BIOINFORMATICS ORIGINAL PAPER

Vol. 28 no. 9 2012, pages 1223–1229
doi:10.1093/bioinformatics/bts131

Structural bioinformatics

Advance Access publication March 30, 2012

Reno: regularized non-parametric analysis of protein lysate array data

Bin Li[1], Feng Liang[1], Jianhua Hu[2,*], and Xuming He[3]

[1]Department of Statistics, University of Illinois at Urbana-Champaign, Champaign, IL, 61820, USA, [2]Department of Biostatistics, University of Texas M.D. Anderson Cancer Center, Houston, TX, 77230, USA and [3]Department of Statistics, University of Michigan, Ann Arbor, MI, 48109, USA

Associate Editor: Martin Bishop

ABSTRACT

Motivation: The reverse-phase protein lysate arrays have been used to quantify the relative expression levels of a protein in a number of cellular samples simultaneously. To avoid quantification bias due to mis-specification of commonly used parametric models, a nonparametric approach based on monotone response curves may be used. The existing methods, however, aggregate the protein concentration levels of replicates of each sample, and therefore fail to account for within-sample variability.

Results: We propose a method of regularization on protein concentration estimation at the level of individual dilution series to account for within-sample or within-group variability. We use an efficient algorithm to optimize an approximate objective function, with a data-adaptive approach to choose the level of shrinkage. Simulation results show that the proposed method quantifies protein concentration levels well. We show through the analysis of protein lysate array data from cell lines of different cancer groups that accounting for within-sample variability leads to better statistical analysis.

Availability: Code written in statistical programming language R is available at: http://odin.mdacc.tmc.edu/∼ jhhu/Reno

Contact: jhu@mdanderson.org

Supplementary information: Supplementary data are available at *Bioinformatics* online.

Received on June 23, 2011; revised on March 10, 2012; accepted on March 12, 2012

1 INTRODUCTION

Protein microarray technologies [e.g. Cahill and Nordhoff (2003); Ivanov *et al*. (2004); MacBeath and Schreiber (2000)] have been developed to measure protein concentrations in a high-throughput fashion. Extensive reviews of this technology can be found in Borrebaeck and Wingren (2007) and Poetz *et al*. (2005). A single nitrocellulose-coated array slide can measure concentrations of a common protein in hundreds of samples in the form of dilution series. The samples are hybridized and label-attached with primary and biotinylated secondary antibodies and the protein concentrations are then measured using streptavadin-linked labels that bind to the biotin. The final product of each array is an image file in which quantified spots represent the observed protein expression levels at various dilutions steps.

Protein lysate array technology has shown its promise in a number of clinical studies [e.g. Pluder *et al*. (2006); Sahin *et al*. (2007)]. In particular, its applications to various cancer studies have been documented extensively; see, for example, Cai *et al*. (2010); Carey *et al*. (2010); Cheng *et al*. (2005); Grote *et al*. (2008); Kim *et al*. (2008); Spurrier *et al*. (2008); Tibes *et al*. (2006). Various procedures to improve the analysis of protein lysate arrays have been proposed recently. For example, Brase *et al*. (2010) proposed antibody-mediated signal amplification to increase the sensitivity of this technology. Neeley *et al*. (2009) introduced a variable slope normalization among arrays to help reduce loading bias and recover true correlation structures among proteins.

In this article, we focus on the protein quantification problem of estimating the relative protein concentrations in the arrayed samples. In this area, the commercial analysis package MicroVigene (http://www.vigenetech.com/products.htm) estimates the protein expression level by fitting a four-parameter logistic model to each dilution series. Kreutz *et al*. (2007) used a linear model between the observed protein expressions (or at logarithmic scale) and the underlying concentration levels. Tabus *et al*. (2006) modeled the mean of the observed expression level as a sigmoidal curve and estimated the model parameters via the non-linear least squares. Alternatively, Zhang *et al*. (2009) modeled the serial dilution curve based on the Sips model (similar to the logistic model) to characterize the relationship between signals in successive dilution steps. These methods can be considered as parametric.

On the contrary, Hu *et al*. (2007) proposed a non-parametric approach by assuming that the median of the observed protein expression is equal to a monotonically increasing function without a parametric form. This non-parametric method, contained in a recently developed statistical tool *RPPanalyzer* for analyzing protein lysate array data (Mannsperger *et al*., 2010), is highly data-adaptive without bias due to mis-specification of g. Simulation studies and real experiments have shown the advantage of the non-parametric approach in producing robust results in a variety of scenarios. However, the existing methods of estimation deteriorate in performance as the noise to signal ratio increases in the data. In addition, the non-parametric estimates, if used to estimate protein concentration at the level of individual dilution series, tend to be unstable due to data sparsity.

Our work aims at improving the accuracy of protein level quantification over the non-parametric procedure described in Hu

[*]To whom correspondence should be addressed.

et al. (2007), by incorporating a method of regularization on estimates within each sample. We start with a general description of the problem.

Let y_{ijl} be the observed expression level for the j-th replicate of the i-th sample at the l-th dilution step ($i=1,\dots,m$, $j=1,\cdots,n_i$, and $l=1,\dots,t$). Each replicate is a dilution series of t steps. The relationship between y_{ijl} and the unobserved protein concentration level x_{ij} (at the $\log_2$ scale) can be modeled as

$$y_{ijl}=g(x_{ij}-d_l)+\epsilon_{ijl}, \tag{1}$$

where g is the protein-specific response curve, d_l defines the corresponding dilution index at the l-th step and ϵ_{ijl} denotes random noise. In a typical dilution series, $d_l-d_{l-1}=1$. In this formulation, we allow each replicate to have its own protein concentration level x_{ij}, which differs from the set-up of Hu *et al.* (2007), where $x_{ij}=x_i$ for all the replicates of the i-th sample. If we adopt the method of Hu *et al.* (2007), we can estimate a non-parametric estimate of g, as well as the concentration levels x_{ij}, by minimizing

$$\sum_{i,j,l}|y_{ijl}-g(x_{ij}-d_l)|+\lambda\max_x|g''(x)|, \tag{2}$$

for some smoothing parameter λ, subject to the constraint that g is a non-decreasing function. Note that x_{ij} are identifiable only up to a constant, and the relative differences between x_{ij}'s are of interest in protein concentration.

As pointed out by Yang and He (2011), the complexity of this quantification problem comes from the dimension of x_{ij}, which increases with the sample size. One assumption made in earlier work is that x_{ij} are constant within the i-th sample. This assumption is practically useful because it reduces the dimensionality of the problem and ensures stable estimates. However, it could lead to bias in estimation, and more importantly, could mask or distort variability across replicates.

We propose to allow x_{ij} to vary with j but regularize the estimation problem by using a penalty term on within-sample variabilities. As a result, we shrink some of the replicate-level estimates to common values but allow within-sample variabilities to be retained in other samples as needed. As with other regularization methods in statistics, our proposed method aims to balance the bias-variance trade-off in a data-adaptive way.

The rest of the article is organized as follows. We describe the new method, *Re*gularized *No*nparametric (Reno) analysis of lysate arrays, in Section 2. We demonstrate the performance of the new estimator through simulation studies in Section 3, and show how the proposed method leads to better significance testing in a real data example in Section 4. Some concluding remarks are given in Section 5.

2 METHODS

2.1 New objective function

We formulate the problem of simultaneously estimating g and the vector $\boldsymbol{x}=\{x_{ij}:i=1,\dots,m;\ j=1,\dots,n_i\}$ as follows

$$\min_{g,\boldsymbol{x}}\sum_{i,j,l}L\Big(y_{ijl},g(x_{ij}-d_l)\Big)+\lambda_1V_1(g)+\lambda_2V_2(\boldsymbol{x}) \tag{3}$$

where

$$L(y,g(x))=|y-g(x)|,$$

is the L_1 loss function measuring the discrepancy between the observed expression level y_{ijl} and its fitted value, and V_1 and V_2 are penalty functions on g and $\boldsymbol{x}$, respectively, with λ_1 and λ_2 to be determined. As in Hu *et al.* (2007), we choose the penalty on g

$$V_1(g)=\max_x|g''(x)|, \tag{4}$$

which leads to a quadratic spline solution of g (Koenker *et al.*, 1994). The L_1 loss function is appealing for lysate data quantification, because the unknown function $g(x)$ corresponds to the median protein expression level, and the relationship is equivariant under any monotone transformation of the response.

The new penalty term $V_2(\boldsymbol{x})$ used in (3) aims to regularize the solution of a high-dimensional vector $\boldsymbol{x}$. Empirical results show that the optimization problem without this penalty (that is, $\lambda_2=0$) is often ill-conditioned and leads to unstable results. The special case of $\lambda_2=\infty$ corresponds to the assumption in earlier work that $x_{ij}=x_i$ for all $j=1,\dots,n_i$. The results obtained under this setting mask the variability across replicates.

In protein microarray experiments, it is often the case that the samples can be divided into m subgroups $G_1\cup\cdots\cup G_m$, and the protein concentration levels are similar within each subgroup. In this article, we focus on a common case where each subgroup contains biological or technical replicates of some kind. In our notation, all the replicates within the sample form a subgroup. Naturally, we use the following penalty on $\boldsymbol{x}$,

$$V_2(\boldsymbol{x})=\sum_{i=1}^{m}\sum_{j,j'}|x_{ij}-x_{ij'}|=\sum_{i=1}^{m}|A_i\boldsymbol{x}_i|, \tag{5}$$

where $\boldsymbol{x}_i=(x_{i1},\dots,x_{in_i})^t$, and A_i is the corresponding $n_i\times n_i$ symmetric matrix such that the L_1 norm of the vector $A_i\boldsymbol{x}_i$ equals $\sum_{j,j'}|x_{ij}-x_{ij'}|$.

In summary, we consider a new objective function

$$\sum_{i,j,l}|y_{ijl}-g(x_{ij}-d_l)|+\lambda_1\max_x|g''(x)|+\lambda_2\sum_i|A_i\boldsymbol{x}_i|. \tag{6}$$

Simultaneous optimization over g and $\boldsymbol{x}$ is numerically challenging, and we propose an approximation to (6) in the next subsection.

2.2 Modified objective function

A natural way to solve (6) is a block-wise coordinate descent algorithm that iterates between the following two steps:

- Step 1: minimize (6) over g given $\boldsymbol{x}$;
- Step 2: minimize (6) over $\boldsymbol{x}$ given g.

Iterative algorithms of this type have been widely adopted (Hu *et al.*, 2007 and Tabus *et al.*, 2006). Step 1 reduces to

$$\min_g\left\{\sum_{i,j,l}|y_{ijl}-g(x_{ij}-d_l)|+\lambda_1\max_x|g''(x)|\right\} \tag{7}$$

subject to the constrain that g is a non-decreasing function. This is the same problem considered by Hu *et al.* (2007), and we use the same procedure as before. The solution of g is a quadratic spline, and it can be found via linear programming with the R package *Cobs* developed by He and Ng (1999).

Given g, Step 2 solves

$$\min_{\boldsymbol{x}}\left\{\sum_{i,j,l}|y_{ijl}-g(x_{ij}-d_l)|+\lambda_2\sum_i|A_i\boldsymbol{x}_i|\right\}, \tag{8}$$

which can be optimized for each i separately. Still, the non-linear nature of g makes this problem harder. We propose to simplify the calculation by using an approximate loss function. Note that

$$|y-g(x)|=|g(g^{-1}(y))-g(x)|\approx|g'(x)|\cdot\ |x-g^{-1}(y)|. \tag{9}$$

Then for the i-th subgroup, we solve, in lieu of (8),

$$\min_{\boldsymbol{x}_i}\sum_{j,l}w_{ijl}|x_{ij}-a_{ijl}|+\lambda_2|A_i\boldsymbol{x}_i|, \tag{10}$$

where $w_{ijl}=|g'(x_{ij}-d_l)|$ and $a_{ijl}=d_l+g^{-1}(y_{ijl})$ are calculated based on the current values of g and x_{ij}'s from the previous iteration. What makes this approximation attractive is that the optimization (10) can be solved by weighted median quantile regression using the R package *quantreg* (Koenker, 2005).

In summary, our proposed method uses the following steps.

- Step 0 (Initialization): following Hu *et al.* (2007), we obtain initial values of $\boldsymbol{x}$ as crude estimates of $x_{ij} = x_i$ by assuming g taking a sigmoidal form $g(x) = \alpha + \beta/(1+2^{-\gamma x})$.
- Perform Step 1 to find an estimate of g from (7).
- Given g from Step 1, solve (10), and then center $\boldsymbol{x}$ by making the median of $\boldsymbol{x}$ to be zero.
- Iterate between the previous two steps until a stopping rule is triggered. A useful stopping rule is to check if the objective function in (3) changes by less than a pre-specified value for two consecutive solutions.

After each iteration, we add a step to check whether the new estimates indeed decrease the original objective function. If not, we do a grid search in a neighborhood around the previous estimates to ensure the monotonic decreasing of the objective function. We also note that centering $\boldsymbol{x}$ in each step of the iteration helps identify g. In our empirical study, we found that the algorithm converges quickly, often within four to five iterations.

2.3 Selection of tuning parameters

The tuning parameter λ_1 is associated with the penalty for g in (6). We use the procedure as in Hu *et al.* (2007). In this section, we consider a data-adaptive choice of λ_2.

We follow the basic idea of cross-validation (CV), but the special structure of the lysate data renders the ordinary CV procedure ineffective. If a dilution series $(y_{ij1},\ldots,y_{ijt})$ were taken as a case to be left out, we would have no validation data for that case. To accommodate the special structure here, we propose a t-fold cross-validation, where the t observations from each dilution series are assigned randomly to each of the t-folds. Each fold contains one observation from each dilution series. Let F_k $(k=1,\ldots,t)$ denote the set of indices (i,j,l) in the resulting k-th fold, and

$$\mathrm{CV}(\lambda_2) = \sum_{k=1}^{t} \sum_{i,j,l \in F_k} |y_{ijl} - \hat{g}^{[-k]}(\hat{x}_{ij}^{[-k]} - d_l)|,$$

where $\hat{g}^{[-k]}$ and $\hat{x}_{ij}^{[-k]}$ are the estimates of g and x_{ij} based on data not in F_k. The parameter λ_2 is chosen to minimize $\mathrm{CV}(\lambda_2)$.

3 SIMULATION STUDIES

To evaluate the performance of the proposed method, we conduct simulation studies to compare our procedure (*Reno*) with the following alternatives:

- *Indep*: the procedure with $\lambda_2 = 0$, which is equivalent to estimating a different concentration level for each dilution series.
- *Same*: the procedure with $\lambda_2 = \infty$, which is equivalent to assuming that the replicates within each sample have exactly the same protein concentration levels.
- *Nonpa*: the procedure with $\lambda_2 = \infty$, but (8) is solved using the procedure of Hu *et al.* (2007), instead of the approximation (10). We include this alternative procedure to evaluate the accuracy of our approximation. It turns out that *Same* is much faster than *Nonpa* without much sacrifice in accuracy.
- *Tabus*: the parametric procedure based on sigmoid curve of Tabus *et al.* (2006).
- *Oracle*: the procedure with $\lambda_2 = \infty$ only for the samples with $x_{ij} = x_{i1}$ $(j=1,\ldots,n_i)$, and $\lambda_2 = 0$ otherwise. This procedure requires the knowledge on the unknown quantities so it is used only as an Oracle in the simulation for comparison purposes.

In the experiments, we simulate data of 300 dilution series, each of length $t = 6$, from $m = 100$ samples with $n_i = 3$ replicates for each sample. The three replicates for each sample form a subgroup. The concentration levels of all the replicates for 60% of the samples are generated to be identical, whereas the remaining 40% of the samples have varying concentration levels from replicate to replicate.

Throughout this section, we evaluate the estimation error by

$$\mathrm{Err}(\hat{\boldsymbol{x}},\boldsymbol{x}) = \sum_{i,j} |g(x_{ij}) - g(\hat{x}_{ij})|, \tag{11}$$

where g is the true response function. Note that $|g(x)-g(z)| \approx |g'(x)| \cdot |x-z|$, so this criterion downweights the estimation error of x where $|g'(x)|$ is small, i.e. in the nearly flat regions of g. Obviously, accuracies in the nearly flat regions of g cannot be expected from any method. We choose *Oracle* as the benchmark method and report the relative error, the ratio of the error of each method to that of *Oracle*.

3.1 Experiment 1

In our first experiment, the response curve is taken from two sigmoid curves in the form of $g(x) = \alpha + \beta/(1+2^{-\gamma x})$ with $\alpha = 3000$ and $\beta = 10000$, but $\gamma = 0.7$ for positive x and $\gamma = 2.1$ for negative x. For positive x, the noise is generated from a scaled t distribution with three degrees of freedom with the scale $\sigma = 6000 \times (x+5)^{-1}$; for negative x, the noise follows the normal distribution $\mathsf{N}(0, 600^2)$.

We conducted 100 simulation trials, and the results are summarized in Figure 1. For the first 20 datasets, the tuning parameter λ_2 was chosen adaptively by CV. The upper left panel of Figure 1 is the boxplot of $\mathrm{CV}(\lambda_2)$ based on these 20 datasets. The value $\lambda_2^* = 1500$ is a good choice for most datasets from this model. To save time in the simulation study, we used this fixed value for all other datasets. But we expect the results for *Reno* to be similar if the CV is done for each trial. The upper right panel of Figure 1 is the boxplot of the relative errors of each competing method over 100 trials. It is clear that the proposed method *Reno* outperforms its competitors.

We repeated the data generating process with two smaller error variances, where the SD is reduced by a factor of 2/3 and 1/2, respectively. The results are summarized in the lower half of Figure 1. In all these cases, *Tabus* performed the worst due to model misspecification. The performances of *Nonpa* and *Same* are similar. When the SNR (signal-to-noise ratio) is low, even if many of the three replicates have different x_{ij} values, the variance deduction from pooling information from the replicates to estimate a single protein concentration level more than offsets the bias. Thus, *Nonpa* and *Same* have smaller errors than *Indep* in the upper right panel of Figure 1. When the SNR is higher, bias becomes dominating, so *Indep* has a smaller error than *Nonpa* and *Same* in the lower two panels of Figure 1. It is worth noting that *Reno* consistently outperforms, and it even beats *Oracle* when the SNR is low. If the calibration curve g were known, *Oracle* would have the best performance. However, when g is unknown and the SNR is low, adaptively borrowing information across replicates can improve estimation accuracy of g and hence the quantification accuracy of $\boldsymbol{x}$.

3.2 Experiment 2

In the second experiment, we generated the response curve from a single sigmoidal curve with $\alpha = 1000$, $\beta = 4000$ and $\gamma = 0.8$. The errors were generated independently from normal distribution

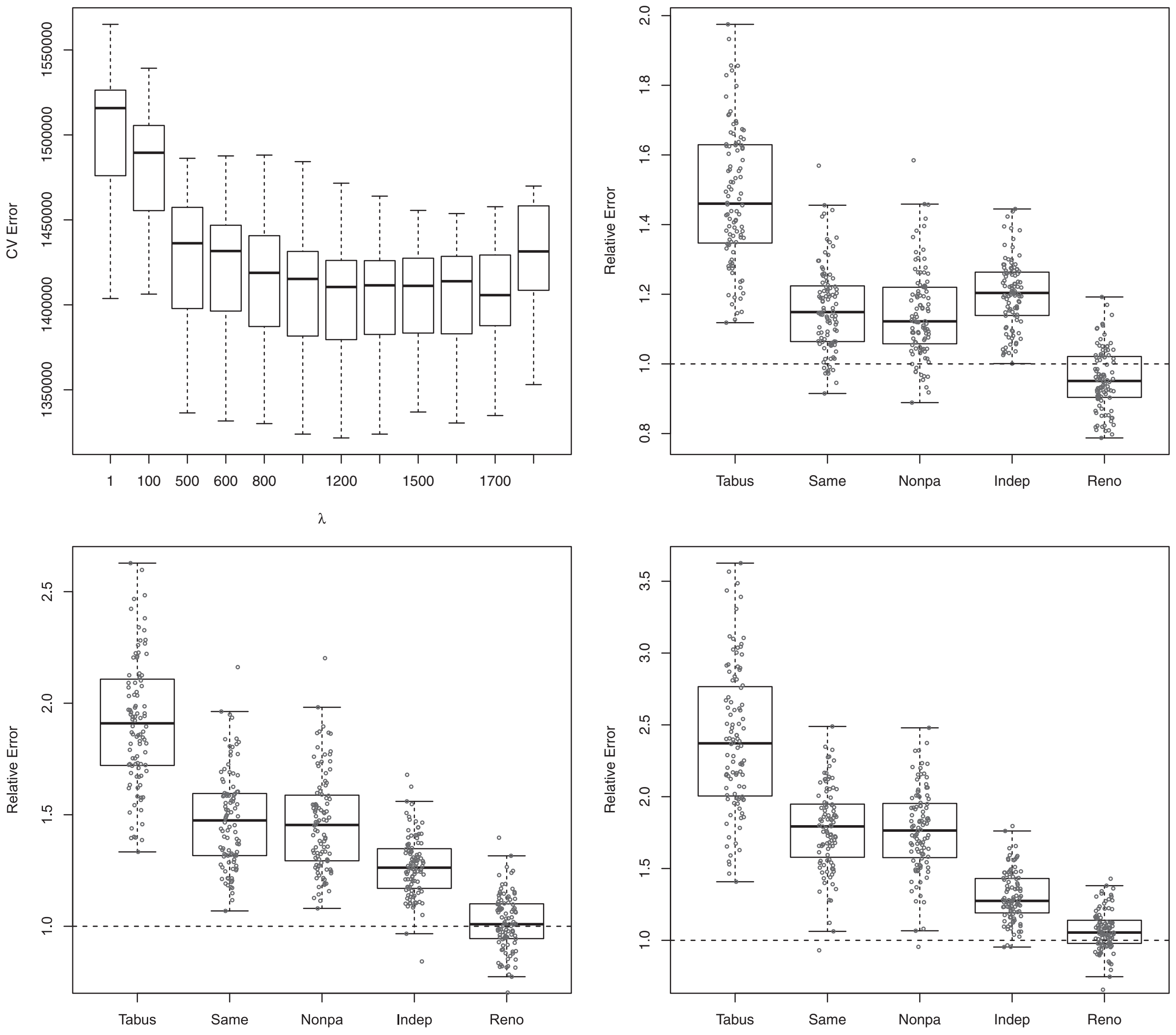

Fig. 1. Results from Experiment 1. The upper left panel displays the boxplot of the CV measures of the first 20 datasets at several values of λ_2. Boxplots for the estimation errors (relative to *Oracle*) on 100 datasets are displayed in the upper right panel (original data generating process, $\lambda_2=1500$), the lower left panel (with the error SD reduced by a factor of $2/3$, $\lambda_2=1000$), and the lower right panel (with the error SD reduced by $1/2$, $\lambda_2=750$).

$\mathrm{N}(0,500^2)$. The true $\boldsymbol{x}$ values and the rest of the set-up are the same as in Experiment 1. The result are now presented in Figure 2.

In this experiment, *Indep* is always inferior to others, confirming that estimating individual x_{ij} for all replicates is usually not recommended. It is interesting to note that *Same* and *Nonpa* outperform *Oracle* when the SNR is small (the upper right panel of Fig. 2), but not in the other cases. Since the data are generated from a sigmoidal curve, the results of *Tabus* are quite decent, but *Reno* remains a top performer.

4 ANALYSIS WITH REAL DATA

We analyzed the protein lysate data from Mendes *et al*. (2007) with the proposed *Reno*, as well as the non-parametric procedure *Nonpa* from Hu *et al.* (2007) and the parametric procedure *Tabus* from Tabus *et al.* (2006). The data consist of the intensity measurements for 90 samples from 52 protein arrays. Each sample has three replicates that are diluted 2-fold six times. The fitted curves from *Reno* and *Nonpa* are very similar for all arrays, but the parametric curves look different on some datasets. This is because the curve estimated by *Tabus* is constrained to take the logistic shape, which may not fit the data well. Between *Reno* and *Nonpa*, we noticed that the estimated protein concentration levels x_{ij} differ on a number of arrays. Those differences have an impact on subsequent statistical analysis, as shown in the example below.

The samples in this study are cell lines from 12 different cancer groups. We use the data to find out whether protein concentration levels are significantly different among cancer groups. We conducted

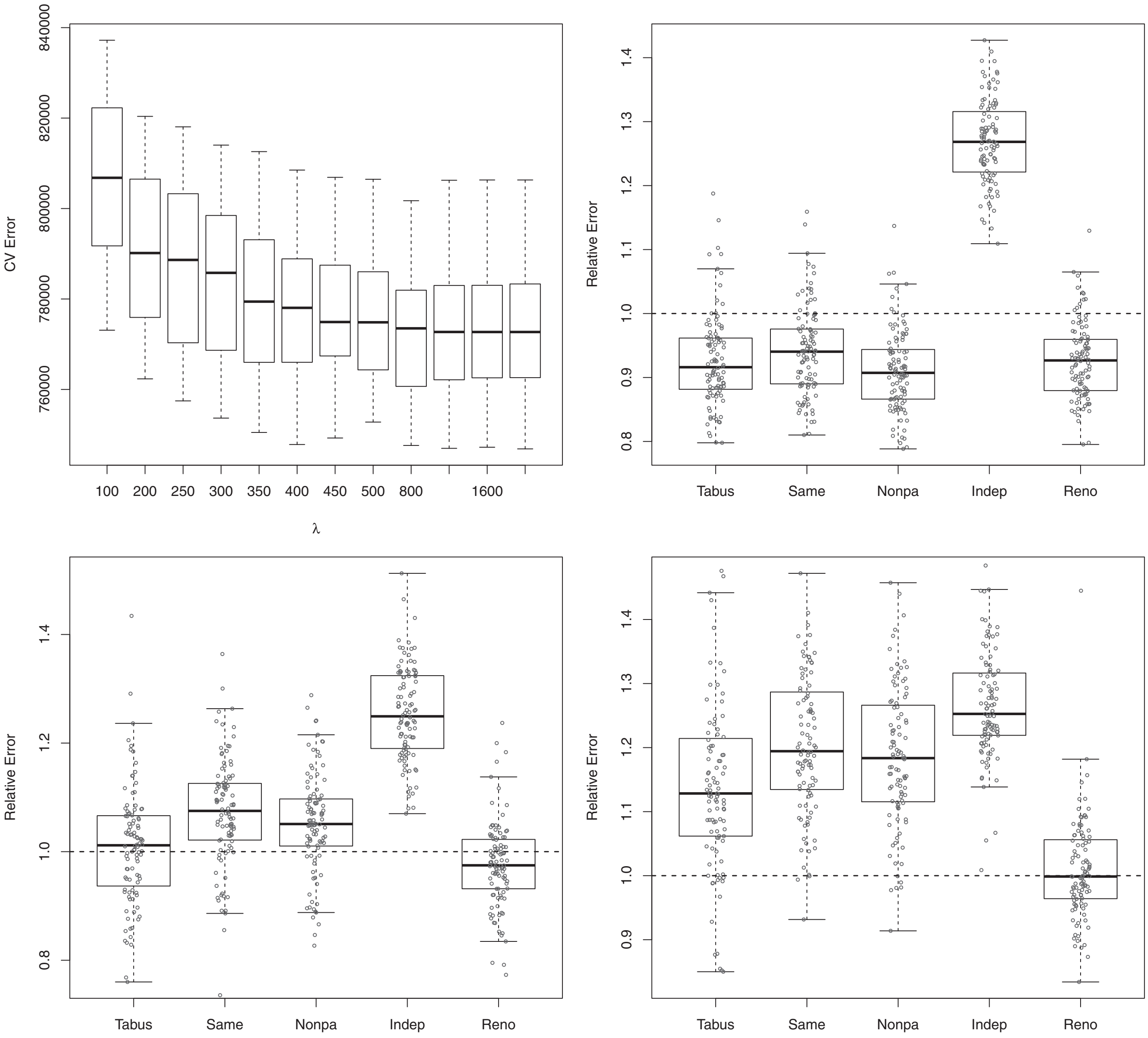

Fig. 2. Results from Experiment 2. The upper left panel displays the boxplot of the CV measures of the first 20 datasets at several values of λ_2. Boxplots for the estimation errors (relative to *Oracle*) on 100 datasets are displayed in the upper right panel (original data generating process, $\lambda_2 = 800$), the lower left panel (with the error SD reduced by a factor of 2/3, $\lambda_2 = 500$), and the lower right panel (with the error SD reduced by 1/2, $\lambda_2 = 400$).

a two-sample t-test based on the average concentration levels $z_i = (x_{i1} + x_{i2} + x_{i3})/3$ by assuming that z_i are normally distributed with mean μ and variance $\sigma^2 + \tau_i^2$, where τ_i^2 is estimated by the sample variance of the replicates x_{i1}, x_{i2} and x_{i3}, and σ^2 denotes the variance component that is homogenous for all samples from the same cancer group.

For *Nonpa*, $\tau_i^2 = 0$, but for *Reno*, τ_i^2 can be non-zero, and the MLE $\hat{\mu}$ is a weighted average of z_i's. Samples with higher within-sample variation are down-weighted, because their estimated x_{ij} values are less reliable. As will be shown in the examples below, the within-sample variation, which is available only with the *Reno* procedure, indeed provides valuable information in the statistical analysis.

We focus on protein *cdk7*, for which the P-values for contrasting the lung cancer group with two other cancer groups (Colon or Sarcoma) are listed in Table 1. The estimates from *Nonpa* have a hard time detecting the difference between the lung cancer group and the other two groups. To see what led to the difference, we display

Table 1. P-values for some selected two-sample t-tests

Protein		*Tabus*	*Nonpa*	*Reno*
cdk7	Lung (13) *versus* Colon (8)	0.060	0.101	0.030
	Lung (13) *versus* Sarcoma (14)	0.018	0.042	0.006
p19	Pancreatic (8) *versus* Colon (8)	0.048	0.041	0.104
Zap70	Sarcoma (14) *versus* Ovarian (5)	0.020	0.030	0.188

The lung cancer group has 13 cell lines/samples, Colon has 8, Sarcoma has 14, Pancreatic has 8 and Ovarian has 5.

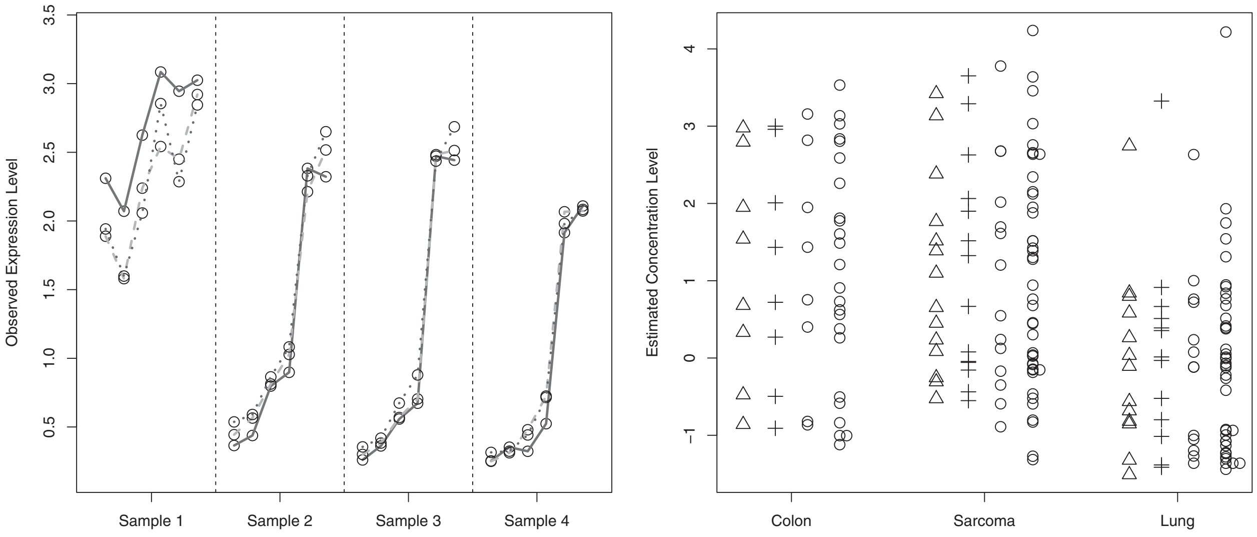

Fig. 3. Plots for protein *cdk7*. Left: plot of the observed dilution series for a subset of samples from the lung cancer group, where Sample 1 is the influential sample mentioned in Section 4. Right: plot of the estimated protein concentration levels x_{ij} for Colon, Sarcoma and Lung cancer groups from *Tabus* (△, 1st column), *Nonpa* (+, 2nd column) and *Reno* (∘, last two columns). The estimated concentration levels from *Reno* are displayed twice, one is the sample average (one point for each sample, the third column) and the other is the raw data (three points for each sample, the last column).

the data of four samples from the lung cancer group in Figure 3 (left panel). Sample 1 has higher expressions than the other samples, and more importantly, there is a visible within-sample variation for Sample 1, but not for the other samples. This information is reflected in the estimates of protein concentrations from *Reno*: 4.2, 1.9 and 1.7 for the three replicates. As a consequence, Sample 1 gets down-weighted in the statistical analysis with the following effects: (i) it decreases the average of the lung cancer group, and therefore enlarges the mean difference between groups; and (ii) it reduces the group variance, making the *t*-test more powerful. This explains why the *P*-values from *Reno* are less than a third of the *P*-values from *Nonpa*.

Note that *Reno* provides more accurate and robust uncertainty analysis than its competitors, but does not always provide smaller *P*-values. In some cases, *Reno* points to the opposite direction when *Tabus* and *Nonpa* report smaller *P*-values by ignoring within-sample variations. Results from two other example slides (proteins *p19* and *Zap70*) listed in Table 1 demonstrate this point.

5 DISCUSSION

In this article, we propose a regularized estimation procedure in non-parametric analysis of the protein lysate array quantification, which enables us to consider replicate-specific quantities when the within-sample variability is significant. We use a simple approximation to the loss function so that the optimization can be carried out by linear programs. We propose a specialized CV method to select the tuning parameter that regulates the within-sample variability. Earlier methods of lysate array quantification have to aggregate the replicates in each sample to avoid unstable estimates, and in doing so, important information may get lost in the aggregation. We demonstrate through simulated and real data that the proposed method is helpful in providing additional information about within-sample variabilities, which has important implications in the subsequent statistical analysis of the lysate array data. Note that the estimated calibration curves from different methods are essentially the same, and the differences are in the concentration estimates of individual samples.

Funding: National Institute of Health Grants R21CA129671, R01GM080503, R01CA158113 and NCI CA97007, and the National Science Foundation Awards DMS-0706818, DMS-0800631 and DMS-1007396.

Conflict of Interest: none declared.

REFERENCES

Borrebaeck,C.A. and Wingren,C. (2007) High-throughput proteomics using antibody microarrays: an update. *Expert. Rev. Mol. Diagn.*, **7**, 673–686.

Brase,J.C. *et al.* (2010) Increasing the sensitivity of reverse phase protein arrays by antibody-mediated signal amplification. *Proteome Sci.*, **8**, 36.

Cahill,D.J. and Nordhoff,E. (2003) Protein arrays and their role in proteomics. *Adv. Biochem. Eng. Biotechnol.*, **83**, 177–187.

Cai,D. *et al.* (2010) Steroid receptor coactivator-3 expression in lung cancer and its role in the regulation of cancer cell survival and proliferation. *Cancer Res.*, **70**, 6477–6485.

Carey,M.S. *et al.* (2010) Functional proteomic analysis of advanced serous ovarian cancer using reverse phase protein array: TGF-β pathway signaling indicates response to primary chemotherapy. *Clin. Cancer Res.*, **16**, 2852–2860.

Cheng,K.W. *et al.* (2005) Assay of Rab25 function in ovarian and breast cancers. *Meth. Enzymol.*, **403**, 202–215.

Grote,T. *et al.* (2008) Validation of reverse phase protein array for practical screening of potential biomarkers in serum and plasma: accurate detection of CA19-9 levels in pancreatic cancer. *Proteomics*, **8**, 3051–3060.

He,X. and Ng,P. (1999) COBS: qualitatively constrained smoothing via linear program. *Comput. Stat.*, **14**, 315–337.

Hu,J. *et al.* (2007) Nonparametric quantification of protein lysate arrays. *Bioinformatics*, **23**, 1986–1994.

Ivanov,S.S. *et al.* (2004) Antibodies immobilized as arrays to profile protein post-translational modifications in mammalian cells. *Mol. Cell Proteomics*, **3**, 788–795.

Kim,W.Y. *et al.* (2008) A novel derivative of the natural agent deguelin for cancer chemoprevention and therapy. *Cancer Prev. Res.*, **1**, 577–587.

Koenker,R. (2005) *Quantile Regression*. Cambridge University Press, New York, USA.

Koenker,R. *et al.* (1994) Quantile smoothing splines. *Biometrika*, **81**, 673–680.

Kreutz,C. *et al.* (2007) An error model for protein quantification. *Bioinformatics*, **23**, 2747–2753.

MacBeath,G. and Schreiber,S.L. (2000) Printing proteins as microarrays for high-throughput function determination. *Science*, **289**, 1760–1763.

Mannsperger,H.A. *et al.* (2010) RPPanalyzer: analysis of reverse-phase protein array data. *Bioinformatics*, **26**, 2202–2203.

Mendes,K.N. *et al.* (2007) Analysis of signaling pathways in 90 cancer cell lines by protein lysate array. *J. Proteome Res.*, **6**, 2753–2767.

Neeley,S. *et al.* (2009) Variable slope normalization of reverse phase protein arrays. *Bioinformatics*, **25**, 1384–1389.

Pluder,F. *et al.* (2006) Proteome analysis to study signal transduction of G proteincoupled receptors. *Pharmacol. Ther.*, **112**, 1–ï¿½11.

Poetz,O. *et al.* (2005) Protein microarrays: catching the proteome. *Mech. Ageing Dev.*, **126**, 161–170.

Sahin,O. *et al.* (2007) Combinatorial RNAi for quantitative protein network analysis. *Proc. Natl Acad. Sci. USA*, **104**, 6579–ï¿½6584.

Spurrier,B. *et al.* (2008) Reverse-phase protein lysate microarrays for cell signaling analysis. *Nat. Protoc.*, **3**, 1796–1808.

Tabus,I. *et al.* (2006) Nonlinear modeling of protein expressions in protein arrays. *IEEE Trans. Signal Process.*, **54**, 2394–2407.

Tibes,R. *et al.* (2006) Reverse phase protein array (RPPA): validation of a novel proteomic technology and utility for analysis of primary leukemia specimens and hematopoetic stem cells (HSC). *Mol. Cancer Ther.*, **5**, 2512–2521.

Yang,J. and He,X. (2011) A multi-step protein lysate array quantification method and its statistical properties. *Biometrics*, **67**, 1197-1205.

Zhang,L. *et al.* (2009) Serial dilution curve: a new method for analysis of reverse phase protein array data. *Bioinformatics*, **25**, 650–654.