

## Ruby master - Feature #10225

### [PATCH] \*math.c: New method Math.normcdf

09/11/2014 06:26 AM - gogotanaka (Kazuki Tanaka)

<b>Status:</b> Open	
<b>Priority:</b> Normal	
<b>Assignee:</b>	
<b>Target version:</b>	
<b>Description</b>	
<p>I know it looks kind of arbitrary or unnecessary feature ,but I believe that to provide such a feature as official is helpful to make people we couldn't reach so far have interest in Ruby.</p> <p>(Actually only this feature is not enough to do that..)</p> <p>By the way I still wonder whether naming method make sense.</p> <p>Debating between ...</p> <pre>Math::pnorm    -&gt; R language use this Math::phi      -&gt; Euler's totient function is more popular as phi, Math::ncdf Math::guesscdf Math::cdf_of_norm, Math::cdf_of_guess</pre> <p>Finally I come to the conclusion Math::normcdf is most proper length, meaning and has no other worries.</p>	

#### History

##### #1 - 09/11/2014 06:43 AM - nobu (Nobuyoshi Nakada)

- Category set to math

- Target version set to 2.2.0

##### #2 - 09/11/2014 09:59 PM - david\_macmahon (David MacMahon)

FWIW, the Ruby/GSL gem already provides support for this (and many other things):

```
$ ruby -r gsl -e 'p GSL::Cdf::gaussian_P(1) '
0.8413447460685429
```

On Sep 10, 2014, at 11:26 PM, [mail@tanakakazuki.com](mailto:mail@tanakakazuki.com) wrote:

Issue [#10225](#) has been reported by gogo tanaka.

---

Feature [#10225](#): [PATCH] \*math.c: New method Math.normcdf  
<https://bugs.ruby-lang.org/issues/10225>

- Author: gogo tanaka
- Status: Open
- Priority: Normal
- Assignee:
- Category:

#### \* Target version:

I know it looks kind of arbitrary or unnecessary feature ,but I believe that

to provide such a feature as official is helpful to make people we couldn't reach so far have interest in Ruby.

(Actually only this feature is not enough to do that..)

By the way I still wonder whether naming method make sense.

Debating between ...

```
Math::pnorm    -> R language use this
Math::phi      -> Euler's totient function is more popular as phi,
Math::ncdf
Math::guesscdf
Math::cdf_of_norm,
Math::cdf_of_guess
```

Finally I come to the conclusion Math::normcdf is most proper length, meaning and has no other worries.

```
---Files-----
_ math.c New method Math.normcdf.PATCH (1.04 KB)
_ test_math.rb Add tests for Math.normcdf.PATCH (609 Bytes)
_ NEWS About Math.normcdf.PATCH (488 Bytes)
```

```
--
https://bugs.ruby-lang.org/
```

### #3 - 09/16/2014 11:14 PM - gogotanaka (Kazuki Tanaka)

[david \(david he\)](#) MacMahon

Actually I know that gem(how nice it is!), I'm disconsolate about a few people knowing that gem.

Even if Ruby/GSL is as strong as python's one, most of people(include me) don't use it.

As for numerical analysis, stability, accuracy or reliability are most important things.

In this respect, I believe that to provide such a feature "as official" is helpful.

Anyway, thank you for telling me that!

gogo.

### #4 - 09/17/2014 04:49 AM - david\_macmahon (David MacMahon)

On Sep 16, 2014, at 4:14 PM, [mail@tanakakazuki.com](mailto:mail@tanakakazuki.com) wrote:

Actually I know that gem(how nice it is!), I'm disconsolate about a few people knowing that gem.

The gsl gem has its issues (memory management and it can be kind of clunky to use sometimes), but it really does provide a great deal of numerical functionality.

Even if Ruby/GSL is as strong as python's one, most of people(include me) don't use it.

As for numerical analysis, stability, accuracy or reliability are most important things.

Is the GSL itself somehow deficient for your needs? It is also an open source project, so I'm (almost) sure patches would be welcome. Ruby/GSL is also open source, though admittedly under-maintained. :-)

In this respect, I believe that to provide such a feature "as official" is helpful.

I agree that having numerical capabilities better supported would be a great thing, but I'm not sure that the Math module is the best place to do so. There are so many things that could be added! Where would it end? :-)

Dave

### #5 - 10/12/2014 06:18 AM - gogotanaka (Kazuki Tanaka)

[david \(david he\)](#) MacMahon

Sorry for my late. It's hard to say but.. ruby-gsl doesn't look good enough.

And I suppose we'd better to make such a library from scratch.

(Actually I start to make other library(called Numrb)

Aside from that, I believe that providing such a feature "as standard" is really important in terms of reliability.

And we can make people we couldn't reach so far have interest in Ruby.

**#6 - 10/13/2014 03:48 AM - david\_macmahon (David MacMahon)**

On Oct 11, 2014, at 11:18 PM, [mail@tanakakazuki.com](mailto:mail@tanakakazuki.com) wrote:

Sorry for my late. It's hard to say but.. ruby-gsl doesn't look good enough.

I agree that ruby-gsl has its shortcomings. GSL is a fairly large library that provides lots of functionality. Exposing all that functionality to Ruby is no small task. Another issue is memory management; I don't think GSL provides a way to hook into its memory allocation so that creates some issues.

And I suppose we'd better to make such a library from scratch.

(Actually I start to make other library(called Numrb)

Aside from that, I believe that providing such a feature "as standard" is really important in terms of reliability.

I think I prefer a sort of middle ground. I'd like some sort of "standard", built-in way of dealing with multidimensional numerical arrays. I do not think we need to include loads of functionality out of the box, but it would be nice to have a standard multidimensional numerical array data type that different Ruby extensions could be created around. IMHO, this would provide a nice modular approach so that things can evolve with a finer granularity. It would somewhat decouple the numerical algorithms from the underlying data structures as well as provide for better interoperability between the different extensions that use multidimensional numerical arrays.

And we can make people we couldn't reach so far have interest in Ruby.

It's always nice to broaden the appeal of Ruby!

**#7 - 10/18/2014 04:31 AM - gogotanaka (Kazuki Tanaka)**

[david \(david he\)](#) MacMahon

Thank you for your valuable comment.

I definitely agree with you. I really want 'built-in way of dealing with multidimensional numerical arrays'

Right now we have Matrix class as standard lib,(it's good!) but it has a problem with performance. (It makes object for each lattice.)

On the other hand SciRuby provide nice gem NMatrix, but we can face some troublesome when installing and updating.(It need ATLAS with CBLAS)

(NumPy does not require any external linear algebra, I like it)

So I have longed for such a built-in way of dealing with multidimensional numerical arrays better.

**#8 - 11/10/2014 10:32 PM - gogotanaka (Kazuki Tanaka)**

Associated revision [10474](#)

**#9 - 01/05/2018 09:01 PM - naruse (Yui NARUSE)**

- Target version deleted (2.2.0)

**Files**

---

<a href="#">_math.c</a> New method Math.normcdf.PATCH	1.04 KB	09/11/2014	<a href="#">gogotanaka (Kazuki Tanaka)</a>
<a href="#">_test_math.rb</a> Add tests for Math.normcdf.PATCH	609 Bytes	09/11/2014	<a href="#">gogotanaka (Kazuki Tanaka)</a>
<a href="#">_NEWS</a> About Math.normcdf.PATCH	488 Bytes	09/11/2014	<a href="#">gogotanaka (Kazuki Tanaka)</a>