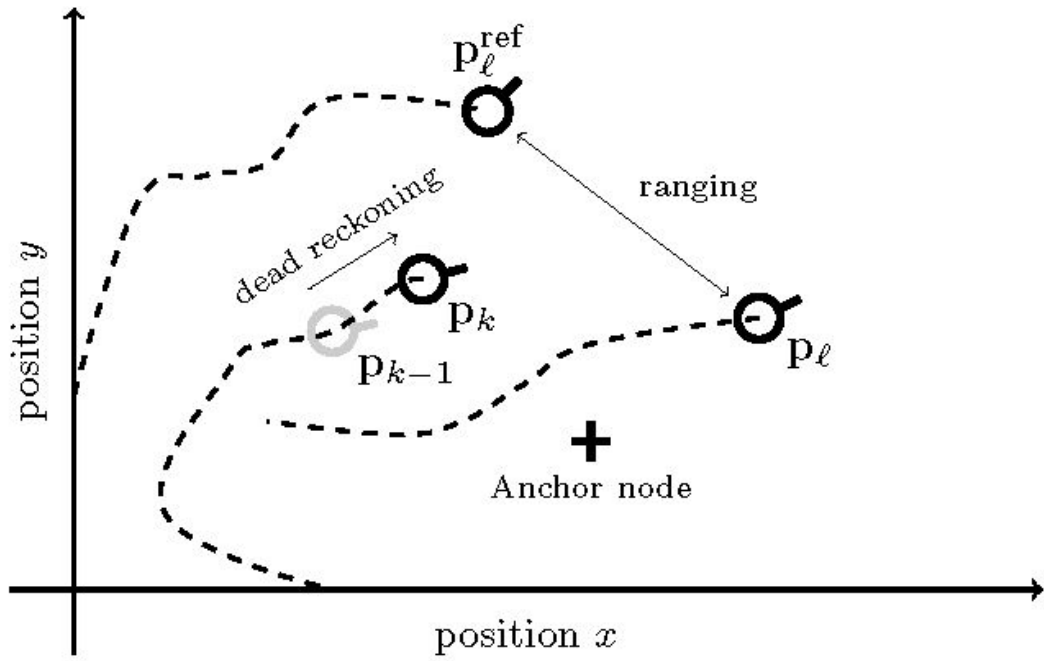


Understanding animal behaviour and location in the landscape using animal-borne technologies

User Guide and Installation

16.03.2016

Installation

1. Download and install Anaconda for python 3 (https://www.continuum.io/downloads)
2. Using Anaconda, install Python 3.4 (It ships with 3.5).
    a. Open a console/terminal window in the Anaconda installation directory. By default, this is located at "`C:\Anaconda3\`"
    b. Run "`conda create -n py34 python=3.4 anaconda`".
3. Download and install PyQt5 (https://www.riverbankcomputing.com/software/pyqt/download5), modifying the installation directory to match Anaconda's Python 3.4 installation (By default, this is located at "`C:\Anaconda3\envs\py34\`").
    a. NB: On a mac you are able to run the following command
        i. `conda install -c https://conda.anaconda.org/mmcauliffe pyqt5`
        ii. Once installed, you must append to your qt.conf file in the same directory as your python binary for your environment to contain a path to your QT5 plugins, this way it can find the cocoa library i.e:
            1. Plugins = /usr/local/opt/qt5/plugins
4. Download the repository's master branch from GitHub (https://github.com/davemccormick/pyAnimalTrack), and extract the files.

User Guide

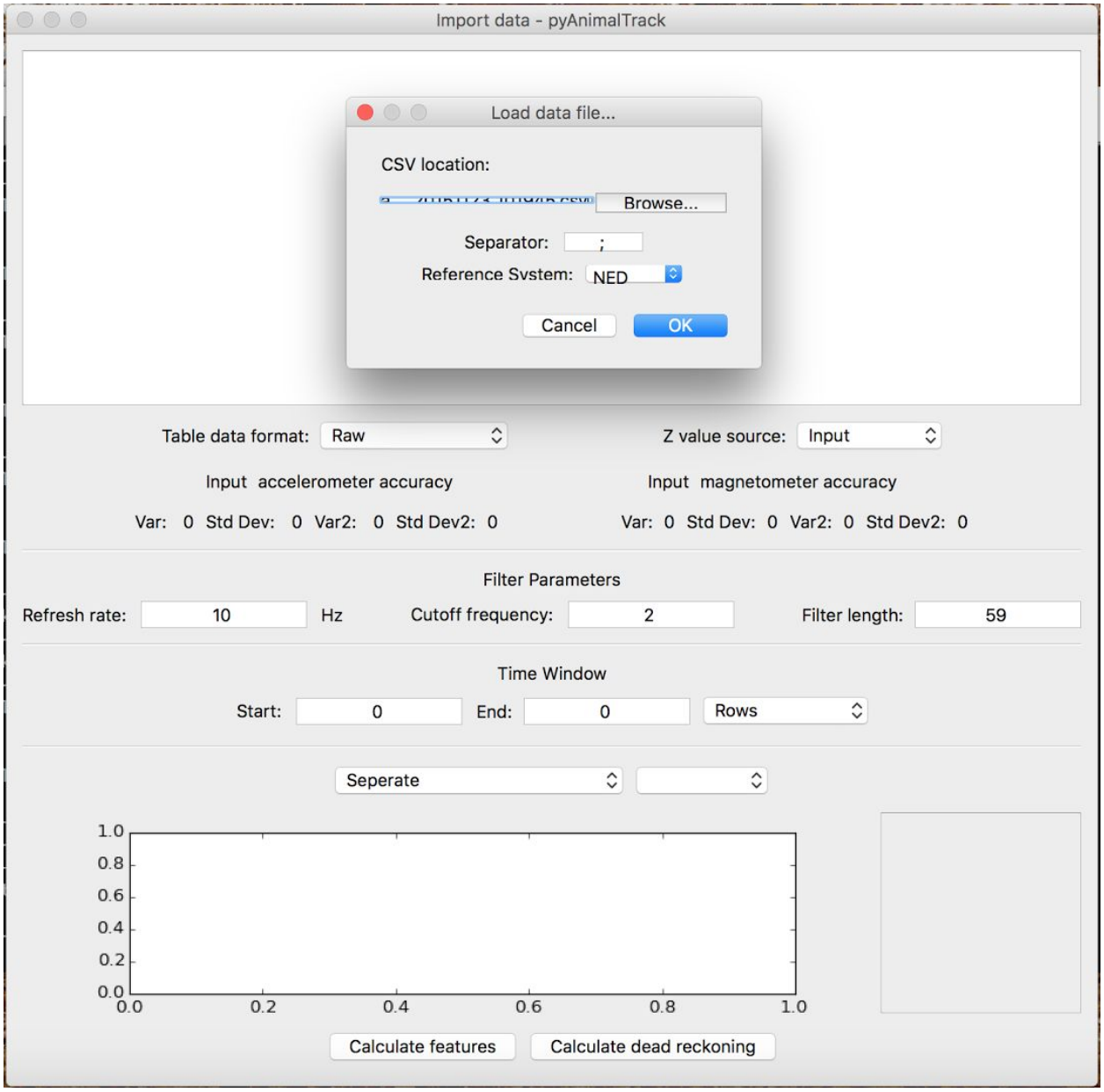
To begin, activate your virtual environment:

"source activate py34"

Once activate, navigate to the directory you extracted pyAnimalTrack to, within this directory move into pyAnimalTrack/src/

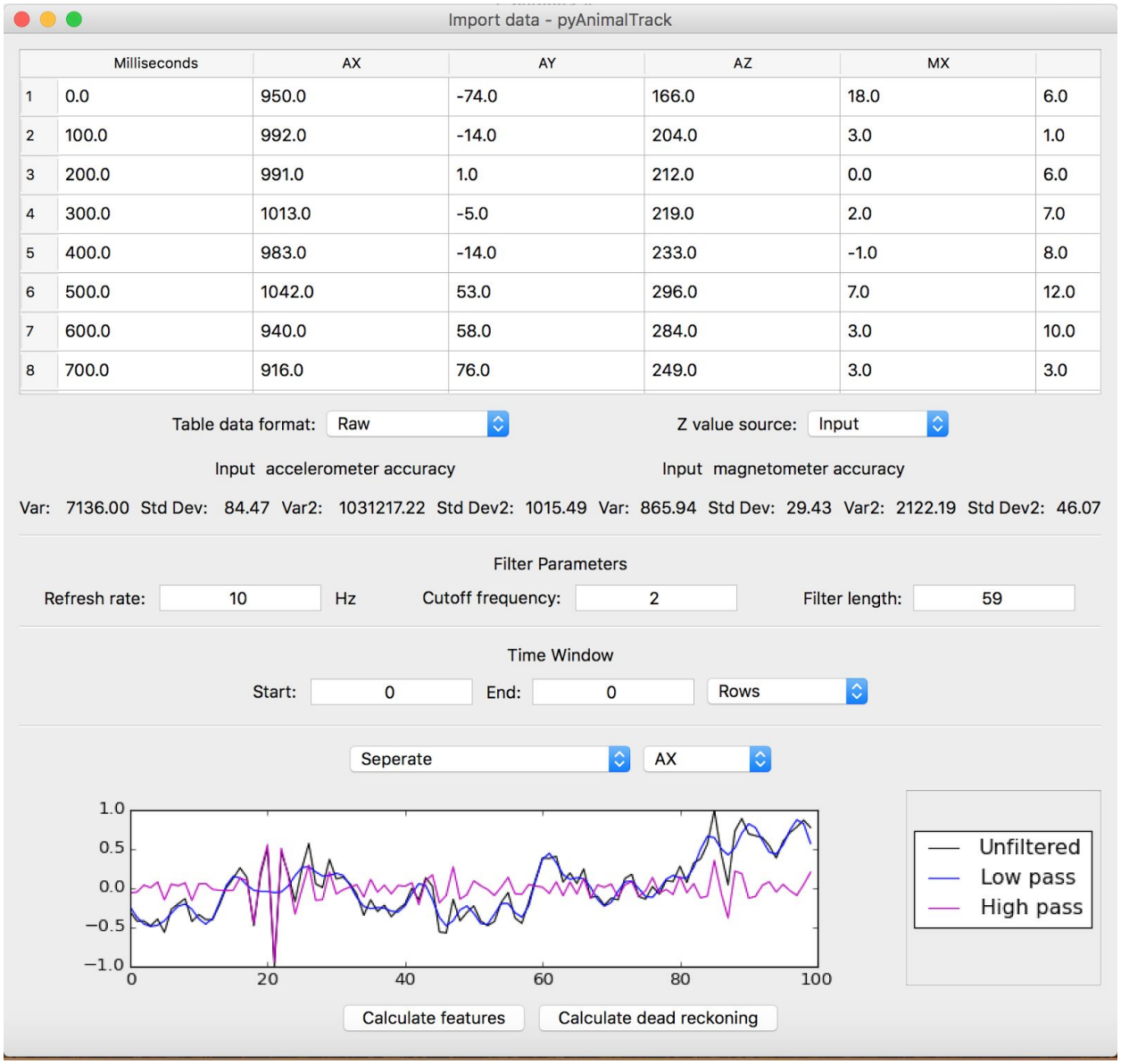
Once in the correct directory, run "python pyAnimalTrackRunner" and the UI should now be open.

File open dialog:



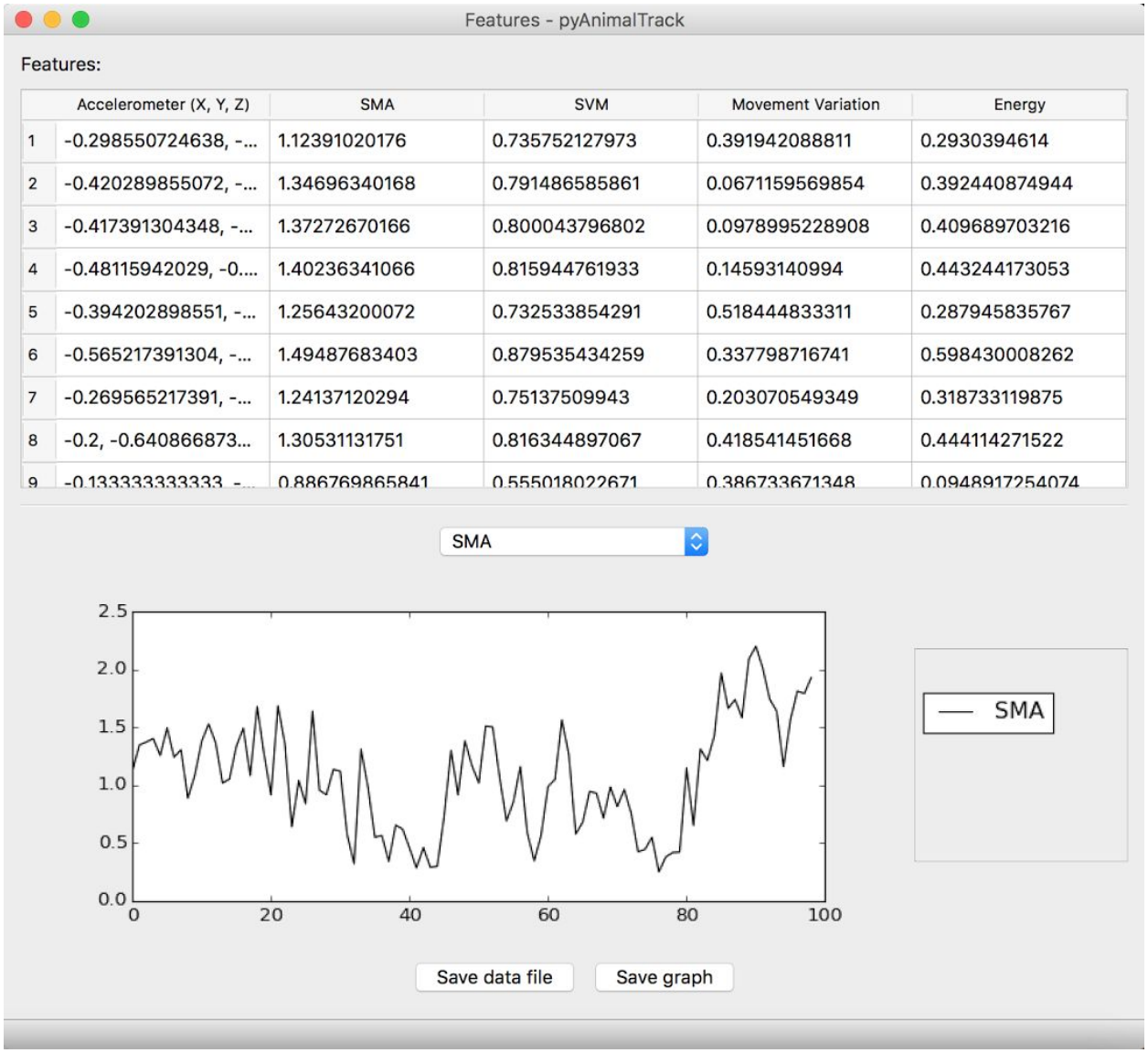
1. File location: The data file to parse.
2. Separator: Which character has been used to deliminate each cell value
3. ?: The orientation of the sensor that was used to record the data file

Main window:



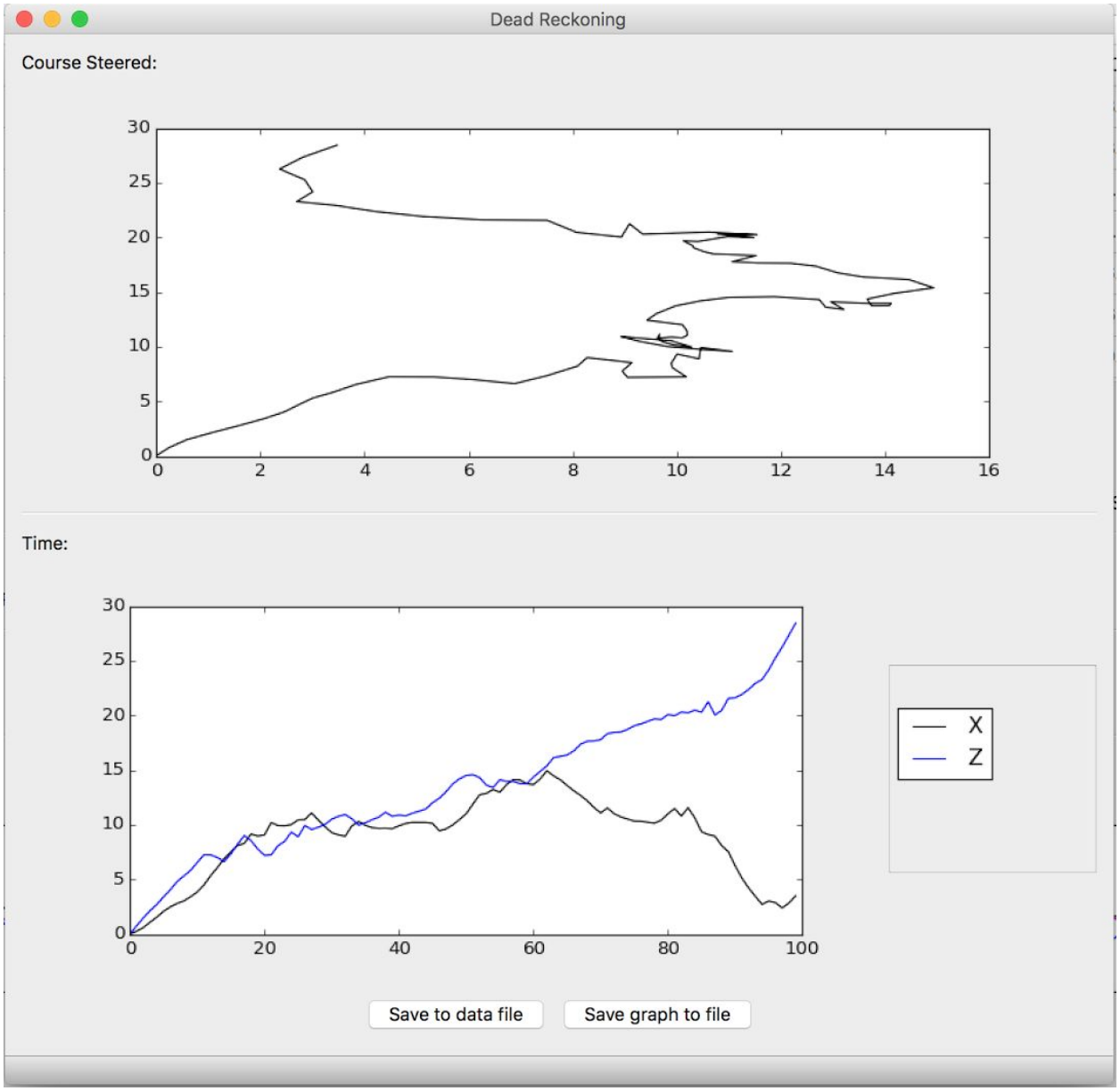
1. Table view: The dataset shown as a table of values. This can be changed to show either the raw imported data, the calibrated data, or the filtered data (Low and High pass filters).
2. Filter parameters: These can be adjusted on a per column basis, and control the operation of the Low and High pass filters.
3. Epoch selection: This can be used to choose a subset of the data for any operations to occur upon, by either a selection of rows, or an amount of time (in milliseconds)
4. Graph:

Features:



1. Table view: The complete feature based dataset is shown, including the accelerometer values, and all calculated features.
2. Feature graph

Dead Reckoning



1. Course Steered: This graph shows the path which the sensor has moved, plotting X and Y against each other.
2. Time: This graph shows how the X and Y coordinates have separately changed over time, independently of each other.
3. Saving: Use these buttons to export the calculated dead reckoning information, both as a CSV, and saving the graphs to file.

For a full walk through, please refer to the github page which links to a software demo of all features. The github page also links to the sphinx generated documentation.