# ECL DOCUMENTATION

Go Up

| | |
|---|---|
| Name | GaussianProcessRegression |
| Version | 1.0 |
| Description | RFF-accelerated Gaussian Process Regression |
| License | http://www.apache.org/licenses/LICENSE-2.0 |
| Copyright | Copyright (C) 2022 HPCC Systems® |
| Authors | HPCCSystems |
| DependsOn | ML_Core |
| Platform | 8.4.0 |

# OVERVIEW

## Gaussian Process Regression (GPR)

This bundle provides a Random Fourier Features accelerated version of Gaussian Process Regressor. It allows Data Scientist, researchers or software programmers to apply Gaussian Process Regressor in the parallelized environment of HPCC Systems.

Random Fourier Features(RFF) map the input data to a randomized low-dimensional feature space. Then one can apply fast existing linear methods to such new space and thus accelerate the training of large scale kernel machines[1]. This bundle is the accelerated version of Gaussian Process Regression(GPR) using such random fourier features.

The module GPRI is the main ECL interface. Three functions are available to the users: getSession, fit and predict.

- getSession function generates a ‘session ID’ for the training and predict process. <br />
- fit function fits the input data and train a GPR model. <br />
- predict funcion uses the trained GPR model to make predictions for the new observations.<br />

For details of each function, see the comments below above each function in GPRI.ecl file. For details of record structure used in these functions, see Types.ecl file. For usage examples of GPR bundle, see the test cases in Test/test.ecl file.

To use GPR bundle, ‘session ID’ is required to feed to each fit or predict function call. However, if the training and predict process are in the same session/workunit, getSession only needs to be called once, i.e. fit and predict share same ‘session ID’ in this case.

## INSTALLATION

Python3 must be installed on each node of HPCC Systems. ML_Core bundle from HPCC Systems Machine Learning Library should be installed as well. To install GPR bundle, run following command via HPCC Systems client tool:

ecl bundle install https://github.com/hpcc-systems/GaussianProcessRegression.git

## EXAMPLES

Test examples are included in the Test folder. Under Test folder, test.ecl file shows the process to define session ID, fit a GPR model and make predictions. The testing data is generated by M_dataGen.ecl file which generates random test data by user defined size.

## OTHER DOCUMENTATIONS

HPCC Systems Machine Learning Library

Using HPCC Systems Machine Learning Library

## ACKNOWLEDGEMENT

This bundle is built upon the orignal python implementaton of GPR module from below source:
https://github.com/gwgundersen/random-fourier-features

## REFERENCE

[1] Ali Rahimi and Benjamin Recht. 2007. Random features for large-scale kernel machines. In Proceedings of the 20th International Conference on Neural Information Processing Systems (NIPS'07). Curran Associates Inc., Red Hook, NY, USA, 1177–1184.

# Table of Contents

# GPRI

---

Go Up

## IMPORTS

python3 | _versions.ML_Core.V3_2_2.ML_Core.Types | std.system.Thorlib | Types | Internal.rffGPR |

## DESCRIPTIONS

### GPRI

| | |
|---|---|
| | **GPRI** |

No Documentation Found

**Children**

1. GetSession : Initialize GPR on all nodes and return a session ID to be used in the following process
2. fit : Train a RFF acclerated GPR model
3. predict : Predict using trained GPR model

---

### GETSESSION

GPRI /

| INTEGER | **GetSession** |
|---|---|
| () | |

Initialize GPR on all nodes and return a session ID to be used in the following process. This function needs to be called before any other process.

**RETURN** **INTEGER8** — sessID session ID to identify this session.

---

## FIT

GPRI /

| DATASET(Layout_model2) | **fit** |
|---|---|
| (INTEGER session, DATASET(NumericField) x, DATASET(NumericField) y, UNSIGNED4 rff_dim = 10, REAL sigma = 1) | |

Train a RFF acclerated GPR model

**PARAMETER** **session** ||| INTEGER8 — No Doc

**PARAMETER** **x** ||| TABLE ( NumericField ) — No Doc

**PARAMETER** **y** ||| TABLE ( NumericField ) — No Doc

**PARAMETER** **rff_dim** ||| UNSIGNED4 — No Doc

**PARAMETER** **sigma** ||| REAL8 — No Doc

**RETURN** **TABLE ( layout_model2 )** — Gaussian process regression model in Layout_model2 foramt.

**SEE** ML_Core.Types.Layout_Model2

**PARAMS** session session ID for the training process.

**PARAMS** x independent training data.

**PARAMS** y dependent training data.

**PARAMS** rff_dim dimesion of random fourier features.

**PARAMS** sigma squre root of the variance.

---

# PREDICT

GPRI /

| DATASET(NumericField) | **predict** |
|---|---|
| (INTEGER session, DATASET(Layout_model2) mod, DATASET(NumericField) x) | |

Predict using trained GPR model

**PARAMETER** **session** ||| INTEGER8 — No Doc

**PARAMETER** **mod** ||| TABLE ( layout_model2 ) — No Doc

**PARAMETER** **x** ||| TABLE ( NumericField ) — No Doc

**RETURN** **TABLE ( NumericField )** — prediction result in NumericField format

**SEE** ML_Core.Types.NumericField

**PARAMS** session session ID for the predicting process.

**PARAMS** mod trained GPR model.

**PARAMS** x input data for prediction.

---

# Types

---

Go Up

## DESCRIPTIONS

### TYPES

**Types**

No Documentation Found

**Children**

1. initParams : No Documentation Found

---

### INITPARAMS

Types /

**initParams**

No Documentation Found

FIELD **nodeid** ||| UNSIGNED4 — No Doc

FIELD **nnodes** ||| UNSIGNED4 — No Doc

---

# Internal

---

Go Up

## Table of Contents

Internal/

# rffGPR

---

Go Up

## IMPORTS

python3 | _versions.ML_Core.V3_2_2.ML_Core.Types | std.system.Thorlib | Types |

## DESCRIPTIONS

### RFFGPR

**rffGPR**

No Documentation Found

**Children**

1. init : No Documentation Found
2. fit : No Documentation Found
3. predict : No Documentation Found

---

### INIT

rffGPR /

| STREAMED DATASET({INTEGER sessID}) | **init** |
|---|---|
| (STREAMED DATASET(initParams) initDat, STRING wuid = WORKUNIT) | |

No Documentation Found

**PARAMETER** **initdat** ||| TABLE ( initParams ) — No Doc

**PARAMETER** **wuid** ||| STRING — No Doc

**RETURN** **TABLE ( { INTEGER8 sessID } )** —

---

## FIT

rffGPR /

| DATASET(Layout_model2) | **fit** |
|---|---|
| (INTEGER session, DATASET(NumericField) x, DATASET(NumericField) y, UNSIGNED4 dim = 10, REAL sig = 1) | |

No Documentation Found

**PARAMETER** **session** ||| INTEGER8 — No Doc

**PARAMETER** **x** ||| TABLE ( NumericField ) — No Doc

**PARAMETER** **y** ||| TABLE ( NumericField ) — No Doc

**PARAMETER** **dim** ||| UNSIGNED4 — No Doc

**PARAMETER** **sig** ||| REAL8 — No Doc

**RETURN** **TABLE ( layout_model2 )** —

---

# PREDICT

rffGPR /

| STREAMED DATASET(NumericField) | **predict** |
|---|---|
| (STREAMED DATASET(Layout_model2) mod, STREAMED DATASET(NumericField) x, INTEGER session) | |

No Documentation Found

**PARAMETER** **mod** ||| TABLE ( layout_model2 ) — No Doc

**PARAMETER** **x** ||| TABLE ( NumericField ) — No Doc

**PARAMETER** **session** ||| INTEGER8 — No Doc

**RETURN** **TABLE ( NumericField )** —

---

# Test

---

Go Up

## Table of Contents

Test/

# M_dataGen

---

Go Up

## IMPORTS

python3 | _versions.ML_Core.V3_2_2.ML_Core.Types |

## DESCRIPTIONS

### M_DATAGEN

| a EXPORT | **M_dataGen** |
|---|---|
| (INTEGER n, INTEGER n_train) | |

No Documentation Found

PARAMETER **n** ||| INTEGER8 — No Doc

PARAMETER **n_train** ||| INTEGER8 — No Doc

**Children**

1. l : No Documentation Found
2. toNF : No Documentation Found
3. generateXData : No Documentation Found
4. x : No Documentation Found

5. generateYData : No Documentation Found
6. y : No Documentation Found
7. X_train : No Documentation Found
8. Y_train : No Documentation Found
9. X_test : No Documentation Found
10. Y_test : No Documentation Found

---

## L

M_dataGen /

| | l |
|---|---|

No Documentation Found

FIELD **x** ||| SET ( REAL8 ) — No Doc

---

## TONF

M_dataGen /

| | toNF |
|---|---|
| (set of REAL input) | |

No Documentation Found

PARAMETER **input** ||| SET ( REAL8 ) — No Doc

RETURN **TABLE ( { UNSIGNED2 wi , UNSIGNED8 id , UNSIGNED4 number , REAL8 value } )** —

---

## GENERATEXDATA

M_dataGen /

| set of real | **generateXData** |
|---|---|
| (INTEGER n) | |

No Documentation Found

**PARAMETER** **n** ||| INTEGER8 — No Doc

**RETURN** **SET ( REAL8 )** —

---

## X

M_dataGen /

| | **x** |
|---|---|

No Documentation Found

---

## GENERATEYDATA

M_dataGen /

| set of real | **generateYData** |
|---|---|
| (set of real x_data) | |

No Documentation Found

**PARAMETER** **x_data** ||| SET ( REAL8 ) — No Doc

**RETURN** **SET ( REAL8 )** —

---

## Y

M_dataGen /

| | |
|---|---|
| | **y** |

No Documentation Found

---

## X_TRAIN

M_dataGen /

| | |
|---|---|
| ; EXPORT | **X_train** |

No Documentation Found

---

## Y_TRAIN

M_dataGen /

| | |
|---|---|
| ; EXPORT | **Y_train** |

No Documentation Found

---

## X_TEST

M_dataGen /

| | |
|---|---|
| ; EXPORT | **X_test** |

No Documentation Found

# Y_TEST

M_dataGen /

| ; EXPORT | **Y_test** |
|---|---|

No Documentation Found

Test/

# pyGPR

---

Go Up

## IMPORTS

python3 | _versions.ML_Core.V3_2_2.ML_Core.Types | Types |

## DESCRIPTIONS

### PYGPR

| DATASET(NumericField) | **pyGPR** |
|---|---|
| (DATASET(NumericField) x, DATASET(NumericField) y) | |

No Documentation Found

**PARAMETER** **x** ||| TABLE ( NumericField ) — No Doc

**PARAMETER** **y** ||| TABLE ( NumericField ) — No Doc

**RETURN** **TABLE ( NumericField )** —

---

Test/

# score

---

Go Up

## IMPORTS

_versions.ML_Core.V3_2_2.ML_Core | _versions.ML_Core.V3_2_2.ML_Core.Types | _versions.PBblas.V3_0_2.PBblas | _versions.PBblas.V3_0_2.PBblas.Types | _versions.PBblas.V3_0_2.PBblas.Converted | _versions.PBblas.V3_0_2.PBblas.MatUtils | _versions.ML_Core.V3_2_2.ML_Core.Math |

## DESCRIPTIONS

### SCORE

| | score |
|---|---|
| | `(DATASET(NumericField) X=empty_data, DATASET(NumericField) y=empty_data, DATASET(NumericField) Yhat=empty_data)` |

No Documentation Found

PARAMETER **x** ||| TABLE ( NumericField ) — No Doc

PARAMETER **y** ||| TABLE ( NumericField ) — No Doc

PARAMETER **yhat** ||| TABLE ( NumericField ) — No Doc

**Children**

1. sumX : No Documentation Found
2. sumy : No Documentation Found
3. sumYY : No Documentation Found
4. n : No Documentation Found
5. x2 : No Documentation Found
6. sumX2 : No Documentation Found
7. y2 : No Documentation Found
8. sumY2 : No Documentation Found
9. p1 : No Documentation Found
10. p2 : No Documentation Found
11. r : No Documentation Found
12. r2 : No Documentation Found

---

## SUMX

score /

> **sumX**

No Documentation Found

**RETURN** **REAL8** —

---

## SUMY

score /

> **sumy**

No Documentation Found

**RETURN** **REAL8** —

---

## SUMYY

score /

| | |
|---|---|
| | **sumYY** |

No Documentation Found

**RETURN** **REAL8** —

---

## N

score /

| | |
|---|---|
| | **n** |

No Documentation Found

**RETURN** **INTEGER8** —

---

## X2

score /

| | |
|---|---|
| | **x2** |

No Documentation Found

**RETURN** **TABLE ( { UNSIGNED2 wi , UNSIGNED8 id , UNSIGNED4 number , REAL8 value } ) —**

---

## SUMX2

score /

```
sumX2
```

No Documentation Found

**RETURN** **REAL8 —**

---

## Y2

score /

```
y2
```

No Documentation Found

**RETURN** **TABLE ( { UNSIGNED2 wi , UNSIGNED8 id , UNSIGNED4 number , REAL8 value } ) —**

---

## SUMY2

score /

```
sumY2
```

No Documentation Found

**RETURN** **REAL8** —

---

## P1

score /

| | |
|---|---|
| | **p1** |

No Documentation Found

**RETURN** **REAL8** —

---

## P2

score /

| | |
|---|---|
| | **p2** |

No Documentation Found

**RETURN** **REAL8** —

---

## R

score /

| | |
|---|---|
| | **r** |

No Documentation Found

**RETURN** **REAL8** —

---

## R2

score /

| | |
|---|---|
| | **r2** |

No Documentation Found

**RETURN** **REAL8** —

---