Abstract

Many regulation networks and control systems may be modeled using systems of ordinary differential equations. These systems are usually unknown, however it is possible to approximate them using multivariate regression. We will use sums of separable functions in the regression models. Through the analysis of these models one can then identify the contributions of individual components to the overall network. Here we outline a multivariate regression model using sums of separable functions. Our focus is on the analysis of the performance of these algorithms and the effect of the structure of tensor product spaces.

Learning and Analysis of Regulatory Networks with Sums of Separable Functions

Ryan T. Botts

Department of Mathematics
Ohio University

Introduction to gene regulatory networks

Genes contain information to perform functions, but do not actually perform them.

Genes code for mRNA, their physical expression, which go into cytoplasm and are converted into proteins to perform specific functions.

Question: What determines when and how the genes will be expressed?

Def: A gene regulatory network (GRN) is a networks of interacting genes which are each regulated by the other genes in the system as well as control parameters, e.g. metabolite levels.

Increasing the levels one gene may cause others to increase (promotion) or decrease (inhibition) or some combination of the two.

Represent d genes at time t as
$\mathbf{x(t)} = (x_1(t), x_2(t), \ldots x_d(t))$.

Represent k control parameters at time t as
$\mathbf{z(t)} = (z_1(t), z_2(t), \ldots z_k(t))$.

Only model the gene expression levels as a system of first order differential equations:

$$\dot{\mathbf{x}} = \mathbf{g}(\mathbf{x}, \mathbf{z}). \tag{1}$$

This is a deterministic system as knowing $\mathbf{g}$ would entirely determine how the genes interact.

Usually $\mathbf{g}$ is unknown.

Our model does not distinguish between control parameters and gene expression levels, so we write $\mathbf{g}(\mathbf{x})$.

Frequently one can perform microarray experiments to obtain data of the form $\{\mathbf{x}^j\}_{j=1}^N$ where $\mathbf{x}^j = (x_1(t_j), x_2(t_j), \ldots x_d(t_j))$.

Method

1. Use finite differences to approximate each $\mathbf{g}$

$$\mathbf{g}(\mathbf{x}(t_j)) = \dot{\mathbf{x}}(t_j) \approx \Big(\mathbf{x}(t_{j+1}) - \mathbf{x}(t_{j-1})\Big) \,/\, \Big(t_{j+1} - t_{j-1}\Big) \,.$$

2. These give us a list of data $D = \{(\mathbf{x}^j, \mathbf{y}_j)\}_{j=1}^N$

3. Fit data using an alternating least squares (ALS) routine with sums of separable functions. (Will fit each component in $\mathbf{y}$ independently.)

4. Develop new techniques to identify these interactions from the approximation to $\mathbf{g}$.

Current Project Goals

- Analyze the ALS algorithm to determine when it does not obtain good approximations.
- Understand when the ALS converges slowly.
- Analyze the structure of tensor product spaces (sums of separable functions).
- Understand what good choices for for a regression model.

Summary of Regression with Sums of Separable Functions

Define the pseudo-norm generated by the data-driven pseudo-inner product (extends to inner products over data)

$$\langle f, h \rangle = \frac{1}{N} \sum_{j=1}^{N} f(\mathbf{x}^j) h(\mathbf{x}^j) \,. \tag{2}$$

We will use **sums of separable functions** of the form

$$f(\mathbf{x}) = \sum_{l=1}^{r} s_l \prod_{i=1}^{d} f_i^l(x_i). \tag{3}$$

Currently we assume that $f_i^l(x_i) = \sum_{k=1}^{M} c_k^{i,l} \phi_k(x_i)$, for some orthogonal set $\{\phi_k\}_{k=1}^{M}$.

We want to minimize
$\|D - f\|^2 = \|D - \sum_{l=1}^{r} s_l \prod_{i=1}^{d} f_i^l(x_i)\|^2$.

Nonlinear optimization problem so we use an alternating least squares (ALS) approach.

Idea: Alternate through each direction x_i independently.

Yields the normal equations $\mathbb{A}\mathbf{z} = \mathbf{b}$ in the first iteration. Where

$$A(k,l;k',l') = \frac{1}{N}\sum_{j=1}^{N}\left(\phi_k(x_1^j)p_j^l\right)\left(\phi_{k'}(x_1^j)p_j^{l'}\right) ,$$

and

$$b(k,l) = \frac{1}{N}\sum_{j=1}^{N} p_j^l \phi_k(x_1^j) y_j ,$$

where $p_j^l = s_l \prod_{i=2}^{d} f_i^l(x_i^j)$.

Solve this and update f_1 and then proceed to next direction.

The solution is not optimal but usually a good approximation. We have also developed variations of this algorithm.

Recognizing Sums of Separables as Tensor Products

The important structure of these functions come from viewing them as sums of products of vectors.

Suppose we have a Hilbert space H with basis $\{e_i\}_{i=1}^{M}$

Let $e(i_1, \ldots, i_d) = e_{i_1} \otimes \cdots \otimes e_{i_d}$. Define the tensor product space

$$H^{\otimes d} = \left\{ \mathbf{V} \quad | \quad \mathbf{V} = \sum_{i_1 \leq M} \cdots \sum_{i_d \leq M} c_{i_1,\ldots,i_d} e(i_1, \ldots, i_d) \right\},$$

$H^{\otimes d}$ is a vector space of dimension M^d. Sum vectors can be written with fewer sums.

Define rank($\mathbb{S}$) $= r$ to be the least r such that

$$\mathbb{S} = \sum_{k=1}^{r} \bigotimes_{i=1}^{d} \mathbf{S}_i^k \text{ where } \mathbf{S}_i^k \in H$$

Find the best rank-r approximation of a tensor $\mathbb{S} \in H^{\otimes d}$.

Difficult problem, so we try a simpler problem.

Consider *symmetric tensors* where permuting the indices i_k results in the same tensor. Example:

$$\mathbb{S} = \bigotimes_{i=1}^{2} [0,1] - \bigotimes_{i=1}^{2} [1,0]$$

Simple case: approximate with a rank-1 tensor

$$\mathbb{F} = \bigotimes_{i=1}^{d} \mathbf{F}_i$$

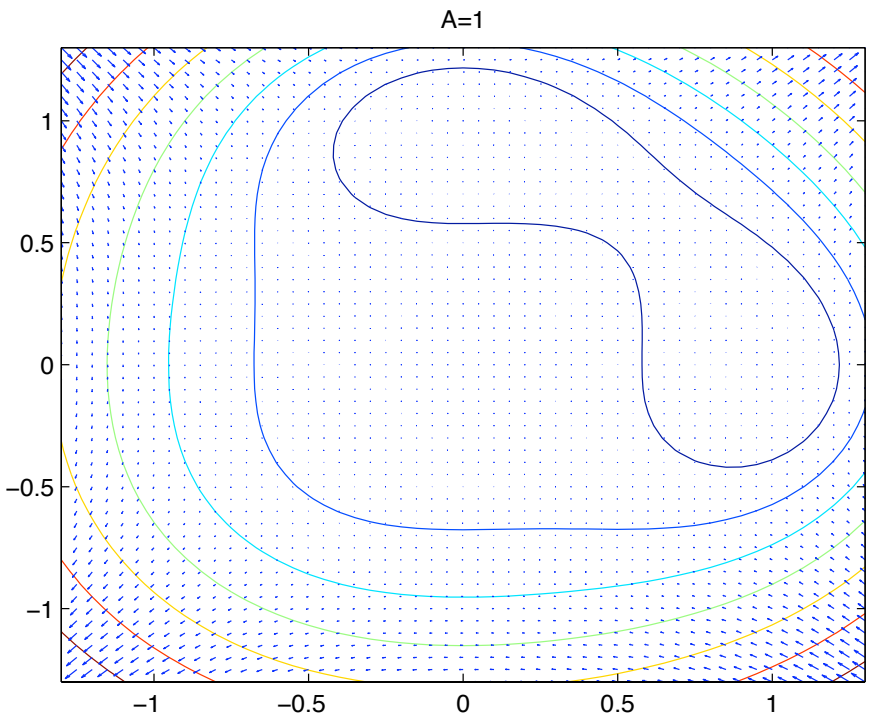
Found the best approximation need not be symmetric, but if it is not there is an equally good symmetric approximation.

Hope to identify the number of disconnected best approximations.

This lead us to identify large regions that are nearly *stationary points*, i.e. gradients of least squares error is very small

A plot of $\|\mathbb{F} - \mathbb{S}\|^2$ in the case where

$$\mathbb{S} = A \bigotimes_{i=1}^{3} [1, 0] + \bigotimes_{i=1}^{3} [0, 1] \text{ and } \mathbb{F} = \bigotimes_{i=1}^{3} [a, b]$$



Current Work

- Identify the number of isolated best approximations.
- Identify the number of local minima.
- Find examples of non-symmetric local minima.
- In many cases, ALS still finds the symmetric approximation. Why?

- When does the ALS avoid local minima?

- What are strategies for avoiding local minima?

- Can we generalize the results of approximating symmetric tensors to non-symmetric tensors?