# RPPA Quantification with SuperCurve

E. Shannon Neeley

## 1 Introduction

Once the raw intensities have been adjusted for background and other spatial trends, the next step is to map these intensities to estimates of the concentration of protein in the sample with the underlying assumption that the brightness of a spot is proportional to the amount of analyte in that spot.

Each sample on an RPPA slide has been printed in a dilution series. The purpose of the dilution series is to ensure at least one spot from the series will be in the "linear range of expression." As with many dilution assays, expression on RPPAs will typically follow a sigmoidal curve. The spots that are highly diluted will often have so little protein that they will not show expression beyond the level of background. Conversely, the undiluted spots will often have so much protein that the reach they level of saturation. The challenge of protein quantification is to use the information provided by the whole dilution series to estimate the relative protein concentration in a given sample.

There have been many different methods used to estimate protein concentration. These methods fall into two general categories: single sample estimation and joint sample estimation. Single sample methods estimate the protein concentration for a sample using only information from that sample. Single sample methods have been proposed by Herrman et al. [2], Nishizuka et al. [6], and Mircean et al. [4]. Joint sample methods use information from all the samples on an array to compute estimates for each sample on the array, as well as global slide parameters. Joint estimation potentially improves estimates by "borrowing strength" across all samples. In this way, all the samples on an array contribute information about the

overall dose-response curve for a slide. Joint sample models have been developed by Tabus et al. [8], Hu et al. [3], and by our group at M.D.Anderson Cancer Center [1]. We call our model "SuperCurve."

# 2 SuperCurve

## 2.1 Joint Estimation and Notation

Joint estimation models use all the information on an array to estimate global (antibody or slide) parameters as well as individual protein concentrations for each sample. This type of model makes sense for several reasons. First, since each microarray slide is probed with a single antibody, protein expression of the samples should have similar chemistry and hybridization behavior. Second, all samples can provide some information about the baseline level, the saturation level, and the rate of signal increase at each dilution point. Third, estimating parameters with more data can yield tighter estimates and smaller variances. Finally, joint estimation can yield estimates with more dynamic range.

In terms of notation, let $Y_{ij}$ be the observed (adjusted for background) intensity for dilution $i$ of sample $j$ with $i = 1, \ldots, I$ and $j = 1, \ldots, n$. Let $\mu_j$ be the true log protein concentration for sample $j$ at some prespecified step within the dilution series (such as the first step or the median step). This is the parameter of interest. Finally, let $x_{ij}$ be the known dilution offset step. For example, Figure 1 shows an RPPA with 1334 separate dilution series, each in 5 spot 1/2 dilutions. In this case $i = 1, \ldots, 5$, where $i = 1$ represents the undiluted spot and $i = 5$ is the spot at 1/16 dilution; $n = 1, \ldots, 1334$; $x_{i=(1,2,3,4,5),j} = -2, -1, 0, 1, 2$ if the dilution series are arbitrarly centered at the median.

## 2.2 Model

SuperCurve models the relationship between intensity and protein concentration with a 3 parameter logistic:

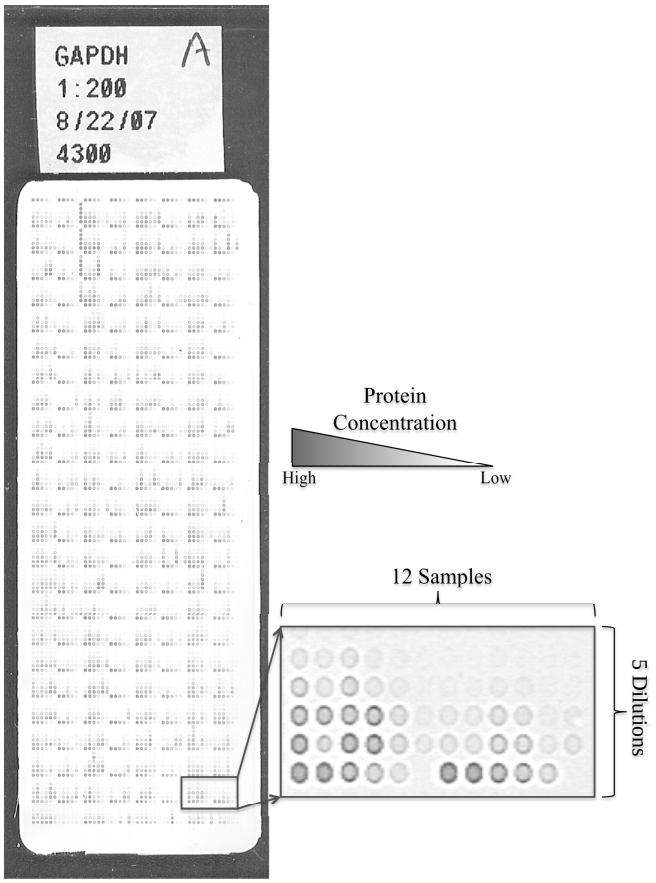


Figure 1: Image of an example RPPA with 1152 separate dilution series. Each dilution series is printed in 5-spot 1/2 dilutions. The zoom-in box shows 12 dilution series on the array. The darker the spot, the higher the concentration of protein. Some of the spot do not appear, even in the zoom-in box because there was no label (i.e. protein) at the spot.

$$Y_{ij} = \alpha + \beta \frac{\exp[\gamma(x_{ij} - \mu_j)]}{1 + \exp[\gamma(x_{ij} - \mu_j)]} + \epsilon_{ij}. \quad (1)$$

Here, $\alpha$, $\beta$, and $\gamma$ are global array parameters that define one logistic curve (a "Super Curve") for all the samples on the array and $\epsilon_{ij} \sim \mathcal{N}(0, \sigma^2)$.

The SuperCurve parameters are estimated by first specifying initial estimates for the protein concentrations and then fitting the logistic model with a nonlinear least squares regression to estimate $\alpha$, $\beta$, and $\gamma$. Next, the estimates for the protein concentrations are updated with the estimated logistic curve. This process is repeated iteratively until convergence is reached.

This model has been implemented in a package for use within the software program R [7]. The package, called "SuperCurve," can be downloaded from `http://bioinformatics.mdanderson.org/software` [1]. The SuperCurve package also has the capability to implement a nonparametric model as describe by Hu et al. [3], and briefly introduced in the following section.

#### 2.2.1 Nonparametric Model

Hu, et al. [3] proposed a more flexible nonparametric joint sample model for the quantification of RPPA data that can improve estimates when the data does not follow a sigmoidal curve. This approach uses a nonparametric model of the form:

$$Y_{ij} = g(x_i - \mu_j) + \epsilon_{ij} \quad (2)$$

where the median of $\epsilon_{ij}$ is assumed to be 0 and $g$ is an unknown but monotonically increasing function. This function is estimated with constrained b-spline approximations as implemented in the R package `cobs` [5].

The approach to parameter estimation is similar to both the Tabus et al. and `SuperCurve` algorithms. First initial estimates for the $\mu_j$s are specified and used to estimate the $g$ function by regressing $Y_{ij}$ on $x_{ij} - \mu_j$. Estimates for the $\mu_j$s are updated with the estimated curve $g$. Finally, the steps are repeated

iteratively until convergence is reached.

### 2.2.2 Parameter Estimation

Model parameters for both the logistic and nonparametric models are estimated with the following algorithm:

1. Form initial estimates of the protein concentration $\hat{c}_{ij}$ for each sample $j$ and dilution $i$. How these estimates are chosen is discussed in more detail below. Note that only one $\hat{c}_{ij}$ is needed for each $j$. For example, let $\hat{c}_{1j}$ be the initial estimate of protein concentration for the undiluted spot for sample $j$. The concentration at the other dilution levels can be expressed on a $log_k$ scale as $\hat{c}_{ij} = \hat{c}_{1j} - i$ (assuming $k$-fold dilutions).

2. Based on the initial estimates, fit the model $\hat{g}$ (which will follow the form of equation (1) for the logistic model or equation (2) for the nonparametric model).

3. Re-estimate sample concentrations from the response curve, $\hat{g}$ by robustly minimizing a goodness-of-fit metric, namely the $L_2$ criterion:

$$\hat{c}_{1j} = \min_{c_{1j}} \left[ \sum_i \left( Y_{ij} - \hat{g}(c_{1j} - i) \right)^2 \right]$$

4. Repeat steps 2 and 3 until convergence is reached.

### 2.2.3 Initial Estimates

It is important to choose sensible initial estimates for the protein concentrations to avoid failed convergence. The `SuperCurve` package uses a logistic relationship to compute the initial estimates for both the logistic and nonparametric models. First rough estimates of the logistic parameters are computed as:

$$\hat{\alpha} = \min(Y_{ij})$$

$$\hat{\beta} = \max(Y_{ij}) - \hat{\alpha}$$

$$\hat{\gamma} = median_j \left( \frac{\max_i(z_{ij}) - \min_i(z_{ij})}{I} \right)$$

where $I$ is the number of dilution steps,

$$z_{ij} = lp\left( \frac{Y_{ij} - \hat{\alpha}}{\hat{\beta}} \right)$$

and the function $lp$ is defined as

$$lp(z) = \begin{array}{ll} log_2\left(\frac{z}{1-z}\right) & \text{if } \epsilon < z < 1-\epsilon \\ log_2\left(\frac{\epsilon}{1-\epsilon}\right) & otherwise \end{array}$$

The value of $\epsilon$ can be changed, but the default in `SuperCurve` is to use $\epsilon = 0.001$ for estimates that are robust to spot intensities that are extremely low or negative. The initial concentrations are defined as

$$\hat{c}_{1j} = median_i \left( \frac{z_{ij}}{\hat{\gamma}} - i \right).$$

# References


[1] Coombes, K., and et al. Object-oriented microarray and proteomics analysis library (oompa). `http://bioinformatics.mdanderson.org/software.html`, 3 Feb 2007.

[2] Herrmann, P. C., Gillespie, J. W., Charboneau, L., Bichsel, V. E., Paweletz, C. P., Calvert, V. S., Kohn, E. C., Emmert-Buck, M. R., Liotta, L. A., and Petricoin, E. F. Mitochondrial proteome: Altered cytochrome c oxidase subunit levels in prostate cancer. *Proteomics 3* (2003), 1801–1810.

[3] Hu, J., He, X., Baggerly, K. A., Coombes, K. R., Hennessy, B. T., and Mills, G. B. Non-parametric quantification of protein lysate arrays. *Bioinformatics 23*, 15 (2007), 1986–1994.

[4] Mircean, C., Shmulevich, I., Cogdell, D., Choi, W., Jia, Y., Tabus, I., Hamilton, S. R., and Zhang, W. Robust estimation of protein expression ratios with lysate microarray technology. *Bioinformatics 21*, 9 (2005), 1935–1942.

[5] Ng, P. T., and Maechler, M. *cobs: COBS – Constrained B-splines (Sparse matrix based)*, 2006. R package version 1.1-3.5.

[6] Nishizuka, S., Charboneau, L., Young, L., Major, S., Reinhold, W. C., Waltham, M., Kouros-Mehr, H., Bussey, K. J., Lee, J. K., Espina, V., Munson, P. J., Petricoin 3rd, E., Liotta, L. A., and Weinstein, J. N. Proteomic profiling of the NCI-60 cancer cell lines using new high-density reverse-phase lysate microarrays. *Proceedings of the National Academy of the Sciences of the United States of America 100*, 24 (2003), 14229–34.

[7] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2008. ISBN 3-900051-07-0.

[8] Tabus, I., Hategan, A., Mircean, C., Rissanen, J., Shmulevich, I., Zhang, W., and Astola, J. Nonlinear modeling of protein expressions in protein arrays. *IEEE Transaction on Signal Processing 54*, 6 (2006), 2394–2407.