Instructions for the use of the of the codes of the near-exact distributions for the product of independent Generalized Gamma random variables

The computational implementation of the near-exact distributions for the product of independent Generalized Gamma random variables was made using the software Mathematica.

The computational modules described are the ones developed in the paper: Marques, F.J. and Loingeville, F. (2016). Improved Near-Exact Distributions for the Product of Independent Generalized Gamma Random Variables. Computational Statistics and Data Analysis, 102, Pages 55-66. The base modules are the modules available in the web-page:

https://sites.google.com/site/nearexactdistributions/

The modules for the density and cumulative distribution functions of a random variable with a Generalized Near-Integer distribution denoted ahead, respectively, by `GNIGpdf[r_, b_, l_, a_, w_]` and `GNIGcdf[r_, b_, l_, a_, w_]` have to be downloaded from the web-page, together with an auxiliary module denoted by `Makec[r_, l_, p_]`. With these modules it is possible to compute, at a given value `w`, the probability density or the cumulative distribution function of a random variable with a GNIG distribution with integer shape parameters given by the vector `r`, non-integer shape parameter `b`, rate parameters give by the vector `l` and by the parameter `a`.

In order to implement the near-exact distributions we need two more modules, the first, in Figure 1, corresponds to the characteristic function of a mixture of shifted Gamma random variables with weights (except the last one, since this is defined inside the module) given by the vector `p`, shape parameters given by the vector `rrr`, scale parameters given by the vector `la` and shift parameters given by the vector `ww`. The second module, in Figure 2, gives the `h`th moment of a mixture of shifted Gamma random variables.

Figure 1: Module for the characteristic function of the mixture of shifted Gamma random variables

```
FCSMG[p_,rrr_,la_,ww_, t_]:=Module[{c,pp},
  c=Length[p];
  pp=Sum[p[[j]],{j,1,c}];
  Sum[p[[j]]*la[[j]]^rrr[[j]]*(la[[j]]-I*t)^(-rrr[[j]])*Exp[ww[[j]]*I*t],{j,1,c}]+
  (1-pp)*la[[1+c]]^rrr[[1+c]]*(la[[1+c]]-I*t)^(-rrr[[1 + c]])*Exp[ww[[1+c]]*I*t]]
```

Figure 2: Module for the evaluation of the h moment of a mixture of shifted Gamma random variables

```
MomSMG[p_,rrr_,la_,ww_,h_]:=I^(-h)*D[FCSMG[p,rrr,la,ww,t],{t,h}]/.t->0
```

The next two modules, in Figures 3 and 4, can be used to evaluate, at a given value `w`, the near-exact probability density and cumulative distribution functions of $Z = \prod_{j=1}^{p} Y_j$ with $Y_j \sim G\Gamma(r_j, \lambda_j, \beta_j)$, $j = 1, \ldots, p$. In these modules `r`=$\{r_1, \ldots, r_p\}$, `lambda`=$\{\lambda_1, \ldots, \lambda_p\}$, `beta`=$\{\beta_1, \ldots, \beta_p\}$, the parameter `gamma` is a parameter that may be used to control the precision together with parameter `m` which is the number of exact moments matched by construction

Figure 3: Module for the near-exact probability density function of Z

```
nearexactPDF[r_,lambda_,beta_,gamma_,m_,w_]:=Module[{mom,mom1,mom2,v,vs,n,isc,rr,shift,Lambda},
  Clear[omega, rho, upsilon, pe, ppp, pppp];pe = Table[pee[j], {j, 1, m}];
  mom=Table[SetPrecision[I^(-h)*D[Product[Gamma[r[[j]]+gamma-I*t/beta[[j]]]/Gamma[r[[j]]+gamma],
        {j,1,Length[r]}],{t,h}]/.t->0,500],{h,1,m}];
  mom1=Table[SetPrecision[I^(-h)*D[omega^rho*(omega-I*t)^(-rho)*Exp[I*t*upsilon],{t,h}]/.t->0,500],{h,1,3}];
 {rho,omega,upsilon}={rho,omega,upsilon}/.Flatten[Solve[{mom[[1]]==mom1[[1]],mom[[2]]==mom1[[2]],
                       mom[[3]]==mom1[[3]]},{rho,omega,upsilon}]];
  mom2=SetPrecision[Table[MomSMG[pe,Table[rho+i,{i,0,m}],Table[omega,{i,1,m+1}],Table[upsilon,{i,1,m+1}],h],
      {h,1,m}],500];
  ppp=Flatten[pe/.NSolve[Table[mom[[h]]==mom2[[h]],{h,1,m}],pe]];
  pppp=Flatten[{ppp,{1-Sum[ppp[[k]],{k,1,m}]}}];
  shift=upsilon+Log[Product[lambda[[j]]^(1/beta[[j]]),{j,1,Length[r]}]];
  v=Flatten[{Table[Table[((r[[j]]+k)*beta[[j]]),{k,0,gamma-1}],{j,1,Length[r]}]}];
  vs=Sort[v];n=Length[v];Lambda={vs[[1]]};rr={1};isc=1;
  Do[If[vs[[i]]==vs[[i-1]],{rr[[isc]]=rr[[isc]]+1},{isc=isc+1,Lambda=Append[Lambda,vs[[i]]],rr=Append[rr,1]}],
     {i,2,n}];
  Sum[pppp[[k]]*(GNIGpdf[rr,rho+(k-1),Lambda,omega,-Log[w]-shift])*1/w,{k,1,m+1}]
   ]
```

Figure 4: Module for the near-exact cumulative distribution function of Z

```
nearexactCDF[r_,lambda_,beta_,gamma_,m_,w_]:=Module[{mom,mom1,mom2,v,vs,n,isc,rr,shift,Lambda},
  Clear[omega, rho, upsilon, pe, ppp, pppp];pe = Table[pee[j], {j, 1, m}];
  mom=Table[SetPrecision[I^(-h)*D[Product[Gamma[r[[j]]+gamma-I*t/beta[[j]]]/Gamma[r[[j]]+gamma],
        {j,1,Length[r]}],{t,h}]/.t->0,500],{h,1,m}];
  mom1=Table[SetPrecision[I^(-h)*D[omega^rho*(omega-I*t)^(-rho)*Exp[I*t*upsilon],{t,h}]/.t->0,500],{h,1,3}];
 {rho,omega,upsilon}={rho,omega,upsilon}/.Flatten[Solve[{mom[[1]]==mom1[[1]],mom[[2]]==mom1[[2]],
                       mom[[3]]==mom1[[3]]},{rho,omega,upsilon}]];
  mom2=SetPrecision[Table[MomSMG[pe,Table[rho+i,{i,0,m}],Table[omega,{i,1,m+1}],Table[upsilon,{i,1,m+1}],h],
      {h,1,m}],500];
  ppp=Flatten[pe/.NSolve[Table[mom[[h]]==mom2[[h]],{h,1,m}],pe]];
  pppp=Flatten[{ppp,{1-Sum[ppp[[k]],{k,1,m}]}}];
  shift=upsilon+Log[Product[lambda[[j]]^(1/beta[[j]]),{j,1,Length[r]}]];
  v=Flatten[{Table[Table[((r[[j]]+k)*beta[[j]]),{k,0,gamma-1}],{j,1,Length[r]}]}];
  vs=Sort[v];n=Length[v];Lambda={vs[[1]]};rr={1};isc=1;
  Do[If[vs[[i]]==vs[[i-1]],{rr[[isc]]=rr[[isc]]+1},{isc=isc+1,Lambda=Append[Lambda,vs[[i]]],rr=Append[rr,1]}],
     {i,2,n}];
  1-Sum[pppp[[k]]*(GNIGcdf[rr,rho+(k-1),Lambda,omega,-Log[w]-shift]),{k,1,m+1}]
   ]
```

For example to evaluate the cumulative distribution function of Z at the value w=1.29127979859 in the following case

```
r = {2, 3, 5};
lambda = {1, 2, 10};
beta = {5, 6, 7};
gamma = 4;
m = 4;
SetPrecision[nearexactCDF[r, lambda, beta, gamma, m, w], 15]
```

and the result is 0.900000 and is given in about 0.2 seconds in a Personal Computer Intel Core i7 @ 2.00GHz. To evaluate the probability density function of Z at the same value the instruction is

```
SetPrecision[nearexactPDF[r, lambda, beta, gamma, m, w], 15]
```

and the result is 0.823125460966845 and is given also in about 0.2 seconds.