^{2p}}{[\sum_{i=1}^n |x_i - M|^p]^2}.$$

Concerning the estimation of the location parameter μ and the scale parameter σ_p, we have used the maximum likelihood method, conditional on the estimate of p that we obtain from the function `estimatep()`. The function we have to use in this case is `paramp()`. We have implemented also a function `simul.mp()`, that allows a simulation study to verify the behavior of the estimators used for the estimation of the parameters μ, σ_p and p. The compared estimators are: the arithmetic mean and the maximum likelihood estimator for the location parameter μ, the standard deviation and the maximum likelihood estimator for the scale parameter σ_p; for the structure parameter p we used the estimation method implemented by `estimatep()`. Through the function `plot.simul.mp()` it is possible to see graphically the behavior of the estimators. A possible use of the function `simul.mp()` is the following:

```
> res <- simul.mp(n = 30, m = 1000, mu = 2,
+     sigmap = 3, p = 3)
> res
              Mean        Mp         Sd
Mean     1.9954033 1.9991151 2.60598964
Variance 0.2351292 0.2849199 0.08791664

                Sp       p
Mean     2.9348828 3.415554
Variance 0.5481126 7.753024

N. samples with a difficult convergence: 26
> plot(res)
```

The command `plot(res)` will produce an histogram for every set of estimates created by the function `simul.mp()`. In figure 2 we have the histogram for $\hat{p}$. For more details see Mineo (1995-a).

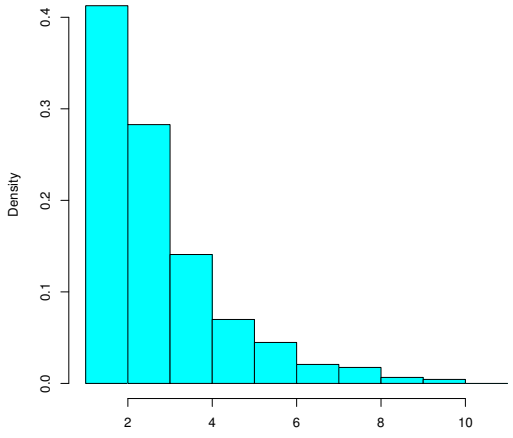

Figure 2: Histogram of $\hat{p}$ obtained with the command `plot.simul.mp(res)`.

It is also possible to estimate linear regression models when we make the hypothesis of random errors distributed according to an order p normal distribution. The function we have to use in this case is `lmp()`, which we can use like the function `lm()` from the **base** package. In fact, the function `lmp()` returns a list with all the most important results drawn from a linear regression model with errors distributed as a normal of order p curve; moreover, it returns an object that can form the argument of the functions `summary.lmp()` and `plot.lmp()`: the function `summary.lmp()` returns a summary of the main obtained results, while the function `plot.lmp()` returns a set of graphs that in some way reproduces the analysis of residuals that usually we conduct when

we estimate a linear regression model with errors distributed as a normal (Gaussian) distribution.

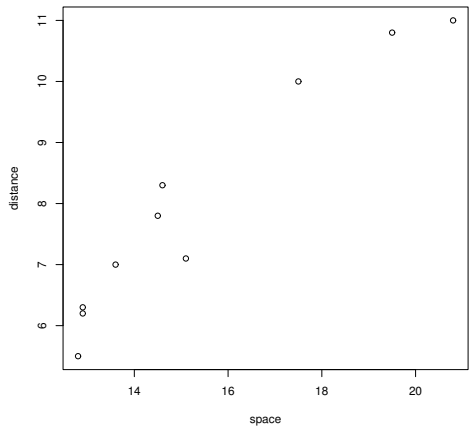

Figure 3: Plot of the data considered in the data frame `cyclist`.

To show an example of use of these functions, we considered a data set reported in Devore (2000). In this data set (see figure 3) the distance between a cyclist and a passing car (variable `distance`) and the distance between the centre line and the cyclist in the bike lane (variable `space`) has been recorded for each of ten streets; by considering the variable `distance` as a dependent variable and the variable `space` as an independent variable, we produce the following example:

```
> data(ex12.21, package = "Devore5")
> res <- lmp(distance ~ space,
+      data = ex12.21)
> summary(res)

Call:
lmp(formula = distance ~ space,
                        data = ex12.21)

Residuals:
    Min      1Q  Median      3Q     Max
-0.7467 -0.5202  0.0045  0.3560  0.8363

Coefficients:
(Intercept)       space
    -2.4075      0.6761

Estimate of p
1.353972

Power deviation of order p: 0.6111
> plot(res)
```

In figure 4 we show one of the four graphs that we have obtained with the command `plot(res)`.

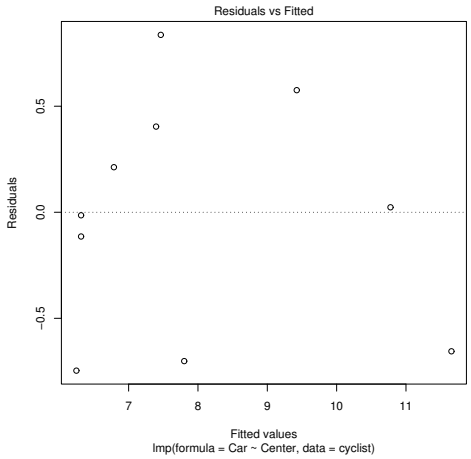

Figure 4: First graph obtained by using the command `plot.lmp(res)`.

Also for a linear regression model with errors distributed as an order p normal distribution we have implemented a set of functions that allow a simulation study to test graphically the suitability of the estimators used. The main function is `simul.lmp()`; besides this function, we have implemented the functions `summary.simul.lmp()` and `plot.simul.lmp()` that allow respectively to visualize a summary of the results obtained from the function `simul.lmp()` and to show graphically the behavior of the produced estimates. A possible use of these functions is the following:

```
> res <- simul.lmp(10, 500, 1, data = 1.5,
+    int = 1, sigmap = 1, p = 3, lp = FALSE)
> summary(res)
Results:
         (intercept)       x1
Mean       0.9959485 1.497519
Variance   0.5104569 1.577187
                  Sp         p
Mean       0.90508039  3.196839
Variance   0.04555003 11.735883

Coefficients: (intercept)          x1
                      1.0         1.5

Formula: y ~ +x1

Number of samples: 500

Value of p: 3

N. of samples with problems on convergence 10

> plot(res)
```

In figure 5 it is showed the result of `plot(res)`. For more details see Mineo (1995-b).

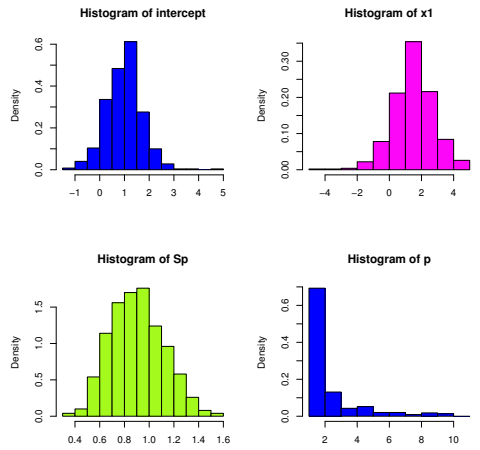

Figure 5: Graphs obtained with the command `plot.simul.lmp(res)`.

Besides the described functions, we have implemented two graphical functions. The command `graphnp()` allows visualization of up to five different order p normal distributions: this is the command used to obtain the plot in figure 1. The command `qqnormp()` allows drawing a Quantile-Quantile plot to check graphically if a set of observations follows a particular order p normal distribution. Close to this function is the command `qqlinep()` that sketches a line passing through the first and the third quartile of the theoretical order p normal distribution, line sketched on a normal of order p Q-Q plot derived with the command `qqnormp()`. In figure 6 there is a graph produced by using these two functions.

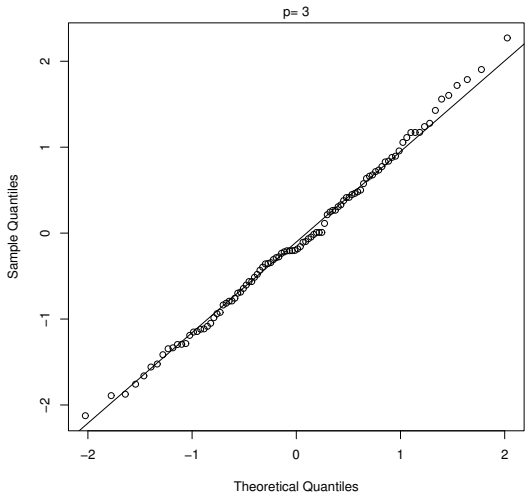

Figure 6: Normal of order p Q-Q plot.

Conclusion

In this article we have described the use of the new package **normalp**, that implements some useful commands where we have observations drawn from an order p normal distribution, known also as general error distribution. The implemented functions deal essentially with estimation problems for linear regression models, besides some commands that generalize graphical tools already implemented in the package **base**, related to observations distributed as a normal (Gaussian) distribution. In the next future we shall work on the computational improvement of the code and on the implementation of other commands to make this package still more complete.

Bibliography

G.E.P. Box and G.C. Tiao. *Bayesian inference in statistical analysis*. Wiley, New York, 1992. First edition for Addison-Wesley, 1973.

M. Chiodi. Procedures for generating pseudo-random numbers from a normal distribution of order p (p>1). *Statistica Applicata*, 1:7-26, 1986.

J.L. Devore. *Probability and Statistics for Engineering and the Sciences (5th edition)*. Duxbury, California, 2000.

G. Lunetta. Di una generalizzazione dello schema della curva normale. *Annali della Facoltà di Economia e Commercio dell'Università di Palermo*, 17:237-244, 1963.

G. Marsaglia and T.A. Bray. A convenient method for generating normal variables. *SIAM rev.*, 6:260-264, 1964.

A.M. Mineo. Stima dei parametri di intensità e di scala di una curva normale di ordine p (p incognito). *Annali della Facoltà di Economia dell'Università di Palermo (Area Statistico-Matematica)*, 49:125-159, 1995-a.

A.M. Mineo. Stima dei parametri di regressione lineare semplice quando gli errori seguono una distribuzione normale di ordine p (p incognito). *Annali della Facoltà di Economia dell'Università di Palermo (Area Statistico-Matematica)*, 49:161-186, 1995-b.

A.M. Mineo. On the estimation of the structure parameter of a normal distribution of order p. To appear on *Statistica*, 2003.

K. Pearson. Contributions to the mathematical theory of evolution. II. Skew variation in homogeneous material. *Philosophical Transactions of the Royal Society of London (A)*, 186:343-414, 1895.

M.T. Subbotin. On the law of frequency of errors. *Matematicheskii Sbornik*, 31:296-301, 1923.

S. Vianelli. La misura della variabilità condizionata in uno schema generale delle curve normali di frequenza. *Statistica*, 23:447-474, 1963.

Angelo M. Mineo
University of Palermo, Italy
elio.mineo@dssm.unipa.it