PyRFBI

A complete framework for the Bayesian joint inversion of teleseismic converted waves arrival times and polarities.

Receiver Functions Bayesian Inversion (RFBI) is a tool written in Python to probabilistically invert teleseismic converted wave arrival times and polarities on the transverse component. It uses the Python package PyRaysum to compute synthetic receiver functions for the forward model.

Installation and requirements

You will neeed a Python environment with the module PyRaySum. I strongly recommend creating a custom conda environment where PyRaySum can be installed along with its dependencies. I define here a a minimal environment for running PyRFBI:

```
conda create -n prs python=3.8 fortran-compiler obspy pandas seaborn cmcrameri -c conda-forge
```

```
conda activate prs
```

```
pip install pyraysum
```

Then, you can download PyRFBI code from github and add its location to your PATH variable:

```
export PATH="$PATH:/absolute/path/to/PyRFBI/"
```

Usage and documentation

0. Input data

Arrival times and polarities needs to be organized in 4 files:

- pick_time.csv : arrival times
- pick_time_error.csv : arrival times uncertainties
- pick_polarity.csv : polarities
- pick_polarity_error.csv : polarities uncertainties

The nomenclatura for phases is based on PyRaySum

For non picked values, put NaN

```
rfbi_gen_input.py
```

With every command, you can use `-h` to have help about the usage of the command.

1. Generate structure

Create working directories, copy the data and create a config file (rfbi.ini):

```
rfbi_make_wkdir.py wkdir datadir
cd wkdir
```

2. Initiate

Initiate the earth model (number of layers and target parameters). You will need to change the parameters_inversion.csv file.

```
rfbi_init_invstruct.py n_layers target_parameters
```

Check that parameters_inversion.csv is correct.

```
rfbi_check_invstruct.py
```

3. Initiate

Select Metropolis or adaptative Metropolis algorithm and add parameters to config file. You will need to change the inversion and sampling parameters in the config file.

```
rfbi_init_inversion.py sampling_method
```

4. Plotting data

Plot inversion data:

```
rfbi_plot_data.py
```

5. Run the inversion

Run the inversion:

```
rfbi_invert.py
```

6. Plotting inversion results

Finally, you can plot the results of the inversion. It will create figures ...

```
rfbi_plot_inv.py
```

Contact

Feel free to contact me and ask questions at emile.denise@ens.psl.eu. I'll try to reply as soon as possible.

Contributing

All constructive contributions are welcome, e.g. bug reports, discussions or suggestions for new features.