Statistical Analysis of a Case-Control Alzheimer's Disease: A Retropective Approach with Sufficient Dimension Reduction

REU Site: Interdisciplinary Program in High Performance Computing

Trevor V. Adriaanse[1], Meshach Hopkins[2], Rebecca Rachan[3], Subodh R. Selukar[4],
Graduate assistant: Elias Al-Najjar[5]
Faculty mentor: Kofi P. Adragni[5], Client: Nusrat Jahan[6]

[1]Department of Mathematics, Bucknell University,
[2]Department of Computer Science and Electrical Engineering, UMBC,
[3]Department of Mathematics, North Central College,
[4]Departments of Biostatistics and Biology, University of North Carolina, Chapel Hill,
[5]Department of Mathematics and Statistics, UMBC,
[6]James Madison University

Technical Report HPCF–2015–23, hpcf.umbc.edu > Publications

Abstract

Alzheimer's Disease is a neurological disorder chiefly present in the elderly that affects functions of the brain such as memory and logic, eventually resulting in death. There is no known cure to Alzheimer's and evidence points to the possibility of a genetic link. This study analyzes microarray data from patients with Alzheimer's disease and disease-free patients in order to evaluate and determine differential gene expression patterns between the two groups. The statistical problem stemming from this data involves many predictor variables with a small sample size, preventing the use of classical statistical approaches from being effective. We turn to a novel three-step approach: first, we screen the genes in order to keep only the genes marginally related to the outcome (presence of Alzheimer's); second, we implemented a sparse sufficient dimension reduction to retain only predictors relevant to the outcome; lastly, we perform a hierarchical clustering method to group genes that exhibit mutual dependence. We adapted this methodology from Adragni et. al and expand on their work by optimizing the existing R code with parallel capabilities in order to enhance performance speed. Thus, our results reflect both an analysis of the microarray data and a performance study of the modified code.

1 Introduction

Alzheimer's Disease (AD) is the most common form of dementia that affects memory, thought, and behavior. It is an irreversible, progressive brain disorder that slowly destroys

memory and thinking skills, and eventually the ability to carry out the simplest tasks. Physiologically, AD progression occurs when irregular protein structures called plaques and tangles destroy brain cells. These protein fragments originate in the hippocampus, where the brain forms memories, and their migration throughout the rest of brain leads to the destruction of other neurological capacities such as the ability to form logical thoughts, to control behaviors, to speak, and to move [1].

There is no known cure to AD, and scientific research is actively underway for a cure. It is suspected that there is a genetic link present in AD. Genetic association studies are one promising area of research to help identify candidate genes or genome regions that contribute to the disease by testing for a correlation between the disease status and the genetic variation.

Microarray data provides relative gene expressions of many genes. With this set of genes, we aim to determine which genes are differentially expressed between patients of Alzheimer's and individuals who do not have the disease. Previous statistical analysis methods of gene expressions data include logistic regression, Bayesian regression, and principal component regression among others. Logistic regression may be the most obvious tool for predicting a binary outcome, and it has been adapted to the microarray problem of many predictors with small sample size. One such adaptation noted by [6] is penalized logistic regression, which addresses multicollinearity and over-fitting. The paper notes that this method may require additional modifications like bootstrapping to provide better prediction. Segal et al. [10] describes a Bayesian regression method using singular value decomposition used to analyze microarray data. This makes use of all of the genes instead of regressing on a subset of the total genes. This would likely be computationally intensive and would have the problem that variation of the regression factors would not necessarily be explained by the phenotypic variation. Principal component regression is another popular technique that has been used in the literature to obtain the so called eigen-genes, a set of genes that are assumed to be important [9].

Regression methods that are considered to analyze gene expression data are often typical for prospective analysis. However, gene expressions data are obtained in a retrospective setup. It is accepted that for a case-control study, both prospective and retrospective analyses yield the same result under a logistic regression setup [11].

In the present work, we consider the statistical analysis using a retrospective approach. Under that setup, we consider the information provided by the genes, given the disease status. This is often referred to as inverse regression. The methodology is in accord with the sampling scheme of the gene expression data and has a built-in sparse sufficient dimension reduction procedure to help identify the most important genes.

The new methodology features principal fitted components, an inverse regression method for sufficient dimension reduction [4]. PFC differs from the well-known principal components

analysis as in PC, the outcome of interest is not used in obtaining the relevant components while PFC does involve the outcome and thus produces a more efficient reduction than a typical PC.

Similar to PC, PFC produces a set of linear combinations of the initial predictors or genes that best explain the response. As we expect a small subset of these genes to be relevant in explaining the outcome, a large number of these genes will be inactive. Our methodology involves an initial screening of the genes to retain those that are marginally related to the outcome. Out of the selected set, a sparse sufficient reduction is obtained to yield a linear combination of the most important sets of genes. The methodology is based on the p-value guided hard-thresholding for sparse sufficient dimension of [2]. It also relates the variation from the regression model with the phenotypic variation and includes a clustering that groups gene expressions based on their interdependence; that is, genes that are dependent occur in the same cluster, while genes that are independent of each other appear in different clusters. This clustering emphasizes the dependence structure among the genes, which may help in better predicting the status of AD. The structure of the grouping of genes is suspected but not known. Therefore, the group-wise sufficient dimension reduction considered by [3] is used to obtain a sparse estimation of a sufficient dimension reduction to better predict the response.

We obtained microarray data that serves as our set of gene expressions. The microarray data provided gives the expression state of many genes, and we aim to find a relationship between a subset of those genes with AD [7].

The initial R codes for the p-value guided hard-thresholding for PFC [2] and for the group-wise sufficient dimension [3] were provided by Dr. Adragni. We merged the two code sources appropriately for the analysis of the AD data. As the dimensions of the data are large, the implemented code tends to be sluggish. We implemented a parallel version of the code on the High Performance Computing hardware to speed up its performance.

The remainder of this report is organized as follows. In Section 2 we introduce dimension reduction, our implementation of the principal component model, and the hierarchical clustering used. Section 3 describes the genes we have found to be related to Alzheimer's. In Section 4 we describe the effectiveness of the parallelization of the existing R code. Section 5 concludes our paper by defending the relationships we determined between Alzheimer's and certain gene expressions.

2 Methodology

A proper regression analysis takes into account the sampling scheme that governed the data acquisition. Let Y denote the outcome of interest, and X be a p-vector predictor. The data

may be obtained in a forward or prospective manner as $Y|X$, in an inverse or retrospective manner as $X|Y$, or jointly as (Y, X). In a prospective study, the outcome Y is observed assuming that the predictors are fixed, not random. In a retrospective study, the covariates are observed given the observed response.

Genetic data are obtained primarily in a retrospective manner. That is, subjects are selected based on the disease status or phenotype. Given the phenotype of the subjects, the gene expressions are observed. A natural statistical study of such data should be via an inverse regression. It is known that when the outcome is a case-control, both retrospective and prospective analyzes yield the same conclusion in terms of the propensity of the disease or phenotype. However, our study shows that more information could be gathered when the appropriate inverse regression method is applied.

We organize the microarray data in a vector of p predictors $X = (X_1, \ldots, X_p)^T$ with a corresponding response variable Y. The response is binary, where $Y = 0$ when Alzheimer's is absent and $Y = 1$ if present. Obviously, a typical microarray data set has a large value for p. With a large p, we hope to reduce the dimensionality of X without losing any regression information of Y contained in X. Reducing the dimensionality of X allows for a better modeling and prediction of future observations, as well as to create a more viewable data. The process of replacing X with a lower dimensional function $R(X)$ is called dimension reduction. When $R(X)$ is obtained to satisfy one of the conditions (i) $X|Y, R(X) \sim X|R(X)$, (ii) $Y|X, R(X) \sim Y|R(X)$, or (iii) $Y \perp\!\!\!\perp X|R(X)$, it is called a sufficient dimension reduction of X [4]. Consider the regression information contained in the genes given the disease, $X|Y$. It can be partitioned into a part that depends on Y, and the other part that is independent of Y. We will write

$$X|Y = \boldsymbol{\nu}(Y) + \boldsymbol{\nu}(Y)^{\perp}$$

We assume that $\boldsymbol{\nu}(Y)$ is a linear function of Y as $\boldsymbol{\nu}(Y) = \Phi Y$, where $\Phi \in \mathbb{R}^p$. Moreover, we assume that $\boldsymbol{\nu}(Y)^{\perp} = \mu + \Delta^{1/2}\boldsymbol{\epsilon}$ where $\mu = E(X)$, $\Delta \in \mathbb{R}^{p\times p}$, and $\boldsymbol{\epsilon}$ is p-dimensional standard normal. Row i of $\Phi = (\phi_1, \cdots, \phi_p)^T$ determines the importance of the associated gene X_i. That is, $\phi_i = 0$ implies X_i is not expressed for the disease, but $\phi_i \neq 0$ is a plausibility of association of the gene to the disease. Writing $\Phi = \Gamma\lambda$ where $\lambda = \|\Phi\|$ and $\Gamma = \Phi/\|\Phi\|$ yields the model

$$X_y = \mu + \Gamma\lambda y + \Delta^{1/2}\boldsymbol{\varepsilon} \tag{2.1}$$

The term Γ is semi-orthogonal, and $\lambda \in \mathbb{R}$. This model is a special case of Cook's principal fitted components (PFC) models [5]. The most important parameters in model (2.1) are Γ and Δ. It is important to understand what information they provide in terms of the Alzheimer's data. Let $\eta = \Delta^{-1}\Gamma$. Under this model, $\eta^T X \in \mathbb{R}$ is a sufficient reduction of X. This reduction is a linear combination of the p predictors. The magnitude of a row of η provide information about the relevance of the corresponding gene. When there is no

relationship between the gene and the disease, the row is close to zero. Thus, when there is a relationship between the gene and the disease, the row is statistically different from zero. Furthermore, the dimension p of the genes is large, and only a small subset of the genes are relevant. Hence, a sparse estimation of Γ is needed to set any entry to zero if the gene is not related to Alzheimer's. We adapt the procedure of [2] to obtain the sparse estimate of Γ.

The covariance Δ provides information about the dependence of the genes after removing the disease information. We assume that there is a grouping structure of the genes, given the disease. Genes within a group are correlated, while the groups are independent. However, the grouping structure is unknown, but identifying this dependence would help better predict the disease status. To discover this grouping structure, we adopt the group-wise PFC procedure of [3]. The details follow.

2.1 Group-wise Sparse PFC

The sparse estimation of Γ is based on the so-called "p-value guided hard-thresolding" of [2]. By ignoring the structure of Δ, model (2.1) can be expressed as p independent linear regressions

$$X_i = \mu_i + \phi_i y + \delta_i \varepsilon_i, \quad i = 1, \ldots, p \tag{2.2}$$

This model is a univariate regression model for each gene $X_i, i = 1, \cdots, p$. Gene X_i is relevant in explaining Alzheimer's if $\phi_i \neq 0$. Now a simple t-test can be carried out for the hypotheses

$$H_{0i} : \phi_i = 0 \text{ against } H_{ai} : \phi_i \neq 0 \tag{2.3}$$

The resulting p-values determine whether H_{0i} is rejected. Let α be the significance level, $\pi = (\pi_1, \cdots, \pi_p)^T$ be the vector of p-values obtained from testing hypotheses (2.3), and let $\mathbf{1}_p$ be the p-vector one ones. A crude sparse estimate of Γ is obtained as

$$\widehat{\Gamma}_\alpha = J(\pi \leq \alpha \mathbf{1}_p) \circ \widehat{\Gamma} \tag{2.4}$$

where $\widehat{\Gamma}$ is an estimator of Γ. The inequality is element-wise, and J represents the indicator function. The operator $\circ$ stands for the Hadamard product of matrices. This screening method is particularly useful because we can immediately remove predictors with corresponding p-values greater than α, thus eliminating some of the irrelevant genes.

After the screening, we compute the F statistic for each of the predictors remaining. Then, a grid of m significance levels between F_{min} and F_{max} is obtained. For each level of significance, a prediction error is computed by cross-validation. The significance level F_j that has the smallest prediction error is retained to be used to obtain the final sparse estimator $\widehat{\Gamma}_{\alpha_k}$.

PFC group-wise sufficient dimension reduction is then used on the set of q genes that are obtained from the sparse estimation. The methodology involves partitioning the genes into sets where genes in the same set are dependent while genes between sets are independent of each other. We determine the grouping structure of the predictors with a hierarchical agglomerative clustering. In order to cluster the genes, it is necessary to evaluate the closeness of the predictors. Given Δ, the correlation matrix R is obtained by $R = D^{-1/2}\Delta D^{-1/2}$ where $D \in \mathbb{R}^{q\times q}$ is the matrix of diagonal elements of Δ. Then, the distance between any two predictors, X_i, X_j, is determined by $d_{ij} = 1 - |R_{ij}|$ where $R_{ij} = \rho(X_i, X_j|Y)$ is the correlation coefficient of the two predictors given Y. In complete linkage clustering two clusters C_k and C_l are merged using the following metric

$$d(C_k, C_l) = \max_{X_i\in C_k, X_j\in C_l} d_{ij}.$$

Finally, since the clustering is a bottom-up approach, it starts with all predictors being their own singleton cluster. From here, the two closest clusters are merged; this process is repeated until all genes form a single cluster. Notice, that for each new clustering, a PFC model can be fitted with a structured Δ induced by the conditional independence of clusters. Assuming that the true structure of Δ is identified by one of the sets of clusters of predictors, we find the structure by comparing the q models with respect to a prediction performance.

3 Results

To proceed with our analysis, we used microarray data that was processed from postmortem human brain tissues donated for the Hisayama study. The RNA samples were prepared front gray matter of the frontal cortex, temporal cortex, and hippocamus. Eighty eight samples were taken, and among these samples, 26 cases were diagnosed with AD or an AD-like disorder, while the remaining 62 samples were non AD. These high-quality RNA samples were run through microarray analysis using the Affymetric Human Gene 1.0 ST platform. Only the samples that passed this analysis were used, to ensure quality control. In total, there were 79 samples that were used, with 32,312 genes analyzed. Of these 79 samples: 33 were from the frontal cortex, 15 AD; 29 taken from temporal cortex, 10 AD; and 17 taken from the hippocampus, 7 AD. In our results we grouped samples by AD (32 samples) and non-AD (47 samples).

The first process of the data analysis was a t-test based screening procedure. This gave us 5260 genes that significantly predicted AD. Of these 5260, we chose the 200 with the smallest p-values to use in our sparse estimation. The sparse estimation then further reduced these 200 to a final selection of 49 genes that most significantly predict Alzheimer's Disease. We see in Figure 3.1 that prediction error is at a minimum at 49 predictors, confirming our

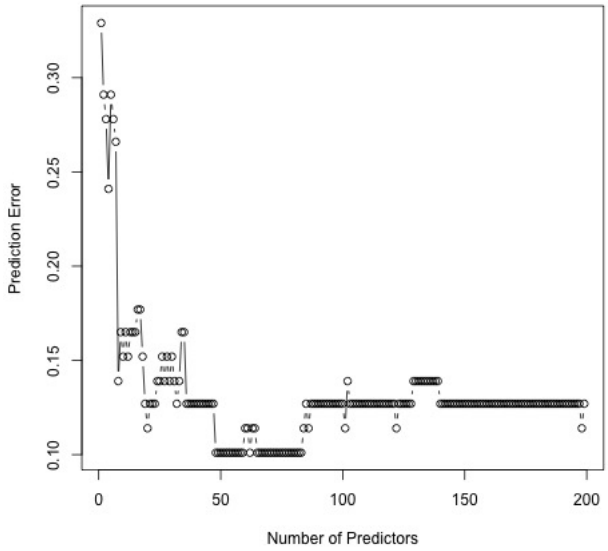

Figure 3.1: Prediction error is minimized when using 49 predictors

sparse estimation method. After determining the final selection of genes, we performed agglomerative hierarchical clustering to find the dependence structure of these genes. From Figure 3.2, the minimum prediction error occurs at a set of 3 clusters. This set refers to a grouping structure with the first 22 (of the 49) in the first group, the next 22 in the second group and the final 5 in a third group. The gene groupings can be seen in Table 3.1.

Table 3.1: Grouping of Genes

Group 1	8028380 8062844 8096663 7937275 7985757 8030448 8075637 8135172 8178561 8050060 7950284 8098576 8170891 7960689 7963235 8062880 8081620 8176230 7903507 7982564 7974895 8051773
Group 2	8039378 7905817 8028791 8002041 8037079 8015835 7992447 8036252 8180371 7902435 8041225 8123739 8014794 7899841 7931479 8062796 8091452 7908867 7979663 8026155 7981566 7997352
Group 3	7894596 7893808 7894185 8024436 8121130

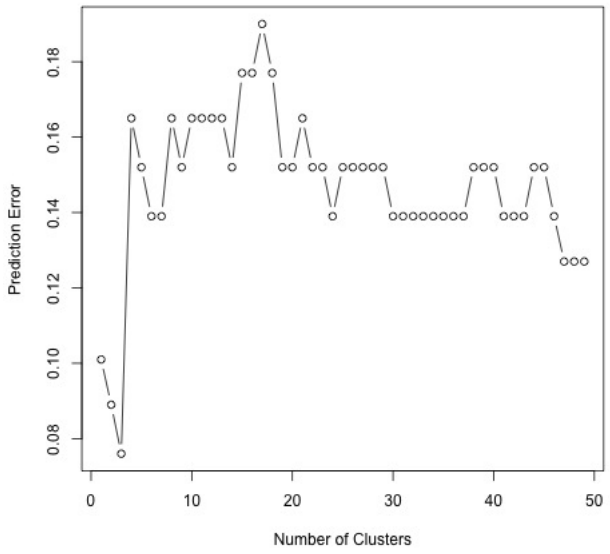

Figure 3.2: Prediction error is minimized when using 3 clusters

4 Computational Performance on HPFC

This section discusses the timing and speedup of our simulations on maya 2013. Maya 2013 is made up of 72 nodes, 67 of which are compute nodes, two are develop nodes, one is a user node, and the last is a management node. Each compute node has room for 64GB of memory [8]. For the sake of this study, we worked on up to 16 compute nodes and up to 16 processes per node, making it possible to calculate up to 256 processes.

To parallelize our code, we used a SLURM submission script that uses `srun` to begin. In the SLURM script, the number of nodes used is determined via `--nodes` and the number of

Table 4.1: Timings and observed speedup for the parallel implementation of the code by the number of processes used with 16 processes per node, with the exception of $p = 1$ which uses 1 process per node.

	$p = 1$	$p = 16$	$p = 32$	$p = 64$	$p = 128$	$p = 256$
Average Time	5.39	4.42	4.59	4.58	4.57	4.53
Observed speedup	1.00	1.22	1.17	1.18	1.18	1.19

tasks per node is specified with `--ntasks-per-node`.

In order to compare the timings of our serial and parallelized code, we conducted 100 simulations of the data set to time our parallelized function named `spfc_pvg`. For each simulated data set, we used various levels of parallelization to time the function: one node with one process, and one, two, four, eight, and sixteen nodes, all with sixteen processes.

The simulated dataset for each simulation run X was generated by the following process using $p = 20000$ predictors and $n = 200$ observations. First a response vector, Y, was obtained from a binomial distribution with $n = 200$ observations, 1 trial per observation and probability of success 0.3. Next, parameter Γ was assigned as a vector of length $p = 20000$ with the first $k = 20$ entries as either 1 (rows 1 to 10) or -1 (rows 11 to 20) and the other entries being 0. The noise parameter ϵ was a matrix of size 200 x 20000 with entries drawn from a normal distribution with mean $\mu = 0$ and variance $\sigma^2 = 0.5$. The dataset X was then computed by the following formula:

$$X = Y\Gamma^T + \epsilon$$

Table 4.1 displays the results of our performance study. The first row shows the average observed wall clock time for each of the processes in minutes. This documents the amount of time it took, in total, for the function to run. The second row displays the speedup, calculated by $S_p = \frac{T_1}{T_p}$. Comparing adjacent columns in Table 3.1 shows that speed-up only occurred between serial and our first parallel implementation $p = 16$. For greater values of p, we did not see consistent speedup as we increased the processes. These results are visualized in Figures 4.1 and 4.2. Overall, we can conclude that parallelization as a whole did decrease the time it took our code to run.

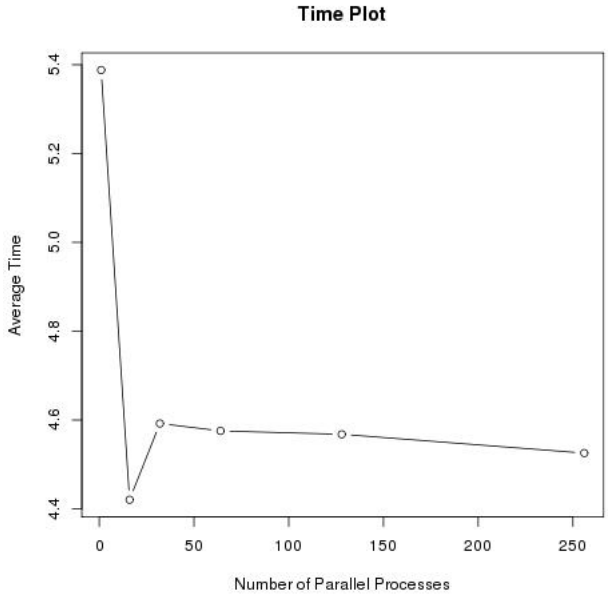

Figure 4.1: Parallel execution is faster than serial implementation

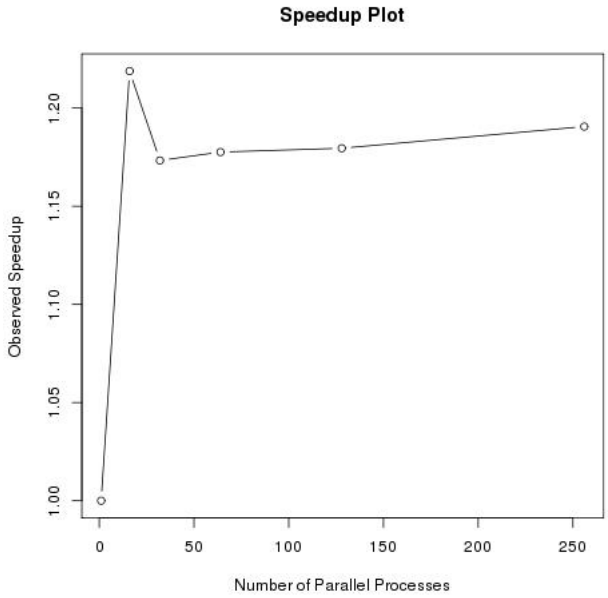

Figure 4.2: Parallelization allows for speedup of `spfc_pvg`

5 Conclusions

The results of our time study show that the use of parallel computing techniques decreases the runtime of our program. However, this decrease is almost negligible. Running our program on 1 node with 16 processes per node produced the fastest runtime while running the program serially on 1 node with 1 process per node produced the slowest time. Yet, the 1 node 16 process per node runtime was only faster than the serial run by a factor of 1.25. With 2 nodes and 16 processes per node the program was faster than the serial run but the slowest of all parallel runs. From 2 nodes to 16 nodes, the program's runtime increases in a linear fashion. There are a few possible explanations for the phenomena we saw. Either the size of the data set or the extra processes it takes to make the program parallel and split the data up into separate nodes, running our program on more than one node does not further decrease the runtime.

Acknowledgments

These results were obtained as part of the REU Site: Interdisciplinary Program in High Performance Computing (hpcreu.umbc.edu) in the Department of Mathematics and Statistics at the University of Maryland, Baltimore County (UMBC) in Summer 2015. This program is funded by the National Science Foundation (NSF), the National Security Agency (NSA), and the Department of Defense (DOD), with additional support from UMBC, the Department of Mathematics and Statistics, the Center for Interdisciplinary Research and Consulting (CIRC), and the UMBC High Performance Computing Facility (HPCF). HPCF is supported by the U.S. National Science Foundation through the MRI program (grant nos. CNS–0821258 and CNS–1228778) and the SCREMS program (grant no. DMS–0821311), with additional substantial support from UMBC. Co-author Meshach Hopkins was supported, in part, by the UMBC National Security Agency (NSA) Scholars Program through a contract with the NSA. Graduate assistant Elias Al-Najjar was supported during Summer 2015 by UMBC.

References

[1] Alzheimer's disease and the brain. https://www.alz.org/braintour/plaques_tangles.asp, 2011. Accessed 2015-07-28.

[2] Kofi P. Adragni and Mingyu Xi. Pruning a sufficient dimension reduction with a p-value guided hard-thresholding. *Statistics*, pages 1–17, 2015.

[3] KP Adragni, E Al-Najjar, S Martin, SK Popuri, and AM Raim. Groupwise sufficient dimension reduction with principal fitted components. *Computational Statistics*, 2015.

[4] R. Dennis Cook. Fisher lecture: Dimension reduction in regression. *Statistical Science*, 22:1–26, 2007.

[5] R. Dennis Cook and Liliana Forzani. Principal fitted components for dimension reduction in regression. *Statistical Science*, 23:485–501, 2008.

[6] Paul H. C. Eilers, Judith M. Boer, Gert-Jan van Ommen, and Hans C. van Houwelingen. Classification of microarray data with penalized logistic regression, 2001.

[7] Leon J Ninomiya T et al. Hokama M, Oka S. Altered expression of diabetes-related genes in alzheimers disease brains: the hisayama study. 2014 Sep;24(9):2476-88. PMID: 23595620.

[8] Samuel Khuvis and Matthias K. Gobbert. Parallel performance studies for an elliptic test problem on the cluster maya. Technical Report HPCF20156, UMBC High Performance Computing Facility, University of Maryland, Baltimore County, 2015.

[9] Huan Liu and Hiroshi Motoda. *Computational Methods of Feature Selection*. Chapman and Hall/CRC, Boca Raton, Florida, 2007.

[10] Eran Segal, Michael Shapira, Aviv Regev, Dana Pe'er, David Botstein, Daphne Koller, and Nir Friedman. Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data. *Nature Genetics*, 34:166–176, 2003.

[11] Ana-Maria Staicu. On the equivalence of prospective and retrospective likelihood methods in case-control studies. *Biometrika*, 4:990–996, 2010.