Practical Integration of Processing, Inversion and Visualization of Magnetotelluric Geophysical Data

Gudni Rosenkjaer,
Lindsey Heagy, Rowan Cockett, Seogi Kang, Doug Oldenburg

Introduction

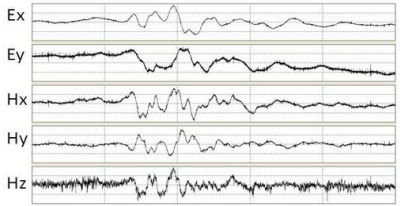
- Who am I?
- What do I do?

- Why do I use python?
- And how do I use python?

Magneto Tellurics

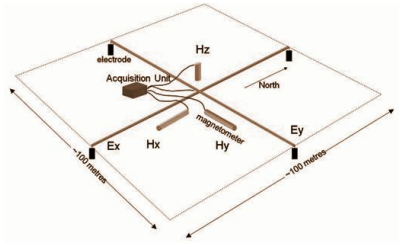
Natural Fields

input energy

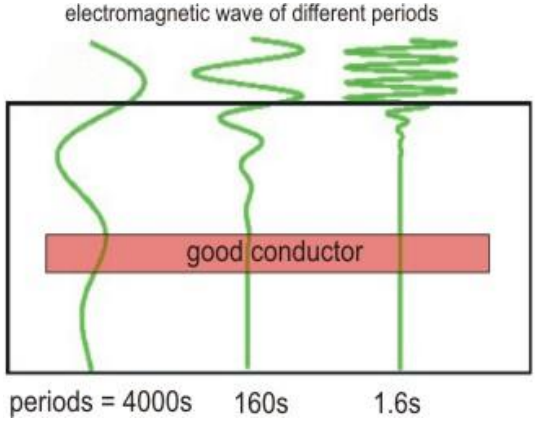


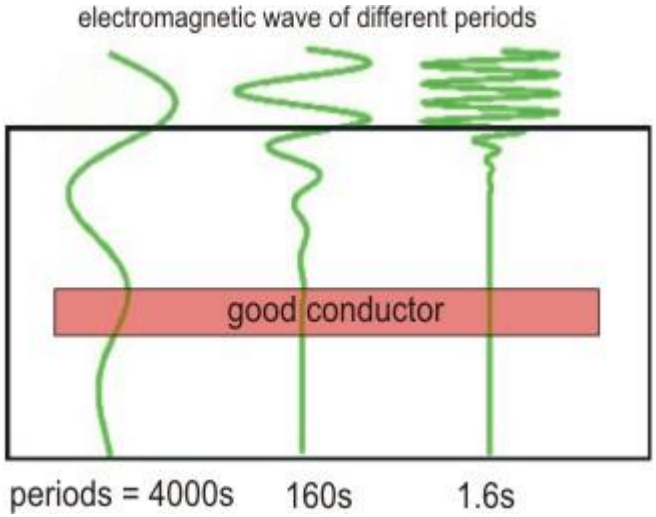
Impedance Data

measure response

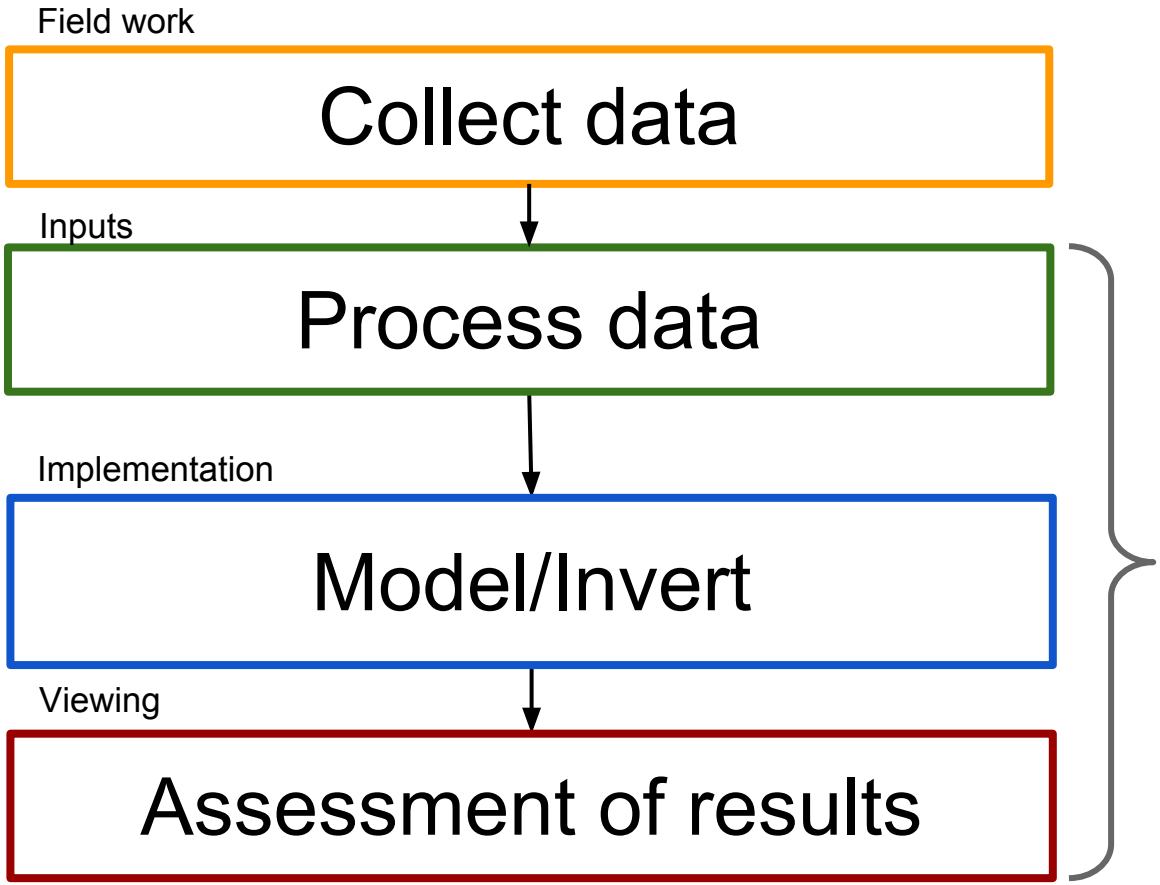


Subsurface:
Electrical Conductivity




electromagnetic wave of different periods
good conductor
periods = 4000s
160s
1.6s

Geophysical workflow

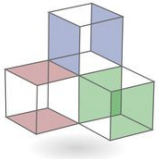


Framework with
- single platform
- easy access
- integratable

Should have
- interactive
- visual
- repeatable

capabilities

SimPEG



Simulation and Parameter Estimation in Geophysics

An open source python package for simulation and gradient based parameter estimation in geophysical applications.

pypi v0.1.3 | downloads 3k/month | license MIT | build passing | coverage 83%

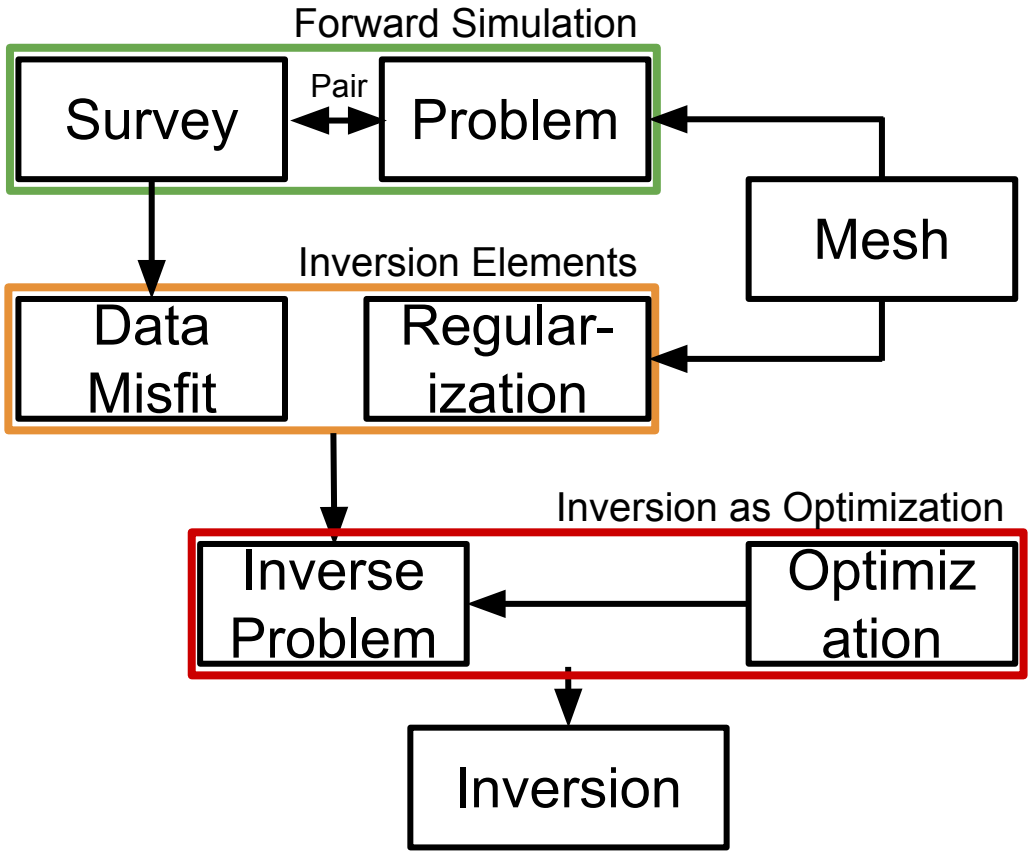
Installation

The easiest way to install SimPEG is from PyPI, using pip:

```
> pip install SimPEG
```

Read more detailed installations instructions in the documentation.

Get the source code at Github.

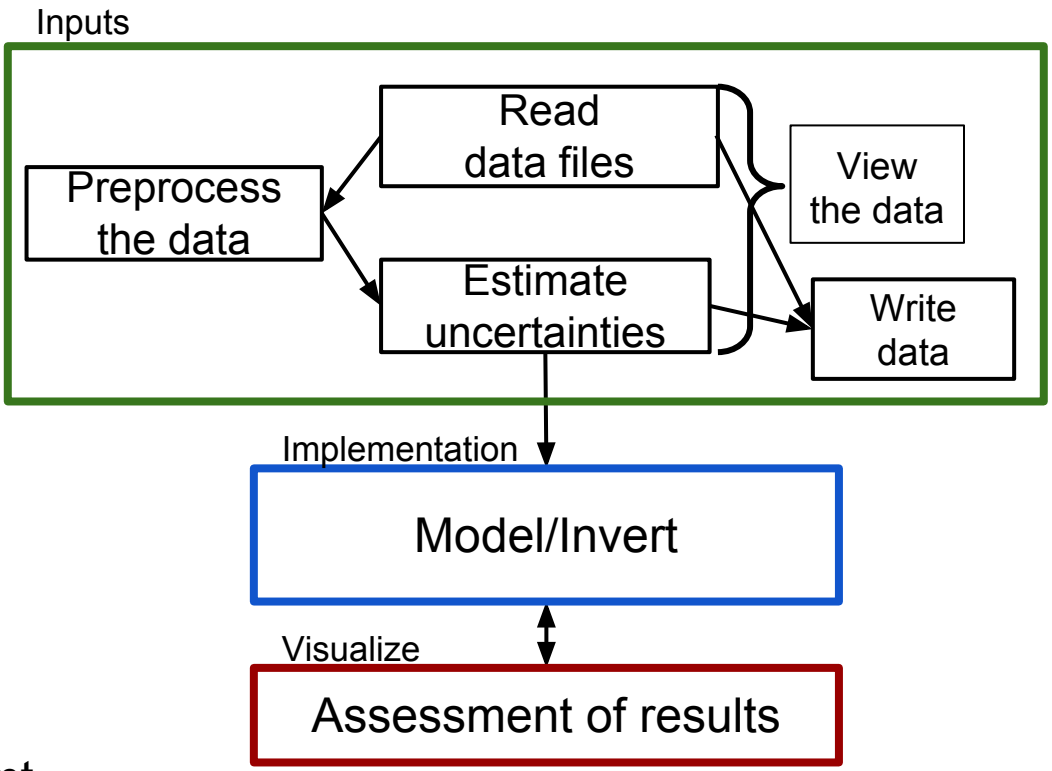


https://github.com/simpeg

simpegMT

Implements Magnetotellurics (MT) in the SimPEG framework

- IO data
- Forward model
  - Predict data
- Invert
  - Recover models
- Interpret
  - Visualize
  - Other data!!!



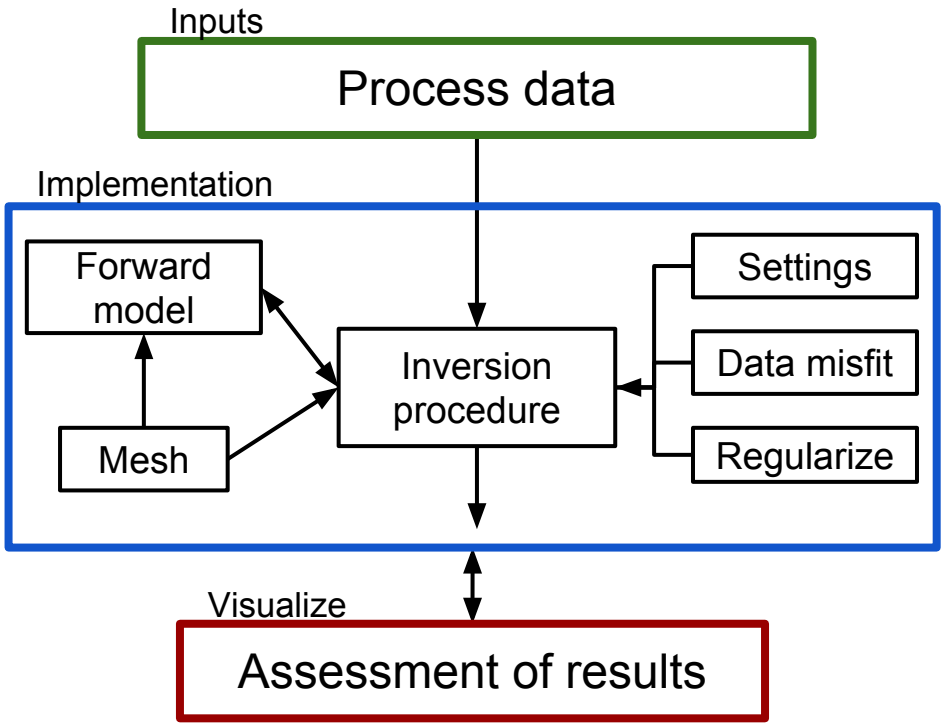
https://github.com/simpeg/simpegmt

Let’s look at some code

simpegMT

Implements Magnetotellurics (MT) in the SimPEG framework

- IO data
- Forward model
  - Predict data
- Invert
  - Recover models
- Interpret
  - Visualize
  - Integrate other data!!!



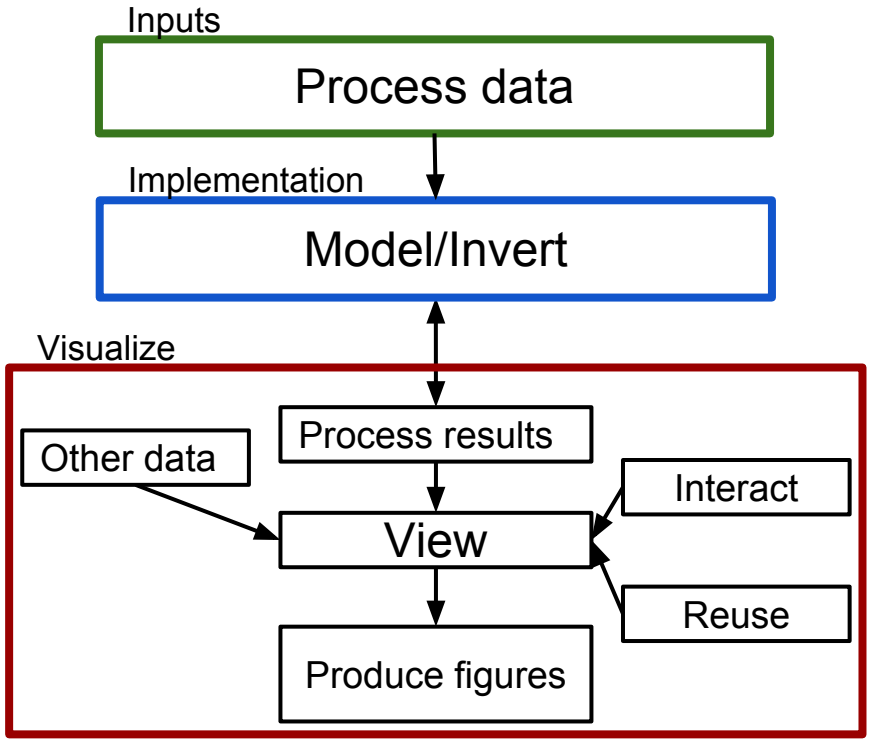
https://github.com/simpeg/simpegmt

Look at some more code

simpegMT

Implements Magnetotellurics (MT) in the SimPEG framework

- IO data
- Forward model
  - Predict data
- Invert
  - Recover models
- Interpret
  - Visualize
  - Integrate other data!!!



https://github.com/simpeg/simpegmt

Play around with some examples

The way forward...

- Continue implement simpegMT
  - Add features
  - Optimize
  - Clean and set benchmarks
- Integrate more with other packages
  - Move more to python
  - Visualization
  - Integration with other data

Explore leveraging further:

- vtk/paraview
- vispy
- h5py/pytables
- parallelism
- gis

Continue enjoying the great open-source community packages and support