Motivations

**Modularity**: setting up a correct methodological workflow requires fulfilling a complex pipeline of basic tasks

**Maintenance**: rapid prototyping of new algorithms allows keeping the library updated to state-of-the-art

**Reproducibility**: the experiments should be repeatable, so every single step should be exactly replicable

**Usability**: researchers should be able to build their own methodological pipeline

**Efficiency**: computing time and memory usage are relevant in most of ML tasks
Our Answer

Dynamic object-oriented programming language
- very clear, readable syntax
- portable
- stable and mature

Python module
- provides fast N-dimensional array manipulation
- basic linear algebra functions
- tools for integrating C/C++ code

Well established and popular programming language
- efficiency
- code portability
- code reusing
mlpy v2.2.1 - Overview

• Multiplatform and Open Source
  – Unix and GNU/Linux
  – Microsoft Windows
  – Mac OS X

• Requirements
  – Python >= 2.5
  – NumPy >= 1.1.0
  – GSL >= 1.8

• (Binary) Packages for
  – MS Windows (Xp, Vista)
  – Ubuntu Linux (8.04, 8.10, 9.04, >)
  – FreeBSD
  – ArchLinux

• Computationally efficient
  – intensive use of NumPy module
    (fast array manipulation and basic Linear Algebra)
  – internal ANSI C99 functions
  – use of GSL library

• Compact
  – Source code size: 610 KB
Provides high level procedures that support the design of rich *Data Analysis Protocols* (DAPs) for preprocessing, predictive classification and feature selection.

Elective application field: *bioinformatics* on high-throughput data.
Data Analysis Protocol

Features: Average position

Samples: Misclassifications

ORIGINAL DATASET

Performance metrics

List indicators
Data Analysis Protocol

Model choice

- SVM - IRELIEF - ERFE
- SRDA - SRDA - ERFE

Accuracy

(in)Stabiliti measure

Accuracy

0 1

(in)Stabiliti measure

0 1
Data Analysis Protocol

Model choice

- SVM - IRELIEF - ERFE
- SRDA - SRDA - ERFE

Accuracy

(in)Stability measure
Data Analysis Protocol

Model choice

- SVM - IRELIEF - ERFE
- SRDA - SRDA - ERFE

Accuracy

(in)Stability measure

Borda

Feats.
List 1 2 3 n

Biomarkers

hsa-mir-024-1-precNo1
hsa-mir-024-1-precNo2
hsa-mir-024-2-prec
hsa-mir-025-prec
hsa-mir-026a-precNo1
hsa-mir-026b-prec
hsa-mir-027a-prec
hsa-mir-027b-prec
hsa-mir-029a-2No1
hsa-mir-029a-2No2
Wavelet Transform

Orig. Signal

Discrete Transform
(DWT, UDWT)

Continuous Transform
(CWT, ICWT)

Unlike Fourier Transform X axis information is preserved. Allows the performing of Multiscale analysis.

\[ [W_\psi f](a, b) = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{\infty} \psi \left( \frac{x - b}{a} \right) f(x) dx \]

\( b = \text{Position} \)
\( a = \text{Dilatation} \)

Morlet -> Paul
D.o.G.
Wavelet Transform for Proteomics

Mass Spectrometry analysis: Peak detection problem
BUT Peak amplitude changes across the spectrum.

Original signal: Spectrum of the quantity of peptides present in a biological sample.

Denoised signal: after baseline-subtraction WT are applied to discover peaks.

Peak alignment: finding corresponding peaks between all the samples.
Feature extraction: assigning numerical values to quantities and dataset unification.
Machine Learning Analysis.
Wavelet Transform for Proteomics

Continuous WT for Proteomics

Multiscale Wavelet Properties
Wavelet Transform
for CNV analysis

Undecimated WT for CNV Segmentation

Problem
Finding boundaries of genomic regions with altered CN (segmentation)

State of the Art
• CNAT/CNAG (Affymetrix)
• GLAD R package
• SODEGIR [Bicciato et al., 2009]
  (computational time: ~2h for the region above)

Our Solution
• Based on [Ben-Yaacov and Edlar, 2008]
• Efficient C/Python implementation of Undecimated Wavelet Transform
• Computational time: ~1.5 sec for the region above
Future Work

- Porting for **Python 3**.
- Improvement of the **Documentation** and development of a **Tutorial**.
- Development of **state-of-the-art** algorithms for machine learning and **preprocessing**.
- **Reorganization** of the classes to make them **easier to use**.
- Revriting of the algorithms with **cuda/OpenCL** (Parallel computing – GPU).

- Calling tool for CNV analysis.
- Development of a GUI for classification and **Biomarker discovery**.
Main Developer:
Davide Albanese
(albanese@fbk.eu)

mlpy.fbk.eu

Also used in...

• Stefan Wuchty et. al. “Gene pathways and subnetworks distinguish between major glioma subtypes and elucidate potential underlying biology”, Journal of Biomedical Informatics 2010
• James Jolly “Efficiently extracting Relationships From Natural Language”, Association for the Advancement of Artificial Intelligence 2009

2nd for downloads and views >200 Projects
I think there are inexorable forces that will elevate open source software to widespread acceptance, and eventually dominance, in the scientific computing market. I have harbored a secret belief that this is true for some time now, but when customers and scientific computing professionals start saying the same thing, it’s time to pay careful attention. (Jan 2010)

Will Schroeder
President, CEO and co-founder of Kitware, Inc.
Are We Ready for the Upscaling to NGS Data? And our Software?