

# **pst-math**

**Special mathematical PostScript functions; v.0.66**

Christoph Jorssen

Herbert Voß

October 17, 2022

## **Contents**

<b>1 Introduction</b>	<b>3</b>
<b>2 Trigonometry</b>	<b>3</b>
<b>3 Hyperbolic trigonometry</b>	<b>5</b>
<b>4 Other operators</b>	<b>6</b>
<b>5 Numerical integration</b>	<b>10</b>
<b>6 Random numbers</b>	<b>13</b>
<b>7 Numerical functions</b>	<b>15</b>
<b>References</b>	<b>15</b>

**Abstract**

`pst-math` is an extension to the PostScript language. The files `pst-math.sty` and `pst-math.tex` are only wrapper files for the `pst-math.pro` file, which defines all the new mathematical functions for use with PostScript.

Thanks to:

Denis Bitouzé; Jacques L'helgoualc'h; Patrice Mégret; Dominik Rodriguez

## 1 Introduction

pst-math defines `\pstPi` on T<sub>E</sub>X level which expects 1,2,3 or 4 as parameter. It is not available on PostScript level.

<code>\pstPi#</code>
----------------------

`\pstPi1` ⇒  $\pi$

`\pstPi2` ⇒  $\frac{\pi}{2}$

`\pstPi3` ⇒  $\frac{\pi}{3}$

`\pstPi4` ⇒  $\frac{\pi}{4}$

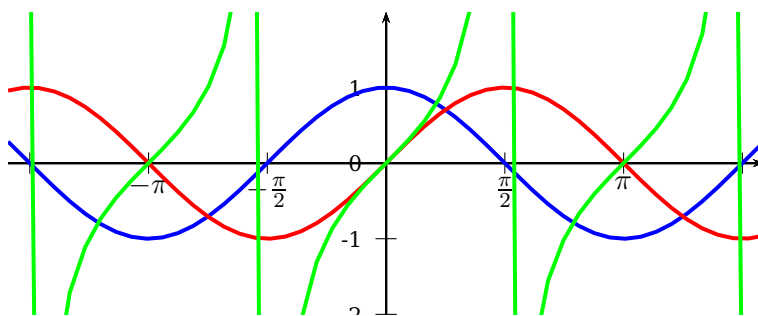
## 2 Trigonometry

pst-math introduces natural trigonometric PostScript operators COS, SIN and TAN defined by

$$\begin{aligned} \cos : & \begin{cases} \mathbb{R} & \rightarrow & [-1, 1] \\ x & \mapsto & \cos(x) \end{cases} \\ \sin : & \begin{cases} \mathbb{R} & \rightarrow & [-1, 1] \\ x & \mapsto & \sin(x) \end{cases} \\ \tan : & \begin{cases} \mathbb{R} \setminus \{k\frac{\pi}{2}, k \in \mathbb{Z}\} & \rightarrow & \mathbb{R} \\ x & \mapsto & \tan(x) \end{cases} \end{aligned}$$

where  $x$  is in *radians*. TAN does *not* produce a PS error<sup>1</sup> when  $x = k\frac{\pi}{2}$ .

Stack	Operator	Result	Description
num	COS	real	Return cosine of num radians
num	SIN	real	Return sine of num radians
num	TAN	real	Return tangent of num radians



```
\begin{pspicture}*(-5,-2)(5,2)
\SpecialCoord % For label positioning
\psaxes[labels=y,Dx=\pstPi2]{->}(0,0)(-5,-2)(5,2)
\uput[-90](!PI 0){$\pi$} \uput[-90](!PI neg 0){$-\pi$}
\uput[-90](!PI 2 div 0){$\frac{\pi}{2}$}
\uput[-90](!PI 2 div neg 0){$-\frac{\pi}{2}$}
\psplot[linewidth=1.5pt,linecolor=blue]{-5}{5}{x COS}
```

<sup>1</sup> TAN is defined with Div, a special PSTricks operator rather than with div, the default PS operator.

```
\psplot[linewidth=1.5pt,linecolor=red]{-5}{5}{x SIN}
\psplot[linewidth=1.5pt,linecolor=green]{-5}{5}{x TAN}
\end{pspicture}
```

pst-math introduces natural trigonometric postscript operators ACOS, ASIN and ATAN defined by

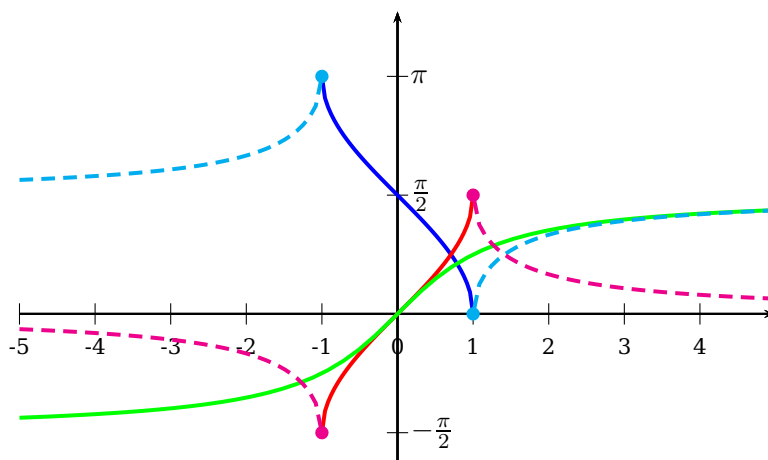
$$\text{acos} : \begin{cases} [-1, 1] & \rightarrow [0, \pi] \\ x & \mapsto \text{acos}(x) \end{cases}$$

$$\text{asin} : \begin{cases} [-1, 1] & \rightarrow [-\frac{\pi}{2}, \frac{\pi}{2}] \\ x & \mapsto \text{asin}(x) \end{cases}$$

$$\text{atan} : \begin{cases} \mathbb{R} & \rightarrow ]-\frac{\pi}{2}, \frac{\pi}{2}[ \\ x & \mapsto \text{atan}(x) \end{cases}$$

Stack	Operator	Result	Description
num	ACOS	angle	Return arccosine of num in radians
num	ASIN	angle	Return arcsine of num in radians
num	ATAN	angle	Return arctangent of num in radians
num	ASEC	angle	Return arcsecans of num in radians
num	ACSC	angle	Return arccosecans of num in radians

ATAN is *not* defined as the already existing PS operator atan. ATAN needs only *one* argument on the stack.



```
\begin{pspicture}(-5,-2)(5,4)
\SpecialCoor % For label positioning
\psaxes[labels=x,Dy=\pstPI2]{->}(0,0)(-5,-2)(5,4)
\uput[0](!0 PI){$\pi$} \uput[0](!0 PI 2 div){$\frac{\pi}{2}$}
\uput[0](!0 PI 2 div neg){$-\frac{\pi}{2}$}
\psset{linewidth=1.5pt,yMaxValue=4}
\psplot[linecolor=blue]{-1}{1}{x ACOS} \psplot[linecolor=red]{-1}{1}{x ASIN}
\psplot[linecolor=green]{-5}{5}{x ATAN}
\psplot[linestyle=dashed,linecolor=cyan,arrows=-*]{-5}{-1}{x ASEC}
\psplot[linestyle=dashed,linecolor=cyan,arrows=-*]{1}{5}{x ASEC}
\psplot[linestyle=dashed,linecolor=magenta,arrows=-*]{-5}{-1}{x ACSC}
\psplot[linestyle=dashed,linecolor=magenta,arrows=-*]{1}{5}{x ACSC}
\end{pspicture}
```

### 3 Hyperbolic trigonometry

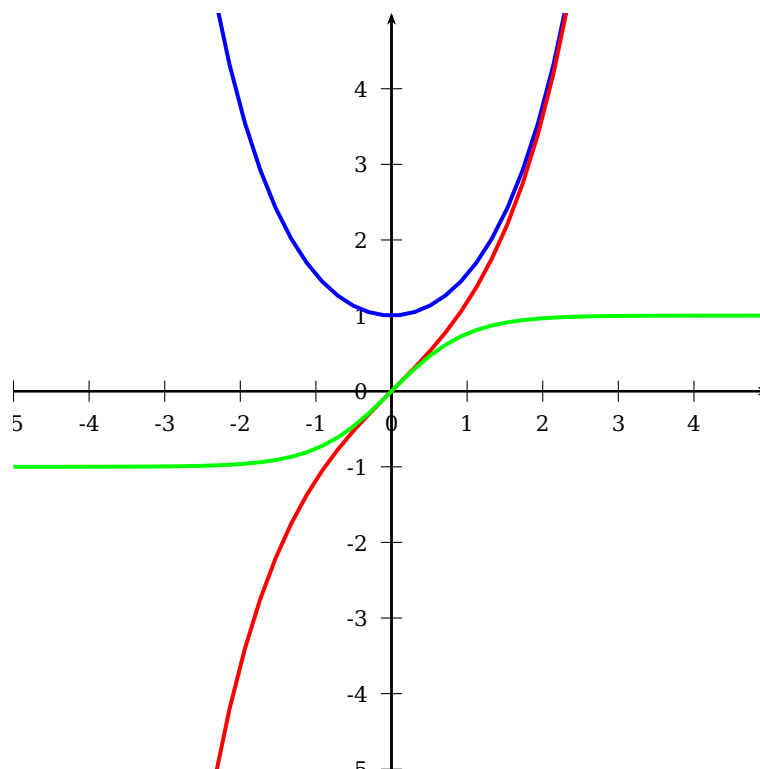
pst-math introduces hyperbolic trigonometric postscript operators COSH, SINH and TANH defined by

$$\cosh : \begin{cases} \mathbb{R} & \rightarrow [1, +\infty[ \\ x & \mapsto \cosh(x) \end{cases}$$

$$\sinh : \begin{cases} \mathbb{R} & \rightarrow \mathbb{R} \\ x & \mapsto \sinh(x) \end{cases}$$

$$\tanh : \begin{cases} \mathbb{R} & \rightarrow ]-1, 1[ \\ x & \mapsto \tanh(x) \end{cases}$$

<i>Stack</i>	<i>Operator</i>	<i>Result</i>	<i>Description</i>
num	COSH	real	Return hyperbolic cosine of num
num	SINH	real	Return hyperbolic sine of num
num	TANH	real	Return hyperbolic tangent of num



```
\begin{pspicture}*(-5,-5)(5,5)
\psaxes{->}(0,0)(-5,-5)(5,5)
\psplot[linewidth=1.5pt,linecolor=blue]{-5}{5}{x COSH}
\psplot[linewidth=1.5pt,linecolor=red]{-5}{5}{x SINH}
\psplot[linewidth=1.5pt,linecolor=green]{-5}{5}{x TANH}
\end{pspicture}
```

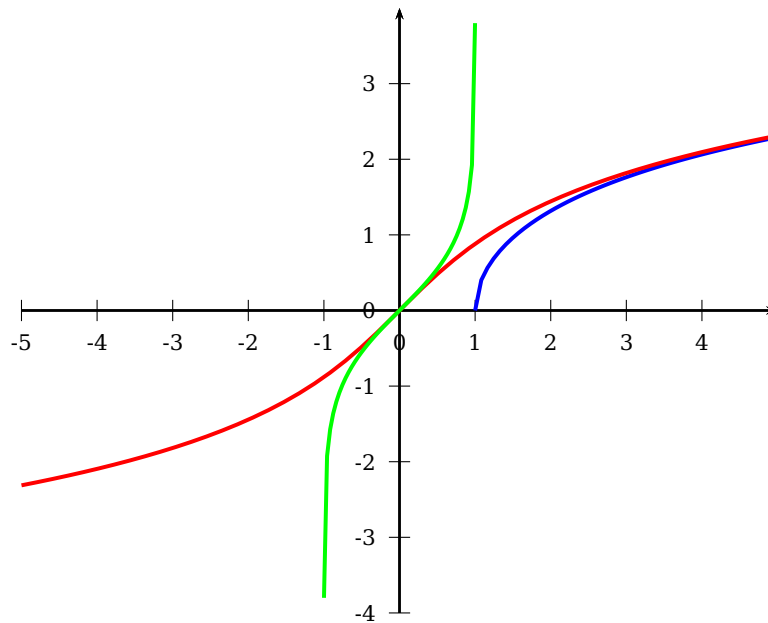
pst-math introduces reciprocal hyperbolic trigonometric postscript operators ACOSH, ASINH and ATANH defined by

$$\operatorname{acosh} : \begin{cases} [1, +\infty[ & \rightarrow \mathbb{R} \\ x & \mapsto \operatorname{acosh}(x) \end{cases}$$

$$\operatorname{asinh} : \begin{cases} \mathbb{R} & \rightarrow & \mathbb{R} \\ x & \mapsto & \operatorname{asinh}(x) \end{cases}$$

$$\operatorname{atanh} : \begin{cases} ]-1, 1[ & \rightarrow & \mathbb{R} \\ x & \mapsto & \operatorname{atanh}(x) \end{cases}$$

<i>Stack</i>	<i>Operator</i>	<i>Result</i>	<i>Description</i>
num	ACOSH	real	Return reciprocal hyperbolic cosine of num
num	ASINH	real	Return reciprocal hyperbolic sine of num
num	ATANH	real	Return reciprocal hyperbolic tangent of num



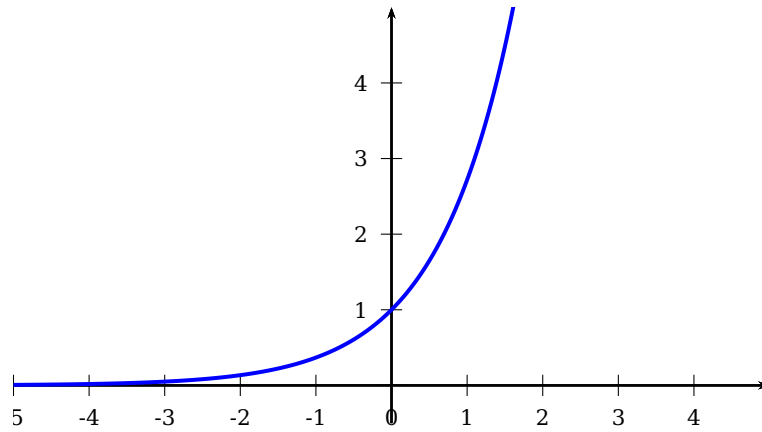
```
\begin{pspicture}(-5,-4)(5,4)
\psaxes{->}(0,0)(-5,-4)(5,4)
\psplot[linewidth=1.5pt,linecolor=blue]{1}{5}{x ACOSH}
\psplot[linewidth=1.5pt,linecolor=red]{-5}{5}{x ASINH}
\psplot[linewidth=1.5pt,linecolor=green]{-.999}{.999}{x ATANH}
\end{pspicture}
```

## 4 Other operators

pst-math introduces postscript operator EXP defined by

$$\operatorname{exp} : \begin{cases} \mathbb{R} & \rightarrow & \mathbb{R} \\ x & \mapsto & \operatorname{exp}(x) \end{cases}$$

<i>Stack</i>	<i>Operator</i>	<i>Result</i>	<i>Description</i>
num	EXP	real	Return exponential of num

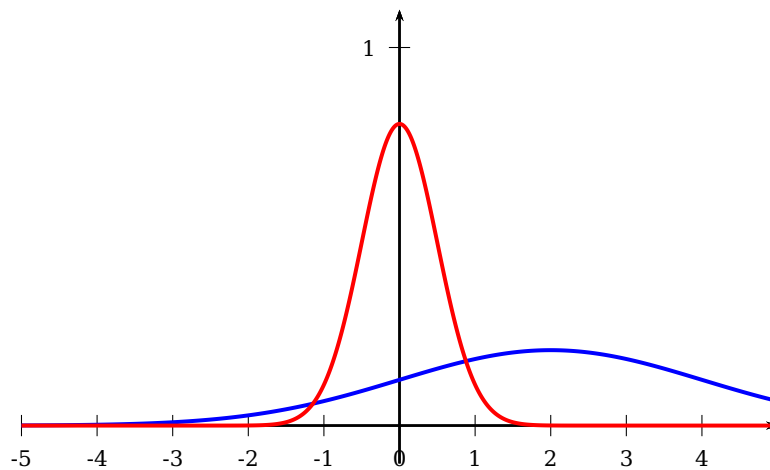


```
\begin{pspicture}*(-5,-1)(5,5)
\psaxes{->}(0,0)(-5,-0.5)(5,5)
\psplot[linecolor=blue,linewidth=1.5pt,plotpoints=1000]{-5}{5}{x EXP}
\end{pspicture}
```

`pst-math` introduces postscript operator `GAUSS` defined by

$$\text{gauss} : \begin{cases} \mathbb{R} & \rightarrow & \mathbb{R} \\ x & \mapsto & \frac{1}{\sqrt{2\pi\sigma^2}} \exp -\frac{(x - \bar{x})^2}{2\sigma^2} \end{cases}$$

<i>Stack</i>	<i>Operator</i>	<i>Result</i>	<i>Description</i>
num1 num2 num3	<code>GAUSS</code>	real	Return gaussian of num1 with mean num2 and standard deviation num3

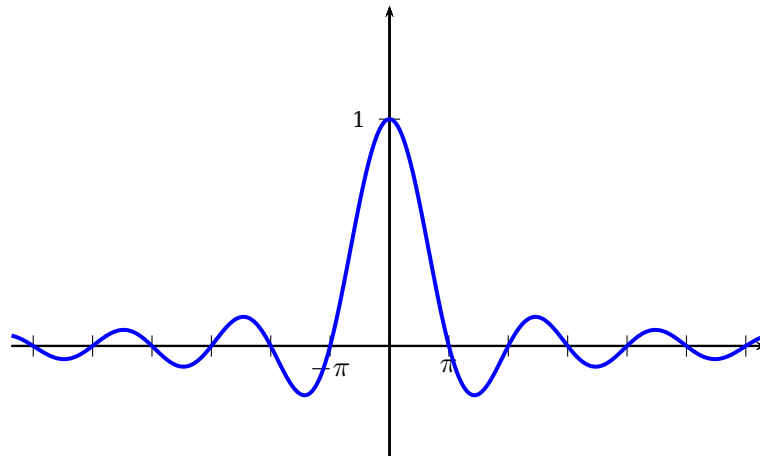


```
\psset{yunit=5}
\begin{pspicture}(-5,-.1)(5,1.1)
\psaxes{->}(0,0)(-5,-.1)(5,1.1)
\psplot[linecolor=blue,linewidth=1.5pt,plotpoints=1000]{-5}{5}{x 2 2 GAUSS}
\psplot[linecolor=red,linewidth=1.5pt,plotpoints=1000]{-5}{5}{x 0 .5 GAUSS}
\end{pspicture}
```

`pst-math` introduces postscript operator `SINC` defined by

$$\text{sinc} : \begin{cases} \mathbb{R} & \rightarrow & \mathbb{R} \\ x & \mapsto & \frac{\sin x}{x} \end{cases}$$

<i>Stack</i>	<i>Operator</i>	<i>Result</i>	<i>Description</i>
num	SINC	real	Return cardinal sine of num radians



```

\psset{xunit=.25,yunit=3}
\begin{pspicture}(-20,-.5)(20,1.5)
\SpecialCoor % For label positioning
\psaxes[labels=y,Dx=\pstPI1]{->}(0,0)(-20,-.5)(20,1.5)
\uput[-90](!PI 0){$\pi$} \uput[-90](!PI neg 0){$-\pi$}
\psplot[linecolor=blue,linewidth=1.5pt,plotpoints=1000]{-20}{20}{x SINC}
\end{pspicture}

```

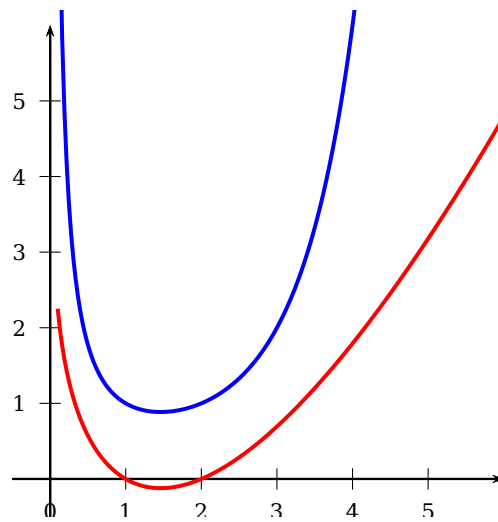
pst-math introduces postscript operator GAMMA and GAMMALN defined by

$$\Gamma : \begin{cases} \mathbb{R} \setminus \mathbb{Z} & \rightarrow \mathbb{R} \\ x & \mapsto \int_0^{\infty} t^{x-1} e^{-t} dt \end{cases}$$

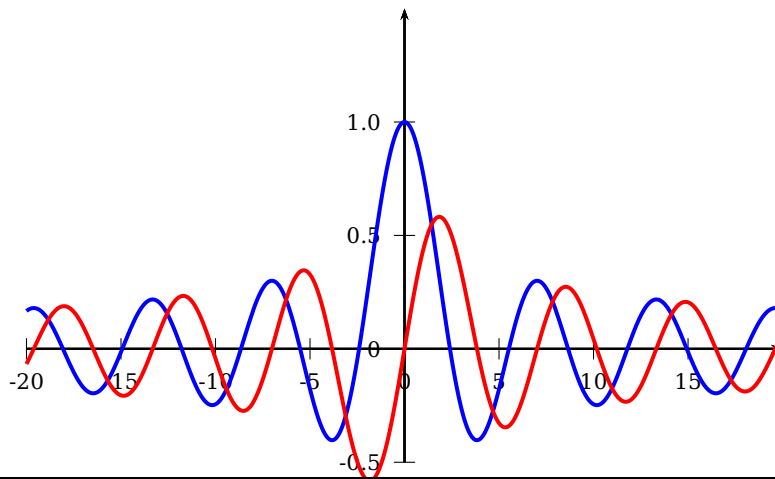
$$\ln \Gamma : \begin{cases} ]0, +\infty[ & \rightarrow \mathbb{R} \\ x & \mapsto \ln \int_0^t t^{x-1} e^{-t} dt \end{cases}$$

<i>Stack</i>	<i>Operator</i>	<i>Result</i>	<i>Description</i>
num	GAMMA	real	Return $\Gamma$ function of num
num	GAMMALN	real	Return logarithm of $\Gamma$ function of num

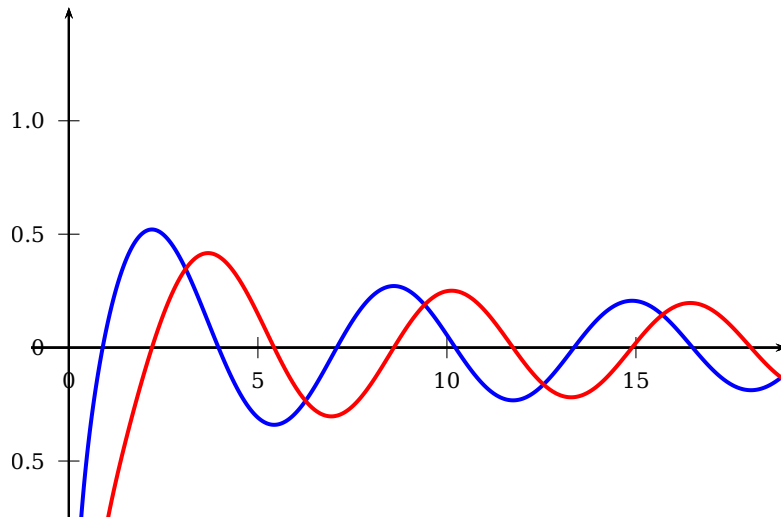




```
\begin{pspicture*}(-.5,-.5)(6.2,6.2)
\psaxes{->}(0,0)(-.5,-.5)(6,6)
\psplot[linecolor=blue,linewidth=1.5pt,plotpoints=200]{.1}{6}{x GAMMA}
\psplot[linecolor=red,linewidth=1.5pt,plotpoints=200]{.1}{6}{x GAMMALN}
\end{pspicture*}
```



```
\psset{xunit=.25,yunit=3}
\begin{pspicture}(-20,-.5)(20,1.5)
\psaxes[Dx=5,Dy=.5]{->}(0,0)(-20,-.5)(20,1.5)
\psplot[linecolor=blue,linewidth=1.5pt,plotpoints=1000]{-20}{20}{x BESSEL_J0}
\psplot[linecolor=red,linewidth=1.5pt,plotpoints=1000]{-20}{20}{x BESSEL_J1}
\end{pspicture}
```

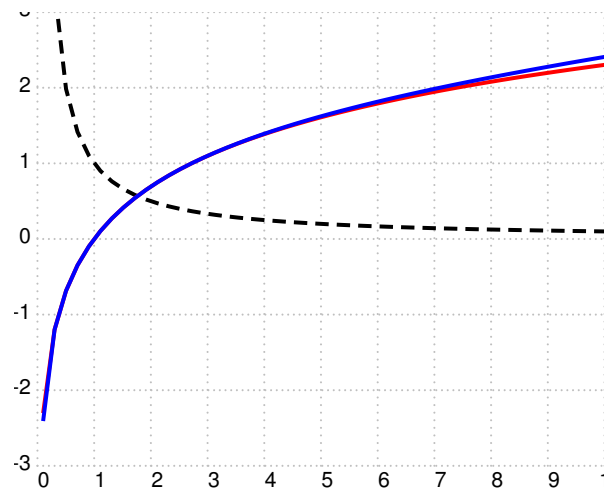


```
\psset{xunit=.5,yunit=3}
\begin{pspicture}*(-1.5,-.75)(19,1.5)
\psaxes[Dx=5,Dy=.5]{->}(0,0)(-1,-.75)(19,1.5)
\psplot[linecolor=blue,linewidth=1.5pt,plotpoints=1000]{0.0001}{20}{x BESSEL_Y0}
\psplot[linecolor=red,linewidth=1.5pt,plotpoints=1000]{0.0001}{20}{x BESSEL_Y1}
%\psplot[linecolor=green,plotpoints=1000]{0.0001}{20}{x 2 BESSEL_Yn}
\end{pspicture}
```

## 5 Numerical integration

<i>Stack</i>	<i>Operator</i>	<i>Result</i>	<i>Description</i>
num num /var { function } num	SIMPSON	real	Return $\int_a^b f(t)dt$
num			

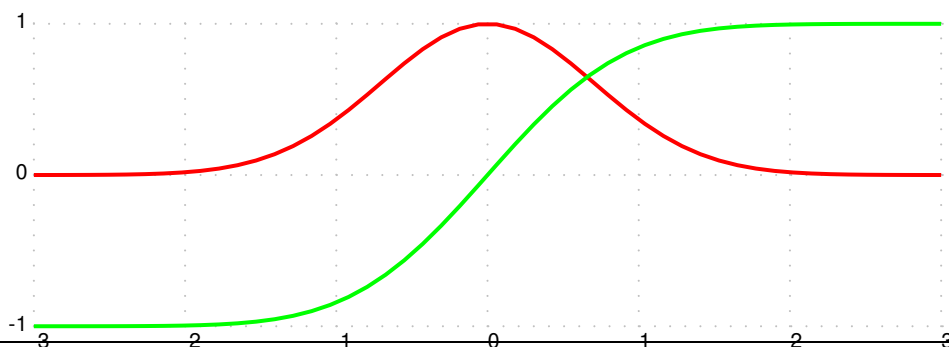
the first two variables are the low and high boundary integral, both can be values or PostScript expressions. /var is the definition of the integrated variable (not x!), which is used in the following function description, which must be inside of braces. The last number is the tolerance for the step adjustment. The function SIMPSON can be nested.



```

\psset{xunit=.75}
\begin{pspicture*}[showgrid=true](-0.4,-3.4)(10,3)
  \psplot[linestyle=dashed,linewidth=1.5pt]{.1}{10}{1 x div}
  \psplot[linecolor=red,linewidth=1.5pt]{.1}{10}{
    1      % start
    x      % end
    /t     % variable
    { 1 t div } % function
    .001  % tolerance
    SIMPSON } %
  \psplot[linecolor=blue,linewidth=1.5pt]{.1}{10}{1 x /t { 1 t div } 1 SIMPSON }
\end{pspicture*}

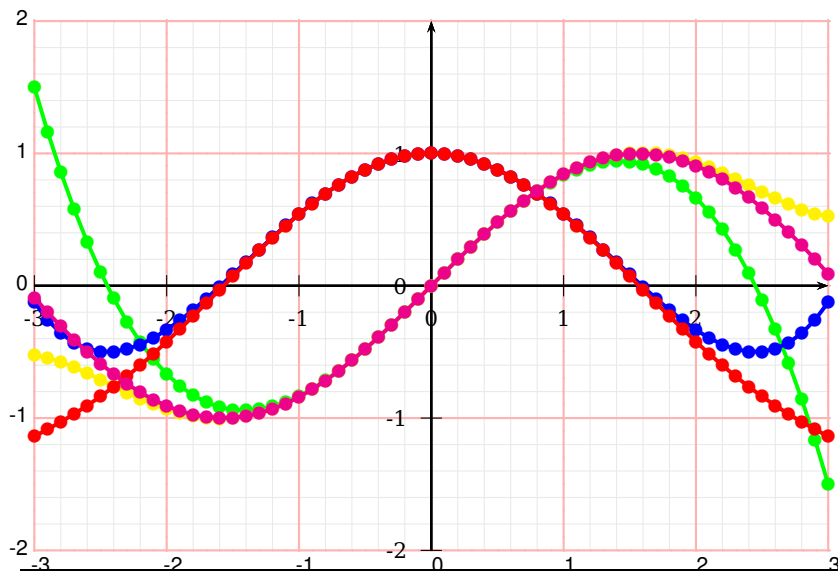
```



```

%% Gaussian and relative integral from -x to x to its value sqrt{pi}
\psset{unit=2}
\begin{pspicture}[showgrid=true](-3,-1)(3,1)
  \psplot[linecolor=red,linewidth=1.5pt]{-3}{3}{Euler x dup mul neg exp }
  \psplot[linecolor=green,linewidth=1.5pt]{-3}{3}
    { x neg x /t { Euler t dup mul neg exp } .001 SIMPSON Pi sqrt div}
\end{pspicture}

```



```

\psset{unit=1.75cm}
%% successive polynomial developments of sine-cosine
\begin{pspicture}[showgrid=true](-3,-2)(3,2)
\psaxes{->}(0,0)(-3,-2)(3,2)
\psplot[linecolor=green, algebraic=false, plotpoints=61, showpoints=true]
{-3}{3}{0 x /tutu
{1 0 tutu /toto { toto } .1 SIMPSON sub}
.01 SIMPSON }
\psplot[linecolor=blue, algebraic=false, plotpoints=61, showpoints=true]
{-3}{3}{1 0 x /tata
{0 tata /tutu
{1 0 tutu /toto { toto } .1 SIMPSON sub}
.01 SIMPSON }
.01 SIMPSON sub}
\psplot[linecolor=yellow, algebraic=false, plotpoints=61, showpoints=true]
{-3}{3}{0 x /titi
{1 0 titi /tata
{0 tata /tutu
{1 0 tutu /toto { toto } .1 SIMPSON sub}
.01 SIMPSON }
.01 SIMPSON sub}
.01 SIMPSON }
\psplot[linecolor=red, algebraic=false, plotpoints=61, showpoints=true]
{-3}{3}{1 0 x /tyty
{0 tyty /titi
{1 0 titi /tata
{0 tata /tutu
{1 0 tutu /toto { toto } .1 SIMPSON sub}
.01 SIMPSON }
.01 SIMPSON sub}
.01 SIMPSON }
.01 SIMPSON sub}
\psplot[linecolor=magenta, algebraic=false, plotpoints=61, showpoints=true]
{-3}{3}{0 x /tete
{1 0 tete /tyty
{0 tyty /titi
{1 0 titi /tata
{0 tata /tutu
{1 0 tutu /toto { toto } .1 SIMPSON sub}
.01 SIMPSON }
.01 SIMPSON sub}

```

```

        .01 SIMPSON }
        .01 SIMPSON sub}
        .01 SIMPSON }%% FIVE nested calls
\end{pspicture}

```

## 6 Random numbers

Package `pst-math` supports the creation of random number lists where a number will appear only once.

```

\defineRandIntervall(min,max)maxNo
\makeSimpleRandomNumberList% multiple values possible
\makeRandomNumberList % no multiple values!
\getNumberFromList{number}

```

The list of the random numbers is `\RandomNumbers`, a comma separated list of the values. It can be used for own purposes.

Random list:

```

12,41,13,11,36,24,47,18,43,7,48,31,22,5,48,36,37,40,18,5,39,6,22,3,25,34,8,8,3,4
12 41 13 11 36 24 47 18 43 7 48 31 22 5 48 36 37 40 18 5 39 6 22 3 25 34 8 8 3 4

```

```

\defineRandIntervall(1,50){30}
\makeSimpleRandomNumberList
Random list: \RandomNumbers

\psforeach{\iA}{1,2,...,30}{\getNumberFromList{\iA}~}

```

In the next example a random number appears only *once* in the list. There are no multiple numbers:

```

17 5 16 10 24 21 11 6 19 30 25 9 8 2 7 4 14 20 13 26 3 27 22 29 23 18 15 1 28 12

```

```

\defineRandIntervall(1,30){30}
\makeRandomNumberList
\psforeach{\iA}{1,2,...,30}{\getNumberFromList{\iA}~}

```

1	18	15	25	8
19	14	24	13	12
11	21	22	3	17
20	7	16	6	10
23	4	2	5	9

25	58	81	64	32	84	63	18	23	96
19	49	98	85	69	100	94	99	77	83
40	24	51	1	76	29	53	52	97	46
11	93	61	10	35	59	38	62	78	82
21	55	70	2	26	33	50	36	54	12
13	4	67	17	92	80	3	41	39	37
22	75	73	45	60	74	15	68	31	47
34	44	87	88	28	72	8	30	43	16
20	91	6	90	89	66	79	56	95	57
7	5	65	27	48	86	71	14	42	9

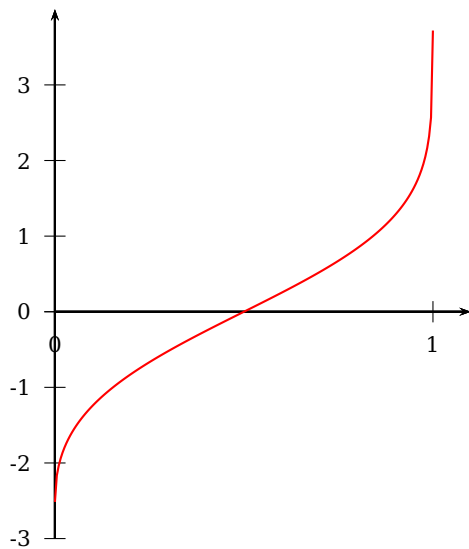
```

\newcounter{RandNo}
\def\n{5} \def\N{\the\numexpr\n*\n}
\defineRandIntervall(1,\N){\N}
\makeRandomNumberList \setcounter{RandNo}{1}
\begin{pspicture}(\n,\n)
  \psgrid[subgriddiv=0,gridlabels=0pt]
  \multido{\rRow=0.5+1.0}{\n}{\multido{\rCol=0.5+1.0}{\n}{%
    \rput(\rCol,\rRow){\getNumberFromList{\theRandNo}}%
    \stepcounter{RandNo}}}
\end{pspicture}
\setcounter{RandNo}{1}
\def\n{10} \def\N{\the\numexpr\n*\n}
\defineRandIntervall(1,\N){\N}
\makeRandomNumberList \setcounter{RandNo}{1}
\begin{pspicture}(\n,\n)
  \psgrid[subgriddiv=0,gridlabels=0pt]
  \multido{\rRow=0.5+1.0}{\n}{\multido{\rCol=0.5+1.0}{\n}{%
    \rput(\rCol,\rRow){\getNumberFromList{\theRandNo}}%
    \stepcounter{RandNo}}}
\end{pspicture}

```

## 7 Numerical functions

Stack	Operator	Result	Description
num	norminv	real	Return $\text{norminv}(num)$



```
\psset{xunit=5}
\begin{pspicture}(-0.1,-3)(1.1,4)
\psaxes{->}(0,0)(0,-3)(1.1,4)
\psplot[plotpoints=200,algebraic,linewidth=red]{0}{0.9999}{norminv(x)}
\end{pspicture}
```

This function returns the inverse normal.

## References

- [1] Denis Girou. “Présentation de PSTricks”. In: *Cahier GUTenberg* 16 (Apr. 1994), pp. 21–70.
- [2] Michel Goossens et al. *The L<sup>A</sup>T<sub>E</sub>X Graphics Companion*. Reading, Mass.: Addison-Wesley Publishing Company, 2007.
- [3] Laura E. Jackson and Herbert Voß. “Die Plot-Funktionen von pst-plot”. In: *Die T<sub>E</sub>Xnische Komödie* 2/02 (June 2002), pp. 27–34.
- [4] Nikolai G. Kollock. *PostScript richtig eingesetzt: vom Konzept zum praktischen Einsatz*. Vaterstetten: IWT, 1989.
- [5] Herbert Voß. “Die mathematischen Funktionen von PostScript”. In: *Die T<sub>E</sub>Xnische Komödie* 1/02 (Mar. 2002).
- [6] Herbert Voß. *Mathematiksatz mit L<sup>A</sup>T<sub>E</sub>X*. Heidelberg/Berlin: DANTE/Lehmanns Media, 2009.
- [7] Herbert Voß. *PSTricks – Grafik für T<sub>E</sub>X und L<sup>A</sup>T<sub>E</sub>X*. 5. Heidelberg/Hamburg: DANTE – Lehmanns, 2008.
- [8] Eric Weisstein. *Wolfram MathWorld*. <http://mathworld.wolfram.com>: Wolfram, 2007.
- [9] Timothy van Zandt. *multido.tex - a loop macro, that supports fixed-point addition*. [CTAN:/graphics/pstricks/generic/multido.tex](http://CTAN.org/graphics/pstricks/generic/multido.tex), 1997.

- 
- [10] Timothy van Zandt. *pst-plot: Plotting two dimensional functions and data*. [CTAN:graphics/pstricks/generic/pst-plot.tex](http://ctan.org/graphics/pstricks/generic/pst-plot.tex), 1999.
- [11] Timothy van Zandt. *PSTricks - PostScript macros for generic T<sub>E</sub>X*. <http://www.tug.org/application/PSTricks>, 1993.
- [12] Timothy van Zandt and Denis Girou. “Inside PSTricks”. In: *TUGboat* 15 (Sept. 1994), pp. 239–246.



## Index

### A

ACOS, 4  
ACOSH, 5, 6  
ACSC, 4  
arccosecans, 4  
arccosine, 4  
arcsecans, 4  
arcsine, 4  
arctangent, 4  
ASEC, 4  
ASIN, 4  
ASINH, 5, 6  
ATAN, 4  
atan, 4  
ATANH, 5, 6

### C

cardinal sine, 8  
COS, 3  
COSH, 5  
cosine, 3

### D

`\defineRandIntervall`, 13

### E

EXP, 6  
exponential, 6

### F

File

– `pst-math.pro`, 2  
– `pst-math.sty`, 2  
– `pst-math.tex`, 2

### G

GAMMA, 8  
 $\Gamma$  function, 8  
GAMMALN, 8  
GAUSS, 7  
`\getNumberFromList`, 13

### H

hyperbolic, 5  
hyperbolic cosine, 5  
hyperbolic sine, 5  
hyperbolic tangent, 5

### I

integral, 10

### L

logarithm, 8

### M

Macro

– `\defineRandIntervall`, 13  
– `\getNumberFromList`, 13  
– `\makeRandomNumberList`, 13  
– `\makeSimpleRandomNumberList`, 13  
– `\pstPI`, 3  
– `\pstPi`, 3  
– `\RandomNumbers`, 13  
`\makeRandomNumberList`, 13  
`\makeSimpleRandomNumberList`, 13

### N

`norminv`, 15

### P

Package

– `pst-math`, 2–8, 13

PostScript, 2

– ACOS, 4  
– ACOSH, 5, 6  
– ACSC, 4  
– ASEC, 4  
– ASIN, 4  
– ASINH, 5, 6  
– ATAN, 4  
– atan, 4  
– ATANH, 5, 6  
– COS, 3  
– COSH, 5  
– EXP, 6  
– GAMMA, 8  
– GAMMALN, 8  
– GAUSS, 7  
– `norminv`, 15  
– SIMPSON, 10  
– SIN, 3  
– SINC, 8  
– SINH, 5  
– TAN, 3  
– TANH, 5

pst-math, 2–8, 13

pst-math.pro, 2

pst-math.sty, 2

pst-math.tex, 2

\pstPI, 3

\pstPi, 3

## R

\RandomNumbers, 13

reciprocal hyperbolic cosine, 6

reciprocal hyperbolic sine, 6

reciprocal hyperbolic tangent, 6

## S

SIMPSON, 10

SIN, 3

SINC, 7

SINC, 8

sine, 3

SINH, 5

standard deviation, 7

## T

TAN, 3

tangent, 3

TANH, 5