Possible DRP Projects

The traditional style for a DRP project is to pick a book not covered in a traditional course, read a few chapters, and do some of the exercises. We can certainly do the same for any applied/computational math topic, substituting some pen/paper exercises for coding ones. We could also take a more problem driven approach. Given a problem you are interested in solving (e.g. denoising an image, modeling forest fires, etc.), we could explore different mathematical approaches to solve it. Another approach is to pick a topic, or even a single concept, term, algorithm, etc., learn as much as possible about it, and write a survey article or Wikipedia style article about it.

We can work on something close to my own research interests (numerical methods for partial differential equations), any other numerical analysis topic, or really anything considered applied and computational math. Obviously, I can advise more closely and provide a concrete plan of study for topics I am more familiar with. But I'm happy to have you choose something less familiar to me if you are willing to be more independent and put up with some bumps. The most important thing is to pick a topic you are excited about.

There is an opportunity to present your work at the end of the semester. It could be a good opportunity to make a LaTeX presentation with Beamer and work on your presentation skills, but this is optional. I will insist that you frequently write up your progress, which will become a nice report at the end of the semester. You could also combine math, code and images in an Jupyter Notebook, like this senior thesis one of my colleagues supervised.

Runge-Kutta Methods

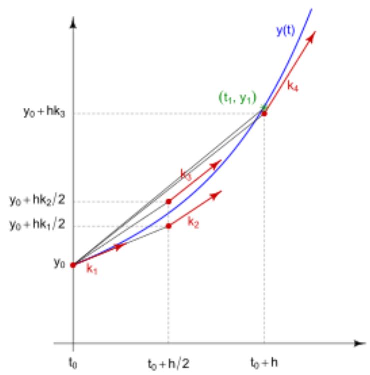
In Math 128a, Runge-Kutta methods are introduced as one of the main time integrators for initial value problems

$$y' = f(t, y) \quad y(t_0) = y_0.$$

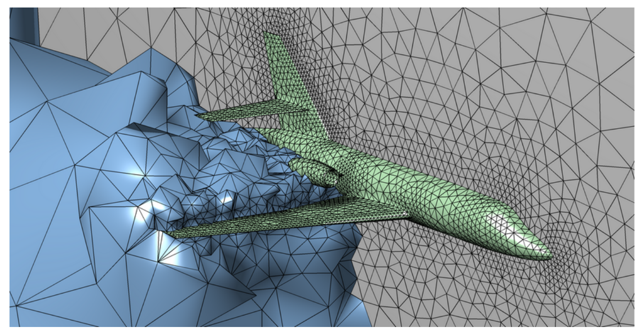
I propose a survey of the major classes of methods (Collocation Based, Diagonally Implicit, RADAU, Lobatto, etc) along with other features like adaptivity. We will examine their theoretical properties and derivation, which draw from many areas of math such as graph theory and linear algebra, in addition to algorithmic considerations and numerical performance. We will also implement them on a variety of test problems, and examine which classes of problems each method is best suited for. Good references begin with any introductory numerical analysis textbook, quickly movinginto the classic texts of Hairer, Norsett, and Wannier. There are also many survey and seminal research papers out there that should also be fairly accessible.

Mesh Generation

Mesh generation is the subdivision of a geometric region into smaller discrete cells, such as triangles. It is often the first step for the numerical solution of a partial differential equation and also frequently used in computer graphics. Good references come from two course pages from Jonathan Shewchuk in the CS department: CS 274: Computational Geometry and CS 294-74: Mesh Generation and Geometry Processing in Graphics, Engineering, and Modeling. Another good reference is a survey paper written by Steve Owen. We can start implementing and studying the three major algorithms for triangular mesh generation, advancing front, octree, and delaunay triangulation. Depending on interest, we can focus on more "high level" meshing issues or more "low level" geometrical aspects. There are plenty of extensions to pursue, such as other shapes (quadrilaterials, hexagons), 3D, or curved meshes. This topic would be fairly coding and visualization intensive, and would result in a lot of pretty pictures!



(a) 4th order Runge-Kutta method



(b) 3D Tetrahedral Mesh of a Falcon Jet

Randomized Numerical Linear Algebra

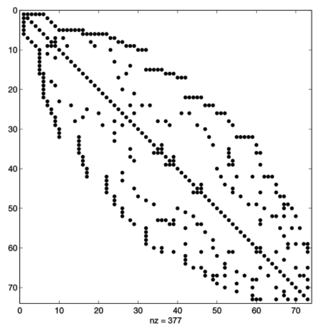
The use of randomization in numerical linear algebra is a fairly recent and powerful development to approach many of the classical problems of NLA: eigenvalue problems, low rank approximation, matrix factorizations, etc. We will study the classical algorithms for some of these problems (texts by Trefethen and Bau or Demmel) and quickly move to their randomized counterparts. The key references are the hugely influential survey by Halko, Martinsson and Tropp (2010) and their recent update (2020).

Singular Value Decomposition (SVD)

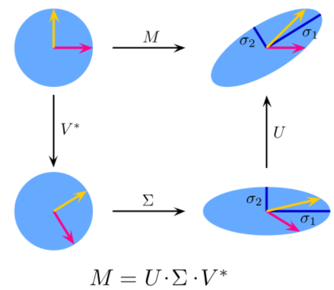
The SVD is a matrix factorization that generalizes eigendecomposition to nonsquare matrices, and has applications to pretty much everything. I propose we begin with studying some of the basic theory, which is well covered in many surveys, such as this survey by Steve Brunton. Then we can proceed its numerical calculation and corresponding algorithms. A huge focus of this project would be to discuss and implement interesting applications, such as the usual data analysis (PCA), image processing (like this), and more interesting possibilities, such as this example where Cleve Moler, the creator of MATLAB, uses the SVD to quantify increasing partisanship in the U.S. Senate.

Top 10 Algorithms of the 20th century

In the January/February 2000 issue of Computing in Science and Engineering, Dongarra and Sullivan selected 10 algorithms with the greatest influence on science and engineering in the 20th century. In March 2016, Nick Higham presented a slightly revised list. These algorithms span a large number of applications and underlying ideas. Each of them will be



(a) Cuthill-McKee ordering of a matrix

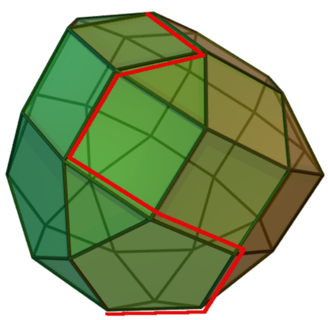


(b) Visual Interpretation of the SVD

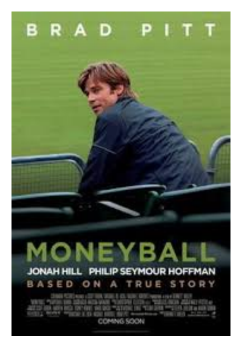
easy to find plenty of good references for, but a good starting point is a course offered by Alex Townsend of Cornell called Top Ten Algorithms from the 20th Century. We can pick one or two of these and implement and analyze them in depth.

Sports Statistics

Is the contract year phenomenon real? Do Icing the kicker strategies work? Should you take NBA players in foul trouble out? How effective are infield/outfield shifts? We could start with a study of the techniques employed in the sabermetrics revolution employed by the Oakland Athletics under Billy Beane. We can also look at papers in the two journals, the Journal of Sports Analytics and the Journal of Quantitative Analysis in Sports. We could also reproduce their statistical methods and apply them to similar problems. This project might involve a decent amount of data wrangling and scraping.



(a) Polyhedron of simplex algorithm in 3D



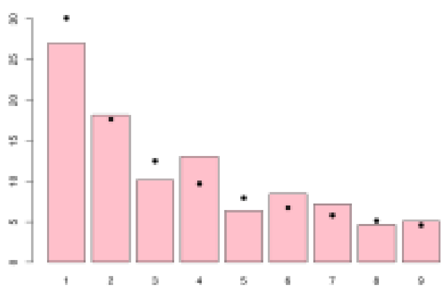
(b) Brad Pitt as Billy Beane in Moneyball

"Laws of Numbers" and Probability in the Real World

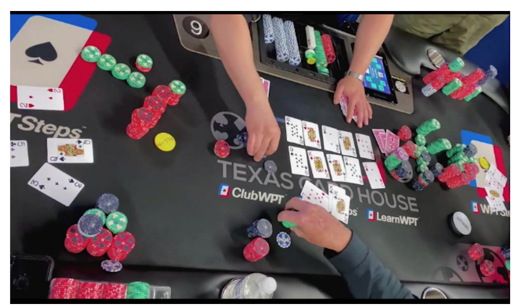
Benford's law is an observation that in real-life sets of numerical data the leading digit is likely to be small. If digits were distributed uniformly, then each digit (1-9) would occur $\frac{1}{9}$ of the time. But in many sets of data that obey the law, we observe that 1 is the leading significant digit 30% of the time, whereas 9 is the leading significant digit less than 5% of the time. We could study many different data sets where the law applies and the assumptions necessary, as well as consider many cool applications such as use in detecting accounting and election fraud. There are also many “laws” of a similar flavor such as Zipf’s law. More broadly, I think anything that falls under the umbrella “Probability in the real world” would be an interesting thing to study, even though I am far from an expert. You might be interested in a class taught by Professor of Mathematics and Statistics David Aldous on such topics.

Poker Solvers and Calculators

The most popular and well known form of poker today is probably Texas Hold’em, which experienced a huge spike of interest around the beginning of the 21st century largely due to online poker and the widespread available of telecasts. The game has since evolved from one being largely played by gamblers relying on ”feel” and ”instinct” to a more mathematically based approach known as “Game Theory Optimal” (GTO), which is seen by many as the holy grail of poker strategy, but also seen by many as ruining the game (not unlike how people feel about sabermetrics). I personally am a recreational low to mid stakes player who relies on feel (despite my quantitative background), but am interested in learning more about how this approach works and in particular the use of “solvers”, which is software that calculates optimal strategies for different user-inputted scenarios. If we were ambitious, we could also think about how these strategies extend to the 4-card game (Pot-Limit Omaha), and my personal favorite, the 4-card 2-board game (PLO Double Board Bomb Pots).



(a) Distribution of first digits in population of the world’s 237 countries



(b) PLO Double Board Action

Choose your own topic!

You can tell from this non-exhaustive list that my interests are rather diverse and also span both academic and non-academic topics. But what they all have in common is that I like to work on practical, applied things (rather than more abstract mathematical ones), get my hands dirty (examples! coding!) and also like to gain a deep understanding of things by doing lots of exposition and background. I highly encourage you to think about your own topic and approach you want to take to it. It can be pretty vaguely formed at first and I can help you turn it into a more concrete idea that’s appropriate in scope for a semester long reading project. We can also tailor it in scope to emphasize certain skillsets you want to further develop. Again, the single most important thing is that we choose a topic you are excited about!