New issue

Jump to bottom

cdf never yields result for certain input #7

Closed **thburghout** opened this issue on Apr 9, 2021 · 4 comments

Labels bug

thburghout commented on Apr 9, 2021

Hi,

thanks you for providing this package. Given a specific data set we found that the function is stuck in a while loop. This can be reproduced reliably through the following example:

```
const xs = [ "-630", "-602", "-663", "-715", "-588", "-630", "-680", "-570", "-655", "-636", "-620", "-680", "-551", "-528", "-829", "-528", "-642", "-861", "-642", "-829", "-801",
"-515", "-708", "-680", "-630", "-663", "-655", "-588", "-776", "-620", "-680", "-843", "-630", "-769", "-663", "-811", "-636", "-751", "-500", "-783", "-708", "-723", "-702",
"-671", "-690", "-751", "-611", "-655", "-762", "-756", "-595", "-655", "-702", "-636", "-740", "-490", "-791", "-731", "-636", "-680", "-723", "-731", "-570", "-655", "-642",
"-671", "-620", "-696", "-740", "-740", "-690", "-655", "-690", "-829", "-630", "-680", "-708", "-769", "-671", "-636", "-663", "-551", "-801", "-611", "-769", "-801", "-822",
"-663", "-776", "-671", "-843", "-630", "-791", "-751", "-680", "-731", "-696", "-690", "-723", "-630", "-791", "-762", "-570", "-731", "-648", "-708", "-811", "-791", "-731",
"-811", "-680", "-715", "-731", "-791", "-542", "-611", "-751", "-762", "-740", "-871", "-776", "-740", "-620", "-702", "-769", "-680", "-663", "-671", "-852", "-696", "-630",
"-791", "-500", "-611", "-715", "-776", "-648", "-723", "-671", "-801", "-671", "-756", "-751", "-636", "-680", "-663", "-582", "-723", "-756", "-690", "-776", "-582", "-769",
"-829", "-811", "-595", "-769", "-671", "-791", "-756", "-769", "-655", "-702", "-769", "-731", "-836", "-671", "-560", "-762", "-723", "-811", "-551", "-801", "-671", "-723",
"-801", "-648", "-731", "-769", "-756", "-611", "-801", "-648", "-482", "-636", "-791", "-791", "-769", "-570", "-602", "-811", "-636", "-861", "-861", "-461", "-648", "-690",
"-620", "-655", "-663", "-762", "-655", "-811", "-648", "-715", "-756", "-731", "-751", "-776", "-762", "-740", "-756", "-731", "-769", "-751", "-769", "-740", "-791", "-751",
"-791", "-751", "-756", "-776", "-756", "-756", "-751", "-783", "-751", "-756", "-756", "-762", "-762", "-582", "-783", "-680", "-811", "-648", "-642", "-690", "-836", "-588",
"-769", "-551", "-723", "-811", "-570", "-723", "-690", "-696", "-588", "-636", "-535", "-648", "-690", "-751", "-551", "-588", "-769", "-822", "-642", "-630"]


const cdf = require('cumulative-distribution-function');
const func = cdf(xs)
console.log("hello")
console.log(func(-791))
console.log("done")
```

I hope you find this useful.

thburghout changed the title ~~cdf never yields result for certain inpu~~ cdf never yields result for certain input on Apr 9, 2021

DrPaulBrewer commented on Apr 26, 2021 • edited Owner

Thanks for this. I confirmed the issue.

The input needs to be numeric and finite -- not strings.

Finding the cdf quickly involves bisection and code like this:

cumulative-distribution-function/index.js
Line 35 in afaf917

```
35      mid = Math.floor((left+right)/2);
```

As you may know, in JS "2"+"2" === "22" and with numbers it is 4 (on most days).

Extracting the numeric data seems to fix your example and yields `0.1609...`

```
...
const xs_numeric = xs.map((v)=>(+v)).filter((v)=>(isFinite(v));
const func = cdf(xs_numeric)
...
```

DrPaulBrewer added bug and removed bug labels on Apr 26, 2021

DrPaulBrewer commented on Apr 26, 2021 • edited Owner

Notes to self:

A fix should aim to eliminate the use of improper data as a denial-of-service attack against an app using the cdf() functionality.

cumulative-distribution-function/index.js
Line 13 in 7c5d951

```
13      sorted = data.slice().sort(function(a,b){ return +a-b; });
```

It is already doing a copy of the data array internally; that copy already costs time and memory.

Converting numeric strings and eliminating non-numeric data in a copy of the data is probably not a terrible idea.

thburghout commented on Apr 26, 2021 Author

Thanks for pointing out the cause! I agree that a check there would be appropriate.

DrPaulBrewer added a commit that referenced this issue on Apr 28, 2021

`Update index.js` ··· ✕ b39b078

DrPaulBrewer mentioned this issue on Apr 28, 2021

The Ensure numeric PR checks for clean data and fixes issue #7 #8

Merged

DrPaulBrewer added a commit that referenced this issue on Apr 28, 2021

`Merge pull request #8 from DrPaulBrewer/ensureNumeric` ··· ✕ 339a6cd

DrPaulBrewer commented on Apr 28, 2021 Owner

The fixed version, v2, will be live on NPM within 24 hours.

DrPaulBrewer closed this as completed on Apr 28, 2021

Assignees

No one assigned

Labels

bug

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

2 participants