Inferring Transcription Factors Protein Activities by Combining Binding Information via Gaussian Process Regression

Sura Z. Alrashid[1] **Nabeel H. Al-Aaraji[2]**
[1]*Lecturer, Software Department/ Information Technology/ Babylon University,*
sura_os@itnet.uobabylon.edu.iq
[2]*Proffesor, Ministry of Higher Education and Scientific Research,*
nhkaghed@yahoo.com

Abstract

The most basic step in understanding gene regulated is performed by identifying the target genes regulated by transcription factors (TFs) Proteins. Protein is produced by Transcription factors Proteins that promote or repress transcription of other genes; they play a very important role in gene networking and affecting for occurring the disease. The analysis of gene expression of time series underpins various biological studies. This work has focused on the difference in transcriptional regulation between two strains of mice. The mice were considered in two forms Wild type SOD1-G93A and Ntg mutations (SOD1 is a transcription factor Protein that induces ALS). The data interest because the phenotype of the two mutant strains differs. One of the strains succumbs to ALS far quicker than the other; we suggested a model to infer Transcription Factor Proteins Activities and correlated with genes targeted. We build Gaussian process with particular covariance function for reconstructing transcription factor activities given gene expression profiles and a connectivity matrix, and we introduce a computational trick, based on Singular Value Decomposition (SVD) to enable us to efficiently fit the Gaussian process in a reduce 'TF activity ' space. Performing the basic step in understanding regulated genes is identifying these genes by transcription factors. Gaussian processes offer an attractive trade-off between usability and efficiency for the analysis of microarray time series. The Gaussian process framework with Coregionalization model offer a natural way of handling biological replicates and correlated output and inferred the activity of Transcription factors Proteins for four cases the genes alter its behavior, we proved the significates TF using DAVID to analysis pathway.

Keywords: Transcription Factors Proteins, Gene expression, Gaussian Processing regression, Coregionalization Model, Covariance Function, Singular Value Decomposition.

الخلاصة

الخطوة الاساسية في فهم الجينات المنظمة المنجزة بتحديد الجينات من قبل عوامل النسخ البروتينية(TF).حيث ان البروتين الممنتج من قبل عوامل النسخ البروتينية يعزز او يقمع نسخ جينات اخرى,هذه الظاهرة تلعب دور مهم جدا في التواصل الجيني وتسبب حدوث المرض. التحليل الجيني للسلاسل الزمنية تدعم الدراسات البيلوجية المختلفة . وقد ركز هذا العمل على الفرق في التنظيم النسخي بين سلالتين من الفئران. حيث أُعتبرت الفئران في شكلين: نوع البرية SOD1−G93A ونوع الطفرات الوطنية للتقنية (SOD1 هو عامل نسخي البروتيني الذي يسبب حدوث مرض التصلب الضموري العصبي الجانبي ALS). تعتبر هذه البيانات مهمة بسبب النمط الظاهري بين سلالتين مختلفتين من الطفرات. واحدة من تلك السلالتين تسبب الALS أسرع بكثير من الاخرى، في هذا البحث اقترحنا نموذجا للاستدلال نشاط العامل النسخي البروتيني والذي مرتبط مع مجموعة من الجينات .حيث بدأءنا ببناء موديل رياضي يعتمد على كاوسين مع دالة التغاير لاستدلال نشاطات تلك العوامل النسخية البروتينية حيث أعطت ملامح التعبير الجيني بالاضافه الى الاستفادة من معلومات الربط بينهم. في هذا البحث استخدمنا طريقة تحليل القيمة المنفردة (SVD) لتقليص التعقيد الاحتسابي لدالة التغاير.هذا المديل قدم الخطوة الأساسية في فهم نسخ الجينات وتحديد عوامل النسخ البروتينية، حيث قدم هذا الموديل سهولة الاستخدام والكفاءة لانجاز هذا الهدف. حيث قدمت عملية التنبا باستخدام كاوسين مع موديل ال Coregionalization وسيلة طبيعية للتعامل مع المكررات البيولوجية والاخراجات المترابطة واستدلال نشاط العوامل النسخية البروتينية لأربع حالات فيها الجينات تغير سلوكها وكذلك برهنة علاقة تلك العوامل المستنتجة من قبل نموذجنا الرياضي باستخدام تحليل المسارات .DAVID

الكلمات المفتاحية: العوامل النسخية البروتينية، التعبير الجيني، الانحدار الخطي كاوسين، موديل Coregionalization، دالة التغاير، تحليل القيمة المفردة.

1 Introduction

The regulatory Quantitative estimation relationship between genes and transcription factors is a basic step to develop cellular processes models. This task, however, is difficult for two reasons: levels of the transcription factor's expression are always noisy and low, as well as most transcription factors are post-transnationally regulated. It is therefore, useful to infer the transcription factors Proteins activity from their target genes expression levels.

In [Sanguinetti *et al.*, 2006b] they proposed a probabilistic model where this method was extended the linear regression model that proposed by [Lio et al., 2003] to model the full probability distribution of each activity of transcription factor on each gene, they used Markov chain model and the covariance structure of the transcription factors, it is shared among genes, that is leading to a manageable parameter space and useful information about the correlation of TFAs. They demonstrated their model on two yeast data sets cell cycle data and metabolic cycle data set. Their model provided new predictions where it light some aspects of the regulatory mechanism of the cell for example the repress of the TF from negative gene-specific.

A probabilistic state space model has been developed by [Sanguinetti *et al.*, 2006b] to allow inference of both concentrations of Transcription factor proteins and their effect on the rates of the transcription of each target gene from microarray data, where they use Expectation and Maximization method as vibrational inference techniques to learn the model parameters and per- form posterior of protein constraints and regulatory strengths with model the temporal structure of the data by using a Markov chain. They applied their model on artificial data and on tow yeast datasets, the exploit the natural sparsity of regulatory network considered the key feature of their model, their model is dynamic and it can account for the temporal structure of data. EMBER is model integrates high-throughput binding data (e.g. CHIP-seq or CHIP-chip)with gene expression data (e.g. DNA microarray) was presented by [Mark *et al.*, 2012] that it is abbreviated (Expectation Maximization of Binding and Expression pRofiles) it worked via an unsupervised machine learning algorithm for inferring the gene targets of sets of TF binding sites. They demonstrated their model by applying it on data for the TFs ERα and RARα and RARγ in breast cancer MCF-7 cells.In [Boulesteix and Strimmer, 2005] proposed a statistical approach based on partial least squares (PLS) regression to infer the true TFAs from a combination of mRNA expression and DNA-protein binding measurements. This method was also statistically sound for small samples and allowed the detection of functional interactions among the transcription factors via the notion of"meta"- transcription factors, [Sanguinetti et al., 2005] Principal Component Analysis (PCA) is one of the most popular techniques of dimensionality reduction for the high-dimensional datasets analysis. However, in its standard form, it does not take into account any error measures associated with the data points beyond a standard spherical noise. They proposed a new model-based approach to PCA that takes into account the variances associated with each gene in each experiment, they developed an efficient EM-algorithm to estimate the parameters of theirs new model. The model provided significantly better results than standard PCA, while remaining computationally reasonable. Most methods aim to infer a matrix of activities of transcription factor Proteins (TFAs), which are supposed to sum up in a single number the concentration of the transcription factor at a certain experimental point and its binding affinity to its target genes. The methods used are modified forms of regression. For example, [Gao *et al.*, 2008] used multivariate regression plus backward variable selection to identify active transcription factors; [Boulesteix and Strimmer, 2005] estimate TFAs using partial least squares, [Liao *et al.*, 2003] proposed analysis of network component, a technique for dimension reduction which takes account of the connectivity information by imposing algebraic constraints on the factors

The aim of our paper is specify the significantly differences genes that infect in speed of ALS disease progression, Deduce the transcription factors' proteins activity from data of the mRNA expression. Suggesting a model depended on [Neil *et al.*, 2006a] to infer Transcription Factor Proteins Activities and correlated with genes that previously selected. Make approach focuses on inference of context specific networks that including all genes targeted and a few interacting transcription factors Proteins. Identification of a set of genes related with significant biological functions associated. Design a covariance function for reconstructing activities of transcription factor given profiles of gene expression and a connectivity matrix (binding data) between transcription factors and genes.

All Methods that are used in this paper where we described MmGmos function from puma package [Richard *et al.*, 2009] and Coregionalization model and Singular Vector Decomposition in Section 2, A Gaussian process approach to model the gene expression profiles, proposed model are discussed in Section 3. The utility of the proposed method is illustrated by real case study in Section 4. We discussed the showed results and conclusion in Section 5 and 6. Respectively.

2 Methods

2.1 Coregionalization model

The linear Coregionalzation model indicates to models the outputs are expressed as combinations that the linear correlation of independent random functions. If these functions are Gaussian processes, then the model result a Gaussian process with covariance function has a positive semi definite [Emery and Maria 2012;Bohling, 2005; Goovaerts, 1992;Goulard and Voltz, 1992].

Assuming D outputs $\{f_d(x)\}_{d=1}^{D}$ with$x \in R^p$, each $f_d$is expressed as:

$$f_d(x) = \sum_{q=1}^{Q} a_{d,q}\, u_q(x) \qquad (2\text{-}1)$$

Where $a_{d,q}$scalar coefficients and the independent functions are$u_q(x)$ have zero mean and covariance

$$\mathrm{cov}\left[u_q(\mathrm{x}), u_{q'}(x')\right] = \begin{bmatrix} k_q(\mathrm{x};\ x') & \text{if } q = q' \\ 0 & \text{Otherwise} \end{bmatrix} \qquad (2\text{-}2)$$

Between any two functions $f_d(x)$ and $f_{d'}(x)$ the cross covariance can then be written as:

$$cov[f_d(x), f_{d'}(x')] = \sum_{q=1}^{Q}\sum_{i=1}^{R_q} a_{d,q}^{i} a_{d',q}^{i} k_q(x,x') \qquad (2\text{-}3)$$
$$= \sum_{q=1}^{Q} b_{d,d'}^{q} k_q(x,x')$$

Where the functions $u_q^i(x)$, with $i = 1, \ldots, R_q$ and $q = 1 \ldots Q$ have mean = 0 and covariance$cov\left[u_q^i(x), u_{q'}^{i'}(x)' = k_q(x,x')\ if\ i = i'\right]\ and\ q = q'$. But $\mathrm{cov}\left[f_q(\mathrm{x}), f_{q'}(x')\right]$ Is given by $((x,x'))_{d,d'}$ Thus the kernel$K(x,x')$ can be written as

$$K(x,x') = \sum_{q=1}^{Q} B_q k_q(x,x') \qquad (2\text{-}4)$$

Where each $B_q \in R^{D\times D}$ is called a Coregionalzation matrix. Therefore, the kernel can be derived from LMC is a sum of the products of two covariance functions, one these models the input dependence, independently of $\{f_d(x)\}_{d=1}^{D}$ (the covariance function $K_q(x,x')$) and one that models the dependence between the outputs, independently of the input vector x (the Coregionalzation matrix B_q) [Han and Micheline, 2006; Finazzi *et. al.*, 2011; Lopez-Kleine *et. al.*, 2013].

2.2 Singular Value Decomposition

The Singular Value Decomposition (SVD) is usually a factorization of the complicate as well as real matrix, basically, the SVD is $R\Lambda V^T$ of an m × n matrix where

Λ is an m × n rectangle-shaped diagonal matrix together with non-negative real numbers within the diagonal, R is real matrix that m× m,

V^T the conjugate transpose associated with V, or just the actual transpose of V when V is actually real) is an n × n real or complex unitary matrix. The diagonal items $\Lambda_{ij}$of Σ are usually referred to as the singular values associated with S. The m columns of R and the n columns of V are called the actual left-singular vectors and right-singular vectors of S, respectively.

The SVD is usually traditionally used technique to decompose a matrix directly into several component matrices, revealing many of the beneficial in addition to useful attributes of the original matrix. The decomposition of a matrix is often known as a factorization. Essentially, the matrix is usually decomposed directly into a collection of factors (often orthogonal or maybe independent) that are best determined by a few requirements [Van *et. al.*, 2010].

3 Data Set

We demonstrate our model by applying it to Mice model for Amyotrophic lateral sclerosis (ALS) (Lou Gehrig's disease) “Amyotrophic lateral sclerosis is a severe neurodegenerative disease, that adult-onset characterized by progressive premature loss of lower and upper motor neurons” [Ana *et. al.*, 2012; Julia, 2012]. where this data generated by affymatrix GeneChip Operating System were analyzed by [Alice *et. al.*, 2013; Giovanni *et al.*, 2013] that Amyotrophic lateral sclerosis is heterogeneous with high variability in the progression speed even in cases with a defined genetic cause such as mutations of superoxide dismutase 1 (SOD1).

3.1 Transcription Factors

Protein-DNA interactions play crucial roles in many key biological processes. One of these processes is transcriptional regulation, in which transcription factors (TFs) bind to specific DNA binding sequences to either activate or repress the expression of their regulated genes [Jiadong *et al.*, 2012]. The term transcription factor is used to refer to the specific transcriptional activators and repressors that activate or repress the transcription of target genes via specific binding to promoter regions [CHENG, 2007; Esther, 2006].

4 The Proposed System

This work highlights a set of key gene and molecular pathway indices of slow or fast progression of disease in the two transgenic mouse models which may prove useful in identifying potential disease modifiers responsible for the heterogeneity of human ALS and which may indicate valid therapeutic targets in humans [Nardo *et. al.*, 2013; Julia, 2012]. The general steps of this work are explained as Block Diagram is showed in figure (1) , In the beginning , we download the Data Sets, then analyzing these .cel files to computing the Gene expressions, then following it standardizes the Data (Y) that is gene expression values for two strains by two changes by four reproduces where its dimensional (num_{genes}(PorbeID), num_{points}) in standardize or normalize step we utilize the statistical-t student as (1)

$$\text{Student's t} - \text{statistic} = \frac{\text{Y} - \bar{Y}}{s} \quad (1)$$

Where $\bar{Y}$ mean of Y, S is standard deviation.
It is input with set of parameters to our model, In building step we mean building the matrix of time series that contains the points of strains and mutations and replicates, where the dimensionality of this matrix is (num_{times} series, corgionalize-dim for two strains and two mutations, Corgionalize_dim for TF_no) in this part the X is (64, 3) .

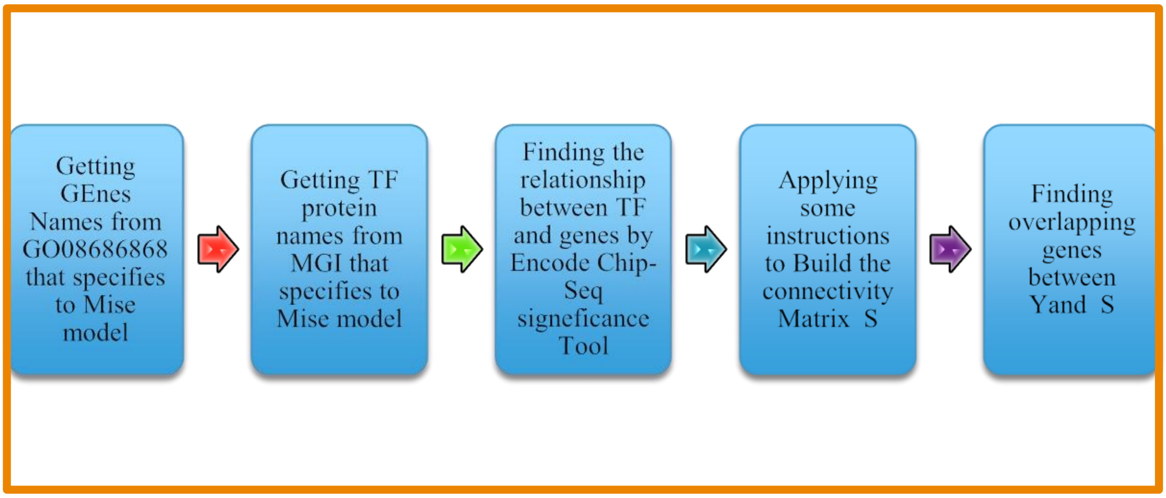


Figure 1: Shows The preprocessing steps of Our Work.

1.1.1 Binding Matrix

We got on Transcription Factors for Mice model from the open source Mouse Genome Informatics (MGI) that is resource of the international database for the laboratory mouse, providing genomic, integrated genetic, and biological data to facilitate the human health and disease study, and then we used the Encode Chip-Seq significance Tool that is a Simple Web Tool to Identify Enriched ENCODE Transcription Factors From a List of Genes or Transcripts via some steps and then built the Binding Matrix that contains 1 if there is relationship between TF and genes else 0.These steps we mention as block Diagram as in figure(above), This data consists of the expression profiles of 45038 genes measured at 4 equally spaced time points (4 stages to progress the ALS) and in each time it contains two strains in each strain contains two mutations and with it's role contains four replicates and then integrate it with 69 transcription factors.

1.1.2 Model for Transcription Factor Activities

We are working with log expression levels in a matrix $\boldsymbol{Y} \in \mathbb{R}^{n \times T}$ and we will assume a linear (additive) model giving the relationship between the expression level of the gene and the corresponding transcription factor activity, which are unobserved, but we represent by a matrix $\boldsymbol{F} \in \mathbb{R}^{q \times T}$. Our basic assumption is as follows. Transcription factors are in time series, so they are likely to be temporally smooth. Further, we assume that the transcription factors are potentially correlated with one another (to account for transcription factors that operate in unison) [Hashimoto, 2014].

1.1.3 Correlation between Transcription Factors

If there are $\boldsymbol{q}$ transcription factors then correlation between different transcription factors is encoded in a covariance matrix, $\boldsymbol{\Sigma}$ which is $\boldsymbol{q} \times \boldsymbol{q}$ in dimensionality [Meng *et. al.*, 2011]. Temporal Smoothness Further we assume that the log of the transcription factors activities is temporally smooth, and drawn from an underlying Gaussian process with covariance $\boldsymbol{K_t}$.

1.1.4 Intrinsic Coregionalzation Model,

We assume that the joint process across all $\boldsymbol{q}$ transcription factor activities and across all time points is well represented by an intrinsic model of Coregionalzation where the covariance is given by the Kronecker product of these terms.

$$\boldsymbol{K_f = K_t \otimes \Sigma} \tag{2}$$

This is known as an intrinsic Coregionalzation model see [Alvarez *et al* 2011] for a machine learning orientated review of these methods. The matrix $\boldsymbol{\Sigma}$ is known as the coregionalization matrix.

1.1.5 Relation to Gene Expressions

We now assume that the $\boldsymbol{j^{th}}$ gene's expression is given by the product of the transcription factors that bind to that gene. Because we are working in log space, that implies a log linear relationship. At the $\boldsymbol{i^{th}}$ time point, the log of the $\boldsymbol{j^{th}}$ gene's expression $\boldsymbol{y_{ij}}$, is linearly related to the log of the transcription factor activities at the corresponding time point $\boldsymbol{f_{i,:}}$. This relationship is given by the binding information from S. We then assume that there is some corrupting Gaussian noise to give us the final observation.

$$\boldsymbol{y_{i,j} = Sf_{:,i} + \epsilon_i} \tag{3}$$

Where the Gaussian noise is sampled

1.1.6 Gaussian Process Model of Gene Expression

We consider a vector operator which takes all the separate time series in Y and stacks the time series to form a new vector n*T length vector y [Strippoli *et. al.,* 2005]. A similar operation is applied to form a q*T length vector f. Using Kronecker products we can now represent the relationship between y and f as follows:

$$\boldsymbol{y = [I \otimes S]f + \epsilon} \tag{4}$$

Standard properties of multivariate Gaussian distributions tell us that

$$\boldsymbol{y \sim \mathcal{N}(0, K)\ where\ K = K_t \otimes S\Sigma S^T + \sigma^2 I} \tag{5}$$

This results in a covariance function that is of size n by T where n is number of genes and T is number of time points. However, we can get a drastic reduction in the size of the covariance function by considering the singular value decomposition of S. The matrix S is n by q matrix, where q is the number of transcription factors. It contains a 1 if a given transcription factor binds to a given gene, and zero otherwise.

$$\boldsymbol{L = -\frac{1}{2}\log|K| - \frac{1}{2}y^T K^{-1} y} \tag{6}$$

In the worst case, because the vector y contains T*n points (T time points for each of n genes) we are faced with $O(T^3 n^3)$ computational complexity. We are going to use a rotation trick to help.

1.1.7 The Main Computational Trick

1.1.7.1 *Rotating the Basis of a Multivariate Gaussian*

For any multivariate Gaussian you can rotate the data set and compute a new rotated covariance which is valid for the rotated data set. Mathematically this works by first inserting RR^T into the likelihood at three points as follows:

$$\boldsymbol{L = -\frac{1}{2}\log|R^T K R| - \frac{1}{2}y^T R^T R K^{-1} R^T R y + const} \tag{7}$$

The rules of determinants and a transformation of the data allows us to rewrite the likelihood as

$$L = -\frac{1}{2}\log|R^T K R| - \frac{1}{2}\hat{y}^T[R^T K^{-1} R]^{-1}\hat{y} + const \quad (8)$$

Where we have introduced the rotated data $\hat{\mathbf{y}} = \mathbf{Ry}$. Geometrically what this says is that if we want to maintain the same likelihood, then when we rotate our data set by R we need to rotate either side of the covariance matrix by R, which makes perfect sense when we recall the properties of the multivariate Gaussian.

1.1.7.2 **A Kronecker Rotation**

In this paragraph, we are using a particular structure of covariance which involves a Kronecker product. The rotation we consider will be a Kronecker rotation. We are going to try and take advantage of the fact that the matrix S is square meaning that $S\Sigma S^T$ is not full rank (it has rank of most q, but is size $n \times n$, and we expect number of transcription factors q to be less than number of genes n).

When ranks are involved, it is always a good idea to look at singular value decompositions (SVDs). The SVD of S is given by:

$$S = Q\Lambda V^T \quad (9)$$

Where $\mathbf{V^T V = I}$, $\boldsymbol{\Lambda}$ is a diagonal matrix of positive values, Q is a matrix of size $n \times q$: it matches the dimensionality of S, but we have $\mathbf{Q^T Q = I}$. Note that because it is not square, Q is not in itself a rotation matrix. However it could be seen as the first q columns of an *n* dimensional rotation matrix (assuming *n* is larger than *q*, i.e. there are more genes than transcription factors).

If we call the n-q missing columns of this rotation matrix U then we have a valid rotation matrix R= [QU] although this rotation matrix is only rotating across the *n* dimensions of the genes, not the additional dimensions across time. In other words, we are choosing K_t to be unrotated. To represent this properly for our covariance we need to set $\mathbf{R = I \otimes [QU]}$. This gives us a structure that when applied to a covariance of the form $\mathbf{K_t \otimes K_n}$ it will rotate $\mathbf{K_n}$ whilst leaving $\mathbf{K_t}$ untouched.

When we apply this rotation matrix to K we have to consider two terms, the rotation of $\mathbf{K_t \otimes S\Sigma S^T}$, and the rotation of $\boldsymbol{\sigma^2}\mathbf{I}$. Rotating the latter is easy, because it is just the identity multiplied by a scalar so it remains unchanged

$$\mathbf{R^T I}\sigma^2\mathbf{R} = \mathbf{I}\sigma^2 \quad (10)$$

The former is slightly more involved, for that term we have

$$[\mathbf{I \otimes QU^T}]\mathbf{K_t \otimes S\Sigma S^T}[\mathbf{I \otimes QU}] = \mathbf{K_t \otimes QU^T S\Sigma S^T QU} \quad (11)$$

Since $S = Q\Lambda V^T$ then we have

$$\mathbf{QU^T S\Sigma S^T QU} = \begin{bmatrix} \mathbf{\Lambda V^T \Sigma V \Lambda} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \quad (12)$$

This prompts us to split our vector $\hat{y}$ into a n - q dimensional vector

$$\hat{y}_u = U^T y \quad (13)$$

And an q dimensional vector

$$\hat{y}_q = Q^T y \quad (14)$$

The Gaussian likelihood can be written as

L = L_u+ L_q + const

Where

$$L_q = -\frac{1}{2}\log\left|\mathrm{K_t} \otimes \Lambda V^T \Sigma V \Lambda + \sigma^2 I\right| - \frac{1}{2}\hat{y}_q{}^T[\mathrm{K_t} \otimes \Lambda V^T \Sigma V \Lambda + \sigma^2 I]^{-1}\hat{y}_q \quad (15)$$

And

$$L_u = -\frac{T(n-q)}{2}\ log\sigma^2\ -\frac{1}{2}\hat{y}_u{}^T\hat{y}_u \tag{16}$$

Firstly, we fit the noise variance σ^2on $\hat{y}_u$ alone using L_u. Once this is done, fix the value of σ^2in L_q and optimize with respect to the other parameters.

$$\frac{\partial L_u}{\partial \sigma} = -\frac{T(n-q)}{2\sigma^2} + \frac{1}{2\sigma^4}\hat{y}_u^T\hat{y}_u, \tag{17}$$

$$0 = \frac{[T(n-q)\sigma^2 + \hat{y}_u^T\hat{y}_u]}{4\sigma^4},$$

$$T(n-q)\sigma^2 = \ \hat{y}_u^T\hat{y}_u,$$

$$\sigma^2 = \frac{1}{T(n-q)}\ \hat{y}_u^T\hat{y}_u$$

In this moment, we make the prediction equations where we are using Kronecker product we can rewrite the Eq(5) as:

$$y_q \sim \mathcal{N}(0, \mathrm{K_t} \otimes \Lambda V^T \Sigma V \Lambda + \ \sigma^2 I) \tag{18}$$

Standard properties of multivariate Gaussian distributions tells us can split it into

$$y_q = g + \epsilon \tag{19}$$

Where g and ϵ are also Gaussian distributions and can be represented by:

$$g \sim \mathcal{N}(0, \mathrm{K_t} \otimes \Lambda V^T \Sigma V \Lambda) \tag{20}$$

$$\epsilon \sim \mathcal{N}(0, \sigma^2 I) \tag{21}$$

Now we can represent the matrix F of transcription factor activity as:

$$F = \mathrm{I} \otimes \Lambda V^T \tag{22}$$

$$\Sigma = WW^T + diag(\kappa) \tag{23}$$

Where κ is the kappa value from Coregionalization Matrix.

$$F \sim \mathcal{N}(0, \mathrm{K_t} \otimes \Sigma) \tag{24}$$

Now we can find the conditional distribution of g for given y_q by:

$$p(g|y_q) \sim \mathcal{N}(\mu_g, C_g) \tag{25}$$

With a mean given by:

$$\mu_q = [\mathrm{K_{t*t}} \otimes \Lambda V^T \Sigma V \Lambda][\mathrm{K_{tt}} \otimes \Lambda V^T \Sigma V \Lambda + \ \sigma^2 I]^{-1} y_q \tag{26}$$

And the covariance given by:

$$C_q = [\mathrm{K_{t*t*}} \otimes \Lambda V^T \Sigma V \Lambda] - [\mathrm{K_{t*t}} \otimes \Lambda V^T \Sigma V \Lambda]^T [\mathrm{K_{tt}} \otimes \Lambda V^T \Sigma V \Lambda + \ \sigma^2 I]^{-1} [\mathrm{K_{t*t}} \otimes \Lambda V^T \Sigma V \Lambda + \ \sigma^2 I] \tag{27}$$

The mean of the conditional distribution is:

$$\mu_F = [\mathrm{K_{t*t}} \otimes \Sigma V \Lambda][\mathrm{K_{tt}} \otimes \Lambda V^T \Sigma V \Lambda + \ \sigma^2 I]^{-1} y_q \tag{28}$$

And the covariance of the conditional distribution given by:

$$C_F = [\mathrm{K_{t*t*}} \otimes \Sigma] - [\mathrm{K_{t*t}} \otimes \Sigma V \Lambda]^T [\mathrm{K_{tt}} \otimes \Lambda V^T \Sigma V \Lambda + \ \sigma^2 I]^{-1} [\mathrm{K_{t*t}} \otimes \Lambda V^T \Sigma + \ \sigma^2 I] \tag{29}$$

Algorithm3-2: General Steps of our third work part
Inputs: Y the Data Set of Mice Models S is Connectivity Matrix between gene$_i$ and TF$_j$, where i=0,...,N, J=0,...,Q
Outputs: Inferred transcription Factors Activity
Step1: Call prepossessing procedure.

Step2: X ← Call Building of matrix of times series.
Step3: M ← Call the Gaussian Process Regression Model.
Step4: $M \leftarrow Optimize(M)$ to estimate and optimize the hyper-parameters.
Step5: $\boldsymbol{\mu_F}$ ← computing the mean using Eq (28)
and $\boldsymbol{C_F}$ ← computing the covariance using Eq(29)
Step6: Call Rank procedure to rank the models depending on Likelihood values.
Step7: Plot the Model depending on Y, mean, and var.
Step8: Compute F (TFA) and select it that effect on the progressing of ALS disease.
Step9: Check the Transcription Factors Names with the selected genes from Second part (Clustering Work), then we go to DAVID to analysis and proving these TF related with ALS disease.

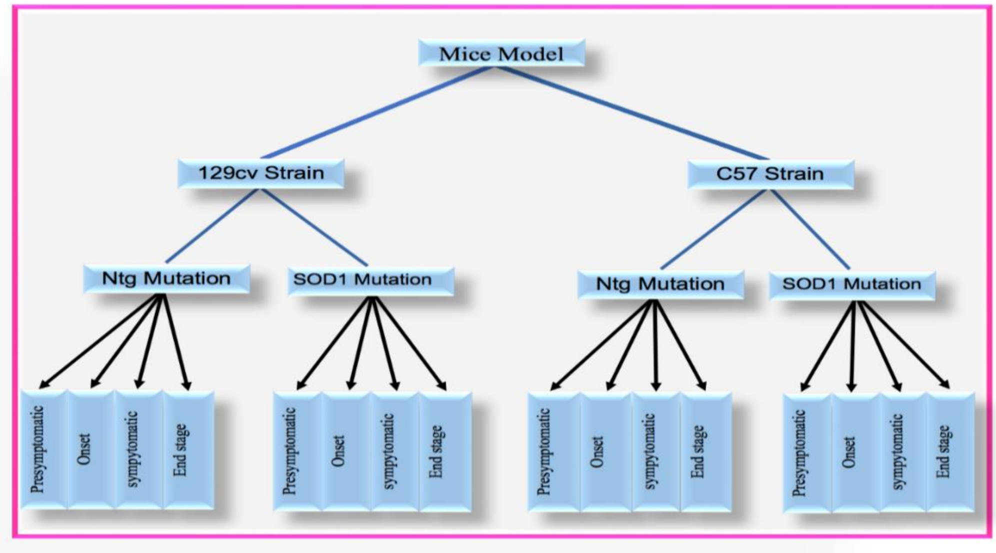
Algorithm3-3: Prepossessing Procedure
Inputs: Y the Data Set of Mice Models S is Connectivity Matrix between $gene_i$ and TF_j, where i=0,...,N, J=0,...,Q
Outputs: Y_q, $sigma^2$, V, Lambda, R
Step1: Filling the missing value by instead it with 0 values. Step2: Finding the overlapped genes between Y and S. Step3: R, $\Lambda\, V^T$ ← Singular Value Decomposition (SVD). Step4: Compute Y_q from Q and Y_u from u as Eq (13) and Eq (14) respectively. Step5: Compute sigma2 from Y_u as (17). Step6: Normalization Y_q values as Eq(1)

Algorithm3-4: Building of matrix of times series
Inputs: Time series, No. of Transcription Factors Protein, No. of mutation, No. of strains, and No. of replicates.
Outputs: X with its dimension.
Step1: s_m ← asarray([0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0, \ 1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1, \ 2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2, \ 3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3]) Step2: x0, x1← make tow vector the theirs dimension asarray(meshgrid(flatten(t), arange(q))) Step3: x2,_ ← vector(meshgrid(flatten(s_m),arange(q))). Step4: X ← concatenation horizontally the three vectors to make X The three vectors are x0, x1,x2.

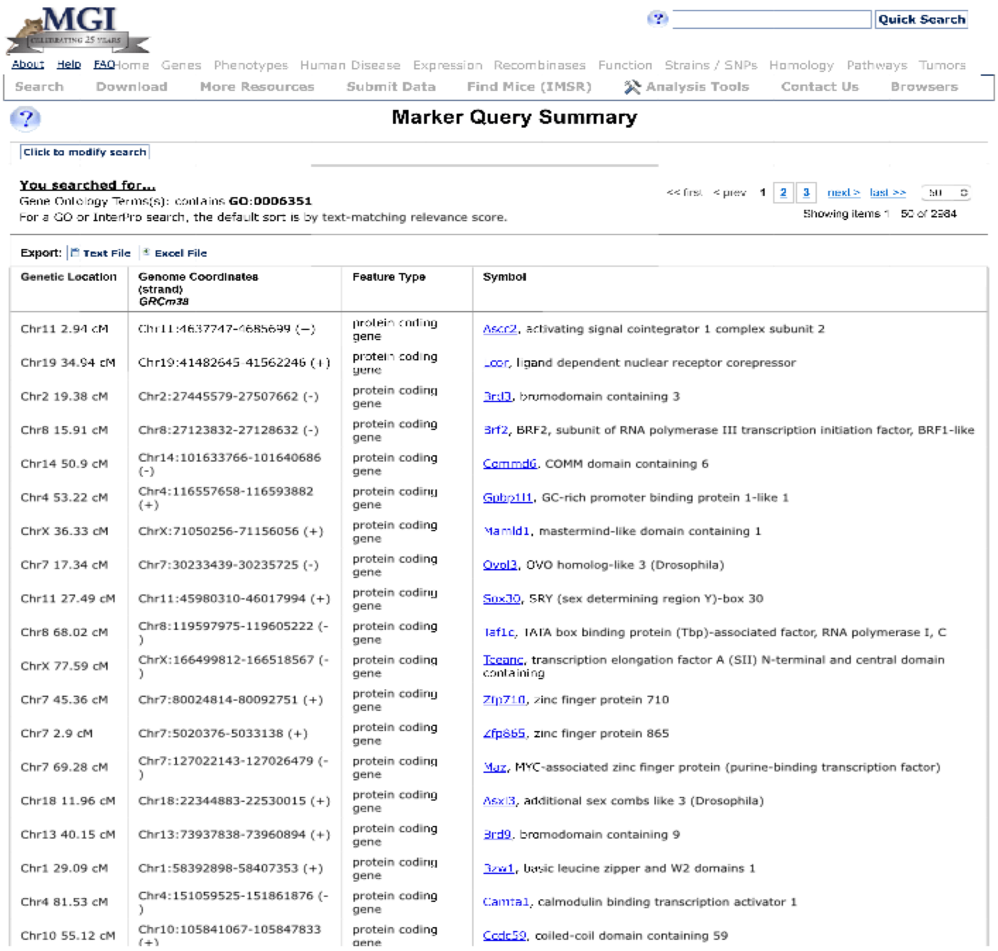
Measurement error is not the only source of noise for consideration. It is unlikely to be identical expression profiles for time series, which leads to the underlying differences in the expression of genes joint organization of genes regulated by the same transcription factor database (s).

5 Results and Discussion

In this section we show and explain the results of the third party that constrains about infer the Activity of Transcription Factors that consider the primary task in Our System, we use the result of clustering work with the results of this work to infer the activity of TF that binding infer with gene expression that discussed in the Second Part.



We build covariance functions to allow the inference of both Transcription factor protein concentrations and their effect on the transcription rates of each target gene from microarray data.



Genetic Location	Genome Coordinates (strand) *GRCm38*	Feature Type	Symbol
Chr11 2.94 cM	Chr11:4637747-4685699 (–)	protein coding gene	Ascc2, activating signal cointegrator 1 complex subunit 2
Chr19 34.94 cM	Chr19:41482645-41562246 (+)	protein coding gene	Lcor, ligand dependent nuclear receptor corepressor
Chr2 19.38 cM	Chr2:27445579-27507662 (-)	protein coding gene	Brd3, bromodomain containing 3
Chr8 15.91 cM	Chr8:27123832-27128632 (-)	protein coding gene	Brf2, BRF2, subunit of RNA polymerase III transcription initiation factor, BRF1-like
Chr14 50.9 cM	Chr14:101633766-101640686 (-)	protein coding gene	Commd6, COMM domain containing 6
Chr4 53.22 cM	Chr4:116557658-116593882 (+)	protein coding gene	Gpbp1l1, GC-rich promoter binding protein 1-like 1
ChrX 36.33 cM	ChrX:71050256-71156056 (+)	protein coding gene	Mamld1, mastermind-like domain containing 1
Chr7 17.34 cM	Chr7:30233439-30235725 (-)	protein coding gene	Ovol3, OVO homolog-like 3 (Drosophila)
Chr11 27.49 cM	Chr11:45980310-46017994 (+)	protein coding gene	Sox30, SRY (sex determining region Y)-box 30
Chr8 68.02 cM	Chr8:119597975-119605222 (-)	protein coding gene	Taf1c, TATA box binding protein (Tbp)-associated factor, RNA polymerase I, C
ChrX 77.59 cM	ChrX:166499812-166518567 (-)	protein coding gene	Tceanc, transcription elongation factor A (SII) N-terminal and central domain containing
Chr7 45.36 cM	Chr7:80024814-80092751 (+)	protein coding gene	Zfp710, zinc finger protein 710
Chr7 2.9 cM	Chr7:5020376-5033138 (+)	protein coding gene	Zfp865, zinc finger protein 865
Chr7 69.28 cM	Chr7:127022143-127026479 (-)	protein coding gene	Maz, MYC-associated zinc finger protein (purine-binding transcription factor)
Chr18 11.96 cM	Chr18:22344883-22530015 (+)	protein coding gene	Asxl3, additional sex combs like 3 (Drosophila)
Chr13 40.15 cM	Chr13:73937838-73960894 (+)	protein coding gene	Brd9, bromodomain containing 9
Chr1 29.09 cM	Chr1:58392898-58407353 (+)	protein coding gene	Bzw1, basic leucine zipper and W2 domains 1
Chr4 81.53 cM	Chr4:151059525-151861876 (-)	protein coding gene	Camta1, calmodulin binding transcription activator 1
Chr10 55.12 cM	Chr10:105841067-105847833 (+)	protein coding gene	Ccdc59, coiled-coil domain containing 59

Figure 5-1: shows the Transcription Factors Protein are douwnloaded from MGI[1].

Then we went to ENCODE[2] to knowing and downloading the all relationship between TF and gene expression, where the ENCODE considers important and simple Web tool to identify Enriched encode TF protein from a list of Genes or Transcriptions We entered the Genes Symbols in it and then select mouse Organism to analyze. The results after submit is shown in Figure 5-2. Where it contains example for some Transcription Factors Proteins and the names of genes of selected proteins.

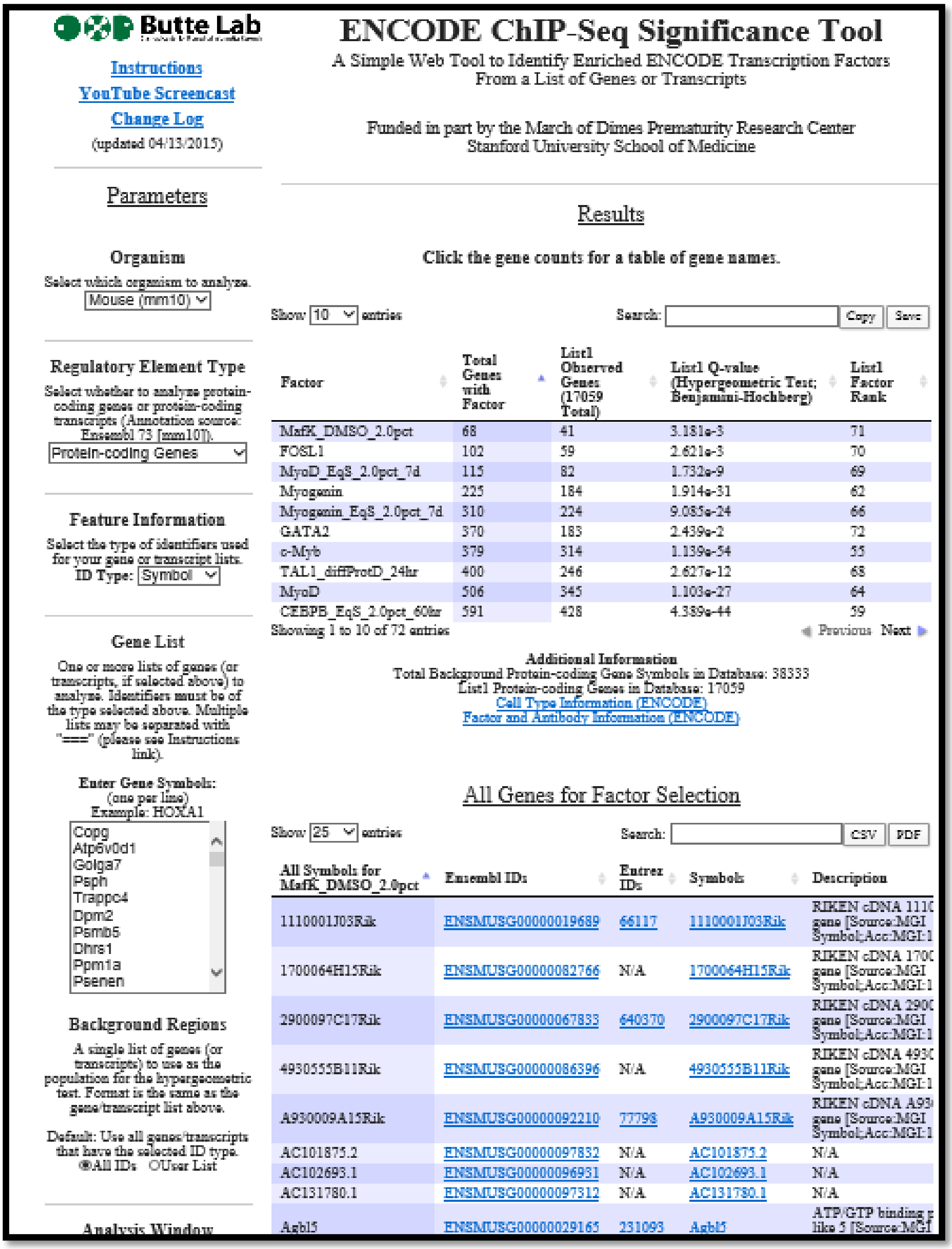


Figure 5-2: shows the relations between the Inputed genes with TF protein[2].

Then we made some codes to Compute (S) that called Connectivity Matrix has 1 if there are relationship between TF and Gene or 0 otherwise. As
Table 1.

Table 1: shows the Connectivity Matrix between TF Proteins and Genes, where 1 indicate there are binding else 0.

		0	1	2		67	68	69
		BHLHE40	c-Jun	c-Myb	…	ZKSCAN 1	ZNF	ZNF384
0	Atp6v0d 1	1	0	0	…	0	0	0
1	Golga7	0	0	0	…	0	1	0
2	Psph0	1	0	0	…	0	1	1
. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .
45035	Zmiz2	1	0	0	…	0	1	1
45036	Cltb	1	0	0	…	0	1	0
45037	D17Wsu 92e	1	0	0	…	0	0	0

5.1.1 Preprocessing Steps

5.1.1.1 Normalization step

After Computing Y Gene expression from Analysis Stage in (), W normalized these expressions using the Normalization equation as was mentioned in Eq 1 where Y Values become between -1 and 1

5.1.1.2 Checking the Zeros' Values

We check and removed rows from the dataset gene expression (Y) were not bound by any TF (S) and columns from Transcription Factors were not bound by any gene. The result of this step was with changing the dimension of Y before this step is (45038, 64) and dimension of S is (45038, 69), After applying that step the dimension of your and S is (26875, 64) and (26875, 69) respectively.

5.1.1.3 Results Ranking Step

We Ranked the Y gene Expression before applying the SVD method where it depended on the () method in 2013 and then Select top 1000 genes to model it as [Kalaitzis, 2013].

5.1.1.4 Result of SVD

The input of the SVD method has been just S and the result it is three Singular Matrixes R Lambda, V that have (1000, 1000) (69, 69) (69, 69) size respectively.

5.1.2 Prepare the data for processing in GP regression

We computed the Yq from Project data (Y) onto the principal subspace of S, (Q) that was computed from R as the size of Yq is (1000 *69, 1) and found sigma2 by looking at

the variance of Y_u from U that was computed also from R. The values of Yq, Yu and sigma2, the all Parameters and matrix are shown in Table 2.
Then we generated the matrix (X) of the Input associated with each Y, The TF and the Time point that has a size (1000*69, 3) as Table 3:

Table 2: Parameters of Clustering Work.

	Shape	**range**
Y	[1000,64]	[-1,1]
S	[1000,69]	0 or 1
R	(1000,1000)	[-.832,0.99]
V	(69,69)	[-.99, 0.957]
Q	(1000,69)	[-0.832, 0.538]
U	(1000,931)	[-0.244, 0.994]
Yq	(4416, 1)	[-1, 1]
Yu	(4416, 1)	[-0.707, 0.528]
Sigma2	(1,)	0.0066

Table 3: The X matrix

	Time Points	Replicates	Transcription Factors Protien
0	**30**	**0**	**0**
1	**30**	**0**	**0**
2	**30**	**0**	**0**
3	**30**	**0**	**0**
4	**60**	**0**	**0**
5	**60**	**0**	**0**
. .	. .	. .	. .
4412	**120**	**3**	**68**
4413	**120**	**3**	**68**
4414	**120**	**3**	**68**
4415	**120**	**3**	**68**

5.1.3 Applying the GP regression

We used the RBF covariance function as kernel and Gaussian Likelihood. The likelihood can be estimated efficiently using the sparsity of the covariance and recursion relations.

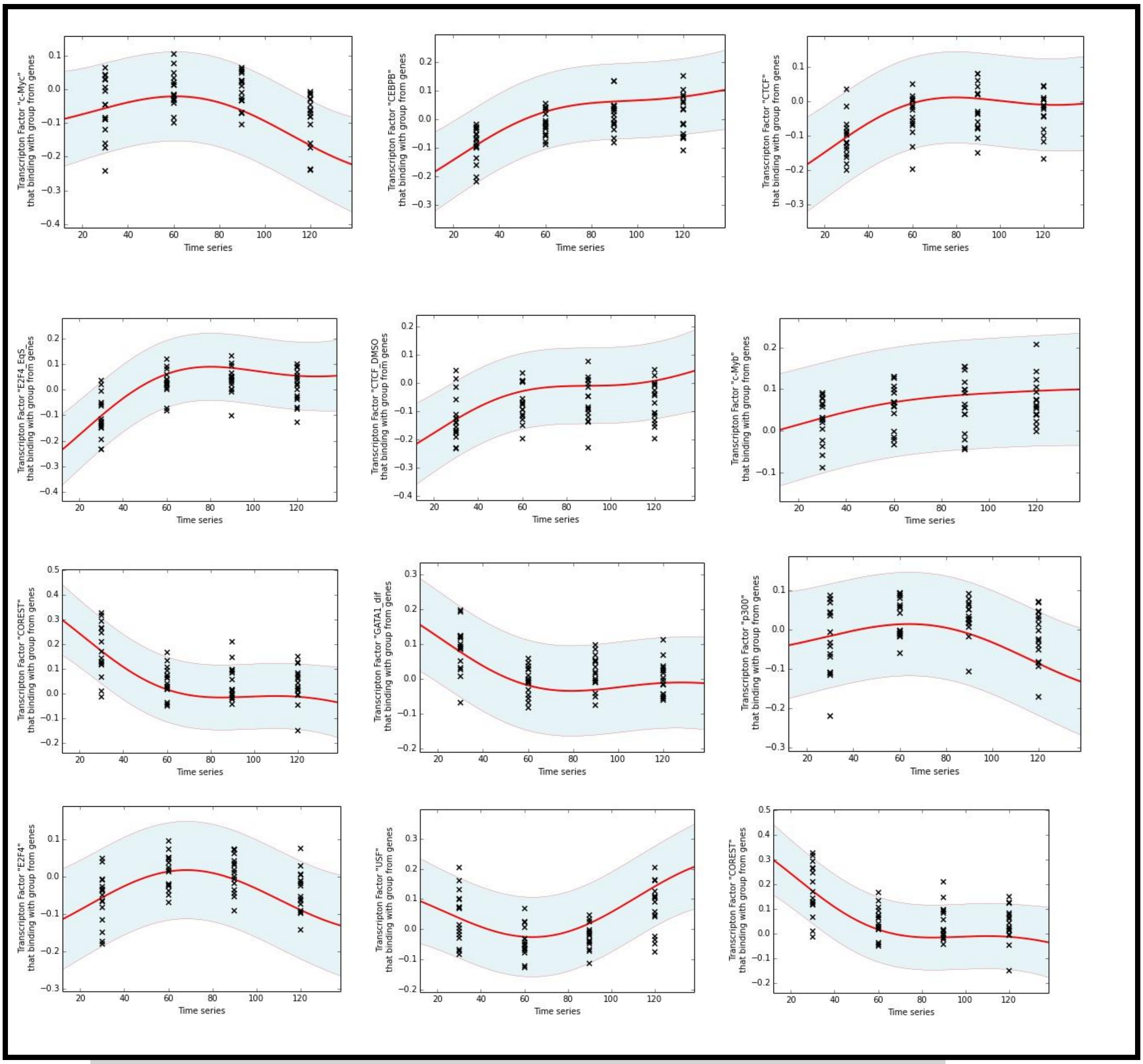


Figure 3: shows the activity of each TF protein for binding set of genes.

We note from Figure 3 the activity (p300)TF protein for example that is binding with set of genes alters its behavior. Here we inferred the gene-specific transcription activities for Mice models and we can determine which regulations significant for a given experimental condition for two mutations for two strains. We checked the consistency of our model on the mice model and used a connectivity matrix obtained via the relationship between TF and genes that obtained from Encode Chip-Seq significance Tool, this data consists of the expression profiles of 45038 genes measured at 4 equally spaced time points (4 stages to progress the ALS) and in each time it contains two strains in each strain contains two mutations and with its role contains four replicates and then integrate it with 69 transcription factors .

6 Conclusion

1- Our proposed work explained the effectiveness of sharing information between different model conditions and replicates when modelling gene expression time series.
2- We suggested a new model depended on to infer Transcription Factor Activities and correlated with genes that previously selected.
3- We suggested accurate methods to recognize what are the genes that is causing a disease and what is its relationship with Transcription Factors using many biology sources to prove that these genes really related with ALS Disease.
4- Analysis of gene pathway of a few specified clusters for a particular group may lead toward identifying features underlying the differential speed of progression of disease.

Acknowledgments

We would like to thanks Prof. Neil Lawrence, Dr. Paul Heath and SiTraN, Sheffield University to theirs research visiting invitation, time, and supervision for my research visiting. I would like to thank The Iraqi ministry of higher Education and Scientific research, Babylon University and Computer Science Department for funding my visiting research at SiTraN, Sheffield University, UK.

7 References

Alice Brockington, Ke Ning, Paul R. Heath, m Elizabeth Wood, Neil Lawrence, et. Al, 2013,"Unravelling the enigma of selective vulnerability in neurodegeneration: motor neurons resistant to degeneration in ALS show distinct gene expression characteristics and decreased susceptibility to excitotoxi- city", Acta Neuropathol (2013) 125:95109

Alvarez, Mauricio a., Lorenzo Rosasco, and Neil D. Lawrence. 2011. "Kernels for Vector-Valued Functions: A Review." 1–37.

Anne-Laure Boulesteix and Korbinian Strimmer, 2005,"Predicting transcription factor activities from combined analysis of microarray and ChIP data: a partial least squares approach", BioMed Central.

Bohling, Geoff. 2005. "Kriging." Kansas Geological Survey (October): 1–20.

Emery, Xavier and María Peláez. 2012. "Reducing the Number of Orthogonal Factors in Linear Coregionalization Modelling." Computers and Geosciences 46:149–56.

Finazzi, Francesco, E. Marian Scott, and Alessandro Fassò. 2011. "The Dynamic Coregionalization Model in Air Quality Risk Assessment." (August): 4537–42.

Gao, Pei, Antti Honkela, Magnus Rattray, and Neil D. Lawrence. 2008. "Gaussian Process Modelling of Latent Chemical Species: Applications to Inferring Transcription Factor Activities." Bioinformatics 24 (16): 70–75.

Goovaerts, P. 1992. "Factorial Kriging Analysis: A Useful Tool for Exploring the Structure of Mulitvariate Spatial Information." Journal of Soil Science 43 (4): 597–619.

Goulard, M. and M. Voltz. 1992. "Linear Coregionalization Model: Tools for Estimation and Choice of Cross-Variogram Matrix." Mathematical Geology 24 (3): 269–86.

H. M. Shahzad Asif, Matthew D. Rolfe, Jeff Green2, Neil D. Lawrence, Magnus Rattray and Guido Sanguinetti, 2010,"TFInfer: a tool for probabilistic inference of transcription factor activities, Vol. 26, pages 2635-2636, doi: 10.1093/Bioinformatics/btq469.

Han, Jiawei and Micheline Kamber. 2006,"Data Mining: Concepts and Techniques".

Hashimoto, Tatsunori B. 2014. "Computation Identification of Transcription Factor Binding Using DNase-Seq".

Julia Morton Caponiti, 2012,"Gene Expression in Motor Neurons with Differential Susceptibility to Amyotrophic Lateral Sclerosis (ALS)", PhD thesis.

Kalaitzis Alfredo, 2013. "Learning with Structured Covariance Matrices in Linear Gaussian Models." (February

López-Kleine, Liliana, Luis Leal, and Camilo López. 2013. "Biostatistical Approaches for the Reconstruction of Gene Co-Expression Networks Based on Transcriptomic Data." Briefings in Functional Genomics 12 (5): 457–67.

Mark Maienschein-Cline, Jie Zhou, Kevin P. White, Roger Sciammas and Aaron R. Dinner, 2012,"Discovering transcription factor regulatory targets using gene expression and binding data", Bioinformatics, Vol. 28 no. 2 2.

Meng, Jia, Jianqiu (Michelle) Zhang, Yidong Chen, and Yufei Huang. 2011. "Bayesian Nonnegative Factor Analysis for Reconstructing Transcription Factor Mediated Regulatory Networks." Proteome Science 9 (Suppl 1): S9.

K Titsias, Antti Honkela, Neil D Lawrence and Magnus Rattray, 2012,"Identifying targets of transcription factors from expression time series by Bayesian model comparison".

Nardo G1, Iennaco R, Fusi N, Heath PR, Marino M, Trolese MC, Ferraiuolo L, Lawrence N, Shaw PJ, Bendotti C., 2013,"Indices of fast and slow disease progression in two mouse models of amyotrophic lateral sclerosis", Brain.

Richard D Pearson, Xuejun Liu, Guido Sanguinetti, Marta Milo, Neil D Lawrence and Magnus Rattray, 2009,"puma: a Bioconductor package for propagating uncertainty in microarray analysis", BMC Bioinformatics, 10:211.

Sanguinetti, Guido, Magnus Rattray, and Neil D. Lawrence. 2006a. "A Probabilistic Dynamical Model for Quantitative Inference of the Regulatory Mechanism of Transcription." Bioinformatics 22 (14): 1753–59.

Sanguinetti, Guido, Marta Milo, Magnus Rattray, and Neil D. Lawrence. 2005. "Accounting for Probe-Level Noise in Principal Component Analysis of Microarray Data." Bioinformatics 21 (19): 3748–54.

Sanguinetti, Guido, Neil D. Lawrence, and Magnus Rattray. 2006b. "Probabilistic Inference of Transcription Factor Concentrations and Gene-Specific Regulatory Activities." Bioinformatics 22 (22): 2775–81.

Strippoli, Pierluigi *et al.* 2005. "Predictting Transcription Factor Activities from Combined Analysis of Microarray and CHIP Data: A Partial Least Squares Approach." Theoretical

Van Barel, Marc, Yvette Vanberghen, and Paul Van Dooren. 2010. "Using Semiseparable Matrices to Compute the SVD of a General Matrix Product/quotient." Journal of Computational and Applied Mathematics 234 (11): 3175–80.

[1] http://www.informatics.jax.org/

[2] http://encodeqt.simple-encode.org/