

---

# GaussBetti

*Release 2.0.0*

**Siddharth**

**Jul 27, 2021**



# CONTENTS

<b>1</b>	<b>topologicalFunc</b>	<b>1</b>
<b>2</b>	<b>Utilities</b>	<b>3</b>
<b>3</b>	<b>gaussClass</b>	<b>7</b>
<b>4</b>	<b>rocGen</b>	<b>9</b>
<b>5</b>	<b>Indices and tables</b>	<b>11</b>
	<b>Python Module Index</b>	<b>13</b>
	<b>Index</b>	<b>15</b>



## TOPOLOGICALFUNC

`topologicalFunc.GaussianFiltration(GaussianRandomField, type='lower')`

Generates Filtration for the Gaussian Random Field.

**Parameters**

- **GaussianRandomField** (*array*) – numpy 2-D array. The Gaussian Random Field generated from the class using `Gen_GRF` method.
- **type** (*string*) – Takes input either 'lower' or 'upper' for lower or upper filtration.
- **nsiz** (*integer*) – Size of the Gaussian Random Fields grid.

**Returns** Filtration Diagram

**Return type** Dionysus object

`topologicalFunc.GenerateBettiP(Filtration, thresholds_start, thresholds_stop, type='lower')`

Generates the Betti numbers from the Filtration diagram.

**Parameters**

- **Filtration** (*Dionysus object*) – Output of GaussianFiltration.
- **thresholds\_start** (*float*) – start value for generating superlevels of the Gaussian Random field .
- **thresholds\_stop** (*float*) – stop value for generating superlevels of the Gaussian Random field.

**Returns** Multidimensional array containing Betti numbers for different dimensions

**Return type** Numpy array

`topologicalFunc.GenerateGenus(Betti_array)`

Generates the Genus curve for gaussian random field using Betti arrays.

**Parameters** **array** (*Betti*) – Betti array from GenerateBettiP.

**Returns** 1-D array containing Genus curve for the Gaussian random field.

**Return type** Numpy array



## UTILITIES

`utilities.Generate_BettiGenus_array(Nsize, power_index_null, power_index_test, average, iteration, filtration_threshold_start, filtration_threshold_stop, type1='lower')`

Generate\_Likelihood\_Array

Generates the Betti and Genus curves for specified parameters.

### Parameters

- **Nsize** (*integer*) – grid size of the Gaussian Random Field
- **power\_index\_null** (*float*) – Power spectral index of Null Hypothesis
- **power\_index\_test** (*float*) – Power spectral index of Test Hypothesis
- **average** (*integer*) – No. of times the betti curves need to be averaged
- **iteration** (*integer*) – Size of the arrays generated
- **filtration\_threshold\_start** (*float*) – Start value for generating filtraion from dionysus
- **filtration\_threshold\_stop** (*float*) – Stop value for generation filtration from dionysus
- **type** – Type of filtration accepted values are 'lower' 'upper'

**Returns** array of Betti and Genus curves

**Return type** numpy array

`utilities.Generate_Likelihood_Array(Nsize, power_index_null, power_index_test, iteration)`

Generates the array of likelihood ratios for making ROC curves.

### Parameters

- **Nsize** (*integer*) – grid size of the Gaussian Random Field
- **power\_index\_null** (*float*) – Power spectral index of Null Hypothesis
- **power\_index\_test** (*float*) – Power spectral index of Test Hypothesis
- **iteration** (*integer*) – Size of the likelihood ratio array generated

**Returns** array of likelihood ratios

**Return type** numpy array

`utilities.KLdivergence(x, y1, y2)`

Calculates the KL divergence for 2 different Gaussian Random Field.

### Parameters

- **x** (*array*) –
- **y1** (*array*) – Gaussian Random Field of null hypothesis as a 1-D array
- **y2** (*array*) – Gaussian Random Field of test hypothesis as a 1-D array

**Returns** KL divergence

**Return type** float

`utilities.plotROC(PFA, PD, nsize, num_iter, H0, H1, type1, Betti='default')`

Plots the PFA and PD ROC graph with the labels provided through parameters.

**Parameters**

- **PFA** (*array*) – numpy vector. The PFA array generated during ROC gen.
- **PD** (*array*) – numpy vector. The PD array generated during ROC gen.
- **nsize** (*integer*) – Size of the Gaussian Random Fields grid.
- **num\_iter** (*integer*) – Number of iteration for which ROC gen is run.
- **H0** (*float*) – Power spectral index of Null Hypothesis.
- **H1** (*float*) – Power spectral index of Test Hypothesis.
- **type1** (*string*) – type1 of the ROC curve generated takes value 'likelihood','betti','genus'
- **Betti** (*integer*) – Dimension of Betti curve not needed when type = likelihood

**Returns** None

**Return type** None

`utilities.readROC(nsize, num_iter, H0, H1, type1, Betti='default')`

Reads the PFA and PD array from the files generated using saveROC.

**Parameters**

- **nsize** (*integer*) – Size of the Gaussian Random Fields grid.
- **num\_iter** (*integer*) – Number of iteration for which ROC gen is run.
- **H0** (*integer*) – Power spectral index of Null Hypothesis.
- **H1** (*integer*) – Power spectral index of Test Hypothesis.
- **type1** (*string*) – type1 of the ROC curve generated takes value 'likelihood','betti','genus'
- **Betti** (*integer*) – Dimension of Betti curve not needed when type = likelihood

**Returns** Returns PFA and PD arrays

**Return type** Numpy Array

`utilities.saveROC(PFA, PD, nsize, num_iter, H0, H1, type1, Betti='default')`

SaveROC

Saves the PFA and PD array with the labels provided through parameters.

**Parameters**

- **PFA** (*array*) – numpy vector. The PFA array generated during ROC gen.
- **PD** (*array*) – numpy vector. The PD array generated during ROC gen.
- **nsize** (*integer*) – Size of the Gaussian Random Fields grid.
- **num\_iter** (*integer*) – Number of iteration for which ROC gen is run.



- **H0** (*float*) – Power spectral index of Null Hypothesis.
- **H1** (*float*) – Power spectral index of Test Hypothesis.
- **type1** (*string*) – type1 of the ROC curve generated takes value ‘likelihood’, ‘betti’, ‘genus’
- **Betti** (*integer*) – Dimension of Betti curve not needed when type = likelihood

**Returns** None

**Return type** None



## GAUSSCLASS

```
class gaussClass.GaussianRandomField(Nsize, n)
    The class for making Gaussian random field with specified spectral index and size of grid.

    Nsize
        size of the grid.
        Type int

    n
        Spectral index of the power law used to generate the Gaussian Random Field.
        Type int

    k_ind
        Grid in the fourier space.
        Type array

    PowerSpectrum
        The power spectrum grid made using the spectral index used to make the Gaussian Random Field.
        Type array

    corr_s
        Correlation matrix in the fourier space.
        Type array

    corr_f
        Correlation matrix in the spatial space.
        Type array

    Gen_GRF(type='grid')
        GenerateBettiP

        Generates the Gaussian Random field with the specified paramters.

        Parameters type (str) – Takes either ‘grid’ or ‘array’ in string format

        Returns: Numpy array: Gaussian Random field

    PowerSpectrum_grid_generator()
        Generates the powerspectrum grid.

    fourier_space_ind()
        Generates the fourier space grid.

    gen_correlation()
        Generates the correlation matrices in fourier and spatial spcae.
```



## ROCGEN

`rocGen.BettiROC(Betti_array0, Betti_array1, threshold_step)`

Generates PFA and PD values from Betti curves using the specified parameters

**Parameters**

- **Betti\_array0** (*array*) – multiple Betti curves for null hypothesis generated from GenerateBetti function for a one dimension.
- **Betti\_array1** (*array*) – multiple Betti curves for test hypothesis generated from GenerateBetti function for a one dimension.
- **threshold\_step** (*float*) – Step value for generating threshold array.

**Returns** PFA and PD values

**Return type** Array

`rocGen.GenusROC(Genus_array0, Genus_array1, threshold_step)`

Generates PFA and PD values from Genus curves using the specified parameters

**Parameters**

- **Genus\_array0** (*array*) – multiple Genus curves for null hypothesis generated from GenerateBetti function for a one dimension.
- **Genus\_array1** (*array*) – multiple Genus curves for test hypothesis generated from GenerateBetti function for a one dimension.
- **index** (*integer*) – index at which the Genus value is calculated and compared with threshold.
- **threshold\_step** (*float*) – Step value for generating threshold array.

**Returns** PFA and PD values

**Return type** Array

`rocGen.LikelihoodROC(likelihoodratio0, likelihoodratio1, threshold_step)`

Generates PFA and PD values from likelihood ratios of the Gaussian Random Field using the specified parameters

**Parameters**

- **likelihoodratio0** (*array*) – arrays containing likelihood ratios of the Gaussian Random field of type 1 see doc for likelihoodratio .
- **likelihoodratio1** – 2-D arrays containing likelihood ratios of the Gaussian Random field of type 2 see doc for likelihoodratio
- **threshold\_step** (*float*) – Step value for generating threshold array.

**Returns** PFA and PD values

**Return type** Array

## INDICES AND TABLES

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





## PYTHON MODULE INDEX

### g

gaussClass, [7](#)

### r

rocGen, [9](#)

### t

topologicalFunc, [1](#)

### u

utilities, [3](#)



## INDEX

### B

BettiROC() (in module rocGen), 9

### C

corr\_f (gaussClass.GaussianRandomField attribute), 7

corr\_s (gaussClass.GaussianRandomField attribute), 7

### F

fourier\_space\_ind() (gauss-  
Class.GaussianRandomField method), 7

### G

gaussClass  
module, 7

GaussianFiltration() (in module topologicalFunc), 1

GaussianRandomField (class in gaussClass), 7

gen\_correlation() (gauss-  
Class.GaussianRandomField method), 7

Gen\_GRF() (gaussClass.GaussianRandomField method),  
7

Generate\_BettiGenus\_array() (in module utilities),  
3

Generate\_Likelihood\_Array() (in module utilities),  
3

GenerateBettiP() (in module topologicalFunc), 1

GenerateGenus() (in module topologicalFunc), 1

GenusROC() (in module rocGen), 9

### K

k\_ind (gaussClass.GaussianRandomField attribute), 7

KLdivergence() (in module utilities), 3

### L

LikelihoodROC() (in module rocGen), 9

### M

module  
gaussClass, 7  
rocGen, 9  
topologicalFunc, 1  
utilities, 3

### N

n (gaussClass.GaussianRandomField attribute), 7

Nzise (gaussClass.GaussianRandomField attribute), 7

### P

plotROC() (in module utilities), 4

PowerSpectrum (gaussClass.GaussianRandomField at-  
tribute), 7

PowerSpectrum\_grid\_generator() (gauss-  
Class.GaussianRandomField method), 7

### R

readROC() (in module utilities), 4

rocGen  
module, 9

### S

saveROC() (in module utilities), 4

### T

topologicalFunc  
module, 1

### U

utilities  
module, 3