PyEEG
A Python Module for EEG Feature Extraction

Forrest Sheng Bao[1,2] and Christina R. Zhang[3]

[1] Department of Computer Science, Texas Tech University, Lubbock, Texas
[2] Department of Electrical Engineering, Texas Tech University, Lubbock, Texas
[3] Department of Physiology, McGill University, Canada

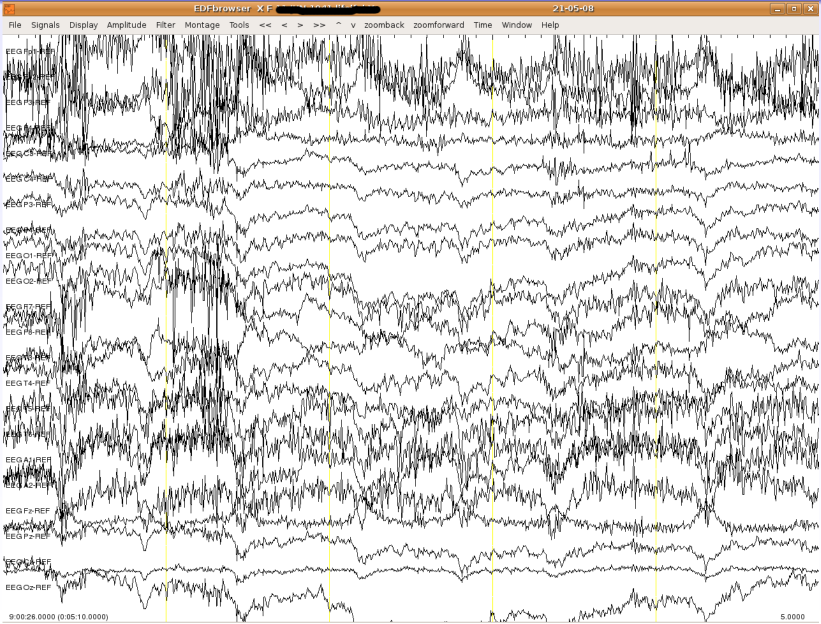
Jun. 30, 2010, Scipy 2010, UT, Austin, Texas

Outline

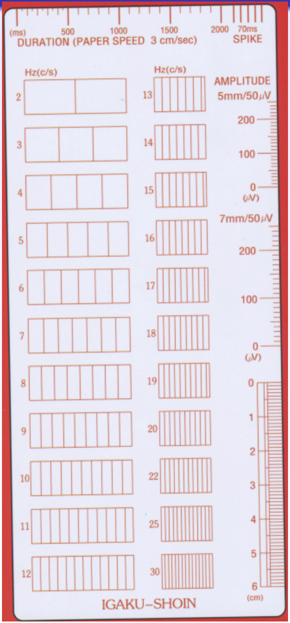
Feature Extraction in Neurological Signal Processing

- Over the past decade, computer-aided diagnosis (CAD) systems based on EEG have emerged in the early diagnosis of several neural diseases such as Alzheimer's disease [1] and epilepsy [2].
- A key component in most such CAD systems is to characterize EEG signals into certain features, a process known as *feature extraction*.
- EEG features can come from different fields that study time series: power spectrum density from classical signal processing, fractal dimensions from computational geometry, entropies from information theory, synchrony measures from nonlinear physics, etc.
- By extracting EEG features, we can use more powerful tools, such as machine learning, to analyze the signal in the next step.

EEG: The Name of the Game



EEG Reading Cards: The Limit of Our Eyes



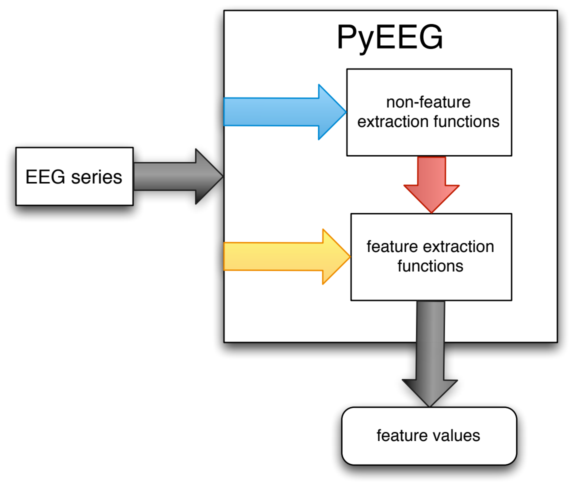
A Case Study on Epilepsy: Features may be better

- Over 50 million people worldwide suffer from epilepsy.
- Conventional epilepsy diagnosis may require long-term or repeated EEG recordings to capture seizures (ictal) or other epileptic activities. Physicians visually inspect lengthy EEG recordings - paging through 24-hr waveforms.
- To help those MDs, research on picking out segments that physicians may interest has been under going for quite a while.
- Recently, researchers have found it promising to use interictal (i.e., non-seizure) EEG records that do not contain particular activities. We can tell whether a EEG segment is epileptic [3] or where the seizure foci are [4], based on EEG features extracted.
- Futhermore, it is possible to predict the state of the brain by tracking the values of those features. This idea is use by seizure predition researchers.

Why PyEEG?

- The computational analysis to EEG signal as described above requires a toolbox to quantify EEG patterns. Turn “*The stock went **rocket high***” into “*NASDAQ increased **5%**.*”
- Such a toolbox can be very useful to computational neuroscience community.
- There is no highly active project as expected out there (e.g., pbrain, ptsa).
- We are not aware of an open-source “simple” (simplifying every aspect of using) Python solution.
- “Software is like sex; it’s better when it’s free.” (Free as in GPL.)

Main Framework



Main Framework (cond.)

- PyEEG consists of two sets of functions, *EEG pre-processing functions*, which do not return any feature values, and *feature extraction functions* that return feature values.
- Besides standard Python functions, PyEEG only uses functions provided by Numpy/SciPy.
- PyEEG does not define any new data structure, using standard Python and NumPy ones only.
- The inputs of all functions are time series in form of a list of floating-point numbers and a set of optional feature extraction parameters. Parameters have default values.
- The output of a feature extraction function is a floating-point number if the feature is a scalar or a list of floating-point numbers if it is a vector.

EEG Pre-processing

- Pre-processing: to build new time series from given time series for further computation.
- PyEEG currently provides two pre-processing functions.
- `embed_seq()`: to build embedding sequence (from given lag and embedding dimension)
- `first_order_diff()`: to compute first-order differential sequence. One can build differential sequences of higher orders by apply first-order differential computing repeatedly.

Features

feature	type
Relative Intensity Ratios (RIRs) [3]	a 1-D vector
Petrosian Fractal Dimension (PFD) [5]	a scalar
Higuchi Fractal Dimension (HFD) [6]	a scalar
Hjorth mobility & complexity [7]	a 1-by-2 vector
Spectral Entropy (entropy of RIRs)	a scalar
SVD Entropy [8]	a scalar
Fisher Information [9]	a scalar
Approximate Entropy (ApEn) [10]	a scalar
Sample Entropy (SampEn) [11]	a scalar
Detrended Fluctuation Analysis (DFA) [12]	a scalar

More features are coming in PyEEG.
No reinvention to the wheel (skewness, etc.).

Basic Usage

SciPy is required to run PyEEG. The latest PyEEG is released as a single Python script, which includes all functions. So users only need to download and place it under a directory that is in Python module search path, such as the working directory. Example:

```
>>>import pyeeg
>>> from numpy.random import randn
>>> for i in xrange(0,10):
...     pyeeg.dfa(randn(4096))
...
0.50473407278667271
0.53339499445571614
0.53034354430841246
0.50844373446375624
0.5162319368337136
0.46319279647779976
0.44515512343867669
0.44077407030262245
0.45894672465613884
0.49135727073171609
```

Future work

- More features.
- More comprehensive documentations.
- Unittest for all functions.
- Faster implementation.
- File I/O.
- (Probably) Integration with nipy/pbrain http://nipy.sourceforge.net/pbrain/.

Questions?

Licensed under GPL v3 at `http://code.google.com/p/pyeeg/`

Bibliography I

- J. Dauwels, F. Vialatte, and A. Cichocki, “A comparative study of synchrony measures for the early detection of Alzheimerâs disease based on EEG,” *Neural Information Processing*, pp. 112–125, 2008.
- F. S. Bao, D. Y.-C. Lie, and Y. Zhang, “A new approach to automated epileptic diagnosis using EEG and probabilistic neural network,” in *Proceedings of 20th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2008)*, vol. 2, 2008.
- F. S. Bao, J.-M. Gao, J. Hu, D. Y. C. Lie, Y. Zhang, and K. J. Oommen, “Automated epilepsy diagnosis using interictal scalp EEG,” in *Proceedings of 31st International Conference of IEEE Engineering in Medicine and Biology Society (EMBC 2009)*, 2009.

Bibliography II

- J. Dauwels, E. Eskandar, and S. Cash, “Localization of seizure onset area from intracranial non-seizure EEG by exploiting locally enhanced sychrony,” in *Proceedings of 31st International Conference of IEEE Engineering in Medicine and Biology Society (EMBC 2009)*, 2009.
- A. Petrosian, “Kolmogorov complexity of finite sequences and recognition of different preictal EEG patterns,” in *Proceedings of 8th IEEE Symposium on Computer-Based Medical Systems*, 1995.
- T. Higuchi, “Approach to an irregular time series on the basis of the fractal theory,” *Physica D*, vol. 31, no. 2, pp. 277–283, 1988.
- B. Hjorth, “EEG analysis based on time domain properties,” *Electroencephalography and Clinical Neurophysiology*, vol. 29, pp. 306–310, 1970.

Bibliography III

- S. Roberts, W. Penny, and I. Rezek, “Temporal and spatial complexity measures for electroencephalogram based brain-computer interfacing,” *Medical and Biological Engineering and Computing*, vol. 37, no. 1, pp. 93–98, 1999.
- C. J. James and D. Lowe, “Extracting multisource brain activity from a single electromagnetic channel,” *Artificial Intelligence in Medicine*, vol. 28, no. 1, pp. 89–104, 2003.
- S. Pincus, I. Gladstone, and R. Ehrenkranz, “A regularity statistic for medical data analysis,” *Journal of Clinical Monitoring and Computing*, vol. 7, no. 4, pp. 335–345, 1991.
- J. S. Richman and J. R. Moorman, “Physiological time-series analysis using approximate entropy and sample entropy,” *American Journal of Physiology - Heart and Circulatory Phsiology*, vol. 278, pp. H2039–H2049, 2000.

Bibliography IV

C.-K. Peng, S. Havlin, H. E. Stanley, and A. L. Goldberger, "Quantification of scaling exponents and crossover phenomena in nonstationary heartbeat time series," *Chaos*, vol. 5, no. 1, pp. 82–87, 1995.