A new family of continuous distributions: properties and applications

Gauss M. Cordeiro[a] and Maria do Carmo[a]

Resumo

This article introduces generalized beta-generated (GBG) distributions. Sub-models include all classical beta-generated, Kumaraswamy-generated and exponentiated distributions. They are maximum entropy distributions under three intuitive conditions, which show that the classical beta generator skewness parameters only control tail entropy and an additional shape parameter is needed to add entropy to the center of the parent distribution. This parameter controls skewness without necessarily differentiating tail weights. The GBG class also has tractable properties: we present various expansions for moments, generating function and quantiles. The model parameters are estimated by maximum likelihood and the usefulness of the new class is illustrated by means of some real data sets.

Key Words:

[a] Departamento de Estatística, Universidade Federal de Pernambuco, Brazil.
e-mail:*gausscordeiro@de.ufpe.br, maria@de.ufpe.br*

1 Introduction

There has been an increased interest in developing generalized families of distributions by introducing additional shape parameters to a baseline cumulative distribution. This mechanism has proved to be useful to make the generated distributions more flexible especially for studying tail properties than existing distributions and for improving their goodness-of-fit statistics to the data under study.

Let $G(x)$ be the cumulative distribution function (CDF) of a baseline distribution and $g(x) = dG(x)/dx$ be the associated probability density function (PDF) depending on a parameter vectot η. We present a generalized family with two additional shape parameters by transforming the CDF $G(x)$ according to two sequential important generators. These families are important for modeling data in several engineering areas. Many special distributions in these families are discussed by Tahir and Nadarajah (2015).

Marshall and Olkin (1997) pioneered a general method to expand a distribution G by adding an extra shape parameter. The CDF of their family (for $\theta > 0$) is

$$F_{\text{MO-G}}(x) = \frac{G(x)}{\theta + (1-\theta)G(x)} = \frac{G(x)}{1-(1-\theta)[1-G(x)]}, \quad x \in \mathbb{R}. \tag{1}$$

The density function corresponding to (1) is

$$f_{\text{MO-G}}(x) = \frac{\theta g(x)}{[\theta + (1-\theta)G(x)]^2}, \quad x \in \mathbb{R}. \tag{2}$$

For $\theta = 1$, $f_{\text{MO-G}}(x)$ is equal to $g(x)$ and, for different values of θ, $f_{\text{MO-G}}(x)$ can be more flexible than $g(x)$. The extra parameter θ is called "tilt parameter", since the HRF of the MO-G family is shifted below ($\theta > 1$) or above ($0 < \theta < 1$) of the baseline HRF. Equation (2) provides a useful mechanism to generate new distributions from existing ones. The advantage of this approach for constructing new distributions lies in its flexibility to model both monotonic and non-monotonic HRFs even when the baseline HRF may be monotonic. Tahir and Nadarajah (2015, Table 2) presented thirty distributions belonging to the MO-G family. Further, this family is easily generated from the baseline QF by $Q_{\text{MO-G}}(u) = Q_G\left(\theta u\left[\theta u + 1 - u\right]\right)$ for $u \in (0,1)$.

Marshall and Olkin considered the exponential and Weibull distributions for the baseline G and derived some structural properties of the generated distributions. The special case that G is an exponential distribution refers to a two-parameter competitive model to the Weibull and gamma distributions. A simple interpretation of (1) can be given as follows. Let $T_1, \ldots, T_N$ be a sequence of independent and identically distributed (i.i.d.) random variables with survival function (SF) $\overline{G}(x) = 1 - G(x)$, and let N be a positive integer random variable independent of the T_i's defined by the probability generating function (PGF) of a geometric distribution with parameter θ, say $\tau(z;\theta) = \theta z\left[1 - (1-\theta)z\right]^{-1}$. Then, the inverse of $\tau(z;\theta)$ becomes $\tau^{-1}(z;\theta) = \tau(z;\theta^{-1})$. We can verify that equation (1) comes from $1 - F_{\text{MO-G}}(x) = \tau(\overline{G}(x);\theta)$ for $0 < \theta < 1$ and $1 - F_{\text{MO-G}}(x) = \tau(\overline{G}(x);\theta^{-1})$ for $\theta > 1$. For both cases, $1 - F_{\text{MO-G}}(x)$ represents the SF of $\min\{T_1, \ldots, T_N\}$, where N has PGF $\tau(z;\cdot)$ with probability parameters θ or θ^{-1}.

Zografos and Balakrishnan (2009) and Ristić and Balakrishnan (2011) defined the gamma-G (Γ-G) family with an extra shape parameter $a > 0$ by the CDF (for $x \in \mathbb{R}$)

$$F_{\Gamma\text{-G}}(x) = \gamma_1\left(a, -\log\left[1 - G(x)\right]\right) = \frac{1}{\Gamma(a)}\,\gamma\left(a, -\log\left[1 - G(x)\right]\right), \tag{3}$$

where $\gamma(a,z) = \int_0^z t^{a-1}\,\mathrm{e}^{-t}\mathrm{d}t$ is the incomplete gamma function and $\gamma_1(a,z) = \gamma(a,z)/\Gamma(a)$ is the incomplete gamma function ratio. The PDF of the Γ-G family takes the form

$$f_{\Gamma\text{-G}}(x) = \frac{1}{\Gamma(a)}\left\{-\log[1 - G(x)]\right\}^{a-1}\, g(x), \quad x \in \mathbb{R}. \tag{4}$$

Each new Γ-G distribution can be determined from a given baseline distribution. For $a = 1$, the G distribution is a basic exemplar of the Γ-G family.

Zografos and Balakrishnan (2009) and Ristić and Balakrishnan (2011) presented a physical motivation for the Γ-G family: if $X_{L(1)}, \ldots, X_{L(n)}$ are lower record values from a sequence of independent random variables with common PDF $g(\cdot)$, then the PDF of the nth lower record value has the form (4). If Z is a gamma random variable with unit scale parameter and shape parameter $a > 0$, then $X = Q_G(1 - \mathrm{e}^{\mathrm{Z}})$ has density (4). So, the Γ-G distribution is easily generated from the gamma distribution and the QF of G.

The rest of the paper is organized as follows. Section 2 describes the distribution and density of the new class of distributions called the *Marshall and Olkin-Gamma-G* (MO-Γ-G) family by combining the above generators. We also present some special models. The maximum likelihood estimation of the model parameters of the new family is described in Section 3. In Section 5, we provide some empirical applications to illustrate the potentiality of the proposed family. A variety of theoretical properties are considered in Section 6. Finally, conclusions are noted in Section 7.

2 The New Family

Let $X \sim$MO-Γ-G denote a random variable having the MO-Γ-G family with two extra shape parameters $\theta > 0$ and $a > 0$ and the baseline vector η. By combining Equations (1) and (3), the CDF of X has the form

$$F_X(x) = \frac{\gamma_1\left(a, -\log\left[1 - G(x)\right]\right)}{\theta + (1-\theta)\gamma_1\left(a, -\log\left[1 - G(x)\right]\right)}, \quad x \in \mathbb{R}. \tag{5}$$

By differentiating (6), we can write the PDF of X as

$$f_X(x) = \frac{\theta\left\{-\log[1 - G(x)]\right\}^{a-1} g(x)}{\Gamma(a)\ \left\{\theta + (1-\theta)\gamma_1\left(a, -\log\left[1 - G(x)\right]\right)\right\}^2}, \quad x \in \mathbb{R}. \tag{6}$$

Distribution	Baseline CDF	Generated PDF
Normal	$G(x) = \Phi(x)$	$f_X(x) = \frac{\theta\{-\log[1-\Phi(x)]\}^{a-1}\,\phi(x)}{\Gamma(a)\,\{\theta+(1-\theta)\gamma_1(a,-\log[1-\Phi(x)])\}^2}$
Logistic	$G(x) = \frac{1}{1+\mathrm{e}^{-\mathrm{x}}}$	$f_X(x) = \frac{\theta\,\mathrm{e}^{-\mathrm{x}}\left\{-\log[1-(1+\mathrm{e}^{-\mathrm{x}})^{-1}]\right\}^{\mathrm{a}-1}}{\Gamma(a)\,(1+\mathrm{e}^{-\mathrm{x}})^2\,\{\theta+(1-\theta)\gamma_1(\mathrm{a},-\log[1-(1+\mathrm{e}^{-\mathrm{x}})^{-1}])\}^2}$
Gumbel	$G(x) = 1 - \exp(-\mathrm{e}^{\mathrm{x}})$	$f_X(x) = \frac{\theta\,\exp(a\,x-\mathrm{e}^{\mathrm{x}})}{\Gamma(a)\,\{\theta+(1-\theta)\gamma_1(a,\mathrm{e}^{\mathrm{x}})\}^2}$
Log-Normal	$G(x) = \Phi(\log x)$	$f_X(x) = \frac{\theta\,\phi(\log x)\,\{-\log[1-\Phi(\log x)]\}^{a-1}}{\Gamma(a)\,x\,\{\theta+(1-\theta)\gamma_1(a,-\log[1-\Phi(\log x)])\}^2}$
Exponential	$G(x) = 1 - \exp(-\lambda x),\ \lambda > 0$	$f_X(x) = \frac{\theta\,\lambda^a\,x^{(a-1)}}{\Gamma(a)\,\{\theta+(1-\theta)\gamma_1(a,\lambda x)\}^2}$
Weibull	$G(x) = 1 - \exp(-(\lambda x)^\gamma),\ \lambda, \gamma > 0$	$f_X(x) = \frac{\theta\,\gamma\lambda^{a\,\gamma}x^{a\,\gamma-1}\,\exp\{-(\lambda\,\gamma)^\gamma\}}{\Gamma(a)\{\theta+(1-\theta)\gamma_1[a,(\lambda\,x)^\gamma]\}^2}$
Gamma	$G(x) = \gamma_1(\alpha, \beta x),\ \alpha,\ \beta > 0$	$f_X(x) = \frac{\theta\,\beta^\alpha\,x^{\alpha-1}\,\mathrm{e}^{-\beta\mathrm{x}}\,\{-\log[1-\gamma_1(\alpha,\beta\mathrm{x})]\}^{\mathrm{a}-1}}{\Gamma(a)\,\{\theta+(1-\theta)\gamma_1(a,-\log[1-\gamma_1(\alpha,\beta x)])\}^2}$
Pareto	$G(x) = 1 - \frac{1}{(1+x)^\nu},\ \nu > 0$	$f_X(x) = \frac{\theta\,\mathrm{e}^{-\mathrm{x}}\,[\nu\log(1+\mathrm{x})]^{\mathrm{a}-1}\,\mathrm{g(x)}}{\Gamma(a)\,(1+\mathrm{e}^{-\mathrm{x}})^2\,\{\theta+(1-\theta)\gamma_1(\mathrm{a},\nu\log[1+\mathrm{x}])\}^2}$

Tabela 1: Special Distributions in the MO-Γ-G family.
($\Phi(x)$ and $\phi(x)$ denote the standard normal distribution and density functions and $\gamma_1(a, z)$ is the incomplete gamma function ratio).

Figure 1 shows the performance of the density and hazard functions of the Marshall-Olkin Gamma Weibull (MO-Γ-W, for short). Note that this extension of the Weibull distribution provides more flexibility for density and hazard shapes.

(a) (b)

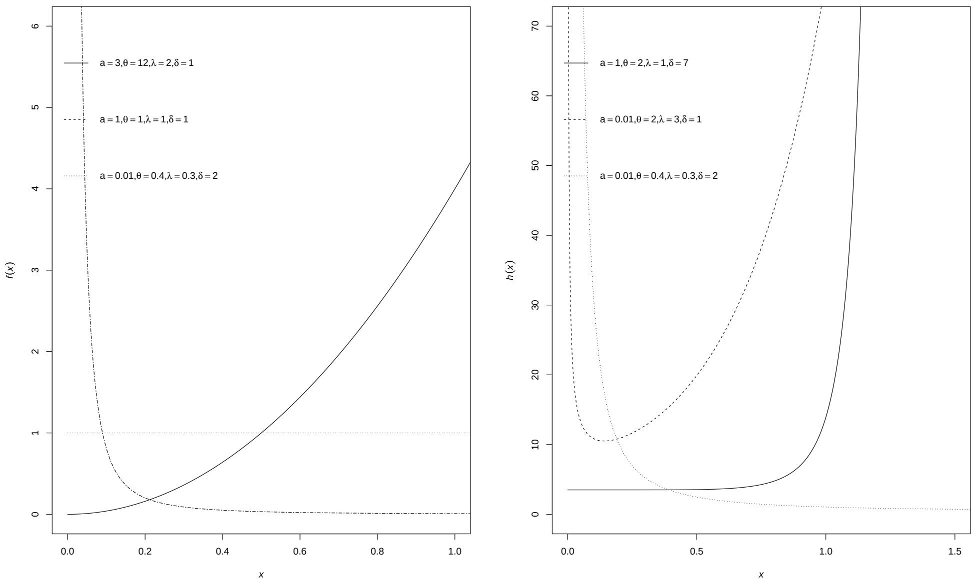

Figura 1: (a) Curves for the MO-Γ-W density; (b) Curves for the MO-Γ-W hazard;

2.1 Quantile Function

We can easily invert the cdf (6) to express the MOGa-G quantile function (qf), say $x = Q_X(u) = G^{-1}(u)$, in terms of the baseline qf $Q_G(u)$. Let u be a uniform $U(0,1)$ random variate. By using (6) and letting $z = \gamma_1\left(a, -\log[1 - G(x)]\right)$, we obtain $z = z(u) = \theta u/[1-(1-\theta)u]$. Then, the qf of X has the form $x = Q_G\left(1 - \mathrm{e}^{-\mathrm{v}}\right)$, where $v = v(z)$ is a random variate generated from a gamma distribution (with shape parameter α and unit scale parameter) corresponding to the quantile z. This scheme is useful because of the existence of fast generators for uniform and gamma random variables.

3 Estimation

The MO-Γ-G family mentioned previously can be fitted to real data sets using the *AdequacyModel* package for the R statistical computing environment (`https://www.r-project.org/`). An important advantage of this package is that it is not necessary to define the log-likelihood function and that it computes the maximum likelihood estimates (MLE), their standard errors and the formal statistics presented in the nest section. We only need to provide the PDF and CDF of the distribution to be fitted to a data set. This *AdequacyModel* package uses the PSO (particle swarm optimization) method obtained by traditional global search approaches such as the quasi-Newton BFGS, Nelder-Mead and simulated-annealing methods to maximized the log-likelihood function. This method does not require initial values. More details are available at `https://rdrr.io/cran/AdequacyModel/`.

Here, we consider estimation of unknown parameters of the MO-Γ-G distribution by the method of maximum likelihood. If x is one observation from **??** and $\boldsymbol{\eta}$ a q-vector parameter vector specifying $G(.)$, thus the log-likelihood function, say $\log L = \log L(a, \theta, \boldsymbol{\eta})$ is

$$\ell(\boldsymbol{\Theta}) = n\log(\theta) + n\log(\gamma) + n\,a\,\gamma\log(\lambda) + (a\gamma - 1)\sum_{i=1}^{n}\log(x_i) - \lambda^{\gamma}\sum_{i=1}^{n}\log(x_i) - n\log[\Gamma(a)]$$
$$- 2\sum_{i=1}^{n}\log\{\theta + (1-\theta)\gamma_1[a, (\lambda\, x_i)^{\gamma}]\} \qquad (7)$$

4 Simulation

Due to the probable absence of maximum likelihood estimators - MLE in closed form for distributions belonging to the MO-Γ-G family, it is necessary to understand the precision of the estimates obtained numerically. For that, we chose to study the bias of the estimators of the MO-Γ-Dagum(θ, a, α, β, p) distribution, in which $G \sim$ Dagum(α, β, p), for different sample sizes, being (10, 20, 60, 100, 200, 400, 600, 1000, 2000, 5000, 10000, 20000, 30000 and 500000).

The simulations considered 10,000 Monte-Carlo - MC iterations for each of the sample sizes to which the numerical estimates were obtained by the BFGS method. Table 2 shows that the BFGS method behaves well as the sample size grows. This is theoretically expected, however, in practice, difficulties can be faced in other families of distributions due to complications in the likelihood function.

All simulations can be reproduced using the code in Annex A. The MC simulations are parallelized and are able to use all threads available by a multicore processor, thus making them more computationally efficient and consequently requiring less time to complete. The simulations were performed on a computer with an Intel Core i5-8265U processor with 8 threads working at a maximum frequency of 3.90 GHz, requiring, on these hardware, a time of 14.36 hours to perform all simulations.

Tabela 2: Mean bias of EMV obtained by the BFGS method in 10,000 Monte Carlo repetitions.

$\boldsymbol{n}$	$\boldsymbol{B(\hat{\theta})}$	$\boldsymbol{B(\hat{a})}$	$\boldsymbol{B(\hat{\alpha})}$	$\boldsymbol{B(\hat{\beta})}$	$\boldsymbol{B(\hat{p})}$	**Time (mins)**
10	0.2376	2.1635	2.7557	1.6282	1.3057	1.1430
20	0.4154	2.4639	1.5728	1.8082	0.7383	1.6248
60	0.7214	2.2432	0.5667	1.8815	0.2872	3.3954
100	0.6146	1.9579	0.3253	1.6148	0.2651	4.9628
200	0.3838	1.3894	0.1827	1.1773	0.3701	8.1457
400	0.2166	0.9635	0.1076	0.6181	0.3957	13.7370
600	0.1269	0.7242	0.0772	0.3968	0.3637	17.9310
1000	0.0553	0.4885	0.0521	0.2328	0.2636	22.8784
2000	0.0456	0.3087	0.0334	0.0990	0.1722	38.4593
5000	-0.0058	0.1307	0.0146	0.0117	0.0171	52.8098
10000	-0.0146	0.0842	0.0095	0.0095	0.0031	95.8380
20000	-0.0090	0.0330	0.0038	0.0005	-0.0099	126.4260
30000	-0.0028	0.0183	0.0012	-0.0036	-0.0029	182.0760
50000	-0.0057	0.0124	0.0015	0.0016	-0.0021	291.9300

5 Applications

In this section, we provide two applications in order to show the performance of our proposed family with another ones. To do this, we consider the Weibull distribution as baseline and for competitive distributions, we consider: beta-Weibull (β-W), proposed by Famoye *et al.* (2005); Kumaraswamy Weibull (KW-W), see Cordeiro and Nadarajah (2010); Marshall-Olkin Weibull (MO-W), studied by Ahmed *et al.* (2017); Marshall-Olkin Extented Weibull (MOE-W), see the family proposed by Cordeiro *et al.* (2019) Exponentiated Weibull (exp-W), introduced by Mudholkar and Srivastava (1993); gamma Weibull (gamma-W), see Cordeiro *et al.* (2016) and exponentiated generalized Weibull (EG-W), see Oguntunde *et al.* (2015), with $a = 1$.

The log-likelihood for the Marshall-Olkin Gamma Weibull (MO-Γ-W) and considering one observation is given by

$$\begin{aligned}\ell(\boldsymbol{\Theta}) =& \log(\theta) + \log(\gamma) + (a\,\gamma)\log(\lambda) + (a\,\gamma - 1)\log(x) - (\gamma\,x)^{\gamma} - \log[\Gamma(a)]\\ &- 2\log\{\theta + (1-\theta)\gamma_1[a, (\lambda\,x)^{\gamma}]\} \end{aligned} \tag{8}$$

where $\boldsymbol{\Theta} = (a, \theta, \lambda, \gamma)^T$. Besides that, the components of the score is given by

$$U_a(\boldsymbol{\Theta}) = \gamma\log(\lambda) + \gamma\log(x) - \psi^{(0)}(a) - \frac{2\left\{(1-\theta)A - (1-\theta)\psi^{(0)}(a)\gamma_1\left[a, (x\lambda)^{\gamma}\right]\right\}}{\theta\,\Gamma(a) + (1-\theta)\gamma_1\left[a, (x\lambda)^{\gamma}\right]},$$

$$U_\theta(\boldsymbol{\Theta}) = \frac{1}{\theta} - \frac{2\left\{\Gamma(a) - \gamma_1\left[a, (\lambda\,x)^{\gamma}\right]\right\}}{\theta\,\Gamma(a) + (1-\theta)\gamma_1\left[a, (\lambda\,x)^{\gamma}\right]},$$

$$U_\lambda(\boldsymbol{\Theta}) = \frac{\gamma}{\lambda}\left[a - (\lambda\,x)^{\gamma}\right] + \frac{2\gamma\,\lambda^{-1}(\lambda\,x)^{a\,\gamma}(1-\theta)\exp\{-(\lambda\,x)^{\gamma}\}}{\theta\,\Gamma(a) + (1-\theta)\gamma_1\left[a, (x\lambda)^{\gamma}\right]}$$

and

$$U_\gamma\boldsymbol{\Theta} = \frac{1}{\gamma} + a\log(\lambda) + a\log(x) - (\lambda x)^{\gamma}\log(\lambda x) + \frac{2(1-\theta)(\lambda x)^{\gamma\,a}\log(\lambda x)\exp\{-(\lambda x)^{\gamma}\}}{\theta\,\Gamma(a) + (1-\theta)\gamma_1\left[a, (x\lambda)^{\gamma}\right]},$$

where

$$A = G_{2,3}^{3,0}\left[(x\lambda)^{\gamma}\middle|\begin{array}{c}1,1\\0,0,a\end{array}\right] + \log\left[(\lambda x)^{\gamma}\right]\gamma_1\left[a, (x\lambda)^{\gamma}\right].$$

where $\psi^{(n)}(x)$ is the n-th derivative of the digamma function and

$$A = \psi^{(0)}(a) - \log\left[(\lambda x)^{\gamma}\right] - G_{2,3}^{3,0}\left((\lambda x)^{\gamma}\middle|\begin{array}{c}1,1\\0,0,a\end{array}\right),$$

where $G_{p,q}^{m,n}\left(z\middle|\begin{array}{c}a_1,\ldots,a_p\\b_1,\ldots,b_q\end{array}\right)$ is the Meijer G function.

To provide the results, in this section we use the R software. In order to adjust the data sets consider here, we use the *AdequacyModel* package mentioned in Section 3. The method used here was the SANN method, that is a variant of simulated annealing (Belisle, 1992). To compare our proposed model with the other ones mentioned above, we use the Anderson Darling (A*) and Cramer Von Mises (W*) statistics, given by the *goodness.fit* function.

For the first data set, we consider a modification of the "FoodExpenditure" from the *betareg* package in R software, that refers on proportion of income spent on food for a random sample of 38 households in a large US city, according to the package information. Here, we consider the household expenditures for food. The modification is

$$data = FoodExpenditure_{food}/\#(FoodExpenditure_{food}),$$

where $FoodExpenditure_{food}$ is the random variable about the household expenditures for food and $\#(.)$ indicates the number of observations on the variable in question. The maximum likelihood estimators (MLE's) and their related standard errors (in parenthesis) are given in Table 3. Besides that, the (W^*) and (A^*) statistics are described in this table too. The results indicate that the proposed model has better performance than the other ones.

Tabela 3: Application 1

Model	a	θ	λ	γ	W^*	A^*
MO-Γ-W$(a,\theta,\lambda,\gamma)$	0.926134 (0.02626926)	1.379664 (0.22381086)	33.323073 (0.28530971)	25.398809 (0.08259864)	0.0339	0.2376
$\beta-$W$(a,\theta,\lambda,\gamma)$	9.92882 (0.02908181)	0.1700880 (0.02049663)	9.759469 (<0.0001)	1.530541 (<0.0001)	0.043567	0.2594618
KW-W$(a,\theta,\lambda,\gamma)$	0.04987575 (0.008090352)	99.99989793 (16.225905058)	1.07602954 (0.003156126)	23.40287099 (0.014646102)	1.330915	6.742609
MOE-W$(a,\theta,\lambda,\gamma)$	0.1366666 (0.1599182)	2.020436 (<0.0001)	62.72201 (<0.0001)	4.2956659 (0.7365535)	0.03541554	0.2579222
EGW(a,b,λ,γ)	5.6189861421 (0.0028147823)	6.1833138579 (0.0009807091)	1.2870816760 (0.1159810838)	1.3798562942 (0.1480747352)	0.03715255	0.2518879
MO-W(a,λ,γ)	0.15920715 (0.07170351)	- (-)	1.58609458 (0.13353716)	4.26713785 (0.16650667)	0.03456333	0.257339
exp-W(a,λ,γ)	6.1102948 (0.4222173)	- (-)	4.4680469 (0.3175876)	1.3858740 (0.1677722)	0.03726684	0.2523749
gamma-W(a,λ,γ)	5.751546 (0.0015006)	- (-)	10.0000 (0.0001163185)	1.208773 (0.00822782)	0.0879	0.6599

As a second application, we consider a data set that can be found in http://biostat.mc.vanderbilt.edu/wiki/ Main/DataSets. This data was collected in a pilot study about hypertension in the Dominican Republic in 1997. The observations are the systolic blood pressure of persons who came to medical clinics in several villages, for a variety of complaints. The MLEs of the model parameters and standard errors and the values of the statistics are listed in Table 4 for the previous models. Overall, by comparing the measures of these formal goodness-of-fit statistics, we conclude that the proposed distribution outperforms all distributions considered in Table 4.

6 Mathematical properties

In this section, we present some main mathematical properties for the MO-Γ-G family based on a general linear representation for its density given in the next section.

Tabela 4: Application 2

Model	a	θ	λ	γ	W^*	A^*
MO-Γ-W$(a,\theta,\lambda,\gamma)$	9.629304 (0.006217876)	3.640779 (0.182495785)	6.260826 (0.024446563)	12.823429 (0.007965221)	0.5093	2.8076
$\beta-$W$(a,\theta,\lambda,\gamma)$	31.08471 (0.01279570)	47.14636 (<0.0001)	0.01698383 (0.00014535)	2.054037 (<0.0001)	0.7540428	4.279418
KW-W$(a,\theta,\lambda,\gamma)$	7363.281 (0.04194304)	0.03925762 (0.0004194304)	1.467640 (<0.0001)	0.6146940 (<0.0001)	0.5351	2.9617
MOE-W$(a,\theta,\lambda,\gamma)$	101.13471834 (46.782882038)	0.42386507 (0.100832102)	0.03095935 (0.003644092)	1.70240332 (0.168142275)	1.14418	6.644617
EGW(a,b,λ,γ)	0.2351691 (0.002540837)	140.0000 (3.278565)	0.4576370 (<0.0001)	0.7425738 (<0.0001)	0.8925	5.3338
MO-W(a,λ,γ)	173.2139 (0.00016394)	- (-)	0.02125455 (0.000212794)	1.601970 (0.0001398109)	1.476088	8.58705
exp-W(a,λ,γ)	69.02916 (0.08389090)	- (-)	0.02405120 (0.0002249256)	1.345536 (<0.0001)	0.8899261	5.094141
gamma-W(a,λ,γ)	9.11229459 (0.96330688)	- (-)	0.02617729 (0.00291875)	1.74641178 (0.08030313)	0.6227098	3.488268

6.1 Linear Representation

A linear representation for the PDF the new family defined previously can be derived using the concept of exponentiated distributions. For an arbitrary baseline CDF $G(x)$, the exponentiated-G (exp-G) distribution with parameter $a > 0$, has CDF and PDF in the forms

$$\Pi_a(x) = G(x)^a \qquad \text{and} \qquad \pi_a(x) = a\, g(x)\, G(x)^{a-1},$$

respectively. The properties of the exponentiated distributions have been studied by many authors in recent years, see Mudholkar and Srivastava (1995) for exponentiated Weibull, Gupta *et al.* (1998) for exponentiated Pareto, Gupta and Kundu (2001) for exponentiated exponential and Nadarajah and Gupta (2007) for exponentiated gamma distribution. Tahir and Nadarajah (2015) cited almost thirty exponentiated distributions in their Table 1.

A linear representation for the PDF of the MO-Γ-G family in terms of exp-G densities is important to determine its mathematical properties from those of the exp-G distributions. They can follow from the papers described below.

First, the MO-G density (2) admits the linear combination (Barreto-Souza *et al.*, 2013)

$$f_{\text{MO-G}}(x) \;=\; \sum_{i=0}^{\infty} w_i^{\text{MO-G}}\, \pi_{i+1}(x),$$

where the coefficients are (for $i = 0, 1, \ldots$)

$$w_i^{\text{MO-G}} = w_i^{\text{MO-G}}(\theta) = \begin{cases} \dfrac{(-1)^i\,\theta}{(i+1)} \sum\limits_{j=i}^{\infty} \dbinom{j}{i} (j+1)\bar{\theta}^j, & \theta \in (0,1), \\ \theta^{-1}(1-\theta^{-1})^i, & \theta > 1, \end{cases}$$

and $\bar{\theta} = 1 - \theta$.

Second, the linear combination for the Γ-G density (4) was derived by Castellares and Lemonte (2015) as

$$f_{\Gamma\text{-G}}(x) = \sum_{i=0}^{\infty} w_i^{\Gamma\text{-G}} \; \pi_{a+i}(x),$$

where

$$w_i^{\Gamma\text{-G}} = w_i^{\Gamma\text{-G}}(a) = \frac{\varphi_i(a)}{(a+i)},$$

$$\varphi_0(a) = \frac{1}{\Gamma(a)}, \quad \varphi_i(a) = \frac{(a-1)}{\Gamma(a)}\,\psi_{i-1}(i+a-2), \quad i \geq 1,$$

and $\psi_{i-1}(\cdot)$ are the Stirling polynomials defined by

$$\psi_{n-1}(x) = \frac{(-1)^{n-1}}{(n+1)!}\left[H_n^{n-1} - \frac{x+2}{n+2}H_n^{n-2} + \frac{(x+2)(x+3)}{(n+2)(n+3)}H_n^{n-3} - \cdots \right.$$
$$\left. + (-1)^{n-1}\frac{(x+2)(x+3)\cdots(x+n)}{(n+2)(n+3)\cdots(2n)}H_n^0\right],$$

where H_n^m are positive integers given recursively by $H_{n+1}^m = (2n+1-m)H_n^m + (n-m+1)H_n^{m-1}$, with $H_0^0 = 1$, $H_{n+1}^0 = 1 \times 3 \times 5 \times \cdots \times (2n+1)$, $H_{n+1}^n = 1$.

7 Conclusions

Falta escrever ...

A Simulation code

The code below was used to perform the simulations whose results are found in Table 2 of Section 4. The code was developed using the programming language R and is available to be reused in other simulations or even to check the results presented in this paper.

Listing 1: Monte-Carlo simulations for different sample sizes.

```
# Title: Marshall Olkin Gamma-G (MOGG)
# Author: Pedro Rafael D. Marinho

# Loading libraries. ------------------------------------------------
library(parallel)
library(tibble)
library(pbmcapply)
library(magrittr)
library(purrr)
library(xtable)

# Baseline functions. -----------------------------------------------
pdf_dagum <- function(x, alpha, beta, p)
  alpha * p / x * (x / beta) ^ (alpha * p) / ((x / beta) ^ alpha + 1) ^ (p + 1)
# integrate(f = pdf_dagum, lower = 0, upper = Inf, alpha = 1.2,
#           beta = 1.6, p = 2.2)

cdf_dagum <- function(x, alpha, beta, p)
  (1 + (x / beta) ^ (-alpha)) ^ (-p)
# cdf_dagum(x = Inf, alpha = 1, beta = 4, p = 1)

# This function creates MOGG functions. ----------------------------
pdf_mogg <- function(g, G) {
  # Using Closures.
```

```r
  function(x, theta, a, ...) {
    if (theta <= 0 || a <= 0)
      warning("The \"a\" and \"theta\" parameters must be greater than zero.")
    num <-
      theta * (-log(1 - G(x = x, ...))) ^ (a - 1) * g(x = x, ...)
    den <-
      gamma(a) *
      (theta + (1 - theta) *
         pgamma(-log(1 - G(x = x, ...)), a, 1L)) ^ 2
    num / den
  }
}

rdagum <- function(n = 1L, alpha, beta, p) {
  beta * (runif(n = n, min = 0, max = 1) ^ (-1 / p) - 1) ^ (-1 / alpha)
}

pdf_mogdagum <- pdf_mogg(g = pdf_dagum, G = cdf_dagum)

# MOG-Dagum -------------------------------------------------------------------
rmogdagum <- function(n = 1L, theta, a, alpha, beta, p) {
  cond_c <- function(x, theta, a, alpha, beta, p) {
    num <- pdf_mogdagum(x, theta, a, alpha, beta, p)
    den <- pdf_dagum(x,
                     alpha = alpha,
                     beta = beta ,
                     p = p)
    - num / den
  }

  x_max <-
    optim(
      fn = cond_c,
      method = "BFGS",
      par = 1,
      theta = theta,
      a = a,
      alpha = alpha,
      beta = beta,
      p = p
    )$par

  c <-
    pdf_mogdagum(x_max, theta, a, alpha, beta, p) / pdf_dagum(x_max,
                                                              alpha = alpha,
                                                              beta = beta,
                                                              p = p)

  criterion <- function(y, u) {
    num <- pdf_mogdagum(y, theta, a, alpha, beta, p)
    den <- pdf_dagum(y,
                     alpha = alpha,
                     beta = beta,
                     p = p)
    u < num / (c * den)
  }

  values <- double(n)
  i <- 1L
  repeat {
    y <- rdagum(
      n = 1L,
      alpha = alpha,
      beta = beta,
      p = p
    )
    u <- runif(n = 1L, min = 0, max = 1)

    if (criterion(y, u)) {
      values[i] <- y
      i <- i + 1L
    }
    if (i > n)
      break
```

```r
  }
  values
}

# Testing the rmogw Function ---------------------------------------------------
theta = 5
a = 1
alpha = 5
beta = 1
p = 1
pdf_mogdagum <- pdf_mogg(g = pdf_dagum, G = cdf_dagum)
sample_data <- rmogdagum(n = 250L, theta, a, alpha, beta, p)
x <- seq(0, max(sample_data), length.out = 500L)
hist(sample_data,
     probability = TRUE,
     xlab = "",
     main = "")
lines(x, pdf_mogdagum(x, theta, a, alpha, beta, p))

# Monte Carlo simulations. -----------------------------------------------------
mc <- function(n = 250L,
               M = 1e3L,
               par_true,
               method = "BFGS") {
  theta <- par_true[1L]
  a <- par_true[2L]
  alpha <- par_true[3L]
  beta <- par_true[4L]
  p <- par_true[5L]

  # Log-likelihood function. ---------------------------------------------------
  pdf_mogw <- pdf_mogg(g = pdf_dagum, G = cdf_dagum)
  log_likelihood <- function(x, par) {
    theta <- par[1L]
    a <- par[2L]
    alpha <- par[3L]
    beta <- par[4L]
    p <- par[5L]

    - sum(log(
      pdf_mogdagum(
        x,
        theta = theta,
        a = a,
        alpha = alpha,
        beta = beta,
        p = p
      )
    ))
  }

  myoptim <-
    function(...)
      tryCatch(
        expr = optim(...),
        error = function(e)
          NA
      )

  one_step_mc <- function(i) {
    sample_data <- rmogdagum(n, theta, a, alpha, beta, p)

    result <- myoptim(
      fn = log_likelihood,
      par = c(1, 1, 1, 1, 1),
      x = sample_data,
      method = method
    )

    while (is.na(result) || result$convergence != 0) {
      sample_data <- rmogdagum(n, theta, a, alpha, beta, p)
      result <- myoptim(
        fn = log_likelihood,
        par = c(1, 1, 1, 1, 1),
```

```
        method = method,
        x = sample_data
      )
    }

    result$par
  }

  result_vector <-
    unlist(
      pbmcapply::pbmclapply(
        X = 1L:M,
        FUN = one_step_mc,
        mc.cores = parallel::detectCores()
      )
    )


  result <-
    tibble::as_tibble(matrix(result_vector, byrow = TRUE, ncol = 5L))

  names(result) <- c("theta", "a", "alpha", "beta", "p")

  result
}

bias_function <- function(x, par_true) {
  x - par_true
}

mse_function <- function(x, par_true) {
  (x - par_true) ^ 2
}

simulate <- function(n) {
  # True parameters (theta, a, alpha, beta and p) -------------------------
  true_parameters <- c(1, 1, 1, 1, 1)

  set.seed(1L, kind = "L'Ecuyer-CMRG")
  t0 <- Sys.time()
  result_mc <-
    mc(
      n = n,
      M = 1e4L,
      par_true = true_parameters,
      method = "BFGS"
    )
  total_time <- Sys.time() - t0

  mc.reset.stream()

  # Average Bias of Estimators --------------------------------------------
  eval(parse(
    text = glue(
      "bias_{n} <- apply(X = result_mc, MARGIN = 1L, FUN = bias_function,
                         par_true = true_parameters) %>%
                   apply(MARGIN = 1L, FUN = mean)"
    )
  ))
  eval(parse(text = glue(
    "save(file = \"bias_{n}.RData\", bias_{n})"
  )))

  # Mean Square Error -----------------------------------------------------
  eval(parse(
    text = glue(
      "mse_{n} <- apply(X = result_mc, MARGIN = 1L, FUN = mse_function,
                        par_true = true_parameters) %>%
                  apply(MARGIN = 1L, FUN = mean)"
    )
  ))
  eval(parse(text = glue(
    "save(file = \"mse_{n}.RData\", mse_{n})"
  )))
```

```r

  # Total Time ----------------------------------------------------------------
  eval(parse(text = glue("time_{n} <- total_time")))
  eval(parse(text = glue(
    "save(file = \"time_{n}.RData\", time_{n})"
  )))

  # Result MC
  eval(parse(text = glue("result_{n} <- result_mc")))
  eval(parse(text = glue(
    "save(file = \"result_{n}.RData\", result_{n})"
  )))

}

walk(
  .x = c(
    10,
    20,
    60,
    100,
    200,
    400,
    600,
    1000,
    2000,
    5000,
    10000,
    20000,
    30000,
    50000
  ),
  .f = simulate
)

first_col <-
  c(10, 20, 60, 100, 200, 400, 600, 1000, 5000, 10000, 20000, 30000, 50000)

tabela <-
  rbind(
    bias_10,
    bias_20,
    bias_60,
    bias_100,
    bias_200,
    bias_400,
    bias_600,
    bias_1000,
    bias_5000,
    bias_10000,
    bias_20000,
    bias_30000,
    bias_50000
  )
tabela <- cbind(n = first_col, tabela)
rownames(tabela) <- NULL

tabela <- tibble::as_tibble(tabela)

latex <-
  print.xtable(
    xtable(tabela,
           caption = "Mean bias of EMV obtained by the BFGS method in 10,000
           Monte Carlo repetitions.",
           digits = 4L),
    print.results = FALSE
  )

writeLines(
  c(
    "\\documentclass[12pt]{article}",
    "\\begin{document}",
    "\\thispagestyle{empty}",
    latex,
```

```
    "\\end{document}"
  ),
  "mc_simulation.tex"
)

tools::texi2pdf("mc_simulation.tex", clean = TRUE)
```

References

A. Alzaatreh, F. Famoye, C. Lee: A new method for generating families of continuous distributions. Metron **71**(1) 63-79 (2013).

W. Barreto-Souza, A.J. Lemonte, G.M. Cordeiro: General results for the Marshall and Olkin's family of distributions. Anais da Academia Brasileira de Ciências **85**(1) 3-21 (2013).

Z.W. Birnbaum, S.C. Saunders: A new family of life distributions. Journal of Applied Probability **6**(2) 319-327 (1969).

F. Castellares, A.J. Lemonte: A new generalized Weibull distribution generated by gamma random variables. Journal of the Egyptian Mathematical Society **23**(2) 382-390 (2015).

G.M. Cordeiro, M. de Castro: A new family of generalized distributions. Journal of Statistical Computation and Simulation **81**(7) 883-898 (2011).

N. Eugene, C. Lee, F. Famoye: Beta-normal distribution and its applications. Communications in Statistics – Theory and Methods **31**(4) 497-512 (2002).

R.C. Gupta, R.D. Gupta, P.L. Gupta: Modeling failure time data by Lehmann alternatives. Communications in Statistics – Theory and Methods **27**(4) 887-904 (1998).

R.D. Gupta, D. Kundu: Exponentiated exponential family: an alternative to gamma and Weibull distributions. Biometrical Journal **43**(1) 117-130 (2001).

A.J. Lemonte, G.M. Cordeiro, G. Moreno-Arenas: A new useful three-parameter extension of the exponential distribution. Statistics **50**(2) 312-337 (2016).

A.W. Marshall, I. Olkin: A new method for adding a parameter to a family of distributions with application to the exponential and Weibull families. Biometrika **84**(3) 641-652 (1997).

G.S. Mudholkar, D.K. Srivastava: The exponentiated Weibull family: a reanalysis of the bus-motor-failure data. Technometrics **37**(4) 436-445 (1995).

S. Nadarajah, G.M. Cordeiro, E.M.M. Ortega: Some general results for the beta-modified Weibull distribution. Journal of Statistical Computation and Simulation **81**(10) 1211-1232 (2011).

S. Nadarajah, G.M. Cordeiro, E.M.M. Ortega: General results for the Kumaraswamy G distribution. Journal of Statistical Computation and Simulation **87**(7) 951-979 (2012).

S. Nadarajah, A.K. Gupta: The exponentiated gamma distribution with application to drought data. Calcutta Statistical Association Bulletin **59**(1-2) 29-54 (2007).

S. Nadarajah, F. Haghighi: An extension of the exponential distribution. Statistics **45**(6) 543-558 (2011).

S. Nadarajah, S. Kotz: The exponentiated-type distributions. Acta Applicandae Mathematicae **92**(2) 97-111 (2006).

M.D. Nichols, W.J. Padgett: A Bootstrap control chart for Weibull percentiles. Quality and Reliability Engineering International **22**(2) 141-151 (2006).

F. Proschan: Theoretical explanation of observed decreasing failure rate. Technometrics **5**(3) 375-383 (1963).

M.M. Ristić, N. Balakrishnan: The gamma-exponentiated exponential distribution. Journal of Statistical Computation and Simulation **82**(8) 1191-1206 (2012).

E.W. Stacy: A Generalization of the Gamma Distribution. Annals of Mathematical Statistics **33**(3) 1187-1192 (1962).

M.H. Tahir, S. Nadarajah: Parameter induction in continuous univariate distributions: Well-established G families. Anais da Academia Brasileira de Ciências **87**(2) 539-568 (2015).

K. Zografos, N. Balakrishnan: On families of beta-and generalized gamma-generated distribution and associate inference. Statistical Methodology **6**(4) 344-362 (2009).