Elements of Scientific Computing with Julia

March 5, 2015

Traditional Scientific Computing

We've learned a lot this quarter! From traditional scientific computing:

Traditional Scientific Computing

We've learned a lot this quarter! From traditional scientific computing:

- The nature and importance of errors in scientific computing;

Traditional Scientific Computing

We've learned a lot this quarter! From traditional scientific computing:

- The nature and importance of errors in scientific computing;
- Quadrature methods for numerical integration;

Traditional Scientific Computing

We've learned a lot this quarter! From traditional scientific computing:

- The nature and importance of errors in scientific computing;
- Quadrature methods for numerical integration;
- Numerical differentiation with the Finite Difference Method;

Traditional Scientific Computing

We've learned a lot this quarter! From traditional scientific computing:

- The nature and importance of errors in scientific computing;
- Quadrature methods for numerical integration;
- Numerical differentiation with the Finite Difference Method;
- Linear algebra concepts and performing matrix computations;

Statistical Learning

From machine learning and statistics:

Statistical Learning

From machine learning and statistics:

- How to perform quantitative regression analysis;

Statistical Learning

From machine learning and statistics:

- How to perform quantitative regression analysis;
- How to perform qualitative regression analysis (or classification);

Statistical Learning

From machine learning and statistics:

- How to perform quantitative regression analysis;
- How to perform qualitative regression analysis (or classification);
- Application of the gradient descent algorithm to find the least-squares solution to problems;

Statistical Learning

From machine learning and statistics:

- How to perform quantitative regression analysis;
- How to perform qualitative regression analysis (or classification);
- Application of the gradient descent algorithm to find the least-squares solution to problems;
- How recommender systems work;

Important Tools

We've also added some important tools in our computer scientist tool-box:

Important Tools

We've also added some important tools in our computer scientist tool-box:

- Julia - a young and promising scientific computing language;

Important Tools

We've also added some important tools in our computer scientist tool-box:

- Julia - a young and promising scientific computing language;
- git - a widely used (in both industry and academia) versioning system;

Important Tools

We've also added some important tools in our computer scientist tool-box:

- Julia - a young and promising scientific computing language;
- git - a widely used (in both industry and academia) versioning system;
- LaTeX- a typesetting language to make elegant documents and presentation (most relevant for research and publishing);

Important Tools

We've also added some important tools in our computer scientist tool-box:

- Julia - a young and promising scientific computing language;
- git - a widely used (in both industry and academia) versioning system;
- LaTeX- a typesetting language to make elegant documents and presentation (most relevant for research and publishing);
- Packages for plotting and making programming visual and easy, so as a scientist you can focus on the problem at hand;

Traditional Scientific Computing

We've learned a lot this quarter! But there is lots more to learn:

Traditional Scientific Computing

We've learned a lot this quarter! But there is lots more to learn:

- The details of computer arithmetic;

Traditional Scientific Computing

We've learned a lot this quarter! But there is lots more to learn:

- The details of computer arithmetic;
- Taylor series function approximation;

Traditional Scientific Computing

We've learned a lot this quarter! But there is lots more to learn:

- The details of computer arithmetic;
- Taylor series function approximation;
- ODE's and PDE's with the Finite Element Method;

Traditional Scientific Computing

We've learned a lot this quarter! But there is lots more to learn:

- The details of computer arithmetic;
- Taylor series function approximation;
- ODE's and PDE's with the Finite Element Method;
- Fast Fourier Transforms;

Traditional Scientific Computing

We've learned a lot this quarter! But there is lots more to learn:

- The details of computer arithmetic;
- Taylor series function approximation;
- ODE's and PDE's with the Finite Element Method;
- Fast Fourier Transforms;
- Different matrix factorization methods such as QR, Cholesky and SVD;

Traditional Scientific Computing

We've learned a lot this quarter! But there is lots more to learn:

- The details of computer arithmetic;
- Taylor series function approximation;
- ODE's and PDE's with the Finite Element Method;
- Fast Fourier Transforms;
- Different matrix factorization methods such as QR, Cholesky and SVD;

Recommendation: Take a 5 credit scientific computing course (Fall 2014 at UWB).

Traditional Scientific Computing

We've learned a lot this quarter! But there is lots more to learn:

- The details of computer arithmetic;
- Taylor series function approximation;
- ODE's and PDE's with the Finite Element Method;
- Fast Fourier Transforms;
- Different matrix factorization methods such as QR, Cholesky and SVD;

Recommendation: Take a 5 credit scientific computing course (Fall 2014 at UWB). Or start by reading Heath (our course text) and wikipedia to see where that takes you. For those interested in linear algebra, I also recommend: Numerical Linear Algebra by Trefethen and Bau III.

Statistical Learning

From machine learning and statistics:

Statistical Learning

From machine learning and statistics:

- Support Vector Machine;

Statistical Learning

From machine learning and statistics:

- Support Vector Machine;
- Neural Networks (see course schedule at UWB);

Statistical Learning

From machine learning and statistics:

- Support Vector Machine;
- Neural Networks (see course schedule at UWB);
- Unsupervised learning techniques;

Statistical Learning

From machine learning and statistics:

- Support Vector Machine;
- Neural Networks (see course schedule at UWB);
- Unsupervised learning techniques;
- Advanced optimization algorithms;

Statistical Learning

From machine learning and statistics:

- Support Vector Machine;
- Neural Networks (see course schedule at UWB);
- Unsupervised learning techniques;
- Advanced optimization algorithms;
- Sentiment Analysis;

Statistical Learning

From machine learning and statistics:

- Support Vector Machine;
- Neural Networks (see course schedule at UWB);
- Unsupervised learning techniques;
- Advanced optimization algorithms;
- Sentiment Analysis;
- Practical ML techniques: test and cross validation sets, model selection, how to focus your energy when solving a problem;

Recommendation: Take a 5 credit machine learning course (Check Graduate Courses Schedule at UWB).

Statistical Learning

From machine learning and statistics:

- Support Vector Machine;
- Neural Networks (see course schedule at UWB);
- Unsupervised learning techniques;
- Advanced optimization algorithms;
- Sentiment Analysis;
- Practical ML techniques: test and cross validation sets, model selection, how to focus your energy when solving a problem;

Recommendation: Take a 5 credit machine learning course (Check Graduate Courses Schedule at UWB).

For those interested in studying more optimization, I recommend starting with Convex Optimization by Boyd and Vandenberghe. Also, Numerical Optimization by Nocedal and Wright is quite good.

Important Tools

Other tools that might be worth considering:

Important Tools

Other tools that might be worth considering:

- MATLAB/Octave - the standard language used by the scientific computing community (i.e. if you want do something in MATLAB, chances are that there's a package for that);

Important Tools

Other tools that might be worth considering:

- MATLAB/Octave - the standard language used by the scientific computing community (i.e. if you want do something in MATLAB, chances are that there's a package for that);
- R - the standard language used by the statistics community;

Important Tools

Other tools that might be worth considering:

- MATLAB/Octave - the standard language used by the scientific computing community (i.e. if you want do something in MATLAB, chances are that there's a package for that);
- R - the standard language used by the statistics community;
- mercurial - an easier to use versioning system (not as widely used as git however);

Important Tools

Other tools that might be worth considering:

- MATLAB/Octave - the standard language used by the scientific computing community (i.e. if you want do something in MATLAB, chances are that there's a package for that);
- R - the standard language used by the statistics community;
- mercurial - an easier to use versioning system (not as widely used as git however);
- Bitbucket - a web-based hosting service that supports mercurial;

Important Tools

Other tools that might be worth considering:

- MATLAB/Octave - the standard language used by the scientific computing community (i.e. if you want do something in MATLAB, chances are that there's a package for that);
- R - the standard language used by the statistics community;
- mercurial - an easier to use versioning system (not as widely used as git however);
- Bitbucket - a web-based hosting service that supports mercurial;
- HTML5 - for “hipper” looking presentations;

Important Tools

Other tools that might be worth considering:

- MATLAB/Octave - the standard language used by the scientific computing community (i.e. if you want do something in MATLAB, chances are that there's a package for that);
- R - the standard language used by the statistics community;
- mercurial - an easier to use versioning system (not as widely used as git however);
- Bitbucket - a web-based hosting service that supports mercurial;
- HTML5 - for "hipper" looking presentations;
- Different programming paradigms - more perspective, better programming;

Recommendation: Play around with this stuff!

Many Thanks!

Finally, I just want to say thank you for participating in my course this quarter. I truly enjoyed teaching it and making all the materials, as well as interacting with all of you!

Many Thanks!

Finally, I just want to say thank you for participating in my course this quarter. I truly enjoyed teaching it and making all the materials, as well as interacting with all of you!

We covered a lot of cool and challenging stuff, and I want to thank you for your attention and effort in this course. I hope you will continue studies in one of the many subtopics of scientific computing. Feel free to contact me in the future to exchange ideas!