
DRAGen.Substructure.stats

Release 1.1

Linghao Kong

May 04, 2022

CONTENTS:

1	stats	1
1.1	stats package	1
2	Indices and tables	3
	Python Module Index	5
	Index	7

1.1 stats package

1.1.1 Submodules

1.1.2 stats.preprocessing module

Time: 2022/5/3 10:22 Author: Linghao Kong Version: V 0.1 File: sample Describe: Writing for DRAGen

class stats.preprocessing.InputDataSampler (data)

Bases: stats.preprocessing.Sampler

fit the real distribution in the data using KDE and sample from fitted distribution

pdf (x)

probability density function for fitted distribution, return the density of the input x

sample (intervals: list)

intervals: sample data from this interval in the fitted distribution

class stats.preprocessing.Sampler (markov_matrix)

Bases: object

base Sampler class: markov_matrix may be introduced in the future to increase sampling efficiency, so far it is set to be None in childclass.

classmethod rejection_sample (intervals: list, pdf, c)

use acceptance-rejection sampling to achieve complex sampling

class stats.preprocessing.SamplerFactory (type)

Bases: object

Factory Design Pattern: avoid tons of if-else, produce sampler objects depending on passed parameters

create_sampler (sampler: [**class** 'stats.preprocessing.InputDataSampler'>, **class** 'stats.preprocessing.UserPakVolumeSampler'>, **class** 'stats.preprocessing.UserBlockThicknessSampler'>], **kwargs)

sampler: must be one of the following samplers: InputDataSampler, UserPakVolumeSampler, UserBlockThicknessSampler. kwargs: the needed parameters for above mentioned 3 samplers. For example, the kwargs for UserBlockThicknessSampler should be average_bt and sigma(defaulted 0.1)

class stats.preprocessing.UserBlockThicknessSampler (average_bt, sigma=0.1)

Bases: stats.preprocessing.Sampler

produce lognorm distribution based on user input: average_bt(average block thickness), sigma(standard variance)

```
pdf (x)  
    probability density function of produced lognorm distribution  
  
sample (intervals: list)  
    intervals: sample data from this interval in the produced distribution  
  
class stats.preprocessing.UserPakVolumeSampler (equiv_d, circularity=1, sigma=0.1)  
    Bases: stats.preprocessing.Sampler  
  
    produce lognorm distribution  
  
    pdf (x)  
        probability density function of produced lognorm distribution  
  
    sample (intervals: list)  
        intervals: sample data from this interval in the produced distribution  
  
stats.preprocessing.test_inputsampler ()  
    test function: test the InputDataSampler, the user usually needs to change the file path  
  
stats.preprocessing.test_usersampler ()  
    test function: test the UserBlockThicknessSampler or UserPakVolumeSampler, the user usually needs to change  
    the file path and the kind of sampler
```

1.1.3 Module contents

Time: 2022/5/3 10:21 Author: Linghao Kong Version: V 0.1 File: __init__.py Describe: Writing at home

C

`create_sampler()` (*stats.preprocessing.SamplerFactory*
method), 1

I

`InputDataSampler` (*class in stats.preprocessing*), 1

P

`pdf()` (*stats.preprocessing.InputDataSampler* *method*),
1

`pdf()` (*stats.preprocessing.UserBlockThicknessSampler*
method), 1

`pdf()` (*stats.preprocessing.UserPakVolumeSampler*
method), 2

R

`rejection_sample()` (*stats.preprocessing.Sampler*
class method), 1

S

`sample()` (*stats.preprocessing.InputDataSampler*
method), 1

`sample()` (*stats.preprocessing.UserBlockThicknessSampler*
method), 2

`sample()` (*stats.preprocessing.UserPakVolumeSampler*
method), 2

`Sampler` (*class in stats.preprocessing*), 1

`SamplerFactory` (*class in stats.preprocessing*), 1

`stats` (*module*), 2

`stats.preprocessing` (*module*), 1

T

`test_inputsampler()` (*in module*
stats.preprocessing), 2

`test_usersampler()` (*in module*
stats.preprocessing), 2

U

`UserBlockThicknessSampler` (*class in*
stats.preprocessing), 1

`UserPakVolumeSampler` (*class in*
stats.preprocessing), 2