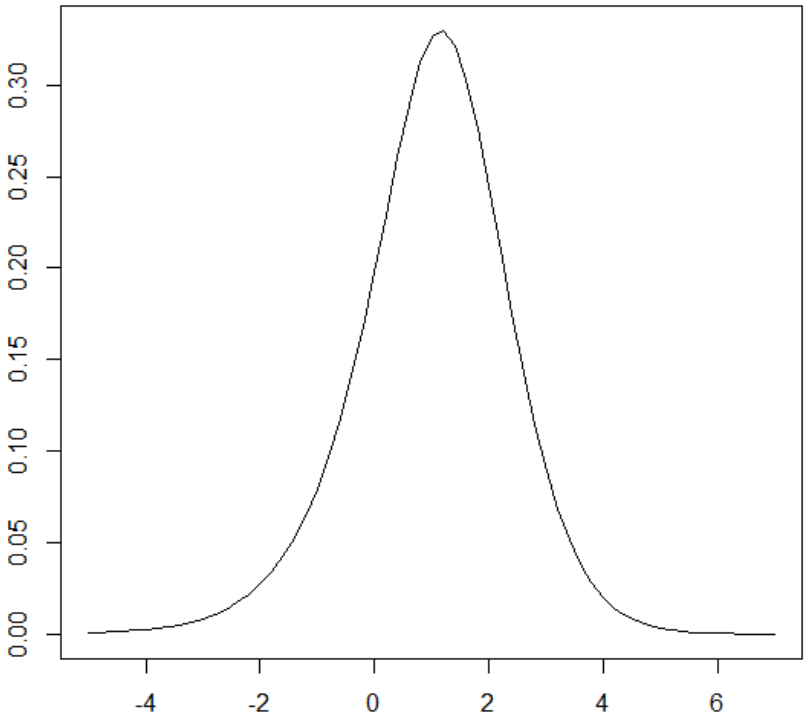
DGIG Workspace

You will need the package Mpfr to run the functions in this Workspace
The function DGIG_help gives help on the user functions, which are:
DGIGpdf, DGIGcdf, DGIGquant
plotDGIGpdf, plotDGIGcdf
and it may also give help on itself.

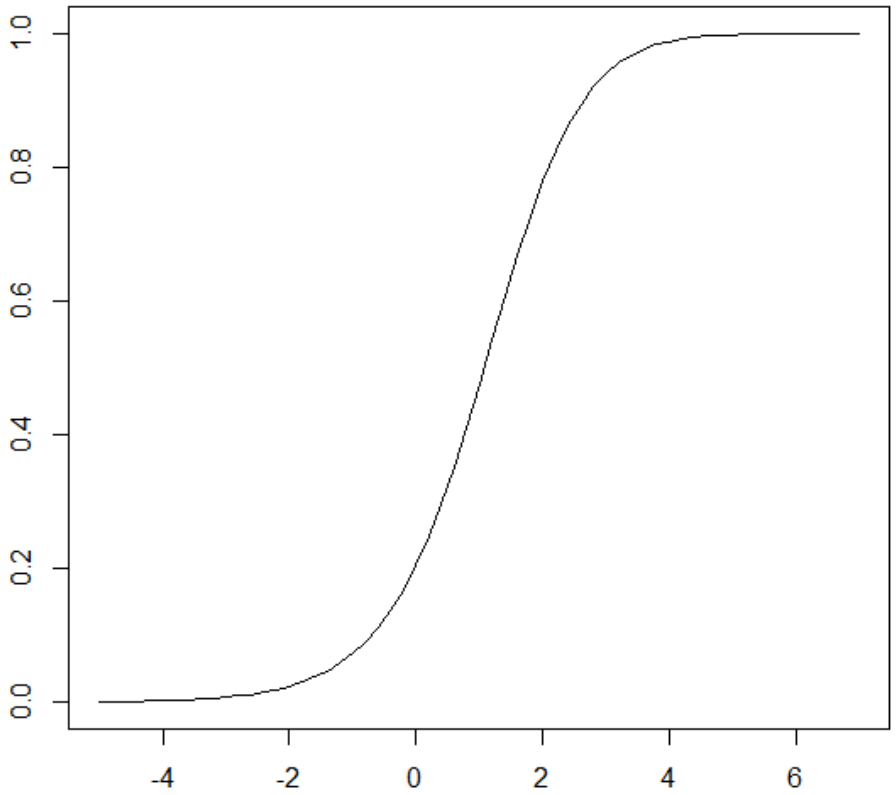
```
> DGIG_help(DGIGpdf)
*********************************************************************************
*  ===> DGIGpdf(r1,r2,l1,l2,z,prec,precp) <===                                  *
* Computes values for the PDF of the DGIG distribution                          *
* It has 7 arguments, the first 5 are mandatory:                                *
*   r1 - list of shape parameters for the Gamma distributions with positive sign *
*   r2 - list of shape parameters for the Gamma distributions with negative sign *
*   l1 - list of rate parameters for the Gamma distributions with positive sign  *
*   l2 - list of rate parameters for the Gamma distributions with negative sign  *
*   z - running value at which the CDF is to be computed                        *
*   prec - (optional argument) number of precision digits for the computations   *
*           (default value: 50)                                                 *
*   precp - (optional argument) number of digits for printing the result        *
*            (default value: 20)                                                *
********************************************************************************* >
DGIGpdf(c(5,3,4),c(2,3),c(3.2,4.5,1.6),c(1.4,1.3),1.2)
    0.19089711134150040569
> # One should be careful and know that for a given value of 'prec', not all 'prec'
> # digits in the result will be exact
> DGIGpdf(c(5,3,4),c(2,3),c(3.2,4.5,1.6),c(1.4,1.3),1.2,,50)
    0.19089711134150040569284558610654807001361124302019
> DGIGpdf(c(5,3,4),c(2,3),c(3.2,4.5,1.6),c(1.4,1.3),1.2,70,50)
    0.19089711134150040569284558610654807001361129292395
> # All user functions, besides giving a printed result, allow for saving the result in
> # a variable which will then have all 'prec' digits
> val<-DGIGpdf(c(5,3,4),c(2,3),c(3.2,4.5,1.6),c(1.4,1.3),1.2)
    0.19089711134150040569
> val
1 'mpfr' number of precision 166    bits
[1] 0.19089711134150040569284558610654807001361124302019 4
> val<-DGIGpdf(c(5,3,4),c(2,3),c(3.2,4.5,1.6),c(1.4,1.3),1.2,70)
    0.19089711134150040569
> val
1 'mpfr' number of precision 232    bits
[1] 0.19089711134150040569284558610654807001361129292394637869101812547108985

> DGIG_help(DGIGcdf)
*********************************************************************************
*  ===> DGIGcdf(r1,r2,l1,l2,z,prec,precp) <===                                  *
* Computes values for the CDF of the DGIG distribution                          *
* It has 7 arguments, the first 5 are mandatory:                                *
*   r1 - list of shape parameters for the Gamma distributions with positive sign *
*   r2 - list of shape parameters for the Gamma distributions with negative sign *
*   l1 - list of rate parameters for the Gamma distributions with positive sign  *
*   l2 - list of rate parameters for the Gamma distributions with negative sign  *
*   z - running value at which the CDF is to be computed                        *
*   prec - (optional argument) number of precision digits for the computations   *
*           (default value: 50)                                                 *
*   precp - (optional argument) number of digits for printing the result        *
*            (default value: 20)                                                *
*********************************************************************************
```

```
> DGIGcdf(c(5,3,4),c(2,3),c(3.2,4.5,1.6),c(1.4,1.3),1.2)
    0.52826935818589085876
> val<-DGIGcdf(c(5,3,4),c(2,3),c(3.2,4.5,1.6),c(1.4,1.3),1.2)
    0.52826935818589085876
> val
1 'mpfr' number of precision 166    bits
[1] 0.52826935818589085875973684292934910811011481427598 9
> val<-DGIGcdf(c(5,3,4),c(2,3),c(3.2,4.5,1.6),c(1.4,1.3),1.2,70,30)
    0.52826935818589085875973684292 9 > val
1 'mpfr' number of precision 232    bits
[1] 0.52826935818589085875973684292934910811011702394545941854056503024295477
```

```
> DGIG_help(plotDGIGpdf)
*****************************************************************************************
*  ===> plotDGIGpdf(r1,r2,l1,l2,ll,ul,step,prec) <===                                   *
* Plots pdfs of the DGIG distribution                                                   *
* arguments are (first 5 are mandatory):                                                *
*   r1 - list of shape parameters for the 'positive' part of the DGIG distribution      *
*   r2 - list of shape parameters for the 'negative' part of the DGIG distribution      *
*   l1 - list of rate parameters for the 'positive' part of the DGIG distribution       *
*   l2 - list of rate parameters for the 'negative' part of the DGIG distribution       *
*   ll - lower limit for plot range                                                     *
*   ul - upper limit for plot range                                                     *
*   step - step for plotting points between ll and ul                                   *
*   prec - number of precision digits for computation of the pdf                        *
*****************************************************************************************
> plotDGIGpdf(c(4,2,3),c(2,1),c(3.6,2.5,4.2),c(1.4,5.3),-5,7,.2)
```



```
> DGIG_help(plotDGIGcdf)
*******************************************************************************
*  ===> plotDGIGcdf(r1,r2,l1,l2,ll,ul,step,prec) <===                          *
* Plots cdfs of the DGIG distribution                                          *
* arguments are (first 5 are mandatory):                                       *
*   r1 - list of shape parameters for the 'positive' part of the DGIG distribution *
*   r2 - list of shape parameters for the 'negative' part of the DGIG distribution *
*   l1 - list of rate parameters for the 'positive' part of the DGIG distribution  *
*   l2 - list of rate parameters for the 'negative' part of the DGIG distribution  *
*   ll - lower limit for plot range                                            *
*   ul - upper limit for plot range                                            *
*   step - step for plotting points between ll and ul                          *
*   prec - number of precision digits for computation of the cdf               *
*******************************************************************************
> plotDGIGcdf(c(4,2,3),c(2,1),c(3.6,2.5,4.2),c(1.4,5.3),-5,7,.2)
```



```
> DGIG_help(DGIGquant)
*************************************************************************
* ===> DGIGquant(r1,r2,l1,l2,quant,xo,eps,prec,precp) <===              *
* Computes quantiles of the DGIG distribution                           *
* It has 9 arguments, the first 6 are mandatory:                        *
*  r1 - list of shape parameters of the positive part                   *
*  r2 - list of shape parameters of the negative part                   *
*  l1 - list of rate parameters corresponding to r1                     *
*  l2 - list of rate parameters corresponding to r2                     *
*  quant - the required quantile (value between 0 and 1)                *
*  xo - initial value for the quantile                                  *
*  eps - (optional argument) largest value of the error for the quantile  *
*        (default value: 10^-10)                                        *
*  prec - (optional argument) number of precision digits for computations *
*         (default value: 50)                                           *
*  precp   (optional argument) number of digits to print the result     *
*          (default value: maximum of -log10(eps) and the required value)*
*************************************************************************
> q05<-DGIGquant(c(4,2,3),c(2,1),c(3.6,2.5,4.2),c(1.4,5.3),.05,7)
    -1.331331651
> q05
1 'mpfr' number of precision 166    bits
[1] -1.331331651457113474930482708933992992454627928574 6
> DGIGcdf(c(4,2,3),c(2,1),c(3.6,2.5,4.2),c(1.4,5.3),q05)
    0.050000000000000000000
> DGIGcdf(c(4,2,3),c(2,1),c(3.6,2.5,4.2),c(1.4,5.3),q05,,30)
    0.050000000000000000000000000000
> DGIGcdf(c(4,2,3),c(2,1),c(3.6,2.5,4.2),c(1.4,5.3),q05,,50)
    0.05000000000000000000000000000000000013337124175301 46
> # For a given value of 'eps', which for default is 10^-10, one gets usually a much
larger precision
> # than only 10 digits
> q05<-DGIGquant(c(4,2,3),c(2,1),c(3.6,2.5,4.2),c(1.4,5.3),.05,-3,10^-20)
    -1.3313316514571134749
> q05
1 'mpfr' number of precision 166    bits
[1] -1.331331651457113474930482708933993016162828705217 79
> DGIGcdf(c(4,2,3),c(2,1),c(3.6,2.5,4.2),c(1.4,5.3),q05)
    0.050000000000000000000
> DGIGcdf(c(4,2,3),c(2,1),c(3.6,2.5,4.2),c(1.4,5.3),q05,,50)
    0.049999999999999999999999999999999999999999983745211
> DGIGcdf(c(4,2,3),c(2,1),c(3.6,2.5,4.2),c(1.4,5.3),q05,100,50)
    0.049999999999999999999999999999999999999999999475064
> q05<-DGIGquant(c(4,2,3),c(2,1),c(3.6,2.5,4.2),c(1.4,5.3),.05,-3,10^-20,100)
    -1.3313316514571134749
> q05
1 'mpfr' number of precision 332    bits
[1] -
1.331331651457113474930482708933993016162828695886468506430741583553605585442710471910 90
326179 40178116
> DGIGcdf(c(4,2,3),c(2,1),c(3.6,2.5,4.2),c(1.4,5.3),q05,100,80)
     0.05000000000000000000000000000000000000000000000000000000000000000000000000000000016726142359
```