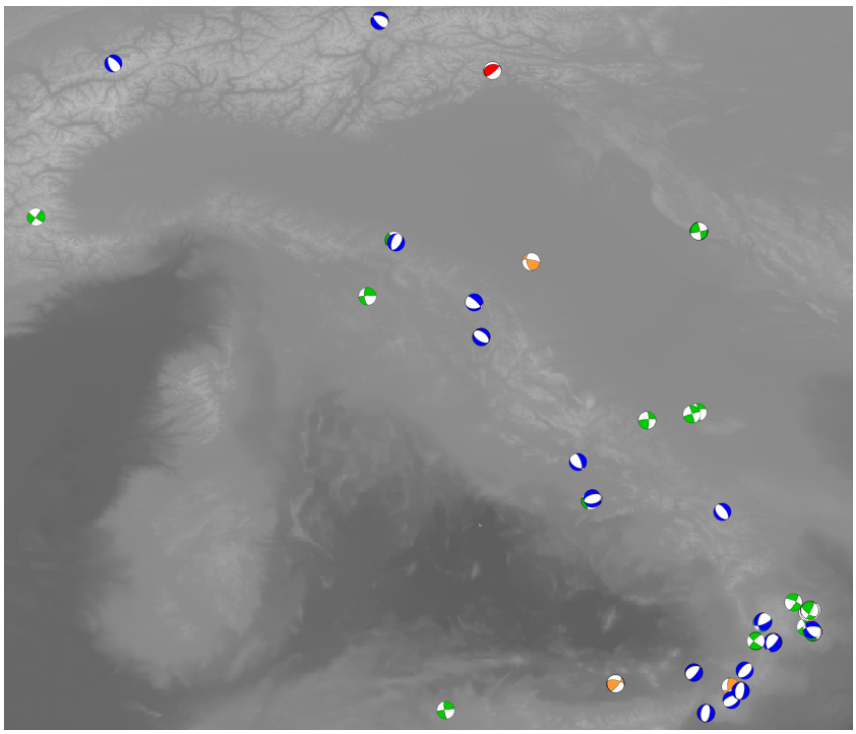
beach_ball - a Qgis plugin

beach_ball is a Qgis plugin written in Python for representing focal mechanisms in map as beachballs.



Hardware and software specification

beach_ball works on both Windows and Linux Operative Systems (tested on Windows 11 v24H2, EndeavourOS Mercury Neo, Ubuntu 24.10). It requires Obspy and some other Python modules that could not be included in the standalone version of Qgis. To avoid further installation to the users, we provided the missing packages inside the plugin (vendoring) under the *./beach_ball/libs/* path.

The process of creating SVG and connecting them to the features can be slow for large datasets, so be patient and wait for Qgis to show the finished green banner.

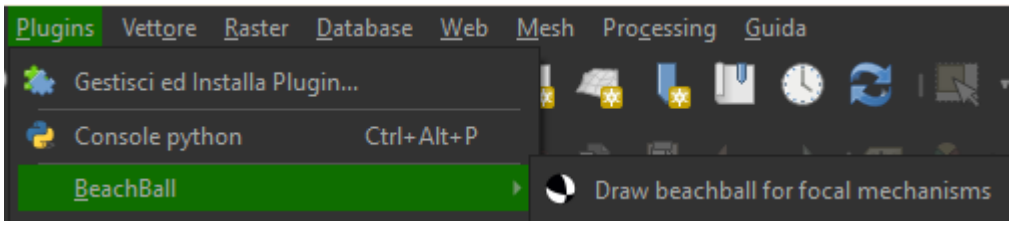
Structure and installation

The repository is composed of:

- a *README.md* file describing the plugin in a nutshell;
- the current *Documentation.md* describing in detail the goals, the structure, and the usage of the plugin;

- a *LICENCE* file describing the licence policy under beach_ball is released;
- the *beach_ball.zip* ready-to-install plugin;
- the not compressed *beach_ball* folder as created by the **Plugin Builder** Qgis plugin and further modified;
- an *example* folder contains a small dataset of focal mechanisms from the Time Domain Moment Tensor (Scognamiglio et al., 2006) for performing tests;
- a *screenshot* folder containing screenshots of the plugin during the usage.

For installing **beach_ball** in Qgis, the users can open the "manage and install plugins" windows, select the "install from zip" tab, and browse to the *beach_ball.zip* file. After selecting the "install" button, in some seconds the users should see a pop-up message informing about the correct installation of the plugin which should be visible in both the plugin menù and the main toolbar.

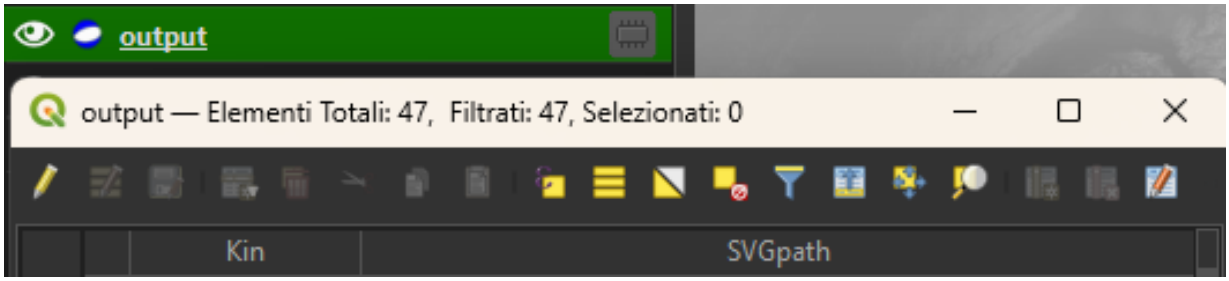


For customizing the plugin, browse to the *beach_ball* folder, apply changes and compress the folder to proceed as before with the installation. Inside *beach_ball* the *beach_ball.py* file is the main code, and *beach_ball_dialog_base.ui* is the graphical user interface built using QtCreator.

How it works

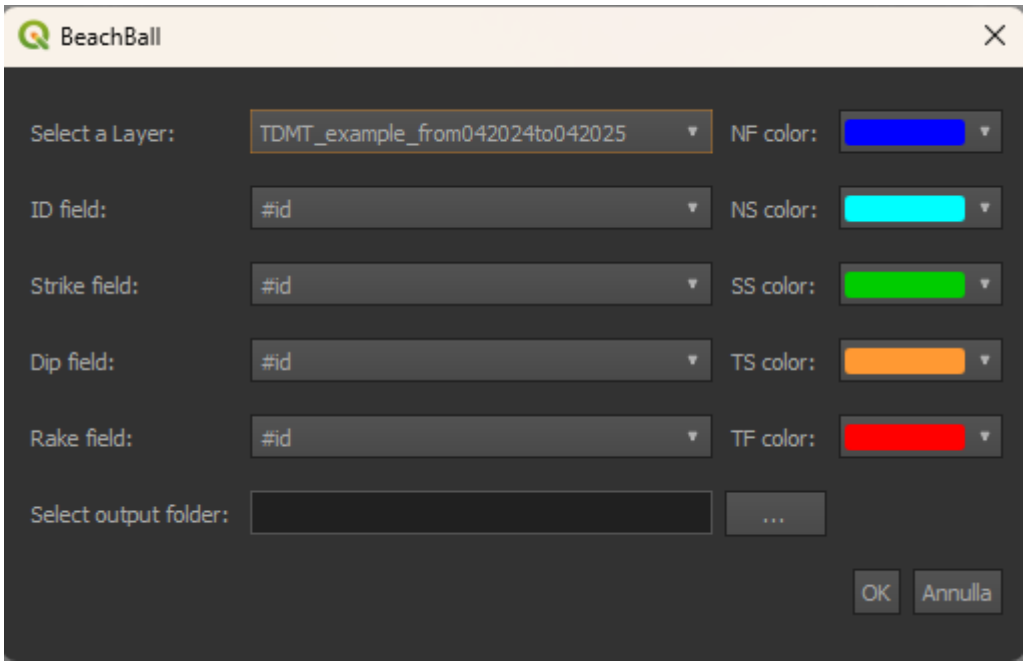
Starting from the strike, dip, and rake values, **beach_ball** produces an SVG icon using the Obspy module. The vector images are stored in a folder called *bb_svg* located in a custom path as selected by the users. The beachballs' names correspond to a unique ID associated with each feature as read in the specified field of the attribute table. The complete path to the corresponding image is added to each feature in the selected layer by creating a new field in the attribute table. The two nodal planes are used for estimating the P-T axes attitude. According to Zobac (1992), the plugin classifies the focal mechanisms of kinematics using the trend and plunge azimuth of two axes. The kinematic information is added as a new field in the attribute table using the following keys: **NF**=Normal Fault, **NS**=Normal Strike, **SS**=Strike Slip, **TS**=Thrust Strike, and **TF**=Thrust fault. Moreover, the beachballs are coloured according to these values using blue for NF, green for SS, and red for TS. The colours can be changed in the plugin GUI. To avoid the problem of number format, **beach_ball** tries to convert the ID, strike, dip, and rake values to float before performing the calculations. In case of the empty cell (NULL) or conversion error, it skips the features representing it as a white circle. This ensures the plugin is able to obtain a valid result and finish the process. The output result is a copy of the original layer (called "**output**") having two more fields in the attribute tables ("Kin" for the kinematics, and

"SVGpath" for the complete path to the SVG image). This is a temporary layer that can be saved as preferred by the users.

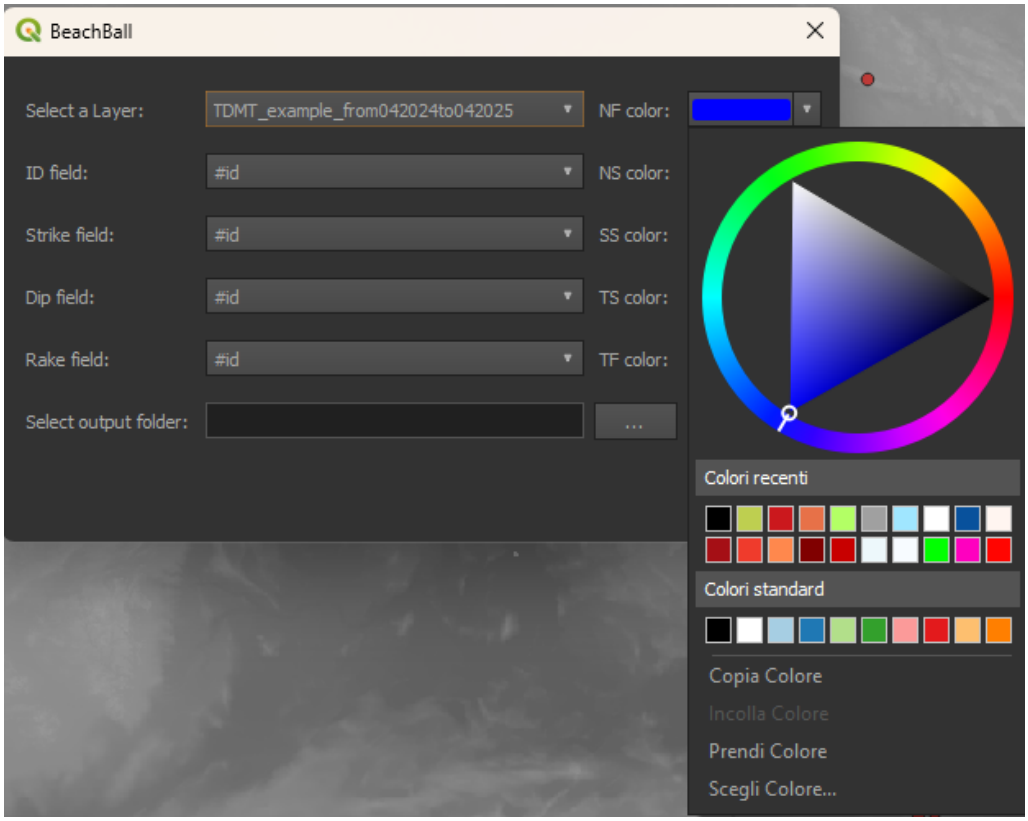


Usage

beach_ball is managed through a graphical user interface.



The users can choose between the layers already loaded in the project, and scroll the fields of the attribute table to indicate the ID, the strike, the dip and the rake field. After selecting the path for storing the SVGs, the users can customize the colour associated with each kinematics (only unknown beachballs are grey by default and cannot be changed).



The "Ok" button effectively executes the code. During the execution, Qgis appears frozen but after some seconds (depending on the dataset dimension) a green banner will appear to confirm the outputs are produced.

Licence

beach_ball is released under MIT licence.