

Statistical Methods for Spot Detection with Macroarray Data

Yi Xie

*Department of Nutrition and Food Sciences and Center for Microbe Detection and
Physiology, Utah State University, Logan UT 84322-8700*

Adele Cutler

Department of Mathematics and Statistics, Utah State University, Logan UT 84322-3900

Bart Weimer

*Department of Nutrition and Food Sciences and Center for Microbe Detection and
Physiology, Utah State University, Logan UT 84322-8700*

Andrejus Parfionovas

Department of Mathematics and Statistics, Utah State University, Logan UT 84322-3900

Abstract

We describe a statistical method for detecting spots and extracting intensities for an oligonucleotide macroarray. The method is freely available, implemented as an ImageJ plugin. Special capabilities include a fast automatic grid finder, detection of very weak spots, and handling saturation of strong spots due to limitations in the dynamic range of the detection device. The method is validated on dilution arrays and illustrated with experimental data.

1. Introduction

A membrane-based oligonucleotide DNA macroarray protocol was recently developed as a low-cost alternative to glass-slide or silicon-based DNA microarrays. (Y. Xie, et al., 2003). The probes were hand-spotted onto a nylon membrane. Hybridization was detected using chemilluminescence, captured on photographic film, and converted into 16 bit grayscale tif images. A more detailed description of the data can be found in Section 2. More details on the protocol are given in Xie et al. (2002a and 2002b).

Statistical analysis of the data was hampered by the lack of freely available software for processing the images and extracting accurate intensity information. While there are several freeware spot-detection packages (see, for example, Yang et al. 2002), we were unable to find one that met all our needs. In particular, we found that the packages had trouble with one or more of the following:

- *Manual placement of grids.* This is tedious and adds a source of variation that is difficult to quantify.

- *Saturation of strong spots.* Although we used photographic film, this phenomenon occurs whenever the abundance levels of the spots have a higher dynamic range than the detection device.
- *Non-detection of weak spots.* The standard approach of performing an edge-detection technique in each grid-region generally results in non-detection of weak spots that are visible to the eye.

Our proposed solutions to the above difficulties are described in Section 3 of this paper and implemented in a “plugin” for the java image processing software ImageJ. The code for the plugin is available upon request from adele@stat.usu.edu. The methods are evaluated on a series of dilution arrays (Section 4) and illustrated using experimental data (Section 5).

2. Description of the Data

Figure 1 gives a typical image of the data of interest. It also shows the GUI for the software. The grid was placed automatically using the procedure described in Section 3. Each bin in the grid contains roughly 10,000 pixels with 16-bit resolution. The spots are subject to diffusion of the probes when spotted on the nylon membrane and diffusion during chemilluminescence, and therefore the diameter of the spots increases with strength. In this particular experiment, 16x24=384 probes were spotted in an array with 16 rows and 24 columns, then the same 384 probes were spotted again, in the same configuration but horizontally offset by half the original horizontal inter-spot distance. In this way, a particular probe was spotted in two adjacent columns (Figure 2a). It is common for the second printing to be imperfectly aligned, resulting in a pattern such as that in Figure 2b. The arrays are also subject to moderate shearing due to the flexibility of the membrane

Cross-sections of individual moderate-intensity spots (Figure 3a) suggest that the intensities follow a bivariate normal pattern:

$$z = \alpha + \beta \exp\{-0.5*[(x-\mu_x)^2 + (y-\mu_y)^2]/\sigma^2\} \quad (1)$$

where x is the horizontal coordinate (in pixels), y is the vertical coordinate (in pixels), and z is the intensity at pixel (x,y) . We use the term “pattern” deliberately because this is an intensity surface, not a density. The normal, pattern might be justifiable based on the theory of diffusion processes, but we view it simply as an empirical model of the intensity.

Cross-sections of strong spots (Figure 3b) reveal truncation of the spots due to the intensity hitting the detection limit of the photographic film. We refer to these spots as “saturated”. They have a pattern similar to (1) on the edges, but show a plateau in the center. Saturation is inevitable because photographic film has a dynamic range of about two orders of magnitude, while these mRNA abundance levels have a dynamic range of at least four orders of magnitude. Failure to deal with this problem during spot detection leads to a nonlinear relationship between true concentration and measured intensity, and makes subsequent statistical analysis more difficult. We choose to deal with the problem at the spot detection stage.

Due to diffusion, many of the strong spots show clear overlap with neighboring regions. In general, it is not possible to find a grid so that only one spot affects the

intensity in each rectangular sub-region of the grid. It is not practical to use larger membranes, and spotting fewer spots on each membrane introduces additional sources of variation, so we choose to handle the overlap in the spot detection process.

Figure 4a shows an image with strong background effects. The red outlines show “spots” located using edge detection. However, numerous spots are visible in the contaminated regions and should be detectable. Of course, it is preferable to improve the experimental protocol to eliminate such contamination, so spot detection in these regions is not a driving force in our research, but our procedure does moderately well at detecting such spots. Our results for the same image are shown in Figure 4b.

In summary, we have spots that diffuse, overlap, and saturate. These problems are largely due to our experimental protocol, and it is reasonable to ask whether the protocol should be changed. However, we argue that all existing protocols have some challenging aspects. For example, diffusion is more challenging to deal with than spots that have a similar radius. However, diffusion can also be useful. For example, spots on glass slides do not diffuse appreciably, but if such a spot hits the limits of the detection device, there is no information about its true intensity.

In the next section, we describe methods to take advantage of the spot structure to automatically place the grid, detect spots, and measure intensities.

3. Grid Placement and Spot Detection

Grid Placement

When using a robotic spotter, grid placement can be automated due to the hardware. Spot detectors that are not tied to particular hardware usually use a manually-placed grid (click and drag to stretch a regular grid with a given number of rows and columns, release to place the grid). Manual placement is tedious and can lead to additional sources of variation due to human judgment. Our automatic grid placement algorithm uses the marginal mean intensities and finds the vertical and horizontal components separately.

Consider determining the horizontal spacing and the location of the left-hand side of the grid. Let $M(c)$ be the mean intensity for all pixels in the c^{th} column. We choose η and δ to maximize the mean, over j , of the following quantity:

$$M(\eta + j\delta) - M(\eta + [\delta/2] + j\delta) \quad (2)$$

where $[\]$ denotes the integer part. Our implementation is a simple grid search over the possible integers η and δ . The procedure for finding the bottom edge and vertical spacing is analogous.

If the image is rotated or shearing is extreme, the marginal intensities may not be useful in determining the grid. Rotation problems can be fixed manually inside ImageJ. It would also be possible to generalize our grid placement strategy to optimize over rotations. Extreme shearing is more difficult, but one possibility would be to divide the image into several sub-regions and place a grid in each region separately.

Spot Detection

In this section, we start with the ideas of our methods and then give an outline of the algorithm. The java code is also available from adele@stat.usu.edu.

For non-saturated spots, a natural approach is to estimate the parameters in model (1) using a nonlinear least squares regression program. Minimizing the residual sum of squares with respect to all 5 parameters (α , β , μ_x , μ_y , σ) is time-consuming. However, note that if μ_x , μ_y , and σ are known, α and β can be estimated by simple linear regression. Now, if α and β are fixed, the residual sum of squares can be minimized with respect to μ_x , μ_y , and σ , an easier problem than the original 5-parameter minimization. This alternating minimization process is iterated as described in the algorithm outline.

For saturated spots we estimate the saturation level of the image and minimize the residual sum of squares only over pixels below the estimated saturation level. This approach fits model (1) to the pixels on the shoulder of the spot, and estimates the intensity that would have been observed in the saturated regions, had the dynamic range of the detection procedure been unlimited. Of course, this assumes that model (1) is a good approximation to the true intensity, which must be verified empirically. The dilution experiments (Section 4) show that model (1) is suitable for our data.

To estimate the saturation level, we find the maximum intensity in each bin, then use the p^{th} percentile of these, where p is a rough estimate of the percentage of saturated spots. We choose $p=40$ for the dilution data described in Section 4 and $p=30$ for the experimental data described in Section 5. The saturation level remains fixed throughout the procedure.

As with most nonlinear regression problems, we must find suitable initial values for the parameters. First, consider estimating μ_x (the procedure for μ_y is identical). For each bin, an initial estimate is obtained by averaging the x values of the pixels for which the intensity is greater than $.2z_{min}+.8z_{max}$, where z_{min} and z_{max} are the smallest and largest intensities in the bin. Let our estimate of μ_x for the i,j^{th} bin be X_{ij} . Typically, this gives a good estimate if the spot is bright, but can give a poor estimate if the spot is very weak, due to background noise. However, when a person looks at the image, he or she has no trouble discerning these dim spots, and differentiating them from background noise. The key is that if a slight increase in intensity is perceived in a location that matches the pattern of the other spots, we believe it is a true signal. We can mimic this behavior automatically using modified estimates from the linear regression:

$$X_{ij} = b_0 + b_1 i + b_2 j + b_3 k$$

where $k=0$ if i is odd and $k=1$ if i is even. The parameter b_1 determines the horizontal spacing. The parameter b_2 allows the spots in each row to be horizontally offset from the spots in the row above, which helps with the shear. Finally, b_3 allows an offset due to the print batch (Figure 2). More complicated designs could easily be implemented for more complex printing schemes. If there is a lot of background noise, it is advisable to down-weight the very weak spots (as measured, for example, by the mean pixel intensity in the bin.) We give zero weight to spots for which the mean intensity was lower than the 40th percentile, and weight the other spots equally.

Finding initial estimates of σ is more challenging. We use a grid search for the first iteration. After that, we notice empirically that σ^2 is roughly proportional to the mean

intensity in the bin. So after the first iteration, we can obtain regression estimates to initialize for future iterations. This keeps the σ estimates from blowing up during the nonlinear regression procedure.

Our algorithm has the following steps:

1. Estimate the saturation level.
2. Find the minimum, maximum, and mean intensity for each bin.

For each bin:

3. Estimate μ_x and μ_y .
4. Assign weight 0 to bins with low mean intensity (below the 30th percentile).
5. Perform weighted linear regressions to improve the estimates of μ_x and μ_y .
6. Perform weighted linear regression to improve the estimate of σ . (second and subsequent iterations only)

For each bin, using model (1):

7. Fix μ_x , μ_y , and σ and estimate α and β using linear regression.
8. Fix α and β , and estimate μ_x , μ_y , and σ using nonlinear least squares.

Return to step 5 and iterate.

Finally, the intensity is estimated by integrating (1) to give $I = 2 \pi \sigma^2 \beta$. Using the integral achieves several goals. It allows us to subtract a constant background from each bin, it allows us to estimate what the total intensity would have been had saturation not interfered, and it allows us to include intensity that has extended beyond the edges of the bin. Furthermore, in steps 5 and 6, we can subtract estimated intensities from neighboring bins before we begin, and therefore improve the estimation for weak spots with bright neighbors.

To perform the nonlinear regression, we use a modification of the Levenberg-Marquardt algorithm, originally found in Minpack and translated to java by Steve Verrill: <http://www1.fpl.fs.fed.us/optimization.html>.

For the linear regression solver, we use the JAMA linear algebra package available from <http://math.nist.gov/javanumerics/jama/>.

ImageJ is available from <http://rsb.info.nih.gov/ij/>.

4. Dilution Arrays

In this section, we compare results from our model-based algorithm to results from a standard edge-detection algorithm described in Xie, 2003. The two algorithms are compared on two slides, shown in Figure 5. Each slide is spotted with 8 rows and 12 columns. The concentration decreases by a factor of two for each column, left to right. There is also a trend in the vertical direction, due to experimental technique, but the column relationship is the one we wish to capture.

The edge detection intensities are normalized by taking the log (base 2), subtracting the minimum, and dividing by the median for each slide. The model-based intensities are normalized by taking the square root, subtracting the minimum and dividing by the median. We do not understand why the model-based intensities needed the square root transformation instead of the usual log, but numerous examples showed that the square root consistently performed better.

The top 4 plots of Figure 6 show the results for both methods on both slides. Each row of spots is shown as a connected path. The edge detection plots show a highly nonlinear relationship that differs between slides. The model-based plots show much more linear results, except where the concentration is very close to zero (slide 2). The relationship between intensities for slide 1 and slide 2 is shown in the bottom two plots. The model-based method gives much higher correlation than the edge detection approach.

5. Illustration: Experimental Data

We consider two slides from a set of stress experiments on *lactococcus lactis* (Xie et al., 2002a and 2002b). Each slide has 768 spots, printed as described in Section 2. Figure 7 shows the results of edge detection and Figure 8 shows the results of the model-based approach. Data were normalized as for the dilution experiments. The top two plots in Figure 9 show the relationship between the edge detection estimates and the model-based estimates. These are similar to what we observed in the dilution experiments. The bottom two plots show the relationship between the slides, for each method. The slides are biological replications (different cultures), so it is perhaps not surprising that the relationship should be weak, but the edge-detection results are clearly far from optimal. Figure 10 shows the relationship between the adjacent pairs of spots on the same slide. These are from the same cultures, so we would expect them to be more similar than the slides. Indeed, the correlations are much higher, but again the model-based approach gives more satisfactory results.

6. Concluding Remarks

With a few exceptions such as Yang et al. (2002), statisticians typically become involved in expression array analysis after the spot detection is complete, by which time it may be too late to correct some of the problems created by an inappropriate spot detector. Choice of a good spot detector depends on many aspects of the experimental protocol. We present software that works for our protocol, in the hopes that it may be useful elsewhere, either as it is or after suitable modifications.

7. Acknowledgements

Thanks to Ross Ihaka, Robert Gentleman and the R development team for providing excellent no-cost software. Although our final software is implemented as an ImageJ plugin, R was used for much of the early development of this code, and all subsequent statistical analyses.

8. Bibliography

- R. Ihaka and R. Gentleman (1996). R: a language for data analysis and graphics. *Journal of Computational and Graphical Statistics*, 5:299-314.
- Y. Xie, L. Chou, P. Joseph, A. Cutler and B. Weimer (2002a) Influence of *Lactococcus lactis* ssp. *lactis* IL 1403 by starvation and extracellular proteolytic enzymes from *Brevibacterium linens* BL2. Technical report, Utah State University.
- Y. Xie, Lan-szu Chou, A. Cutler and B. Weimer (2002b) Expression Profiling of *Lactococcus lactis* ssp. *lactis* IL 1403 under stresses with DNA Macroarray. Technical report, Utah State University.
- Y. Xie, (2002) MS Thesis, Computer Science, Utah State University.
- Y. H. Yang, M. J. Buckley, S. Dudoit, and T. P. Speed (2002). Comparison of methods for image analysis on cDNA microarray data. *Journal of Computational and Graphical Statistics*, Vol. 11, No. 1, p. 108--136.

Figure 1

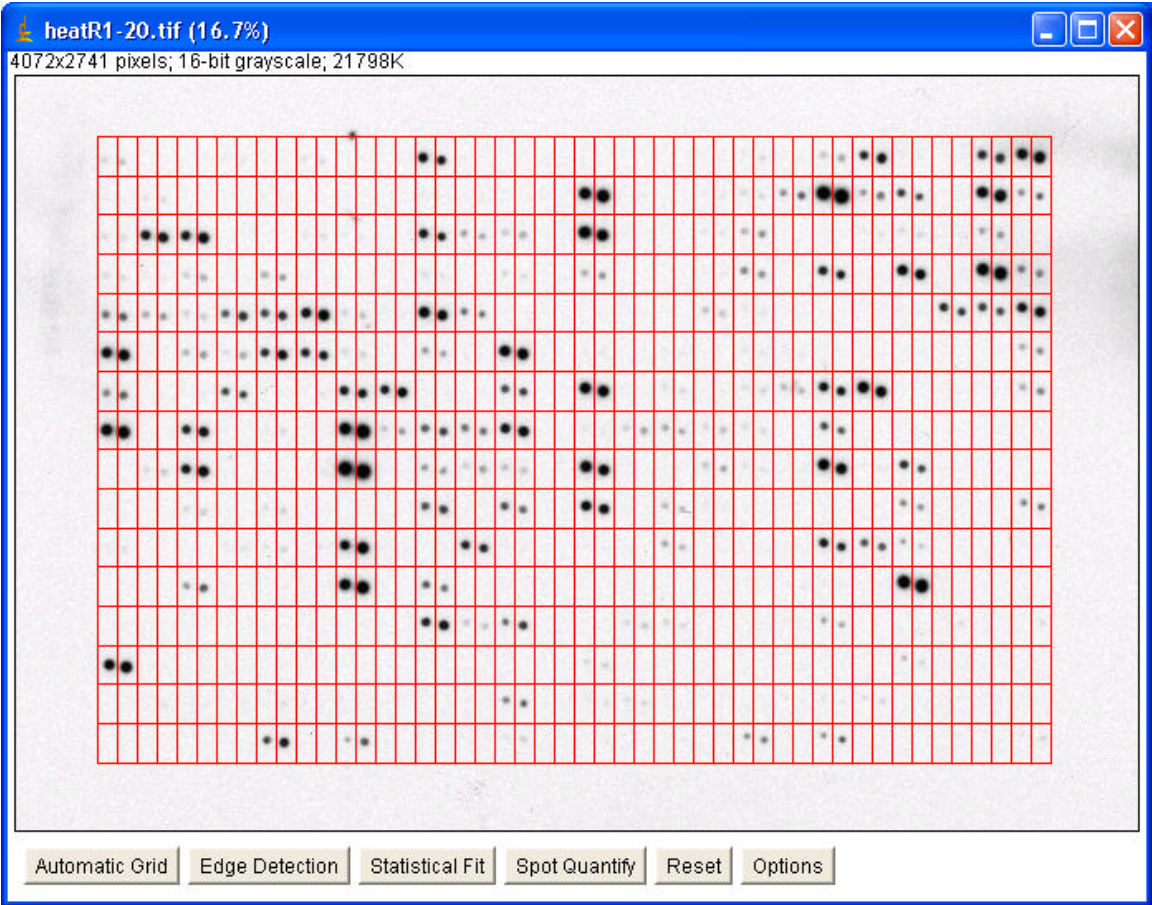


Figure 2a)

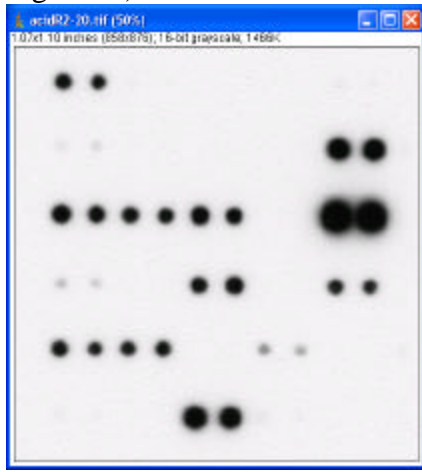


Figure 2b)

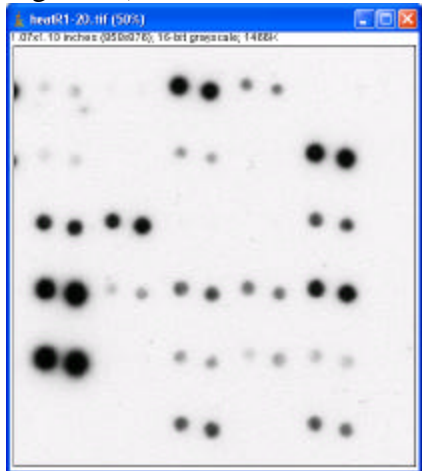


Figure 3a

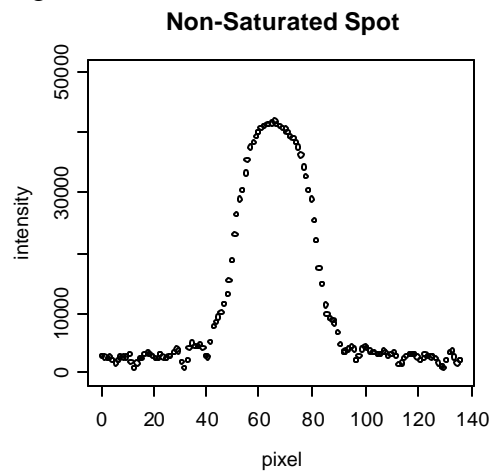


Figure3b

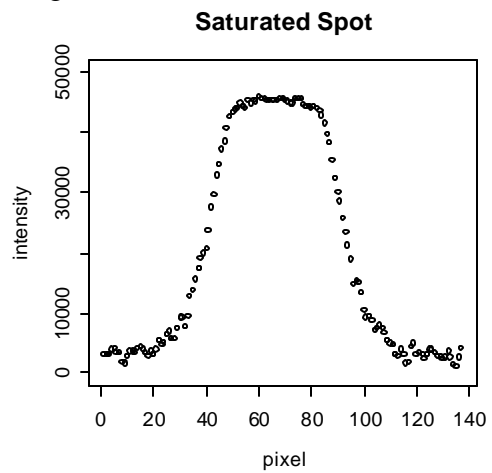


Figure 4a

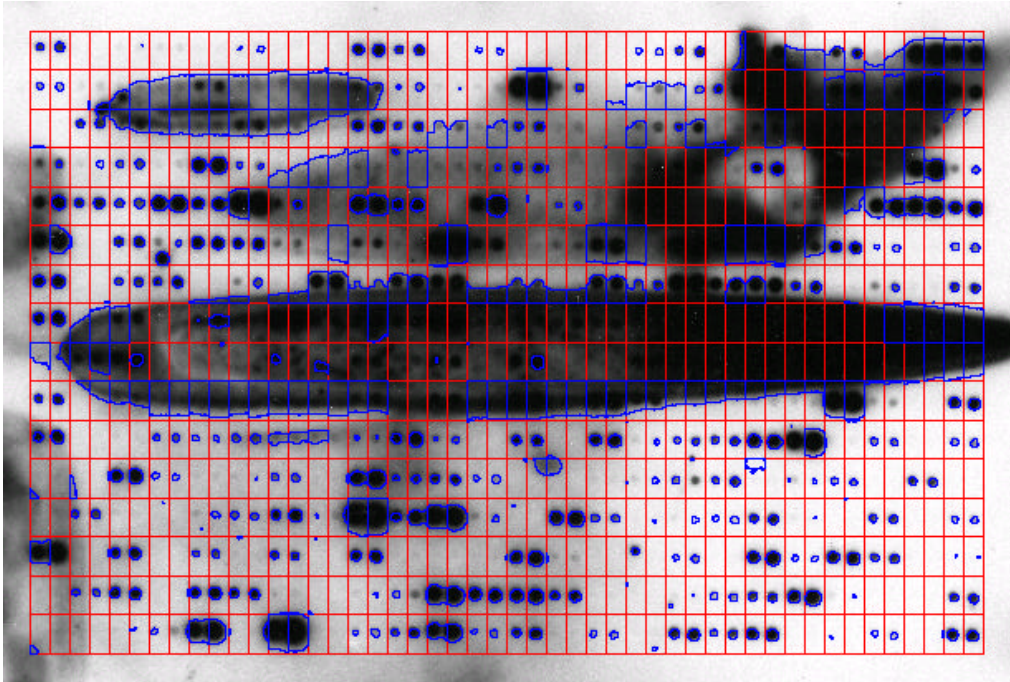


Figure 4b

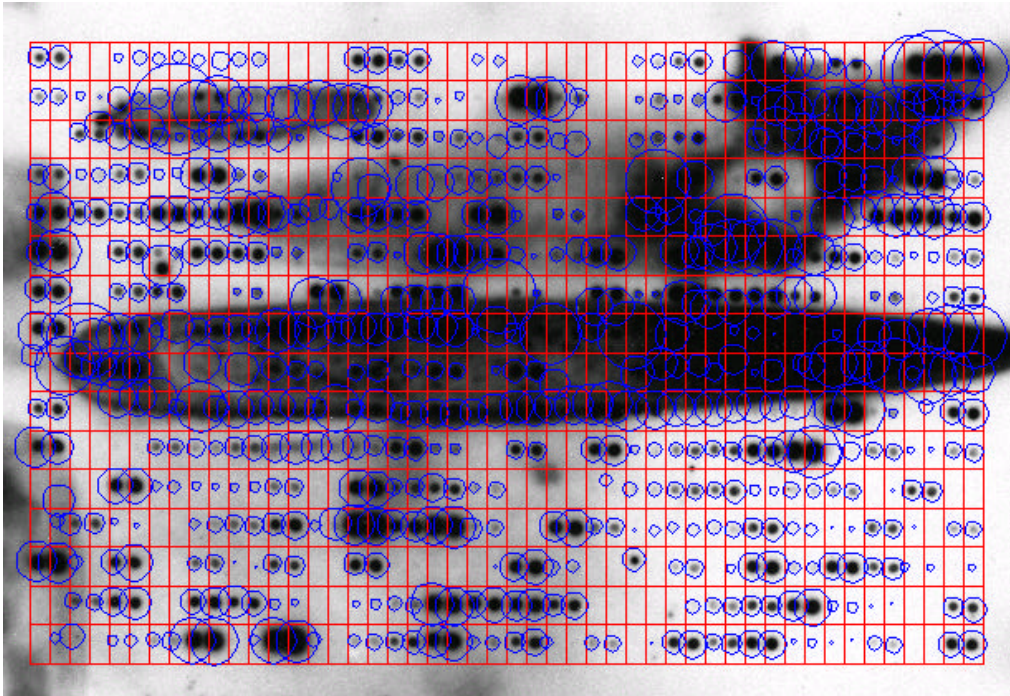
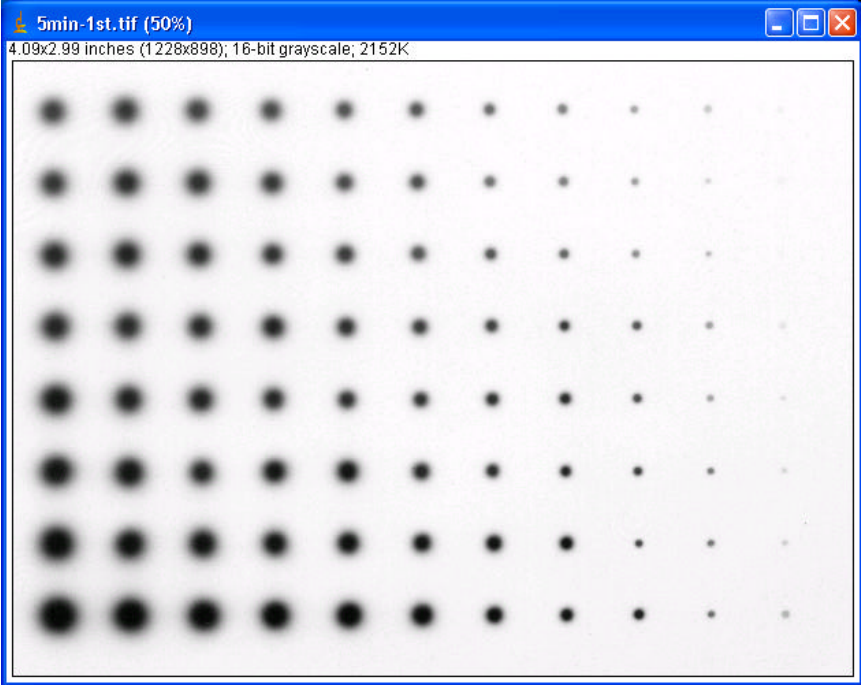


Figure 5

Dilution slide 1



Dilution slide 2



Figure 6

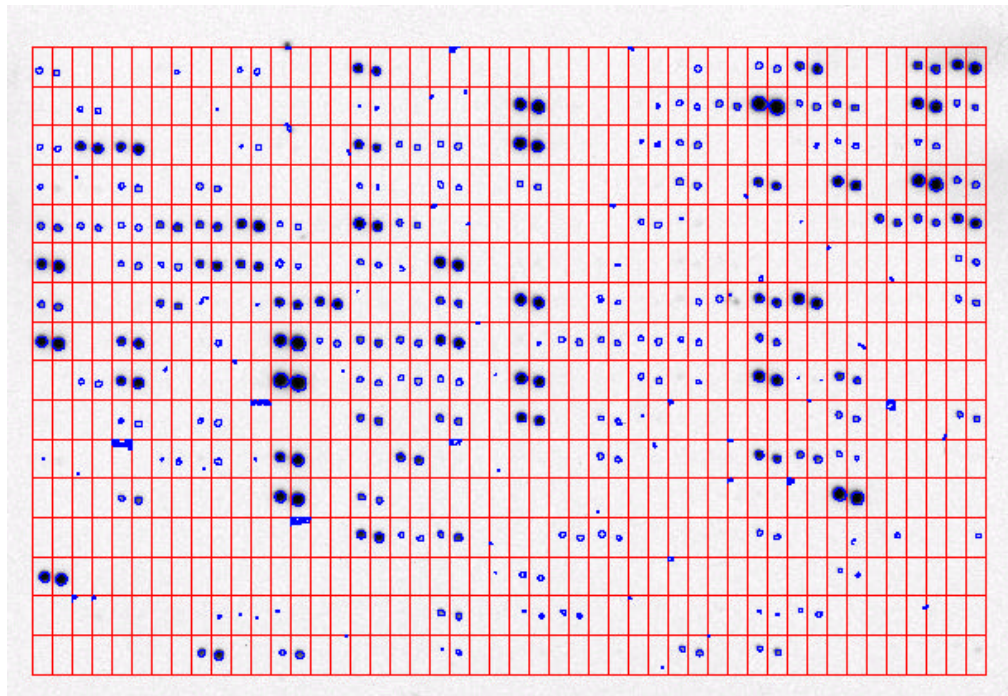


Figure 7

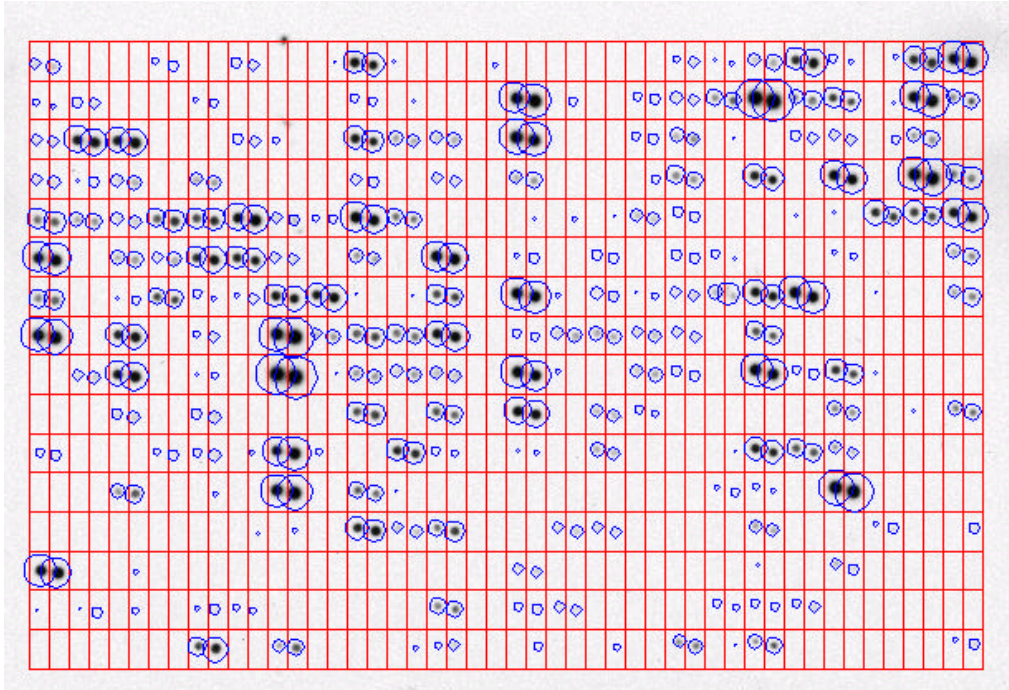


Figure 8 Dilution experiments

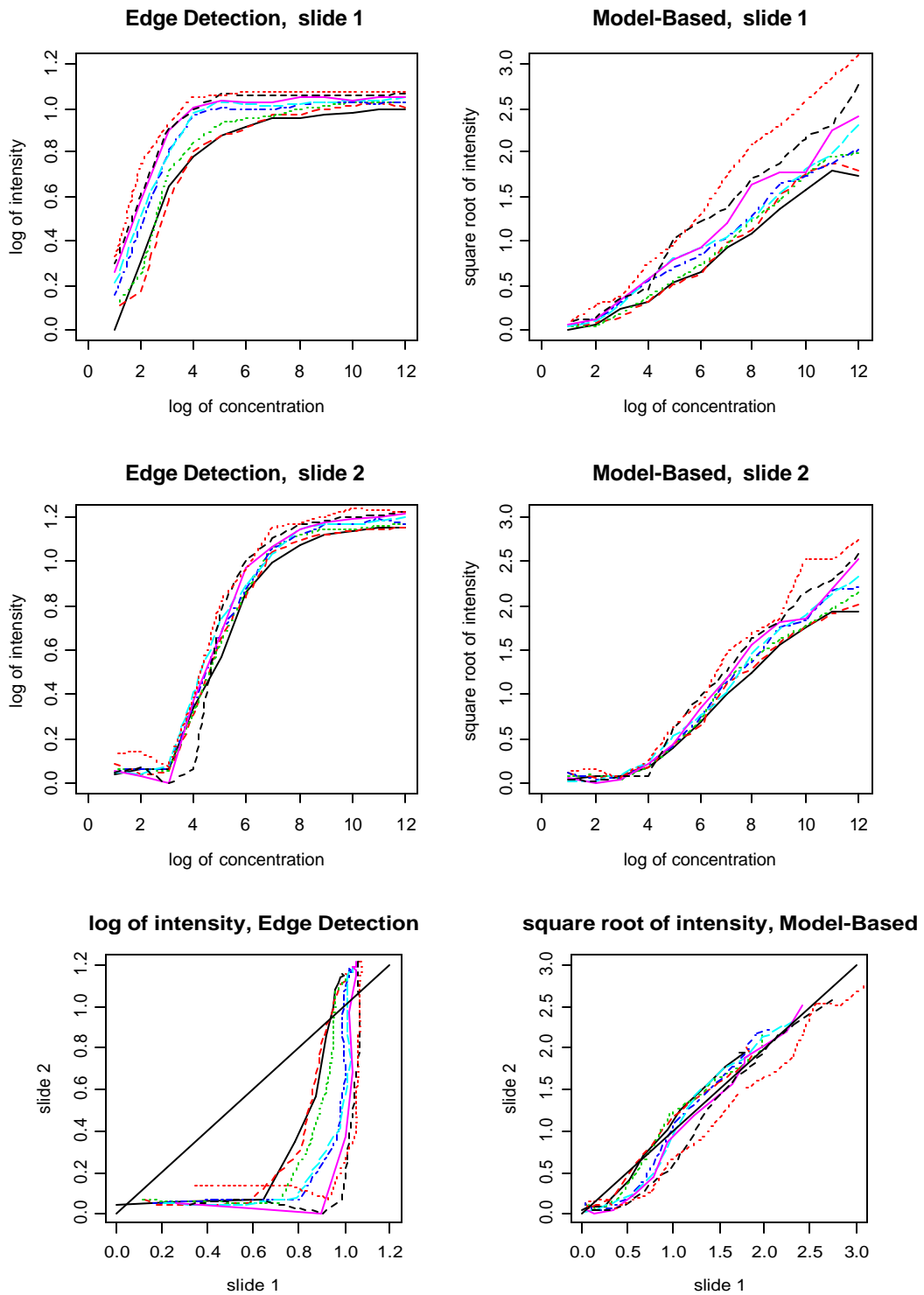


Figure 9 Relationship between methods and between slides

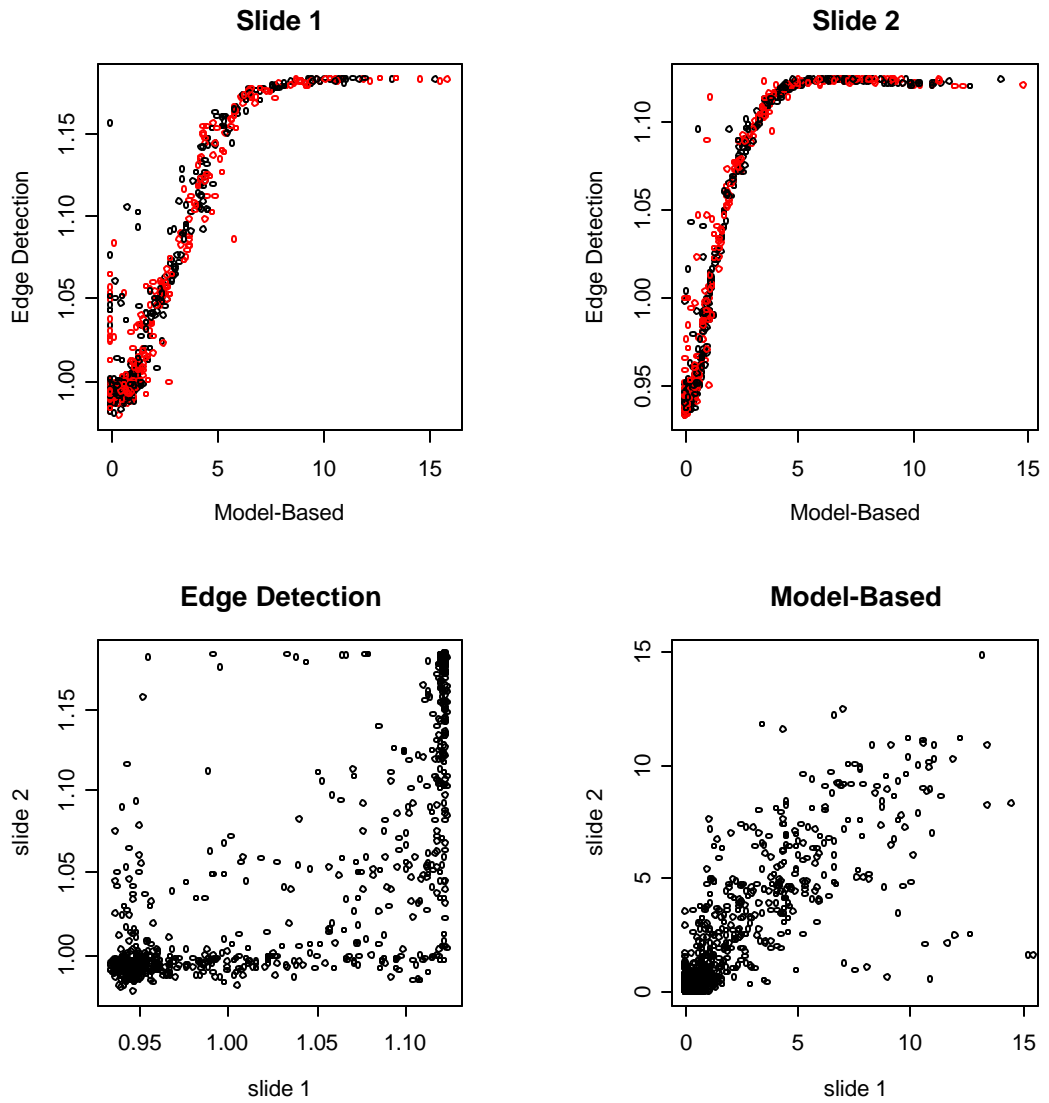


Figure 10 Relationship between replications on the same slide

