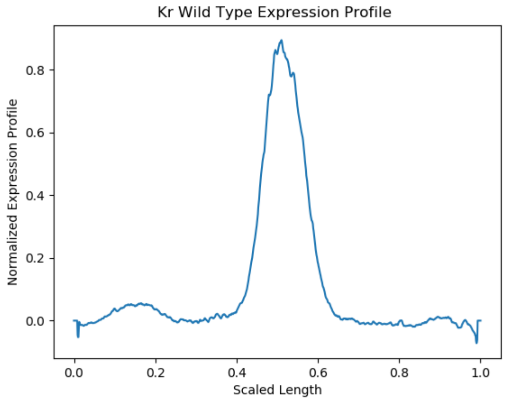
18.337 Project Proposal

Alasdair Hastewell

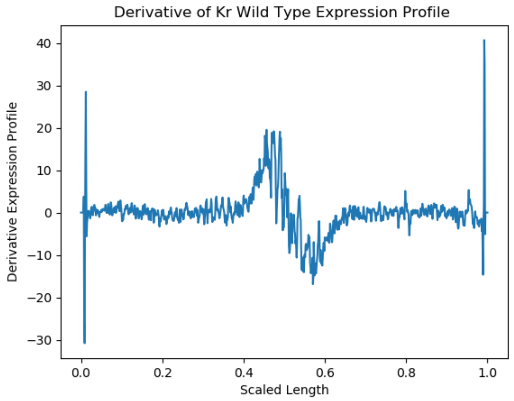
October 11, 2018

1 Introduction and Background

Often theory makes predictions about quantities that cannot be measured experimentally and, therefore, have to be calculated from other measurements. This is fine when data is smooth and noise free but becomes challenging when data contains fluctuations due to noise. As an example, one possible extension of the theory of positional information is to consider the effect of including gap gene expression gradients, in the calculation [4]. A sample expression profile is shown below,



Plot of the wild type expression profile of the gap gene Kr in *Drosophila* embryos. Data taken from [3]. We can see that there are small oscillations in the main profile shape.



Derivative of the wild type expression profile of the gap gene Kr using a centered second order finite difference scheme. We can see that profile is very jerky because of the noise in the raw profile.

The derivatives of this profile found using a finite difference scheme are jerky and do not represent a good approximation of the derivative of the smooth underlying profile. We cannot, therefore, use this derivative in calculation of quantities to compare to theory.

Another main goal of theory is to derive equations of motion of problems. In general we do this using conservation laws and other physical principles. This process can become untenable, however, when the system we are considering becomes arbitrarily complicated. As this has happened, however, our ability to collect and store increasingly complicated data sets has increased. This has led to the question, is it possible to use data do find dynamics? The problem has been study in depth by Brunton et al . [1, 5, 6] using sparse regression on a possible library of functions calculated from the data. Matlab and Python scripts exist that implement these algorithms but they have not yet been implemented in Julia. This methodology is highly sensitive to noise, however, which means that using this method on raw data yields meaningless results. How then can we use this method to find the dynamics of biological systems that are inherently noisy and stochastic?

First introduced in 2002 the Chebfun package in Matlab and its newer close Julia sibling ApproxFun provide functionality to manipulate functions in a similar way to vectors [2]. We can use these packages to manipulate datasets and calculate desired quantities The idea is to find expansions of the data in a given basis and then use these to calculate the desired quantities is an efficient method. Similar to a high pass filter we can also use the expansions to smooth the profiles. I used such an approach for the senior thesis using Chebfun.

Many issues still exist, however, Chebfun and ApproxFun are set up to use points on non-equally spaced grids such as the Chebyshev points. Datasets are normally not taken on such grids however. Chebfun provides a simple 'equi' flag to take care of this for equally spaced data. This functionality does not yet exist in ApproxFun, however, which makes using the package with data more complicated.

2 Problem Statement

The problem I hope to address in this project is how can we manipulate data and calculate theoretical values from them using Julia. The first half of the project would involve translating code from my thesis in Matlab into Julia. This would involve writing the equivalent of an 'equi' flag for ApproxFun in both one and two dimensions. It would also require me to write a Monte Carlo algorithm. The breadth of ApproxFun also allows for the possibility of comparing different Spaces easily.

The second half of the project would involve writing sparse regression algorithms in Julia and attempt to learn the dynamics of the gap gene network using them. The project will provide the opportunity to compare Julia to Matlab where much of the previous work in this field has been done.

3 Scope and Importance

Having an 'equi' flag for ApproxFun would make using ApproxFun when values are known rather than the functional form possible. This extends the power of ApproxFun to many other situations, similar to how Chebfun can be used now in Matlab. This can calculating quantities from data efficient and easy.

Having a smooth way to feed the output of this procedure into a dynamics learner would greatly increase the efficiency with which the dynamics of experimental systems can be studied. The predictions from this model can then be used to guide future experiments.

4 Software

The main packages that I would use are,

- DelimitedFiles
- CSV
- ApproxFun
- (maybe) FastTransforms
- (maybe) FastGaussQuadrature

The goal is to make the algorithms as general as possible so that they can be applied to a broad range of possible data sets.

References

[1] Steven L Brunton, Joshua L Proctor, and J Nathan Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences*, page 201517384, 2016.

[2] Tobin A Driscoll, Nicholas Hale, and Lloyd N Trefethen. Chebfun guide, 2014.

[3] Julien O Dubuis, Reba Samanta, and Thomas Gregor. Accurate measurements of dynamics and reproducibility in small genetic networks. *Molecular systems biology*, 9(1):639, 2013.

[4] Alasdair Hastewell. Infuence of gene expression gradients on positional information content in fly embryos, June 2018.

[5] Niall M Mangan, Steven L Brunton, Joshua L Proctor, and J Nathan Kutz. Inferring biological networks by sparse identification of nonlinear dynamics. *IEEE Transactions on Molecular, Biological and Multi-Scale Communications*, 2(1):52–63, 2016.

[6] Samuel H Rudy, Steven L Brunton, Joshua L Proctor, and J Nathan Kutz. Data-driven discovery of partial differential equations. *Science Advances*, 3(4):e1602614, 2017.