Kumaraswamy Normal and Azzalini's skew Normal modeling asymmetry

Michelle A. Correa[1†], *Denismar A. Nogueira*[2], *Eric B. Ferreira*[2]

[1] *Degree in Mathematics, Federal University of Alfenas.*

[2] *Adjunct Lecturer III, Exact Sciences Institute, Federal University of Alfenas.*

Resumo: *Este trabalho apresenta a comparação de duas distribuições de probabilidade com parâmetros específicos para determinação da assimetria. As distribuições kum-normal e a normal assimétrica foram escolhidas por apresentarem, como caso particular, a distribuição normal. A qualidade do ajuste, a flexibilidade de assimetria e a quantidade de parâmetros foram fatores usados para comparação. Pesquisas afirmam que a normal assimétrica possui limitações em relação à flexibilidade da cauda, apresentando uma certa resistência na modelagem da assimetria, pois, com o aumento do valor absoluto do parâmetro que modela a assimetria esta tende a uma* half-*normal. Os objetivos deste trabalho foram: implementar a distribuição kum-normal e, com o uso de simulação Monte Carlo, gerar dados com níveis crescentes de assimetria, para eleger o melhor ajuste. As distribuições também foram comparadas quanto ao ajuste do dados reais de besouros* Tribolium cofusum*, cultivados a 29° C. Para a implementação foi utilizado o pacote* `gamlss` *do software R, que permitiu o ajuste dos modelos, simulação de dados de distribuições generalizadas, e obtenção do critério de informação de Akaike, critério de informação bayesiano e o teste da razão de verossimilhança, utilizados para comparação. A distribuição kum-normal ajustou-se melhor com o aumento do nível de assimetria, quando comparada à distribuição normal assimétrica. Para os dados reais as duas distribuições não diferiram significativamente, apresentando equivalente estimação do grau de assimetria destes dados.*

Palavras-chave: Estatística; Distribuições de Probabilidade; Software R.

Abstract: *This paper presents the comparison of two probability distributions with specific parameters for modelling asymmetry. Kum-normal and Azzalini's skew normal distributions were chosen because they turn, in special case, into the normal distribution. The quality of the fit, flexibility and amount of asymmetry parameters were factors used for comparison. Researches state that the Azzalini's skew normal distribution has limitations regarding the flexibility of the tail, presenting certain resistance in modelling asymmetry since, by increasing the absolute value of the asymmetry parameter, it tends to a* half-*normal distribution. The objectives of this study were to implement a kum-normal distribution and, using Monte Carlo simulation to generate data with increasing levels of asymmetry, choose the best fit. The distributions were also compared in modelling a beetle data set (*Tribolium cofusum*), grown at 29° C. For implementation we used the R package* `gamlss`*, that allows adjusting of the models, simulating data of generalized distributions and obtaining the Akaike information criterion, Bayesian information criterion and likelihood ratio test, used for comparison. The kum-normal distribution was better adjusted by increasing the level of asymmetry compared to Azzalini's skew normal distribution. For real data the two distributions do not differ significantly, showing equivalent estimation of the degree of asymmetry of these data.*

Keywords: Statistics. Probability Distributions. R software.

† *Corresponding author:* michellecorrea2007@hotmail.com.

Introduction

In practical situations, data modelling is a great challenge. There is the need to find an appropriate distribution to describe the data that represents them in the best way. Furthermore, we invariably want to perform some sort of inference about a bigger and unobserved population. One of the most common assumptions of inference procedures is normality of the data, which is not always met. Among the cases in which the data is non-normal stand out asymmetric data. An alternative that can be used for handle asymmetric data are probability distributions that can model this asymmetry and, if possible, have as a special case the normal distribution.

Azzalini, in 1985, has proposed the asymmetric (skew) normal distribution, comprised by three parameters. One of those parameters (λ) is responsible for controlling the asymmetry. Azzalini states that the asymmetric normal has some limitations, such as the flexibility of its tail, resisting adjusting since, as the parametric value increases, this distribution tends to a *half*-normal.

Oliveira (2009) states the definition and properties of the Azzalini's skew normal distribution. The parameter responsible for asymmetry indicates that for positive (negative) values the density assumes positive (negative) asymmetry.

When $\lambda = 0$ we have the symmetry situation. Therefore, the standard normal distribution is a particular case. The probability density function of the Azzalini's skew normal is given by:

$$f(x|\mu, \sigma, \lambda) = \frac{2}{\sigma}\, \phi\left(x\right) \Phi\left(\lambda x\right),\ x \in (-\infty, \infty).$$

where $\mu \in \mathbb{R}$ is the mean or location parameter; σ^2 is the variance or scale parameter; $\phi(.)$ and $\Phi(.)$ are normal probability density function and distribution function, respectively.

The distribution function is given by

$$F(x|\mu, \sigma, \lambda) = 2\Phi_2\left(x|\mathbf{0}, \Omega\right),$$

where

$$\Omega = \begin{bmatrix} 1 & -\delta \\ -\delta & 1 \end{bmatrix},\ \delta = \frac{\lambda}{\sqrt{1+\lambda^2}} \qquad x \in (-\infty, \infty)$$

and $\Phi_2(.,.|\mathbf{0}, \Omega)$ is the cumulative bivariate normal function with null vector of means and covariance matrix Ω.

The two parameter Kumaraswamy distribution was created by Pondi Kumaraswamy in 1980, in applications in hydrology (KUMARASWAMY, 1980). In 2011, Cordeiro and Castro have created a family of generalized distributions derived from the distribution initially proposed by Kumaraswamy. Honouring this author, Cordeiro and Castro (2011) called this family of *kum*. The kum family of distributions was constructed from the mixture of existing distributions, for instance the kum-Weibull, kum-gamma and kum-normal distributions. Relating the works of Eugene, Lee and Famoye (2002) and Jones (2004, 2008) it has been constructed a new class of generalized kum (kum-G), and its probability density function is defined as:

$$f(x|\theta, a, b) = abg(x)G(x)^{a-1}\left[1 - G(x)^a\right]^{b-1}$$

where θ is a vector of parameters from the density $g(x)$, $G(x)$ is the cumulative function, $a > 0$ and $b > 0$ are additonal parameters, with the role of introducing asymmetry and varying the weight of the tail.

From this definition of kum-G families, Cordeiro and Castro (2011) created various distributions stemmed from this generalization, as mentioned, but in this study we evaluated only its mixture to normal, called kum-normal and denoted by kumN, which has probability density function is given by:

$$f(x|\mu, \sigma, a, b) = \frac{ab}{\sigma}\, \phi\left(x\right)\left[\Phi\left(x\right)\right]^{a-1}\left[1 - \Phi\left(x\right)^a\right]^{b-1}, \tag{1}$$

where $x \in \mathbb{R}$, $a > 0$, $b > 0$, $\mu \in \mathbb{R}$, is the location parameter (mean) and $\sigma > 0$ is the scale parameter (standard deviation), $\phi(.)$ is the normal probability density function and $\Phi(.)$ is the distribution function.

A random variable with density $f(x)$ (1) is denoted as $X \sim kumN(\mu, \sigma, a, b)$. For $\mu = 0$ and $\sigma = 1$ we obtain the standard kum-normal distribution. Furthermore, kum-normal distribution with parameters $a = 2$ and $b = 1$ coincides with a normal distribution with asymmetric shape parameter $\lambda = 1$. It is noticed that both the kum-normal and Azzalini's skew normal have the same goal of modelling the tail, ie, each with their respective parameters of asymmetry tuning. For kum-normal two parameters are designated for the adjustment of asymmetry.

`Gamlss` is an R package with numerous functions and some ramifications. One of them is the `gamlss.dist`, where are all probability distribution adjusted by the package. It also provides the Akaike information criterion (AIC), the likelihood ratio test (LRT), Bayesian information criterion (BIC), likelihood function (LF) and graphical settings, among others.

The aim of this work is to implement the kum-normal distribution proposed by Cordeiro and Castro (2011) using the R package `gamlss` and verify the performance of kum-normal and Azzalini's skew normal on modelling asymmetry, using criteria information on the quality of fit and likelihood ratio test. For comparison, we used a real example of grain beetles (*Tribolium cofusum*) grown at 29°C (EUGENE; LEE; FAMOYE, 2002) and simulated data, generated from the distribution created by Tukey, called *g and h*, with increasing levels of asymmetry. Everything was done using R (R Development Core Team, 2011). The partial derivatives of the log-likelihood function of kum-normal were obtained for its implementation using the package `gamlss` (STASINOPOULOS; RIGBY, 2007).

In the second section we present the methodology used in this study, as well as the generalization of the normal distribution function to find the partial derivatives for implementation of kum-normal distribution, the real data and data from `g and h` distribution, tests for normality, information criteria and likelihood ratio test used to compare the distributions.

The results obtained with the implementation and comparison of the distributions are discussed in the third section, the conclusions presented in section four and the implementation of the kum-normal distribution is is the appendix.

Methodology

The kum-normal likelihood function is generated by the generalization of the generalized kum distributions proposed by Cordeiro e Castro (2011). They used, as base distribution, the probability density functions and cumulative functions.

Let θ be the p-dimensional vector of parameters. Let $X_1, \ldots, X_n$ random variables from the kum-normal distribution with vector of parameters $\theta = (a, b, \mu, \sigma)$. The log-likelihood function $S = ln[L(\theta)]$ is set as follows:

$$S_{KN} = n\{ln(a) + ln(b)\} + \sum_{i=1}^{n} ln[\phi(x_i)] + (a-1)\sum_{i=1}^{n} ln\left[\Phi(x_i)\right] + + (b-1)\sum_{i=1}^{n} ln\left[1 - \Phi(x_i)^a\right] \quad (2)$$

Then, the partial derivatives can be written as

$$\frac{\partial S_{KN}}{\partial a} = \frac{n}{a} + \sum_{i=1}^{n} ln\left[\Phi(x_i)\right]\left[1 - \frac{(b-1)\Phi(x_i)^a}{1-\Phi(x_i)^a}\right], \quad (3)$$

$$\frac{\partial S_{KN}}{\partial b} = \frac{n}{b} + \sum_{i=1}^{n} ln\left[1 - \Phi(x_i)^a\right], \tag{4}$$

$$\frac{\partial S_{KN}}{\partial \mu} = \sum_{i=1}^{n}\left[\frac{1}{\phi(x_i)}\frac{\partial \phi(x_i)}{\partial \mu} + \frac{1}{\Phi(x_i)}\frac{\partial \Phi(x_i)}{\partial \mu}\left(1 - \frac{a(b-1)}{\Phi(x_i)^{-a} - 1}\right)\right], \tag{5}$$

$$\frac{\partial S_{KN}}{\partial \sigma} = \sum_{i=1}^{n}\left[\frac{1}{\phi(x_i)}\frac{\partial \phi(x_i)}{\partial \sigma} + \frac{1}{\Phi(x_i)}\frac{\partial \Phi(x_i)}{\partial \sigma}\left(1 - \frac{a(b-1)}{\Phi(x_i)^{-a} - 1}\right)\right]. \tag{6}$$

From the equations above the maximum likelihood estimators can be obtained. Furthermore, partial derivatives of the kum-normal density are needed for programming it in the `gamlss` package. However, they are not straightforward obtaining, mainly due to the normal cumulative function. Eugene, Lee and Famoye (2002) used the following generalizations for such derivatives:

$$\frac{\partial \Phi(x)}{\partial \mu} = \frac{-\phi(x)}{\sigma}$$

and

$$\frac{\partial \Phi(x)}{\partial \sigma} = -\left(\frac{x-\mu}{\sigma^2}\right)\phi(x)$$

From them, all the second and mixed derivatives from kum-normal were derived.

The insertion of kum-normal distribution in package `gamlss` was performed substituting a distribution which contained the same number of parameters, following the guidelines of Rigby and Stasinopoulos (2007). Thus, kum-normal distribution can be part of the family of probability distributions implemented in the package, allowing use of resources already programmed.

Fitting

Fitting distributions was made in two situations. First random values were simulated from *h and g* distribution. This distribution has some facility generating asymmetric values, controlling a parameter.

The *g and h* distribution been suggested by John Wilder Tukey in 1977, and discussed by Hoaglin and Peters (1979) and Hoaglin (1983). It has a great flexibility in modelling, contains only two parameters (g, h), where g controls asymmetry and h models the weight of the tail. This distribution is defined in terms of the quantile function of the standard normal distribution:

$$f(Z|g,h) = \frac{exp(gZ-1)}{g} exp\left(\frac{h}{2}Z^2\right)$$

where Z is the p-th quantile of the standard normal distribution. Thus, $f(Z|g,h)$ is its probability density function with parameters $g \in (-1,1)$ and $h \in (0,1)$. A particular case for the *g and h* distribution with $g = 0$ and $h = 0$, is the standard normal distribution. The *g and h* density is derived taking the numeric derivative of the cumulative function.

Values generated from this distribution were split by degree of asymmetry. To generate such data, we used the function `rgh (random)` for generating vectors containing 100 random values. For each configuration, the `g` parameter was used as an asymmetry scale, set to the values: 0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9 and 1; and h was set to only two values, 0 and 0.2. Thus, 21 simulated cases were performed with 10 levels of asymmetry[ii].

[ii] The set $g = 0$ and $h = 0.2$ was not performed.

The second situation was fitting real data - regarding the number adults of *Tribolium cofusum*[iii], grown at 29^oC (CONSTANTINO; DESHARNAIS, 1981).

According to Constantino and Desharnais (1981), the experiment started placing a container with flour and some young adults beetles, who were kept in incubators in a range of 14 to 30 days. After that period the number of beetles and stages of life were computed. This procedure was repeated during a year and a half to two and a half years, until they reach an adult age or reaches a stationary phase. Thus, they were counted to generate such a frequency distribution that has a degree of asymmetry to the right. Eugene Lee and Famoye (2002) also used these same data for studies about asymmetric distributions.

Asymmetry tests and comparison criteria

To make the comparison between distributions, first we performed tests to verify the normality and symmetry to ensure that the simulated data from the *h and g* distribution - and the real data - were statistically asymmetric.

For such verification we performed the Shapiro-Wilk (FERREIRA, 2009) test, to confirm the data does not follow a normal distribution. Following test we used the test of D'Agostino (D'Agostino, 1978) that verifies if the asymmetry parameter is zero. Both are implemented in R, where the Shapiro-Wilk (`shapiro.test()`) is part of the basic package and test of D'Agostino (`agostino.test()`) is part of the package `moments`.

To compare the fittings we used the likelihood ratio test and to select the best fit we used AIC and BIC information criteria. The likelihood ratio test (CASELLA; BERGER, 2010) was performed to compare what distribution, kum-normal or Azzalini's skew normal, best fits the simulated data, according to the degree of asymmetry. Floriano et al. (2006) claim that the AIC (AKAIKE, 1973) and BIC (SCHWARZ, 1978) are the main criteria used in the comparison of models in computer programs. They are based on the likelihood value of the model and on the number of observations and the number of parameters thereof. It is assumed here that we have nested models, although BIC can also be used in comparison of non-nested models, according to Schwarz (1978).

Results and discussion

Second and mixed derivatives of kum-normal distribution were obtained from the log-likelihood function (2) proposed by Cordeiro and Castro (2011). Those derivatives are of great importance, since they are needed in the construction of kum-normal distribution and likelihood, through the their implementation in the package `gamlss.dist`. Due to notation facility, we cal $\phi(x) = \phi$ the normal probability density function and $\Phi(x) = \Phi$ the normal cumulative function.

$$\frac{\partial S^2_{(KN)}(\theta)}{\partial \mu^2} = \left(\frac{-1}{\sigma^2}\right) - \left[\frac{\Phi\phi + \phi^2}{(\Phi\sigma)^2}\right] + \left[\frac{(\Phi\sigma)(\Phi^{-a}-1)(ab-a)\left(\frac{x-\mu}{\sigma^2}\right)\phi}{[(\Phi\sigma)(\Phi^{-a}-1)]^2}\right] - \\ - \left[\frac{\phi(ab-a)(\phi - \phi\Phi^{-a} - \sigma a \Phi^{-a})}{[(\Phi\sigma)(\Phi^{-a}-1)]^2}\right]. \quad (7)$$

$$\frac{\partial S^2_{(KN)}(\theta)}{\partial \sigma^2} = \left[\frac{1}{\sigma^2}\right] - \left[\frac{(\Phi\sigma)(x-\mu)\phi\left(\frac{(x-\mu)^2}{\sigma^2} - 1\right) + 2\phi\Phi\sigma(x-\mu) - \phi^2(x-\mu)^2}{(\Phi\sigma^2)^2}\right] +$$

[iii]Beetles that attack stored grains.

$$+\left[\frac{-3(x-\mu)^2}{\sigma^4}\right]+\left[\frac{(\Phi\sigma^2)(\Phi^{-a}-1)(ab-a)(x-\mu)\phi\left(\frac{(x-\mu)^2}{\sigma^3}-\frac{1}{\sigma}\right)}{\left[(\Phi\sigma^2)(\Phi^{-a}-1)\right]^2}\right]$$

$$-\left[\frac{\phi(x-\mu)(ab-a)(2\sigma\Phi^{1-a}-\Phi^{-a}\phi(x-\mu)-a\sigma^2\Phi^{-a}-2\Phi\sigma+(x-\mu)\phi)}{\left[(\Phi\sigma^2)(\Phi^{-a}-1)\right]^2}\right]. \qquad (8)$$

$$\frac{\partial S^2_{(KN)}(\theta)}{\partial\mu\partial\sigma}=\left[\frac{-2(x-\mu)}{\sigma^3}\right]-\left[\frac{\Phi\phi\left(\frac{(x-\mu)^2}{\sigma^2}-1\right)\phi\Phi+\phi^2\left(\frac{x-\mu}{\sigma}\right)}{(\Phi\sigma)^2}\right]+$$

$$+\left[\frac{(\Phi\sigma)(\Phi^{-a}-1)(ab-a)\phi\left(\frac{(x-\mu)^2}{\sigma^3}-\frac{1}{\sigma}\right)}{\left[(\Phi\sigma)(\Phi^{-a}-1)\right]^2}\right]-$$

$$-\left[\frac{\phi(ab-a)\left[\Phi^{-a}\left(\Phi-\frac{\phi(x-\mu)}{\sigma}\right)-a\sigma\Phi^{-a}-\Phi+\phi\frac{(x-\mu)}{\sigma}\right]}{\left[(\Phi\sigma)(\Phi^{-a}-1)\right]^2}\right]. \qquad (9)$$

$$\frac{\partial S^2_{(KN)}(\theta)}{\partial\mu\partial a}=\left[\frac{(\Phi\sigma)(\Phi^{-a}-1)\phi(b-1)+\phi(ab-a)(\Phi^{1-a})\sigma\ln(\Phi)}{\left[(\Phi\sigma)(\Phi^{-a}-1)\right]^2}\right]. \qquad (10)$$

$$\frac{\partial S^2_{(KN)}(\theta)}{\partial\mu\partial b}=\frac{\phi a}{(\Phi\sigma)(\Phi^{-a}-1)}. \qquad (11)$$

$$\frac{\partial S^2_{(KN)}(\theta)}{\partial\sigma\partial a}=\frac{(\Phi\sigma^2)(\Phi^{-a}-1)\phi(x-\mu)(b-1)+\phi(x-\mu)(ab-a)(\Phi^{1-a})\ln(\Phi)\sigma^2}{\left[(\Phi\sigma^2)(\Phi^{-a}-1)\right]^2}. \qquad (12)$$

$$\frac{\partial S^2_{(KN)}(\theta)}{\partial\sigma\partial b}=\frac{\phi a(x-\mu)}{(\Phi\sigma^2)(\Phi^{-a}-1)}. \qquad (13)$$

$$\frac{\partial S^2_{(KN)}(\theta)}{\partial a^2}=\left(\frac{-1}{a^2}\right)-\ln(\Phi)\left[\frac{(b-1)\ln(\Phi)\Phi^a}{(1-\Phi^a)}+\frac{(b-1)(\Phi^{2a})\ln(\Phi)}{(1-\Phi^a)^2}\right]. \qquad (14)$$

$$\frac{\partial S^2_{(KN)}(\theta)}{\partial a\partial b}=-\frac{\ln(\Phi)\Phi^a}{1-\Phi^a}. \qquad (15)$$

$$\frac{\partial S^2_{(KN)}(\theta)}{\partial b^2}=-\frac{1}{b^2}. \qquad (16)$$

The implementation of kum-normal was performed in the package *gamlss* replacing a distribution of four parameters already implemented. To use the distribution of its properties just load the package functions `rKumN`, `qKumN`, `dKumN` and `pKumN` for random number generation, quantile function, density function and probability function, respectively. See Appendix for R routines.

Figure 1 shows the probability distribution function of kum-normal, for a scene with parameters $\mu = 10$, $\sigma = 2$, $a = 2$ and $b = 1$, showing the shape of the distribution in according to the implemented functions.

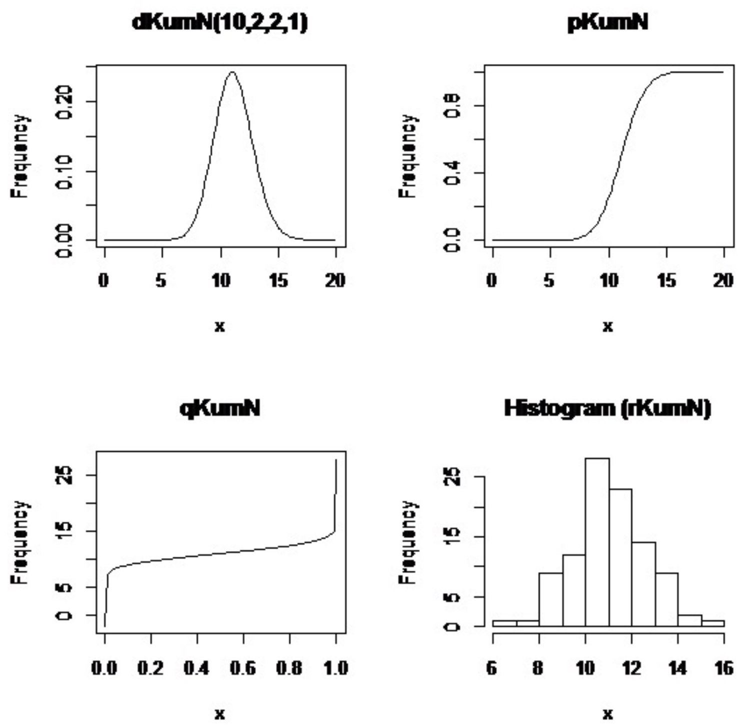

Figure 1: Kum-Normal distribution, $\mu = 10, \sigma = 2, a = 2, b = 1$.

Comparing distributions

Data simulated from the *h and g* distribution, setting $h = 0$ and varying the parameter g, were exposed to Shapiro-Wilk and D'Agostino tests. The p-values and estimates of asymmetry are shown in Table 1.

Observing Table 1 we note that the p-values in scene $r0$, for Shapiro-Wilk and D'Agostino tests, are greater than 5%, ie the simulated data follow a normal distribution and therefore are symmetrical. It happens due the definition of *h and g* distribution under the such parametric values. For the other scenarios, the p-values are less the level of significance for both tests, characterizing asymmetric distributions. It is worth noting also the increase of the degree of asymmetry while increases the g parameter.

Table 2 presents the results of information criteria and likelihood ratio test for each simulated scenario, for both kum-normal and Azzalini's skew normal.

Note in 2 that in the first two scenarios the likelihood ratio test confirms the equality of the fitting for both models. That occurred because, in those scenes, the asymmetry was not well

Table 1: Results for the normality tests and estimates of asymmetry of the data simulated from the *g and h* distribution

Scenes	Shapiro-Wilk p-value	D'Agostino p-value	Asymmetry level
r0(0.0; 0)	6.7×10^{-1}	9.8×10^{-1}	0.007
r1(0.1; 0)	8.0×10^{-3}	4.5×10^{-2}	0.771
r2(0.2; 0)	4.5×10^{-12}	1.0×10^{-2}	1.056
r3(0.3; 0)	1.2×10^{-5}	4.0×10^{-2}	1.225
r4(0.4; 0)	2.3×10^{-8}	3.0×10^{-4}	1.687
r5(0.5; 0)	1.2×10^{-8}	2.0×10^{-4}	1.784
r6(0.6; 0)	9.2×10^{-11}	1.0×10^{-4}	1.887
r7(0.7; 0)	2.8×10^{-10}	9.4×10^{-5}	1.940
r8(0.8; 0)	1.2×10^{-12}	2.2×10^{-5}	2.235
r9(0.9; 0)	1.3×10^{-13}	4.1×10^{-6}	2.596
r10(1.0; 0)	3.3×10^{-13}	1.9×10^{-6}	2.765

Table 2: Results of the information criteria (AIC and BIC) and likelihood ratio test (LRT) for kum-normal (KumN) and asymmetric normal (AN)

Scenes	Distribution	AIC	BIC	LRT p-value
r0(0.0; 0)	KumN	304.53	314.95	0.259
	AN	303.81	311.63	
r1(0.1; 0)	KumN	277.30	287.72	0.532
	AN	275.69	283.50	
r2(0.2; 0)	KumN	299.62	310.04	0.014
	AN	303.63	311.45	
r3(0.3; 0)	KumN	270.06	280.48	0.039
	AN	272.32	280.12	
r4(0.4; 0)	KumN	357.37	397.79	0.006
	AN	362.79	370.61	
r5(0.5; 0)	KumN	325.53	335.95	0.011
	AN	330.01	337.82	
r6(0.6; 0)	KumN	377.90	388.32	0.008
	AN	382.76	390.57	
r7(0.7; 0)	KumN	321.86	332.28	0.007
	AN	327.20	335.08	
r8(0.8; 0)	KumN	409.49	419.91	0.003
	AN	416.37	424.19	
r9(0.9; 0)	KumN	402.32	412.74	0.012
	AN	406.66	414.47	
r10(1.0; 0)	KumN	347.35	357.77	1.6×10^{-12}
	AN	395,24	403,06	

established. Despite the test of D'Agostino characterize the presence of asymmetry parameter (table 1) as significant in scenario $r1$ the distributions also showed no significant differences in adjustment, which confirms similar fitting in situations of none or low degree of asymmetry.

Regarding the other scenarios, the likelihood ratio test detected difference in fitting distributions to the data. The information criteria were used to determine the best fitting model.

Note that the values of AIC for kum-normal are less than the values for the asymmetric normal in all scenarios. Therefore the kum-normal best fitted all cases where the degree of asymmetry was higher. There was some disagreement in the scenarios $r3$, $r4$, $r12$ and $r14$ in relation to the information criteria, but preference was given to the AIC results.

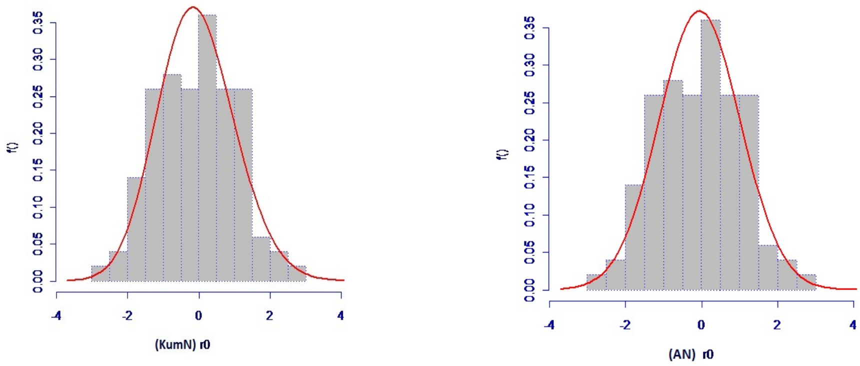

Figure 2: Graphic representations under estimated asymmetry of 0.007 for KumN (-0.87; 0.15; 0.58; -0.27) and AN (-0.08; 0.07; 0.03).

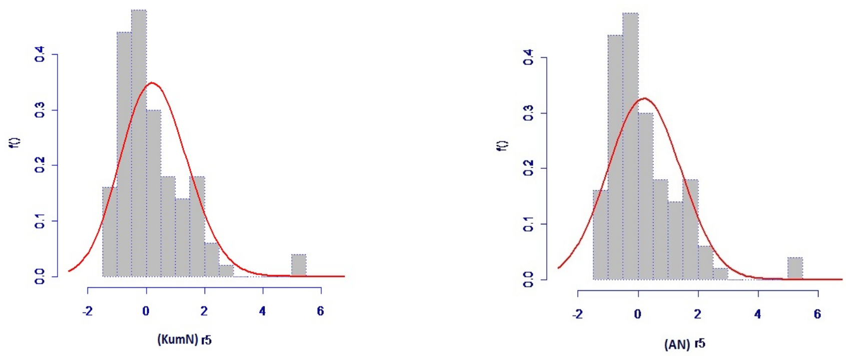

Figure 3: Graphic representations under estimated asymmetry of 0.771 for KumN (-0.36; 6.7×10^{-6}; 0.16; $-0, 55$) and AN (0.22; 0.20; -0.01).

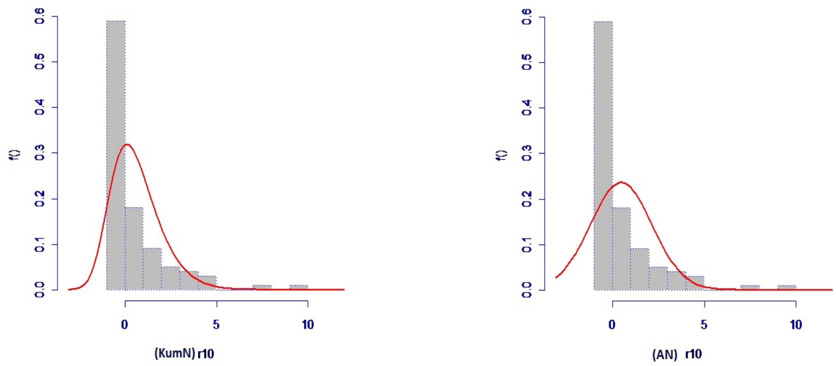

Figure 4: Graphic representations under estimated asymmetry of 0.0839 for KumN (-3.65; 0.64; 3.05; -0.60) and AN (0.49; 0.52; -0.01).

Figures 2, 3 and 4 show the graphs for scenes $r0$, $r5$ and $r10$. The histogram displays the simulated values and the density, adjusted according to the estimated parameters. It appears that the adjustment for the two simulated cases exhibit similar behaviours. Visually we can see a slight difference in the estimation of the mean. Kum-normal presents lower estimates for the average while Azzalini's skew normal presents characteristics of a standard normal, ie estimate of λ close to zero.

Table 3 brings the test results to verify the simulated data with respect to normality and asymmetry.

These data were generated with h set to 0.2, simulating a distribution with heavier tails, differentiating a little bit more the asymmetric form.

Table 3: Results for the normality tests and estimates of asymmetry of the data simulated from the *g and h* distribution

Scenes	Shapiro-Wilk p-value	D'Agostino p-value	Asymmetry level
r11(0.1; 0.2)	2.0×10^{-3}	3.7×10^{-2}	0.839
r12(0.2; 0.2)	4.0×10^{-4}	2.2×10^{-2}	0.905
r13(0.3; 0.2)	2.0×10^{-6}	3.0×10^{-3}	1.240
r14(0.4; 0.2)	3.9×10^{-8}	1.0×10^{-3}	1.350
r15(0.5; 0.2)	4.0×10^{-7}	9.0×10^{-4}	1.500
r16(0.6; 0.2)	7.3×10^{-9}	4.0×10^{-4}	1.630
r17(0.7; 0.2)	4.9×10^{-9}	3.0×10^{-4}	1.720
r18(0.8; 0.2)	4.3×10^{-10}	2.0×10^{-4}	1.800
r19(0.9; 0.2)	1.6×10^{-8}	2.0×10^{-4}	1.820
r20(1.0; 0.2)	3.0×10^{-12}	2.0×10^{-4}	1.840

Results of the Shapiro-Wilk test presented in Table 3 indicate that the null hypothesis that the data follow a normal distribution should be rejected, with p-values ranging from 4.3×10^{-10} to 0.002. The test of D'Agostino also presents significance, indicating that the data follow a skewed distribution. P-values vary from 0.0009 to 0.037.

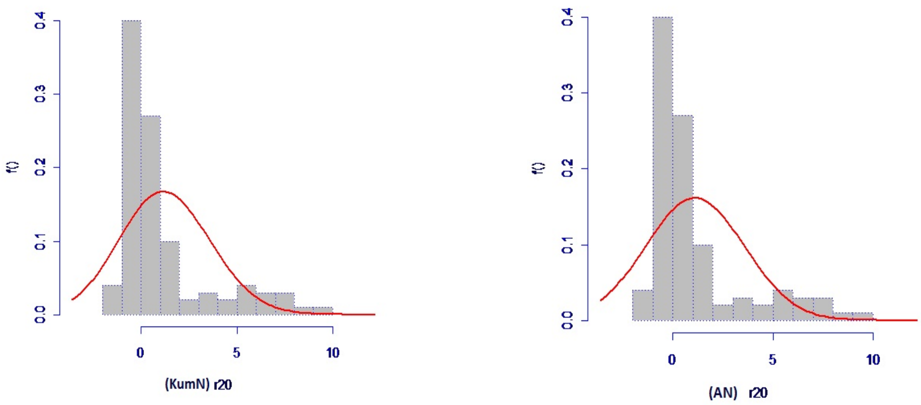

Figure 5: Graphic representations under estimated asymmetry of 1.84 for KumN (0.31; 0.96; 0.37; −0.06) and AN (1.18; 0.90; −0.04).

Figure 5 presents the histogram and the fit of each distribution to the data, with increasing values of asymmetry. The fittings were considered different, where the kum-normal distribution has the best fit. It can be seen that although the kum-normal presents the a better fit there are limitations on the shape of both distributions. In a visual evaluation, the dependence of normality seems to prevent the distributions to follow the format of the histogram peaks. In this study we did not consider the quality of fit but the comparison between them.

Table 4 shows the result of the comparison between distributions, highlighting the information criteria and likelihood ration test results.

Table 4: Comparison between distributions kum-normal (KumN) and Azzalini's asymmetric normal (AN) for scenes from $r11$ to $r20$

Scene	Distribution	AIC	BIC	LRT p-value
r11(100; 0.1; 0.2)	KumN	283.332	293.753	0.019
	AN	286.809	294.624	
r12(100; 0.2; 0.2)	KumN	316.109	326.529	0.039
	AN	318.351	326.167	
r13(100; 0.3; 0.2)	KumN	368.655	379.076	0.021
	AN	371.947	379.763	
r14(100; 0.4; 0.2)	KumN	349.409	359.829	0.033
	AN	351.953	359.768	
r15(100; 0.5; 0.2)	KumN	366.629	377.05	0.011
	AN	371.096	378.912	
r16(100; 0.6; 0.2)	KumN	387.885	398.306	0.006
	AN	393.481	401.297	
r17(100; 0.7; 0.2)	KumN	353.051	363.472	0.003
	AN	360.177	367.993	
r18(100; 0.8; 0.2)	KumN	430.477	440.898	0.009
	AN	435.269	443.084	
r19(100; 0.9; 0.2)	KumN	398.582	409.002	0.010
	AN	403.139	410.954	
r20(100; 1.0; 0.2)	KumN	467.846	478.266	0.028
	AN	470.689	478.505	

It can be seen in Table 4 that the p-values range from 0.003 to 0.039, confirming the difference between distributions' fittings. Thus, according to the likelihood ratio test, we can say that the fittings are different between the distributions, for those scenes, at 5% of significance. Analyzing the information criteria, the values of the AIC for kum-normal values are smaller than the one for Azzalini's asymmetric normal, in all cases. So kum-normal distribution fitted better for scenes from $r11$ to $r20$.

Regarding the comparison between the distributions, the kum-normal best fit at 90.9% of the simulated cases. Only in the scenarios $r0$ and $r1$ - with a low degree of asymmetry - the Azzalini's skew normal performed best.

Real data modelling

The real data used here refer to the number of adults of *Tribolium cofusum* grown at 29°C. Such data set is asymmetric, with a long tail to the right.

Table 5 presents the results of Shapiro-Wilk and D'Agostino tests and the asymmetry level of the data.

It can be seen in Table 5 that the p-values are all significant at 5%, confirming the lack of normality and symmetry, indicating an asymmetry to the right of 0.8.

Table 6 presents the results for the information criteria and likelihood ratio test.

Table 5: Results for the normality tests and estimates of asymmetry of the real data set

Shapiro-Wilk p-value	D'Agostino p-value	Asymmetry level
8.04×10^{-16}	3.6×10^{-7}	0.80

Table 6: Comparing the distributions in relation to the real data

Distribution	AIC	BIC	LRT p-value
KumN	7266.54	7284.68	0.225
AN	7266.01	7279.62	

Analysing Table 6, through the likelihood ratio test, the hypothesis of equality of fittings should not be rejected at 5% of significance. Thus, distributions showed the same fitting capacity. In the simulated case $r2$ (Table 2), the estimation of asymmetry was also 0.8, and the result confirms the presented adjustment.

From Figure 6 we can see similar fittings and similar parameter estimates from both distributions.

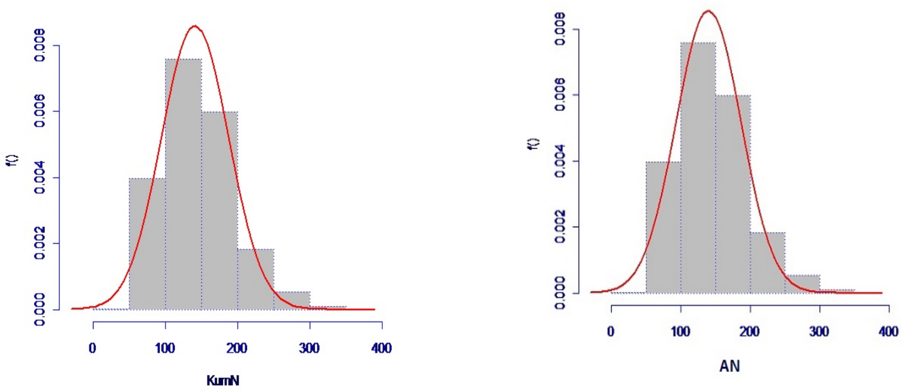

Figure 6: Graphic representations under estimated asymmetry of 0.80 for KumN (139; 3.8; 0.04; −0.014) and AN (139.7; 3.8; 0.004).

Conclusions

Second and mixed derivatives were obtained enabling the implementation of kum-normal distribution in the package `gamlss R`.

The kum-normal distribution proved to be effective in adjusting asymmetric data as much as the Azzalini's skew normal distribution. As the level of asymmetry increases the kum-normal distribution shows better fitting than the asymmetric normal.

For the real data, distribution kum-normal and normal asymmetric had the same fitting quality.

References

AKAIKE, H. Information theory and an extension of the maximum likelihood principle. International Symposium on Information Theory, 1973, Budapest. *Proceeding of...* Budapest: Academiai Kiado. 1973. pp.267-281.

AZZALINI, A. A class of distributions which the normal ones. *Journal Statistical*,v.12, n.2, p.171-178, 1985.

CASELLA, G.; BERGER, R. L. *Inferência Estatística.* São Paulo, 2010, 588p.

CONSTANTINO, R. F.; DESHARNAIS, R. A. Gamma distributions of adult numbers for tribolium populations in the region of their steady states. *Journal of Animal Ecology*, v.50, p.667-681, 1981.

CORDEIRO, G. M.; CASTRO, M. A new family of generalized distributions. *Journal of Statistical Computation & Simulation*, v.81, n.7, p.883-898, 2011.

D'AGOSTINO, R. B. Transformation to Normality of the Null. *Biometrika*, v.57, n.3, p.679-681, 1978.

EUGENE, N.; LEE, C.; FAMOYE, F. Beta-normal distribution and its applications. *Communications in Statistics - Theory and Methods*, Michigan, USA, v.31, p.497-512, 2002.

FERREIRA, D.F. *Estatítica Básica.* Lavras: UFLA, 2009, 664p.

HOAGLIN, D. C.; PETERS, S. C. Software for Exploring Distributional Shapes. IN: *Proceedings of...* Computer Science and Statistics: 12th Annual Symposium on the Interface, Ontario, Canada: Iniverty of waterloo, p. 418-443, 1979.

HOAGLIN, D. C. Summarizing shape numerically: *The g-and-h Distributions*, in Exploring Data, tables, Trends and Shapes. New York: Wiley, p.461-513, 1983.

JONES, M. C. Families of distributions arising from distributions of order statistics. *Test*, vol 13, n.1, p.1-43, 2004.

JONES, M. C. Kumaraswamy's distribution: A beta-type distribution with some tractability advantages. *Statistical Methodology*, v.6 , p.70-81, 2008.

KUMARASWAMY, P. Generalized probability densy-function for double-bounded random-process. *Journal of Hydrology*, v.462, p.79-88, 1980.

OLIVEIRA, M. S. *Comparações múltiplas Baysianas com erro normal assimétrico.* 2009, 154f. Tese (Doutorado em Estatítica e Experimentação Agropecuária), Departamento de Ciências Exatas, Universidade Federal de Lavras, Lavras, 2009.

R DEVELOPMENT CORE TEAM. *R*: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria, 2011. ISBN 3-900051-07-0, URL http://www.R-project.org/.

SCHWARZ, G. Estimating the dimension of a model. In: STATISTICS, 6, n.2, 1978, *Annals...* Ann Arbor: Institute of Mathematical Statistics, 1978. pp.461-464.

STASINOPOULOS, D. M.; RIGBY, R. A. Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, v.23, p.1-46, 2007.

TUKEY, J. W. Modern Techniques in Data Analysis. *Proceeding of...* NSF-Sponsored Regional Research Conference at Southeastern Massachusetts University, North Dartmouth, MA, 1977.

Appendix

Kum-normal distribution source code

```
library("gamlss")
KumN = function(mu.link="identity", sigma.link="log", nu.link ="log",
tau.link="log"){
mstats=checklink("mu.link","kumaraswamy-Normal",substitute(mu.link),
        c("$1/mu^2$", "log", "identity"))
dstats = checklink("sigma.link", "kumaraswamy-Normal", substitute
      (sigma.link), c("inverse", "log", "identity"))
vstats=checklink("nu.link","kumaraswamy-Normal",substitute(nu.link),
        c("$1/nu^2$", "log", "identity"))
tstats=checklink("tau.link","kumaraswamy-Normal",substitute(tau.link),
        c("$1/tau^2$", "log", "identity"))
structure (
list(family = c("KumN", "kumaraswamy-Normal"),
parameters = list(mu=TRUE, sigma=TRUE, nu=TRUE, tau=TRUE),
nopar = 4,
type = "Continuous",
mu.link = as.character(substitute(mu.link)),
sigma.link = as.character(substitute(sigma.link)),
nu.link = as.character(substitute(nu.link)),
tau.link = as.character(substitute(tau.link)),
mu.linkfun = mstats$linkfun,
sigma.linkfun = dstats$linkfun
nu.linkfun = vstats$linkfun,
tau.linkfun = tstats$linkfun,
mu.linkinv = mstats$linkinv,
sigma.linkinv = dstats$linkinv,
nu.linkinv = vstats$linkinv,
tau.linkinv = tstats$linkinv,
mu.dr = mstats$mu.eta,
sigma.dr = dstats$mu.eta,
nu.dr = vstats$mu.eta,
tau.dr = tstats$mu.eta,
```

#**dldm**: First derivative of eq. (2) in relation to μ (eq. 5)

```
dldm = function(y, mu, sigma, nu, tau) {
   d = dnorm(y, mu, sigma)
   p = pnorm(y, mu, sigma)
   n = (nu*tau-nu)
   dfdm = (d*(y-mu)/sigma^2)
   dpdm = (-d/sigma)
   dldm = ((1/d)*( dfdm)+(1/p)*(dpdm)*(1-(n/((p^(-nu))-1))))
},
```

#**d2ldm2**: Second derivative of eq. (2) in relation to μ (eq. 7)

```
d2ldm2 = function(y, mu, sigma, nu, tau){
   d = dnorm(y, mu, sigma)
   p = pnorm(y, mu, sigma)
   n = (nu*tau-nu)
   C = ((p*sigma*((p^(-nu))-1))^2)
   A = ((sigma*p*((p^(-nu))-1)*n*(d*(y-mu)/(sigma^2)))/ C )
   B = ((d*n*(d-d*(p^(-nu))-sigma*nu*(p^(-nu)))))/ C )
   d2ldm2=(-1/sigma^2)-((p*d*((y-mu)/sigma)+d^2)/((p*sigma)^2))+A-B
},
```

#**dldd**: First derivative of eq. (2) in relation to σ (eq. 6)

```
dldd = function(y, mu, sigma, nu, tau) {
   d = dnorm(y, mu, sigma)
   p = pnorm(y, mu, sigma)
   n = (nu*tau-nu)
   dpdd = ((-(y-mu)*d)/(sigma^2))
   dfdd = (d*((((y-mu)^2)/sigma^3)-(1/sigma)))
   dldd = (1/d)*(dfdd)+((1/p)*dpdd*(1-(n/((p^(-nu))-1))))
},
```

#**d2ldd2**: Second derivative of eq. (2) in relation to σ (eq. 8)

```
d2ldd2 = function(y, mu, sigma, nu, tau) {
   p=pnorm(y, mu, sigma)
   d=dnorm(y, mu, sigma)
   n=( nu*tau - nu)
   G=(((y-mu)^2)/sigma^3)
   E=(p*(sigma^2)*((p^(-nu))-1)*n*(y-mu)*d*(G-(1/sigma)))
   A=E/(((p*(sigma^2))*((p^(-nu))-1))^2)
   H=((p^(-nu))*d*(y-mu))
   B=(d*n*(y-mu)*((2*sigma*(p^(1-nu)))-H-(nu*(sigma^2)*(p^{(-nu)}))))-F)
   F=((2*sigma*p)+((y-mu)*d))
   C=(((p*(sigma^2))*((p^(-nu))-1))^2)
   D=(((2*d*(y-mu)*sigma*p)-((d^2)*((y-mu)^2)))/((p*(sigma^2))^2))
   G=((p*sigma*(y-mu)*d*((((y-mu)^2)/sigma^2)-1))/(p*(sigma^2))^2)
   d2ldd2=(-3*((y-mu)^2)/sigma^4)+(1/sigma^2)-G+D+A-B/C
},
```

#**dldv**: First derivative of eq. (2) in relation to a (eq. 3)

```
dldv = function(y, mu, sigma, nu, tau) {
   p = pnorm(y, mu, sigma)
   dldv = (1/nu)+(\log(p))*(1-(((tau-1)*(p^nu))/(1-(p^nu))))
},
```

#**dldv2**: Second derivative of eq. (2) in relation to a (eq. 14)

```
d2ldv2 = function(y, mu, sigma, nu, tau) {
   p = pnorm(y, mu, sigma)
   A = (((tau-1)*\log(p)*(p^nu))/(1-(p^nu)))
   B = (((tau-1)*\log(p)*(p^(2*nu)))/(1-(p^nu))^2)
   d2ldv2 = (-1/(nu^2))-(\log(p))*( A + B)
},
```

#**dldt**: First derivative of eq. (2) in relation to b (eq. 4)

```
dldt = function(y, mu, sigma, nu, tau) {
   p  = pnorm(y, mu, sigma)
   dldt = (1/tau)+\log(1-(p^nu))
},
```

#**d2dt**: Second derivative of eq. (2) in relation to b (eq. 16)

```
d2ldt2 = function(y, mu, sigma, nu, tau) {
   d2ldt2 = (-(1/(tau^2)))
},
```

#**d2ldmdd**: Mixed derivative of eq. (2) in relation to μ and σ (eq. 9)

```
d2ldmdd = function(y, mu, sigma, nu, tau){
   d=dnorm(y, mu, sigma)
   p=pnorm(y, mu, sigma)
   n=(nu*tau-nu)
   E=(d*n*((p^(-nu))
   A=E*(p-((y-mu)/sigma)*d-((p^(-nu))*sigma*nu)-p+((y-mu)/sigma)*d)
   F=((p*sigma*((p^(-nu))-1))^2)
   B=((p*sigma*((p^(-nu))-1)*n*d*((((y-mu)^2)/(sigma^3))-(1/sigma)))/F)
   C=(A/(((p*sigma)*((p^(-nu))-1))^2))
   D=(-2*(y-mu)/sigma^3)
   G=((((y-mu)^2)/sigma^2)-1)
   d2ldmdd=D-((p*d*G-(d*p)+(d^2)*((y-mu)/sigma))/(p*sigma)^2)+B-C
},
```

#**d2ldmdv**: Mixed derivative of eq. (2) in relation to μ and a (eq. 10)

```
d2ldmdv = function(y, mu, sigma, nu, tau){
   d = dnorm(y, mu, sigma)
   p = pnorm(y, mu, sigma)
   n = (nu*tau-nu)
   A = (p*sigma*((p^(-nu))-1)*d*(tau-1))
d2ldmdv=(A+(d*n*sigma*\log(p)*(p^(1-nu))))/((p*sigma*((p^{(-nu)})-1))^2)
},
```

#**d2ldmdt**: Mixed derivative of eq. (2) in relation to μ and b (eq. 11)

```
d2ldmdt = function(y, mu, sigma, nu, tau){
   d = dnorm(y, mu, sigma)
   p = pnorm(y, mu, sigma)
   d2ldmdt = (nu*d)/(p*sigma*((p^(-nu))-1))
},
```

#**d2ldddv**: Mixed derivative of eq. (2) in relation to σ and a (eq. 12)

```
d2ldddv = function(y, mu, sigma, nu, tau) {
   p = pnorm(y, mu, sigma)
   n = (nu*tau-nu)
   d = dnorm(y, mu, sigma)
   A = (p*(sigma^2)*((p^(-nu))-1)*d*(y-mu)*(tau-1))
   B = ((p*(sigma^2)*((p^(-nu))-1))^2)
   d2ldddv = (A+(d*(y-mu)*n*(p^(1-nu))*\log(p)*(sigma^2)))/B
},
```

#**d2ldddt**: Mixed derivative of eq. (2) in relation to σ and b (eq. 13)

```
d2ldddt = function(y, mu, sigma, nu, tau) {
   p = pnorm(y, mu, sigma)
   d = dnorm(y, mu, sigma)
   n = (nu*tau-nu)
   d2ldddt = (d*(y-mu)*nu)/(p*(sigma^2)*((p^(-nu))-1))
},
```

#**d2ldvdt**: Mixed derivative of eq. (2) in relation to a and b (eq. 15)

```
d2ldvdt = function(y, mu, sigma, nu, tau){
   p = pnorm(y, mu, sigma)
   d2ldvdt = (-(\log(p)*(p^(nu)))/(1-(p^(nu))))
},
G.dev.incr = function(y, mu, sigma, nu, tau,...)
   -2*dKumN(y, mu, sigma, nu, tau, \log=TRUE),
   rqres = expression (
   rqres(pfun = "pKumN", type="Continuous", y=y, mu=mu,
   sigma=sigma, nu=nu, tau=tau) ),
   mu.initial = expression(mu = (y+mean(y))/2),
   sigma.initial = expression(sigma = rep(sd(y), length(y))),
   nu.initial = expression(nu = rep(1, length(y))) ,
   tau.initial = expression(tau = rep(0.5, length(y))) ,
   mu.valid = function(mu) TRUE,
   sigma.valid = function(sigma) all(sigma > 0),
   nu.valid = function(nu) all(nu > 0),
   tau.valid = function(tau) all(tau > 0),
   y.valid = function(y) TRUE),
   class = c("gamlss.family","family")) }
```

Kum-normal probability density function

```
dKumN=function(y, mu=0, sigma=1, nu=2, tau=1, \log=FALSE) {
if(any(sigma<0)) stop(paste("sigma must be positive", "\ n", ""))
if(any(tau<0)) stop(paste("tau must be positive", "\ n", ""))
if(any(nu<0)) stop(paste("nu must be positive", "\ n", ""))
p=pnorm(y, mu, sigma)
f = (1/(sqrt(2*pi*sigma^2))*exp(-0.5*((y-mu)/sigma)^2))
loglik=\log(nu)+\log(tau)+\log(f)+(nu-1)*\log(p)+(tau-1)*\log(1-(p^(nu)))
if(log==FALSE) ft = exp(loglik) else ft = loglik
ft
}
```

Kum-normal distribution function

```
pKumN=function(q, mu=0, sigma=1, nu=2, tau=1,
lower.tail = TRUE, \log.p = FALSE) {
if(any(sigma<0)) stop(paste("sigma must be positive", "\ n", ""))
if(any(tau<0)) stop(paste("tau must be positive", "\ n", ""))
if(any(nu<0)) stop(paste("nu must be positive", "\ n", ""))
p=1-(1-(pnorm(q, mu, sigma))^nu)^tau
if(lower.tail==TRUE) p = p else p = 1-p
if(log.p==FALSE) p = p else p=- \log(p)
p
}
```

Kum-normal quantile function

```
qKumN=function(p, mu=0, sigma=1, nu=2, tau=1,lower.tail=TRUE,
log.p=FALSE, lower.limit=mu-10*(sigma/(nu*tau)),
upper.limit = mu+10*(sigma/(nu*tau)) ){
h1 = function(q) {
pKumN(q, mu=mu[i], sigma=sigma[i], nu=nu[i], tau=tau[i])-p[i]}
h = function(q) {
pkumN(q, mu=mu[i], sigma=sigma[i], nu=nu[i], tau=tau[i])}
if(any(sigma<=0))
stop(paste("sigma must be positive", "\ n", ""))
if(\log.p == TRUE)
p = exp(p)
else p = p
if(lower.tail==TRUE)
p = p
else p = 1 - p
if (any(p < 0)  any(p > 1))
stop(paste("p must be between 0 and 1", "\ n", ""))
lp = max(length(p), length(mu), length(sigma), length(nu), length(tau))
p = rep(p, length = lp)
sigma = rep(sigma, length = lp)
mu = rep(mu, length = lp)
nu = rep(nu, length = lp)
tau = rep(tau, length = lp)
q = rep(0, lp)
for(i in seq(along = p)) {
```

```
if(h(mu[i]) < p[i]) {
interval = c(mu[i],mu[i] + sigma[i])
j = 2
while (h(interval[2]) < p[i]) {
interval[2] = mu[i] + j *sigma[i]
j = j + 1 }
}
else {
interval = c(mu[i] - sigma[i], mu[i])
j = 2
while(h(interval[1]) > p[i]) {
interval[1] = mu[i] - j * sigma[i]
j = j + 1$ } }
q[i] = uniroot(h1, interval) \ root
} q }
```

Kum-normal random data generating function

```
rKumN=function(n, mu=0, sigma=1, nu=2, tau=1) {
if(any(sigma<=0)) stop(paste("sigma must be positive", "\ n", ""))
if(any(tau<0)) stop(paste("tau must be positive", "\ n", ""))
if(any(nu<0)) stop(paste("nu must be positive", "\ n", ""))
n = ceiling(n)
p = runif(n)
r = qKumN(p, mu=mu, sigma=sigma, nu=nu, tau=tau)
}
```