V1mb121514

**Direct Energy - Analytical Software Selection – Comparison of Major Applications**

| **Functionality/Application** | **MatLab (ML)** | **R (SPLus)** | **Python (PY)** | **Scilab** | **Freemat** | **Octave** | **@Risk** | **JAVA/C#** | **C++** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |
| * Mathematics functionality
 |  |  | Add packages/ML (NumPy, SciPy, Matplotlib) |  |  |  |  |  |  |
| * Numerical Capabilities
 |  |  |  |  |  |  |  |  |  |
| * + Simulation
 | Some unique to MatLab: Simulink (scheduling algorithm) | Struggles for complicated MCMC / ML |  |  |  |  |  |  |  |
| * + Optimization
 | Some unique to MatLab: CVX, YalMip | R can do almost all/ML<http://127.0.0.1:19321/library/stats/html/optim.html> |  |  |  |  |  |  |  |
| * + Numerical integration
 |  |  |  |  |  |  |  |  |  |
| * + Interpolation/extrapolation
 |  |  |  |  |  |  |  |  |  |
| * + Other numerical procedures related to statistics
 |  |  |  |  |  |  |  |  |  |
| * Statistics
 |  | Superior: ARIMA modelsfractional ARIMA modelsGARCH models WaveletsGLM modellingRegressionsMCMC algorithmsNumerical integration |  |  |  |  |  |  |  |
| * + Ordinary Least Squares (OLS)/Nonlinear Schroedinger (NLS) estimation techniques
 |  |  |  |  |  |  |  |  |  |
| * + Time Series techniques
 |  | numerous |  |  |  |  |  |  |  |
| * + Forecasting
 |  |  |  |  |  |  |  |  |  |
| * + Hypothesis testing
 |  |  |  |  |  |  |  |  |  |
| * Data structures
 |  |  | Superior (support for dictionaries i.e. hashes) /ML. |  |  |  |  |  |  |
| * Calendar functionality
 |  |  |  |  |  |  |  |  |  |
| * Plotting
 |  |  |  |  |  |  |  |  |  |
| * Geospatial Functionality
 |  |  |  |  |  |  |  |  |  |
| * Financial and Economics packages
 |  | Numerous <http://en.wikipedia.org/wiki/Rmetrics> |  |  |  |  |  |  |  |
| * Reporting
 |  |  |  |  |  |  |  |  |  |
| * Graphical User Interface (including third party GUI)
 | Superior |  | It is possible to create applications using any of the mayor GUI libraries (TK, WX, GTK, QT, ...), |  |  |  |  |  |  |
| * Speed
 | MATLAB is about 10 times slower than “good” C++ and Fortran programs. However, with vectorization MATLAB can get faster and “comparable” to C++/Fortran. | R is slowest/C++, even if it is compiled. Superior efficiency of some libraries | Python is about 44 to 270 times slower than “good” C++ and Fortran code. Can enhance the speed of algorithms by converting Python to C code (pyrex/cython ) |  |  |  |  |  | C++ and Fortran are the fastest in execution. C++ can edge Fortran, but in High Performance Computing (i.e. supercomputer racing) the language primarily used is Fortran. C++ is more demanding language than Fortran, and it requires longer experience. Note Julia is only 2-3 times slower than C++ and Fortran. Sometimes new compiler technology such just-in-time compilers for interpreted languages like Java, ML, quite often better optimize performance than the C/C++. |
| * Integration and cross platform sharing/ ability for one application to work with another
 | Can be called from Python (Matlab-Wrap, -Bridge, -COM). Harder in some cases |  | Can be called from most other. it is very easy for other parties to design packages or other software tools that extend Python. Use OpenGL, drive your USB port |  |  |  |  |  |  |
| * Create stand-alone application
 | The portability solution (ML Component Runtime; MCR) works fine, but the application must be exactly the same version as the installed MCR (done to prevent generic Matlabing with it) can be a nuisance considering that a new version is released every 6 months | can also be invoked from within Excel (RExcel) | Can create a standalone application from your source (py2exe) |  |  |  |  |  |  |
| * OS (windows 8) and automatic scheduling
 |  |  |  |  |  |  |  |  |  |
| * Proprietary/Opensource
 | Math toolbox main engine is Opensource. Other mostly proprietary including algorithms equations (you have to trust they are correct). Hard for 3rd parties to extend or create tools for Matlab |  | OpenSource. |  |  |  |  |  |  |
| * User group driven/Internally driven
 | Sometimes unreceptive to outside suggestions |  |  |  |  |  |  |  |  |
| * Cost
 | Expensive in particular if many toolboxes. A new version is released every month with “new” but “delayed” features?. |  | Free |  |  |  |  |  |  |
| * Usability
 | Easier for beginner. The concept of Matlab refers to the whole package, including the IDE |  | Need to install separateIDE |  |  |  |  |  |  |
| * Scalability
 | None |  | Easier |  |  |  |  |  |  |
| * Resource availability in employment market
 |  |  |  |  |  |  |  |  |  |
| * Information sources
 | [wiki](http://en.wikipedia.org/wiki/MATLAB) | <http://cran.ma.imperial.ac.uk/> | <https://sites.google.com/site/pythonforscientists/python-vs-matlab> |  |  |  |  |  |  |
| **SUMMARY** |  **A popular numerical computing environment and programming language. The standard library does not contain as much generic programming functionality/PY , but does include matrix algebra and an extensive library for data processing and plotting.** |  |  |  |  |  |  |  |  |

**APPENDIX**

==========================================================

Compare Python and MatLab:


**==========================================================**

