Java Essential Dynamics: JED

Protein Trajectory Analysis in Java

Charles C. David

University of North Carolina at Charlotte

Department of Bioinformatics and Computational Biology

Donald J. Jacobs

University of North Carolina at Charlotte

Department of Physics and Optical Science

Abstract (needs to be cut to 350 words…)

Background

Essential Dynamics (ED) is the application of principal component analysis (PCA) to a protein trajectory in order to extract the most biologically relevant or essential motions from the dataset. Investigators choose a covariance (Q) or correlation (R) based form of PCA for the chosen set of atoms that yields a set of eigenvalues and eigenvectors. When Cartesian coordinates are used in the analysis (cPCA), one generally also constructs PCA modes to assess the displacements of the residues in each eigenvector. To extract the essential motions of the protein, the displacement vectors (DV) calculated from the trajectory are projected onto the top set of eigenvectors to produce a set of principal components (PC). These may then be investigated to determine the details of the atomic motions and determine clusters of similar displacements in the trajectory. The essential subspace (SS) defined by the top set of eigenvectors defines the essential motions of the trajectory, and often one wishes to compare these SS for similarity assessment. Unfortunately, no ED package exists that had all the above features and could handle very large data files.

Results

We developed a Java package, Java Essential Dynamics (JED), that is capable of performing complete ED for a protein trajectory. This implementation can read the trajectory data from sets of PDB files or from a matrix of atomic coordinates. The PDB files may be single or multi chained. A set of residues is chosen that may be non-contiguous. The analysis proceeds using the alpha carbon atoms from each of the selected residues. Cartesian based analysis is done using both Q and R based PCA. There is an option to specify a second subset of residues for PCA based on internal distance coordinates (dPCA). The program outputs all results as text files that can be reviewed using common tools such as Excel, Gnu Plot, etc. Key results are the transformed coordinates, the conformation and residue RMSDs, the DVs; plus the eigenvalues, eigenvectors, pca-modes, and projections of the DVs onto the top eigenvectors for both the Q and R based PCA for both the cPCA and dPCA. Additionally, a set of PDB files is generated along with a Pymol™ script to visualize each of the top modes derived from both Q and R cPCA. Finally, JED has classes for comparing the subspaces defined by the top eigenvectors using a variety of metrics. This allows one to compare trajectories directly to assess similarity within the essential subspaces defined by multiple analyses.

Conclusions

We present a Java package that performs “best-practices” ED. JED allows one to compare the results derived from Q and R based PCA and allows one to readily see the abstract motions that are specified by the eigenvectors. There is also the option for comparing the essential subspaces of multiple trajectories to each other. JED offers a complete ED analysis package that is unique, functional, and unbiased.

**Keywords**

Essential Dynamics, PCA, distance PCA, vector space, subspace, RMSIP, principal angle

**Background**

The Background section should be written in a way that is accessible to researchers without specialist knowledge in that area and must clearly state - and, if helpful, illustrate - the background to the research and its aims. It should clearly described the relevant context and the specific issue which the software described is intended to address.

**Implementation**

This should include a description of the overall architecture of the software implementation, along with details of any critical issues and how they were addressed.

**Results and Discussion**

The Results and Discussion may be combined into a single section or presented separately. They may also be broken into subsections with short, informative headings. In any case what should be described is the functionality of the software together with data on how its performance and functionality compare with and improve on functionally similar existing software. There should then be a discussion of the intended use of the software, and the benefits that are envisioned together, if possible, with an outline for the planned future development of new features.

**Conclusions**

This should state clearly the main conclusions of the article and give a clear explanation of the importance and relevance of the software.

**Availability and requirements**

* **Project name:** Java Essential Dynamics: JED
* **Project home page:** http://sourceforge.net/projects/JED
* **Operating system(s):** Platform independent
* **Programming language:** Java
* **Other requirements:** Java JDK 1.5 or higher, an amount RAM appropriate to the size of Q or R: JED performs a full eigenvalue decomposition.
* **License:** GNU GPL.
* **No restrictions to use:** For reproduction and development, cite thelicense