GPCorrelation

De Wiki
Aller à : navigation, rechercher
GPCorrelation

This widget allows to manage **"pseudo correlation"** or **covariance** matrices.

Why **"pseudo correlation"** ? In fact a real correlation matrix has its diagonal only with values of "1", the sigma values being given independantly. Here, we will consider a **"pseudo correlation"** with the diagonal filled with these sigma values !

Sommaire

- 1 How to call it
- 2 Display
- 3 How to use it
- 4 How it is stored

How to call it

For using the GPCorrelation class, the developer has only to create such an object as is:

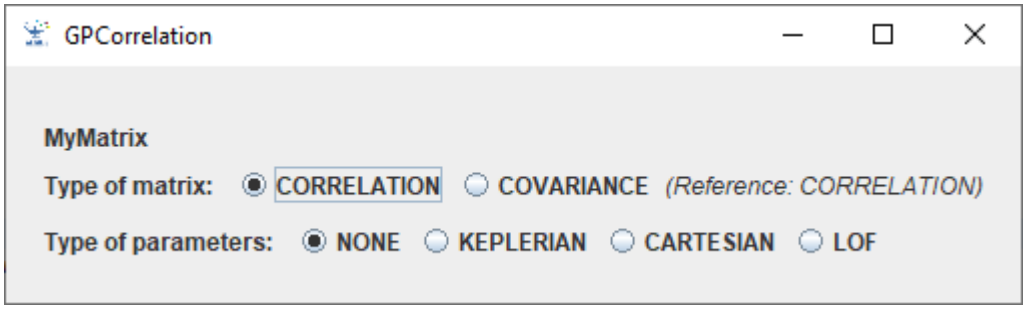
```
gpCor = new GPCorrelation();
```

or:

```
gpCor = new GPCorrelation("widgetId", "MyMatrix");
```

Display

We will get:

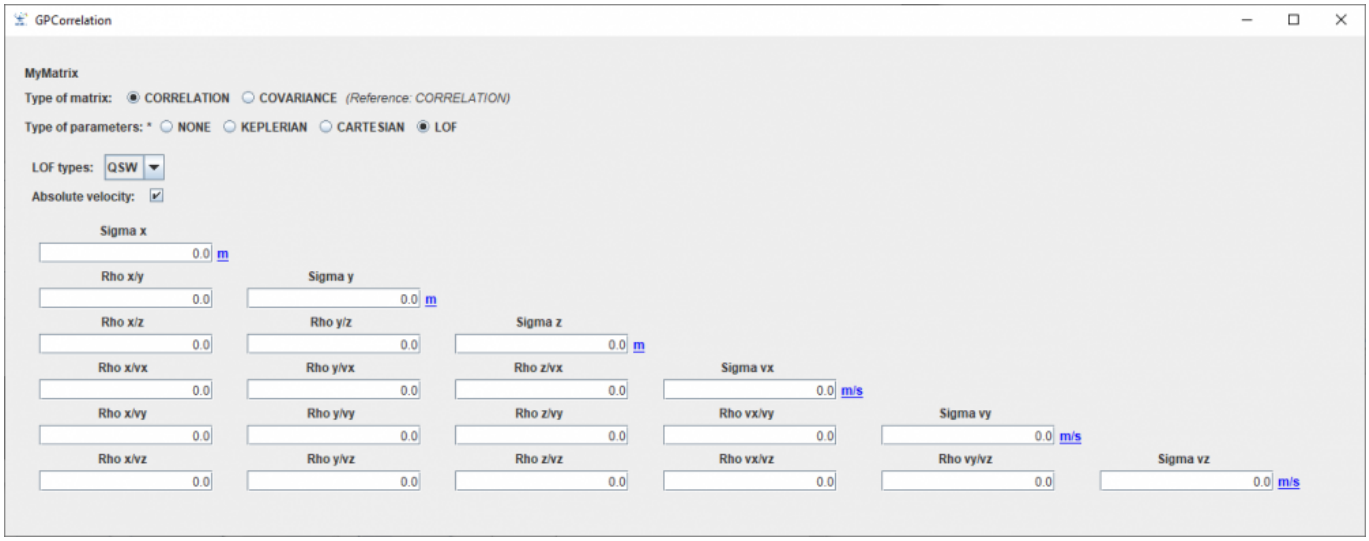


By selecting the type of parameters, we may enter data for pseudo correlation/covariance matrices for Keplerian/Cartesian/LOF cartesians parameters. The different displays will be:

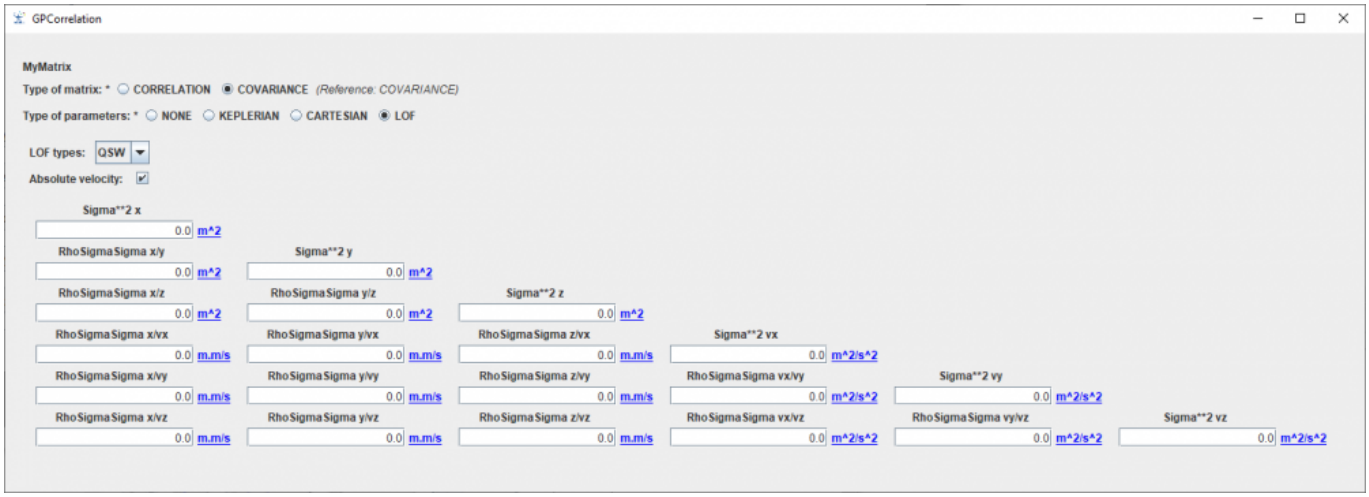
- In case of Keplerian parameters:

- In case of cartesian parameters:

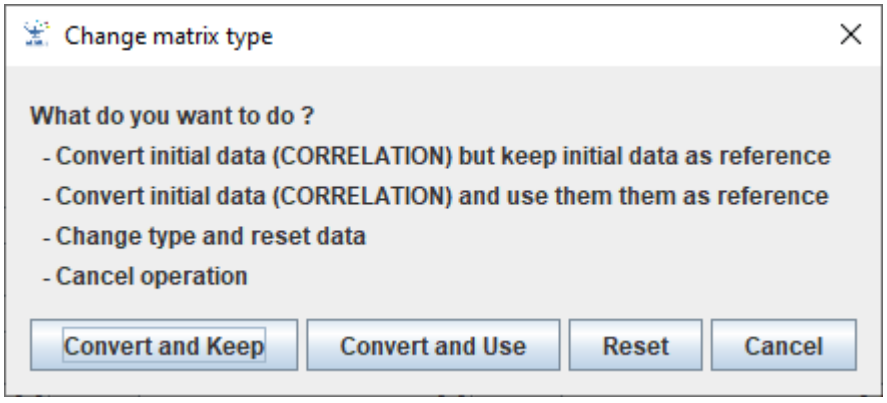
- In case of LOF cartesian parameters:



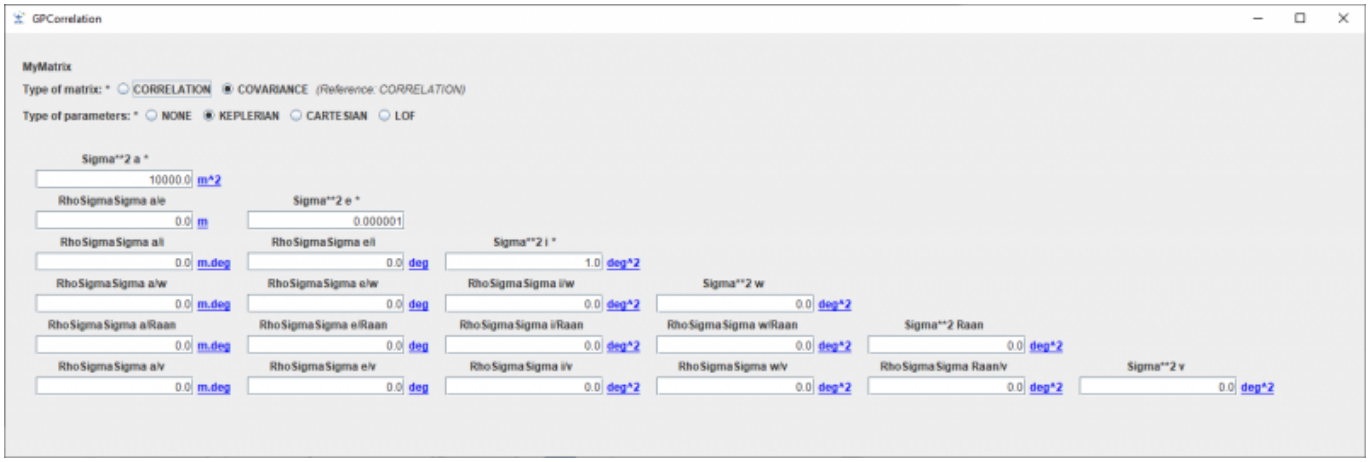
So we may have a covariance matrix rather than a pseudo correlation one:



If the initial pseudo correlation (resp. covariance) matrix is not null, a specific pop-up window will appear to ask for the type of conversion. A bit as for the GPOrbit widget, it will be possible to do this conversion between pseudo correlation and covariance matrices (but not between the different type of parameters) using a kind of “pivot” notion.



=>



How to use it

There are no exactly corresponding correlation/covariance matrices inside PATRIUS but **GENOPUS** gives some utility classes and methods to use them.

In the examples below, we can recover the widget data inside PATRIUS RealMatrix or Realvector objects.

```
final GPCorrelationData data = gpCor.getGpCorrelationData();

final RealMatrix corMat = data.getCorrelationMatrix();
final RealMatrix covMat = data.getCovarianceMatrix();
final RealMatrix sqrtRootcovMat = data.getSqrtCovarianceMatrix();
final RealVector sigma = data.getSigma();
```

We have also the possibility to use these utility methods:

```
final RealMatrix mat1 = GPCorrelationUtils.corToCov(sigma, corMat);
final RealMatrix mat2 = GPCorrelationUtils.pseudoCorToCov(corMat);
final RealMatrix mat3 = GPCorrelationUtils.covToPseudoCor(covMat);
```

How it is stored

Here is an example of the XML format:

```
  <!--Type of matrix:-->
  <String name="matrixType">CORRELATION</String>
  <!--Type of parameters:-->
  <String name="type">LOF</String>
  <!--LOF types:-->
  <String name="lofType">LVLH</String>
  <!--Absolute velocity:-->
  <Boolean name="velocityType">true</Boolean>
  <Real name="Sigma_x" unit="m">1.0E1</Real>
  <Real name="Rho_x/y">5.0E-1</Real>
  <Real name="Rho_x/z">0.0E0</Real>
```

```
  <Real name="Rho_x/vx">0.0E0</Real>
  <Real name="Rho_x/vy">0.0E0</Real>
  <Real name="Rho_x/vz">0.0E0</Real>
  <Real name="Sigma_y" unit="m">2.0E4</Real>
  <Real name="Rho_y/z">0.0E0</Real>
  <Real name="Rho_y/vx">0.0E0</Real>
  <Real name="Rho_y/vy">0.0E0</Real>
  <Real name="Rho_y/vz">0.0E0</Real>
  <Real name="Sigma_z" unit="m">3.0E4</Real>
  <Real name="Rho_z/vx">0.0E0</Real>
  <Real name="Rho_z/vy">0.0E0</Real>
  <Real name="Rho_z/vz">0.0E0</Real>
  <Real name="Sigma_vx" unit="m/s">1.0E2</Real>
  <Real name="Rho_vx/vy">0.0E0</Real>
  <Real name="Rho_vx/vz">0.0E0</Real>
  <Real name="Sigma_vy" unit="m/s">2.0E2</Real>
  <Real name="Rho_vy/vz">0.0E0</Real>
  <Real name="Sigma_vz" unit="m/s">3.0E2</Real>
```


Menu de navigation

Rechercher

Rechercher Lire

GENOPUS

- Welcome
- Quick Start
- News

User Manual

- BasicPrinciples
- GPAbsoluteDate
- GPOrbit
- GPFramesConfiguration
- GPVehicle
- GPForceModels
- GPManeuverSequence
- GPAttitudeSequence
- GPIntegrator
- GPAxisCoordinates
- GPGeodeticPoint
- GPOneAxisEllipsoid
- GPRotation
- GPConstants
- Events
- GPCorrelation

Evolutions

- Main differences between V2.4.1 and V2.4.2
- Main differences between V2.3.3 and V2.4.1
- Main differences between V2.2.1 and V2.3.3
- Main differences between V2.2 and V2.2.1
- Main differences between V2.1.1 and V2.2
- Main differences between V2.1 and V2.1.1
- Main differences between V2.0.1 and V2.1
- Main differences between V2.0 and V2.0.1
- Main differences between V1.3.1 and V2.0
- Main differences between V1.3 and V1.3.1
- Main differences between V1.2.1 and V1.3

Training

- Make your own propagator tool!
- Tutorials package for V2.4.1
- Tutorials package for V2.3.3
- Tutorials package for V2.2
- Tutorials package for V2.1.1
- Tutorials package for V2.0 and V2.0.1
- Tutorials package for V1.3 and V1.3.1
- Training slides

Links

- CNES freeware server

Outils

- Pages liées
- Suivi des pages liées
- Pages spéciales
- Adresse de cette version
- Information sur la page
- Citer cette page

- Dernière modification de cette page le 13 décembre 2021 à 16:38.

- Politique de confidentialité
- À propos de Wiki
- Avertissements
-