

---

# phuzzy Documentation

*Release 0.5.15*

**Lepy**

**Apr 19, 2018**



---

## Contents:

---

<b>1</b>	<b>phuzzy</b>	<b>1</b>
1.1	Features . . . . .	1
1.2	Credits . . . . .	1
<b>2</b>	<b>Installation</b>	<b>3</b>
2.1	Stable release . . . . .	3
2.2	From sources . . . . .	3
<b>3</b>	<b>Usage</b>	<b>5</b>
3.1	available shapes . . . . .	5
3.1.1	Uniform . . . . .	5
3.1.2	Triangle . . . . .	5
3.1.3	Trapezoid . . . . .	5
3.1.4	TruncNorm . . . . .	7
3.1.5	TruncGenNorm . . . . .	7
3.1.6	Superellipse . . . . .	7
3.2	basic operations . . . . .	9
3.2.1	Addition . . . . .	9
3.2.2	Substraction . . . . .	10
3.2.3	Multiplication . . . . .	10
3.2.4	Division . . . . .	11
3.2.5	Power . . . . .	11
<b>4</b>	<b>Shapes</b>	<b>13</b>
4.1	Uniform . . . . .	13
4.2	Triangle . . . . .	14
4.3	Trapezoid . . . . .	14
4.4	TruncNorm . . . . .	14
4.5	TruncGenNorm . . . . .	16
4.6	Superellipse . . . . .	16
<b>5</b>	<b>Contributing</b>	<b>19</b>
5.1	Types of Contributions . . . . .	19
5.1.1	Report Bugs . . . . .	19
5.1.2	Fix Bugs . . . . .	19
5.1.3	Implement Features . . . . .	19
5.1.4	Write Documentation . . . . .	20

5.1.5	Submit Feedback . . . . .	20
5.2	Get Started! . . . . .	20
5.3	Pull Request Guidelines . . . . .	21
5.4	Tips . . . . .	21
5.5	Deploying . . . . .	21
<b>6</b>	<b>Credits</b>	<b>23</b>
6.1	Development Lead . . . . .	23
6.2	Contributors . . . . .	23
<b>7</b>	<b>History</b>	<b>25</b>
7.1	0.4.0 (2018-04-13) . . . . .	25
7.2	0.5.0 (2018-04-16) . . . . .	25
<b>8</b>	<b>Indices and tables</b>	<b>27</b>

Documentation: <https://phuzzy.readthedocs.io>.

## 1.1 Features

- TODO

## 1.2 Credits

This package was created with [Cookiecutter](#) and the [audreyr/cookiecutter-pypackage](#) project template.



### 2.1 Stable release

To install phuzzy, run this command in your terminal:

```
$ pip install phuzzy
```

This is the preferred method to install phuzzy, as it will always install the most recent stable release.

If you don't have `pip` installed, this [Python installation guide](#) can guide you through the process.

### 2.2 From sources

The sources for phuzzy can be downloaded from the [Github repo](#).

You can either clone the public repository:

```
$ git clone git://github.com/lepy/phuzzy
```

Or download the [tarball](#):

```
$ curl -OL https://github.com/lepy/phuzzy/tarball/master
```

Once you have a copy of the source, you can install it with:

```
$ python setup.py install
```





To use phuzzy in a project:

```
import phuzzy
tn = phuzzy.TruncNorm(alpha0=[2, 3], alpha1=[], number_of_alpha_levels=15, name="t")
tri = phuzzy.Triangle(alpha0=[1, 4], alpha1=[2], number_of_alpha_levels=5)
f = tn + tri
print(f.df)
```

## 3.1 available shapes

### 3.1.1 Uniform

```
1 import phuzzy.mpl as phm
2 uni = phm.Uniform(alpha0=[1, 4], number_of_alpha_levels=5, name="x")
3 uni.plot(show=True, filepath="/tmp/uniform.png", title=True)
```

### 3.1.2 Triangle

```
1 import phuzzy.mpl as phm
2
3 tri = phm.Triangle(alpha0=[1, 4], alpha1=[2], number_of_alpha_levels=5)
4 tri.plot(show=False, filepath="/tmp/triangle.png", title=True)
```

### 3.1.3 Trapezoid

```
1 import phuzzy.mpl as phm
2 trap = phm.Trapezoid(alpha0=[1, 5], alpha1=[2, 3], number_of_alpha_levels=5)
3 trap.plot(show=False, filepath="/tmp/trapezoid.png", title=True)
```

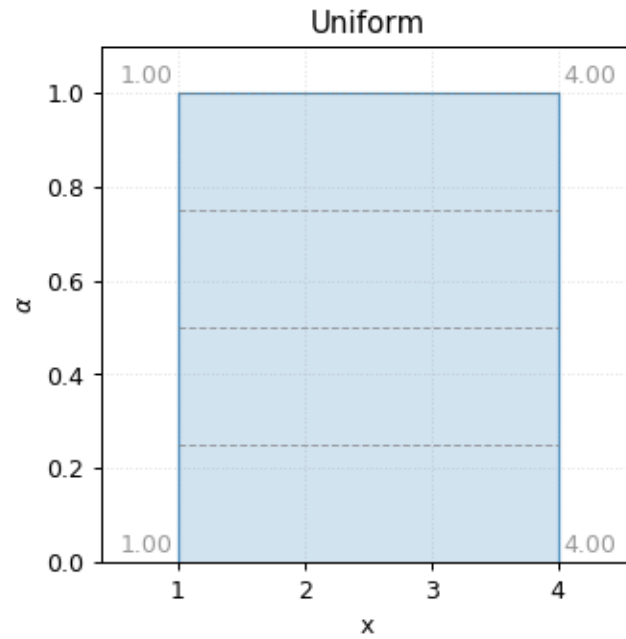


Fig. 3.1: Uniform fuzzy number (this is just an interval)

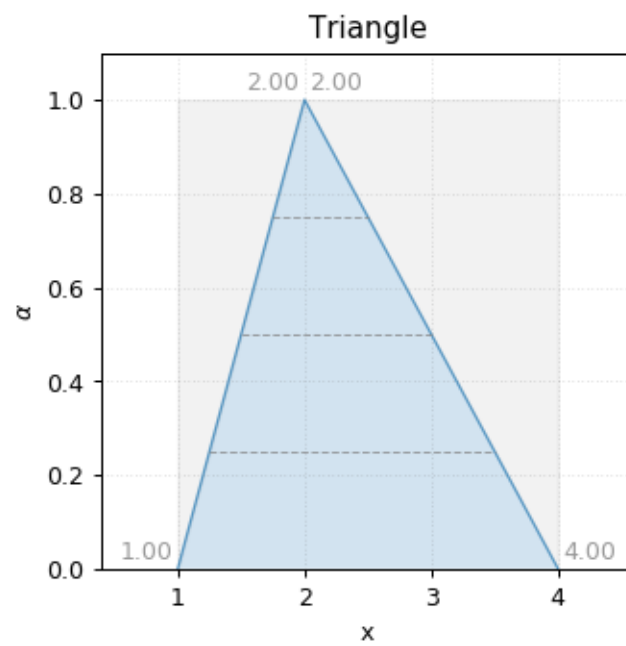


Fig. 3.2: Triangle fuzzy number

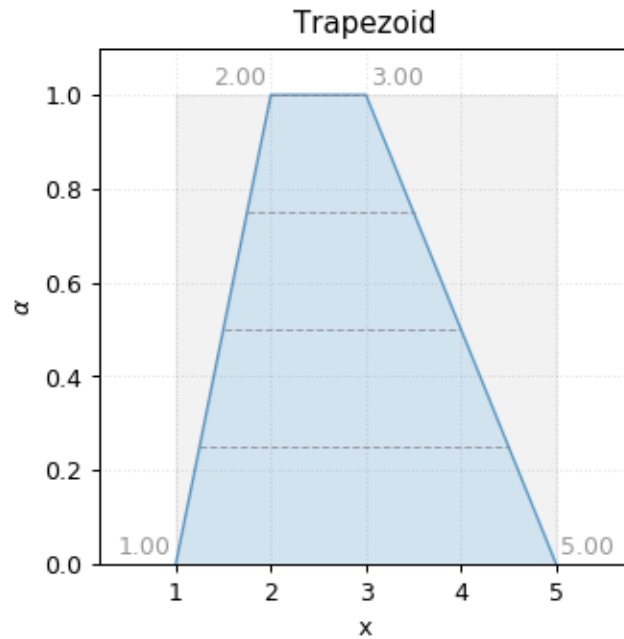


Fig. 3.3: Trapezoid fuzzy number

### 3.1.4 TruncNorm

```

1 import phuzzy.mpl as phm
2 tn = phm.TruncNorm(alpha0=[1, 3], number_of_alpha_levels=15, name="x")
3 tn.plot(show=False, filepath="/tmp/truncnorm.png", title=True)

```

### 3.1.5 TruncGenNorm

```

1 import phuzzy.mpl as phm
2 tgn = phm.TruncGenNorm(alpha0=[1, 4], alpha1=[2, 3], number_of_alpha_levels=15,
3 ↪ beta=3.)
3 tgn.plot(show=False, filepath="/tmp/truncgennorm.png", title=True)

```

### 3.1.6 Superellipse

```

1 import phuzzy.mpl as phm
2 se = phm.Superellipse(alpha0=[-1, 2.], alpha1=None, m=1.0, n=.5, number_of_alpha_
3 ↪ levels=17)
3 se.plot(show=True, filepath="/tmp/superellipse.png", title=True)

```

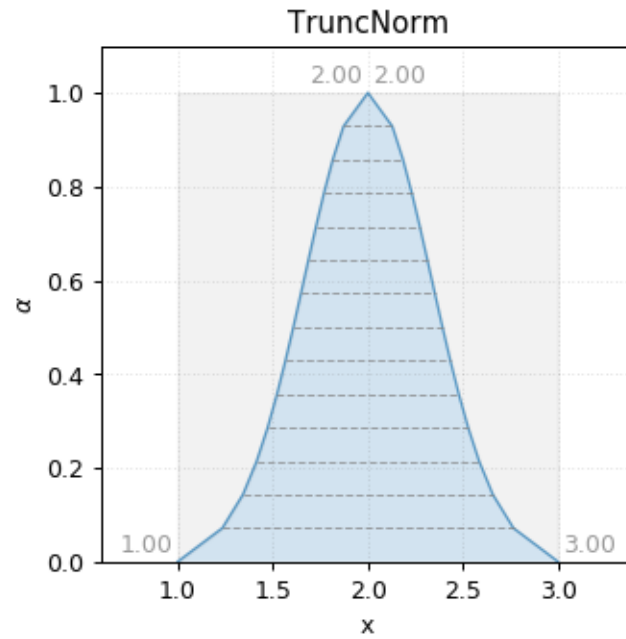


Fig. 3.4: TruncNorm fuzzy number

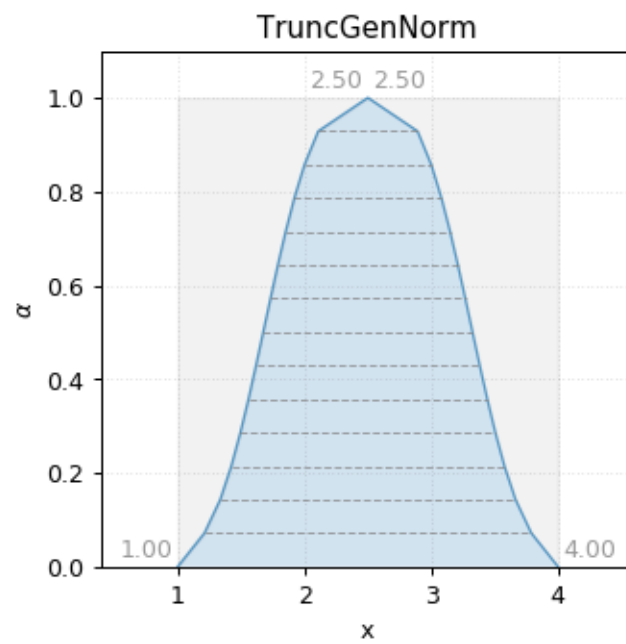


Fig. 3.5: TruncGenNorm fuzzy number

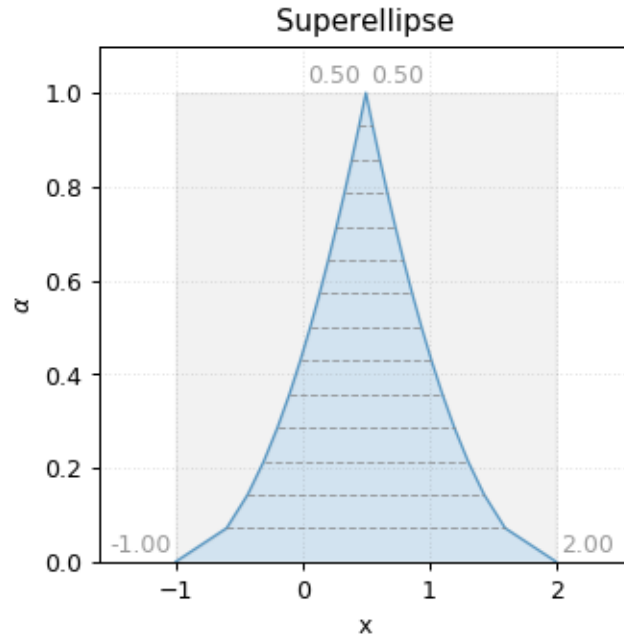


Fig. 3.6: Superellipse fuzzy number

## 3.2 basic operations

### 3.2.1 Addition

$$z = x + y$$

```

1 x = phuzzy.Trapezoid(alpha0=[0, 4], alpha1=[2, 3], number_of_alpha_levels=5)
2 y = phuzzy.TruncNorm(alpha0=[1, 3], number_of_alpha_levels=15, name="y")
3 z = x + y
4 z.name = "x+y"

```

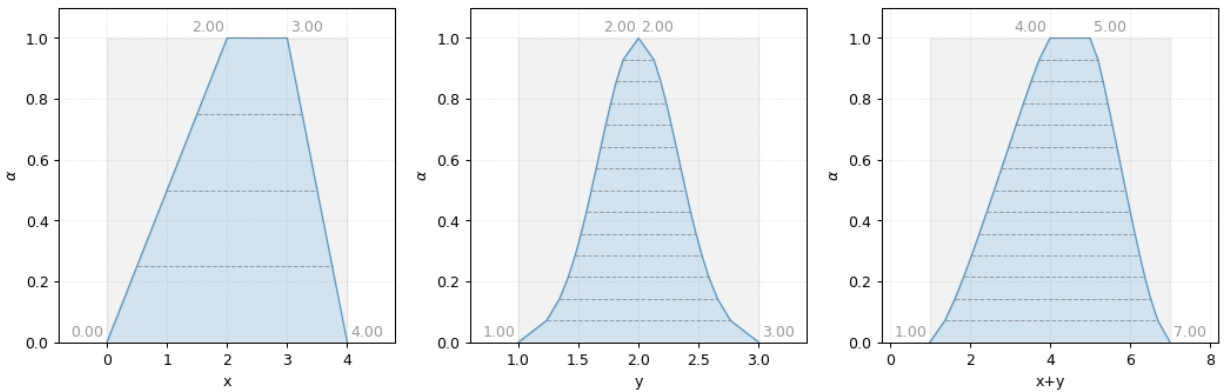


Fig. 3.7: Addition of fuzzy numbers

### 3.2.2 Substraction

$$z = x - y$$

```

1 x = phuzzy.Trapezoid(alpha0=[0, 4], alpha1=[2, 3], number_of_alpha_levels=5)
2 y = phuzzy.TruncNorm(alpha0=[1, 3], number_of_alpha_levels=15, name="y")
3 z = x - y
4 z.name = "x-y"

```

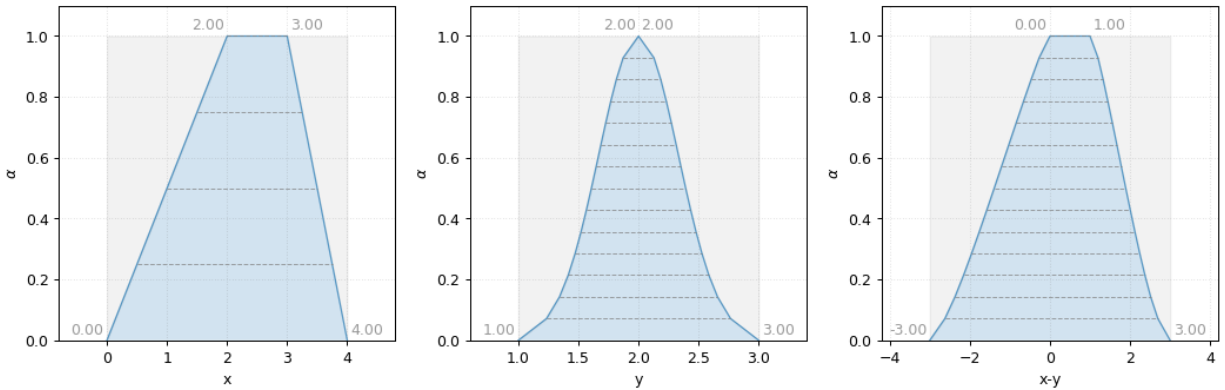


Fig. 3.8: Substraction of fuzzy numbers

### 3.2.3 Multiplication

$$z = xy$$

```

1 x = phuzzy.Trapezoid(alpha0=[0, 4], alpha1=[2, 3], number_of_alpha_levels=5)
2 y = phuzzy.TruncNorm(alpha0=[1, 3], number_of_alpha_levels=15, name="y")
3 z = x * y
4 z.name = "x*y"

```

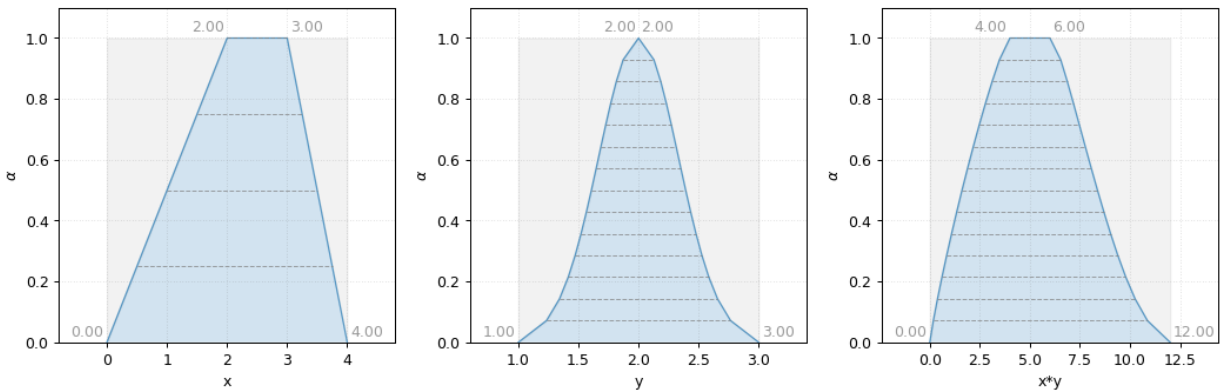


Fig. 3.9: Multiplication of fuzzy numbers

### 3.2.4 Division

$$z = \frac{x}{y}$$

```

1 x = phuzzy.Trapezoid(alpha0=[0, 4], alpha1=[2, 3], number_of_alpha_levels=5)
2 y = phuzzy.TruncNorm(alpha0=[1, 3], number_of_alpha_levels=15, name="y")
3 z = x / y
4 z.name = "x/y"

```

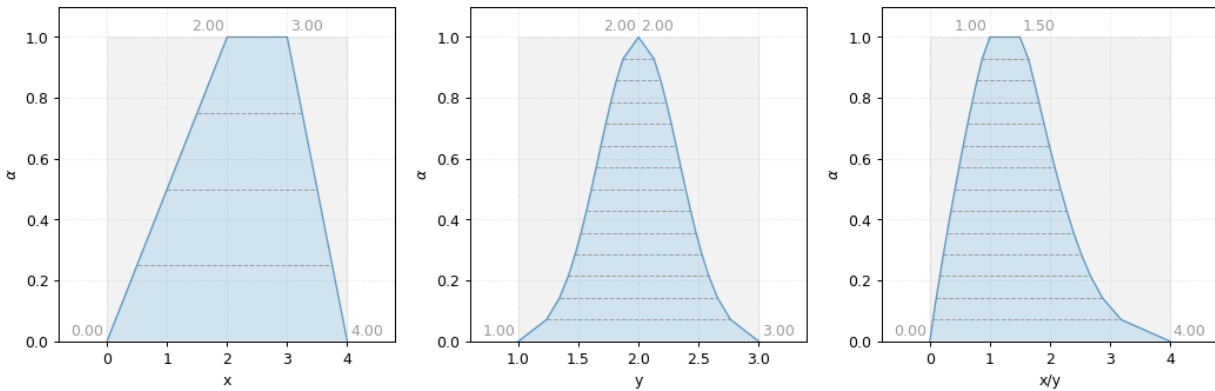


Fig. 3.10: Division of fuzzy numbers

### 3.2.5 Power

$$z = x^y$$

```

1 x = phuzzy.Trapezoid(alpha0=[0, 4], alpha1=[2, 3], number_of_alpha_levels=5)
2 y = phuzzy.TruncNorm(alpha0=[1, 3], number_of_alpha_levels=15, name="y")
3 z = x ** y
4 z.name = "x^y"

```

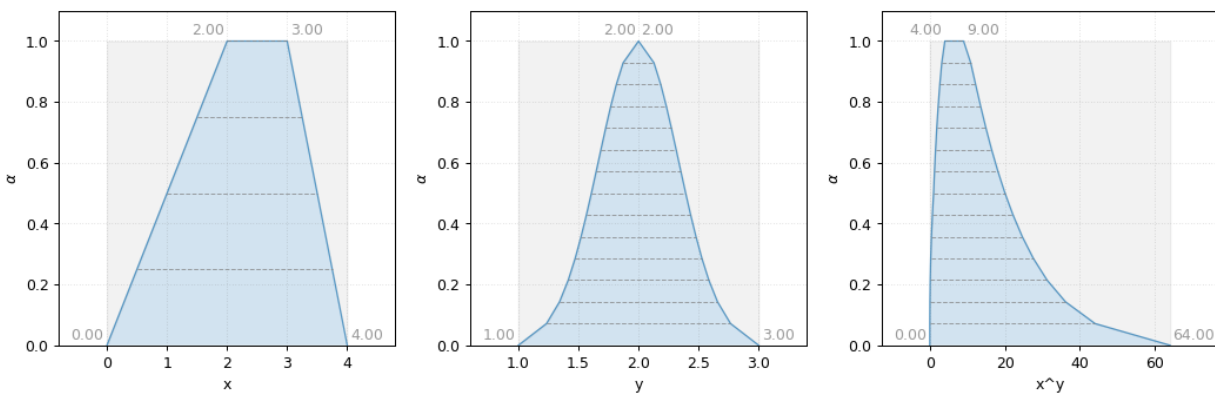


Fig. 3.11: Power operation with fuzzy numbers





## 4.1 Uniform

```
1 import phuzzy.mpl as phm
2 uni = phm.Uniform(alpha0=[1, 4], number_of_alpha_levels=5, name="x")
3 uni.plot(show=True, filepath="/tmp/uniform.png", title=True)
```

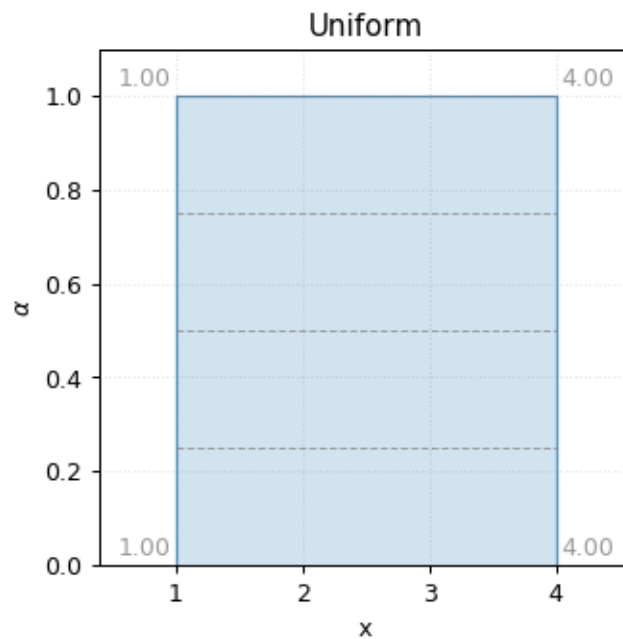


Fig. 4.1: Uniform fuzzy number

## 4.2 Triangle

```

1 import phuzzy.mpl as phm
2
3 tri = phm.Triangle(alpha0=[1, 4], alpha1=[2], number_of_alpha_levels=5)
4 tri.plot(show=False, filepath="/tmp/triangle.png", title=True)

```

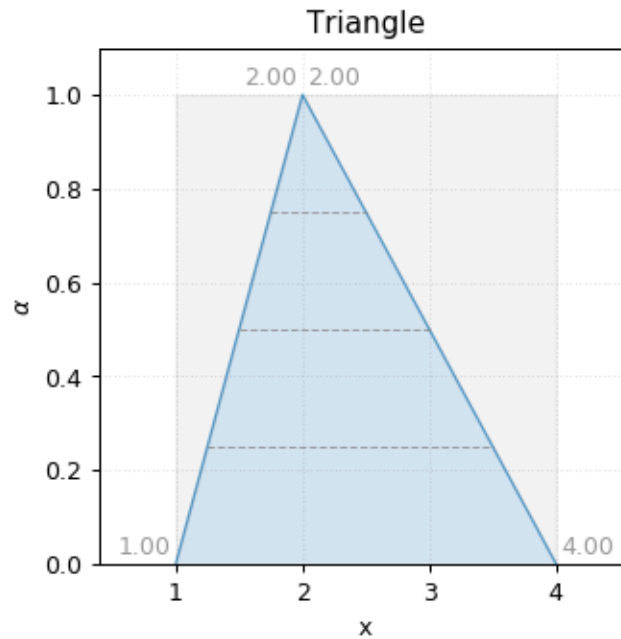


Fig. 4.2: Triangle fuzzy number

## 4.3 Trapezoid

```

1 import phuzzy.mpl as phm
2 trap = phm.Trapezoid(alpha0=[1, 5], alpha1=[2, 3], number_of_alpha_levels=5)
3 trap.plot(show=False, filepath="/tmp/trapezoid.png", title=True)

```

## 4.4 TruncNorm

```

1 import phuzzy.mpl as phm
2 tn = phm.TruncNorm(alpha0=[1, 3], number_of_alpha_levels=15, name="x")
3 tn.plot(show=False, filepath="/tmp/truncnorm.png", title=True)

```

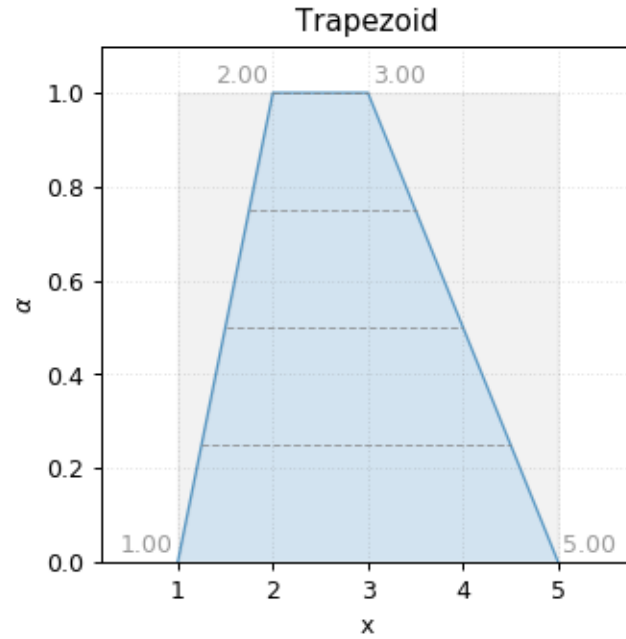


Fig. 4.3: Trapezoid fuzzy number

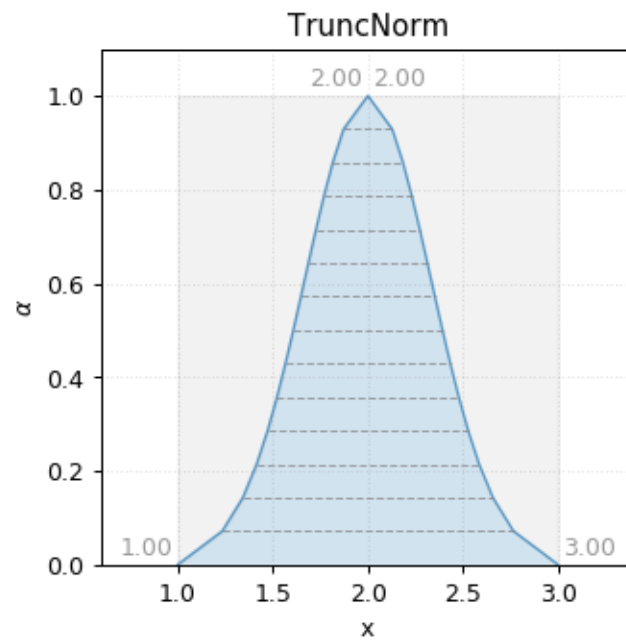


Fig. 4.4: TruncNorm fuzzy number

## 4.5 TruncGenNorm

```

1 import phuzzy.mpl as phm
2 tgn = phm.TruncGenNorm(alpha0=[1, 4], alpha1=[2, 3], number_of_alpha_levels=15,
3   ↪beta=3.)
4 tgn.plot(show=False, filepath="/tmp/truncgennorm.png", title=True)

```

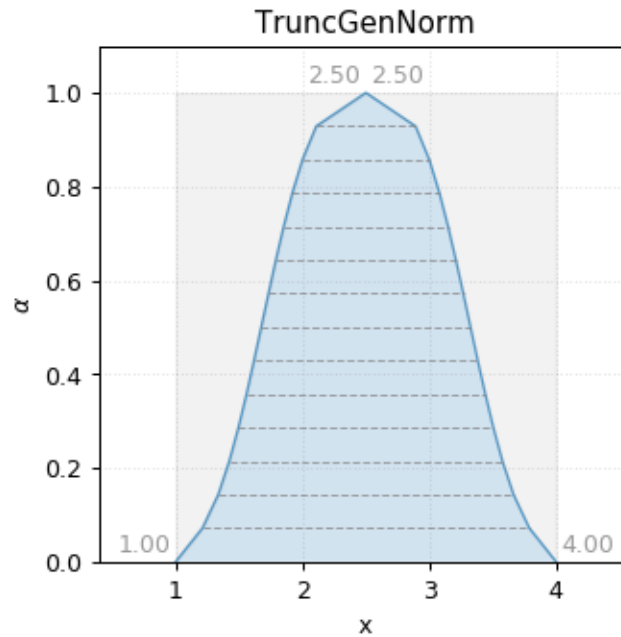


Fig. 4.5: TruncGenNorm fuzzy number

## 4.6 Superellipse

```

1 import phuzzy.mpl as phm
2 se = phm.Superellipse(alpha0=[-1, 2.], alpha1=None, m=1.0, n=.5, number_of_alpha_
3   ↪levels=17)
4 se.plot(show=True, filepath="/tmp/superellipse.png", title=True)

```

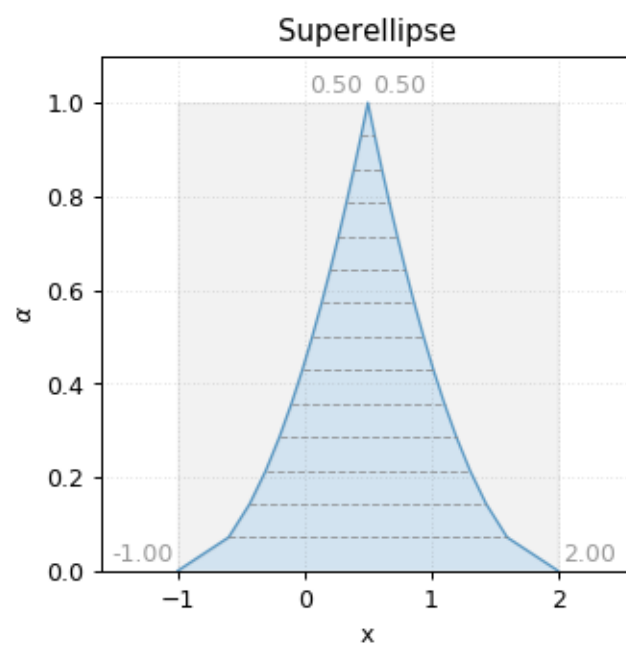
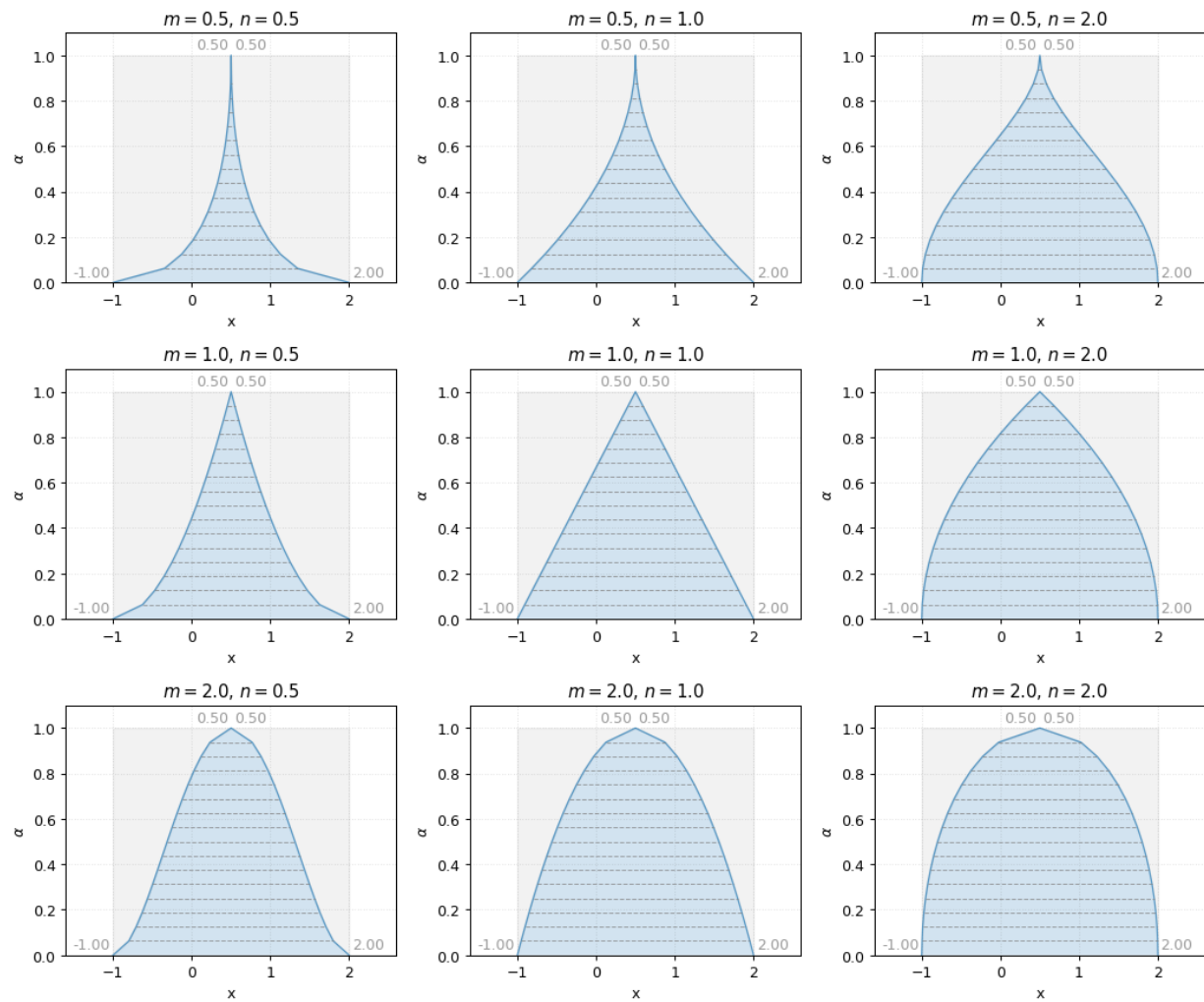


Fig. 4.6: Superellipse fuzzy number

Fig. 4.7: Superellipse fuzzy number (variation  $m, n$ )

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given. You can contribute in many ways:

## 5.1 Types of Contributions

### 5.1.1 Report Bugs

Report bugs at <https://github.com/lepy/phuzzy/issues>.

If you are reporting a bug, please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

### 5.1.2 Fix Bugs

Look through the GitHub issues for bugs. Anything tagged with “bug” and “help wanted” is open to whoever wants to implement it.

### 5.1.3 Implement Features

Look through the GitHub issues for features. Anything tagged with “enhancement” and “help wanted” is open to whoever wants to implement it.

### 5.1.4 Write Documentation

phuzzy could always use more documentation, whether as part of the official phuzzy docs, in docstrings, or even on the web in blog posts, articles, and such.

### 5.1.5 Submit Feedback

The best way to send feedback is to file an issue at <https://github.com/lepy/phuzzy/issues>.

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.
- Remember that this is a volunteer-driven project, and that contributions are welcome :)

## 5.2 Get Started!

Ready to contribute? Here's how to set up *phuzzy* for local development.

1. Fork the *phuzzy* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/phuzzy.git
```

3. Install your local copy into a virtualenv. Assuming you have virtualenvwrapper installed, this is how you set up your fork for local development:

```
$ mkvirtualenv phuzzy
$ cd phuzzy/
$ python setup.py develop
```

4. Create a branch for local development:

```
$ git checkout -b name-of-your-bugfix-or-feature
```

Now you can make your changes locally.

5. When you're done making changes, check that your changes pass flake8 and the tests, including testing other Python versions with tox:

```
$ flake8 phuzzy tests
$ python setup.py test or py.test
$ tox
```

To get flake8 and tox, just pip install them into your virtualenv.

6. Commit your changes and push your branch to GitHub:

```
$ git add .
$ git commit -m "Your detailed description of your changes."
$ git push origin name-of-your-bugfix-or-feature
```

7. Submit a pull request through the GitHub website.



## 5.3 Pull Request Guidelines

Before you submit a pull request, check that it meets these guidelines:

1. The pull request should include tests.
2. If the pull request adds functionality, the docs should be updated. Put your new functionality into a function with a docstring, and add the feature to the list in README.rst.
3. The pull request should work for Python 2.7, 3.4, 3.5 and 3.6, and for PyPy. Check [https://travis-ci.org/lepy/phuzzy/pull\\_requests](https://travis-ci.org/lepy/phuzzy/pull_requests) and make sure that the tests pass for all supported Python versions.

## 5.4 Tips

To run a subset of tests:

```
$ py.test tests.test_phuzzy
```

## 5.5 Deploying

A reminder for the maintainers on how to deploy. Make sure all your changes are committed (including an entry in HISTORY.rst). Then run:

```
$ git push
$ git push --tags
```

Travis will then deploy to PyPI if tests pass.



### 6.1 Development Lead

- Lepy <[lepy@mailbox.org](mailto:lepy@mailbox.org)>

### 6.2 Contributors

None yet. Why not be the first?



### 7.1 0.4.0 (2018-04-13)

- First release on PyPI.

### 7.2 0.5.0 (2018-04-16)

- rename `FuzzyNumber.df.columns` = ["alpha", "l", "r"]  
lsdyna



## CHAPTER 8

---

### Indices and tables

---

- `genindex`
- `modindex`
- `search`