MicroRNA Target Identification by Reverse Phase Protein Array

A Dissertation Presented

by

Jiawen Zhu

to

The Graduate School

in Partial Fulfillment of the

Requirements

for the Degree of

Doctor of Philosophy

in

Applied Mathematics and Statistics

Stony Brook University

January 2015

Stony Brook University

The Graduate School

Jiawen Zhu

We, the dissertation committee for the above candidate for the

Doctor of Philosophy degree, hereby recommend

acceptance of this dissertation.

Jie Yang – Dissertation Advisor
Assistant Professor, Department of Preventive Medicine, Director, Statistical Consulting Core, School of Medicine, Adjunct Assistant Professor, Department of Applied Mathematics and Statistics

Song Wu – Dissertation Co-advisor
Assistant Professor, Department of Applied Mathematics and Statistics

Wei Zhu – Chairperson of Defense
Professor, Deputy Chair, Department of Applied Mathematics and Statistics

Jingfang Ju – Outside Member
Associate Professor, Co-director of Translational Research Laboratory, Department of Pathology, School of Medicine

This dissertation is accepted by the Graduate School

Charles Taber
Dean of the Graduate School

Abstract of the Dissertation

MicroRNA Target Identification by Reverse Phase Protein Array

by

Jiawen Zhu

Doctor of Philosophy

in

Applied Mathematics and Statistics

Stony Brook University

2015

Understanding functions of microRNAs (or miRNAs), particularly their effects on protein degradation, is biologically important. Emerging technologies, including the reverse-phase protein array (RPPA) for quantifying protein concentration and RNA-seq for quantifying miRNA expression, provide a unique opportunity to study miRNA-protein regulatory mechanisms. A naïve and commonly used way to analyze such data is to directly examine the correlation between the raw miRNA measurements and protein concentrations estimated from RPPA through simple linear regression models. However, the uncertainty associated with protein concentration estimates is ignored, which may lead to less accurate results and significant power loss.

Here we propose an integrated nonlinear hierarchical model for detecting miRNA targets through original RPPA intensity data. The model is fitted within a maximum likelihood framework and the significance of the correlation between miRNA and protein is assessed using the Wald test. Our extensive simulation studies demonstrated that the integrated method performed consistently better than the simple method, especially when the RPPA intensity levels are close to the boundaries of image intensity limits. The proposed model was also illustrated through real datasets from The Cancer Genome Atlas (TCGA) program.

In addition, we extend the model to a semi-parameter model by incorporating a nonparametric curve fitting technique, which relaxes the assumption of a specific parametric form for the RPPA response curve. The performance of this model is also demonstrated by simulation studies and real data analyses.

I dedicate this dissertation to my father, Wei Zhu and mother, Jianqin Hu

Table of Contents

List of Figures

List of Tables

Acknowledgments

I would like to thank my dissertation committee members. Without their guidance, I would never have been able to finish the dissertation.

I would like to express my deepest gratitude to my advisors, Dr. Jie Yang and Dr. Song Wu, for their excellent guidance, encouragement, patience and leading me to the biostatistics research field. As my advisors, they are very approachable and enlighten me on every research question I had. What I learnt from them are not only brilliant ideas but also a rigorous attitude to science research.

I would like to thank Prof. Wei Zhu, Dr. Xuefeng Wang, and Dr. Pei Fen Kuan who provided great comments and suggestions to my work. I would like to thank my prelim committee member, Prof. Stephen Finch who provided me the change to firstly collaborate with non-statisticians in a research project. Also, I would like to thanks all my lab mates for the insightful discussions to my research.

At the end, I would like to thanks my family, who support me and encourage me all the time when I work on my PhD research.

Publications

Jiawen Zhu, Song Wu, Jie Yang "An Integrated Method for Detecting MicroRNA Target Proteins through Reverse-phase Protein Arrays" accepted by *Journal of Computer Science & Systems Biology*

Chapter 1. Introduction

1.1. Biological Background

1.1.1. Gene Expression and Protein Biosynthesis

Gene expression is a conversion of the information encoded in a gene into messenger RNA (mRNA) and then to a protein. The process of gene expression is fundamental in all known organisms, such as eukaryotes, bacteria and archaea, and it generates the macromolecular machinery for life.

In general, the mechanism of gene expression for producing proteins involves two steps: transcription and translation (Figure 1.1). Firstly, coding information is transferred from DNA to an mRNA molecule with complementary base-paring. At this stage, a pre-mRNA molecule, which is later processed to form a mature mRNA, is generated by an enzyme called RNA polymerase. The mature mRNA molecule is a single-stranded copy of the gene (Figure 1.2). Secondly, mRNA is surrounded by ribosomes and decoded to produce specific polypeptides, according to rules specified by trinucleotide genetic codes. In translation, the mRNA is served as a template to guide the protein synthesis, which consists of four phases: activation, initiation, elongation, and termination.

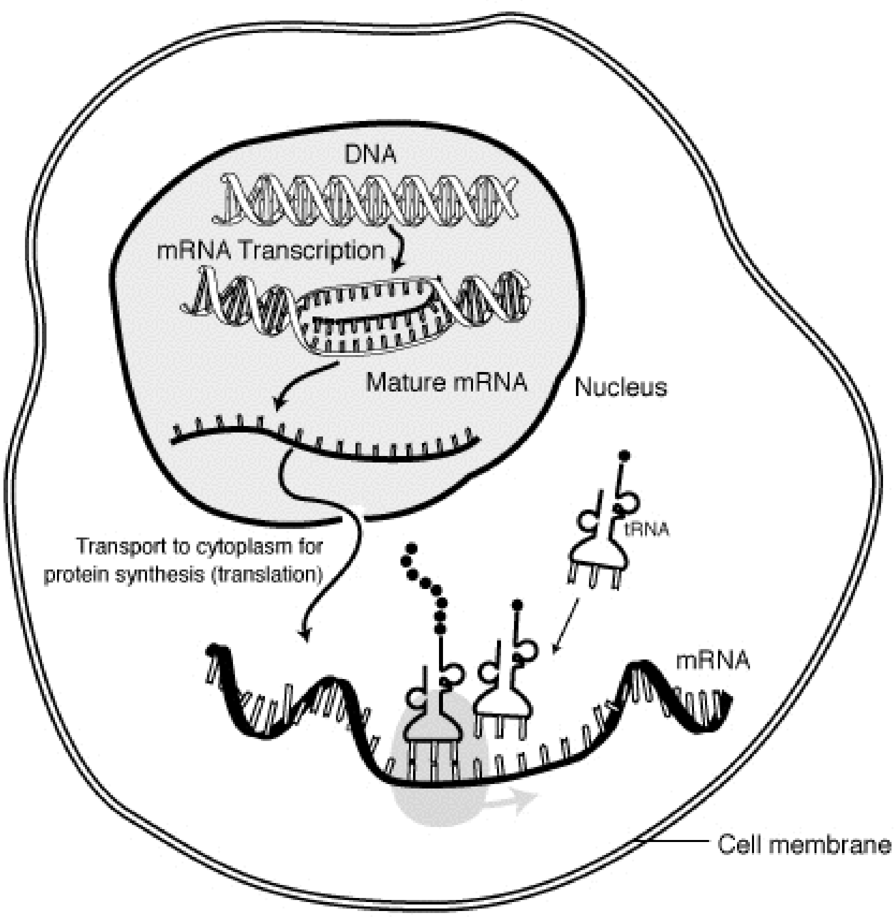


Figure 1.1| Two main procedures of the gene expression: transcription and translation (Wikipedia, 2014). Transcription is a procedure of creating a complementary RNA copy based on a DNA sequence. Translation is a procedure in mRNAs are decoded to produce specific sequences of amino acids in polypeptide chains.

In the activation phase of the translation, amino acids (AAs) are coupled with their corresponding transfer RNAs (tRNAs). tRNAs is called to be "charged" when an AA links to it. The second phase, initiation, involves small subunits of the ribosome binding to 5' end of mRNA with the help of initiation factors and other proteins. Elongation occurs when the "charged" tRNA in line binds to the ribosome along with GTP and an elongation factor. At the end, termination phase happens when the A site of the ribosome faces a stop codon (UAA, UAG, or UGA). When this happens, a releasing factor that recognizes the stop codon releases the synthesized polypeptide chain.

After translation, protein exists as an unfolded polypeptide which latter fold into their characteristic and functional three-dimensional structures by a physical process called protein folding. The total protein components present at the same time in a cell or cell type are referred as proteome. A study of such large-scale data sets defines the field of proteomics, analogy to the related field of genomics.

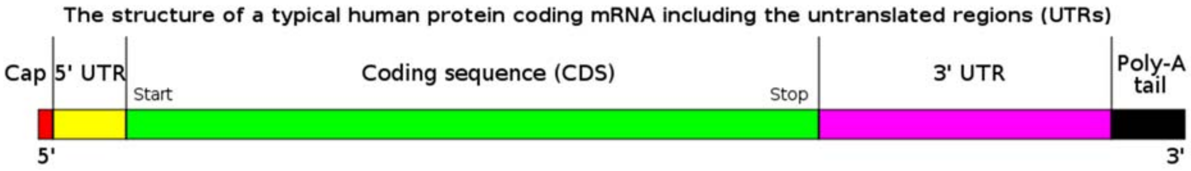

The structure of a typical human protein coding mRNA including the untranslated regions (UTRs)
Cap
5' UTR
Coding sequence (CDS)
Start
Stop
3' UTR
Poly-A tail
5'
3'

Figure 1.2|The structure of a typical human protein coding mRNA (Wikipedia, 2014). A mRNA contains an exact transcribed copy of the original DNA sequence in coding sequence area with 5’cap, 5’-untranslated region (UTR), 3’-UTR and Poly (A) tail. The poly (A) tail is a long sequence of adenine nucleotides (often several hundred) added to the 3' end of the pre-mRNA.

1.1.2. MicroRNA and Protein Expression

MicroRNA (miRNA) is a set of functional molecules that serves as regulators of gene expression. Recent evidence indicates that some miRNAs can function as tumor suppressors or oncogenes, and they are therefore referred to as ‘oncomires’(Esquela-Kerscher & Slack, 2006). MiRNAs have shown promise as biomarkers for many other diseases (Jeffrey, 2008; Jones, Nourse, Keane, Bhatnagar, & Gandhi, 2014; W. Zhang et al., 2012), which is one of the reasons that research in miRNA becomes very important nowadays.

miRNAs were originally discovered in 1993 by Victor Ambros, Rosalind Lee and Rhonda Feinbaum during a study for gene *lin-14* in the development of *C. elegans* (Lee, Feinbaum, & Ambros, 1993), which led to the discovery of the first miRNA, *lin-4*. However, the second miRNA, *let-7,* was not characterized until the year of 2000(Reinhart et al., 2000). Since then, thousands of miRNAs have been identified in different organisms such as plants, animals and some viruses (Hsu et al., 2014).

Although different miRNAs have different characteristics, they generally consist of 21-25 nucleotides. In animals, miRNA biogenesis usually consists of two steps: first, a newly generated microRNA transcript called pre-miRNA is processed into a precursor of ~70-nucleotide; then, the pre-miRNA is cleaved to generate a mature miRNA which is around 20 to 25 nucleotide (Figure 1.3).

In general, a miRNA regulates the expression of its target genes through two mechanisms – mRNA degradation or translation inhibition. That is, if a miRNA and its target gene can complement extensively, the miRNA-mRNA target may form a double-strand RNA (dsRNA) structure, which can be cleaved and degraded to reduce the mRNA expression and subsequently protein expression (Tang, Reinhart, Bartel, & Zamore, 2003; Xie, Kasschau, & Carrington, 2003). While if a miRNA and its target mRNA can only complement partially, the target mRNA will not be directly degraded but its translation may be repressed (Doench & Sharp, 2004; Zeng, Wagner, & Cullen, 2002). So, in either mechanism, the total protein level of miRNA targets would be reduced, resulting in their functional losses.

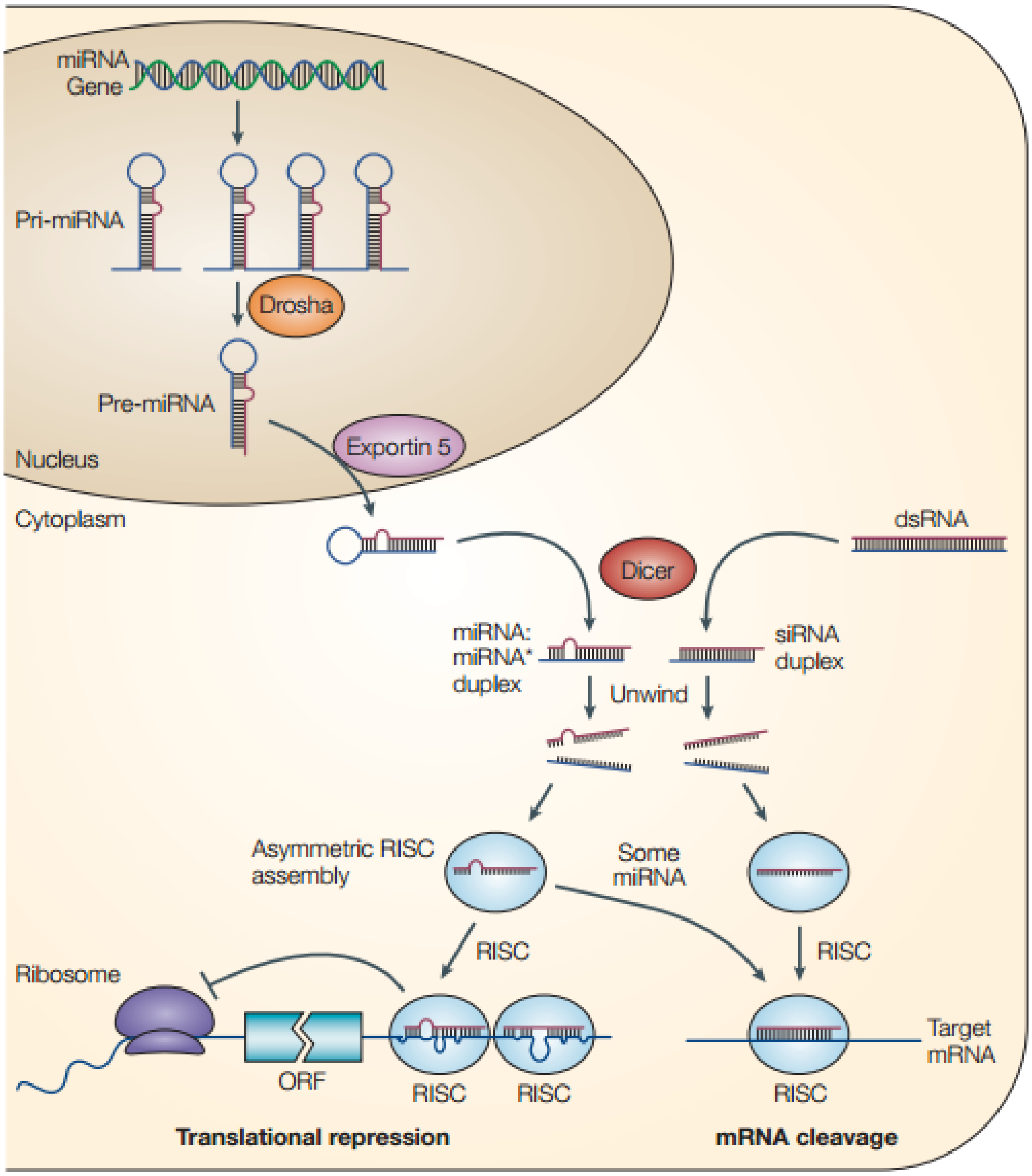


Figure 1.3| Mechanism of miRNA functions (L. He & Hannon, 2004). The upper part shows the procedure inside animal nucleus and the lower part shows how a pre-miRNA becomes a mature miRNA and how it regulates gene expression. If a miRNA and its target extensively complement, the RNA target is cleaved. If they partially complement, the target mRNA will not be depredated but its translation is repressed.

Many research have been done to investigate biological functions of miRNAs. In 2005, J. Brennecke *et al*. provides evidence that on average a miRNA has approximately 100 target sites, which indicates that miRNAs regulate a large fraction of protein-coding genes. It has also been shown that the 3'-ends of mRNAs are key determinants of target specificity for miRNA families (Brennecke, Stark, Russell, & Cohen, 2005). In 2008, two groups (Daehyun Baek et al., 2008; Selbach et al., 2008) have used variants of a technique known as SILAC (stable isotope labeling with amino acids in cell culture) to measure proteome-wide changes. They found that while miRNAs can directly repress the translation of hundreds of genes, additional indirect effects result in changes in the expression of thousands of other genes. And many of the changes they observed were less than two-fold in magnitude. Their findings indicate that either directly or indirectly, miRNAs can fine-tune protein synthesis to match the needs of a cell at any given time. Nonetheless, the studies cannot provide information in how relevant miRNA regulation and protein production are, and there is no systematic statistical method to model the relationship.

Although the biological importance of miRNAs is clear, their biological functions still remain largely unknown. So far, only few miRNAs have been functionally characterized. Knowing miRNA target genes will help understand miRNA functions in many different situations, and hence research for the miRNA target identification is in great need.

1.1.3. MiRNA Targeting

Based on the fact that the sequences of miRNAs and their target genes complement to each other, or at least partially, one way for the miRNA target identification is through *in silico* prediction. Several software tools, such as miRanda (D. Baek et al., 2008) and TargetScan (Lewis, Shih, Jones-Rhoades, Bartel, & Burge, 2003), have been developed for such purpose. miRanda scores the likelihood of mRNA downregulation (in which process the targeted cellular component decreases) according to a regression based machine learning method--the mirSVR, which is trained on sequence and contextual features of the predicted miRNA::mRNA duplex (Betel, Koppal, Agius, Sander, & Leslie, 2010). In contrast, TargetScan studies the RNA::RNA duplex interactions according to a thermodynamics-based modeling and comparative sequence analysis to predict miRNA targets conserved across multiple genomes. Several databases under *microRNA.org* and *targetscan.org* have been generated from these computation-based analyses. However, one major limitation form these *in silico* predictions is that they all suffer from big false positive rates, which hinder their practical use. Usually these resources are best used as candidates in the preliminary screening step or as supporting evidences for findings from other methods.

Another popular way to determine miRNA targets is through experimental data by measuring downstream effects of miRNAs. Currently, scanning of the miRNA targets is mainly through testing negative correlations between miRNAs' and mRNAs' expression levels. For example, high-throughput techniques, such as miRNA and mRNA gene microarray, can be applied to measure their expression levels, and then the correlation analyses can be conducted subsequently to filter out miRNA-mRNA pairs that show significant negative correlations as potential candidate pairs for further analyses(Brennecke et al., 2005). More recently, with the advent and rapid advance of sequencing techniques, the miRNA sequencing (miRNA-seq) and RNA sequencing (RNA-seq) platforms have become more and more popular for the quantification of the miRNA and RNA expressions. However, the main drawback for the miRNA/mRNA correlation analysis is that they can only identify miRNA targets that may change at mRNA levels, but is determined to fail for those modulated through translation inhibition.

Recent evidence has shown that the regulation of a miRNA on its targeted mRNA level is moderate, and its effect on protein levels is more profound (Bartel, 2009). Since miRNAs can induce protein reduction via both functional mechanisms, logically it should be more sensible to examine correlations between miRNA levels and targeted protein levels directly. However, little or no study has explored the miRNA targets based on protein expression data, largely due to extreme difficulties in quantifying protein expression through high-throughput screening. Taking advantage of a recently emerged high-throughput technique for protein quantification, the Reverse-phase Protein Array (RPPA) assays (see section 1.22), we aim to develop statistical methodologies that can identify miRNA targets directly using this new protein expression data type.

1.2. Molecular Detection

1.2.1. MiRNA Detection

Currently, there are mainly three methods for miRNA profiling: quantitative reverse transcription PCR-based methods (QRT-PCR), miRNA microarray and RNA-sequencing (RNA-seq) (Pritchard, Cheng, & Tewari, 2012). Each method has its unique pros and cons. QRT-PCR is very sensitive and specific, but is expensive and can examine only one gene at a time. The second method, miRNA microarray, has fairly low-cost and is high-throughput with respect to the number of samples that can be processed per day; however, it typically has lower specificity than qRT-PCR. miRNA-seq is the most recent method that is based on the next-generation sequence (NGS) technique. It has reasonable cost while maintains high specificity in distinguishing miRNAs that are very similar in sequence. In our study here, miRNA data are generated from miRNA-seq.

Usually, miRNA-seq has the following four steps (Lu, Meyers, & Green, 2007): Firstly, isolate low molecular weight (LMW) RNA from the tissue of interest; secondly, based on polyacrylamide gel-based size fractionation, purify small RNAs (20–30 nucleotides) from the LMW RNA fraction and ligate them to a 5′-end RNA adapter. An excess of adapter over small RNAs is used to prevent self-ligation of small RNAs; thirdly, ligate a 3′-end RNA adapter which is modified to prevent circularization to the gel-purified product from the 5′-end adapter ligation. In this step, chemical synthesis of an oligonucleotide containing a 3′-end non-nucleotidic group will block the 3′-end hydroxyl; fourthly, use a low number of PCR cycles to obtain sufficient amount of templates for sequencing after reverse transcription.

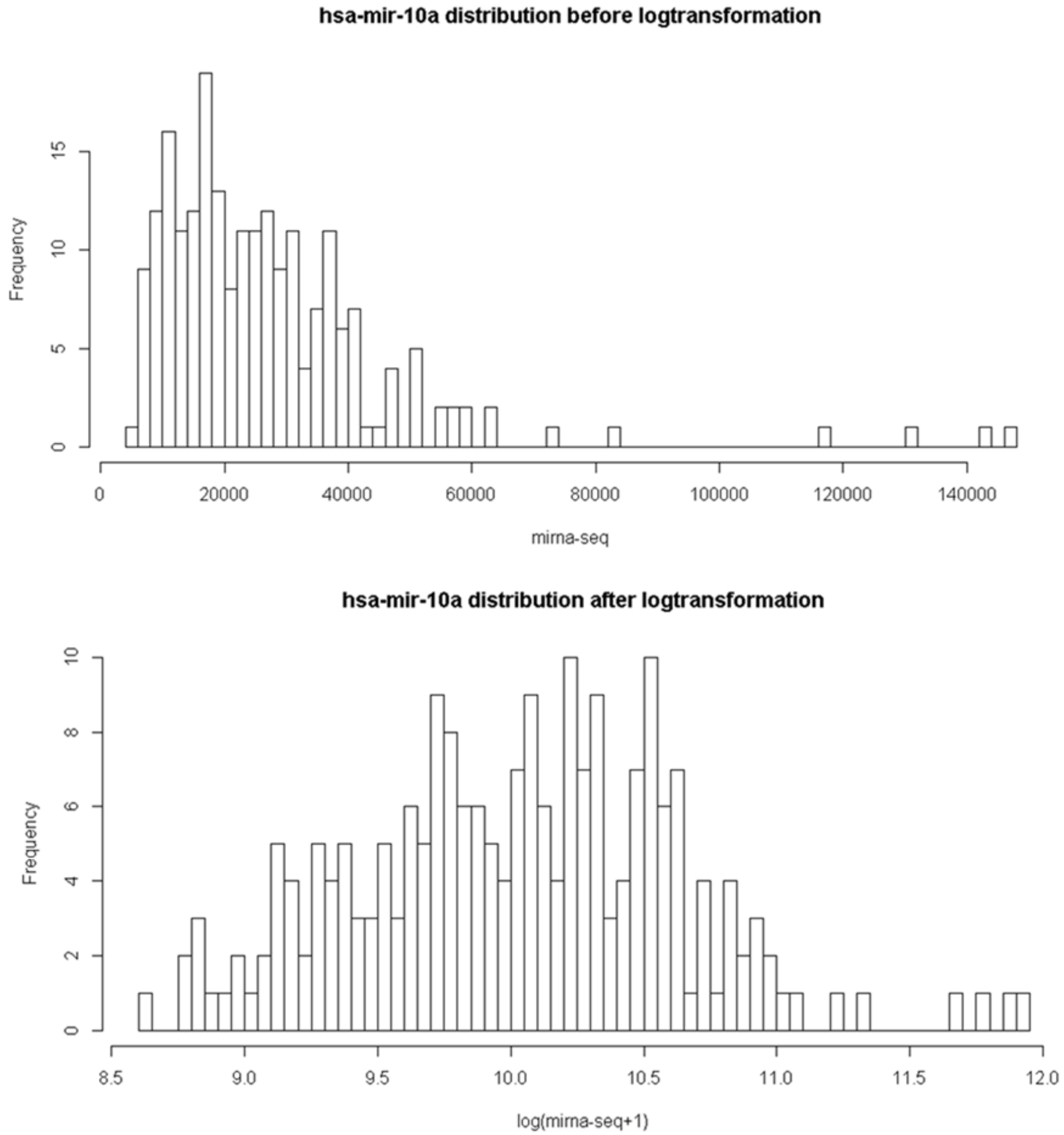


Figure 1.4| Distribution of *has-mir-10a* miRNA-seq data (Upper) before and (Lower) after log-transformation. Usually miRNA-seq data is right skewed, and log-transformation is a common strategy to normalize the data.

The miRNA-seq data we used are available in a database from The Cancer Genome Atlas (TCGA) program, provided by BC Cancer Agency (bcgsc.ca). Under platform Illumina Hiseq, *"*.mirna.quantification.txt*" files which contain the expression levels of miRNAs in particular samples were used in our study. Figure 1.4 is an example of miRNA-seq data from TCGA ovarian cancer dataset. For most cases, miRNA-seq data are right skewed (upper part of Figure 1.4) and a log-transformation strategy-- $log(miRNA + 1)$ is used to normalize the data in our analysis (lower part of Figure 1.4).

1.2.2. Reverse Phase Protein Array

The principles of reverse phase protein array (RPPA) technology were largely described by Roger Ekins (Ekins, 1998) in his work on "ligand assays" more than 20 years ago. However, this technology has not been used in clinical trials until 5 years ago (Mueller, Liotta, & Espina, 2010). There are three techniques to detect specific proteins in a given sample: RPPA, sandwich array, forward phase array (Figure 1.5). Among them, RPPA technology, which is also called protein lysate array, is an emerging technology and a new means for estimating protein expression levels. It results from an attempt to extend the microarray approach to measure proteins. The term "reverse phase", comparing to the foreword phase array, refers to the fact that the antigen is immobilized, rather than an antibody being immobilized as capturing molecules.

Recently, RPPA has become more and more popular. One of its advantages is that it is very sensitive and only requires a minimal amount of protein extracts for the array. Another advantage is that multiple replicates and dilutions can be incorporated into experiment design, thus making the protein level quantification more accurate.

In RPPA, the biological samples of interest are first lysed, yielding a homogeneous mixture (lysates), and then these lysates are printed onto an array according to a set of dilution series. The arrays are typically glasses coated with a nitrocellulose membrane on one side, and the lysates are printed on the nitrocellulose.

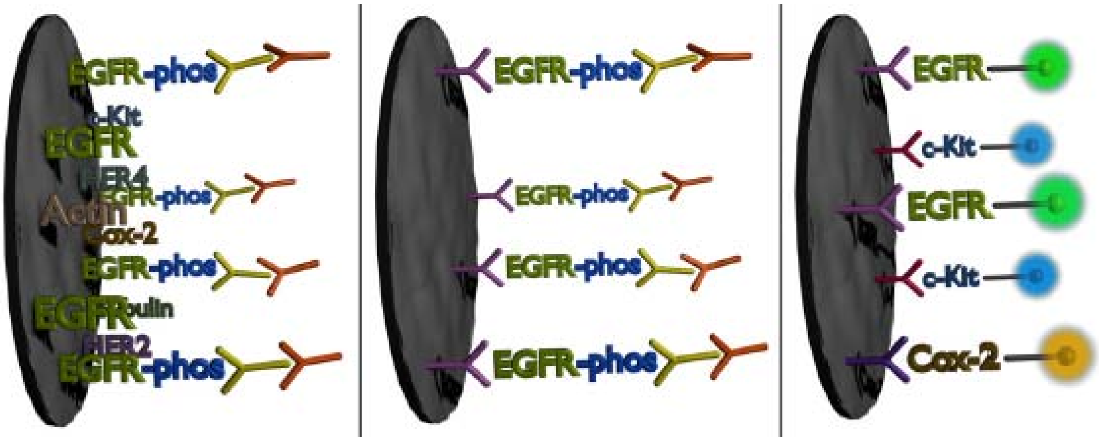


Figure 1.5| A directive view of three protein arrays (Mueller et al., 2010). (Left): RPPA; (Middle): Sandwich array; (Right): Forward Phase Array. The term "reverse phase" of RPPA, comparing to the forward phase array, refers to the fact that the antigen is immobilized, rather than an antibody being immobilized as a capturing molecule. Sandwich arrays require a pair of antibodies to capture the protein of interest and to detect unique epitopes of the same protein on the sample. In a forward phase, antibodies are immobilized rather than proteins on a surface to capture proteins from a sample.

A serial dilution is a stepwise dilution of a substance in solution. Usually the dilution factor at each step is constant, resulting in a geometric progression of the concentration in a logarithmic fashion. A 2-fold serial dilution is 1-unit, 0.5-unit, 0.25-unit, and 0.125-unit and so on. Serial dilutions can result in concentration curves with a logarithmic scale, which gives more accurate estimation of protein concentrations.

To measure a specific protein, the array is first incubated with its antibody. Then the array is interrogated with a labeled secondary antibody, which recognizes the primary antibody. The secondary antibody is linked to an enzyme to generate detectable signals. Thirdly, the enzyme substrate is introduced to react with the enzyme, causing precipitate. More protein of interest at a spot attracts more enzyme molecules, which subsequently yields more precipitate. After a short reacting period, loose substrate is then washed away. At the end, the array is imaged, typically with a flatbed scanner, producing a TIF image file (Figure 1.6). By using appropriate software such as MicroVigene, the printed spots in the image file are quantified.

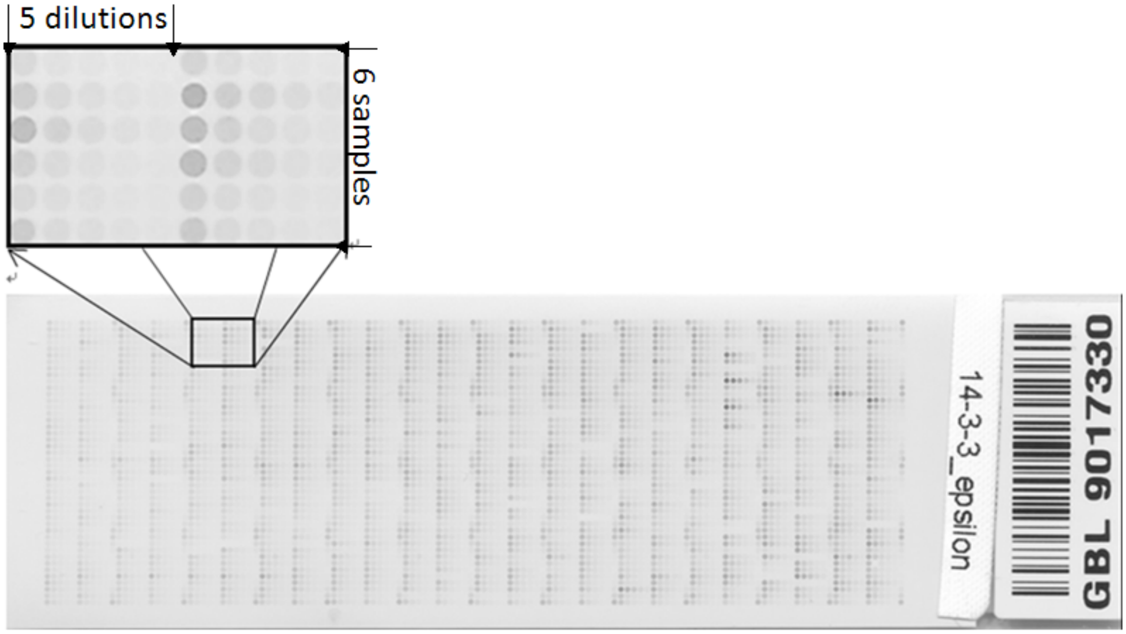


Figure 1.6| A sample image file of RPPA from TCGA program. The magnified part contains 12 samples and each sample is in 5 dilutions. The pixel values of each spot are combind to give the intensity level of a spot.

Figure 1.6 shows a typical RPPA array, in which there are many dark “lines” as columns. After magnifying one small area, it shows that many small dark spots form those lines. The image intensity level of each spot measured by pixels directly reflects the protein expression level. Usually each sample has several spots with decreasing darkness. For example, in the magnified part of Figure 1.6, the first 5 spots in the first row are from one sample, even though with different intensities, which is because those are from the serial dilution of one sample. The more a sample dilutes, the lighter the spot will be.

Serial dilution is a special feature of RPPA compared with general microarray techniques. It is designed to have accurate measurements of protein concentrations over a wide dynamic range. Because protein concentrations can vary over large orders of magnitude in patients or cell line samples, proteins with very high concentration may saturate the image and make them inestimable. This is mainly an issue related to digital image quantification, in which the intensity of a pixel is stored as an 8-bit integer, giving a range of possible values from 0 to 255. Once all pixels at a spot are saturated, it will cause problem in quantifying protein concentration. Therefore, diluting each sample multiple times on an RPPA slide is a good way to solve this problem. In this case, if a protein concentration on the original sample is close to saturation, the sample can still be well measured at other diluted concentrations.

The relationship of gray level intensities in the image and protein concentrations can be reflected through a response curve (Figure 1.7). Usually it is modeled as a sigmoid shape, which reflects the key characteristics of these data. The flatness at the lower end reflects background noises and the plateau at the higher end reflects the saturated signals. Note that the protein concentrations estimated from the response curve are relative quantities, and usually one protein is arbitrarily chosen as the reference and the other protein levels are expressed as ratios to the reference. However, the relative quantification suffices for our purposes in this miRNA target study.

The Cancer Genome Atlas (TCGA) program contains both RPPA data and miRNA-seq data from same patients, which will be used later in this work to illustrate our proposed method.

RPPA slide image measurements, i.e. Level 1 data, were provided by MD Anderson Cancer Center (mdanderson.org).

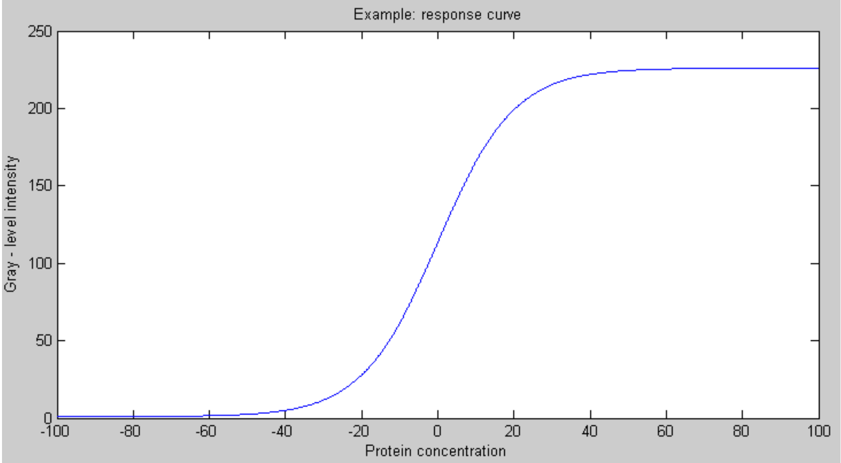


Figure 1.7| The image intensity-protein concentration response curve. The curve has an S-shape because the range of intensity levels has a natural upper bound and lower bound. Value "0" in $\boldsymbol{x}$-axis stands for the reference protein concentration.

1.3. Mixed Models

Linear regression models with random errors are classical statistical models for continuous variables. A linear mixed model incorporates random effects in additional to fixed effects in the classical linear regression models. It can be represented as $\boldsymbol{Y} = \boldsymbol{X\beta} + \boldsymbol{Z\gamma} + \boldsymbol{\varepsilon}$, where $\boldsymbol{Y}$ is the dependent variable; $\boldsymbol{\beta}$ is the coefficient vector of fixed effects; $\boldsymbol{\gamma}$ represents the random effects variables; $\boldsymbol{X}$ *and* $\boldsymbol{Z}$ are known design matrix relating to the dependent variable $\boldsymbol{Y}$. In linear mixed model, we assume $\boldsymbol{\gamma}$ and $\boldsymbol{\varepsilon}$ are normally distributed and independent to each other, i.e. $\boldsymbol{\gamma} \sim \boldsymbol{MVN}(\boldsymbol{0}, \boldsymbol{G})$, $\boldsymbol{\varepsilon} \sim \boldsymbol{MVN}(\boldsymbol{0}, \boldsymbol{R})$ and $\boldsymbol{\gamma} \perp \boldsymbol{\varepsilon}$. $\boldsymbol{Z\gamma}$ and $\boldsymbol{\varepsilon}$ can describe a complex dependence structure among $\boldsymbol{Y}$, and the variance of $\boldsymbol{Y}$ is then $\boldsymbol{V} = \boldsymbol{ZGZ'} + \boldsymbol{R}$. Therefore, a linear mixed model provides a general solution for repeated measurements on each subject over time or space, or multiple related measurements at one time. In addition, a linear mixed model allows using data from subjects with missing measurements as long as the missing mechanism is missing at random, in which the missing is not related to the value of the variable that has missing data. A likelihood-based method is commonly used to estimate unknown covariance parameters in linear mixed models, e.g., the PROC MIXED in SAS.

A nonlinear mixed model (NLMM) is a straightforward extension of a linear mixed model with random effects appearing in a nonlinear function. For example, it is widely used in pharmacokinetics or used to describe over-dispersed binomial data. A general NLMM can be written as

$$y_{ij} = \eta(\boldsymbol{\tau_i}, t_{ij}) + \epsilon_{ij}$$

$$\boldsymbol{\tau_i} = \boldsymbol{A_i}\boldsymbol{\beta_i} + \boldsymbol{B_i}\boldsymbol{b_i}$$

$$\boldsymbol{b_i} \sim MVN(0, D), \boldsymbol{\epsilon_i} \sim MVN(0, R), \boldsymbol{b_i} \perp \boldsymbol{\epsilon_i}, \boldsymbol{\epsilon_i} = \{\epsilon_{ij} | j = 1,2, \ldots J\}$$

Where y_{ij} is the observation from the ith subject and the jth measurement, η is a known function which depends on parameter vector $\boldsymbol{\tau_i}$ and t_{ij}, a variable related to the time or space of measurement. Unknown parameters in a NLMM are generally estimated using a maximum likelihood principle. Many statistical software tools provide efficient functions to do so, such as PROC NLMIXED in SAS 9.3 (SAS Institute Inc., Cary, NC).

1.4. Splines

A spline is a piecewise-polynomial real function that possesses a sufficiently high degree of smoothness at certain pre-specified connection points (or called knots). The smoothness of a spline is controlled by the order of each piecewise-polynomial function. A good property of splines is that it can fit to any smooth function with sufficient order and knots. Some details about the spline representation are given below.

Define a spline function $S: [a,b] \rightarrow R$ on an interval $[a,b]$ composed of k order (degree=$k-1$) disjoint subintervals $[t_{i-1}, t_i]$ with $a = t_0 < t_1 < \cdots < t_{n-1} < t_n = b$.

The restriction of S to an interval i is a polynomial

$$P_i: [t_{i-1}, t_i] \rightarrow R$$

so that

$$S(t) = P_1(t), t_0 \leq t \leq t_1,$$

$$S(t) = P_2(t), t_1 \leq t \leq t_2,$$

$$\vdots$$

$$S(t) = P_n(t), t_{n-1} \leq t \leq t_n,$$

The highest order of the polynomials $P_i(t)$ is said to be the order of the spline S. If all subintervals are of the same length, the spline is said to be uniform; otherwise non-uniform. For a spline of order k, S is continuously differentiable to order $k-1$ at the interior points t_i: for all $i = 1, 2, \ldots, n-1$ and all $j, 0 < j < k-1$,

$$P_i^{(j)}(t_i) = P_{i+1}^{(j)}(t_i).$$

A B-spline (basis spline), which is relatively complex to be constructed, is a spline function that has minimal support to maintain a given degree. Besides the $n+1$ internal knots with the $n-1$ inner knots and the two boundary knots, there are $2k$ additional knots, where k is the degree of B-spline functions. For B-splines, the first and last k knots are "clamped": $t_0 = t_1 = \ldots = t_k$ and $t_{n+k} = t_{n+k+1} = \cdots = t_{2k+n}$. The B-spline basis functions with degree k, $\{B_{j,k}\}$, can be constructed recursively using splines with lower degrees $\{B_{j,p}, p = 0, \ldots, k-1\}$ as following (the Cox-de Boor recursion formula):

$$B_{j,0}(t) = \begin{cases} 1, & if\ t_j \le t < t_{j+1} \\ 0, & otherwise \end{cases} \quad j = 0, \ldots, 2k+n-1;$$

$$B_{j,p}(t) = \frac{t - t_j}{t_{j+p} - t_j} B_{j,p-1}(t) + \frac{t_{j+p+1} - t}{t_{j+p+1} - t_{j+1}} B_{j+1,\,p-1}(t)\,, j = 0, \ldots, 2k+n-p-1;$$

Thus the B-spline function $S(t)$ can be written as $\sum_{j=0}^{k+n-1} \beta_j B_{j,k}$, which is a linear combination of B-spline basis functions. The number of pieces will be determined once the number of knots, *2k+n+1*, and the degree of a B-spline function, *k*, is usually chosen. At the $n-1$ inner knots, basis functions satisfy C^{p-1} continuity when they are not zero. In general, the lower the degree of B-spline function is, the closer it follows the polyline formed by control points $\{\beta_j\}$ (control polyline). And a B-spline curve has a strong convex hull property. That is, it is contained in the convex hull of the control polyline.

1.5. Outline of the Dissertation

Several methods have been published to quantify protein levels using RPPA and they were broadly categorized into two groups: single sample estimation and joint sample estimation methods. Those methods will be introduced in Chapter 2, and based on those methods, a naïve method to identify miRNA targets will be described.

In Chapter 3, we propose a parametric integrated model (PIM) based on a nonlinear mixed model to identify miRNA targets. Simulation studies and real data analyses are conducted to compare the naïve method and the integrated method (Sections 3.5-3.7). Chapter 3 of this dissertation in part (including Figures and Tables) is from the materials as it appears in Journal of Computer Science & Systems Biology. The co-authors, Dr. Song Wu and Dr. Jie Yang listed in the publications directed and supervised the research that forms the basis for this Chapter.

To further improve the robustness of our integrated method, we also propose a semi-parametric model by incorporating a nonparametric curve fitting technique for RPPA data, which relaxes the assumption of a specific parametric form for the RPPA response curve. This model is called the semi-parametric integrated model (SPIM). Comparison between SPIM and PIM is illustrated using simulation studies and real data analyses (Sections 4.4-4.5).

In the last chapter, Chapter 5, the advantages and disadvantages of these models are summarized and the future research directions is discussed also.

Chapter 2. A Naïve Method for Detecting miRNA Targets

2.1. Motivation

As discussed in Chapter 1, miRNA may regulate the expression of its target genes through two mechanisms: mRNA degradation or translation inhibition. In either case, the final effect of miRNA to its targets is to reduce their protein levels, resulting in functional losses. The conventional approach for screening miRNA target identification is to study the correlations between mRNA and miRNA levels, which is determined to miss targets regulated through translation inhibition. Therefore, a better way to identify miRNA targets should be based on protein expression data. However, due to difficulties in quantifying protein expression through high-throughput screening, little or no study has been done in this way.

Emerging technologies, RPPA for quantifying protein concentration and RNA-seq for quantifying miRNA expression, provide a unique opportunity to study miRNA-protein regulatory mechanisms. Since protein concentrations can be estimated from the RPPA and miRNA expression levels can be measured from miRNA-seq, a straightforward way to examine the miRNA-protein relationship is through the Pearson's correlation coefficients or simple linear regression models, which is referred as the *naïve model* in this work.

In the following, we will first review several statistical methods of protein quantification from RPPA data, and then introduce the naïve model in details.

2.2. Methods for Protein Quantification from RPPA Data

RPPA generates data with serial dilutions in terms of imaging intensities, which requires further process to quantify protein concentrations. The main goal of protein quantification is to estimate the relative protein concentrations for different samples based on their imaging intensity data. Several methods have been proposed so far, and they roughly fall into two general categories: single sample estimation methods and joint sample estimation methods.

2.2.1. Single Sample Methods

Inverse Linear Spline Interpolation Method

Single sample methods estimated protein concentrations using information from a single sample only. Such a method was first proposed in 2003 (Nishizuka et al., 2003). The idea is to use linear spline interpolation to generate a piecewise linear curve f_i passing through all (j, y_{ij}) for the ith sample and the jth dilution, and calculate a global "reading point" λ to "read" the protein expression level, $\hat{\varphi}_i$, from $f_i^{-1}(\lambda)$. Here y_{ij} is the image intensity level of the jth dilution from the ith sample. One main drawback of the method is that the observational error is not able to be considered in the model. Also, protein concentration of a sample cannot be estimated if its intensity interval does not contain the reading point λ.

Robust Estimation Methods

Later in 2005, two robust estimation methods were proposed by C. Mircean *et al.* (Mircean et al., 2005). They used statistical methods to handle measurement errors and aimed to improve estimation accuracies based on a standard simple linear regression approach. In this study, RPPAs with three technical replicates and 6 dilutions of one biological sample were used. The idea of their first robust estimation method is to fit a simple linear regression line through median values of log-transformed intensity y_{ij} among three technical replicates and the dilution index j for each biological sample. Their second method is to use a robust least square approach to fit a simple linear regression line through all log-transformed intensities of technical replicates and their dilution indexes in each biological sample. Briefly, each linear regression line contains the protein concentration information for one biological sample.

To estimate the difference between two biological samples, the distance of their fitted line was used as the log-ratio of their protein expression. When the lines are not parallel, a summarized statistics of the distance is needed. To estimate this, the method was based on the intuition that "the higher the dispersion for a particular dilution, the less the weight this dilution should get when calculating the distance between two fitted lines and consequently, the less the influence this dilution should have on the final estimate of the ratio of protein expressions between the two samples" (Mircean et al., 2005). Thus the weight can be estimated by the inverse of the interquartile range or the coefficient of variation (standard deviation divided by the mean) of the intensity values for each dilution (Figure 2.1).

However, when protein intensities are close to saturation, the linear relationship of log-intensity and dilution index is unlikely to be true, and it leads to inaccurate estimations of protein concentrations, which is the main disadvantage of these methods.

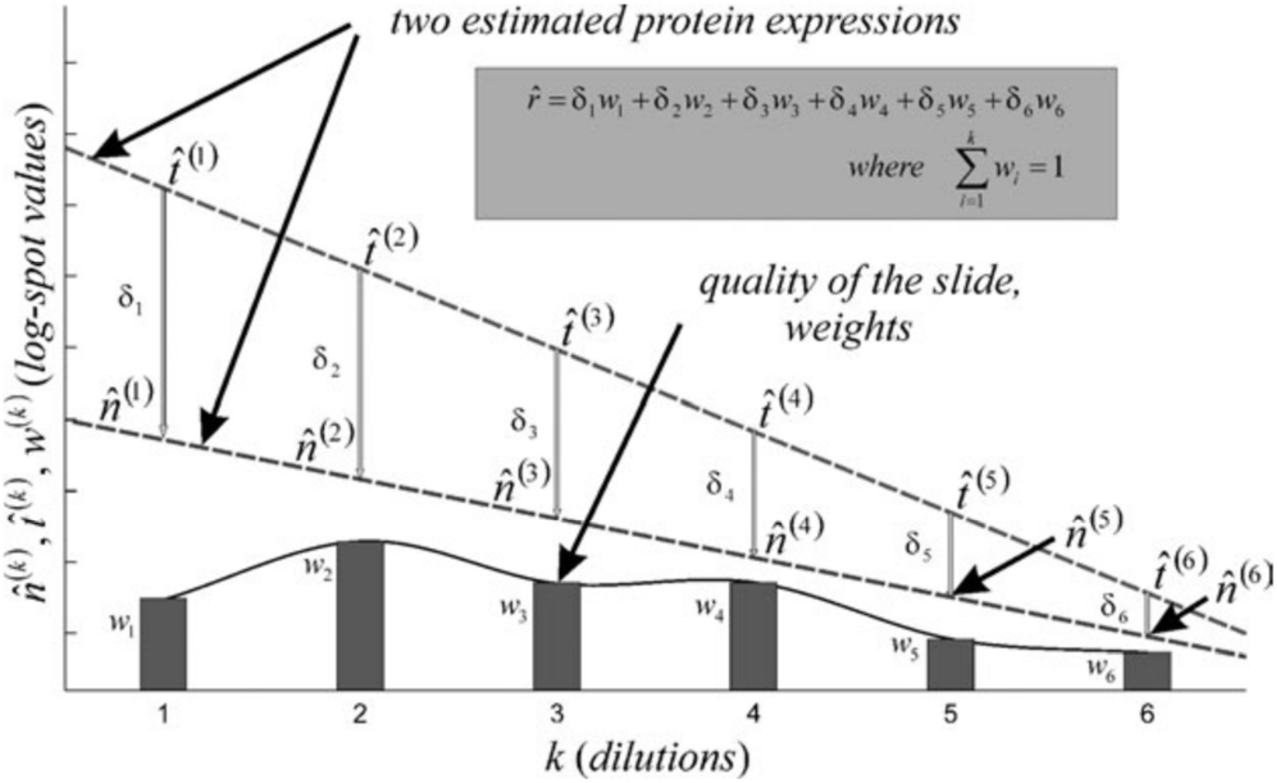


Figure 2.1| Regression lines of two biological samples (Mircean et al., 2005). The protein expression ratio $\boldsymbol{r}$ can be measured by the distance of two regression line. When the lines are not parallel, a weighted sum of distance of two fitted lines at different dilutions is used as a summarized statistics.

2.2.2. Joint Sample Methods

Since each RPPA slide is probed with a single antibody for each kind of protein, protein expression of different samples should have similar chemistry and hybridization behaviors. For instance, all samples should share the same baseline level, saturation level and the rate of signal increase at each dilution. Joint sample methods take into consideration of this by using information from all the samples on an array to compute sample parameters (protein concentration estimates), as well as global slide parameters. That is, the joint estimation potentially improves estimation by summarizing information across all samples and hence all samples on an array contribute to an overall protein concentration-intensity response curve. In this way, sample parameter estimates are expected to have smaller errors. Below, we review four different methods with this philosophy: nonlinear parametric regression method, nonparametric regression method, serial dilution curve method, and multistep protein lysate array quantification method.

Nonlinear parametric regression method

A nonlinear parametric regression method was proposed by Tabus *et. al.* in 2006 (Tabus et al., 2006), which used sigmoidal or polynomial models to mimic the RPPA response curve.

Denote the image intensity from the tth technical replicates of biological sample i at the jth dilution step as y_{ijt}, $(i = 1, \ldots, I,\ j = 1,2, \ldots J, t = 1,2, \ldots, K)$.The binary logarithm of the median effective protein concentration level (EC_{50}), a single quantity per dilution series to represent the concentration of the protein in the ith sample, is denoted as x_i. $x_{ij} = x_i + l_j$ is the binary logarithm of the protein concentration in the jth dilution step where $l_j = \frac{(1+J)}{2} - j$ for 2-fold dilution cases. Assuming the functional relationship between intensity level y_{ijk} in the jth spot and x_{ij} is described as

$$E(y_{ijk}) = g(x_{ij}, \boldsymbol{\beta})$$

Specifically, a polynomial model to describe such relationship is:

$$g_p(x, \boldsymbol{\beta}) = \beta_0 + \beta_1 x + \cdots + \beta_k x^P$$

where β_0 represents a reference intensity when $x = 0$ which can be set to zero.

And a sigmoidal model is:

$$g_s(x, \boldsymbol{\beta}) = \beta_1 + \frac{\beta_2}{(1 + 2^{-\beta_3(x-\beta_0)})}$$

In the sigmoidal curve, all parameters can be interpretable: β_1 is the baseline saturation level ; β_2 is the increment from β_1 to the saturated intensity level (since when $x - \beta_0$ is very large, $g_s(x, \beta) \to \beta_2 + \beta_1$, and when $x - \beta_0$ is very small, $g_s(x, \beta) \to \beta_1$); β_0 is a reference intensity which can be set to zero as well. In addition, when $x = 0$, we have

$$\frac{d^2 g}{dx^2}|_{x=0} = 0$$

and

$$\frac{dg}{dx}|_{x=0} = \left(\frac{\beta_3 \beta_2}{4}\right).$$

So the steepness of a sigmoidal curve can be directly controlled by the parameter β_3, when β_1, β_2 are fixed. A model for the variance is assumed to be:

$$\mathrm{var}\left(y_{ijk} \middle| x_{ij}, \boldsymbol{\beta}\right) = \sigma_0^2 g(x_{ij}, \boldsymbol{\beta})^{2\alpha},$$

where σ_0^2 is a variance parameter. Furthermore, $y_{ijk}|\theta = [x^T\ \boldsymbol{\beta}^T\ \sigma_0\ \alpha]^T$ are assumed to be normally distributed:

$$y_{ijk}|\theta \sim N(g\left(x_{ij}, \boldsymbol{\beta}\right), \sigma_0^2 g\left(x_{ij}, \boldsymbol{\beta}\right)^{2\alpha})$$

Four models were examined by Tabus *et al.* (2006):

1) Model M_1 in which $g = g_p$ and $\alpha = 0$;

2) Model M_2 in which $g = g_p$ and α is in the interval [-3,3];

3) Model M_3 in which $g = g_s$ and $\alpha = 0$;

4) Model M_4 in which $g = g_s$ and α is in the interval [-3,3].

The parameter estimation algorithm for a polynomial model

The estimation of parameters in polynomial models with degree P is based on an optimization of weighted nonlinear least square:

$$\sum_{\substack{i=1,2,\dots I\\ j=1,2,\dots J\\ k=1,2,\dots K}} w_{ijk}^2 \left(y_{ijk} - \sum_{m=1}^{P} \beta_m x^m \right)^2$$

where w_{ijk} is a weight which can be set to zero to eliminate data points with poor image quality. The parameter estimation algorithm for a polynomial model with degree P can be described as following:

Firstly, iteration starts by generating initial values of $\{x_i\}$ from a polynomial function with order 1.

Secondly, set the parameters $\widehat{\boldsymbol{\beta}}^{(h)}$ to be

$$\widehat{\boldsymbol{\beta}} = \left(\sum_{\substack{i=1,2,\dots I\\ j=1,2,\dots J\\ k=1,2,\dots K}} w_{ijk}^2 \widehat{\boldsymbol{\varphi}}_{\boldsymbol{ijk}} \widehat{\boldsymbol{\varphi}}_{\boldsymbol{ijk}}^{\boldsymbol{T}} \right)^{-1} \sum_{\substack{i=1,2,\dots I\\ j=1,2,\dots J\\ k=1,2,..K}} w_{ijk}^2 \widehat{\boldsymbol{\varphi}}_{\boldsymbol{ijk}} y_{ijk}$$

where $\widehat{\boldsymbol{\varphi}}_{\boldsymbol{ijk}} = [(\hat{x}_{ij}), (\hat{x}_{ij})^2, \dots, (\hat{x}_{ij})^P]^T$ and $\hat{x}_{ij} = \hat{x}^{(h-1)}{}_i + l_j$ in the hth iteration step. A standard linear least square routine with constraints is employed if the function $g_p(x_{ijk}, \hat{\beta})$, $i = 1, 2 \dots, I; j = 1,2.., J; k = 1,2, \dots K$, is not monotonically increasing.

Thirdly, find the domains of increasing monotonicity of $g_p(x, \hat{\beta})$ which is a function of x, and take $A = [x_{low}, x_{up}]$ as the interval of increasing monotonicity containing the largest number of $x_{ij}, i = 1, 2, 3, \dots, I, j = 1,2, \dots J$.

Fourthly, calculate the set $S^*\left(P'(x_i; \hat{\beta})\right)$ of roots in the following polynomial function of degree $(2P - 1)$, which are located inside interval A for biological sample i:

$$P'(x_i; \hat{\beta}) = -2 \sum_{\substack{j=1,2,..J \\ k=1,2,\dots K}} w_{ijk}^2 (y_{ijk} - \sum_{m=1}^{P} \hat{\beta}_m (x_i + l_j)^m) \sum_{m=1}^{P} m\hat{\beta}_m (x_i + l_j)^{m-1}$$

And add the bounds $\{x_{low}, x_{up}\}$ to the set $S^*\left(P'(q_j; \hat{\beta})\right)$. Then update the ith element of $\hat{x}^{(h)}$ to be

$$\hat{x}_i = arg \min_{x_i \in S^*\left(P'(x_i; \hat{\beta})\right)} \sum_{\substack{j=1,2,\dots J \\ k=1,2,\dots K}} w_{ijk}^2 \left(y_{ijk} - \sum_{m=1}^{P} \hat{\beta}_m \left(x_i + l_j\right)^m \right)^2$$

If there is no significant improvement of the weighted nonlinear least square, the algorithm stops. Otherwise, the algorithm goes to the $h + 1$ iteration step.

Parameter estimation algorithm for sigmoidal models

The parameter estimation problem for sigmoidal models can be simplified to the case of polynomial models.

Firstly, define a variable $r(x)$ where

$$r(x) = \exp(-\beta_3 x) = \frac{\beta_1 + \beta_2 - g_s(x)}{g_s(x) - \beta_1}$$

In order to minimize the nonlinear least square function

$$\tau(x_i) = \sum_{\substack{j=1,2,\ldots I \\ k=1,2,\ldots K}} w_{ijk}^2 (y_{ijk} - g_s(x_i + l_j))^2,$$

the derivative of it is computed and set to zero.

Since

$$r(x_i + l_j) = exp(-\beta_3 l_j) r(x_i) = \gamma^{-l_j} r(x_i), \ \ \gamma = \exp(\beta_3)$$

$$\frac{dg_s(x_i + l_j)}{dx_i} = \beta_3 \beta_2 \frac{r(x_i + l_j)}{\left(1 + r(x_i + l_j)\right)^2} = \beta_3 \beta_2 \frac{\gamma^{-l_j} r(x_i)}{\left(1 + \gamma^{-l_j} r(x_i)\right)^2}$$

the derivative of $\tau(x_i)$ can be computed as

$$\frac{d\tau(x_i)}{dx_i} = -2 \sum_{\substack{j=1,2,\ldots J \\ k=1,2,\ldots K}} w_{ijk}^2 \left(y_{ijk} - g_s(x_i + l_j)\right)^2 \frac{dg_s(x_i + l_j)}{dx_i}$$

$$= -2\beta_2 \beta_3 r(x_i) \sum_{\substack{j=1,2,\ldots J \\ k=1,2,\ldots K}} w_{ijk}^2 \frac{\gamma^{-l_j} \left(y_{ijk} - \beta_1 - \beta_2 + \gamma^{-l_j}(y_{ijk} - \beta_1) r(x_i)\right)}{\left(1 + \gamma^{-l_j} r(x_i)\right)^3}$$

After summation, the numerator is a polynomial function of degree $3J - 2$ in the unknown $r(x_i)$ where J is the total dilutions in each sample. A similar algorithm with the polynomial case can solve the roots of this function in order to minimize $\tau(x_i)$.

The methods assume parametric forms of RPPA response curve, which incorporate information from all biological samples. Nonlinear least square methods are employed to estimate protein concentrations, which are the unknown parameters. The dimensionality of the unknown parameter space would be high when the sample size is large, which polemically give rise to the risk of missing the global optimization. In addition, when the response curve does not follow the pattern of polynomial functions or sigmoidal functions, bias may be introduced in the estimates of protein concentrations.

Nonparametric Quantification Method

Hu *et al.* (Hu et al., 2007) proposed a more flexible nonparametric joint sample model for the quantification of RPPA data that could improve estimation when the data does not follow a known response curve. This approach used a nonparametric model of the form:

$$y_{ij} = g(l_j + x_i) + \epsilon_{ij}, \epsilon_{ij} \sim \mathrm{N}(0, \sigma^2)$$

where σ^2 is the variance of ϵ_{ij} y_{ij} is the observed expression level at the jth dilution step of the ith sample where $i = 1,2, \dots I$, $j = 1,2, \dots J$; l_j indicates the corresponding dilution level index at the jth step. $\{x_i, i = 1,2, \dots I\}$ are quantities estimated to represent protein concentrations in each dilution series.

In nonparametric regressions, predictors do not take a predetermined form; rather it is constructed according to information derived from the data. It usually requires larger sample sizes than regression based on parametric models, because the data need to supply the model structure as well as the model estimates. Specifically, Hu *et al.* proposed to use B-splines to estimate $g(x)$.

Algorithm used for estimation has the following steps:

Step 1: Generate initial values of x_i for each biological sample. Assuming the imaging intensities are linearly related to the true protein concentration level,

$$y_{ij} = \alpha + \beta(l_j + x_i) + \epsilon_{ij}$$

where the estimate $\hat{\alpha}$ is the minimum of the intensities of all samples. $\hat{\beta}$ is the median slope over all the dilution series. Then set the initial value of x_i as the median of $\{\frac{y_{ij}-\widehat{\alpha}-\widehat{\beta}x_i}{\widehat{\beta}} : i = 1,2, \ldots I\}$.

Step 2: A qualitatively constrained (regression) smoothing method (X. He & Ng, 1999) is used to obtain a monotonically increasing function g by regressing y_{ij} on $l_j + x_i$, $i = 1,2, \ldots I$, $j = 1,2, \ldots J$.

Step 3: Update $\hat{x}_i$ by minimizing target function $\sum_j \left|y_{ij} - g(x_i + l_j)\right|$ conditional on estimated curve g.

Step 4: The iteration stops when there is no significant improvement of target function. Otherwise go to step (2).

Through data simulation and real data analysis, Hu *et al.* have demonstrated the advantage of the nonparametric quantification method that it reduces the estimation bias due to model misspecification in Tabus' models. However, since the dimension of unknown parameter space in nonparametric methods is usually larger than that in the nonlinear parametric regression method, for accurate parameter estimation they may need a relative larger sample size.

Serial Dilution Curve Method

Zhang *et al*. (2009) (L. Zhang et al., 2009) proposed an alternative approach to RPPA data analysis that models a serial dilution curve instead of a RPPA response curve. Briefly, their method characterized the relationship between signals in successive dilution steps.

Zhang *et al.* points out that the response curve, which is monotonically increasing and s-shaped, is uniquely determined by the relationship between signals in successive dilution steps. The response curve in their method is described as

$$\mathrm{S} = a + \frac{x^{\gamma}}{1 + \frac{x^{\gamma}}{M - a}}$$

which is similar to the sigmoidal curve in Tabus' model with parameter $\{\beta_1, \beta_2, \beta_3\}$:$S$ is the RPPA intensity level; $a = \beta_1$ stands for the background noise; $M = \beta_1 + \beta_2$ is the maximum or saturation level; $\gamma = \beta_3$ controls the steepness of the response curve; x is the protein concentration corresponding to S while the variable in Tabus' model is $\beta_2^{\beta_3}\log(protein\ concentration)$.

After a transformation of the response curve, a function, which is called the serial dilution curve (Figure 2.2), without any parameters of protein concentrations can be obtained as following:

$$S_k = a + \frac{d^{\gamma}(S_{k+1} - \mathrm{a})}{1 + \frac{(d^{\gamma} - 1)(S_{k+1} - a)}{M - a}}$$

where d is the dilution fold. The model displays raw data in an impressive way since the parameters of protein concentrations are cancelled out. Comparing with a general nonlinear model that has much more unknown parameters (three plus the number of protein samples), a serial dilution curve model only has three unknown parameters: a, M and γ, which can be estimated through a weighted non-linear regression model. The weight is set as $\frac{1}{m+|S|}$ where $m = 5$, the minimal error from the RPPA scanner to generate image intensities.

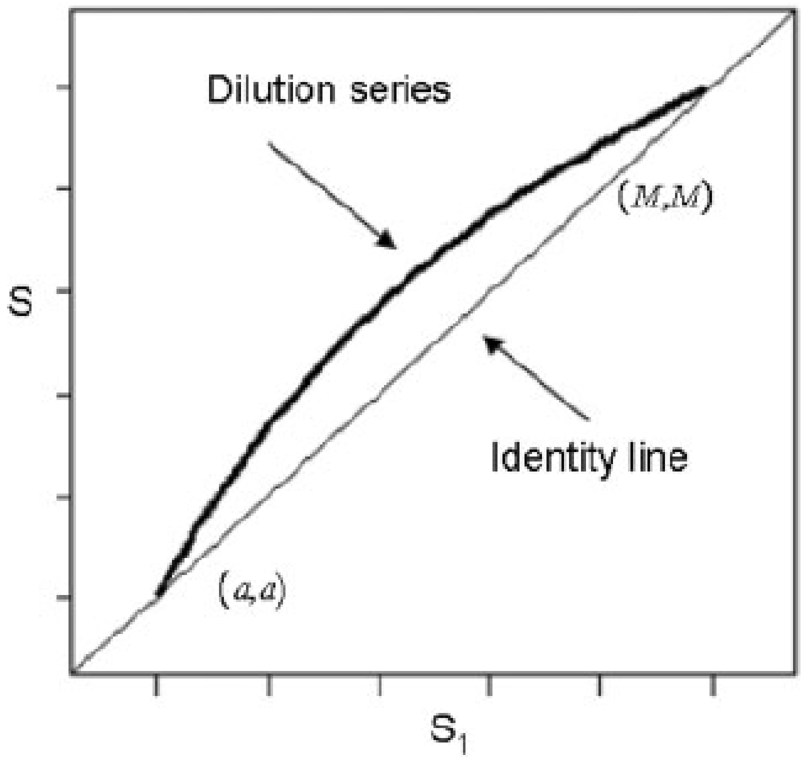


Figure 2.2| A dilution series curve (L. Zhang et al., 2009). In the dilution series curve, the maximal intensity is M and the minimal intensity is a. γ controls the shape of dilution series curve. x-axis is the observational intensity level at the jth dilution step and y-axis is the intensity level from the same biological sample at the $j+1$ dilution step.

Conditioning on the three parameters a, M, γ, protein concentration can be obtained by an algorithm as below:

Firstly, a protein's signals are marked as saturated if its measurements in all serial dilution are greater than a threshold value M/r. The index r that is used to adjust the threshold of saturation is generally greater than 1, and can be reduced if the precision of signals is improved. Besides, if all the signals from one biological sample except one are greater than M/r, and it is not the lowest dilution signal, then the protein concentration in that sample is also marked as saturated. In additional, if all the signals are below ar, its protein concentration is marked to be undetected.

Similar to the saturation cases, if all of them except one are less than ar, and the exception is not the original one, which is not diluted, the protein concentration is also marked to be undetected.

Secondly, if the protein concentration from the ith sample, denoted as x_i, is not marked as saturated or undetected, estimate x_{ij} the protein concentration of sample i at the jth dilution step using the following formula:

$$x_{ij} = d^j \left[\frac{1}{S_j - a} - \frac{1}{M - a}\right]^{-1/\gamma}$$

An outliers among $x_{ij}, j = 1,2, \dots J$, is defined as

$$\left|x_{ij} - median\left(\{x_{i1}, x_{i2}, \dots x_{iJ}\}\right)\right| >$$
$$3 * median(\left|x_{ij} - median\left(\{x_{i1}, x_{i2}, \dots x_{iJ}\}\right)\right|, j = 1,2, \dots J)$$

Thirdly, the protein concentration of biological sample i is obtained from weighted average of $\{x_{ij}; j = 1,2, \dots J\}$:

$$\widehat{x_i} = \frac{\sum_{j=2}^{J}\left(x_{ij} w_{ij}\right)}{\sum_{j=1}^{J} w_{ij}}$$

where

$$w_j = \frac{1}{\left(\frac{\partial x_{ij}}{\partial a}\Delta a\right)^2 + \left(\frac{\partial x_{ij}}{\partial M}\Delta M\right)^2 + \left(\frac{\partial x_{ij}}{\partial \gamma}\Delta \gamma\right)^2}$$

and the partial derivatives are derived and computed according to equation

$$x_{ij} = d^j \left(\frac{1}{S_{ij} - a} - \frac{1}{M - a}\right)^{-1/\gamma}$$

In this method, samples that are not marked as saturated and undetected are related to each other in an explicit formula, which does not contain parameters for unknown protein concentrations. By solving a low-dimensional nonlinear optimization problem, protein concentrations in different biological samples can be estimated based on parameters in the dilution series curve and signal intensities. Moreover, data quality is easier to check by displaying raw data in this way, and model can be interpreted intuitively. However, the measurement error structure in RPPA is difficult to be incorporated into the model.

Multistep Protein Lysate Array Quantification Method

The multistep protein lysate array (PLA) quantification method was proposed by Yang *et al.* (Yang & He, 2011). Similar to Tobus *et al.* (2005), this method also used a sigmoidal model for the relationship between the intensity level and the protein concentration level:

$$y_{ijk} = \beta_1 + \frac{\beta_2}{1 + e^{-\beta_3(x_i + l_j)}} + \epsilon_{ijk}, \epsilon_{ijl} \sim N(0, \sigma^2)$$

where $i = 1,2 \ldots I, j = 1,2 \ldots, J, k = 1, \ldots, K$ And y_{ijk} is the gray-level intensity from the kth replicates from ith biological sample in jth dilution, and x_i is the binary logarithm of the median effective protein concentration level (EC$_{50}$). $x_{ij} = x_i + l_j$ is the binary logarithm of the protein concentration in the jth dilution step where $l_j = \frac{(1+J)}{2} - j$ for 2-fold dilution cases. $\boldsymbol{\beta} = \{\beta_1, \beta_2, \beta_3\}$ is a vector of parameters in the sigmoidal curve, whose properties have been discussed in the nonlinear parametric regression method.

The multistep quantification method is based on a nonlinear regression framework and uses a modified multistep model fitting procedure with two components: a divide-and-conquer component and a pooling component. There are 4 steps in this procedure:

Firstly, I biological samples are divided into $[I/r]$ (the integer part of I/r) groups, ordered increasingly by the median intensity values from the same biological sample. r is a small value, and based on author's experience, $r = 3\ or\ 4$ works reasonably well. For simplification, we assume I is divisible by r.

Secondly, $\{\beta_1, \beta_2, \beta_3\}$ in sigmoidal function are estimated separately in each group. $\hat{\beta}^m$ and $\hat{x}^m$ can be obtained by minimize:

$$\sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{k=1}^{K}\left(y_{ijk}^{(m)} - \beta_1^{(m)} - \frac{\beta_2^{(m)}}{1 + e^{-\beta_3^{(m)}\left(x_i^{(m)} + l_j\right)}}\right)^2$$

The curve parameters and the concentration levels are denoted as: $\boldsymbol{\beta}^{(m)} = (\beta_1^{(m)}, \beta_2^{(m)}, \beta_3^{(m)})'$ and $\boldsymbol{x}^{(m)} = (x_1^{(m)}, \dots, x_r^{(m)})'$, respectively, in the mth group. $m = 1, \dots, I/r$. $y_{ijk}^{(m)}$s are the observed image intensity level of the ith sample in the mth group.

Thirdly, a pooled estimate $\widehat{\boldsymbol{\beta}}^{(c)}$ is obtained through $\widehat{\boldsymbol{\beta}}^{(c)} = \sum_{m=1}^{r} V^{(m)}\,\widehat{\boldsymbol{\beta}}^{(m)}$. $V^{(m)}$s are weight matrixes, and two kinds of weight matrices, M-T and M-C, were discussed by Yang in their paper.

Fourthly, conditioning on $\hat{\beta}^{(c)}$, the concentration estimate $\hat{x}_i$ of biological sample i can be obtained by minimizing:

$$\sum_{k=1}^{K}\sum_{j=1}^{J}\left(y_{ijk}^{(m)}-\beta_1^{(c)}-\frac{\beta_2^{(c)}}{1+e^{-\beta_3^{(c)}(x_i+l_j)}}\right)^2$$

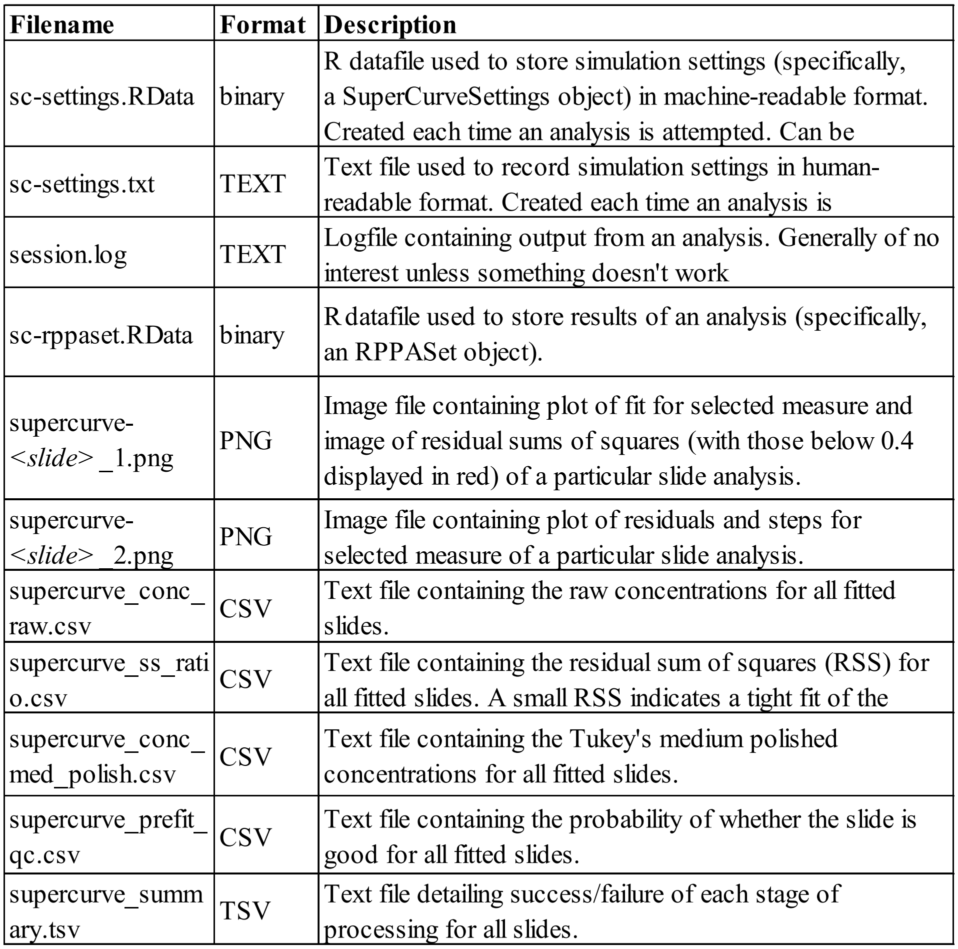
Depending on different weight matrices in the third step, their method can be divided into two subtypes. The first subtype is M-T (T stands for trace), employing a trace minimization criterion in which the trace of $var(\widehat{\boldsymbol{\beta}}^{(c)})$ is minimized. The second subtype is M-C (C stands for component), employing component-wise minimization criterion which treats the variance of each parameter $\beta^{(c)}$ independently. The results by using those two weight matrixes are found to be quite similar. The authors carried out simulation studies to evaluate the performance of the proposed multistep procedures, M-C and M-T, depending on the weight matrix used, in comparison with the least squares procedure in Tabus *et al.* (2006)'s paper (called M-S). They showed that the estimated relative concentration levels from M-C and M-T models had smaller differences to the relative real concentration level. And the estimated parameters $\boldsymbol{\beta}$ had less mean square error (MSE) in M-C and M-T model than M-S model.

Protein concentration estimates resulting from this model have been proved to be consistent and have the asymptotic normality property in Yang's study. This modified parameter estimation procedure is more stable in terms of numerical calculation, and also more robust in practice in terms of less restrictions on RPPA intensities before model fitting, compared to the nonlinear least square methods proposed by Tabus *et al.* in 2006.

2.2.3. An Example

Here we use the RPPA file 14-3-3_epsilon-M-C_GBL9017330 (Figure 1.5) from TCGA database as an example to demonstrate how to estimate protein concentrations. An R package called *SuperCurve* can be used for quantifying protein expression level through the RPPA raw data, which are image intensity file preprocessed by the software MicroVigene. The nonlinear parametric method and nonparametric regression method are demonstrated. Output files from R package *SuperCurve* include raw concentration of fitted slides and residual sum of square (Table 2.1).

Table 2.1| Outputs from *SuperCurve* Package (SuperCurveGUI, 2011).



Filename	Format	Description
sc-settings.RData	binary	R datafile used to store simulation settings (specifically, a SuperCurveSettings object) in machine-readable format. Created each time an analysis is attempted. Can be
sc-settings.txt	TEXT	Text file used to record simulation settings in human-readable format. Created each time an analysis is
session.log	TEXT	Logfile containing output from an analysis. Generally of no interest unless something doesn't work
sc-rppaset.RData	binary	R datafile used to store results of an analysis (specifically, an RPPASet object).
supercurve-<*slide*> _1.png	PNG	Image file containing plot of fit for selected measure and image of residual sums of squares (with those below 0.4 displayed in red) of a particular slide analysis.
supercurve-<*slide*> _2.png	PNG	Image file containing plot of residuals and steps for selected measure of a particular slide analysis.
supercurve_conc_raw.csv	CSV	Text file containing the raw concentrations for all fitted slides.
supercurve_ss_ratio.csv	CSV	Text file containing the residual sum of squares (RSS) for all fitted slides. A small RSS indicates a tight fit of the
supercurve_conc_med_polish.csv	CSV	Text file containing the Tukey's medium polished concentrations for all fitted slides.
supercurve_prefit_qc.csv	CSV	Text file containing the probability of whether the slide is good for all fitted slides.
supercurve_summary.tsv	TSV	Text file detailing success/failure of each stage of processing for all slides.

Figure 2.3 is a plot for intensity vs. dilution step. We can easily tell that the effect of dilution steps to intensities is not linear. Through Figure 2.3 and Figure 2.4, we compare the results of using nonlinear regression for sigmoidal (Tabus et al., 2006) model and nonparametric quantification methods (Hu et al., 2007). The results from these two methods are very similar and the data appears to follow the estimated curves quite well (intensity as y-axis and log concentration

as x-axis; upper part of Figure 2.4). On the other side, larger residuals were obtained in nonlinear parametric method than those in nonparametric regression method (lower part of Figure 2.4).

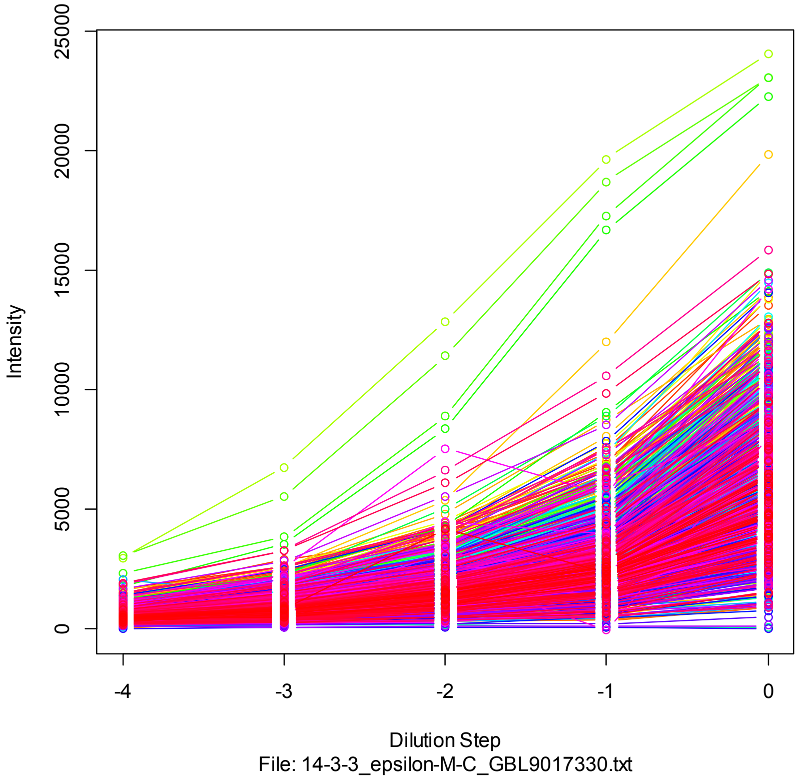
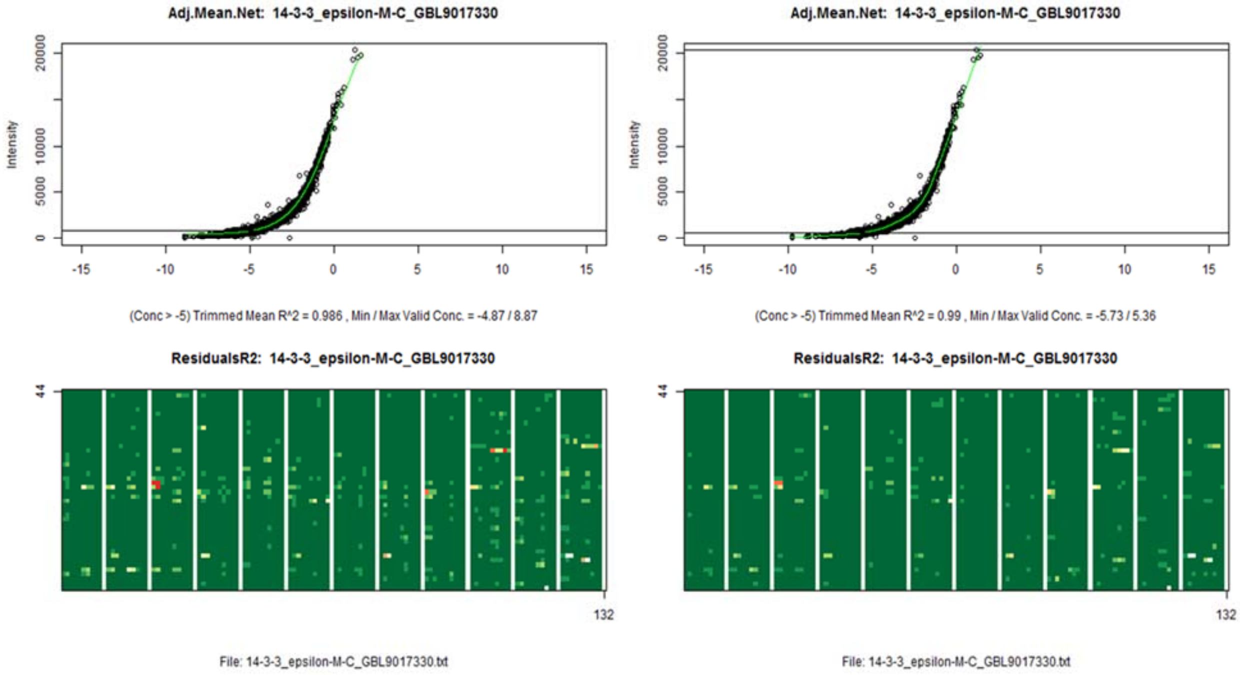


Figure 2.3| The intensity vs. dilution step plot. There are 5 dilutions for each sample here. We can easily see that the relationship between dilution steps and intensity is not linear.



a. non-linear method

b. nonparametric method

Figure 2.4| Plot of residuals and model fitting for selected measure of a particular slide; analyses used (a) nonlinear model assuming a sigmoidal curve and (b) nonparametric model with constrained B-splines. In the lower part images, light color means something odd may have happened and we need to pay special attention to the results from this patch. Trimmed mean $\mathbf{R^2}$ represents goodness of fit on *SuperCurve* at the point beyond the indicated concentration (−5). Upper and lower lines represent cutoff levels for upper and lower limits of signal reflecting minimum/maximum valid concentration.

2.3. A Naïve Model for Detecting MiRNA Target Proteins

Once the protein concentrations and miRNA levels have been estimated, an intuitive way to screen miRNA target genes is to search for proteins whose expressions have significant negative correlations with the miRNA. We call this approach as the naïve method, as it is simple and straightforward. The linear relationship between a protein and miRNA in the naïve model can be expressed as: $x_i = f(z_i) = \alpha_1 + \alpha_2 z_i + \eta_i$, where $\eta_i \sim N(0, \sigma_0^2)$ and $\{z_i,\ i = 1,2 \dots I\}$ are log-transformed expression levels of a specific miRNA from sample $i = 1,2 \dots I$.

To quantify the protein expression on a RPPA array, we utilized a sigmoidal model, which is commonly assumed to describe the relationship between the intensity level and the protein concentration as Gelman et al. (2004), Tubas et al. (2006) and Yang et al. (2011):

$$y_{ij} = g\left(x_i, l_j, \boldsymbol{\beta}\right) + \varepsilon_{ij} = \beta_1 + \frac{\beta_2}{1 + 2^{-\beta_3(x_i + l_j)}} + \varepsilon_{ij},$$

where y_{ij} is the gray-level intensity from sample i at jth dilution, $i = 1, \dots, I$ and $j = 1, \dots, J$, x_i is the binary logarithm of the median effective protein concentration level, EC_{50}; $x_i + l_j$ is the binary logarithm of the protein concentration after jth dilution where $l_j = \frac{(1+J)}{2} - j$, ε_{ij} is the error term assumed to have a normal distribution with mean 0 and variance σ_1^2, and $\boldsymbol{\beta} = \{\beta_1, \beta_2, \beta_3\}$. Since $\beta_1 = \lim_{x_i \to -\infty} E\left(y_{ij}\right)$ and $\beta_1 + \beta_2 = \lim_{x_i \to +\infty} E\left(y_{ij}\right)$, β_1 is interpreted as the lowest intensity level without noise, and β_2 is the increment from the lowest to the highest intensity or the saturation level.

By using EC_{50}s estimated from the sigmoidal function and the miRNA data, the parameter estimates $\hat{\alpha}_1, \hat{\alpha}_2, \hat{\sigma}_0$ can be calculated based on simple linear regression. $\hat{\alpha}_2$ is our parameter of interest, describing the relationship between a miRNA and protein pair. Hypothesis test $H_0 \colon \alpha_2 = 0$ vs $H_1 \colon \alpha_2 \leq 0$ can be conducted to determine if a particular pair of miRNA and protein is related or not.

2.4. Discussion

Owing to the newly developed RPPA technique, protein concentrations can be measured in a fast and accurate way. In the case that the correlation between a pair of miRNA and protein is strong, the naïve model serves as a good solution to study the protein-miRNA relationship. The biggest advantage of the naïve model is simple and easy to be understood. The computational burden of the naïve model is also very small (usually less than 1min/case for sample size 300 in Intel® Core™ i7-2600 CPU @ 34.0GHz). However, the naïve model ignores the variations associated with the estimates of protein concentration, which could result in significant power loss. This disadvantage of naïve method will be further illustrated from simulation studies and real data analyses in later chapters.

Chapter 3. A Parametric Integrated Method for Detecting MicroRNA Target Proteins

3.1. Motivation

The naïve model identifies potential miRNA targets by searching evidences for significant negative correlations between estimated protein expression levels and miRNA levels. The uncertainty associated with protein expression level estimates is ignored in this method, thus it may lead to less accurate findings and significant power loss. That is, such model may miss certain number of miRNA targets. Hence, developing a model with higher detection power is imperative.

Since we are more interested in the miRNA/protein relationship instead of the absolute magnitude of protein concentration, in this sense, results from protein expression estimation are not that important to us. A hierarchical model that treats protein concentrations as latent variables may improve the detection power by avoiding a direct estimation of protein levels. Such a model is named as an *integrated model* thereafter. In this Chapter, we will propose a parametric integrated model (PIM) assuming a sigmoidal RPPA response curve.

3.2. Statistical Model

Similar to the naïve model, for simplicity, a sigmoidal model is used to describe the relationship between the imaging intensity levels and the protein concentrations in RPPA data with additive error:

$$y_{ij} = g\left(x_i, l_j, \boldsymbol{\beta}\right) = \beta_1 + \frac{\beta_2}{1 + 2^{-\beta_3(x_i + l_j)}} + \varepsilon_{ij},$$

where y_{ij} is the gray-level intensity from sample i at jth dilution, $i = 1, \dots, I$ and $j = 1, \dots, J$, x_i is the binary logarithm of the median effective protein concentration level (EC_{50}) which represents the concentration of the protein, $x_i + l_j$ is the binary logarithm of the protein concentration after jth dilution, $l_j = \frac{(1+J)}{2} - j$ is used and ε_{ij} is the error term assumed to have a normal distribution with mean 0 and variance σ_1^2. $\boldsymbol{\beta} = \{\beta_1, \beta_2, \beta_3\}$. To easily illustrate our model, the dilution number J was chosen as a fixed number, 5, and no technical replicates are in RPPA in the rest of the article.

Our proposed hierarchical model directly models the relationship between miRNA and protein signals from RPPA without estimating protein concentration. A general model is given as follows:

$$\begin{cases} \boldsymbol{y_i} = \boldsymbol{g}(x_i, \boldsymbol{\beta}) + \boldsymbol{\varepsilon_i} = \boldsymbol{\beta_1} + \begin{pmatrix} \frac{\beta_2}{1 + 2^{-\beta_3(x_i - 2)}} \\ \frac{\beta_2}{1 + 2^{-\beta_3(x_i - 1)}} \\ \frac{\beta_2}{1 + 2^{-\beta_3(x_i)}} \\ \frac{\beta_2}{1 + 2^{-\beta_3(x_i + 1)}} \\ \frac{\beta_2}{1 + 2^{-\beta_3(x_i + 2)}} \end{pmatrix} + \boldsymbol{\varepsilon_i} \\ x_i = f(z_i) + \eta_i \end{cases}$$

$$i = 1,2, \dots I, \eta_i \sim N(0, \sigma_0^2), \boldsymbol{\varepsilon_i} \sim N(\boldsymbol{0}, \Sigma_1^2)$$

Here $f(.)$ is a general function to describe how x_i, the protein concentration level, and z_i, the miRNA expression level, are related. $\boldsymbol{\beta} = \{\beta_1, \beta_2, \beta_3\}$ is the parameter vector for the response curve function $\boldsymbol{g}(.)$. η_i is a random error term in miRNA regulation and $\boldsymbol{\varepsilon_i}$ is a measurement error vector of image intensity. In this hierarchical framework, the relationship between miRNA and protein expression levels will be estimated without explicitly quantifying the protein concentration levels based on intensity data first.

To directly compare with the naïve model, we assume $f(z_i)$ to be linear, that is, $f(z_i) = \alpha_1 + \alpha_2 z_i$. We further assume that the two error terms, η_i and $\boldsymbol{\varepsilon_i}$ are independent of each other, and intensity levels from one subject are independent to each other, that is, $\Sigma_1^2 = \sigma^2 I$ as in(Hu et al., 2007; Tabus et al., 2006; Yang & He, 2011). A simplified version of the integrated model is showed below:

$$y_{ij} = g\left(x_i, l_j, \boldsymbol{\beta}\right) = \beta_1 + \frac{\beta_2}{1 + 2^{-\beta_3(x_i - l_j)}} + \varepsilon_{ij},$$

$$x_i = f(z_i) = \alpha_1 + \alpha_2 z_i + \eta_i,$$

$$\eta_i \sim N(0, \sigma_0^2), \varepsilon_{ij} \sim N(0, \sigma_1^2), \eta_{i'} \perp \varepsilon_{ij}$$

$$i, i' = 1,2, \dots, I \; and \; j = 1,2, \dots, 5$$

This hierarchical model is not a traditional mixed model since random effects appear on the nonlinear part of the function. Thus a different strategy is needed to fit such a model. In the simplified setting, the likelihood function for $\boldsymbol{Y}$ and $\boldsymbol{Z}$ can be written as a joint probability function of and $\boldsymbol{Z}$ **:**

$$L(\boldsymbol{\phi}, \sigma_0, \sigma_1 | \boldsymbol{Y}, \boldsymbol{Z}) = \prod_{\substack{i=1,2\dots I \\ j=1,2,..J}} \int_{-\infty}^{+\infty} f_{\epsilon_{ij}}\left(y_{ij} \middle| z_i, \boldsymbol{\phi}, \eta_i, \sigma_1\right) q_{\eta_i}(\eta_i | \sigma_0) d\eta_i$$

where $\boldsymbol{\phi} = \{\alpha_1, \alpha_2, \beta_1, \beta_2, \beta_3\}$ is a vector including parameters in function $f(.)$ and $g(.)$, $\boldsymbol{Y} = \{y_{ij} \mid i = 1,2 \dots I, j = 1,2 \dots 5\}$ represents the RPPA imaging intensity levels, and $\boldsymbol{Z} = \{z_i | i\ 1,2 \dots I\}$ represents the normalized variable--log-transformed miRNA expression levels.

3.3. Computational Algorithm

Since the likelihood function of the model can be explicitly written, the unknown parameters $\boldsymbol{\phi} = \{\alpha_1, \alpha_2, \beta_1, \beta_2, \beta_3\}$ can be estimated within the maximum likelihood framework. To get the MLE, a typical way is to do an integral approximation first and then to maximize the function after integration. Two methods are commonly used in integral approximation: the first order method and the adaptive Gaussian quadrature method. Several numerical algorithms can be further applied in maximizing the likelihood function. Those methods will be introduced in this section.

3.3.1. Integral approximation

The First order method

The first order method is based on Taylor expansion with the first order (S. Beal & Sheiner, 1988; S. L. Beal & Sheiner, 1981; Sheiner & Beal, 1985). The equation of Taylor expansion at number zero is as following:

$$f(x) = f(0) + \frac{f'(0)}{1!}x + \frac{f''(0)}{2!}x^2 + \frac{f^{(3)}(0)}{3!}x^3 + o(x^3) = \sum_{n=0}^{\infty} \frac{f^{(n)}(0)}{n!}x^n$$

In our case, the probability density function of $\boldsymbol{\epsilon_i} = \{\epsilon_{ij} | j = 1,2, ..5\}$ of the ith sample is

$f_{\boldsymbol{\epsilon_i}}(\boldsymbol{y_i}|z_i, \boldsymbol{\phi}, \eta_i, \sigma_1)$

$$= \left(\frac{1}{\sqrt{2\pi}}\right)^5 |R(z_i, \boldsymbol{\phi}, \sigma_1)|^{-\frac{1}{2}} \exp\{-\left(\frac{1}{2}\right)[\boldsymbol{y}_i - m(z_i, \boldsymbol{\phi}, \eta_i)]^T R(z_i, \boldsymbol{\phi}, \sigma_1)^{-1}[y - m(z_i, \boldsymbol{\beta}, \eta_i)]\}$$

where the 5x1 location vector is

$$m(z_i, \boldsymbol{\beta}, \eta_i) = \{\beta_1 + \frac{\beta_2}{\left(1 + 2^{-\beta_3(l_j + \alpha_1 + \alpha_2 x_i + \eta_i\ )}\right)} |j = 1,2,..5\}$$

and the 5x5 covariance matrix $R(x_i, \boldsymbol{\phi}, \sigma_1) = \sigma_1^2 I$

Thus,

$f(\boldsymbol{y_i}|z_i, \boldsymbol{\phi}, \eta_i, \sigma_1)$

$$\approx \left(\frac{1}{\sqrt{2\pi}}\right)^5 |\sigma_1^2 I|^{-\frac{1}{2}} \exp\{-\left(\frac{1}{2}\right)[\boldsymbol{y}_i - m(z_i, \boldsymbol{\phi}, 0) - Z(z_i, \boldsymbol{\phi},)\eta_i]^T (\sigma_1^2 I)^{-1}[y - m(z_i, \boldsymbol{\phi}, 0) - Z(z_i, \boldsymbol{\phi},)\eta_i]\}$$

where $Z(z_i, \boldsymbol{\phi})$ is the first derivative:

$$\frac{\partial m(z_i, \boldsymbol{\phi}, \eta_i)}{\partial \eta_i}|_{\eta_i = 0}$$

Therefore

$$Z(z_i, \boldsymbol{\phi}) = \begin{pmatrix} \dfrac{ln2 * \beta_2\beta_3 2^{\left(-\beta_3(-2+\alpha_1+\alpha_2 z_i\ )\right)}}{\left(1+2^{\left(-\beta_3(-2+\alpha_1+\alpha_2 z_i\ )\right)}\right)^2} \\ \dfrac{ln2 * \beta_2\beta_3 2^{\left(-\beta_3(-1+\alpha_1+\alpha_2 z_i\ )\right)}}{\left(1+2^{\left(-\beta_3(-1+\alpha_1+\alpha_2 z_i\ )\right)}\right)^2} \\ \dfrac{ln2 * \beta_2\beta_3 2^{\left(-\beta_3(\alpha_1+\alpha_2 z_i\ )\right)}}{\left(1+2^{\left(-\beta_3(\alpha_1+\alpha_2 z_i\ )\right)}\right)^2} \\ \dfrac{ln2 * \beta_2\beta_3 2^{\left(-\beta_3(1+\alpha_1+\alpha_2 z_i\ )\right)}}{\left(1+2^{\left(-\beta_3(1+\alpha_1+\alpha_2 z_i\ )\right)}\right)^2} \\ \dfrac{ln2 * \beta_2\beta_3 2^{\left(-\beta_3(2+\alpha_1+\alpha_2 z_i\ )\right)}}{\left(1+2^{\left(-\beta_3(2+\alpha_1+\alpha_2 z_i)\right)}\right)^2} \end{pmatrix}$$

Assuming that $q(\eta_i|\sigma_0)$ is a normal density function with mean 0 and variance σ_0^2, the first order integral approximation is computable in a closed form after completing the square:

$$\int_{-\infty}^{+\infty} f_{\epsilon_{ij}}(\boldsymbol{y_i}|z_i, \boldsymbol{\phi}, \eta_i, \sigma_1) q_{\eta_i}(\eta_i|\sigma_0) d\eta_i$$

$$\approx \int \left(\frac{1}{\sqrt{2\pi}}\right)^5 |R(z_i, \boldsymbol{\phi}, \sigma_1)|^{-\frac{1}{2}} \exp\left\{-\left(\frac{1}{2}\right)[\boldsymbol{y}_i - m(z_i, \boldsymbol{\phi}, 0)\right.$$

$$\left. - Z(z_i, \boldsymbol{\phi},)\eta_i]^T R(z_i, \boldsymbol{\phi}, \sigma_1)^{-1}[y - m(z_i, \boldsymbol{\phi}, 0) - Z(z_i, \boldsymbol{\phi},)\eta_i]\right\}$$

$$* (2\pi\sigma_0)^{-\frac{1}{2}} \exp\left\{-\frac{\eta_i^2}{2\sigma_0^2}\right\} d\eta_i$$

$$= \left(\frac{1}{\sqrt{2\pi}}\right)^5 |R(z_i, \boldsymbol{\phi}, \sigma_1)|^{-\frac{1}{2}} (2\pi\sigma_0)^{-\frac{1}{2}} \int \exp\left\{-\left(\frac{1}{2}\right)[y - m(z_i, \boldsymbol{\phi}, 0)\right.$$

$$\left. - Z(z_i, \boldsymbol{\phi})\eta_i]^T R(z_i, \boldsymbol{\phi}, \sigma_1)^{-1}[y - m(z_i, \boldsymbol{\phi}, 0) - Z(z_i, \boldsymbol{\phi})\eta_i] - \frac{\eta_i^2}{2\sigma_0^2}\right\} d\eta_i$$

$$\approx (2\pi)^{-\frac{5}{2}} |V(z_i, \boldsymbol{\phi}, \sigma_0)|^{-\frac{1}{2}} \exp\{-\left(\frac{1}{2}\right)[\boldsymbol{y_i} - m(z_i, \boldsymbol{\phi}, 0)]^T V(z_i, \boldsymbol{\phi}, \sigma_0)^{-1}[\boldsymbol{y_i} - m(z_i, \boldsymbol{\phi}, 0)]\}$$

where $(z_i, \boldsymbol{\phi}, \sigma_0) = \sigma_0^2 Z(z_i, \boldsymbol{\phi}) Z(z_i, \boldsymbol{\phi})^T + R(z_i, \boldsymbol{\phi}, \sigma_1)$.

Thus the likelihood function of the PIM can be approximated as

$L(\boldsymbol{\phi}, \sigma_0, \sigma_1 | \boldsymbol{Y}, \boldsymbol{Z}) =$

$$\prod_{\substack{i=1,2\ldots I \\ j=1,2,..J}} (2\pi)^{-\frac{5}{2}} |V(z_i, \phi, \sigma_0)|^{-\frac{1}{2}} \exp\{-\left(\frac{1}{2}\right) [\boldsymbol{y_i} - m(x_i, \phi, 0)]^T V(x_i, \phi, \sigma_0)^{-1} [\boldsymbol{y_i} - m(x_i, \phi, 0)]\}$$

Since the Tayler expansion is expanded at the point of zero, the integration approximation may not be precise when η_is are far away from zero. In our simulation studies, the type-I error from using the first order method is much higher than the pre-specified significant level. The imprecise integration approximation by using the first order method led to inflated type-I error in our case (Figure 3.2).

The adaptive Gaussian quadrature (AGQ) method

The Gaussian quadrature method is based on the idea of Hermite integration for function$f(x)$:

$$\int_{-\infty}^{+\infty} f(x) dx = \sum_{k=1}^{m} w_k e^{x_k^2} f(x_k)$$

According to it, our likelihood function can be rewritten as

$\boldsymbol{\mathcal{L}}(\boldsymbol{\phi}, \sigma_0, \sigma_1 | \boldsymbol{Y}, \boldsymbol{Z}) =$

$$\prod_{\substack{i=1,2,\ldots I\\ j=1,2,\ldots 5}} \int_{-\infty}^{+\infty} f_{\epsilon_{ij}}\left(y_{ij}\middle|z_i, \boldsymbol{\phi}, \eta_i, \sigma_1\right) q_{\eta_i}(\eta_i|\sigma_0)\, d\eta_i$$

$$\approx \prod_{\substack{i=1,2,\ldots I\\ j=1,2,\ldots 5}} \sqrt{2}|\Gamma_i|^{-\frac{1}{2}} \sum_{k=1}^{m} f_{\epsilon_{ij}}\left(y_{ij}\middle|z_i, \boldsymbol{\phi}, \sqrt{2}\Gamma^{-\frac{1}{2}}x_k + \hat{\eta}_i, \sigma_1\right) q_{\eta_i}\left(\sqrt{2}\Gamma^{-\frac{1}{2}}x_k + \hat{\eta}_i\middle|\sigma_0\right) w_k e^{x_k^2}$$

m is the number of quadrature points which is set to 5 in our analysis. x_k and w_k denote the standard Gauss-Hermite abscissas and weights; $\hat{\eta}_i$ minimizes

$$-\log\left(f_{\epsilon_{ij}}\left(y_{ij}\middle|z_i, \boldsymbol{\phi}, \eta_i, \sigma_1\right) q_{\eta_i}(\eta_i|\sigma_0)\right)$$

and Γ_i is the Hessian matrix (a matrix of second derivatives of the log-likelihood function respect to the parameters) from the minimization.

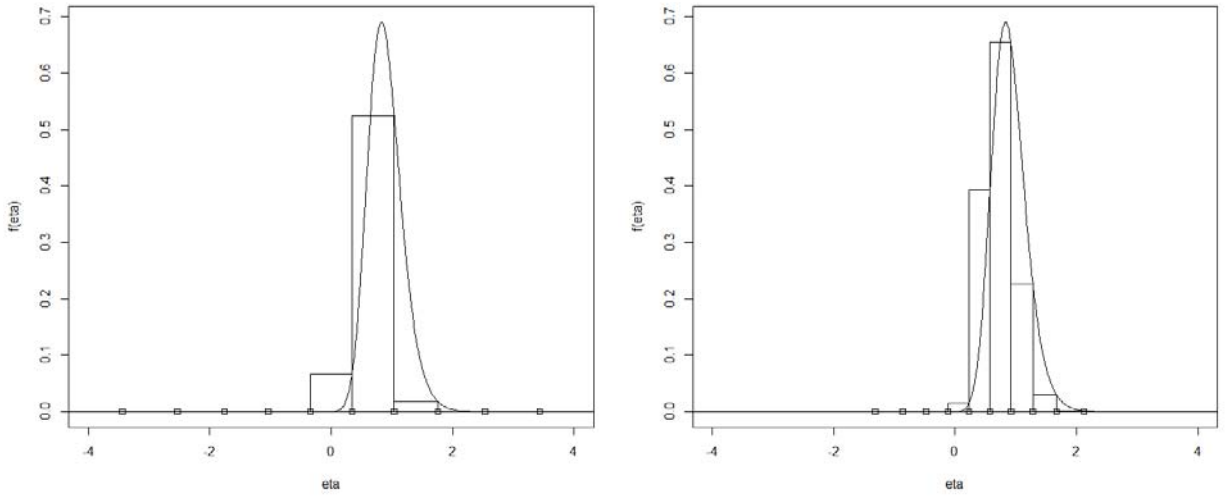


Figure 3.1| An illustration of general Gaussian quadrature method and adaptive Gaussian quadrature method with 10 knots. (Left) Gaussian quadrature method (Right) adaptive Gaussian quadrature method. The adaptive Gaussian quadrature method finds the main part of the function and sets most of the knots in that area which makes it more efficient than the general Gaussian quadrature method under same number of knots.

By calculating $\hat{\eta}_i$ and Γ_i, the adaptive Gaussian quadrature method adjusts knots to locate on the x-axis corresponding to the main part of the function which needs integration. It is more efficient than the general Gaussian quadrature method under same number of knots (Figure 3.1). Comparing with the First order method, adaptive Gaussian quadrature method can better estimate the integral function and further better control the type-I error (Figure 3.2).

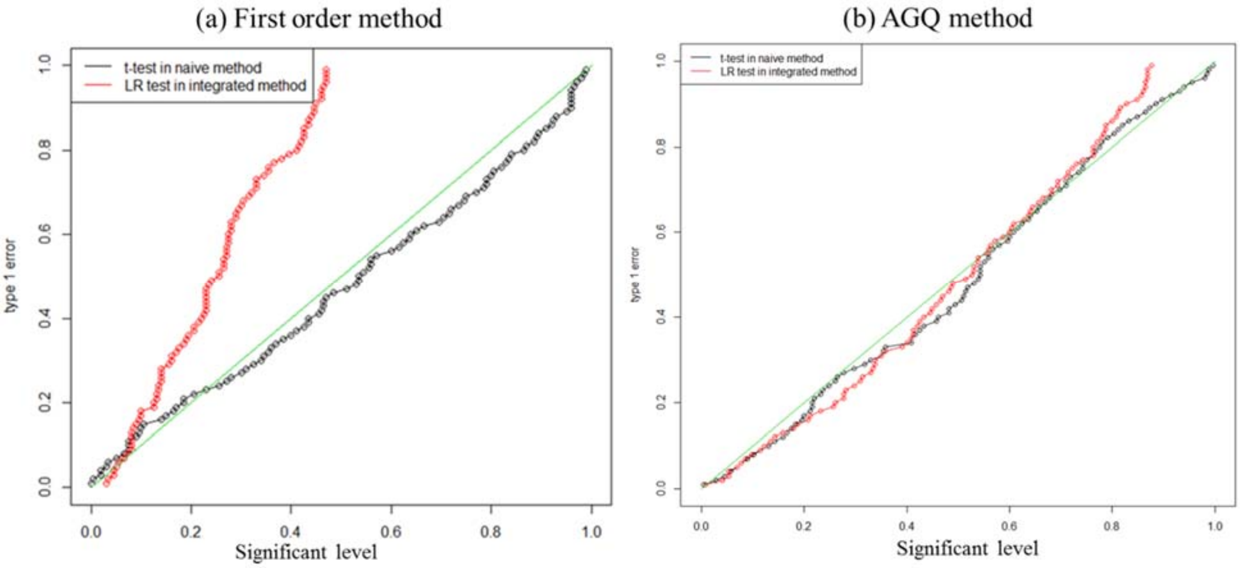


Figure 3.2| A comparison of estimated and pre-specified type I error. (a) Type I error in model using the first order method (b) Type-I error in model using adaptive Gaussian Quadrature (AGQ) method. The AGQ method can better estimate the integral and further better control the type-I error.

3.3.2. Likelihood Function Maximization

Nelder-Mead method

Without using any derivatives and assuming that the objective function has continuous derivatives, the Nelder-Mead method, proposed by John Nelder and Roger Mead (Nelder & Mead, 1965), uses a special polytope of $N + 1$ vertices in N-dimension called simplex. For instance, to minimize a function with two unknown parameters, a simplex is a triangle and the method compares the function values at the three vectors of the triangle and replaces the vector which corresponding to the highest function value to a new vector by using reflection, expansion, contraction and shrinkage. When the triangle is close to the optimal point, it will shrink to the optimal point. The iterative process stops for convergence when the difference between the best function value in the new simplex and old simplex is less than a tolerance threshold.

This method usually gives rather big improvements in the first few iteration steps. Also, this method requires much lower number of function evaluations and does not use any derivatives of the objective function which makes it appealing in cases of very complex objective functions. However, this method is lack of convergence theory and with less advantage when optimizing a function in a lower dimension space.

Newton-Raphson method

The Newton-Raphson method is one of the most famous optimization methods. It is a second-derivative method which derives from the Taylor expansion and uses the gradient and Hessian matrix. The new improved estimate of unknown parameter vector in function $f(.)$ on the $i+1$th iteration is given by:

$$\boldsymbol{x}^{(i+1)} = \boldsymbol{x}^{(i)} - f'\left(\boldsymbol{x}^{(i)}\right)/f''\left(\boldsymbol{x}^{(i)}\right)$$

Usually computing the Hessian matrix takes much more time than compute the gradient and the objective function value, especially for functions in high dimensions. However, methods using Hessians matrix usually converge more quickly than methods without using Hessian matrix. Although it can convert to multiple dimensions, Newton-Raphson method is pretty time consuming on calculating Hessian matrix, $H(x)$. Also, the Newton-Raphson method is sensitive to its initial value: if the initial value is not close to the maximal point, the method may converge to a local maximal.

Quasi-Newton methods

Unlike the Newton-Raphson method, the quasi-Newton method does not compute Hessian matrixes $H(x)$ in every iteration steps, instead, it updates them, which makes it a better candidate in our likelihood function maximization problems, even though this method may require more iterations to converge than the Newton-Raphson method. This algorithm suits our problem best in which there are 7 unknown parameters.

The algorithm of quasi-Newton methods to minimize function $f(.)$ is as following:

Step 1: Get initial values of unknown parameters $\boldsymbol{x}^{(0)}$.

Step 2: In the pth iteration, compute the quasi-Newton direction Δx through

$$\Delta x = -{H^{(p-1)}}^{-1} \nabla f(x^{(p-1)})$$

And determine the step size t by line searching methods to satisfy the Goldstein conditions, which tests whether the movement from x to $x + t\Delta x$ achieves a sufficient decrease in function $f(.)$.

Step 3: Update parameters x value using $x^{(p)} = x^{(p-1)} + t\Delta x$.

Step 4: Update Hessian matrix using Broyden–Fletcher–Goldfarb–Shanno (BFGS) approach:

$$H^{(p)} = H^{(p-1)} + \frac{y^{(p-1)} {y^{(p-1)}}'}{{y^{(p-1)}}' s^{(p-1)}} - \frac{H^{(p-1)} s^{(p-1)} {s^{(p-1)}}' H^{(p-1)}}{s^{(p-1)} H^{(p-1)} s^{(p-1)}}$$

where $s^{(p-1)} = t\Delta x$ and $y = \nabla f\left(x^{(p)}\right) - \nabla f(x^{(p-1)})$.

Step 5: Check if there is significant improvement of $(.)$. If not, go to Step 2.

3.3.3. Initial Value Selection

To select initial values for our integrated model, one way is assuming there is no error term in protein-miRNA link function $x_i = f(z_i) + \eta_i = \alpha_1 + \alpha_2 z_i + \eta_i$ and estimating unknown parameters using nonlinear least square method. But this way cannot estimate the variance of the error term η_i in protein-miRNA link function $f(z_i)$. Therefore, another approach is to directly use the parameter estimates from the naïve method as initial values. *SuperCurve* package we introduced in Chapter 2 doesn't provide estimations of known parameters, so we used an algorithm similar to Hu (2007)'s model fitting algorithm (Hu et al., 2007) to estimate the relative protein concentration in RPPA:

The initial intensity data are first transformed as

$$y_{linear_{ij}} = \mathrm{logit}_2 \frac{y_{ij} - \min(Y)}{range(Y)}$$

where $= \{y_{ij} |\ i = 1, \dots, I\ and\ j = 1, \dots, J\}\ and\ range(Y) = \max(Y) - \min(Y),$. Initial values of $\boldsymbol{\beta}$ are set as $Y\ \hat{\beta}_1^{(0)} = \min(Y), \widehat{\beta}_2^{(0)} = max(Y) - \min(Y), \hat{\beta}_3^{(0)} = \hat{\beta}_2^{(0)}/(J-1).$ J is set to 5.

The initial median effective protein concentration level x_i are estimated by using:

$$x_i^{(0)} = median_{j=1,2,..5}\left(\frac{y_{lineari.}}{\hat{\beta}_3^{(0)}} + l_j\right)$$

where $y_{lineari.}$ is the mean value among $\{y_{linearij}\ |j = 1,2, \dots 5\}$.

To update the parameters ($\hat{\beta}_1^{(0)}, \hat{\beta}_2^{(0)}, \hat{\beta}_3^{(0)}$) in the nonlinear model, the nonlinear least-squares estimates of ($\hat{\beta}_1^{(1)}, \hat{\beta}_2^{(1)}, \hat{\beta}_3^{(1)}$) were calculated based on the following model (Golub & Pereyra, 2003):

$$y_{ij} = g\left(x_i, l_j, \boldsymbol{\beta}\right) = \beta_1 + \frac{\beta_2}{1 + 2^{-\beta_3(x_i - l_j)}} + \varepsilon_{ij}.$$

After obtaining $\hat{\beta}_1^{(1)}, \hat{\beta}_2^{(1)}, \hat{\beta}_3^{(1)}$, the nonlinear least-square method is used again to update the relative protein level $\boldsymbol{X} = \{x_i, i = 1 \ldots I\}$ and σ_1^2. And then update $\hat{\beta}_1^{(1)}, \hat{\beta}_2^{(1)}, \hat{\beta}_3^{(1)}$, conditioning on the relative protein level $\boldsymbol{X}$. This iteration continues until convergence.

The initial value of α_1 and α_2 are calculated from linear regression, and the MSE of the linear regression is used as the initial value of σ_0. The initial value of σ_1^2 is from residual sum of square (RSS).

3.3.4. Computational Algorithm

No numerical optimization algorithm can guarantee to find the global maximum. A preliminary grid search step is further used in our model fitting algorithm with hope to find a global maximum. As its name suggests, a grid searching method searches points over a grid. Our algorithm calculates the likelihood function values among points on the grid with initial values provided by the naïve method as its center, and the point with highest likelihood was chosen as the modified initial values for our PIM.

Our computational algorithm is illustrated in Figure 3.3. Our SAS macro using PROC NLMIXED implement the same algorithm and can directly applied to miRNA and RPPA intensity data. The same analysis can be performed in R, a free and widely used software, through our R code. The algorithm stops when the change in the normalized likelihood value is less than 10e-8.

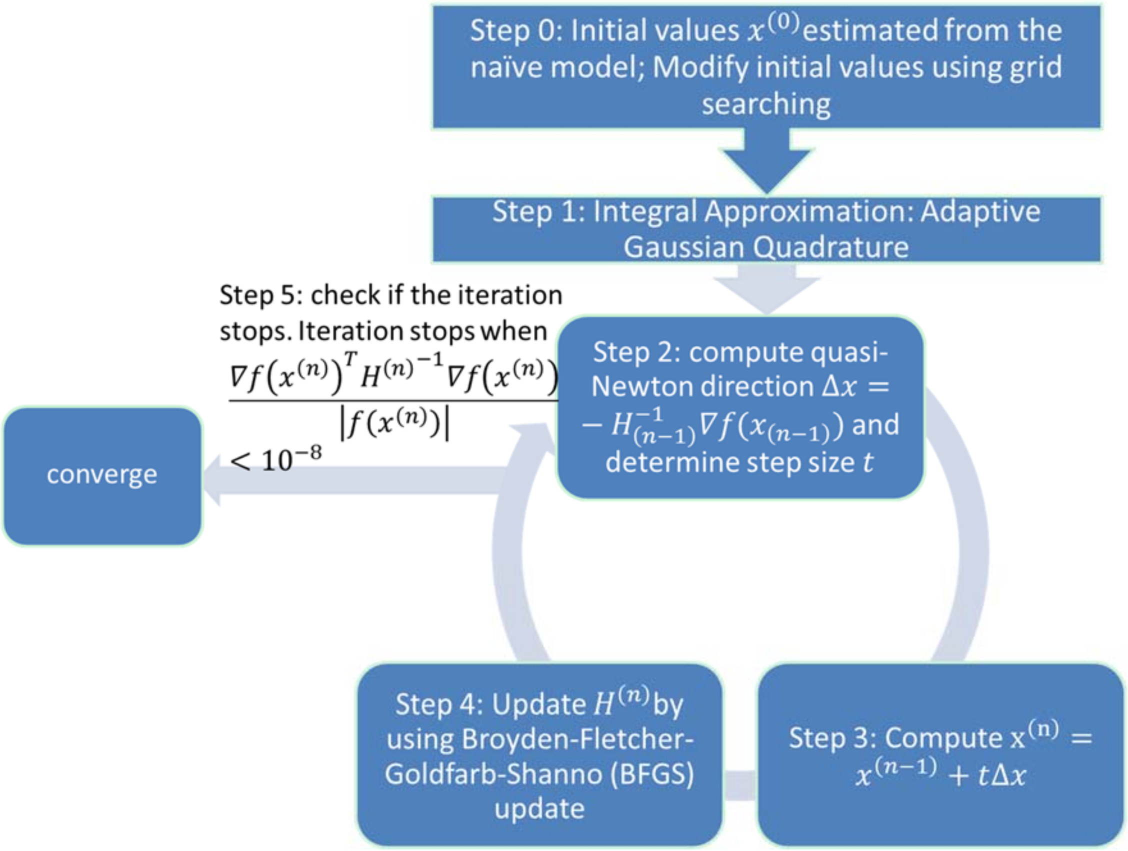

Step 0: Initial values $x^{(0)}$estimated from the naïve model; Modify initial values using grid searching
Step 1: Integral Approximation: Adaptive Gaussian Quadrature
Step 2: compute quasi-Newton direction $\Delta x = -H_{(n-1)}^{-1} \nabla f(x_{(n-1)})$ and determine step size t
Step 3: Compute $\mathrm{x}^{(\mathrm{n})} = x^{(n-1)} + t\Delta x$
Step 4: Update $H^{(n)}$by using Broyden-Fletcher-Goldfarb-Shanno (BFGS) update
Step 5: check if the iteration stops. Iteration stops when $\frac{\nabla f(x^{(n)})^T {H^{(n)}}^{-1} \nabla f(x^{(n)})}{|f(x^{(n)})|} < 10^{-8}$
converge

Figure 3.3| A flow chart of computational algorithm to fit our parametric integrated model based on adaptive Gaussian quadrature method and quasi-Newton algorithm. Step 0: estimate the initial value of $\boldsymbol{\phi}, \boldsymbol{\sigma_0}, \boldsymbol{\sigma_1}$, denoted as $\boldsymbol{x}^{(\mathbf{0})}$, by using the naïve model; a grid searching method was applied; Step 1: generate the approximate likelihood function by using adaptive Gaussian Quadrature method; Step 2: compute the quasi-Newton direction $\Delta\boldsymbol{x}$, determine the step size $\boldsymbol{t}$ to satisfy the Goldstein conditions; Step 3: update parameters $\boldsymbol{x}$ value; Step 4: update Hessian matrix; Step 5: check if the iteration stops. If not, go to Step 2.

3.4. Hypothesis Testing

Since it is expected that miRNA negatively regulates the protein expression levels of its target genes, to test if there is a significant relationship between a specific pair of miRNA and protein, the hypothesis test can be set up as a one-sided test:

$$H_0: \alpha_2 = 0 \ vs \ H_a: \alpha_2 < 0$$

Once the maximum likelihood estimates are obtained, a likelihood ratio test (LRT), a Wald test or a Score test can be constructed:

Likelihood Ratio Test (LRT):

$$T.S. = -2ln\big(\sup\big(L(\boldsymbol{\phi}, \sigma_0, \sigma_1 | y_{ij}, z_i): \alpha_2 \geq 0\ \big)\big)$$
$$+ 2\ ln\left(sup\left(L(\boldsymbol{\phi}, \sigma_0, \sigma_1 | y_{ij}, z_i)\right)\right) \sim\chi_1^2 \ under \ H_0$$

Wald Test:

$$T.S. = \frac{\hat{\alpha}_2}{I(\hat{\boldsymbol{\phi}})_{\alpha_2}^{-\frac{1}{2}}} \sim N(0,1) \ under \ H_0$$

Score Test:

$$T.S. = \frac{\hat{\alpha}_2}{I(\boldsymbol{\phi}_0)_{\alpha_2}^{-\frac{1}{2}}} \sim N(0,1) \ under \ H_0$$

where $I(\hat{\boldsymbol{\phi}})$ represent the Fisher information matrix of the likelihood function.

However, LRT can be very time consuming and the confidence interval of α_2 is difficult to calculate for Score test. Thus Wald tests were used in our simulation and real data example.

Information Matrix and Standard Errors

Fisher Information matrix $I(\theta)$ where θ is the unknown parameters is commonly used to calculate the variance-covariance matrix of the maximum likelihood estimates $\hat{\theta}$. The variances of MLEs are the diagonal values in the inverse of the information matrix, and the information matrix is calculated from the negative of the expected value of Hessian matrix $H(\theta)$. Theoretically the variance equals to the Cramer-Rao lower bound which implies that MLEs are efficient estimators of the parameters. But in practice, it is hard to calculate the mean value of Hessian matrix when the likelihood function is rather complex, so we directly use the calculated Hessian matrix in the last iteration step to calculate the standard errors of MLEs.

3.5. A Simulation Study

Extensive simulation studies were carried out to examine the performance of our proposed integrated model and to compare with the naïve model approach. Protein intensities were generated by using a sigmoidal response curve (Figure 3.4-a). And a typical miRNA expression distribution in TCGA dataset was borrowed in this simulation to mimic the real data and generate protein EC_{50}s (Figure 3.4-b). Also, the true values of $\{ \beta_1, \beta_2, \beta_3, \sigma_0, \sigma_1 \}$ were set as {50, 30000, 1, 1, and 500} to mimic parameter values estimated from a real TCGA ovarian cancer data set. Different strengths of correlation between miRNA and protein expression levels, as characterized by α_2, were examined in a range from 0, which represents the null hypothesis, to -1.5, which yields the power of 1 for the integrated method. In order to investigate the performance of two models with protein intensity values located in different areas of the response curve, α_1 was set as 0 and 5 corresponding to the middle part and upper part (boundary) of sigmoidal curve, respectively. The upper part of a sigmoidal curve corresponds to a scenario where most of intensity levels are close to the saturation point. The RPPA intensity levels range between 10 and 30100. An illustration of the sigmoidal curve used to generate simulated data was showed in Figure 3.4-a. The locations of protein intensity center were marked by circles. If simulated intensity values are beyond the imaging boundary, they would be replaced with the boundary value with small error (Gaussian distributed with mean 0 and standard deviation 5). 1000 simulations were carried out for each parameter setting under different sample sizes (N=20, 50, 100 and 300). Generally, there are 5 diluted samples in one dilution series, so $J = 5$ was used in our simulation setting. Pre-specified type-I error was set to be 0.05.

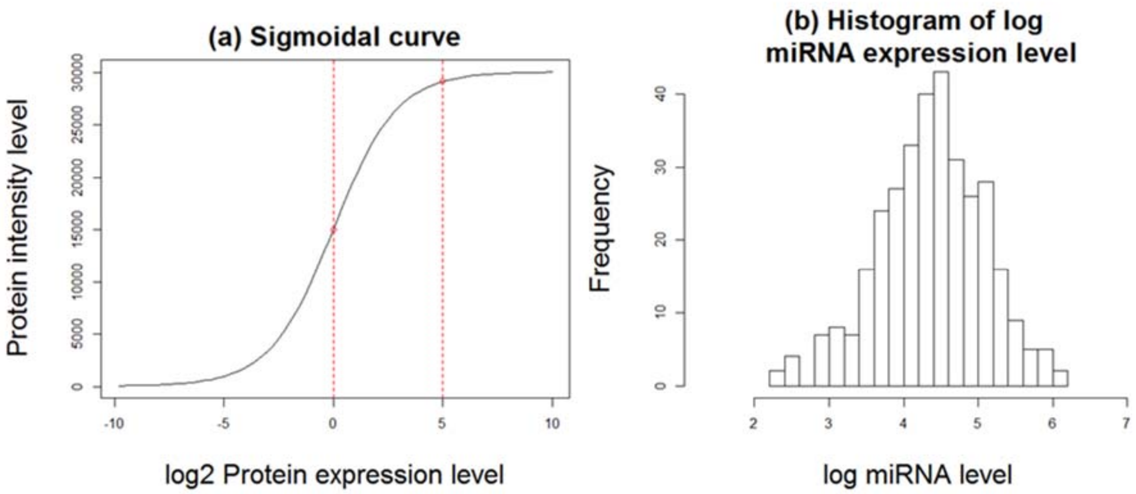


Figure 3.4| An illustration of (a) a sigmoidal shape response curve (b) histogram of a typical miRNA expression levels in TCGA ovarian cancer data. When $\boldsymbol{\alpha_1}$ was set to be 0, the center of the EC$_{50}$s would located at 0; when $\boldsymbol{\alpha_1}$ was set to be 5, the center of EC$_{50}$s would located at 5.

The false positive rates and detection powers for miRNA targets for both the integrated model and the naïve model under different sample sizes were shown in Figures 3.5 and 3.7 for two sets of simulated data. It is clear that when there was no relationship between miRNA and protein ($\alpha_2 = 0$), both models can well control the pre-specified type-I error when sample size were bigger than 50. The integrated model was consistently more powerful than the naïve model, especially when the RPPA intensity levels are close to the boundaries of imaging limits (Figure 3.7). Figures 3.6 and 3.8 illustrated the variations of $\hat{\alpha}_2$ under different simulation settings. The integrated model consistently yielded parameter estimates of α_2 with similar or much less standard errors than those from the naïve model. Table 3.1-3.2 listed the detailed point estimates of all parameters and their corresponding standard errors which also supported the conclusion. When the RPPA intensities reached the upper flatter part of the sigmoidal curve, which caused information loss because of intensity level truncation at the saturation points, both the naïve and the integrated method over-estimated β_1, which represents the lower imaging limits. However, in this situation the integrated method still had a much larger detection power than the naïve method (Figure 3.7).

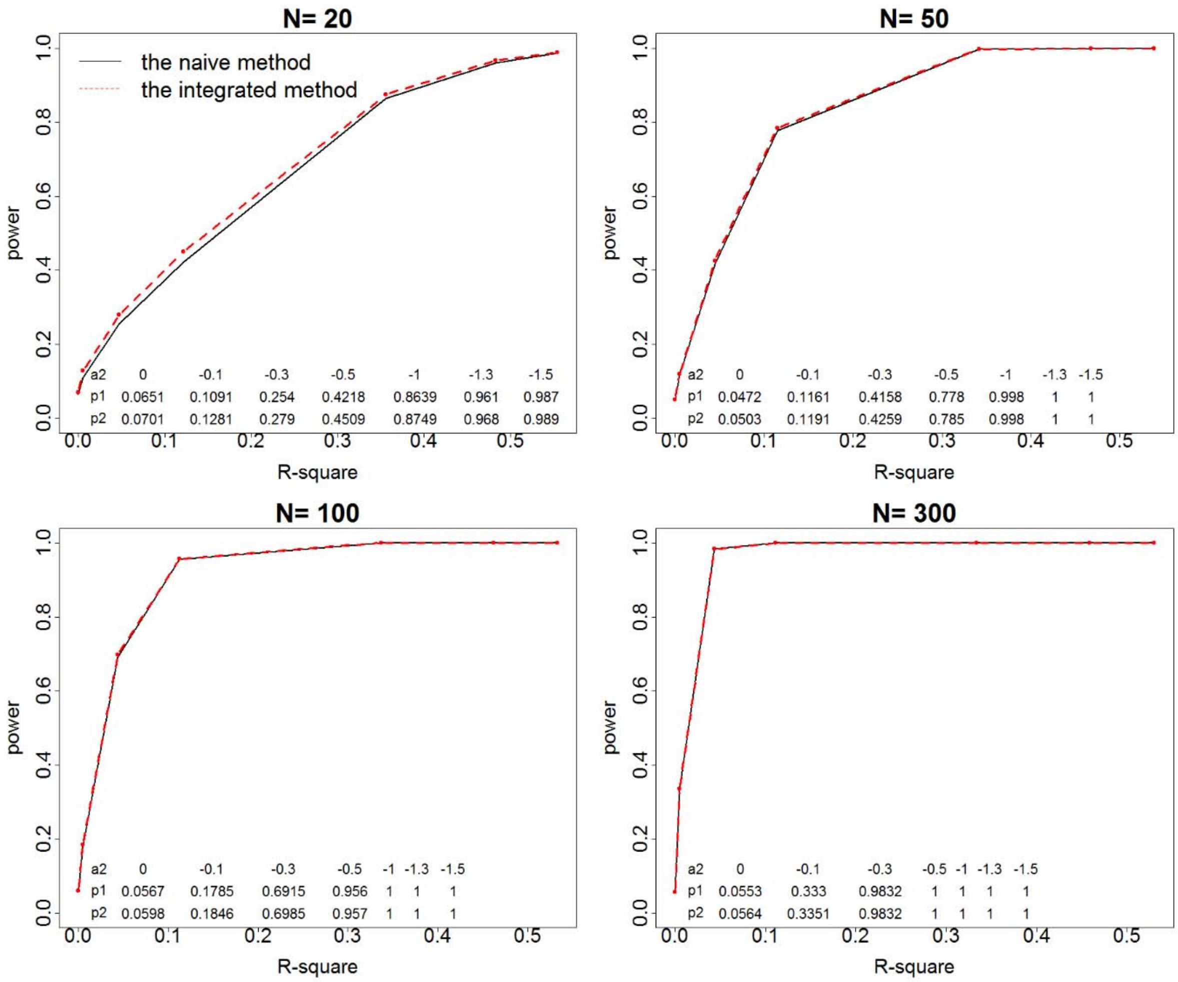


Figure 3.5| Power curve of the naïve model (solid line) and the integrated model (dashed line) according to different simulation scenarios: sample size ranged from 20 to 300 and the protein intensities were located on the middle part of a sigmoidal curve. Detection powers (type-I error if $\boldsymbol{\alpha_2 = 0}$) denoted by $\boldsymbol{p_1}$ and $\boldsymbol{p_2}$ under different correlation strengths were report on the bottom of each plot for the naïve and integrated models, respectively. Both models can well control the pre-specified type-I error when sample size were bigger than 50. Two models had similar detection performance, especially when sample size increased.

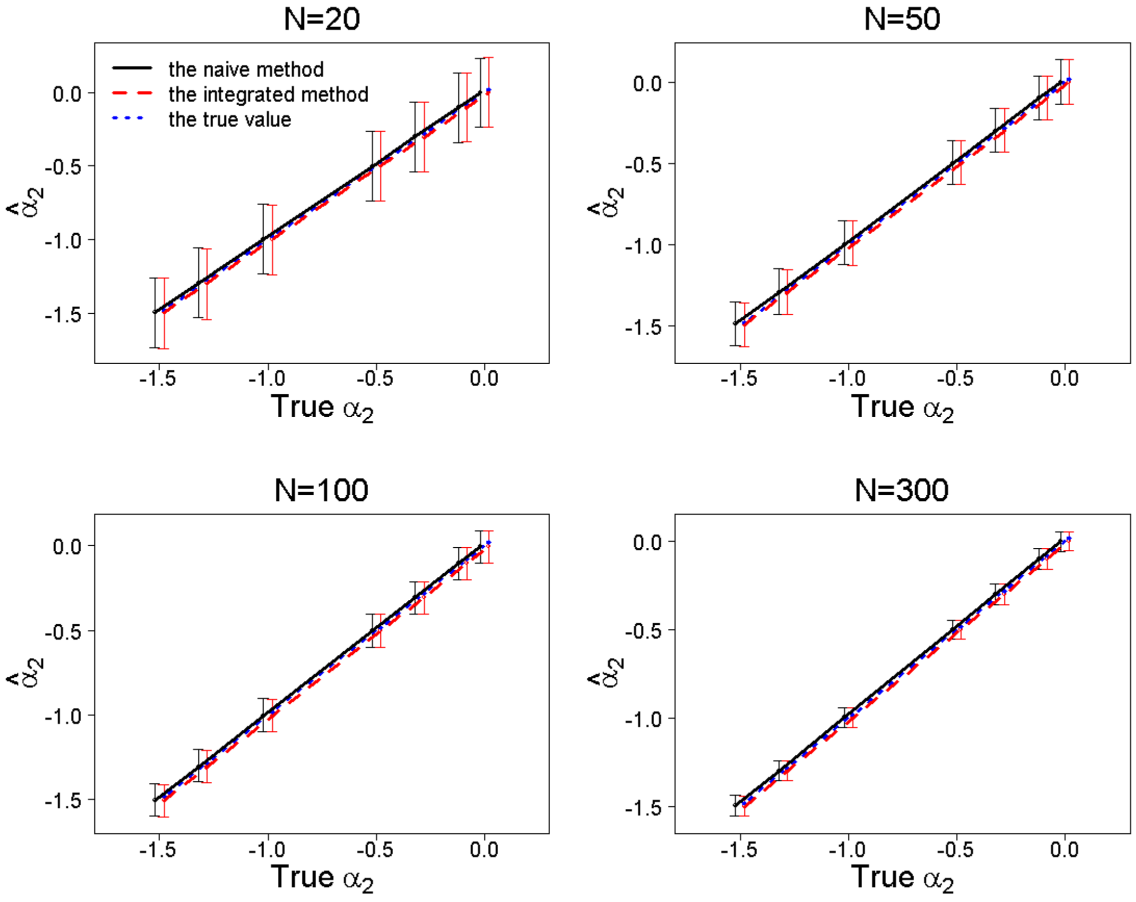


Figure 3.6| Error bars for point estimators of α_2, $\widehat{\alpha}_2$, by the integrated model (dashed line) and naïve model (solid line) comparing to the real α_2(dotted line in blue) value when protein intensities were located on the middle part of a sigmoidal curve. The middle points of the bars were the median values of $\widehat{\boldsymbol{\alpha}}_2$s, the upper bars and the lower bars represented the Q3 and Q1 of $\widehat{\boldsymbol{\alpha}}_2$s, respectively. Sample size was from 20 to 300 and the protein intensity located on the middle part of sigmoidal curve; the integrated model had a similar performance as the naïve model.

Table 3.1| Table for the detailed point estimates of all unknown parameters and their standard error when protein intensities were located on the middle part of a sigmoidal curve. Sample size was from 20 to 300. The integrated model had a similar performance as the naïve model.

	α_1	α_2	β_1	β_2	β_3	σ_0	σ_1
Sample size	20						
true value	0	0	50	30000	1	1	500
naive model	0.0213 (0.0517)	-0.0063 (0.0117)	78.13 (22.56)	29958.77 (31.48)	1.0022 (0.0014)	0.9809 (0.0053)	463.61 (1.34)
integrated model	0.015 (0.052)	-0.0062 (0.0117)	142.07 (15.3)	29881.73 (28.8)	1.0077 (0.0014)	0.9298 (0.0051)	490.89 (1.24)
true value	0	-0.1	50	30000	1	1	500
naive model	-0.0062 (0.0057)	-0.1066 (0.0117)	78.43 (22.69)	29954.12 (31.17)	1.0023 (0.0014)	0.981 (0.0053)	463.75 (1.34)
integrated model	-0.0106 (0.0068)	-0.1067 (0.0117)	136.82 (15.25)	29875.35 (28.62)	1.0078 (0.0014)	0.9299 (0.005)	490.51 (1.23)
true value	0	-0.3	50	30000	1	1	500
naive model	-0.0015 (0.0057)	-0.3067 (0.0117)	83.04 (23.4)	29904.9 (31.25)	1.0044 (0.0014)	0.9811 (0.0053)	464.85 (1.38)
integrated model	-0.0126 (0.0068)	-0.3071 (0.0117)	168.89 (15.35)	29827.69 (29.14)	1.0102 (0.0014)	0.93 (0.0051)	491.49 (1.23)
true value	0	-0.5	50	30000	1	1	500
naive model	-0.0015 (0.0057)	-0.5063 (0.0117)	81.23 (22.74)	29909.99 (30.21)	1.0045 (0.0014)	0.9807 (0.0053)	464.58 (1.36)
integrated model	-0.0119 (0.0068)	-0.5069 (0.0117)	161.52 (14.84)	29838.19 (28.02)	1.0096 (0.0014)	0.9297 (0.0051)	491.18 (1.23)
true value	0	-1	50	30000	1	1	500
naive model	-0.0043 (0.0061)	-1.0021 (0.0117)	23.03 (20.93)	30048.43 (26.41)	0.9997 (0.0012)	0.9795 (0.0053)	465.42 (1.4)
integrated model	-0.0105 (0.0068)	-1.005 (0.0117)	91.21 (13.22)	29964.21 (24.28)	1.003 (0.0012)	0.9298 (0.0051)	491.06 (1.27)
true value	0	-1.3	50	30000	1	1	500
naive model	-0.0055 (0.006)	-1.3033 (0.0117)	39.67 (18.69)	30027.79 (24.25)	1.0018 (0.0012)	0.9792 (0.0053)	465.7 (1.37)
integrated model	-0.0099 (0.0067)	-1.3077 (0.0118)	96.34 (11.97)	29952.63 (22.46)	1.0035 (0.0011)	0.9301 (0.005)	491.25 (1.25)
true value	0	-1.5	50	30000	1	1	500
naive model	-0.0096 (0.0063)	-1.5032 (0.0117)	83.81 (17.38)	29973.69 (22.32)	1.0042 (0.0011)	0.9789 (0.0053)	464.47 (1.35)
integrated model	-0.0111 (0.0069)	-1.5086 (0.0117)	124.24 (11.18)	29909.02 (20.75)	1.0054 (0.0011)	0.93 (0.0051)	491.11 (1.28)
Sample size	50						
true value	0	0	50	30000	1	1	500
naive model	-0.023 (0.0283)	0.0054 (0.0064)	-11.3 (16.58)	30137.24 (20.17)	0.9952 (9e-04)	0.9923 (0.0032)	457.55 (0.85)
integrated model	-0.0259 (0.0284)	0.0053 (0.0064)	43.36 (9.77)	30057.24 (18.46)	0.9982 (9e-04)	0.9719 (0.0032)	495.97 (0.79)
true value	0	-0.1	50	30000	1	1	500
naive model	-0.0012 (0.0041)	-0.0946 (0.0064)	-6.46 (16.12)	30141.23 (19.69)	0.9951 (9e-04)	0.9925 (0.0032)	457.14 (0.84)
integrated model	-0.0026 (0.0043)	-0.0947 (0.0064)	39.43 (9.46)	30061.21 (18.05)	0.9982 (8e-04)	0.972 (0.0032)	495.79 (0.79)

true value	0	-0.3	50	30000	1	1	500
naive model	0.0021 (0.0041)	-0.2942 (0.0064)	-14.38 (15.88)	30131.06 (18.59)	0.9954 (8e-04)	0.9923 (0.0032)	458.04 (0.86)
integrated model	-9e-04 (0.0044)	-0.2945 (0.0064)	36.9 (8.99)	30053.19 (17.01)	0.9983 (8e-04)	0.9719 (0.0032)	496.54 (0.78)
true value	0	-0.5	50	30000	1	1	500
naive model	0.0031 (0.004)	-0.4939 (0.0064)	-10.89 (15.49)	30113.24 (17.77)	0.9965 (8e-04)	0.9919 (0.0032)	457 (0.84)
integrated model	-0.002 (0.0044)	-0.4946 (0.0064)	46.42 (8.56)	30042.02 (16.43)	0.9988 (8e-04)	0.9716 (0.0032)	496.05 (0.79)
true value	0	-1	50	30000	1	1	500
naive model	-0.0031 (0.0042)	-0.9927 (0.0064)	16.45 (13.86)	30112.04 (16.06)	0.998 (8e-04)	0.9915 (0.0032)	457.22 (0.82)
integrated model	-0.0022 (0.0044)	-0.9948 (0.0064)	46.61 (7.86)	30043.54 (14.75)	0.9986 (7e-04)	0.9719 (0.0032)	496.19 (0.79)
true value	0	-1.3	50	30000	1	1	500
naive model	-0.0068 (0.0043)	-1.2911 (0.0064)	48.01 (13.45)	30080.17 (14.29)	1.0002 (7e-04)	0.9911 (0.0032)	457.21 (0.82)
integrated model	-0.0018 (0.0044)	-1.2949 (0.0064)	60.83 (7.18)	30013.75 (13.47)	0.9997 (7e-04)	0.9721 (0.0032)	496.15 (0.79)
true value	0	-1.5	50	30000	1	1	500
naive model	-0.0078 (0.0041)	-1.4898 (0.0064)	77.09 (11.46)	30027.67 (13.43)	1.0032 (7e-04)	0.9909 (0.0032)	456.69 (0.78)
integrated model	-0.0026 (0.0043)	-1.4944 (0.0064)	82.21 (6.6)	29975.26 (12.74)	1.0017 (7e-04)	0.972 (0.0032)	496.5 (0.77)
Sample size	100						
true value	0	0	50	30000	1	1	500
naive model	0.0124 (0.0207)	-0.0032 (0.0047)	-21.78 (12.82)	30189.17 (14.32)	0.9927 (6e-04)	0.997 (0.0022)	455.17 (0.62)
integrated model	0.0143 (0.0208)	-0.0033 (0.0047)	7.17 (6.74)	30116.73 (12.96)	0.9951 (6e-04)	0.9863 (0.0022)	498.45 (0.57)
true value	0	-0.1	50	30000	1	1	500
naive model	0.0029 (0.0032)	-0.1028 (0.0047)	-46.45 (12.49)	30192.3 (13.75)	0.9929 (6e-04)	0.9965 (0.0022)	454.27 (0.6)
integrated model	-5e-04 (0.0032)	-0.1029 (0.0047)	1.55 (6.47)	30125.41 (12.4)	0.9948 (6e-04)	0.9859 (0.0022)	498.08 (0.56)
true value	0	-0.3	50	30000	1	1	500
naive model	3e-04 (0.0033)	-0.3033 (0.0047)	-26.94 (12.46)	30177.98 (13.77)	0.9932 (6e-04)	0.9968 (0.0022)	455.26 (0.6)
integrated model	-9e-04 (0.0032)	-0.3036 (0.0047)	11.3 (6.47)	30110.92 (12.44)	0.9951 (6e-04)	0.9861 (0.0022)	498.77 (0.56)
true value	0	-0.5	50	30000	1	1	500
naive model	0.0034 (0.0033)	-0.5024 (0.0047)	-29.46 (12.41)	30152.94 (12.87)	0.9948 (6e-04)	0.9968 (0.0022)	454.68 (0.59)
integrated model	-0.002 (0.0032)	-0.5029 (0.0047)	23.79 (6.41)	30091.34 (11.81)	0.996 (6e-04)	0.9863 (0.0022)	498.68 (0.56)
true value	0	-1	50	30000	1	1	500
naive model	-0.0063 (0.0033)	-1.0002 (0.0047)	18.58 (10.56)	30140.4 (11.11)	0.9961 (5e-04)	0.9965 (0.0022)	454.05 (0.58)
integrated model	-0.0024 (0.0032)	-1.0018 (0.0047)	31.28 (5.38)	30081.86 (10.35)	0.9964 (5e-04)	0.9864 (0.0022)	498.21 (0.57)
true value	0	-1.3	50	30000	1	1	500
naive model	-0.0072 (0.0033)	-1.3008 (0.0047)	47.51 (9.63)	30088.26 (9.9)	0.9992 (5e-04)	0.9967 (0.0022)	453.78 (0.59)

integrated model	-0.0024 (0.0032)	-1.3036 (0.0047)	52.33 (4.87)	30038.92 (9.46)	0.9982 (5e-04)	0.9871 (0.0022)	498.32 (0.59)
true value	0	-1.5	50	30000	1	1	500
naive model	-0.0066 (0.0032)	-1.5001 (0.0047)	65.64 (8.32)	30046.58 (8.83)	1.0018 (4e-04)	0.996 (0.0022)	452.58 (0.56)
integrated model	-8e-04 (0.0032)	-1.5037 (0.0047)	62.17 (4.51)	30006.08 (8.55)	0.9998 (4e-04)	0.9864 (0.0022)	498.25 (0.58)
Sample size	300						
true value	0	0	50	30000	1	1	500
naive model	-0.0029 (0.0118)	-7e-04 (0.0027)	-7.69 (8.36)	30141.18 (8.44)	0.9945 (4e-04)	0.9998 (0.0014)	451.18 (0.34)
integrated model	-0.0039 (0.0118)	-7e-04 (0.0027)	17.74 (3.87)	30099.42 (7.63)	0.9956 (4e-04)	0.9953 (0.0014)	499.45 (0.32)
true value	0	-0.1	50	30000	1	1	500
naive model	3e-04 (0.0023)	-0.1009 (0.0026)	-45.21 (8.74)	30156.77 (7.84)	0.9938 (4e-04)	0.9999 (0.0014)	451.88 (0.34)
integrated model	-0.0084 (0.002)	-0.1009 (0.0026)	11.41 (3.65)	30117.72 (7.15)	0.9947 (3e-04)	0.9955 (0.0014)	500 (0.32)
true value	0	-0.3	50	30000	1	1	500
naive model	-0.0038 (0.0024)	-0.3005 (0.0027)	-19.8 (8.81)	30144.6 (8.1)	0.9945 (4e-04)	0.9996 (0.0014)	451.86 (0.35)
integrated model	-0.0069 (0.0019)	-0.3006 (0.0027)	14.95 (3.71)	30101.39 (7.38)	0.9955 (4e-04)	0.9953 (0.0014)	499.85 (0.31)
true value	0	-0.5	50	30000	1	1	500
naive model	-0.0041 (0.0023)	-0.5006 (0.0026)	-16.25 (8.32)	30142.07 (7.35)	0.9948 (3e-04)	0.9996 (0.0014)	451.4 (0.33)
integrated model	-0.0085 (0.0019)	-0.5009 (0.0026)	22.01 (3.48)	30102.75 (6.82)	0.9954 (3e-04)	0.9953 (0.0014)	499.8 (0.32)
true value	0	-1	50	30000	1	1	500
naive model	-0.0121 (0.0023)	-0.999 (0.0027)	30.28 (7.06)	30107.97 (6.46)	0.9968 (3e-04)	0.9996 (0.0014)	451 (0.33)
integrated model	-0.0082 (0.002)	-0.9999 (0.0027)	32.49 (3.08)	30070.72 (6.22)	0.9967 (3e-04)	0.9955 (0.0014)	499.92 (0.33)
true value	0	-1.3	50	30000	1	1	500
naive model	-0.0101 (0.0021)	-1.2999 (0.0028)	47.45 (5.33)	30058.55 (5.39)	0.9997 (3e-04)	1.0019 (0.0027)	448.96 (0.31)
integrated model	-0.0077 (0.002)	-1.3008 (0.0028)	49.28 (2.79)	30036.58 (5.22)	0.9983 (3e-04)	0.9955 (0.0014)	499.31 (0.34)
true value	0	-1.5	50	30000	1	1	500
naive model	-0.0106 (0.0021)	-1.4962 (0.0027)	47.23 (4.56)	30059.42 (5.23)	1.0004 (3e-04)	0.9997 (0.0015)	450.03 (1.52)
integrated model	-0.0091 (0.0021)	-1.4986 (0.0028)	49.02 (2.61)	30043.6 (5.15)	0.9979 (3e-04)	0.9955 (0.0014)	499.52 (0.35)

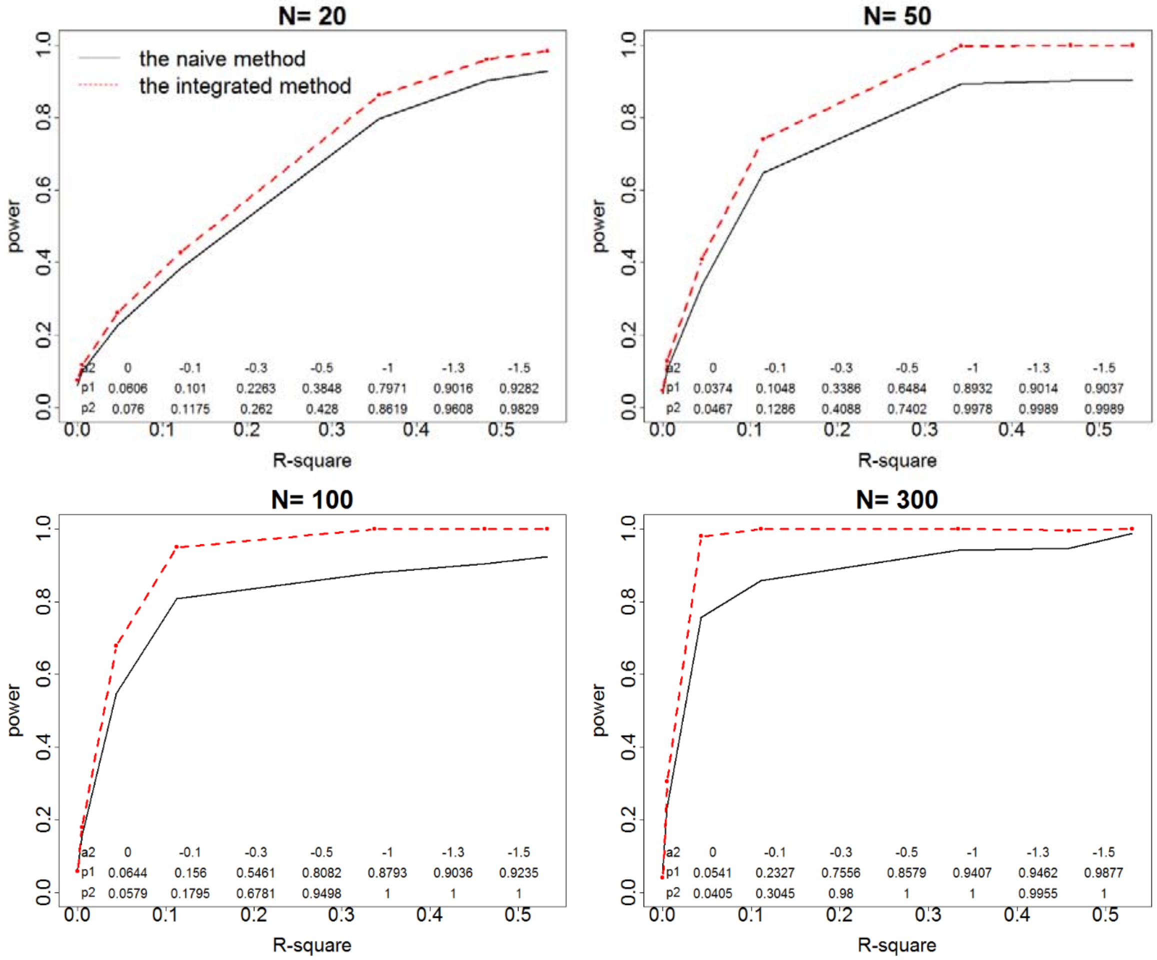


Figure 3.7| Power curves of the naïve model (solid line) and the integrated model (dashed line) according different simulation scenarios: sample size ranged from 20 to 300 and the protein intensities were located on the upper part of a sigmoidal curve. Detection powers (type-I error if $\boldsymbol{\alpha_2 = 0}$) denoted by $\boldsymbol{p_1}$ and $\boldsymbol{p_2}$ under different correlation strength were report on the bottom of each figure for the naïve and integrated models, respectively. Both models can well control the pre-specified type-I error when sample size were bigger than 50. The integrated model was consistently more powerful than the naïve model.

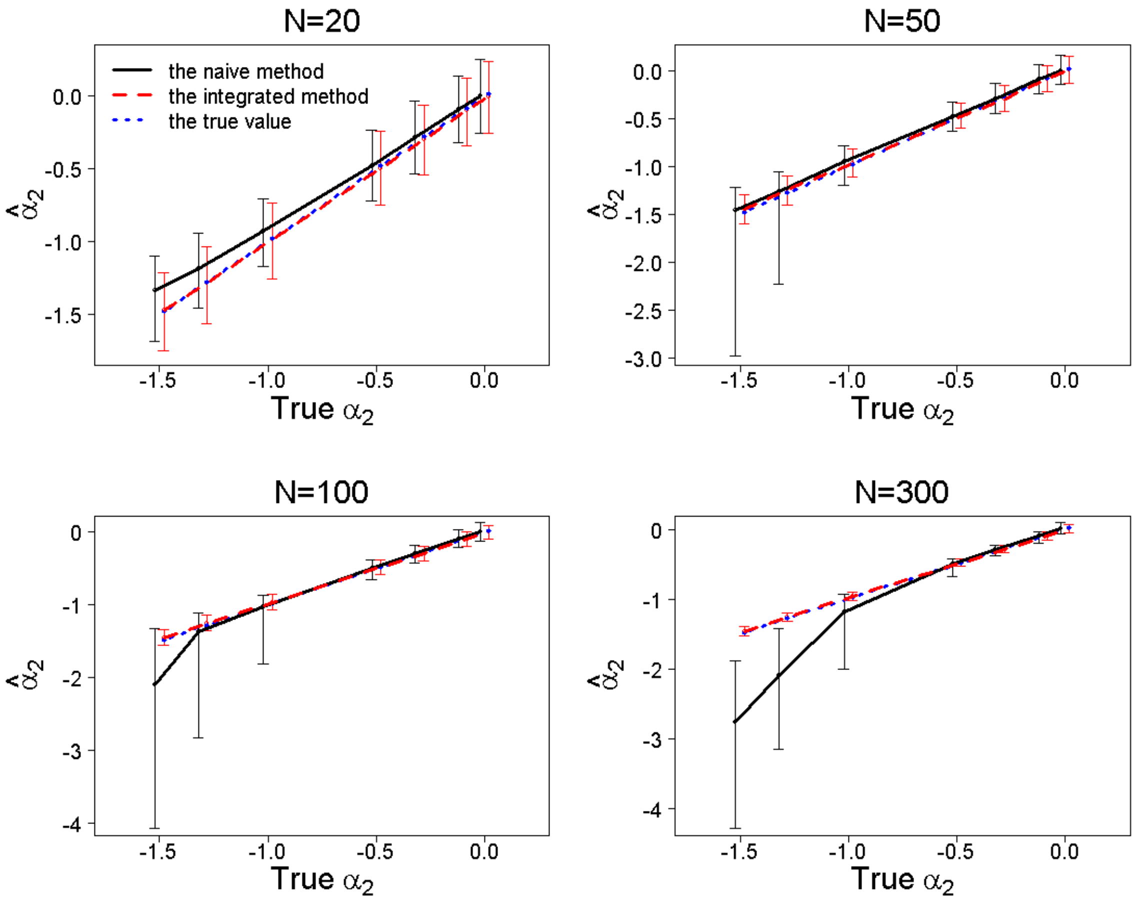


Figure 3.8| Error bars for point estimates of α_2, $\widehat{\alpha}_2$, by the integrated model (dashed line) and naïve model (solid line) comparing to the real α_2(dotted line in blue) value when protein intensities were located on the middle part of a sigmoidal curve. The middle points of the bars were the med ian value of $\widehat{\boldsymbol{\alpha}}_2$s, the upper bars and the lower bars represented the Q3 and Q1 of $\widehat{\boldsymbol{\alpha}}_2$s, respectively. Sample size was from 20 to 300 and the protein intensity located on the upper part of sigmoidal curve; the $\boldsymbol{\alpha}_2$s estimated by the integrated model had much narrower bars than the naïve model.

Table 3.2| Table for the detailed point estimates of all unknown parameters and their standard error when protein intensities were located on the upper part of a sigmoidal curve. Sample size was from 20 to 300. Truncation was applied to the boundary of intensity level; the integrated model consistently yielded parameter estimates of α_2 with similar or much less standard errors than the naïve model.

	$\boldsymbol{\alpha_1}$	$\boldsymbol{\alpha_2}$	$\boldsymbol{\beta_1}$	$\boldsymbol{\beta_2}$	$\boldsymbol{\beta_3}$	$\boldsymbol{\sigma_0}$	$\boldsymbol{\sigma_1}$
Sample size	20						
true value	5	0	50	30000	1	1	500
naive model	2.3538 (0.0337)	0.0108 (0.0348)	21673.99 (73.67)	8063.19 (72.22)	1.8432 (0.0062)	1.5515 (0.1377)	505.9 (4.52)
integrated model	4.4653 (0.08)	-0.0161 (0.0139)	-274276.76 (52197.04)	304133.74 (52200.42)	1.2805 (0.011)	0.9099 (0.0111)	457.39 (1.95)
true value	5	-0.1	50	30000	1	1	500
naive model	2.3656 (0.0399)	-0.0895 (0.0592)	21700.84 (71.7)	8041.22 (70.25)	1.8401 (0.0059)	1.6134 (0.1588)	504.84 (3.24)
integrated model	4.5686 (0.0859)	-0.0693 (0.0418)	-330800.61 (56594.51)	360639.78 (56591.54)	1.2918 (0.0288)	0.9732 (0.0742)	458.96 (2)
true value	5	-0.3	50	30000	1	1	500
naive model	2.3695 (0.0275)	-0.3577 (0.0482)	21516.03 (74.6)	8221.13 (73.15)	1.8385 (0.0057)	1.5135 (0.1068)	505.61 (3.97)
integrated model	4.403 (0.0769)	-0.308 (0.0123)	-249677.33 (50408.57)	279512.95 (50405.41)	1.2762 (0.0105)	0.8937 (0.0099)	456.25 (1.72)
true value	5	-0.5	50	30000	1	1	500
naive model	2.4446 (0.0374)	-0.7107 (0.0782)	21269.05 (74.86)	8464.35 (73.51)	1.8481 (0.0059)	1.6564 (0.1521)	507.56 (2.77)
integrated model	4.5227 (0.0783)	-0.5035 (0.0123)	-250395.86 (48590.31)	280258.79 (48592.42)	1.2486 (0.0097)	0.8891 (0.0102)	457.35 (1.82)
true value	5	-1	50	30000	1	1	500
naive model	2.7712 (0.0505)	-1.4615 (0.1208)	19902.57 (90.02)	9814.9 (88.76)	1.8388 (0.0058)	2.1685 (0.1959)	527.34 (4.87)
integrated model	4.1187 (0.0636)	-1.0024 (0.0148)	-52274.05 (21283.7)	82100.65 (21281.33)	1.2984 (0.0263)	0.8417 (0.0365)	469.74 (2.81)
true value	5	-1.3	50	30000	1	1	500
naive model	2.9411 (0.0548)	-1.7328 (0.0838)	18770.02 (98.58)	10936.33 (97.49)	1.828 (0.0054)	2.2838 (0.2277)	539.5 (4.31)
integrated model	4.1799 (0.0528)	-1.3159 (0.0145)	-81574.65 (30566.75)	111394.19 (30567.8)	1.2845 (0.0095)	0.8634 (0.0125)	478.65 (2.39)
true value	5	-1.5	50	30000	1	1	500
naive model	3.2206 (0.0717)	-2.3778 (0.1593)	18002.82 (105.51)	11696.6 (104.59)	1.8231 (0.0057)	2.9557 (0.2621)	560.41 (7.25)
integrated model	3.6493 (0.6203)	-0.964 (0.6069)	-51974.4 (22391.12)	81776.04 (22386.56)	1.2932 (0.0201)	0.8548 (0.0355)	488.64 (3.71)
Sample size	50						
true value	5	0	50	30000	1	1	500
naive model	2.7437 (0.0369)	-0.0469 (0.0526)	20499.39 (63.98)	9225.41 (62.87)	1.7402 (0.0037)	2.7635 (0.2377)	485.25 (2.14)
integrated model	3.5198 (0.0405)	0.0089 (0.0067)	8598.57 (1581.12)	21172.62 (1582.55)	1.4285 (0.0103)	0.8595 (0.0138)	489.77 (1.59)
true value	5	-0.1	50	30000	1	1	500

naive	2.7316	-0.1643	20463.66	9261.85	1.738	2.6338	484.29
model	(0.0347)	(0.0384)	(65.62)	(64.57)	(0.0038)	(0.2266)	(1.72)
integrated	3.4811	-0.0898	9927.86	19842.78	1.434	0.8728	491.19
model	(0.0385)	(0.0067)	(1621.29)	(1622.67)	(0.0104)	(0.0129)	(1.61)
true value	5	-0.3	50	30000	1	1	500
naive	2.8069	-0.4472	20293.49	9426.7	1.7416	2.9565	486.58
model	(0.0401)	(0.047)	(66.02)	(65.05)	(0.0037)	(0.2576)	(1.97)
integrated	3.4894	-0.2882	12135.89	17635.35	1.4246	0.8701	491.18
model	(0.0357)	(0.0067)	(534.51)	(537.21)	(0.0101)	(0.0131)	(1.66)
true value	5	-0.5	50	30000	1	1	500
naive	2.828	-0.6441	19946.91	9770.55	1.7402	2.8163	491.23
model	(0.0387)	(0.0451)	(69.44)	(68.46)	(0.0038)	(0.2509)	(2.77)
integrated	3.5327	-0.4769	11365.82	18404.38	1.4188	0.8762	493.4
model	(0.0365)	(0.0067)	(735.99)	(738.2)	(0.01)	(0.0124)	(1.7)
true value	5	-1	50	30000	1	1	500
naive	3.2183	-1.7113	18367.63	11331.51	1.7405	3.9333	506.65
model	(0.047)	(0.0905)	(79.4)	(78.56)	(0.0037)	(0.301)	(2.69)
integrated	3.4259	-0.9716	13477.54	16260.6	1.4497	0.7908	512.56
model	(0.029)	(0.0072)	(288.17)	(291.47)	(0.0086)	(0.0171)	(1.84)
true value	5	-1.3	50	30000	1	1	500
naive	3.5889	-2.5651	17174.96	12514.49	1.7375	5.4373	521.52
model	(0.0593)	(0.1361)	(86.16)	(85.49)	(0.0037)	(0.3774)	(3.13)
integrated	3.2866	-1.4043	13337.94	16383.67	1.4403	0.8708	528.37
model	(0.1729)	(0.1428)	(260.58)	(263.67)	(0.0146)	(0.1229)	(2.55)
true value	5	-1.5	50	30000	1	1	500
naive	3.8296	-2.9831	16141.61	13546.24	1.7198	6.1208	523.94
model	(0.0621)	(0.1416)	(89.86)	(89.49)	(0.0039)	(0.3784)	(2.07)
integrated	3.305	-1.3648	13436.45	16273.32	1.4225	0.9029	536.53
model	(0.1669)	(0.105)	(198.21)	(201.3)	(0.0332)	(0.1719)	(2.56)
Sample size	100						
true value	5	0	50	30000	1	1	500
naive	2.9087	-0.039	19661.6	10058.46	1.6748	3.337	468.53
model	(0.0269)	(0.0434)	(56.83)	(56.2)	(0.003)	(0.2366)	(1.22)
integrated	3.0493	-0.0055	16387.53	13333.07	1.5292	0.8232	501.83
model	(0.0261)	(0.006)	(641.05)	(642.19)	(0.0075)	(0.0178)	(1.1)
true value	5	-0.1	50	30000	1	1	500
naive	2.9097	-0.1508	19584.67	10137.22	1.6675	3.2625	469.76
model	(0.0291)	(0.043)	(57.9)	(57.29)	(0.0028)	(0.2545)	(1.53)
integrated	3.0247	-0.0985	17062.36	12657.31	1.5329	0.8748	502.86
model	(0.0234)	(0.0065)	(191.54)	(194.2)	(0.0073)	(0.0133)	(1.06)
true value	5	-0.3	50	30000	1	1	500
naive	2.9106	-0.3621	19313.04	10404.89	1.6712	3.0649	469.78
model	(0.0275)	(0.0361)	(62.78)	(62.09)	(0.003)	(0.2302)	(0.91)
integrated	3.0442	-0.3021	16981.12	12738.09	1.5236	0.8185	504.55
model	(0.0224)	(0.0073)	(176.62)	(179.23)	(0.007)	(0.0172)	(1.09)
true value	5	-0.5	50	30000	1	1	500
naive	3.0186	-0.7679	18919.14	10793.17	1.674	3.701	474.8
model	(0.0323)	(0.044)	(64.29)	(63.66)	(0.003)	(0.2824)	(1.61)
integrated	3.0784	-0.4853	16620.4	13095.57	1.5193	0.8203	508.61
model	(0.0226)	(0.0049)	(198.41)	(200.87)	(0.0068)	(0.0165)	(1.12)
true value	5	-1	50	30000	1	1	500

naive	3.5281	-2.0565	17213.8	12482.15	1.6745	5.9013	486.43
model	(0.0442)	(0.0999)	(74.03)	(73.63)	(0.003)	(0.3507)	(0.92)
integrated	3.1546	-0.9644	16182.07	13518.68	1.5056	0.7447	521.9
model	(0.0163)	(0.0055)	(131.53)	(133.43)	(0.0053)	(0.0202)	(1.12)
true value	5	-1.3	50	30000	1	1	500
naive	3.8155	-2.6141	15813.29	13875.57	1.6668	7.1509	504.16
model	(0.0476)	(0.0999)	(80.09)	(79.85)	(0.0034)	(0.3873)	(2.87)
integrated	3.28	-1.2628	15230.07	14467.41	1.4792	0.7278	531.28
model	(0.0142)	(0.0074)	(110.54)	(111.99)	(0.0047)	(0.0212)	(1.16)
true value	5	-1.5	50	30000	1	1	500
naive	4.23	-3.6045	14844.17	14844.17	1.6571	9.335	513.6
model	(0.0626)	(0.1408)	(83.8)	(83.59)	(0.0034)	(0.4778)	(2.09)
integrated	3.4028	-1.4779	14282.26	15415.65	1.4536	0.6573	539.12
model	(0.016)	(0.0122)	(116.92)	(118.47)	(0.0046)	(0.0252)	(1.27)
Sample size	300						
true value	5	0	50	30000	1	1	500
naive	3.0135	0.0285	18437.53	11288.37	1.5908	3.2476	451.57
model	(0.0208)	(0.0232)	(67.92)	(67.65)	(0.003)	(0.263)	(1.21)
integrated	2.9	5e-04	18348.42	11365.95	1.5243	0.8799	498.43
model	(0.0112)	(0.0039)	(75.58)	(76.43)	(0.0043)	(0.0187)	(0.68)
true value	5	-0.1	50	30000	1	1	500
naive	3.0361	-0.1808	18456.78	11268.42	1.592	3.577	454.1
model	(0.0228)	(0.0329)	(69.96)	(69.74)	(0.0032)	(0.2943)	(2.08)
integrated	2.9366	-0.0989	18113.81	11603.85	1.516	0.879	496.92
model	(0.0163)	(0.0074)	(121.05)	(122.77)	(0.0054)	(0.0201)	(0.77)
true value	5	-0.3	50	30000	1	1	500
naive	3.0804	-0.3768	18175.54	11545.95	1.5939	4.1281	453.35
model	(0.027)	(0.0336)	(74.23)	(73.97)	(0.0032)	(0.3787)	(1.29)
integrated	2.9395	-0.2858	18058.62	11655.43	1.5166	0.8667	498.87
model	(0.0132)	(0.004)	(94.19)	(95.21)	(0.0046)	(0.0213)	(0.69)
true value	5	-0.5	50	30000	1	1	500
naive	3.1502	-0.7234	17717.61	11998.41	1.5974	3.9477	454.13
model	(0.0254)	(0.0412)	(82.68)	(82.46)	(0.0034)	(0.3312)	(0.75)
integrated	2.9937	-0.4811	17680.24	12031.57	1.5063	0.8291	500.36
model	(0.0129)	(0.0042)	(87.96)	(88.86)	(0.0047)	(0.0251)	(0.78)
true value	5	-1	50	30000	1	1	500
naive	3.5669	-1.7535	15740.25	13964.36	1.5926	6.4576	471.3
model	(0.043)	(0.09)	(113.86)	(113.86)	(0.0043)	(0.5025)	(3.33)
integrated	3.2607	-0.9702	15650.42	14066.12	1.4461	0.8718	505.15
model	(0.0177)	(0.0074)	(130.59)	(132.08)	(0.0057)	(0.0259)	(0.98)
true value	5	-1.3	50	30000	1	1	500
naive	4.0045	-2.2565	13772.02	15929.25	1.5632	12.2208	480.58
model	(0.1514)	(0.5547)	(120.38)	(120.41)	(0.0045)	(2.356)	(3.11)
integrated	3.1969	-0.7316	13720.47	16004.76	1.3919	2.9218	514.38
model	(0.3036)	(0.5495)	(126.43)	(128)	(0.0097)	(2.2545)	(4.56)
true value	5	-1.5	50	30000	1	1	500
naive	4.4603	-3.707	12633.68	17067.83	1.5489	12.0708	485.18
model	(0.0723)	(0.1756)	(117.4)	(117.75)	(0.0043)	(0.7155)	(1.18)
integrated	3.6474	-1.4779	12443.71	17289.63	1.3512	0.6374	510.48
model	(0.0193)	(0.0129)	(145.16)	(147.48)	(0.0055)	(0.0466)	(1.31)

3.6. Real Data Examples

Many studies of breast cancer and ovarian cancer have been done on gene expression level. The naïve model and integrated model were applied to ovarian cancer (OV) and breast cancer (BRCA) dataset from The Cancer Genome Atlas (TCGA) program which started from the year 2006 and is supported by the NCI and the NHGRI. In OV dataset, there were 333 ovarian cancer samples with both miRNA and RPPA data available. 352 miRNAs having more than 50% of non-zero counts and 165 proteins were included in our analyses in order to have sufficient information. In BRCA dataset, there were 239 breast cancer samples with both miRNA and RPPA data available. 417 miRNAs having more than 50% of non-zero counts and 165 proteins were included in our analyses.

3.6.1. Analysis of TCGA Ovarian Cancer (OV) Data

The results from both the naïve and integrated models on predicting miRNA targets were displayed in Table 3.3. False Discover Rate (FDR) at 10% was used to adjust for multiple testing (Benjamini & Hochberg, 1995). The integrated model approach we proposed found 1106 potential miRNA-protein pairs, 797 of which were on non-phosphorylated protein array. Totally 822 pairs were found on non-phosphorylated protein array: 250 out of them were found by the integrated model only and 25 pairs were found by the naïve model only. The integrated model found significantly more number of potential miRNA-protein pairs ($P<0.0001$ according to the McNemar’s test). Furthermore, we compared our results with miRNA targets identified by the miRanda algorithm (Enright et al., 2004; Hofacker et al., 1994; McCaskill, 1990; Zuker & Stiegler, 1981). 98 targets, which were found by both the integrated and the naïve model, and 31 targets, which were found only by the integrated model, were confirmed by the miRanda database.

However, only 6 targets found by the naïve model only were confirmed by the miRanda database. MirTarBase, a dataset based on manually surveying pertinent literature (Hsu et al., 2014) was further used to verify our results. 15 suggested targets found by both the integrated and the naïve model were supported by the MirTarBase dataset. 11 of the 15 suggested targets found by both the integrated and the naïve model were supported by strong experimental evidences according to the MirTarBase dataset. One suggested target found by the integrated model only were supported by strong experimental evidences according to the MirTarBase dataset. None of the suggested targets found by the naïve model only were supported by strong experimental evidences according to the MirTarBase dataset. This suggests that there could be a number of experimentally undiscovered miRNA targets included in the findings of integrated and naïve models. Those found miRNA-protein pairs with literature support, which were sorted by ascending order of adjusted p-values from the integrated model and the naïve model, were listed in Table 3.4.

Table 3.3 | Analysis results by the naïve and the integrated model compared with the miRanda and the MirTarBase based on TCGA ovarian cancer dataset. The integrated model found significantly more potential miRNA & non-phosphorylated protein pairs (P<0.0001 according to the McNemar's test). Results were compared with miRNA targets identified by miRanda algorithm and supported by the MirTarBase. Numbers in parentheses are percentage of pairs among total pairs which were found by either the naïve model method or the integrated model method.

Method[1]	All protein arrays found as miRNA targets	None phosphorylated protein arrays	Found in the miRanda	Found in the MirTarBase	Found in the MirTarBase[2]
Naïve+ Integrated+	719 (62.4%)	547 (66.6%)	98 (72.6%)	15 (88.2%)	11 (91.7%)
Naïve+ Integrated-	46 (4.0%)	25 (3.0%)	6 (4.4%)	1 (5.9%)	0 (0%)
Naïve- Integrated+	387 (33.6%)	250 (30.4%)	31 (23%)	1 (5.9%)	1 (8.3%)
Naïve- Integrated-	41250	25225	4010	557	285

1. Naïve+ Integrated+: Pairs found by both the integrated method and the naïve method; Naïve+ Integrated-: Pairs found by the naïve method only; Naïve- Integrated+: Pairs found by the integrated method only; Naïve- Integrated-: Pairs found by neither the integrated method nor the naïve method;
2. Supported by strong experimental evidences

Table 3.4| miRNA-protein corresponded genes pairs with literature support in TCGA ovarian cancer dataset. A number "1" was marked under the column for pairs found by the naïve model, integrated model, the MirTarBase, MirTarBase with strong experimental evidences (listed on the top) or the miRanda. Pairs were sorted by ascending order of adjusted p-values from the integrated model and the naïve model.

M1[1]	M2[1]	miRNA	Corresponding genes	P-values from M1	P-values from M2	MirTarBase	MirTarBase[2]	miRanda	References PMID
1	1	hsa-mir-150	NOTCH3	2.53E-08	1.66E-08	1	1	1	21551231
1	1	hsa-mir-150	TP53	4.67E-07	8.30E-07	1	1		23747308
1	1	hsa-mir-214	CTNNB1	6.28E-06	6.06E-06	1	1		23068095
1	1	hsa-mir-181a-1	BCL2L11	2.71E-05	1.84E-05	1	1		20841506
1	1	hsa-mir-223	IGF1R	1.92E-05	4.14E-05	1	1		22073238
1	1	hsa-mir-139	IGF1R	2.40E-05	4.43E-05	1	1		22580051
1	1	hsa-mir-181a-1	CDKN1B	8.68E-04	5.41E-04	1	1		19273599
1	1	hsa-mir-18a	ESR1	9.97E-04	6.37E-04	1	1	1	19684618
1	1	hsa-mir-145	IGF1R	9.20E-04	7.96E-04	1	1		19391107
1	1	hsa-mir-155	SMAD3	1.20E-03	1.08E-03	1	1		21036908
1	1	hsa-mir-21	MSH6	1.47E-03	1.46E-03	1	1		21078976
	1	hsa-mir-605	TP53	6.06E-03	1.67E-03	1	1		21217645
1	1	hsa-mir-7-1	CAV1	1.24E-03	1.29E-03	1		1	19073608
1	1	hsa-let-7a-2	BCL2L11	7.11E-04	1.64E-03	1		1	23622248
1	1	hsa-let-7a-1	BCL2L11	7.38E-04	1.68E-03	1		1	23622248
1	1	hsa-let-7a-3	BCL2L11	7.07E-04	1.69E-03	1		1	23622248
1		hsa-mir-146b	AKT1-3	9.22E-04	3.11E-03	1			23622248

1. M1 stands for the naïve model and M2 stands for the integrated model;
2. Supported by strong experimental evidences;

3.6.2. Analysis of TCGA Breast Cancer (BRCA) Data

The results from both the naïve and the integrated models on predicting miRNA targets were reported in Table 3.5. False Discover Rate (FDR) at 10% was used to adjust for multiple testing as well. The integrated model approach we proposed found 5159 potential miRNA-protein pairs, 3434 of which were on non-phosphorylated protein array. 3471 pairs were found on non-phosphorylated protein array: 3306 out of them were found by integrated model only and 128 pairs were found by naïve model only. Integrated model found significantly more number of potential miRNA-protein pairs (P<0.0001 according to the McNemar's test). 449 targets, which were found by both the integrated and the naïve model, and 21 targets, which were found only by the integrated model, were confirmed by the miRanda database. However, only 7 targets found by the naïve model only were supported by the miRanda database. 49 suggested targets found by both the integrated and the naïve model were supported by the MirTarBase dataset. 37 of the 49 suggested targets found by both the integrated and the naïve model were supported by strong experimental evidences according to the MirTarBase dataset. None of the suggested target found by the integrated model only were supported by strong experimental evidences according to the MirTarBase dataset. 2 of the suggested targets found by the naïve model only were supported by strong experimental evidences according to the MirTarBase dataset. Found miRNA-protein pairs with literature support, sorted by ascending order of adjusted p-values from the integrated model and the naïve model, were listed in Table 3.6.

Table 3.5| Analysis results by the naïve and the integrated model compared with the miRanda and the MirTarBase based on TCGA breast cancer dataset. Integrated model found significantly more potential miRNA & non-phosphorylated protein pairs (P<0.0001 according to the McNemar's test). Results were compared with miRNA targets identified by the miRanda algorithm and supported by the MirTarBase. Numbers in parentheses are percentage of pairs among total pairs which were found by either the naïve model method or the integrated model method.

Method[1]	All protein arrays found as miRNA targets	None phosphorylated protein arrays	Found in the miRanda	Found in the MirTarBase	Found in the MirTarBase[2]
Naïve+ Integrated+	4930 (94.63%)	3306 (95.25%)	449 (94.13%)	49 (96.08%)	37 (94.87%)

Naïve+ Integrated-	51 (0.98%)	37 (1.07%)	7 (1.47%)	2 (3.92%)	2 (5.13%)
Naïve- Integrated+	229 (4.39%)	128 (3.69%)	21 (4.40%)	0 (0%)	0 (0%)
Naïve- Integrated-	51319	32212	4933	634	304

1. Naïve+ Integrated+: Pairs found by both the integrated method and the naïve method; Naïve+ Integrated-: Pairs found by the naïve method only; Naïve- Integrated+: Pairs found by the integrated method only; Naïve- Integrated-: Pairs found by neither the integrated method nor the naïve method;
2. Supported by strong experimental evidences

Table 3.6| miRNA-protein corresponded genes pairs with literature support in TCGA breast cancer dataset. A number "1" was marked under the column for pairs found by the naïve model, integrated model, the MirTarBase, MirTarBase with strong experimental evidences (listed on the top) or the miRanda. Pairs were sorted by ascending order of adjusted p-values from the integrated model and the naïve model, respectively.

M1[1]	M2[1]	miRNA	Corresponding genes	P-values from M1	P-values from M2	Mir Tar Base	Mir Tar Base[2]	miRanda	References PMID
1	1	hsa-mir-99a	IGF1R	8.07E-11	6.07E-11	1	1		21687694
1	1	hsa-mir-18b	ESR1	2.58E-11	1.64E-10	1	1		19684618
1	1	hsa-mir-10b	NOTCH1	5.63E-07	3.89E-07	1	1		23034333
1	1	hsa-mir-101-1	PTGS2	7.95E-07	7.80E-07	1	1	1	19133256
1	1	hsa-mir-101-1	STMN1	1.62E-05	2.96E-06	1	1		23071542
1	1	hsa-mir-125b-2	BCL2	5.74E-06	4.31E-06	1	1		22293115
1	1	hsa-let-7c	BCL2L1	6.93E-06	4.65E-06	1	1		20347499
1	1	hsa-let-7c	BCL2L1	1.42E-05	8.85E-06	1	1		20347499
1	1	hsa-mir-125b-2	ERBB3	2.38E-05	1.84E-05	1	1		17110380
1	1	hsa-mir-100	IGF1R	2.00E-05	1.86E-05	1	1		21643012
1	1	hsa-mir-143	KRAS	3.10E-05	3.06E-05	1	1		19137007
1	1	hsa-let-7c	MYC	6.28E-05	4.68E-05	1	1	1	17877811
1	1	hsa-mir-143	AKT1-3	1.20E-04	7.33E-05	1	1		23104321
1	1	hsa-mir-125b-2	RAF1	8.43E-05	8.39E-05	1	1		19825990
1	1	hsa-mir-19a	ESR1	1.83E-05	1.03E-04	1	1		20080637
1	1	hsa-mir-10b	CDKN1A	1.18E-04	1.23E-04	1	1		21471404
1	1	hsa-mir-19b-1	ESR1	3.53E-05	1.70E-04	1	1		19706389
1	1	hsa-mir-130a	ESR1	1.46E-04	2.15E-04	1	1		21712254
1	1	hsa-mir-125b-2	ERBB2	2.27E-04	2.25E-04	1	1		19825990
1	1	hsa-mir-125b-1	BCL2	3.03E-04	2.70E-04	1	1		22293115

1	1	hsa-mir-125b-2	AKT1-3	6.41E-04	5.96E-04	1	1		18649363
1	1	hsa-mir-125b-2	BAK1	7.85E-04	7.48E-04	1	1		23497288
1	1	hsa-mir-222	ESR1	6.40E-04	1.32E-03	1	1		18790736
1	1	hsa-mir-199a-1	SMAD4	2.68E-03	2.10E-03	1	1		22819820
1	1	hsa-mir-483	SMAD4	3.17E-03	2.34E-03	1	1	1	21112326
1	1	hsa-mir-376c	IGF1R	2.60E-03	2.36E-03	1	1		22747855
1	1	hsa-mir-494	BCL2L11	4.05E-03	2.90E-03	1	1		23012423
1	1	hsa-let-7a-2	EGFR	3.24E-03	3.06E-03	1	1		23032975
1	1	hsa-mir-221	ESR1	2.96E-03	3.32E-03	1	1		18790736
1	1	hsa-let-7a-1	EGFR	3.54E-03	3.35E-03	1	1		23032975
1	1	hsa-mir-143	BCL2	3.88E-03	3.63E-03	1	1		19843160
1	1	hsa-mir-19b-2	ESR1	1.13E-03	4.37E-03	1	1		19706389
1	1	hsa-let-7a-3	EGFR	4.92E-03	4.65E-03	1	1		23032975
1	1	hsa-mir-144	PTEN	5.06E-03	4.85E-03	1	1		23125220
1	1	hsa-mir-139	IGF1R	6.13E-03	6.16E-03	1	1		22580051
1	1	hsa-mir-21	BCL2	7.05E-03	6.78E-03	1	1		17072344
1	1	hsa-mir-21	MSH2	6.86E-03	8.16E-03	1	1	1	18591254
1		hsa-mir-217	PTEN	6.36E-03	9.39E-03	1	1		20216554
1		hsa-mir-204	BIRC2	8.41E-03	9.92E-03	1	1		21282569
1	1	hsa-mir-101-1	MAP2K1	5.34E-07	4.14E-07	1			20371350
1	1	hsa-mir-30a	EGFR	1.12E-05	5.37E-06	1		1	18668040
1	1	hsa-mir-7-3	CAV1	3.59E-05	3.50E-05	1		1	19073608
1	1	hsa-mir-99a	RB1	6.44E-05	4.17E-05	1			23622248
1	1	hsa-mir-101-1	MSH2	5.47E-05	4.82E-05	1			20371350
1	1	hsa-mir-99a	RPS6	8.48E-05	6.49E-05	1			23622248
1	1	hsa-mir-100	RB1	1.32E-04	1.11E-04	1			23622248
1	1	hsa-mir-7-2	CAV1	3.75E-04	3.69E-04	1		1	19073608
1	1	hsa-let-7c	FOXO3	8.91E-04	7.84E-04	1		1	23622248
1	1	hsa-let-7b	EEF2	3.67E-03	2.72E-03	1			23622248
1	1	hsa-mir-21	PTK2	3.52E-03	3.34E-03	1		1	18591254
1	1	hsa-mir-132	GATA3	4.06E-03	4.01E-03	1			17612493

1. M1 stands for the naïve model and M2 stands for the integrated model;
2. Supported by strong experimental evidences;

3.7. Discussion

The traditional way to detect direct targets of miRNA using miRNA-mRNA experiment method is limited, due to the fact that miRNAs may regulate their targets post-transcriptionally. In addition, other computational methods, which were based on optimal sequence complementarity of miRNA and mRNA, suffer from large percentage of false positives and of limited practical use. Taking the advantage of recent technique advance in measuring of miRNA expression and protein concentration levels in a high-throughput scale, we proposed to search for potential miRNA targets through a nonlinear hierarchical model. Computationally, this integrated model measures the correlation between miRNA and its targeting protein without making estimation of protein expression levels first as in the naïve method. We used both simulation studies and an application to the real data to compare our proposed method and the naïve method. Our simulation results suggested that both integrated and naïve methods can well control their type-I errors with sufficient sample size, while the integrated method consistently showed higher detection powers than the naïve method under different scenarios, particularly when the protein intensity values were located close to the saturation point or the background noise level. In the real data example, our proposed integrated method detected much more potential miRNA targets than the naïve method. Furthermore, the number of potential miRNA targets, which can be confirmed by computational methods or literatures, is larger in the integrated method than that in the naïve method.

A significant association between a miRNA and protein can be either direct or indirect. For example, a miRNA may directly target and degrade a transcription factor (TF), which in turn induces indirect cascading effects of down-regulating the TF's target genes. The association analyses from the simple or our integrated model would reveal both direct and indirect associations. In contrast, the other computer-based algorithms, e.g. miRanda, can only predict direct miRNA targets based on sequence comparison. In the real data analyses (Tables 3.3 and Table 3.5), the relatively smaller percentage of overlap between our findings and the miRanda database suggests that our algorithm may detect more indirect targets. This is showed our algorithm is more powerful, as demonstrated by our simulation studies, and hence is capable of detecting smaller indirect associations. With the cross-reference to the miRanda database, those direct miRNA targets of more biological relevance could be filtered out to serve as top candidates for further biological validations. It is worth noting that our algorithm can indeed detect more direct miRNA targets in absolute number. Also, in Tables 3.3 and 3.5, the results were based on a FDR of 10% for the multiple test adjustment; however, we also checked a FDR at 5% level and found the conclusion remained the same. That is, the proposed integrated method found more miRNA targets that appear in other existing databases, demonstrating its advantage over the naïve method.

Unknown parameters in our proposed model were estimated within the maximum likelihood framework. Using the asymptotic properties of maximum likelihood estimates, test statistics were straightforward to construct. However, some improvement can be made to further improve the proposed model. For example, we assumed a linear relationship between miRNA and protein to directly compare with the naïve method and to illustrate our model using simple examples, but in reality, the relationship between miRNAs and proteins could follow a nonlinear relationship, such as a dose-response curve. In this case, $f(z_i)$ can be replaced by other parametric or nonparametric functions. With some simple modifications, our model can be easily extended to relax these assumptions. Additionally, in this Chapter the random error terms for different dilution steps were set to be independent and identically distributed as proposed in other RPPA analysis papers (Hu et al., 2007; Tabus et al., 2006; Yang & He, 2011). However, it is possible that the errors may be highly correlated. In this case, more complicated dependence matrix among serial dilution steps can also be readily incorporated into our model framework.

Chapter 4. A Semi-parametric Integrated Method for Detecting MicroRNA Target Proteins

4.1. Motivation

Even though the parametric integrated model (PIM) can better estimate the correlation between miRNA and protein than the naïve model, we need to be aware that the PIM was built under a critical assumption that the protein intensity curve follows a parametric sigmoidal shape. Thus, if the real response curve does not have such a function format, bias could exist when the PIM is applied. Furthermore, the parametric model is relative sensitive to initial values in parameter estimation, i.e., a poor set of initial values could lead to model failure or convergence to unreasonable locations. Therefore, more robust approaches are still in need.

In this chapter, we will propose a semi-parameter integrated model (SPIM), which fits the response curve through a B-spline function. Owing to the good properties of B-splines, the model is easy to be constructed and the fitting accuracy can be adjusted by the degree of spine functions or the number of knots. To illustrate the performance of the SPIM, simulation studies have been performed. Again, the model has been applied to TCGA datasets to demonstrate its practical usage.

4.2. Statistical Model and Hypothesis Testing

The overall model is considered to be semi-parametric because the link function between protein and miRNA concentrations is still assumed to be linear and parametric, even though we relax the parametric assumption for the RPPA response curve. Such a model should fit the RPPA data more flexibly and further improve the detection power of miRNA targets in specific cases.

Previously, a few methods have been established to solve such a nonlinear semi-parametric problem. Particularly, Kacrcher et.al. proposed a general semi-parametric nonlinear mixed effects model (Karcher & Wang, 2001) in which they used smoothing spline ANOVA decomposition defined on general domains. However, the computation of the log-likelihood function by using Markov chain Monte Carlo is really time consuming. Later, Elmi (Elmi, Ratcliffe, Parry, & Guo, 2011) proposed a general semi-parametric model by using the idea of B-splines with a three-step algorithm which estimates B-spline coefficients, fix effects and variance components in each step, respectively. Fixing the knots of a B-spline function, it becomes a linear combination of peace-wise polynomial functions with unknown coefficients, and the semi-parametric model can be solved through a likelihood framework. Elmi's semi-parametric model is relatively easy to construct and estimates are with good property through maximal likelihood estimation.

With a similar idea, our semi-parametric integrated model (SPIM) is a nature extension of our parametric integrated model (PIM) by replacing the sigmoidal function with a B-spline function. We applied a modified algorithm based on Elmi's in this chapter to improve the computational efficiency.

Similar to the PIM, our basic model is established as following:

$$y_{ij} = g\left(x_i, l_j, \boldsymbol{\beta}\right) + \varepsilon_{ij}$$

$$x_i = f(z_i) = \alpha_1 + \alpha_2 z_i + \eta_i,$$

$$\eta_i \sim N(0, \sigma_0^2), \varepsilon_{ij} \sim N(0, \sigma_1^2), \eta_{i'} \perp \varepsilon_{ij}$$

$$i, i' = 1,2, \dots, I \; and \; j = 1,2, \dots, 5$$

where y_{ij} is the observed protein intensity level from the ith sample and under the jth dilution, z_i is the log-transformed miRNA level from the ith sample, and x_i is the corresponding log-transformed protein level, $i = 1,2, \dots, I \; and \; j = 1,2, \dots, 5$. $l_j = 3 - j$ is a dilution index in the jth dilution step. ε_{ij} is the error term assumed to have a normal distribution with mean 0 and variance σ_1^2 and η_i is the error term in the protein-miRNA link function with a normal distribution $N(0, \sigma_0^2)$. We assume $\eta_{i'}$ and ε_{ij} are independent. Now the $g(.)$ is a function that can be modeled with B-spline functions:

$$g(x_i, l_j, \boldsymbol{\beta}) = \boldsymbol{B}(x_i + l_j)'\boldsymbol{\beta}$$

where $\boldsymbol{B}(t) = \{B_1(t), B_2(t), B_3(t), \dots B_{n+k}(t)\}$ is a vector of B-spline basis functions and $\boldsymbol{\beta} \in \boldsymbol{R}^{n+k}$ is the B-spline coefficient vector where n+1 is the number of internal knots and k denotes the degree of the B-spline function. More specifically, a quadratic B-spline function with 4 internal knots will be utilized in the following analysis.

With a similar model structure as the PIM, the unknown parameters of the SPIM can be solved under a likelihood scale. We center the miRNA data before fitting the model so that α_1 can be set to zero to cancel one degree of freedom in fitting the SPIM. The likelihood function for $\boldsymbol{Y}$ and $\boldsymbol{Z}$ can be written as their joint probability function:

$$L(\boldsymbol{\phi}, \sigma_0, \sigma_1 | \boldsymbol{Y}, \boldsymbol{Z}) = \prod_{\substack{i=1,2\dots I \\ j=1,2,..J}} \int_{-\infty}^{+\infty} f_{\epsilon_{ij}}(y_{ij} | z_i, \boldsymbol{\phi}, \eta_i, \sigma_1) q_{\eta_i}(\eta_i | \sigma_0) d\eta_i$$

where $\boldsymbol{\phi} = \{\alpha_2, \boldsymbol{\beta}\}$ is a vector of unknown parameters included in function $f(.)$ and $g(.)$, $\boldsymbol{Y} = \{y_{ij} \,|\, i = 1,2 \dots I, j = 1,2 \dots 5\}$ represents the RPPA intensity levels and $\boldsymbol{Z} = \{z_i | i = 1,2 \dots I\}$

represents the log-transformed miRNA expression levels. The number of total unknown parameters would be 8 (i.e. $3 + n + k$).

Since the response curve, intuitively, is a monotonically increasing function, we set a monotonically increasing constraint in our B-spline representation. That is, $\beta_i \geq \beta_j$ when $i \geq j, i, j = 0,1, \dots, k + n - 1$ (De Boor, 1978). The constraints can be set up naturally by transformed parameters $\{b_i\}$ with the following formula:

$$b_i = \begin{cases} \log(\beta_{i+1} - \beta_i) & i = 1,2, .. n + k - 2 \\ \beta_0 & i = 0 \end{cases}$$

By using $\{b_i\}$ to replace B-spline coefficients $\{\beta_i\}$ in the likelihood function, the monotonicity feature of the response curve is guaranteed.

We construct a one-sided test since a negative correlation is expected when a miRNA regulates a protein:

$$H_0: \alpha_2 = 0 \; vs \; H_a: \alpha_2 < 0$$

And the Wald test is used in our simulation and real data example with the test statistic:

$T.S. = \frac{\hat{\alpha}_2}{I(\hat{\boldsymbol{\phi}})_{\alpha_2}^{-\frac{1}{2}}} \sim N(0,1) \; under \; H_0$, where $I(\hat{\boldsymbol{\phi}})$ represent the fisher information matrix of the likelihood function. The null hypothesis is rejected when the p-value is above the present significance level 0.05.

4.3. Computational Algorithm and Issues

With a three-step iteration in Elmi's algorithm, the B-spline coefficients, fixed effects and variance components can be estimated in each step, respectively. However, the initial value setting and B-spline knots re-selection in every iteration steps make the computational algorithm less efficiency and even unable to converge for our case. In this section, we will introduce our computational algorithm and also discuss issues related to it.

The strategy to solve the maximum likelihood estimators, unknown parameters $\{\alpha_2, \boldsymbol{\beta}, \sigma_0, \sigma_1\}$, of SPIM is similar to the one of the PIM. The adaptive Gaussian quadrature (AGQ) method is used to approximate the integral of η_i, and the quasi-Newton method is applied in maximizing the likelihood function. However, different from the PIM, a two-step optimization strategy is utilized here to optimize the objective likelihood function. The two-step optimization strategy reduces the computational time enormously by breaking a high dimensional problem to several lower dimensional ones. Also, the situation of singular Hessian matrixes happens more frequently to a high dimensional problem, thus a two-step algorithm in our cases will have much less cases of negative variance estimates of α_2. Details of the algorithm is introduced in Section 4.3.2.

4.3.1. Initial Value Selection

Good initial values will greatly reduce computational time for our algorithm, thus they need to be carefully set up. The initial values of B-spline coefficients are set from the RPPA nonparametric method introduced in section 2.2.2 (Hu et al., 2007), and we generate initial α_2, σ_0 from the slope and the estimated mean square error (MSE), respectively, of a linear regression fitting centered protein concentrations from the nonparametric RPPA quantification method and centered miRNA levels. Initial σ_1 is estimated from the square root residual sum of square of protein intensity level. Actually, the way to estimate initial values for SPIM is a semi-parametric naïve analogue to the naïve method we described in Chapter 3, so it will be called as the semi-paramedic naïve method in the following article. The naïve method introduced in chapter 2 and 3 will be called as the parametric naïve method hereafter.

4.3.2. Computational Algorithm

Our algorithm of SPIM can be summarized as four steps showed below:

Step 0: Generate initial values: $\alpha_2^{(0)}, \boldsymbol{\beta}^{(0)}, \sigma_0^{(0)}, \sigma_1^{(0)}$ and knots of B-Splines based on the RPPA nonparametric quantification method. Calculate $\boldsymbol{b}^{(0)}$ for the monotonically increasing constrains and the initial log-likelihood function; knots of the B-spline function are equally spaced in percentile levels from the RPPA nonparametric quantification method. MiRNA levels are centered before fitting the model.

Step 1: In the nth iteration step, fixing $\alpha_2, \sigma_0, \sigma_1$, update $\boldsymbol{b}^{(n)}$by using the maximum-likelihood estimation (MLE). The AGQ method is used to approximate the likelihood function value and the quasi-Newton method is employed to optimize the approximated log-likelihood function. Specifically, R function *optim* with BFGS method, a quasi-Newton method, is used to optimize the approximated likelihood function value.

Step 2: Fixing $\boldsymbol{b}$**,** update $\alpha_2^{(n)}, \sigma_0^{(n)}, \sigma_1^{(n)}$ by using the MLE and calculate the log-likelihood value corresponding to them. The AGQ method and the quasi-Newton method are also employed.

Step 3: Stop the iteration and claim convergence when $\left|\frac{loglikelihood^{(n)}-loglikelohood^{(n-1)}}{loglikelihood^{(n-1)}}\right| < 10^{-5}$, otherwise, go back to step (1).

The algorithm has been written in an R code.

4.4. A Simulation Study

To compare with PIM, SPIM was applied on the data generated in middle part of sigmoidal curve in chapter 3 (called Scenario 1 in this chapter) and one newly generated data based on an artificial, monotonic response curve constructed by a truncated line with a positive slope (called Scenario 2 in this chapter). The function to generate Scenario 2 is showed below:

$$y_{ij} = 3000 + 1000(x_i + l_j) * I_{x_i+l_j>0} + \varepsilon_{ij}$$

$$x_i = f(z_i) = \alpha_2 z_i + \eta_i,$$

$$\eta_i \sim N(0,1), \varepsilon_{ij} \sim N(0, 500^2), \eta_{i'} \perp \varepsilon_{ij}$$

$$i, i' = 1,2, \dots, I \; and \; j = 1,2, \dots, 5$$

where 3000 was the base level of imaging intensity and 1000 was the slope once the relative *log2*-transformed protein level above 0. The miRNA expression level was generated using the same distribution in chapter 3 to mimic the real data. The correlations between miRNAs and protein expression levels, as characterized by α_2, were examined under both the null hypothesis (True $\alpha_2 = 0$) and the alternative hypothesis (True $\alpha_2 = -0.3$). 1000 simulations (I=1000) were carried out for each parameter setting under different sample sizes (N=20, 50, 100 and 300). The type-I error was set to be 0.05.

Results from SPIM and PIM were reported. In additional, the parametric and the semi-parametric naïve methods were used as references. The SPIM can successfully test α_2 among all cases based on our initial values setting. The PIM can successfully test over 98% cases with samples in simulation scenario 1. However, PIM only successfully tested average 80% cases in scenario 2 with sample size 20, and the rate decreased to a much lower number when sample size was 300. Those failures were due to negative variance estimates of α_2.

The upper part of Figure 4.1 illustrated the detection powers of four methods over different sample sizes, and the lower part of Figure 4.1 illustrated the false positive rates (FPR) of four methods. The SPIM can correctly control type-I error when sample size was sufficient (e.g. sample size=300, False positive rate =0.0530) in scenario 1, of which most protein intensity values fell on the middle part and the response curve was in S-shape. Also, the SPIM had a slightly lower power than the PIM (detection power=0.9830 vs. 0.9832, Table 4.1) with sample size 300 when correlations between miRNAs and proteins existed. However, when sample size was small, especially as small as 20, the SPIM could not well control type-I error (FPR=0.1349, Table 4.1). This could result from an over fitting of response curve in small sample size, which is a typical drawback of non-parametric (semi-parametric) methods. Point estimates in terms of $\hat{\alpha}_2$,$\hat{\sigma}_0$ and $\hat{\sigma}_1$ from cases solved by both the SPIM and the PIM are very close to each other (Table 4.2).

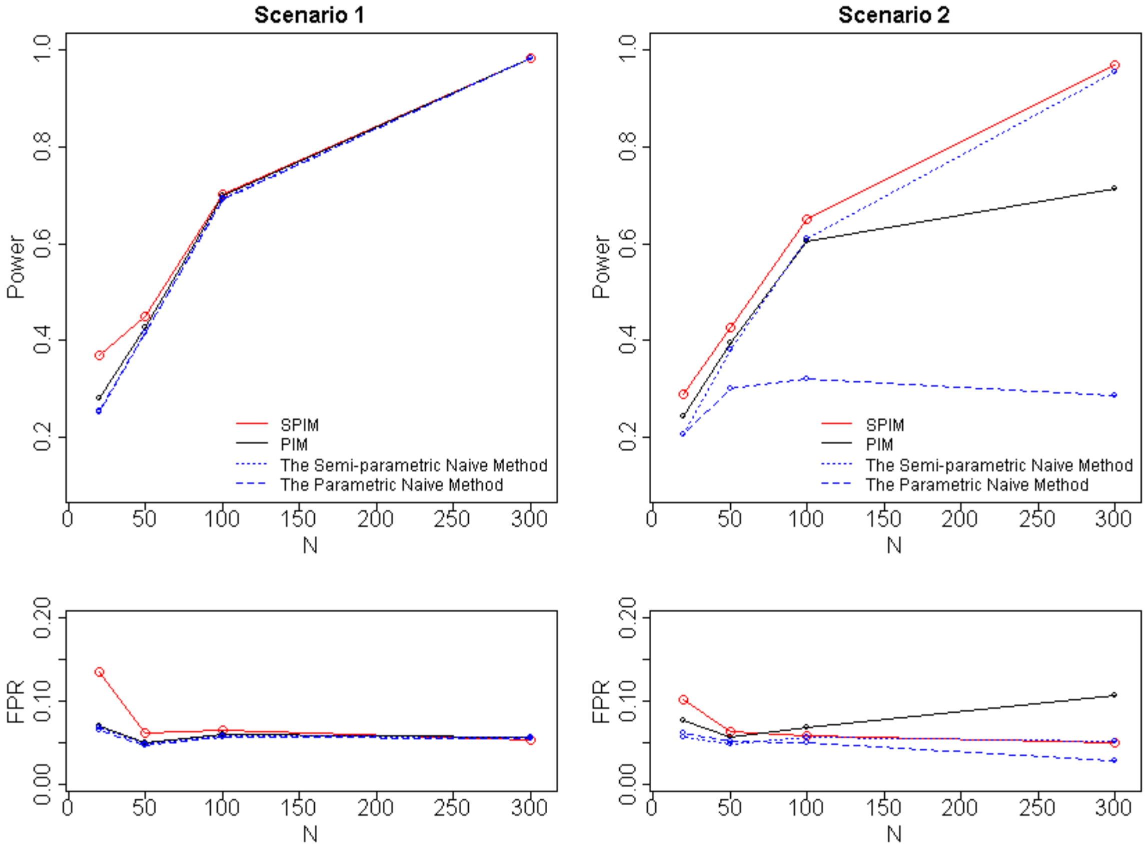


Figure 4.1| Curves of false positive rate (FPR) and detection power of four methods by simulation sample size. (Upper) the detection power of the four methods over different sample size in two scenarios; (Lower) the type-I errors (FPR) of methods in two scenarios. The SPIM and the PIM had similar detection power in scenario 1 when sample size was large; the SPIM had significant higher detection power than the PIM in scenario 2 when sample size was large. Type-I errors of the SPIM got controlled when sample size was greater than 50.

Table 4.1| Detection power or false positive rate if true $\alpha_2 = 0$ in simulation scenario 1; the SPIM and the PIM had similar performances when sample size was as large as 300; SPIM got type-I error controlled when sample size was over 50.

True α_2	Sample Size	The semi-paramedic Naïve method	SPIM	The Naïve method	PIM
-0.3	300	0.9830	0.9830	0.9832	0.9832
	100	0.6930	0.7020	0.6915	0.6985
	50	0.4160	0.4489	0.4158	0.4259
	20	0.2520	0.3696	0.2540	0.2790
0	300	0.0570	0.0530	0.0553	0.0564
	100	0.0580	0.0650	0.0567	0.0598
	50	0.0490	0.0611	0.0472	0.0503
	20	0.0680	0.1349	0.0651	0.0701

Table 4.2| Table for the detailed point estimates of shared unknown parameters in the SPIM and the PIM (Scenario 1). Mean values and standard errors (in the parenthesis) were reported; all methods had similar estimation ability in terms of small estimation bias and variations.

N	Method[1]	True α_2	α_2	σ_0 (True value=1)	σ_1 (True value=500)
20	3	0	-0.0061(0.0118)	0.9848(0.0054)	482.0903(1.2956)
	2	0	-0.0064(0.0117)	0.9366(0.0052)	487.4254(1.2434)
	1	0	-0.0062(0.0117)	0.9298(0.0051)	490.8855(1.2402)
	0	0	-0.0063(0.0117)	0.9809(0.0053)	463.6075(1.3352)
50	3	0	0.0052(0.0065)	0.9957(0.0033)	484.4161(0.8455)
	2	0	0.0053(0.0064)	0.9783(0.0034)	498.9683(0.8156)
	1	0	0.0053(0.0064)	0.9719(0.0032)	495.9713(0.7876)
	0	0	0.0054(0.0064)	0.9923(0.0032)	457.5466(0.8527)
100	3	0	-0.0034(0.0047)	0.9988(0.0022)	486.6351(0.6572)
	2	0	-0.0032(0.0047)	0.9928(0.0024)	506.0391(0.6516)
	1	0	-0.0033(0.0047)	0.9863(0.0022)	498.4465(0.5748)
	0	0	-0.0032(0.0047)	0.997(0.0022)	455.1726(0.6192)
300	3	0	-6e-04(0.0027)	0.9992(0.0014)	493.6454(0.5503)
	2	0	-6e-04(0.0027)	1.0097(0.002)	517.345(0.5095)
	1	0	-7e-04(0.0027)	0.9953(0.0014)	499.4539(0.3246)
	0	0	-7e-04(0.0027)	0.9998(0.0014)	451.1762(0.3433)
20	3	-0.3	-0.3075(0.0118)	0.9852(0.0054)	484.3206(1.2927)
	2	-0.3	-0.3067(0.0117)	0.9375(0.0053)	488.7848(1.239)
	1	-0.3	-0.3071(0.0117)	0.93(0.0051)	491.4855(1.2291)
	0	-0.3	-0.3067(0.0117)	0.9811(0.0053)	464.8493(1.3814)
50	3	-0.3	-0.295(0.0065)	0.9954(0.0033)	486.0782(0.8563)

	2	-0.3	-0.2941(0.0064)	0.9789(0.0034)	500.5824(0.8356)
	1	-0.3	-0.2945(0.0064)	0.9719(0.0032)	496.5411(0.7847)
	0	-0.3	-0.2942(0.0064)	0.9923(0.0032)	458.0395(0.8593)
100	3	-0.3	-0.3039(0.0047)	0.999(0.0022)	489.1494(0.6984)
	2	-0.3	-0.3033(0.0047)	0.9947(0.0025)	508.1468(0.6753)
	1	-0.3	-0.3036(0.0047)	0.9861(0.0022)	498.7687(0.5616)
	0	-0.3	-0.3033(0.0047)	0.9968(0.0022)	455.2566(0.5977)
300	3	-0.3	-0.3002(0.0027)	0.9991(0.0014)	496.7987(0.5852)
	2	-0.3	-0.3002(0.0027)	1.0149(0.0022)	519.7791(0.5156)
	1	-0.3	-0.3006(0.0027)	0.9953(0.0014)	499.8458(0.3148)
	0	-0.3	-0.3005(0.0027)	0.9996(0.0014)	451.8615(0.3463)

1, method 0 to 3 were the parametric naïve method, the PIM method, the SPIM method and the semi-parametric naïve method, respectively;

In scenario 2, simulation results showed that the SPIM had better performance than the PIM when the response curve is not S-shaped (the right part of Figure 4.1). The SPIM can better control the type-I error than the PIM (sample size=300, FPR=0.0500 vs. 0.1058, respectively, Table 4.3), and had higher detection power than the PIM (sample size=300, detection power=0.9710 vs. 0.7143, respectively, Table 4.3). In additional, the SPIM had higher power than the semi-parametric naïve method (sample size=300, detection power=0.9710 vs. 0.9550, respectively, Table 4.3). In Table 4.4, we compare the point estimates of shared unknown parameters in the PIM and the SPIM from cases solved by both types of methods. The parametric based methods did not correctly estimate α_2 values while the semi-parametric types of methods did, especially when sample size is big, e.g. 300. Also, the PIM was not able to successful test α_2 in many cases in scenario 2. The simulation study suggested that the SPIM method is more robust and powerful than the PIM method when response curve are not sigmoidal.

Table 4.3| Detection power or false positive rate if true $\alpha_2 = 0$ in simulation scenario 2. The SPIM has much better performance than the PIM under this condition in terms of higher detection power and better controlled type-I error.

True α_2	Sample Size	The semi-paramedic Naïve method	SPIM	The Naïve method	PIM
-0.3	300	0.9550	0.9710	0.2857	0.7143
	100	0.6100	0.6496	0.3213	0.6041
	50	0.3800	0.4259	0.2995	0.3947
	20	0.2040	0.2880	0.2051	0.2415
0	300	0.0510	0.0500	0.0288	0.1058
	100	0.0570	0.0590	0.0504	0.0680
	50	0.0490	0.0630	0.0522	0.0567
	20	0.0560	0.1020	0.0618	0.0772

Table 4.4| Table for the detailed point estimates of shared unknown parameters in four methods and their standard error (Scenario 2). The parametric based methods did not correctly estimate α_2 values while the semi-parametric types of methods did, especially when sample size is big.

N	Method[1]	True α_2	α_2	σ_0 (True value=1)	σ_1 (True value=500)
20	3	0	0.0023(0.0141)	1.0995(0.0078)	507.2353(1.6754)
	2	0	-0.0012(0.0132)	0.9138(0.0067)	494.8362(1.3455)
	1	0	-1.6318(1.918)	10.355(4.7727)	507.3892(2.4273)
	0	0	-0.3242(0.3732)	7.0797(1.4518)	492.5723(1.7325)
50	3	0	-0.0018(0.0088)	1.1062(0.0056)	486.6809(1.1655)
	2	0	-0.003(0.0083)	0.9541(0.0047)	497.7188(0.9644)
	1	0	-2.2536(1.6159)	16.6359(4.2935)	525.0788(4.1128)
	0	0	-0.8754(0.6847)	21.2222(4.1322)	480.3866(1.3207)
100	3	0	5e-04(0.0075)	1.1087(0.005)	478.9898(0.7994)
	2	0	0.0025(0.0071)	0.9692(0.0038)	499.1161(0.798)
	1	0	0.6864(1.3524)	31.2586(9.0262)	528.3946(4.6303)
	0	0	-0.4422(0.7166)	42.0133(8.7666)	473.7157(1.0266)
300	3	0	-0.0041(0.0104)	1.1307(0.0072)	479.0157(0.9882)
	2	0	-0.0087(0.0098)	0.9791(0.0057)	502.4604(0.9904)
	1	0	-5.7375(5.5085)	344.8306(65.6849)	611.8913(20.2568)
	0	0	-5.4565(5.5833)	366.5218(65.0571)	468.3396(1.0279)
20	3	-0.3	-0.2787(0.0144)	1.1037(0.0083)	507.334(1.707)
	2	-0.3	-0.2878(0.0135)	0.9175(0.007)	494.382(1.3599)
	1	-0.3	-1.0032(0.7571)	3.5806(1.2039)	504.2361(2.1717)

	0	-0.3	-1.0644(0.4188)	6.5531(1.3729)	491.699(1.8231)
50	3	-0.3	-0.2956(0.0089)	1.0993(0.0057)	485.9764(1.1347)
	2	-0.3	-0.2904(0.0085)	0.9561(0.0049)	497.8892(0.9731)
	1	-0.3	2.0399(2.2698)	16.4983(5.2172)	517.2476(2.7556)
	0	-0.3	-1.2422(0.5922)	19.2075(4.3996)	479.8681(1.2262)
100	3	-0.3	-0.306(0.0079)	1.1094(0.005)	479.4046(0.811)
	2	-0.3	-0.2947(0.0074)	0.9688(0.0039)	499.3543(0.8044)
	1	-0.3	-3.078(1.2961)	33.652(9.9803)	531.3021(5.2062)
	0	-0.3	-4.4507(1.0441)	49.311(10.1107)	473.7148(1.0005)
300	3	-0.3	-0.3061(0.0107)	1.1415(0.0085)	479.6509(1.1017)
	2	-0.3	-0.2943(0.01)	0.9837(0.0057)	502.3416(1.1047)
	1	-0.3	-26.9698(15.6214)	513.4009(100.7311)	622.6092(22.1766)
	0	-0.3	-30.3285(5.993)	538.1615(100.2324)	468.3017(1.1993)

1, method 0 to 3 were the parametric naïve method, the PIM method, the SPIM method and the semi-parametric naïve method, respectively;

4.5. Real Data Examples

4.5.1. Analysis of TCGA Ovarian Cancer (OV) Data

The SPIM was applied to the ovarian cancer dataset from TCGA program introduced in Chapter 3.6. In this dataset, there were 333 ovarian cancer samples with both miRNA and RPPA data available. The SPIM was applied onto the 574 miRNA-protein pairs with literature supported in MirTarBase (Hsu et al., 2014). FDR at 10% was used to adjust for multiple testing and the results were listed in Table 4.5.

Totally 27 targets were suggested by either the PIM method or the SPIM method (Table 4.6). 7 targets were found by the SPIM only and 4 targets were found by the PIM only. Two suggested targets found by the SPIM only were supported by strong experimental evidences according to the MirTarBase database, and one of the suggested targets found by the PIM only were supported by strong experimental evidences according to the MirTarBase database. Among the 297 targets with strong experimental evidences, 75 were identified by the miRanda algorithm.

12 targets were supported by the PIM or SPIM but not suggested by the miRanda. This implies our methods discover additional miRNA targets to the computational based method miRanda by studying protein-miRNA relationship. The 27 targets found by either the PIM or the SIM were listed in Table 4.6.

Table 4.5| Analysis results by the PIM and the SPIM compared with the miRanda on TCGA ovarian cancer dataset with literature supports in the MirTarBase.

Method[1]	Found in the MirTarBase	Found in the MirTarBase[2]	Found in the miRanda with strong experimental evidences
PIM+ SPIM+	16	12	2
PIM+ SPIM-	4	1	0
PIM- SPIM+	7	2	0
PIM- SPIM-	547	282	73

1. + stands for positive findings, i.e. targeted proteins with a specified miRNA, in the method, and – stands for negative results;
2. Supported by strong experimental evidences

Table 4.6 | Details of targets suggested by either the PIM or the SPIM in TCGA ovarian cancer dataset. A number "1" was marked under the column for pairs found by the PIM, the SPIM, in the MirTarBase, MirTarBase with strong experimental evidences (listed on the top) or miRanda database; pairs were sorted by ascending order of adjusted p-values from the SPIM and the PIM.

PIM	SPIM	miRNA	Corresponding genes	P-values from the PIM	P-values from the SPIM	MirTarBase	MirTarBase[1]	miRanda	References PMID
1	1	hsa-mir-150	TP53	8.30E-07	2.68E-07	1	1		23747308
	1	hsa-mir-155	KRAS	1.63E-02	3.83E-07	1	1		18668040
1	1	hsa-mir-150	NOTCH3	1.66E-08	3.98E-06	1	1	1	21551231
1	1	hsa-mir-181a-1	BCL2L11	1.84E-05	4.65E-05	1	1		20841506
1	1	hsa-mir-223	IGF1R	4.14E-05	4.00E-05	1	1		22073238
1	1	hsa-mir-139	IGF1R	4.43E-05	7.71E-05	1	1		22580051
1	1	hsa-mir-214	CTNNB1	6.06E-06	2.27E-04	1	1		23068095
1	1	hsa-mir-181a-1	CDKN1B	5.41E-04	5.75E-04	1	1		19273599
1	1	hsa-mir-18a	ESR1	6.37E-04	7.12E-04	1	1	1	19684618

1	1	hsa-mir-145	IGF1R	7.96E-04	8.84E-04	1	1		19391107
	1	hsa-mir-217	KRAS	4.41E-02	1.54E-03	1	1		20675343
1	1	hsa-mir-605	TP53	1.67E-03	1.73E-03	1	1		21217645
1	1	hsa-mir-181a-2	CDKN1B	2.62E-03	2.20E-03	1	1		23622248
1	1	hsa-mir-155	SMAD3	1.08E-03	3.76E-03	1	1		21036908
1		hsa-mir-21	MSH6	1.46E-03	1.06E-02	1	1		21078976
	1	hsa-mir-125a	EIF4EBP1	1.25E-02	3.40E-13	1			20371350
	1	hsa-mir-181a-2	KRAS	8.89E-02	1.16E-06	1			20371350
	1	hsa-mir-181a-1	KRAS	5.84E-02	9.19E-05	1			20371350
	1	hsa-mir-155	CTNNA1	1.79E-02	2.70E-04	1			18668040
	1	hsa-mir-877	EEF2K	1.02E-02	1.08E-03	1			23622248
1	1	hsa-let-7a-1	BCL2L11	1.69E-03	3.11E-03	1		1	23622248
1	1	hsa-let-7a-2	BCL2L11	1.64E-03	3.06E-03	1		1	23622248
1	1	hsa-let-7a-3	BCL2L11	1.69E-03	3.07E-03	1		1	23622248
1	1	hsa-mir-766	MAPK1	3.13E-03	3.21E-03	1			23622248
1		hsa-mir-146b	AKT1-AKT3	3.11E-03	4.21E-03	1			23622248
1		hsa-mir-221	BCL2L11	3.45E-03	5.19E-03	1		1	23622248
1		hsa-mir-7-1	CAV1	1.29E-03	2.07E-02	1		1	19073608

1, Supported by strong experimental evidences

4.5.2. Analysis of TCGA Breast Cancer (BRCA) Data

In BRCA dataset of TCGA program introduced in Chapter 3.6, there were 239 samples with both miRNA and RPPA data available. The SPIM was applied onto the 685 miRNA-protein pairs with literature supported in the MirTarBases. FDR at 10% was used to adjust for multiple testing and the results were listed in Table 4.7.

Totally 51 targets were suggested by either the PIM method or the SPIM method (Table 4.8). Three targets were found by the SPIM method only and 5 targets were found by the PIM method only. Two suggested targets found by the SPIM only were supported by strong experimental evidences according to the MirTarBase database, and 3 of the suggested targets found by the PIM only were supported by strong experimental evidences according to the MirTarBase database. Among the 343 targets with strong experimental evidences, 73 were suggested by the miRanda algorithm. Three targets were supported by all three methods. The 51 targets found by either the PIM or the SIM were listed in Table 4.8.

Table 4.7| Analysis results by the PIM and the SPIM compared with the miRanda on TCGA breast cancer dataset with literature support in the MirTarBase.

Method[1]	Found in the MirTarBase	Found in the MirTarBase[2]	Found in the miRanda with strong experimental evidences
PIM+ SPIM+	43	33	3
PIM+ SPIM-	5	3	0
PIM- SPIM+	3	2	0
PIM- SPIM-	633	305	70

1. + stands for positive findings in the method, i.e. targeted proteins with a specified miRNA, and – stands for negative results;
2. Supported by strong experimental evidences

Table 4.8| Details of targets suggested by either the PIM or the SPIM on TCGA breast cancer dataset. A number “1” was marked under the column for pairs found by the PIM, SPIM, in the MirTarBase, MirTarBase with strong experimental evidences (sorted on the top) or the miRanda; pairs were sorted by ascending order of adjusted p-values from the SPIM and the PIM.

PIM	SPIM	miRNA	Corresponding genes	P-values from the PIM	P-values from the SPIM	MirTarBase	MirTarBase[1]	miRanda	References PMID
1	1	hsa-mir-99a	IGF1R	6.07E-11	5.97E-12	1	1		21687694
1	1	hsa-mir-18b	ESR1	1.64E-10	2.37E-09	1	1		19684618
1	1	hsa-mir-10b	NOTCH1	3.89E-07	2.30E-07	1	1		23034333
1	1	hsa-mir-101-1	STMN1	2.96E-06	2.31E-06	1	1		23071542
1	1	hsa-mir-100	IGF1R	1.86E-05	9.79E-06	1	1		21643012
1	1	hsa-mir-143	KRAS	3.06E-05	1.08E-05	1	1		19137007
1	1	hsa-mir-125b-2	ERBB3	1.84E-05	1.34E-05	1	1		17110380
1	1	hsa-let-7c	MYC	4.68E-05	1.94E-05	1	1	1	17877811
1	1	hsa-mir-101-1	PTGS2	7.80E-07	2.26E-05	1	1	1	19133256
1	1	hsa-let-7c	BCL2L1	4.65E-06	2.53E-05	1	1		20347499
	1	hsa-mir-30a	SMAD1	3.27E-02	3.86E-05	1	1		22253433
1	1	hsa-let-7c	BCL2L1	8.85E-06	4.87E-05	1	1		20347499
1	1	hsa-mir-125b-2	BCL2	4.31E-06	9.83E-05	1	1		22293115

1	1	hsa-mir-143	AKT1-AKT3	7.33E-05	1.44E-04	1	1		23104321
1	1	hsa-mir-125b-2	RAF1	8.39E-05	1.64E-04	1	1		19825990
1	1	hsa-mir-19a	ESR1	1.03E-04	1.54E-04	1	1		20080637
1	1	hsa-mir-10b	CDKN1A	1.23E-04	1.36E-04	1	1		21471404
1	1	hsa-mir-125b-2	ERBB2	2.25E-04	1.62E-04	1	1		19825990
1	1	hsa-mir-19b-1	ESR1	1.70E-04	3.11E-04	1	1		19706389
1	1	hsa-mir-130a	ESR1	2.15E-04	7.46E-04	1	1		21712254
1	1	hsa-mir-125b-2	AKT1-AKT3	5.96E-04	1.08E-03	1	1		18649363
1	1	hsa-mir-125b-1	BCL2	2.70E-04	1.57E-03	1	1		22293115
1	1	hsa-mir-125b-2	BAK1	7.48E-04	1.77E-03	1	1		23497288
1	1	hsa-mir-222	ESR1	1.32E-03	1.80E-03	1	1		18790736
	1	hsa-mir-126	PGR	1.79E-01	1.95E-03	1	1		21526342
1	1	hsa-mir-376c	IGF1R	2.36E-03	2.11E-03	1	1		22747855
1	1	hsa-mir-199a-1	SMAD4	2.10E-03	2.41E-03	1	1		22819820
1	1	hsa-let-7a-2	EGFR	3.06E-03	3.18E-03	1	1		23032975
1	1	hsa-let-7a-1	EGFR	3.35E-03	3.58E-03	1	1		23032975
1	1	hsa-mir-144	PTEN	4.85E-03	3.91E-03	1	1		23125220
1	1	hsa-mir-483	SMAD4	2.34E-03	4.10E-03	1	1	1	21112326
1	1	hsa-let-7a-3	EGFR	4.65E-03	4.88E-03	1	1		23032975
1	1	hsa-mir-139	IGF1R	6.16E-03	4.97E-03	1	1		22580051
1	1	hsa-mir-19b-2	ESR1	4.37E-03	5.58E-03	1	1		19706389
1	1	hsa-mir-221	ESR1	3.32E-03	6.01E-03	1	1		18790736
1		hsa-mir-494	BCL2L11	2.90E-03	9.05E-03	1	1		23012423
1		hsa-mir-143	BCL2	3.63E-03	1.24E-02	1	1		19843160
1		hsa-mir-21	BCL2	6.78E-03	1.93E-02	1	1		17072344
1	1	hsa-mir-99a	RPS6	6.49E-05	0.00E+00	1			23622248
	1	hsa-mir-30a	PCNA	2.50E-02	1.94E-11	1			23622248
1	1	hsa-mir-101-1	MSH2	4.82E-05	1.72E-10	1			20371350
1	1	hsa-mir-101-1	MAP2K1	4.14E-07	3.57E-07	1			20371350
1	1	hsa-mir-30a	EGFR	5.37E-06	8.40E-06	1		1	18668040
1	1	hsa-mir-99a	RB1	4.17E-05	1.60E-04	1			23622248
1	1	hsa-mir-100	RB1	1.11E-04	2.47E-04	1			23622248
1	1	hsa-let-7c	FOXO3	7.84E-04	7.45E-04	1		1	23622248
1	1	hsa-mir-7-3	CAV1	3.50E-05	1.82E-03	1		1	19073608
1	1	hsa-mir-132	GATA3	4.01E-03	4.26E-03	1			17612493
1	1	hsa-mir-7-2	CAV1	3.69E-04	6.70E-03	1		1	19073608
1		hsa-let-7b	EEF2	2.72E-03	9.32E-03	1			23622248
1		hsa-mir-21	PTK2	3.34E-03	3.25E-02	1		1	18591254

1. Supported by strong experimental evidences

4.6. Discussion

In this chapter, we extend the parametric integrated model (PIM) developed in Chapter 3 to a more flexible semi-parametric integrated model (SPIM) by incorporating a nonparametric function for RPPA response curve, which relaxes the assumption of a specific sigmoid function. The performance of the SPIM is demonstrated by both simulation studies and real data analyses. According to our simulation results, the SPIM was flexible enough to fit a non-sigmoidal intensity response curve with much less power loss comparing to PIM, and yielded the type-I error when the intensity response curve is non-sigmoidal and sample size is over 50 (Scenario 2). Moreover, it had more accurate estimates of model parameters. Importantly, when the response curve is sigmoidal (Scenario 1), the SPIM achieved similar performance as PIM in terms of detection power and parameter estimates even though the PIM is slightly more efficient. In the real data example, our proposed SPIM suggested additional miRNA targets with literature support to the PIM in both OV and BRCA datasets, however, no evidence showed that the SPIM had higher discover rate than the PIM, which implied the SPIM is still not a replacement of the PIM but a complement when the RPPA response curve is not sigmoidal.

Unknown parameters in our proposed model were estimated within the maximum likelihood framework. Using the asymptotic properties of maximum likelihood estimates, test statistics were straightforward to construct. By adjusting the number of B-spline knots and order, our semi-parametric model can fit different pattern of protein concentration-intensity response curve. While more knots and higher order of B-spline functions can result in a more sensitive model, it also has more parameters to be estimate and can potentially cause over-fitting when sample size is limited.

Chapter 5. Discussion and Future Work

The RPPA technique provides a new prospect to study the miRNA targets, which could serve as potential biomarkers of different diseases. Comparing to the naïve models, our proposed integrated models (PIM and SPIM) are flexible and have higher detection powers while controlling type-I error with a sufficient sample size. In reality, more complicated relationships between miRNA and proteins could exist. For instance, if the influence from miRNA to proteins could start only when miRNA reach a certain level, such relationship can be formulated by replacing the miRNA/protein link function as

$$c = \alpha_1 + \alpha_2 * x * I_{x>a_3}.$$

where c is protein levels, x is miRNA level and α_3 is the critical value that miRNA start to regulate proteins. Currently, we assume a constant variance in the miRNA-protein link function and independence among intensity levels from different dilution steps. A non-constant error variance and different correlation structure can be implement into our models. In addition, there could be confounding variables relating to function of miRNA, our models are able to estimate the correlation between miRNA and proteins relationship after adjusting for these information.

According to our simulation studies and real data analyses, the SPIM is not a replacement of the PIM but a complement. When the real RPPA response curve follows a sigmoidal shape, the

PIM is more efficient than the SPIM. While the real response curve is not under a sigmoidal shape, the SPIM has a higher detection power and better point estimates than those of the PIM. Thus a method to classified RPPA data into two groups which favor different models before fitting the PIM and the SPIM will highly improve the efficiency and accuracy of miRNA targets screening using both methods.

Another important issue in our study is how to better estimate the FDR level. In the current analyses, we applied the traditional Benjamini-Hochberg procedure (Benjamini & Hochberg, 1995), which works well when all tests are independent. However, given the special testing structure of cross-examining multiple miRNA-protein pairs, the tests are not independent. For example, one miRNA may be used to test its correlations with several proteins, and these tests are dependent as they share the same miRNA data. Several studies have been conducted in this area. Pawitan *et al.* (Pawitan, Calza, & Ploner, 2006) have addressed the deleterious effect due to ignore biological and technical correlations. Ghosal *et al.* (Ghosal & Roy, 2011) proposed an error measure prediction method by modeling the distribution of probit transformed p-values. Bean et al. (Bean et al., 2013) estimated true negative rates in multiple testing through finite skew-mixture models. But those methods are not perfect fits for our study. Thus, finding a better way to adjust the bias of FDR estimates is an important topic to be conducted in the future.

In addition, relative miRNAs could work together to regulate the protein synthesis. Our current method detects miRNA targets through examining correlations between one specific miRNA and one specific protein. A more general model considering combination effects from multiple miRNAs could tell whether a protein is regulated by a set of miRNAs and further improves the detection power. Moreover, our methods detect the miRNA and protein pairs no matter if the effects are direct or indirect. Causal models could be used in our framework to further separate the direct or indirect targets of a miRNA. Inferences about miRNA regulation mechanism are surely enhanced by incorporating additional information such as mRNA. Alternatively, it is worthwhile to develop models that simultaneously consider miRNA, mRNA and protein information.

Reference

Baek, D., Villen, J., Shin, C., Camargo, F. D., Gygi, S. P., & Bartel, D. P. (2008). The impact of microRNAs on protein output. *Nature, 455*(7209), 64-71. doi: 10.1038/nature07242

Baek, D., Villén, J., Shin, C., Camargo, F. D., Gygi, S. P., & Bartel, D. P. (2008). The impact of microRNAs on protein output. *Nature, 455*(7209), 64-71.

Bartel, D. P. (2009). MicroRNAs: target recognition and regulatory functions. *cell, 136*(2), 215-233.

Beal, S., & Sheiner, L. (1988). Heteroscedastic nonlinear regression. *Technometrics, 30*(3), 327-338.

Beal, S. L., & Sheiner, L. B. (1981). Estimating population kinetics. *Critical reviews in biomedical engineering, 8*(3), 195-222.

Bean, G. J., Dimarco, E. A., Mercer, L. D., Thayer, L. K., Roy, A., & Ghosal, S. (2013). Finite skew-mixture models for estimation of positive false discovery rates. *Statistical Methodology, 10*(1), 46-57.

Benjamini, Y., & Hochberg, Y. (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society. Series B (Methodological)*, 289-300.

Betel, D., Koppal, A., Agius, P., Sander, C., & Leslie, C. (2010). Comprehensive modeling of microRNA targets predicts functional non-conserved and non-canonical sites. *Genome biology, 11*(8), R90.

Brennecke, J., Stark, A., Russell, R. B., & Cohen, S. M. (2005). Principles of microRNA–target recognition. *PLoS Biol, 3*(3), e85.

De Boor, C. (1978). *A practical guide to splines*. New York: Springer.

Doench, J. G., & Sharp, P. A. (2004). Specificity of microRNA target selection in translational repression. *Genes Dev, 18*(5), 504-511.

Ekins, R. P. (1998). Ligand assays: from electrophoresis to miniaturized microarrays. *Clinical chemistry, 44*(9), 2015-2030.

Elmi, A., Ratcliffe, S. J., Parry, S., & Guo, W. (2011). A B-spline based semiparametric nonlinear mixed effects model. *Journal of computational and Graphical Statistics, 20*(2), 492-509.

Enright, A. J., John, B., Gaul, U., Tuschl, T., Sander, C., & Marks, D. S. (2004). MicroRNA targets in Drosophila. *Genome biology, 5*(1), R1-R1.

Esquela-Kerscher, A., & Slack, F. J. (2006). Oncomirs—microRNAs with a role in cancer. *Nature Reviews Cancer, 6*(4), 259-269.

Ghosal, S., & Roy, A. (2011). Predicting false discovery proportion under dependence. *Journal of the American Statistical Association, 106*(495).

Golub, G., & Pereyra, V. (2003). Separable nonlinear least squares: the variable projection method and its applications. *Inverse problems, 19*(2), R1.

He, L., & Hannon, G. J. (2004). MicroRNAs: small RNAs with a big role in gene regulation. *Nature Reviews Genetics, 5*(7), 522-531.

He, X., & Ng, P. (1999). COBS: qualitatively constrained smoothing via linear programming. *Computational Statistics, 14*(3), 315-338.

Hofacker, I. L., Fontana, W., Stadler, P. F., Bonhoeffer, L. S., Tacker, M., & Schuster, P. (1994). Fast folding and comparison of RNA secondary structures. *Monatshefte für Chemie/Chemical Monthly, 125*(2), 167-188.

Hsu, S.-D., Tseng, Y.-T., Shrestha, S., Lin, Y.-L., Khaleel, A., Chou, C.-H., . . . Ho, S.-Y. (2014). miRTarBase update 2014: an information resource for experimentally validated miRNA-target interactions. *Nucleic acids research, 42*(D1), D78-D85.

Hu, J., He, X., Baggerly, K. A., Coombes, K. R., Hennessy, B. T., & Mills, G. B. (2007). Non-parametric quantification of protein lysate arrays. *Bioinformatics, 23*(15), 1986-1994. doi: 10.1093/bioinformatics/btm283

Jeffrey, S. S. (2008). Cancer biomarker profiling with microRNAs. *Nature biotechnology, 26*(4), 400-401.

Jones, K., Nourse, J. P., Keane, C., Bhatnagar, A., & Gandhi, M. K. (2014). Plasma microRNA are disease response biomarkers in classical Hodgkin lymphoma. *Clinical Cancer Research, 20*(1), 253-264.

Karcher, P., & Wang, Y. (2001). Generalized nonparametric mixed effects models. *Journal of computational and Graphical Statistics, 10*(4), 641-655.

Lee, R. C., Feinbaum, R. L., & Ambros, V. (1993). The C. elegans heterochronic gene lin-4 encodes small RNAs with antisense complementarity to lin-14. *cell, 75*(5), 843-854.

Lewis, B. P., Shih, I.-h., Jones-Rhoades, M. W., Bartel, D. P., & Burge, C. B. (2003). Prediction of mammalian microRNA targets. *cell, 115*(7), 787-798.

Lu, C., Meyers, B. C., & Green, P. J. (2007). Construction of small RNA cDNA libraries for deep sequencing. *Methods, 43*(2), 110-117.

McCaskill, J. S. (1990). The equilibrium partition function and base pair binding probabilities for RNA secondary structure. *Biopolymers, 29*(6-7), 1105-1119.

Mircean, C., Shmulevich, I., Cogdell, D., Choi, W., Jia, Y., Tabus, I., . . . Zhang, W. (2005). Robust estimation of protein expression ratios with lysate microarray technology. *Bioinformatics, 21*(9), 1935-1942. doi: 10.1093/bioinformatics/bti258

Mueller, C., Liotta, L. A., & Espina, V. (2010). Reverse phase protein microarrays advance to use in clinical trials. *Mol Oncol, 4*(6), 461-481.

Nelder, J. A., & Mead, R. (1965). A simplex method for function minimization. *The computer journal, 7*(4), 308-313.

Nishizuka, S., Charboneau, L., Young, L., Major, S., Reinhold, W. C., Waltham, M., . . . Espina, V. (2003). Proteomic profiling of the NCI-60 cancer cell lines using new high-density reverse-phase lysate microarrays. *Proceedings of the National Academy of Sciences, 100*(24), 14229-14234.

Pawitan, Y., Calza, S., & Ploner, A. (2006). Estimation of false discovery proportion under general dependence. *Bioinformatics, 22*(24), 3025-3031.

Pritchard, C. C., Cheng, H. H., & Tewari, M. (2012). MicroRNA profiling: approaches and considerations. *Nature Reviews Genetics, 13*(5), 358-369.

Reinhart, B. J., Slack, F. J., Basson, M., Pasquinelli, A. E., Bettinger, J. C., Rougvie, A. E., . . . Ruvkun, G. (2000). The 21-nucleotide let-7 RNA regulates developmental timing in Caenorhabditis elegans. *Nature, 403*(6772), 901-906.

Selbach, M., Schwanhäusser, B., Thierfelder, N., Fang, Z., Khanin, R., & Rajewsky, N. (2008). Widespread changes in protein synthesis induced by microRNAs. *Nature, 455*(7209), 58-63.

Sheiner, L. B., & Beal, S. L. (1985). Pharmacokinetic parameter estimates from several least squares procedures: superiority of extended least squares. *Journal of pharmacokinetics and biopharmaceutics, 13*(2), 185-201.

SuperCurveGUI. (2011). User Guide. from http://bioinformatics.mdanderson.org/Software/supercurve/UserGuide2.html

Tabus, I., Hategan, A., Mircean, C., Rissanen, J., Shmulevich, I., Zhang, W., & Astola, J. (2006). Nonlinear modeling of protein expressions in protein arrays. *Signal Processing, IEEE Transactions on, 54*(6), 2394-2407.

Tang, G., Reinhart, B. J., Bartel, D. P., & Zamore, P. D. (2003). A biochemical framework for RNA silencing in plants. *Genes Dev, 17*(1), 49-63.

Wikipedia. (2014). Messenger RNA --- Wikipedia *The Free Encyclopedia*. from http://en.wikipedia.org/w/index.php?title=Messenger_RNA&oldid=637717685

Xie, Z., Kasschau, K. D., & Carrington, J. C. (2003). Negative Feedback Regulation of Dicer-Like1 in Arabidopsis by microRNA-Guided mRNA Degradation. *Current Biology, 13*(9), 784-789.

Yang, J. Y., & He, X. (2011). A multistep protein lysate array quantification method and its statistical properties. *Biometrics, 67*(4), 1197-1205. doi: 10.1111/j.1541-0420.2011.01567.x

Zeng, Y., Wagner, E. J., & Cullen, B. R. (2002). Both natural and designed micro RNAs can inhibit the expression of cognate mRNAs when expressed in human cells. *Molecular cell, 9*(6), 1327-1333.

Zhang, L., Wei, Q., Mao, L., Liu, W., Mills, G. B., & Coombes, K. (2009). Serial dilution curve: a new method for analysis of reverse phase protein array data. *Bioinformatics, 25*(5), 650-654. doi: 10.1093/bioinformatics/btn663

Zhang, W., Zhang, J., Hoadley, K., Kushwaha, D., Ramakrishnan, V., Li, S., . . . Song, S. W. (2012). miR-181d: a predictive glioblastoma biomarker that downregulates MGMT expression. *Neuro-oncology, 14*(6), 712-719.

Zuker, M., & Stiegler, P. (1981). Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information. *Nucleic acids research, 9*(1), 133-148.