Ranking Documentation

Release 0.3.2

Heungsub Lee

Oct 09, 2020

Contents

strategies to assign ranks

CHAPTER 1

Introduction

In most cases, *enumerate* a Python standard function is a best tool to make a ranking. But how about tie scores? You may end up with giving different rank for tie scores. And I'm quite sure that will make you and your users dissatisfied. Solution? You are on the right page.

```
>>> list(enumerate([100, 80, 80, 70]))
[(0, 100), (1, 80), (2, 80), (3, 70)]
```

This module implements *Ranking* that looks like *enumerate* but generates ranks instead of indexes and various strategy to assign ranks to tie scores.

```
>>> list(Ranking([100, 80, 80, 70])) # same scores have same ranks
[(0, 100), (1, 80), (1, 80), (3, 70)]
```

CHAPTER 2

API

class ranking.**Ranking**(*sequence*, *strategy=COMPETITION*, *start=0*, ***kwargs*)

This class looks like *enumerate* but generates a *tuple* containing a rank and value instead.

```
>>> scores = [100, 80, 80, 70, None]
>>> list(Ranking(scores))
[(0, 100), (1, 80), (1, 80), (3, 70), (None, None)]
```

Parameters

- **sequence** – sorted score sequence. *None* in the sequence means that no score.
- **strategy** – a strategy for assigning rankings. Defaults to *COMPETITION()*.
- **start** – a first rank. Defaults to 0.
- **key** – (keyword-only) a function to get score from a value
- **reverse** – (keyword-only) *sequence* is in ascending order if *True*, descending otherwise. Defaults to *False*.
- **no_score** – (keyword-only) a value for representing "no score". Defaults to *None*.

rank(*value*)

Finds the rank of the value.

Raises **ValueError** – the value isn't ranked in the ranking

ranks()

Generates only ranks.

CHAPTER 3

Strategies to assign ranks

Ranking follows the strategy function when it assigns ranks to tie scores. Also this module provides most common 5 strategies:

`ranking.`**`COMPETITION`**(*start*, *length*)

`ranking.`**`MODIFIED_COMPETITION`**(*start*, *length*)

`ranking.`**`DENSE`**(*start*, *length*)

`ranking.`**`ORDINAL`**(*start*, *length*)

`ranking.`**`FRACTIONAL`**(*start*, *length*)

You can also implement your own strategy function. A strategy function has parameters *start*, a rank of the first tie score; *length*, a length of tie scores. Then it returns *length* + 1 for each scores for tie scores and the next rank.

Here's an example of custom strategy function that assigns no ranks to tie scores:

```
>>> def exclusive(start, length):
...     return [None] * length + [start]
>>> list(Ranking([100, 80, 80, 70], exclusive))
[(0, 100), (None, 80), (None, 80), (1, 70)]
```

CHAPTER 4

Installing

The package is available in PyPI. To install it in your system, use *easy_install*:

```
$ easy_install ranking
```

Or check out developement version:

```
$ git clone git://github.com/sublee/ranking.git
```

CHAPTER 5

Changelog

5.1 Version 0.3.2

Released on Oct 9th 2020.

- Stopped supporting Python 2.5, 2.6, 3.2, and 3.3.
- Supported Python 3.4, 3.5, 3.6, 3.7, and 3.8.

Thanks to Nolan Gilley.

5.2 Version 0.3.1

Released on Mar 4th 2013.

Fixes an error of *`FRACTIONAL()`* strategy with multiple tie ranks. Thanks to Yunwon Jeong.

5.3 Version 0.3

Released on Jan 30th 2013.

Doesn't support old `cmp` style just like Python 3.

- Removes‘‘cmp‘‘ parameter from *`Ranking`*.
- Removes `score_comparer()`.
- `key`, `reverse`, `no_score` parameter of *`Ranking`* is now keyword-only.

5.4 Version 0.2.4

Released on Jan 9th 2013.

Supports Python 3 and Jython.

5.5 Version 0.2.3

Released on Oct 26th 2012.

- Adds `no_score` parameter to use other value for "no score" instead of `None`.
- Implements `score_comparer()`.

5.6 Version 0.2.2

Released on Oct 26th 2012.

- Adds `reverse` parameter for *`Ranking`*.
- Adds *`Ranking.rank()`* to find the rank of the value.

5.7 Version 0.2.2

Released on Oct 26th 2012.

- Adds `reverse` parameter for *`Ranking`*.
- Adds *`Ranking.rank()`* to find the rank of the value.

5.8 Version 0.1.2

Released on Oct 9th 2012.

- Uses `None` for "no score".
- Works with `sequence` as an iterator instead of only a list or tuple object.
- *`Ranking`* yields values instead of found scores.

5.9 Version 0.1.1

Released on Oct 7th 2012.

- Adds `start` parameter for *`Ranking`*.
- Renames `score` parameter for *`Ranking`* to `key`.

5.10 Version 0.1

First public preview release.

CHAPTER 6

Licensing and Author

This project is licensed under BSD. See LICENSE for the details.

I'm Heungsub Lee, a game developer. Any regarding questions or patches are welcomed.

Index