PlotSeis Documentation

Release 0.1.0

Joshua Jones

Mar 15, 2020

Contents

CHAPTER 1

Installation

1. Ensure Python is installed and working on your machine.
2. [Install matplotlib for Python.](https://matplotlib.org/users/installing.html)
3. From the Julia prompt, press "]" to enter package mode, then type "add PlotSeis"
4. Exit package mode with "^C", then type "using PlotSeis"

1.1 Time-Series Plots

plotseis (*S::SeisData*[, *fmt="auto"*, *use_name=false*, *nxt=5)*])

Normalized trace plot of data in `S`. Time-series data use lines; irregularly- sampled data are plotted with circles.

uptimes (*S::SeisData*[, *summed=false*, *fmt="auto"*, *use_name=false*])

Plot uptimes of each channel in `S` using filled, colored bars.

If `summed==true`, plot uptimes for all channels in S that record timeseries data, scaled so that y=1 corresponds to 100% of channels active. Non-timeseries channels in S are not counted toward the cumulative total in a summed uptime plot.

1.1.1 Keywords

- `fmt=FMT` formats x-axis labels using C-language `strftime` format string `FMT`. If unspecified, the format is determined by when data in `S` start and end.
- `use_name=true` uses `S.name`, rather than `S.id`, for trace labels.
- `n=N` sets the number of X-axis ticks to N.

1.2 Time-Frequency Plots

`logspec`(*S::SeisData*, *k::Union{Int64*, *String}*[, *nx=1024*, *ov=0.5*, *fmin=0.5*fs/nx*, *fmax=0.5*fs*, *fmt="auto"*])

Spectrogram of trace number or channel ID `k` with logarithmic scaling of the y-axis (frequency).

1.2.1 Keywords

- **nx** window length
- **ov** overlap fraction between adjacent windows
- **fmin** lowest frequency to plot
- **fmax** highest frequency to plot

Index