



# Processing potential field data with Fatiando a Terra

Santiago Soler, Lu Li and Lindsey Heagy



## Process gravity data

```
import boule

# Compute gravity disturbance
normal_gravity = boule.WGS84.normal_gravity(
    data.latitude, data.height_m
)
gravity_disturbance = data.gravity_mgal - normal_gravity
```

```
import harmonica as hm

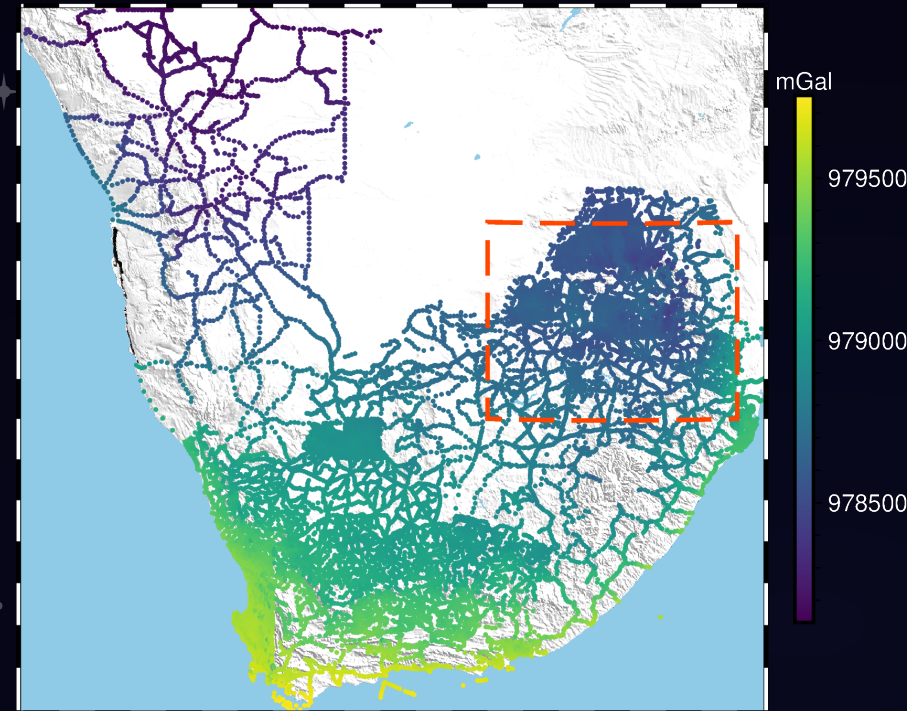
# Bouguer disturbance
topography_model = hm.prism_layer(
    coordinates=(topo.easting, topo.northing),
    surface=topo,
    reference=0,
    properties={"density": density},
)
topo_effect = topography_model.gravity(coordinates)
bouguer = gravity_disturbance - topo_effect
```

600k prisms  
4500 data points

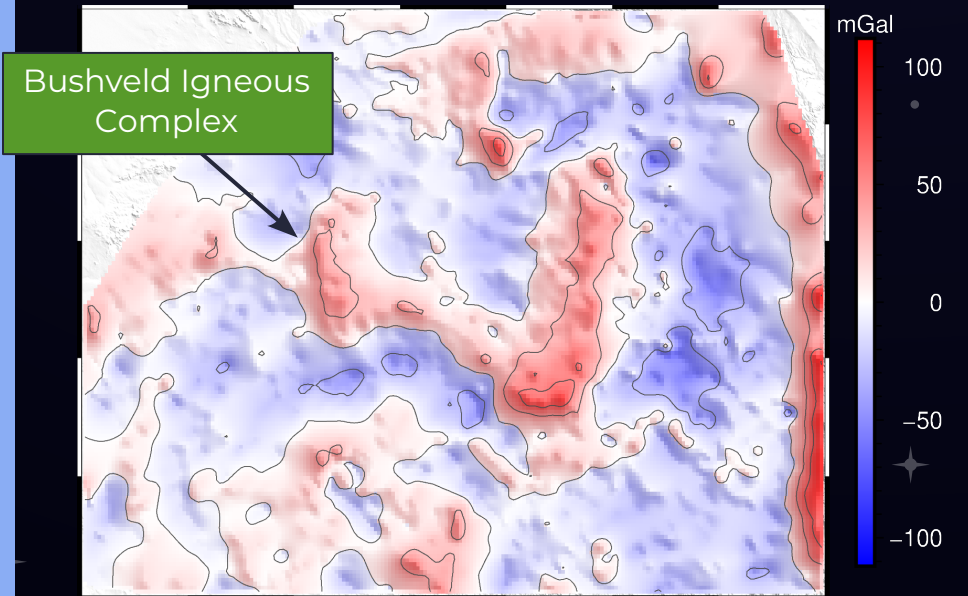
```
# Regional separation
deep_sources = hm.EquivalentSources(
    depth=500e3, damping=1000
)
deep_sources.fit(coordinates, bouguer)
regional_gravity = deep_sources.predict(coordinates)
residual_gravity = bouguer - regional_gravity
```

```
# Grid
eq_sources = hm.EquivalentSources(depth=5e3, damping=10)
eq_sources.fit(coordinates, residual_gravity)
grid = eq_sources.grid(grid_coors)
```

### Raw gravity data



### Gridded residual gravity



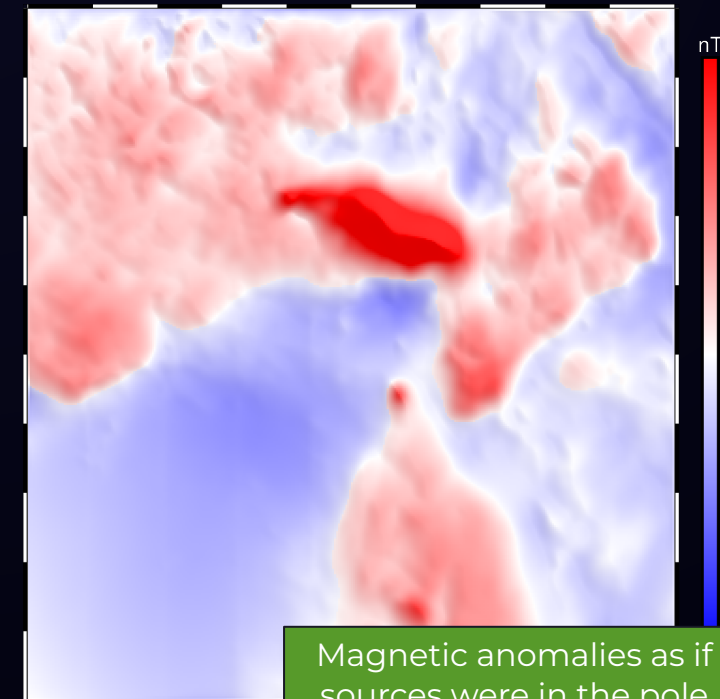
## Filter magnetic data

```
import harmonica as hm

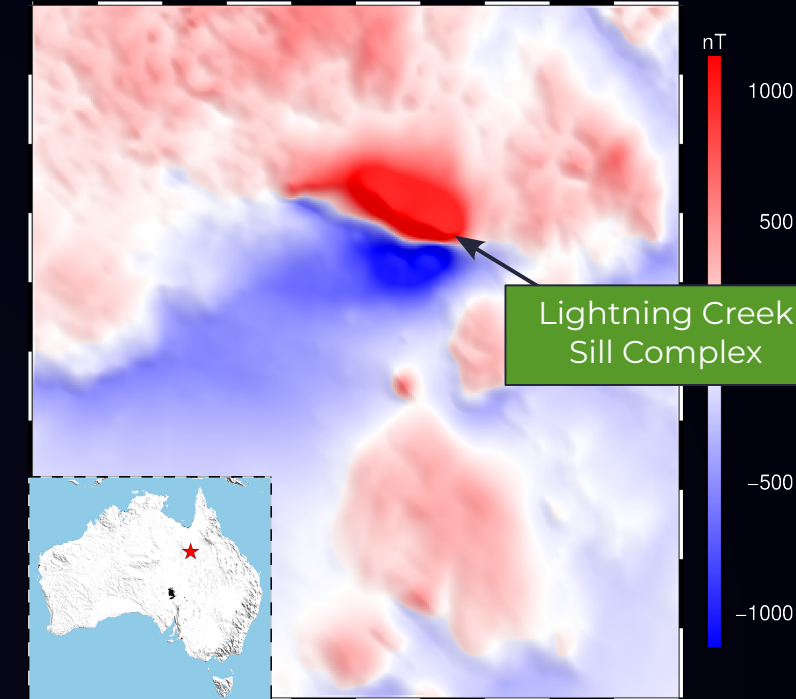
# Reduce to the pole
rtp_grid = hm.reduction_to_pole(
    magnetic_grid,
    inclination=-52.98,
    declination=6.51,
)

# Upward continuation
upward_cont = hm.upward_continuation(
    rtp_grid, height_displacement=200
)
```

### Reduced to the pole



### Gridded magnetic data



### Upward continued

