Title
Statistical Problems in DNA Microarray Data Analysis

Author
Wang, Nancy Naichao

Publication Date
2009

Peer reviewed|Thesis/dissertation

Statistical Problems in DNA Microarray Data Analysis

by

Nancy Naichao Wang

A dissertation submitted in partial satisfaction of the
requirements for the degree of
Doctor of Philosophy

in

Biostatistics

and the Designated Emphasis

in

Computational and Genomic Biology

in the

GRADUATE DIVISION
of the
UNIVERSITY OF CALIFORNIA, BERKELEY

Committee in charge:
Professor Terence P. Speed, Chair
Professor Sandrine Dudoit
Professor Daniel A. Portnoy

Fall 2009

Statistical Problems in DNA Microarray Data Analysis

Abstract

Statistical Problems in DNA Microarray Data Analysis

by

Nancy Naichao Wang

Doctor of Philosophy in Biostatistics

and the Designated Emphasis in Computational and Genomic Biology

University of California, Berkeley

Professor Terence P. Speed, Chair

DNA microarrays are powerful tools for functional genomics studies. Each array contains thousands of microscopic spots of DNA oligonucleotides with specific sequences, which can hybridize with their complementary DNA sequences. Thus each microarray experiment consists of parallel assays about thousands of genomic fragments. This thesis concerns some statistical issues in the analysis of DNA microarray data.

One common usage of DNA microarrays is to monitor the dynamic levels of gene expression in response to a stimulus. This is often achieved through a time course experiment, in which RNA samples are extracted at various time points after exposing the organism to the stimulus. A particularly interesting type of time course experiments involve replicated series of longitudinal samples. In 2006, Tai and Speed proposed a multivariate empirical Bayes model for analyzing this type of data. The *MB*-statistic derived from this model was shown useful for ranking the genes according to changes in their temporal expression profiles. In the first part of this thesis, we propose an empirical Bayes false discovery rate (FDR)-controlling procedure for multiple hypothesis testing using the *MB*-statistic. A null distribution is obtained using the parametric bootstrap. Critical values are determined according to the empirical Bayes FDR procedure. This method was compared, through simulations, to the frequentist FDR procedure, which requires a theoretical null distribution for calculating the nominal p-values. Although our method is slightly anti-conservative, it is more robust to the variability in the estimates of the hyperparameters, when the degree of moderation is small.

Another common usage of DNA microarrays is to detect genomic locations that are associated with DNA-binding proteins. This is often achieved through ChIP-chip experiments that combine chromatin immunoprecipitation with the microarray technology. Traditional DNA microarrays designed for gene expression studies contain only a few probes for each gene. A special type of DNA microarrays, called tiling arrays, are often used in ChIP-chip experiments. They typically contain probes that are placed densely along the chromosomes to cover either the entire genome or contigs of the genome. A couple of challenges in the analysis of ChIP-chip tiling array data have not been met satisfactorily in the literature.

When large scale genomic studies are carried over a long period of time, tiling arrays with different probe designs are often used for practical reasons. The first challenge is the integration of replicate experiments performed using different tiling array designs. When the biological process of interest involves a large protein complex, the investigators often perform ChIP-chip experiments on each component DNA-binding protein individually. DNA targets that are shared by the individual proteins are thought to be the localization sites of the protein complex. The second challenge is the joint analysis of multiple DNA-binding proteins, aimed at identifying their shared targets. In the second part of this thesis, we propose a nonhomogeneous hidden Markov model (HMM) for addressing these two challenges. The nonhomogeneous time axis represents the genomic positions of the probes. The hidden states represent the binding statuses of the proteins. The state-conditional emission distributions of the tiling array data are protein-specific and design-specific. We derived a modified Baum-Welch algorithm for fitting the model parameters. We also developed a procedure that converts the probe level summaries into peaks, which represent the putative binding sites, based on both signal strength and peak shape. To compare our method with existing methods, we curated a set of positive and negative genomic regions from a *C. elegans* dataset, and performed some receiver operating characteristics (ROC) analyses. When applied to each experiment separately, our method performs similarly as the three best existing methods. When applied to the combined data set, which consists of tiling arrays with different probe designs, our method shows a drastic improvement in performance. A generalization of the nonhomogeneous HMM enables the joint analysis of the ChIP-chip data of multiple proteins. We present an application of this method to identify the shared localization sites of two DNA-binding proteins, under two different conditions.

To my parents.

Contents

List of Figures **iv**

List of Tables **xv**

1 Introduction and outline **1**
1.1 Overview of DNA Microarrays 1
1.2 Gene expression time course analysis 3
1.3 ChIP-chip and tiling arrays 5
1.4 Outline of the thesis 8

2 An FDR-controlling procedure for analyzing replicated microarray time course data with the multivariate empirical Bayes statistic **10**
2.1 Motivation 10
2.2 The multivariate empirical Bayes statistic 11
2.2.1 Distribution of the moderated Hotelling T^2 statistic 12
2.3 Frequentist FDR-controlling procedure 14
2.4 Empirical Bayes FDR-controlling procedure 14
2.5 Comparisons of the FDR procedures through simulations 16
2.5.1 Effects of sample size 17
2.5.2 Effects of moderation 21
2.6 Application to real data 21
2.7 Summary of results 30

3 A nonhomogeneous hidden Markov model for integrating ChIP-chip data from multiple tiling array designs **35**
3.1 Motivation 35
3.2 Nonhomogeneous hidden Markov model for one protein 37
3.2.1 Approximation of the transition matrix 39
3.2.2 Modified forward-backward algorithm 40
3.2.3 Modified Baum-Welch algorithm 40
3.2.4 Initialization of the parameter estimates 45
3.3 Simulation study for one protein 46
3.4 From candidate probes to binding peaks 57
3.5 Comparisons with existing methods for ChIP-chip data analysis 70

3.5.1 Review of the three best existing methods 72
3.5.2 Results of comparisons by ROC 74

4 A peak-detection method for the joint analysis of ChIP-chip data from multiple DNA binding proteins 85
4.1 Motivation 85
4.2 Conditional independence of the observations 86
4.3 Nonhomogeneous hidden Markov model for multiple proteins 92
4.4 Simulation study for two proteins 94
4.5 Comparison with the alternative approach 109
4.6 Issues in the application to real data 115
4.6.1 Preprocessing of tiling array data 115
4.6.2 Estimation of emission parameters 119
4.6.3 Analysis of IgG control data 122
4.6.4 Summary of results 133

5 Conclusions and discussion 138
5.1 FDR procedure for microarray time course data 138
5.2 Nonhomogeneous HMM for ChIP-chip data 139

Bibliography 141

List of Figures

1.1 Principle of DNA microarrays 2

2.1 Distributions of the F-transformed $\widetilde{T}^2$ in small samples. The dimension of the multivariate normal is $k = 6$ and the sample size is $n = 6$. The true prior degrees of freedom used to simulate the data is $\nu = 12$. The estimate obtained from the simulated data set is $\hat{\nu} = 7$. (a) 100% null genes; (b) 90% null genes. 18

2.2 QQ-plots of the bootstrap null distribution vs. the observed (mixture) distribution and the theoretical null distribution. The dimension of the multivariate normal is $k = 6$ and the sample size is $n = 6$. (a) Bootstrap null versus observed, 100% null genes. (b) Bootstrap null versus theoretical, 100% null genes. (c) Bootstrap null versus observed, 90% null genes. (d) Bootstrap null versus theoretical, 90% null genes. 19

2.3 Distributions of the F-transformed $\widetilde{T}^2$ in large samples. The dimension of the multivariate normal is $k = 6$ and the sample size is $n = 30$. The true prior degrees of freedom used to simulate the data is $\nu = 12$. The estimate obtained from the simulated data set is $\hat{\nu} = 7$. (a) 100% null genes; (b) 90% null genes. 20

2.4 Comparison of FDR procedures applied to the moderated Hotelling T^2 statistics. The dimension of the multivariate normal is $k = 6$. (a) sample size $n = 6$; (b) sample size $n = 30$. 22

2.5 Effects of moderation on the distribution of the moderated Hotelling T^2 statistics. Each condition was examined by simulating two data sets. For each data set, the hyperparameters were estimated and the $\widetilde{T}^2$ statistics were computed using the *timecourse* package. The dimension of the multivariate normal is $k = 6$ and the sample size is $n = 6$. (a) true $\nu = 6$, estimate $\hat{\nu} = 1$, 100% null genes. (b) true $\nu = 60$, estimate $\hat{\nu} = 53$, 100% null genes. (c) true $\nu = 6$, estimate $\hat{\nu} = 1$, 90% null genes. (d) true $\nu = 60$, estimate $\hat{\nu} = 61$, 90% null genes. 23

2.6 Effects of moderation on the distribution of the moderated Hotelling T^2 statistics. The known true hyperparameters were plugged-in. The dimension of the multivariate normal is $k = 6$ and the sample size is $n = 6$. (a) true $\nu = 6$, 100% null genes. (b) true $\nu = 60$, 100% null genes. (c) true $\nu = 6$, 90% null genes. (d) true $\nu = 60$, 90% null genes. 24

2.7 Effects of moderation on the FDR-controlling procedures for the moderated Hotelling T^2 statistics. The dimension of the multivariate normal is $k = 6$ and the sample size is $n = 6$. (a) true $\nu = 6$; (b) true $\nu = 60$. 25
2.8 Distribution of the F-transformed moderated Hotelling T^2 statistics for the two subsets of the real data, and simulations based on the parameters estimated from the real data. The dimension of the multivariate normal is $k = 3$ and the sample size is $n = 4$ in each subset. Top panels: Subset A contains the 0, 1, 5 hours time points. Bottom panels: Subset B contains the 0.25, 3, 7 hours time points. Left panels: real data. Right panels: simulated data. . . 27
2.9 Comparison of the two FDR procedures applied to the moderated Hotelling T^2 statistics computed from the individual subsets. The dimension of the multivariate normal is $k = 3$ and the sample size is $n = 4$ in each subset. Top panels: Subset A contains the 0, 1, 5 hours time points. Bottom panels: Subset B contains the 0.25, 3, 7 hours time points. Left panels: real data. Right panels: simulated data. 28
2.10 Temporal expression profile of Nonexpressor of pathogenesis-related genes 1 (NPR1). Shown at the top of this plot is the rank of the moderated Hotelling T^2 statistic obtained from the analysis of the full data set, containing 6 time points. Each curve represents an independent longitudinal replicate of the time course, indexed by 11 to 14. 31
2.11 Temporal expression profiles of 4 genes that were selected by the empirical Bayes IUT, but not by the frequentist IUT, at 5% FDR. Shown at the top of each plot is the rank of the moderated Hotelling T^2 statistic obtained from the analysis of the full data set, containing 6 time points. Each curve represents an independent longitudinal replicate of the time course, indexed by 11 to 14. 32
2.12 Temporal expression profiles of 4 genes that were selected by the frequentist IUT, but not by the empirical Bayes IUT, at 5% FDR. Shown at the top of each plot is the rank of the moderated Hotelling T^2 statistic obtained from the analysis of the full data set, containing 6 time points. Each curve represents an independent longitudinal replicate of the time course, indexed by 11 to 14. 33

3.1 Data integration approach 37
3.2 Nonhomogeneous hidden Markov model 38
3.3 Emission Parameters (Design 1): 100 simulations of tiling array data were generated according to the chromosomal coordinates of the probes in a 1 MB region of Chromosome X. Each simulated data set contained one array of each design, summing up to 3 "replicates". Estimates of the emission parameters for Design 1 are summarized as boxplots. In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the estimates. The black dashed lines represent Mean +/− 2 SDs of the estimates. 49

3.4 Emission Parameters (Design 2): 100 simulations of tiling array data were generated according to the chromosomal coordinates of the probes in a 1 MB region of Chromosome X. Each simulated data set contained one array of each design, summing up to 3 "replicates". Estimates of the emission parameters for Design 2 are summarized as boxplots. In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the estimates. The black dashed lines represent Mean +/− 2 SDs of the estimates. 50

3.5 Emission Parameters (Design 3): 100 simulations of tiling array data were generated according to the chromosomal coordinates of the probes in a 1 MB region of Chromosome X. Each simulated data set contained one array of each design, summing up to 3 "replicates". Estimates of the emission parameters for Design 3 are summarized as boxplots. In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the estimates. The black dashed lines represent Mean +/− 2 SDs of the estimates. 51

3.6 Stationary Distribution: 100 simulations of tiling array data were generated according to the chromosomal coordinates of the probes in a 1 MB region of Chromosome X. Each simulated data set contained 3 replicates, with one from each design. Estimates of the initial probabilities are summarized as boxplots. In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates. 52

3.7 Transition Matrix (variable Δ_k): 100 simulations of tiling array data were generated according to the chromosomal coordinates of the probes in a 1 MB region of Chromosome X. Each simulated data set contained one array of each design, summing up to 3 "replicates". Estimates of the transition probabilities are summarized as boxplots. In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the estimates. The black dashed lines represent Mean +/− 2 SDs of the estimates. 53

3.8 Transition Matrix ($\Delta_k = 20$): 100 simulations of tiling array data were generated according to a hypothetical design with probes placed at every 20 base pairs. Each simulated data set contained one array. Estimates of the transition probabilities are summarized as boxplots. In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates. 54

3.9 Transition Matrix ($\Delta_k = 10$): 100 simulations of tiling array data were generated according to a hypothetical design with probes placed at every 10 base pairs. Each simulated data set contained one array. Estimates of the transition probabilities are summarized as boxplots. In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates. 55

3.10 Transition Matrix ($\Delta_k = 2$): 100 simulations of tiling array data were generated according to a hypothetical design with probes placed at every other base pair. Each simulated data set contained one array. Estimates of the transition probabilities are summarized as boxplots. In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates. 56

3.11 Confusion Matrix (variable Δ_k): 100 simulations of tiling array data were generated according to the chromosomal coordinates of the probes in a 1 MB region of Chromosome X. Each simulated data set contained one array of each design, summing up to 3 "replicates". The inferred states of the observed positions were compared against the true states. A confusion matrix was tabulated for each simulated data set, and its entries were normalized into percentages of the true states. Each panel summarizes one entry in the confusion matrix across 100 simulations. The inner-quartile range is delineated by the borders of the box, with a thick line drawn at the median. The whiskers extend to the extreme values. The black dashed lines represent Mean +/− 2 SDs. . . . 58

3.12 Confusion Matrix ($\Delta_k = 20$): 100 simulations of tiling array data were generated according to a hypothetical design with probes placed at every 20 base pairs. Each simulated data set contained one array. The inferred states of the observed positions were compared against the true states. A confusion matrix was tabulated for each simulated data set, and its entries were normalized into percentages of the true states. Each panel summarizes one entry in the confusion matrix across 100 simulations. The inner-quartile range is delineated by the borders of the box, with a thick line drawn at the median. The whiskers extend to the extreme values. The black dashed lines represent Mean +/− 2 SDs. 59
3.13 Confusion Matrix ($\Delta_k = 10$): 100 simulations of tiling array data were generated according to a hypothetical design with probes placed at every 10 base pairs. Each simulated data set contained one array. The inferred states of the observed positions were compared against the true states. A confusion matrix was tabulated for each simulated data set, and its entries were normalized into percentages of the true states. Each panel summarizes one entry in the confusion matrix across 100 simulations. The inner-quartile range is delineated by the borders of the box, with a thick line drawn at the median. The whiskers extend to the extreme values. The black dashed lines represent Mean +/− 2 SDs. 60
3.14 Confusion Matrix ($\Delta_k = 2$): 100 simulations of tiling array data were generated according to a hypothetical design with probes placed at every other base pair. Each simulated data set contained one array. The inferred states of the observed positions were compared against the true states. A confusion matrix was tabulated for each simulated data set, and its entries were normalized into percentages of the true states. Each panel summarizes one entry in the confusion matrix across 100 simulations. The inner-quartile range is delineated by the borders of the box, with a thick line drawn at the median. The whiskers extend to the extreme values. The black dashed lines represent Mean +/− 2 SDs. 61
3.15 An ideal symmetric triangular peak. 62
3.16 A false positive rectangular interval. 63
3.17 Non-ideal peaks. The first peak has the shape of an asymmetric triangle. The second peak is the product of multiple nearby binding sites being merged together. Both peaks are probably real binding sites. 64
3.18 A repetitive region that has a false positive interval due to non-specific binding. The middle track has missing data in this region, because the repetitive regions were masked out on Design 2. 66
3.19 Scatter plots of all peaks along the first two principal components. Peaks that overlap with the curated positive regions in the training set are colored in red. Peaks that overlap with the curated negative regions in the training set are colored in green. 69

3.20 ROC analysis of the peak filtering procedure for various data sets. True positive and false positive rates are defined in terms of the intervals that overlap with the curated positive and negative regions in the training set. . . 71
3.21 ROC curves for wild type Dpy-27. Each experiment was analyzed separately by: HMM (red, dashed), TAMALPAIS (green, dashed), NimbleScan (cyan, dashed), MA2C (blue, dashed). The combined set of 3 replicates was analyzed by HMM (red, solid). Right-hand side panels are zoomed in versions of left-hand side panels. 76
3.22 ROC curves for wild type Sdc-3. Each experiment was analyzed separately by: HMM (red, dashed), TAMALPAIS (green, dashed), NimbleScan (cyan, dashed), MA2C (blue, dashed). The combined set of 3 replicates was analyzed by HMM (red, solid). Right-hand side panels are zoomed in versions of left-hand side panels. 77
3.23 ROC curves for *smo-1* mutant Dpy-27. Each experiment was analyzed separately by: HMM (red, dashed), TAMALPAIS (green, dashed), NimbleScan (cyan, dashed), MA2C (blue, dashed). The combined set of 3 replicates was analyzed by HMM (red, solid). Right-hand side panels are zoomed in versions of left-hand side panels. 78
3.24 ROC curves for *smo-1* mutant Sdc-3. Each experiment was analyzed separately by: HMM (red, dashed), TAMALPAIS (green, dashed), NimbleScan (cyan, dashed), MA2C (blue, dashed). The combined set of 2 replicates was analyzed by HMM (red, solid). Right-hand side panels are zoomed in versions of left-hand side panels. 79
3.25 Weak peaks that were called by the other methods but not by the nonhomogeneous HMM, when applied to each replicate separately. Yellow intervals represent peaks called by the combined version of the nonhomogeneous HMM. Red intervals represent peaks called by the single experiment version of the nonhomogeneous HMM. Here, red intervals are missing because the single experiment analysis did not recognize any peaks. But the combined analysis recognized a peak. Green intervals represent peaks called by TAMALPAIS. Cyan intervals represent peaks called by NimbleScan. Blue intervals represent peaks called by MA2C. 80
3.26 Weak peaks that were called by the other methods but not by the nonhomogeneous HMM, when applied to each replicate separately. Yellow intervals represent peaks called by the combined version of the nonhomogeneous HMM. Red intervals represent peaks called by the single experiment version of the nonhomogeneous HMM. Here, red intervals are missing because the single experiment analysis did not recognize any peaks. But the combined analysis recognized a peak. Green intervals represent peaks called by TAMALPAIS. Cyan intervals represent peaks called by NimbleScan. Blue intervals represent peaks called by MA2C. 81

3.27 Peaks that were called by the other methods but not by our method due to postprocessing. Yellow intervals represent peaks called by the combined version of the nonhomogeneous HMM. Red intervals represent peaks called by the single experiment version of the nonhomogeneous HMM. Green intervals represent peaks called by TAMALPAIS. Cyan intervals represent peaks called by NimbleScan. Blue intervals represent peaks called by MA2C. The intervals in the region between 1,172,000 and 1,176,000 were filtered out by our method due to their roughly rectangular shapes. 82
3.28 Peaks that were called by the other methods but not by our method due to postprocessing. Yellow intervals represent peaks called by the combined version of the nonhomogeneous HMM. Red intervals represent peaks called by the single experiment version of the nonhomogeneous HMM. Green intervals represent peaks called by TAMALPAIS. Cyan intervals represent peaks called by NimbleScan. Blue intervals represent peaks called by MA2C. The intervals in the region between 2,712,500 and 2,716,500 were filtered out by our method due to their roughly rectangular shapes. 83

4.1 Correlation plots for tiling arrays of Design 2. Wild type data for a pair of ChIP-chip experiments are stratified by the inferred states in the two-protein model. Each data point represents a probe on tiling array Design 2. There appears to be a positive correlation between Dpy-27 and Sdc-3 in State 11. . 87
4.2 Correlation plots for tiling arrays of Design 3. Wild type data for a pair of ChIP-chip experiments are stratified by the inferred states in the two-protein model. Each data point represents a probe on tiling array Design 3. There appears to be a positive correlation between Dpy-27 and Sdc-3 in State 11. . 88
4.3 Confusion matrices of true states versus inferred states were computed for 100 simulations. The differences between the diagonal elements in the confusion matrices of the two models were divided by their average counts, to obtain the percent differences. Solid lines represent the medians. Dashed lines represent 2 SDs away from the means. 91
4.4 Four Hidden States of the Two-Protein Model 93
4.5 Emission Parameters (Protein 1, Design 1): 100 simulations of tiling array data were generated according to the chromosomal coordinates of the probes in a 1 MB region of Chromosome X. Each simulated data set contained 3 replicates for each protein, with one replicate from each design. Estimates of the emission parameters for Protein 1 on Design 1 are summarized as boxplots. In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean $+/-$ 2 SDs of the parameter estimates. 98

4.6 Emission Parameters (Protein 1, Design 2): 100 simulations of tiling array data were generated according to the chromosomal coordinates of the probes in a 1 MB region of Chromosome X. Each simulated data set contained 3 replicates for each protein, with one replicate from each design. Estimates of the emission parameters for Protein 1 on Design 2 are summarized as boxplots. In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates. 99
4.7 Emission Parameters (Protein 1, Design 3): 100 simulations of tiling array data were generated according to the chromosomal coordinates of the probes in a 1 MB region of Chromosome X. Each simulated data set contained 3 replicates for each protein, with one replicate from each design. Estimates of the emission parameters for Protein 1 on Design 3 are summarized as boxplots. In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates. 100
4.8 Emission Parameters (Protein 2, Design 1): 100 simulations of tiling array data were generated according to the chromosomal coordinates of the probes in a 1 MB region of Chromosome X. Each simulated data set contained 3 replicates for each protein, with one replicate from each design. Estimates of the emission parameters for Protein 2 on Design 1 are summarized as boxplots. In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates. 101
4.9 Emission Parameters (Protein 2, Design 2): 100 simulations of tiling array data were generated according to the chromosomal coordinates of the probes in a 1 MB region of Chromosome X. Each simulated data set contained 3 replicates for each protein, with one replicate from each design. Estimates of the emission parameters for Protein 2 on Design 2 are summarized as boxplots. In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates. 102

4.10 Emission Parameters (Protein 2, Design 3): 100 simulations of tiling array data were generated according to the chromosomal coordinates of the probes in a 1 MB region of Chromosome X. Each simulated data set contained 3 replicates for each protein, with one replicate from each design. Estimates of the emission parameters for Protein 2 on Design 3 are summarized as boxplots. In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates. 103
4.11 Stationary Distribution: 100 simulations of tiling array data were generated according to the chromosomal coordinates of the probes in a 1 MB region of Chromosome X. Each simulated data set contained 3 replicates for each protein, with one replicate from each design. Estimates of the initial probabilities are summarized as boxplots. In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates. 104
4.12 Transition Matrix (variable Δ_k): In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates. 105
4.13 Transition Matrix ($\Delta_k = 20$): In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates. 106
4.14 Transition Matrix ($\Delta_k = 10$): In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates. 107
4.15 Transition Matrix ($\Delta_k = 2$): In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates. 108

4.16 Confusion Matrix (variable Δ_k): A confusion matrix was tabulated for each simulated data set, and its entries were normalized into percentages of the true states. Each panel summarizes one entry in the confusion matrix across 100 simulations. The inner-quartile range is delineated by the borders of the box, with a thick line drawn at the median. The whiskers extend to the extreme values. The black dashed lines represent Mean +/− 2 SDs. 110
4.17 Confusion Matrix ($\Delta_k = 20$): A confusion matrix was tabulated for each simulated data set, and its entries were normalized into percentages of the true states. Each panel summarizes one entry in the confusion matrix across 100 simulations. The inner-quartile range is delineated by the borders of the box, with a thick line drawn at the median. The whiskers extend to the extreme values. The black dashed lines represent Mean +/− 2 SDs. 111
4.18 Confusion Matrix ($\Delta_k = 10$): A confusion matrix was tabulated for each simulated data set, and its entries were normalized into percentages of the true states. Each panel summarizes one entry in the confusion matrix across 100 simulations. The inner-quartile range is delineated by the borders of the box, with a thick line drawn at the median. The whiskers extend to the extreme values. The black dashed lines represent Mean +/− 2 SDs. 112
4.19 Confusion Matrix ($\Delta_k = 2$): A confusion matrix was tabulated for each simulated data set, and its entries were normalized into percentages of the true states. Each panel summarizes one entry in the confusion matrix across 100 simulations. The inner-quartile range is delineated by the borders of the box, with a thick line drawn at the median. The whiskers extend to the extreme values. The black dashed lines represent Mean +/− 2 SDs. 113
4.20 ROC curves comparing the nonhomogeneous HMM with the MA2C cross-listing method, when used to identify the shared binding sites of Dpy-27 and Sdc-3. The nonhomogeneous HMM is shown in red; MA2C is shown in blue. The right-hand side panels are zoomed-in versions of the left-hand side panels. 116
4.21 QQ-plots of wild type Dpy-27 replicate data on Designs 1 and 2 before and after quantile normalization. The 45° line is drawn in red. 118
4.22 QQ-plots of wild type Dpy-27 replicate data on Designs 2 and 3 before and after quantile normalization. The 45° line is drawn in red. 118
4.23 QQ-plots of wild type Dpy-27 replicate data on Designs 3 and 1 before and after quantile normalization. The 45° line is drawn in red. 119
4.24 Boxplots of pair-wise differences between replicates of wild type Dpy-27, before and after quantile normalization. The 25th and 75th quantiles are represented by the boundaries of each box. The median is drawn as a thick black horizontal line. The whiskers extend to the most extreme data points within one interquartile range from the box. 120
4.25 Histograms of Dpy-27 observations on tiling arrays of Design 1 123
4.26 Histograms of Dpy-27 observations on tiling arrays of Design 2 124
4.27 Histograms of Dpy-27 observations on tiling arrays of Design 3 125
4.28 QQ-plots of Dpy-27 observations on tiling arrays of Design 1 126

4.29 QQ-plots of Dpy-27 observations on tiling arrays of Design 2 126
4.30 QQ-plots of Dpy-27 observations on tiling arrays of Design 3 127
4.31 An IgG interval of State 1: The log ratios of three IgG control experiments are displayed in cyan. The posterior probabilities of State 1 are shown in orange. The posterior probabilities of State 2 are shown in pink. 129
4.32 An IgG interval of State 1: The log ratios of three IgG control experiments are displayed in cyan. The posterior probabilities of State 1 are shown in orange. The posterior probabilities of State 2 are shown in pink. 130
4.33 An IgG interval of State 2: The log ratios of three IgG control experiments are displayed in cyan. The posterior probabilities of State 1 are shown in orange. The posterior probabilities of State 2 are shown in pink. 131
4.34 An IgG interval of State 2: The log ratios of three IgG control experiments are displayed in cyan. The posterior probabilities of State 1 are shown in orange. The posterior probabilities of State 2 are shown in pink. 132
4.35 Peak filtering results (training set): Scatter plots of the peaks along the principal components. Peaks that overlap with the curated positive (negative) regions in the training set are colored in red (green). The peak score cutoffs are represented by the solid blue lines. 135
4.36 Peak filtering results (testing set): Scatter plots of the peaks along the principal components. Peaks that overlap with the curated positive (negative) regions in the testing set are colored in red (green). The peak score cutoffs are represented by the solid blue lines. 136

List of Tables

2.1 Hyperparameters used in the simulations aimed at emulating the setting of the real data. 29

3.1 Summary of tiling array designs 36
3.2 Summary of ChIP-chip experiments 36
3.3 Curated sets of positive and negative regions 67
3.4 Summary of peak calls by various methods 84

4.1 Emission parameters for simulating Dpy-27 data 89
4.2 Emission parameters for simulating Sdc-3 data 89
4.3 States reconstruction by the conditional independence model 89
4.4 States reconstruction by the bivariate model 90
4.5 Percent differences between the diagonal elements in the confusion matrices of the two models 90
4.6 Emission parameters for Protein 1 96
4.7 Emission parameters for Protein 2 96
4.8 Emission parameters for IgG on tiling arrays of Design 3 123
4.9 Emission parameters for Dpy-27 133
4.10 Emission parameters for Sdc-3 133
4.11 Overall error rates of the final peak calls 137

Acknowledgments

I would like to express my deepest appreciation to my advisor, Professor Terry Speed, for not only his tireless guidance throughout my graduate career, but also his personal impacts that will last a life time. He taught me statistical rigor when my reasoning was less than rigorous. He encouraged me to move forward when I fidgeted in the face of uncertainties. He gave me patience when I needed time to grow... He showed me what it means to be both a great statistician and a great mentor. I also would like to express my gratitude to my committee members, Professor Sandrine Dudoit and Professor Dan Portnoy, for their helpful advices and supports.

I thank my collaborators in Mary Wildermuth's group for providing the data that motivated the study described in Chapter 2 of my thesis. I am very thankful to my collaborators in Barbara Meyer's group for motivating the project described in Chapter 3 and 4 of my thesis. This project would not have been possible without their help. I especially enjoyed working with Rebecca Pferdehirt and Ed Ralston, who made learning about *C. elegans* genetics a fun experience.

I would like to thank all members of the Speed group for providing such a pleasant learning and research environment. In particular, I benefited substantially from the insightful discussions with Yu Chuan Tai and Soyeon Ahn, about microarray time course data analysis and ChIP-chip data analysis, respectively. I also thank my classmates Jing Lei and Frances Tong for proof-reading this manuscript.

Finally, I thank my family for their love and care since the first day of my life. I thank my parents for bringing me to this wondrous world and raising me in the most nurturing environment. I thank my sister for looking out for me since childhood. I thank my fiancé for being everything that I could ever ask for...

Chapter 1

Introduction and outline

1.1 Overview of DNA Microarrays

Ever since the discovery of the double helix in 1953, scientists and amateurs alike have been marveled at the vast genetic information encoded by DNA. The Human Genome Project was launched in 1990 to dissect the genetic makeup of the human species. Although a complete draft of the human genome was release in 2003, a long journey still lies ahead before the mystery of the genetic code can be unraveled. A relatively new field of biology called "functional genomics" aims at making use of the data produced by genome sequencing projects to better understand gene functions. An important technology that enables research in functional genomics is the DNA microarray technology. According to the American Heritage Science Dictionary (2005), a DNA microarray is "a small solid support, usually a membrane or glass slide, on which sequences of DNA are fixed in an orderly arrangement. DNA microarrays are used for rapid surveys of the expression of many genes simultaneously, as the sequences contained on a single microarray can number in the thousands." Figure 1.1 illustrates the basic principle of DNA microarrays. It relies on the affinity of single stranded nucleic acids to bind, or hybridize, with their complementary sequences. A piece of single stranded DNA that matches with a unique segment of the genome is called a probe. Thousands of probes are attached to the surface of a chip in a rectangular grid, with one spot representing a specific sequence. The sample of interest, which could be either DNA or RNA, is converted into single strands of fluorescently labeled target DNA. After mixing on the chip, each piece of target DNA pairs up with its complementary probe, and the corresponding spot is detected by a fluorescence scanner. What distinguishes DNA microarrays from traditional molecular biology techniques is that thousands of assays can be carried out simultaneously.

Different types of DNA microarrays are produced using different fabrication methods. One class of fabrication methods involve the synthesis of DNA probes as the first step, and the attachment of probes on the solid surface as the second step. The cDNA microarrays pioneered by Pat Brown's group [57] are produced using polymerase chain reaction (PCR) in the first step and a robot-controlled printer in the second step. Some other related technologies use ink-jet like printers to spray chemically synthesized oligonucleotide probes on the microarrays. One example is the Agilent custom microarrays. Another class of

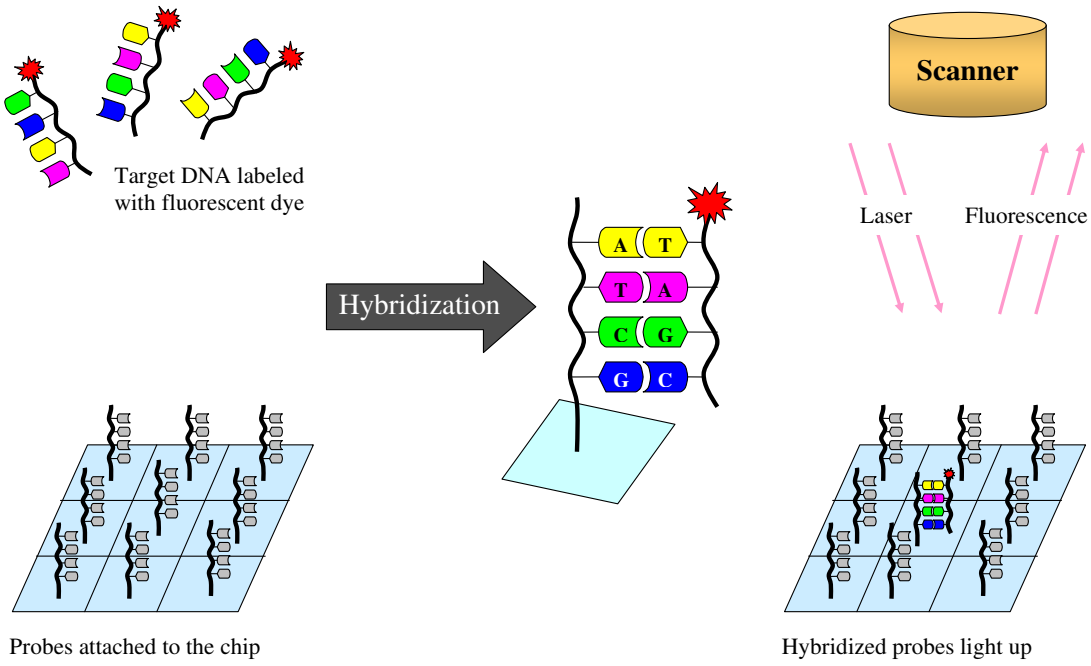

Figure 1.1: Principle of DNA microarrays

fabrication methods involve the synthesis of DNA oligonucleotides directly on the microarray using photo-activated chemistry. In each cycle, sites that will have the next base added are deprotected by UV light, and coupling of the next nucleotide (A, T, C or G) follows immediately. Affymetrix uses a physical photolithographic mask to ensure that only specific sites are activated by UV light in each cycle. Roche NimbleGen builds its arrays using the Maskless Array Synthesizer technology. A Digital Micromirror Device creates "virtual masks" that reflect the desired pattern of UV light to be projected on the microarray slide. They ensure that UV activation occurs at the precise locations where the next nucleotide is to be coupled.

Aside from the fabrication methods, DNA microarrays can be categorized based on the number of channels in the fluorescence detection system. An Affymetrix microarray is almost always hybridized with a single sample labeled by one fluorescent dye. A NimbleGen microarray is often hybridized with two samples simultaneously, also known as competitive hybridization. The two samples are labeled separately with two different dyes. Often, one of the two samples contains the treatment of interest and the other one is the control sample. The log ratios of fluorescent intensities between treatment and control represent the relative abundances of the RNA or DNA sequences in the samples. Agilent and cDNA microarrays are also platforms for two-color assays. In the literature, "single-channel" refers to the one-color assay system and "two-channel" refers to the two-color assay system.

DNA microarrays have wide applications in functional genomics. These include gene expression profiling, comparative genomic hybridization, chromatin immunoprecipitation on chip (ChIP-chip), SNP detection, alternative splicing detection and fusion genes detection etc. Regardless of the application type, common statistical issues in DNA microarray data analysis include image analysis, accounting for the effects of background noise, normalization of the intensity ratios, and quality controls. These topics have been reviewed by Smyth et al. [59] and Bolstad et al. [9, 7]. This thesis is concerned with the statistical problems in two types of DNA microarray applications. The first type is gene expression profiling in time course experiments. The second type is the determination of protein binding sites through ChIP-chip experiments. Data analysis issues unique to these two types of DNA microarray applications are discussed in the following two sections.

1.2 Gene expression time course analysis

Gene expression refers to the process by which the genetic code is used to synthesize a functional gene product, which is often a protein. A copy of the genomic DNA is made in the form of messenger RNA (mRNA) in a process called transcription. The mRNA transcript directs the formation of polypeptides (proteins) in a process called translation. There is often a positive correlation between the amount of mRNA transcribed in the first step and the amount of proteins produced in the final step. Transcriptional regulation refers to the molecular mechanisms that control the number of copies of mRNA made for a particular gene, or a set of genes. Whenever a biological system, i.e. a cell or an organism, experiences a change in the environment, it responds with transcriptional regulation that results in changes to the expression levels of certain genes. Thus the gene expression profile of a biological system is

dynamic. DNA microarrays are powerful tools for gene expression profiling, because mRNA levels of thousands of genes can be measured simultaneously. To study transcriptional regulation, biologists often collect mRNA samples at various time points following the induction of a stimulus. Gene expression profiles are then constructed for the different time points using DNA microarrays. We refer to this type of studies as microarray time course experiments. Data analysis methods for microarray time courses have been reviewed extensively elsewhere. [1, 66].

A time course can be either "periodic" or "developmental" depending on the temporal pattern. Periodic time courses are characterized by temporal profiles that follow regular patterns, such as cell cycles [61] and circadian rhythms. Developmental time courses are characterized by aperiodic processes, such as natural growth, response to a treatment, or response to an infection [45]. The temporal profiles in developmental time courses tend to have arbitrary patterns. Time course experiments may be categorized as either "longitudinal" or "cross-sectional" depending on the sampling scheme. In longitudinal experiments, samples are collected from the same experimental unit repeatedly. For example, blood samples may be drawn from the same individual at different time points during the course of an infection. In cross-sectional experiments, each sample is collected from a different experimental unit. When a time course experiment is performed using a batch of cell cultures, the experimental unit may be a plate of cells. At each time point, one plate is removed from the batch for RNA extraction. Thus each sample comes from a different experimental unit. We are particularly interested in the analysis of developmental microarray time courses with longitudinal replications. This type of experiments often have small numbers of time points and small numbers of replicates.

Gene expression profiling studies often aim at identifying sets of genes that are differentially expressed under one or more conditions. In time course experiments, "differential expression" has special meanings. If the experiment is carried out under one condition, the aim is to determine which genes have non-constant expression levels across the time points. This situation is referred as the one-sample problem. If two or more conditions are involved, "differential expression" means the temporal profiles of the given gene are different under the different conditions. This situation is referred as the multi-sample problem. For example, a gene that exhibits increasing levels of expression under the wild type condition may be expressed at a constant level under the mutant condition. This gene would be considered differentially expressed, because its temporal profiles are different under the two conditions. A common approach for analyzing short time courses is to treat time as a factor in F-tests. This approach ignores any autocorrelations that may exist in the longitudinal samples. Another approach is to consider the temporal profile of each gene as a multivariate vector. However, estimation of the covariance matrix is difficult due to the small number of replicates. Tai and Speed developed multivariate empirical Bayes (*MB*-) statistics for ranking genes based on "differential expression," in both the one-sample and the multi-sample problems [67, 68]. The multivariate Gaussian model is used to capture the correlations between gene expression measurements at the different time points. The empirical Bayes approach addresses the small-sample challenge with moderation of the covariance matrix.

Biologists are often unsatisfied with ranking alone, and would like to make significance

statements about differentially expressed genes. The selection of significant genes based on the *MB*-statistic is a multiple hypothesis testing problem. A comprehensive review of multiple testing procedures for microarray data can be found in a recently published book by Dudoit and van der Laan [23]. For each gene on the microarray, a null hypothesis is tested using the *MB*-statistic. Since the microarray contains thousands of genes, thousands of hypotheses are tested simultaneously. Any decision about gene selection may incur two types of errors. Type I errors refer to the rejection of true null hypotheses, i.e. false positives. Type II errors refer to the failure to reject false null hypotheses, i.e. false negatives. The "power" of the test is defined as one minus the false negative rate. The goal is to minimize the Type II error rate subject to a constraint on the Type I error rate. There are numerous definitions of Type I error rates. One group of Type I error rates are based on the distribution of the number of Type I errors. These include the family-wise error rate (FWER), the generalized family-wise error rate (gFWER), the per-comparison error rate (PCER), the per-family error rate (PFER), the median-based per-family error rate (mPFER), and the quantile number of false positives (QNFP). Another group of Type I error rates are based on the proportion of Type I errors among the rejected hypotheses. These include the false discovery rate (FDR), the proportion of expected false positives (PEFP), the quantile proportion of false positives (QPFP). Because of the popularity of the false discovery rate among the biologists, we propose in this thesis an FDR-controlling procedure for multiple testing using the *MB*-statistic.

1.3 ChIP-chip and tiling arrays

Traditional DNA microarrays designed for gene expression studies often contain only a few probes for each gene. A special type of DNA microarrays, called tiling arrays, contain probes that are placed densely along the chromosomal coordinates. They are generally designed to cover either the entire genome or contigs of the genome. Tiling array designs may differ in probe lengths and the spacing between adjacent probes. Affymetrix uses 25-mer probes, and offer 6 million features on each chip. Agilent and NimbleGen use longer oligonucleotides with fewer features on each chip. The genome may be tiled with either partially overlapping probes, or non-overlapping probes with small gaps in between the neighbors. The most common usage of tiling arrays is to determine the genomic locations of DNA binding sites for a particular protein, through chromatin immunoprecipitation on chip (ChIP-chip) experiments. This type of experiments often involve 5 steps. In the first step, DNA-binding proteins are cross-linked to the DNA that they are associated with in vivo, by applying formaldehyde to the cells. In the second step, the cells are lysed and the DNA is fragmented into pieces of roughly 500 base pairs by sonication. In the third step, the DNA-protein complex is precipitated out of the cell lysate using an antibody that specifically binds to the protein of interest. Serial washes are often performed to remove some non-specific materials pulled down by immunoprecipitation. In the fourth step, the cross-links in the purified DNA-protein complex are reversed to release the DNA fragments. In the fifth step, the purified DNA fragments are labeled with a fluorescent dye and hybridized to a tiling array. The input material obtained from Step 2 is often analyzed in parallel.

If single-channel arrays are used, then the immunoprecipitated (IP) sample and the input sample are hybridized to two different chips separately. If two-channel arrays are used, then the immunoprecipitated sample and the input sample are labeled with two different dyes and co-hybridized to the same chip. The fluorescence intensity log ratios of IP over input are used to quantify the enrichment of DNA fragments associated with the protein. The investigator may also perform some control experiments using non-specific immunoglobulin G (IgG) in lieu of the antibody specific to the protein of interest. The IgG control experiments help the investigator identify genomic fragments that are prone to non-specific binding in the immunoprecipitation step. Reviews of the ChIP-chip technology can be found in Buck and Lieb [12], Mockler and Ecker [47] and Bulyk [13].

A few different methods have been used for normalizing tiling array data. Model-based Analysis of Tiling arrays (MAT) was proposed by Johnson et al. [37] to remove the probe effects due to the sequence and the genome copy number of the 25-mer on Affymetrix arrays. Model-based Analysis for 2-Color arrays (MA2C) was proposed by Song et al. [60] to normalize the probe level log ratios based on the GC-content of the probe sequences. Quantile normalization was originally developed by Bolstad et al. [8] for normalizing probe level intensities across multiple gene expression microarrays. It has been accepted widely as the preferred normalization method for tiling array data [69, 42, 34].

A distinguishing characteristic of tiling arrays is the sequential order among the probes. Analysis methods developed for the traditional gene expression microarrays often assume independence among the genes. Although this is not entirely true, the independence assumption is often applicable because the probes are mapped to genomic locations that are far part from each other. Tiling arrays, on the other hand, contain probes that are mapped to nearby genomic locations in a sequential order, such that the independence assumption is completely violated. So the analysis of tiling array data requires a different perspective. Whereas gene expression profiling aims at finding differentially expressed genes, ChIP-chip analysis aims at detecting chromosomal fragments enriched in the immunoprecipitated sample, which are putative targets of the DNA-binding protein. When the probe level log ratios of fluorescence intensities are plotted against their chromosomal positions, a binding target has the characteristic shape of a triangular peak. The signals are the strongest at the center of the peak, and taper off near the ends. The peak position corresponds to the center of the putative binding event. Thus computational methods aim at detecting the peaks in ChIP-chip tiling array data.

The most common approach to peak detection scans the genome with a sliding window. This approach generally involves three steps. In the first step, the signal of each probe is standardized into a test statistic. In the second step, a smoothed score is computed for each probe position, using the test statistics of the probes that lie within the sliding window centered at this position. In the third step, neighboring windows with high scores are joined into peaks according to some criteria. One of the earliest ChIP-chip analysis methods, developed by Cawley et al. in 2004 [14], takes the sliding window approach. It applies the Wilcoxon rank-sum test to the probe intensity differences in each sliding window, to compute a p-value of enrichment of IP over control. Then, a threshold is applied to the p-values to identify the positions of interest, which are eventually joined into peaks. This method is implemented

in the Affymetrix Tiling Array Software (TAS). TileMap by Ji et al. [35] is another sliding window method designed for Affymetrix data. It standardizes the probe-level data using an empirical Bayes test statistic that resembles the *Student-t*. The smoothed score is a moving average of the t-like statistics. An updated version of TileMap was introduced recently as the internal peak caller in CisGenome, which is an integrated software system for analyzing ChIP-chip and ChIP-seq data [34]. The two versions of TileMap offer different options for FDR computation. The old version of TileMap estimates FDRs based on an unbalanced mixture subtraction (UMS) method, which compares the data distributions of the paired IP and input control experiments. The new version of TileMap estimates FDRs based on the left tail in the histogram of the moving average statistics, also known as the symmetric null method. TiMAT, developed by Bourgon and Speed [42], introduced the symmetric null method for estimating the FDR of window score cutoffs. This method assumes that the background window score distribution is symmetric about its mean. Thus values less than the observed mode of the genome-wide window scores are used to estimate the null distribution. Each window score is assigned a p-value according to the symmetric null distribution. Subsequently, the p-values are converted into FDRs using the method due to Storey [62]. MAT by Johnson et al. [37] and MA2C by Song et al. [60] apply the sliding window approach to peak detection, after model-based standardization of the probe level data. MAT uses the trimmed mean of all the probe t values in the window as the smoothed score. MA2C offers a few options for computing the smoothed score: median, pseudo-median, median polish, or trimmed mean of the probes in the window. The default option is the median of the probe level normalized log ratios in the window. Tilescope by Zhang et al. [69] also takes the sliding window approach to peak detection. The smoothed window score used there is either a p-value calculated by the Wilcoxon signed-rank test, or the pseudo-median (the one-sample Hodges-Lehmann estimator) of the log ratios within the window. ChIPOTle by Buck et al. [11] and ACME by Scacheri et al. [56] are two other sliding window peak callers developed for NimbleGen tiling arrays. ChIPOTle computes a moving average of the log ratios within each window, and assigns a p-value using either a Gaussian or a permutation-based null distribution. ACME uses χ^2 analysis to determine if any given window contains a higher than expected number of probes with log ratios above the user-defined threshold.

Hidden Markov models have been proposed previously for the analysis of tiling array data. TileMap provides the option of computing smoothed scores of the t-like statistics based on an HMM [35]. Each probe has a hidden hybridization state: 1 if the probe is IP-enriched; 0 otherwise. The two states have stationary probabilities π_0 and $\pi_1 = 1 - \pi_0$. For each pair of adjacent probes, the transition probabilities between the hidden states depend on the distance between the center positions of the probes. If the distance is greater than the parameter d_0, then the Markov chain is reset and the stationary probabilities are used as the transition probabilities. If the distance is less than or equal to the parameter d_0, then the transition matrix is used. Notice that probe spacings of all lengths less than or equal to the parameter d_0 are treated in the same way. This implies the assumption that the transition probabilities are constant for steps of all sizes, which is only a crude approximation. The state-conditional emission distributions are estimated using the UMS method, which aims at recovering different components of a mixture distribution. The parameter d_0 is

estimated based on the average length of the IP fragments. The transition probabilities are estimated based on the typical length of hybridizations in base pairs. The smoothed score is the posterior probability of the IP-enriched state, obtained using the forward-backward algorithm. Li et al. [41] also proposed a two-state hidden Markov model for analyzing ChIP-chip data. This method has two main distinctions from TileMap: 1) the emission distributions are assumed to be Gaussian; 2) the log odds of the IP-enriched state against the non-enriched state is used as the smoothed score, instead of the posterior probability of the IP-enriched state. Du et al. [20] proposed to incorporate external knowledge, such as gene annotation or experimental validations, into the analysis of ChIP-chip experiments using a hidden Markov model.

Besides the sliding windows and the HMMs, a few other approaches have been used in the analysis of ChIP-chip data. Keles [38], Gottardo et al. [31] and Sun et al. [64] proposed mixture models for the ChIP-chip data generation process. Zheng et al. [70] proposed to model DNA fragmentation by the Poisson point process, and concluded that the peak shape of single binding events should be triangular on the log transformed scale. They developed a peak detection method called MPeak that incorporates the shape information by fitting a double regression model. Qi et al. [52] and Reiss et al. [55] proposed deconvolution methods for resolving multiple nearby binding events. Whereas Qi et al. use a Bayesian graphical model, Reiss et al. use Kernel regression. Other methods that are frequently used in the scientific community include TAMALPAIS by Bieda et al. [6] and the permutation method by Lucas et al. [43] that is implemented in NimbleScan.

More than half of the previously developed methods can analyze only one ChIP-chip experiment at a time. Although some methods can incorporate replicate data, they all require the same design of tiling arrays being used in all experiments. However, when a large scale genomic study is carried over a long period of time (i.e. 3+ years), the investigators may be forced to use tiling arrays with different probe designs, due to practical considerations. Thus there is a compelling need to address the unmet challenge of integrating replicate data with different tiling array designs. Another challenge that has not been addressed satisfactorily in the literature is the identification of binding sites shared by two or more DNA-binding proteins. This is a consequence of the fact all of the existing methods were developed for analyzing one protein at a time. The current practice is to cross-list the peaks called for the individual proteins. Because it requires making judgmental decisions at various steps, there is no standard protocol for cross-listing the peaks of individual proteins. Clearly, the field needs a method that enables the joint analysis of multiple proteins. In this thesis, we propose a nonhomogeneous hidden Markov Model to address these two challenges: 1) the integration of ChIP-chip data from different tiling array designs, 2) the joint analysis of two or more DNA-binding proteins.

1.4 Outline of the thesis

Chapter 2 describes an FDR-controlling procedure for analyzing replicated microarray time course data with the multivariate empirical Bayes statistic. Following a brief introduction on the background of the study, the multivariate empirical Bayes model for microarray

time course data is reviewed. We derive the distribution of the moderated Hotelling T^2 statistic, and establish its relationship to the F-distribution. Two FDR-controlling procedures are proposed under either the frequentist framework or the empirical Bayes framework. The two FDR procedures are compared under various conditions through simulations. We then present an application of the FDR procedures to a real data set. Both the simulation results and the analysis of real data suggest that the empirical Bayes FDR-controlling procedure is more robust to the variability of hyperparameter estimation, and is a more powerful method when the sample size is small.

Chapter 3 describes a nonhomogeneous Hidden Markov Model for integrating ChIP-chip data from multiple tiling array designs. After a brief introduction on the scientific context that motivated our study, we present a nonhomogeneous hidden Markov model for analyzing one protein at a time. This model integrates different tiling array designs at the single base pair level, thus preserves the highest possible resolution in the merged data set. We derive a modified Baum-Welch algorithm for fitting this nonhomogeneous HMM. We also present some simulation results confirming that the algorithm is well-behaved under the settings relevant to our study. We then describe a procedure that converts candidate probes into peaks that represent putative binding sites. Finally, we present some ROC analyses comparing our method to the existing methods, using a set of positive and negative regions pre-defined by visual inspection. When applied to single experiments, our method performs similarly as the best existing methods for analyzing NimbleGen tiling array data. When applied to the combined data set, which consists of replicate experiments performed on different tiling array designs, our method shows a drastic improvement in performance.

Chapter 4 describes a peak-detection method for the joint analysis of ChIP-chip data from multiple DNA binding proteins. This method is based on a generalization of the nonhomogeneous Hidden Markov Model presented in Chapter 3. One assumption of the multi-protein model is that each protein emits observations independently of the other proteins, conditional on the hidden states. We present some simulation results verifying the applicability of this assumption. We then describe a procedure for fitting the nonhomogeneous HMM for multiple proteins. Simulations of the two-protein model confirm that the algorithm performs well in identifying the joint binding sites of different proteins. Another ROC analysis shows that our joint analysis method out-performs the current practice of cross-listing the individual peak calls. Finally, we discuss some practical issues in the application of our method to real data, and present some solutions.

Chapter 2

An FDR-controlling procedure for analyzing replicated microarray time course data with the multivariate empirical Bayes statistic

2.1 Motivation

Microarray time course experiments provide a powerful tool for monitoring the dynamic expression levels of virtually the entire genome simultaneously. A time course can be either "periodic" or "developmental" depending on the temporal pattern. Periodic time courses are characterized by temporal profiles that follow regular patterns, such as cell cycles [61] and circadian rhythms. Developmental time courses are characterized by aperiodic processes, such as natural growth, response to a treatment, or response to an infection [45]. The temporal profiles in developmental time courses tend to have arbitrary patterns. The replication of time course experiments can be categorized as either "longitudinal" or "cross-sectional" depending on the sampling scheme. In longitudinal experiments, the mRNA samples collected at different time points are extracted from the same biological unit, such that the expression measurements are correlated. In cross-sectional experiments, the mRNA samples are extracted from different units at different time points. This chapter is concerned with the analysis of developmental microarray time courses with longitudinal replications. Two major sources of challenges are the small number of time points (4-10) and the few number of replicates (6 or less).

In 2006, Tai and Speed [67] proposed a multivariate empirical Bayes model for developmental, longitudinal replicated microarray time course data. To rank genes in the order of interest, they derived the one-sample and two-sample *MB*-statistics, which reduce to the moderated Hotelling T^2 statistics when the sample size is the same for all genes. They later extended this model to the multi-sample situation [68]. The one-, two- or multi- samples refer to the number of biological conditions being investigated. The null hypothesis for the one-sample problem states that the expected temporal profile is constant. The null hypothesis

for the multi-sample problem states that the expected temporal profiles under the different conditions are the same. Biologists are often unsatisfied with ranking alone. They would like to select for a set of differentially expressed genes, while keeping the false discovery rate (FDR) within a pre-specified level. We propose an empirical Bayes FDR-controlling procedure for hypothesis testing with the *MB*-statistics. Since in most microarray studies, the same layout of features applies to all the chips used in an experiment, the sample size is usually the same for all genes. Hence, we devote our attention to the moderated Hotelling T^2 statistics. Moreover, we use the one-sample problem to illustrate the fundamental principles, which are readily generalizable to the multi-sample problem.

The organization of this chapter is as follows. We begin with a brief review of the multivariate empirical Bayes model for replicated time course data. We then introduce two procedures for controlling the false discovery rate, one being a frequentist approach and the other being an empirical Bayes approach. Following this, we present some simulation results showing the effects of sample size and moderation on the two FDR procedures. After comparing the performances of the two FDR procedures in different scenarios, we present an application of the FDR procedures to a real data set. Finally, we conclude by summarizing our findings and making recommendations.

2.2 The multivariate empirical Bayes statistic

We adopt the notation used in Tai and Speed (2006). Suppose there are n independent, replicated time series measurements of mRNA expression levels for each gene. Let k denote the number of time points at which the measurements are taken, and let g index the genes. The replicated time series, $\mathbf{X}_{g1}, \ldots, \mathbf{X}_{gn}$, are modeled as an *i.i.d.* samples of size n drawn from the k-variate Gaussian distribution. The gene-specific mean and covariance parameters are μ_g and $\mathbf{\Sigma}_g$. The objective is to test the null hypothesis H_g that $\mu_g = \mu_0$ against the alternative hypothesis K_g that $\mu_g \neq \mu_0$. When the microarray data are normalized log-ratios from two-color arrays, or when they are differences of paired experiments, we can assume that $\mu_0 = \mathbf{0}$. Since the same model is applied to every gene on the microarray, we will drop the subscript g in expressions representing an arbitrary single gene. Let I denote a Bernoulli random variable with success probability p, $0 < p < 1$, that indicates the status of the gene.

$$I = \begin{cases} 1 & \text{if } K \text{ is true;} \\ 0 & \text{if } H \text{ is true.} \end{cases}$$

The multivariate hierarchical Bayesian model states the following. The gene-specific covariance matrix $\mathbf{\Sigma}$ is generated from an inverse-Wishart distribution, with ν degrees of freedom and scale matrix $\nu\mathbf{\Lambda}$:

$$\mathbf{\Sigma} \sim \textit{Inv-Wishart}_{\nu}((\nu\mathbf{\Lambda})^{-1}),$$

where $\nu > 0$ and $\nu\mathbf{\Lambda} > 0$. The prior mean of $\mathbf{\Sigma}$ is $\mathbf{\Lambda}$ for all genes. Given $\mathbf{\Sigma}$, the multivariate normal priors for the gene-specific mean parameter μ are generated as follows:

$$\begin{cases} \mu|\mathbf{\Sigma} & \sim N(\mathbf{0}, \eta^{-1}\mathbf{\Sigma}) & \text{if } I = 1, \\ \mu|\mathbf{\Sigma} & \equiv \mu_0 & \text{if } I = 0, \end{cases}$$

where $\eta > 0$ is the scale parameter. For the one-sample or paired two-sample problem, $\mu_0 = 0$. The true status I of the gene is unobserved. Given μ and $\mathbf{\Sigma}$, the observed data $(\mathbf{X}_1, \ldots, \mathbf{X}_n)$ come from the multivariate normal distribution $N(\mu, \mathbf{\Sigma})$, with the mean parameter μ dependent on I.

This Bayesian model leads to a moderated estimate of the covariance matrix that is shrunken towards the prior mean. The amount of shrinkage is determined by the parameter ν. Tai and Speed [67] derived the moderated Hotelling T^2 statistic, denoted by $\widetilde{T}^2$, from a likelihood ratio test. A large value of $\widetilde{T}^2$ provides evidence against the null hypothesis. Let $\mathbf{S}$ denote the sample covariance matrix, and let $\widetilde{\mathbf{S}}$ denote the inverse of the posterior mean of the precision matrix given $\mathbf{S}$, namely

$$\mathbf{S} = \frac{1}{n-1}\sum_{i=1}^{n}(\mathbf{X}_i - \bar{\mathbf{X}})(\mathbf{X}_i - \bar{\mathbf{X}})',$$
$$\widetilde{\mathbf{S}} = [E(\mathbf{\Sigma}^{-1}|\mathbf{S})]^{-1} = \frac{(n-1)\mathbf{S} + \nu\mathbf{\Lambda}}{n+\nu-1}.$$

The one-sample moderated Hotelling T^2 statistic is:

$$\begin{aligned}\widetilde{T}^2 &= n(\bar{\mathbf{X}} - \mu_0)'\widetilde{\mathbf{S}}^{-1}(\bar{\mathbf{X}} - \mu_0),\\ &= n\bar{\mathbf{X}}'\widetilde{\mathbf{S}}^{-1}\bar{\mathbf{X}} \quad \text{when } \boldsymbol{\mu_0} = \mathbf{0}.\end{aligned}$$

It can be shown that the moderated Hotelling T^2 statistic is equivalent to the conventional Hotelling T^2 statistic with augmented degrees of freedom (see Section 2.2.1). Under the null hypothesis, $\widetilde{T}^2$ is proportional to an F-statistic:

$$\frac{(n+\nu-k)}{k(n+\nu-1)}\widetilde{T}^2 \sim \mathcal{F}(k, n+\nu-k),$$

where $\mathcal{F}(a, b)$ denotes the F-distribution with a and b degrees of freedom.

2.2.1 Distribution of the moderated Hotelling T^2 statistic

Claim

In the one-sample (or paired two-sample) problem, the moderated Hotelling T^2 statistic is distributed as the conventional Hotelling T^2 statistic with $(n+\nu-k-1)$ degrees of freedom.

Proof

Let $\mathbf{S}$ denote the sample covariance matrix, and let $\tilde{\mathbf{t}}$ denote the moderated multivariate t-statistic:

$$\tilde{\mathbf{t}} = n^{\frac{1}{2}}\tilde{\mathbf{S}}^{-\frac{1}{2}}\bar{\mathbf{X}}.$$

Then $\widetilde{T}^2$ can be re-written as:

$$\widetilde{T}^2 = \tilde{\mathbf{t}}'\tilde{\mathbf{t}}.$$

Following the derivation in Tai & Speed [67], we obtained

$$P(\tilde{\mathbf{t}}|\mathbf{S}, I=0) = \pi^{-\frac{1}{2}k}(n+\nu-1)^{-\frac{1}{2}k}\frac{\Gamma[\frac{1}{2}(n+\nu)]}{\Gamma[\frac{1}{2}(n+\nu-k)]}\left[1+\frac{\tilde{\mathbf{t}}'\tilde{\mathbf{t}}}{(n+\nu-1)}\right]^{-\frac{1}{2}(n+\nu)}.$$

This expression can be rewritten as:

$$\begin{aligned}P(\tilde{\mathbf{t}}|I=0) =& \frac{\Gamma[\frac{1}{2}(n+\nu)]}{(n+\nu-k)^{\frac{1}{2}k}\pi^{\frac{1}{2}k}\Gamma[\frac{1}{2}(n+\nu-k)]}\left(\frac{n+\nu-1}{n+\nu-k}\right)^{-\frac{1}{2}k} \\ & \left[1+\frac{1}{(n+\nu-k)}\left(\frac{n+\nu-1}{n+\nu-k}\right)^{-1}\tilde{\mathbf{t}}'\tilde{\mathbf{t}}\right]^{-\frac{1}{2}(n+\nu)}.\end{aligned}$$

According to the definition in [32], $\tilde{\mathbf{t}}$ is distributed as multivariate t with parameters:

$$\begin{aligned}d.f. &= (n+\nu-k), \\ \Sigma &= \left(\frac{n+\nu-1}{n+\nu-k}\right)\mathbf{I}_k.\end{aligned}$$

Thus $\tilde{\mathbf{t}}$ can be constructed as the following:

$$\tilde{\mathbf{t}} = \left(\frac{x}{n+\nu-k}\right)^{-\frac{1}{2}}\mathbf{y},$$

where x is distributed as *Chi*-squared with $(n+\nu-k)$ degrees of freedom; $\mathbf{y}$ is distributed as k-dimensional multivariate normal with mean zero and covariance $\Sigma = (\frac{n+\nu-1}{n+\nu-k})\mathbf{I}_k$.

The moderated Hotelling T^2 statistic can be re-written as:

$$\begin{aligned}\widetilde{T}^2 = \tilde{\mathbf{t}}'\tilde{\mathbf{t}} &= (n+\nu-k)x^{-1}\mathbf{y}'\mathbf{y}, \\ \frac{(n+\nu-k)}{(n+\nu-1)}\widetilde{T}^2 &= (n+\nu-k)x^{-1}\mathbf{y}'\Sigma^{-1}\mathbf{y}, \\ \frac{(n+\nu-k)}{k(n+\nu-1)}\widetilde{T}^2 &= \frac{(n+\nu-k)}{k}\frac{z}{x},\end{aligned}$$

where $z = \mathbf{y}'\Sigma^{-1}\mathbf{y}$ is distributed as *Chi*-squared with k degrees of freedom [48]. Standard calculations lead to the following conclusion.

$$\begin{aligned}\frac{(n+\nu-k)}{k(n+\nu-1)}\widetilde{T}^2 &\sim \mathcal{F}(k, n+\nu-k), \\ \widetilde{T}^2 &\sim \mathcal{T}^2(k, n+\nu-1).\end{aligned}$$

2.3 Frequentist FDR-controlling procedure

The false discovery rate (FDR) was introduced by Benjamini and Hochberg in 1995 [3]. In comparison to FWER-controlling methods, FDR-controlling methods are more powerful. The advantage of FDR increases with the number of non-null hypotheses and the total number of tests. Because many microarray studies are fishing expeditions, a conservative control of Type I errors is often less important than power. Thus the false discovery rate is a very popular Type I error rate among the biologists in the microarray field. For this reason, we aim at developing an FDR-controlling procedure for hypothesis testing using the moderated Hotelling T^2 statistic.

FDR is defined as the expected value of the proportion of false positives incurred by applying a given rejection rule. Let $\mathcal{R}$ denote the rejection rule of choice.

$$FDR(\mathcal{R}) \triangleq E\left[\frac{\text{number true nulls rejected by } \mathcal{R}}{\text{total number of rejected hypotheses}}\right]$$

The FDR-controlling procedure originally proposed by Benjamini and Hochberg is reviewed briefly. Let H_g denote the null hypothesis associated with gene g, for $g = 1, ..., N$. Let P_g denote the nominal p-value of gene g, obtained according to the null distribution of the test statistic. Let $P_{(1)} \leq P_{(2)} \leq \ldots \leq P_{(N)}$ be the ordered p-values. For an arbitrary level $\alpha \in (0, 1)$, find the largest index i_α that satisfies the following.

$$i_\alpha = arg\max_i \left\{P_{(i)} \leq \frac{i}{N}\alpha\right\}$$

For independent test statistics, the following rejection rule controls the FDR at level α.

$$\mathcal{R}_\alpha = \{\text{reject all } H_g \text{ with } P_g \leq P_{(i_\alpha)}\}$$

Smyth [58] proposed an FDR-controlling procedure for the one-dimensional moderated t-statistics as follows. First, convert the test statistics into nominal p-values according to the theoretical null distribution. Then, apply the FDR-controlling procedure described above to adjust for genome-wide multiple testing. This approach can be emulated for the $\tilde{T}^2$ statistics. We refer to it as the frequentist FDR-controlling procedure. Please note that specification of the null distribution requires estimates of the hyperparameters. Thus the frequentist FDR procedure may be sensitive to hyperparameter estimation errors.

2.4 Empirical Bayes FDR-controlling procedure

An empirical Bayes FDR procedure proposed by Efron et al. [26] has the flexibility that a theoretical null distribution is not required. Instead, the null distribution is estimated empirically through either permutations or resampling. We briefly review the empirical Bayes FDR procedure below.

In the Bayesian framework, the false discovery rate is defined as the posterior probability that a hypothesis is truly null, given its observed test statistic exceeds the critical value. Let

p_0 denote the prior proportion of null hypotheses, and let $F_0(y)$ denote the null distribution of the test statistics evaluated at the critical value y. The Bayesian definition is written as the following,

$$Fdr(y) \triangleq P(I_g = 0 | Y_g \geq y) = p_0 \frac{1 - F_0(y)}{1 - F(y)}.$$

Let $\widehat{F}(y) = |\{Y_g \leq y\}|/n$ denote the ordinary empirical cdf of the test statistic evaluated at y. In our case, the test statistic is the moderated Hotelling T^2 statistic, $Y_g = \widetilde{T}_g^2$. The choice of the empirical null distribution $\widehat{F}_0(y)$ is crucial to the success of this procedure. The empirical Bayes estimate of FDR is the proportion of test statistics exceeding the critical value in the empirical null distribution, divided by that in the overall empirical distribution.

$$\widehat{Fdr}(y) \approx \tilde{p}_0 \frac{\{\text{proportion of } Y_g \geq y \text{ in } \widehat{F}_0\}}{\{\text{proportion of } Y_g \geq y \text{ in } \widehat{F}\}},$$

where $\tilde{p}_0$ is an upper-bound for the prior probability of H.

The rejection rule is chosen according to the minimum test statistic, for which the FDR is controlled at the level α. In microarray studies, gene g is considered non-null if $Y_g \geq y_\alpha$ and

$$y_\alpha = \min_y \left\{ \widehat{Fdr}(y) \leq \alpha \right\}.$$

In order to compute the empirical Bayes false discovery rates, we need to estimate the null distribution of the moderated Hotelling T^2 statistics. Because longitudinal time course data consist of dependent samples, permutation-based methods are inappropriate for this purpose. Dudoit, van der Laan and Pollard proposed the null shift and scale-transformed test statistics null distribution, which can be estimated using the non-parametric bootstrap [21, 22, 51]. This choice of the null distribution provides asymptotic control of the FDR when the step-up procedure due to Benjamini and Hochberg is applied. However, a challenge in our application is that longitudinal time course data typically contain very few replicates. In order to achieve the asymptotic results, we need a sufficient pool of replicated time series from which many bootstrap resamples can be drawn. When the original data set contains only a few replicates, the number of unique bootstrap resamples that could be drawn is very limited. Therefore, we decided to estimate the null distribution using the parametric bootstrap. By taking this approach, we assume that the genes on the microarray generate observations independently of each other. Although it is known that genes may function in clusters, the independence assumption is applicable for the majority of cases.

The first step of the parametric bootstrap procedure is to estimate the Gaussian parameters from the real data. Under the null hypothesis, the gene-specific mean μ_g is the vector of zeros for all genes. The gene-specific covariance matrix is estimated as:

$$\mathbf{S}_g = \frac{1}{n} \sum_{i=1}^{n} \mathbf{X}_{gi} \mathbf{X}'_{gi}.$$

The moderated sample covariance matrix is computed by plugging-in the estimated hyper-parameters:

$$\widetilde{\mathbf{S}}_g = [E(\mathbf{\Sigma}_g^{-1}|\mathbf{S}_g)]^{-1} = \frac{(n-1)\mathbf{S}_g + \hat{\nu}\widehat{\mathbf{\Lambda}}}{n-1+\hat{\nu}}.$$

Because the Bayes estimator for $\mathbf{\Sigma}_g$ is more efficient than the conventional sample covariance, $\widetilde{\mathbf{S}}_g$ is preferable over $\mathbf{S}_g$ in the microarray setting with small sample sizes.

In the second step, bootstrap samples of gene expression data are generated by simulating from the Gaussian distribution specified by the estimated parameters. The b^{th} bootstrap sample of expression data for gene g is simulated from:

$$\mathbf{X}_g^b \sim N(\mathbf{0}, \widetilde{\mathbf{S}}_g) \text{ for } b = 1, \ldots, B.$$

Finally, a vector of moderated Hotelling T^2 statistics are computed for each bootstrap sample using the *timecourse* package [67], available from the Bioconductor. The estimated null distribution is constructed by pooling all the bootstrap test statistics. We typically use $B = 20$ iterations to obtain a bootstrap estimate of the test statistics null distribution. The large number N of genes ensures that the bootstrap distribution converges quickly with small B. In the simulation experiments to be described later, we used $N = 10000$ and did not observe any noticeable changes with larger values of B.

2.5 Comparisons of the FDR procedures through simulations

In the frequentist framework, the null distribution is specified by the number of time points k, the sample size n, and the prior degrees of freedom ν. In the empirical Bayes framework, the bootstrap null distribution is also influenced by n and ν. This is because n affects the estimation of the sample covariance matrix, and ν affects how the moderated sample covariance matrix is weighted towards $\mathbf{\Lambda}$. Thus the two FDR procedures are both influenced by the parameters n and ν. We performed simulation experiments to investigate the effects of varying these parameters, while holding the other parameters fixed.

Microarray data were simulated according to the multivariate hierarchical Bayesian model described in Section 2.2. The parameters that were fixed in each simulation include $k = 6, \eta = 0.25, \mu_0 = 0$, and $\mathbf{\Lambda}$ given below. The matrix $\mathbf{\Lambda}$ was obtained by making some modifications to the matrix used by Tai & Speed (2006) [67] in their simulation study.

$$\mathbf{\Lambda} = \begin{bmatrix} 0.1500 & 0.0250 & 0.005 & 0.003 & 0.0015 & 0.0005 \\ 0.0250 & 0.1400 & 0.025 & 0.005 & 0.0030 & 0.0015 \\ 0.0050 & 0.0250 & 0.085 & 0.025 & 0.0050 & 0.0030 \\ 0.0030 & 0.0050 & 0.025 & 0.100 & 0.0250 & 0.0050 \\ 0.0015 & 0.0030 & 0.005 & 0.025 & 0.1150 & 0.0250 \\ 0.0005 & 0.0015 & 0.003 & 0.005 & 0.0250 & 0.0950 \end{bmatrix}.$$

To investigate the effects of sample size, we fixed the value of ν at 12 and varied the value of n. To investigate the effects of the prior degrees of freedom, we fixed the value of n at 6, and

varied the value of ν. For each chosen set of parameters, we simulated two data sets with different values of p, the proportion of non-null genes. The total number of genes was fixed at 10,000 for both data sets. In the first data set, p was set to zero; in the second data set, p was set to 10%. These two data sets will be referred as the "observed" data sets in the succeeding discussion about the simulations. The moderated Hotelling T^2 statistics were computed for each data set. We also recorded the estimates of the prior degrees of freedom, obtained by running the hyperparameter estimation program provided in the *timecourse* package. We then applied each of the two FDR procedures to estimate the false discovery rates incurred at various cutoffs of the moderated Hotelling T^2 statistics.

2.5.1 Effects of sample size

Figure 2.1 compares the F-transformed distributions of the following: 1) the observed $\widetilde{T}^2$ statistics with the hyperparameters estimated from the data, 2) the observed $\widetilde{T}^2$ statistics with the true hyperparameters plugged-in, 3) the bootstrap null $\widetilde{T}^2$ statistics with the hyperparameters estimated from the data, 4) the bootstrap null $\widetilde{T}^2$ statistics with the true hyperparameters plugged-in, 5) the F-distribution according to the true value of ν. Keep in mind that each bootstrap distribution was obtained by pooling 20 bootstrap samples, with each sample having the same size as the observed data set. Notice that in the case of 100% null genes, the distribution of the observed $\widetilde{T}^2$ statistics, computed with the true hyperparameters, coincided with the theoretical null distribution, as expected. Figure 2.2 displays the quantile-quantile plots of the observed distribution against either the bootstrap null distribution or the theoretical null distribution. The top two panels represent the situation with 100% null genes. The bottom two panels represent the situation with 90% null genes.

The discrepancy between the bootstrap null distribution and the theoretical null distribution is worth mentioning. This was primarily due to the small sample size (i.e. the number of replicated time series). We found that the minimum sample size n required for either of the two FDR procedures to work properly is the dimension k of the multivariate Gaussian vectors. The examples shown in Figures 2.1 & 2.2 represent the small sample size scenario, with $n = k = 6$. The bootstrap null distribution biased toward the low-range values. As the sample size increased, the bias was reduced. A sample size of $n = 5k$ was sufficient for the bootstrap null distribution to converge to its theoretical counterpart, as shown in Figure 2.3.

To make the two FDR procedures comparable [25], we took the most conservative choice of the prior probability, $\widetilde{p}_0 = 1$. Figure 2.4 compares the results of the two FDR procedures, when applied to the data set simulated with 90% null genes. The horizontal axis of each plot represents critical values of $\widetilde{T}^2$ after scaling to the F-distribution by a constant. The vertical axis represents the false discovery rates incurred by applying various critical values of the test statistic. Figure 2.4(a) illustrates the situation with a small sample size, $n = k$. Figure 2.4(b) illustrates the situation with a large sample size, $n = 5k$. The true FDR is plotted as a function of the critical values, in dashed black lines. The frequentist procedure (blue) was excessively conservative, because the hyperparameter ν was under-estimated, leading to a misspecified theoretical null distribution. The empirical Bayes procedure (red) was slightly

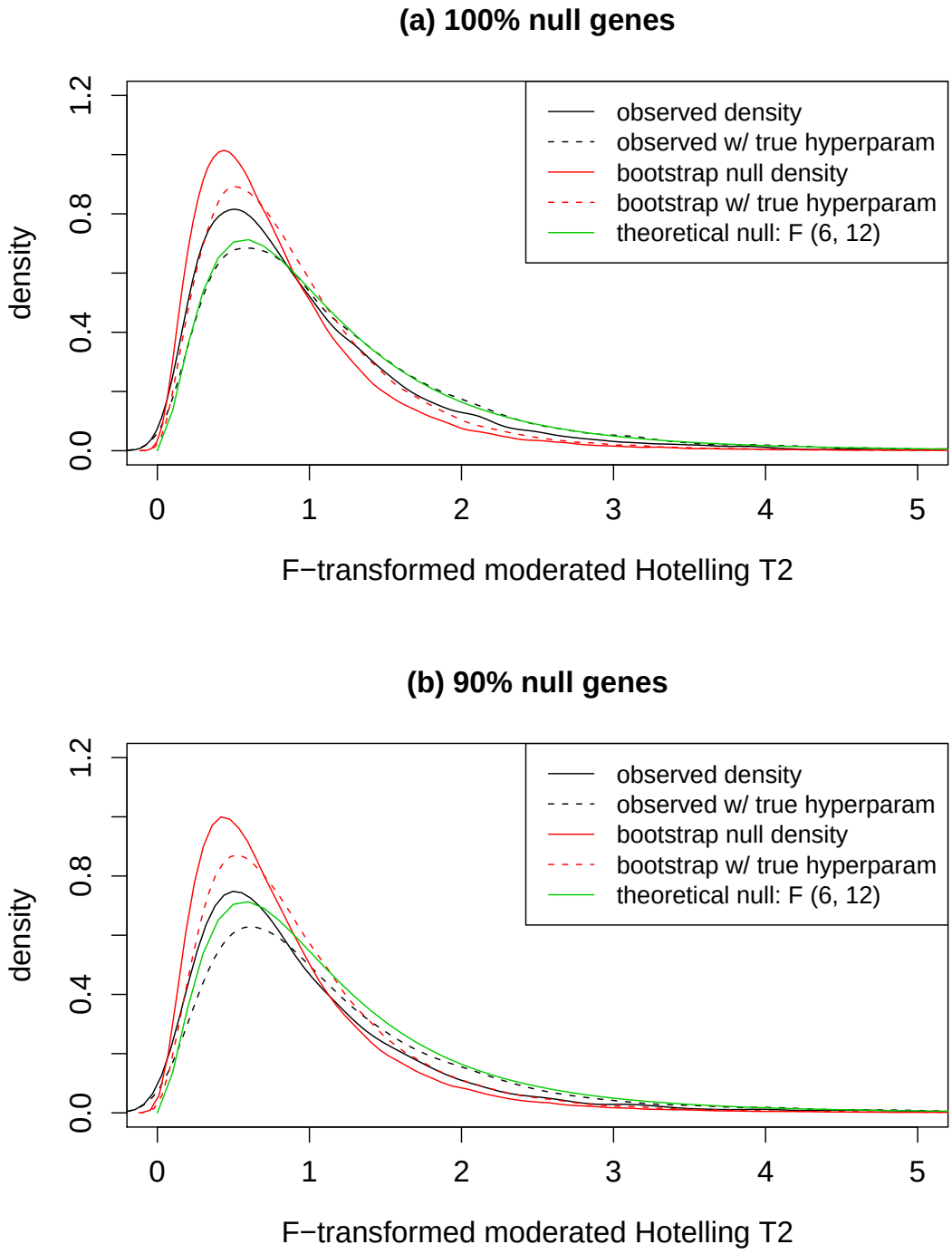

Figure 2.1: Distributions of the F-transformed $\widetilde{T}^2$ in small samples. The dimension of the multivariate normal is $k = 6$ and the sample size is $n = 6$. The true prior degrees of freedom used to simulate the data is $\nu = 12$. The estimate obtained from the simulated data set is $\hat{\nu} = 7$. (a) 100% null genes; (b) 90% null genes.

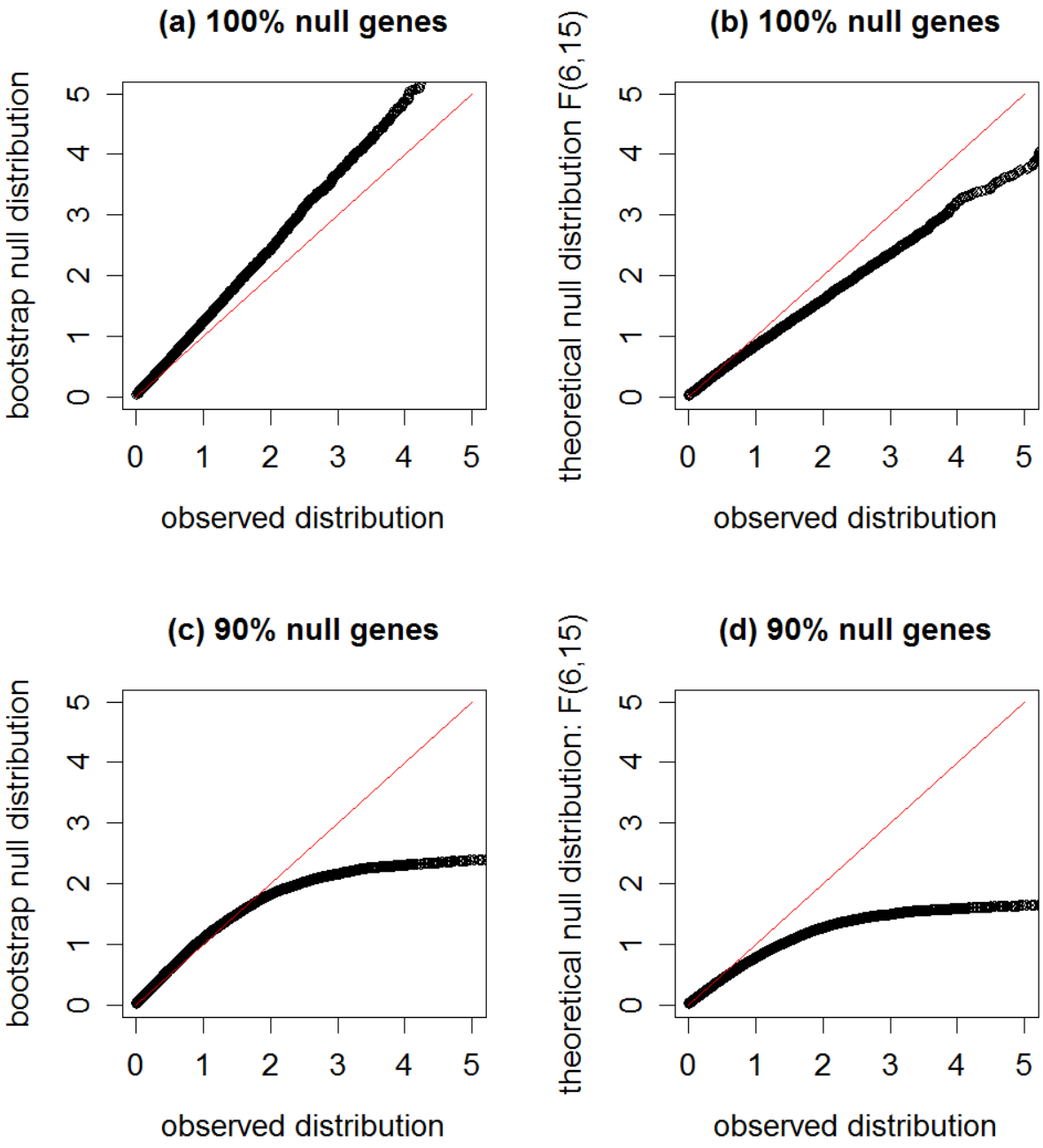

Figure 2.2: QQ-plots of the bootstrap null distribution vs. the observed (mixture) distribution and the theoretical null distribution. The dimension of the multivariate normal is $k = 6$ and the sample size is $n = 6$. (a) Bootstrap null versus observed, 100% null genes. (b) Bootstrap null versus theoretical, 100% null genes. (c) Bootstrap null versus observed, 90% null genes. (d) Bootstrap null versus theoretical, 90% null genes.

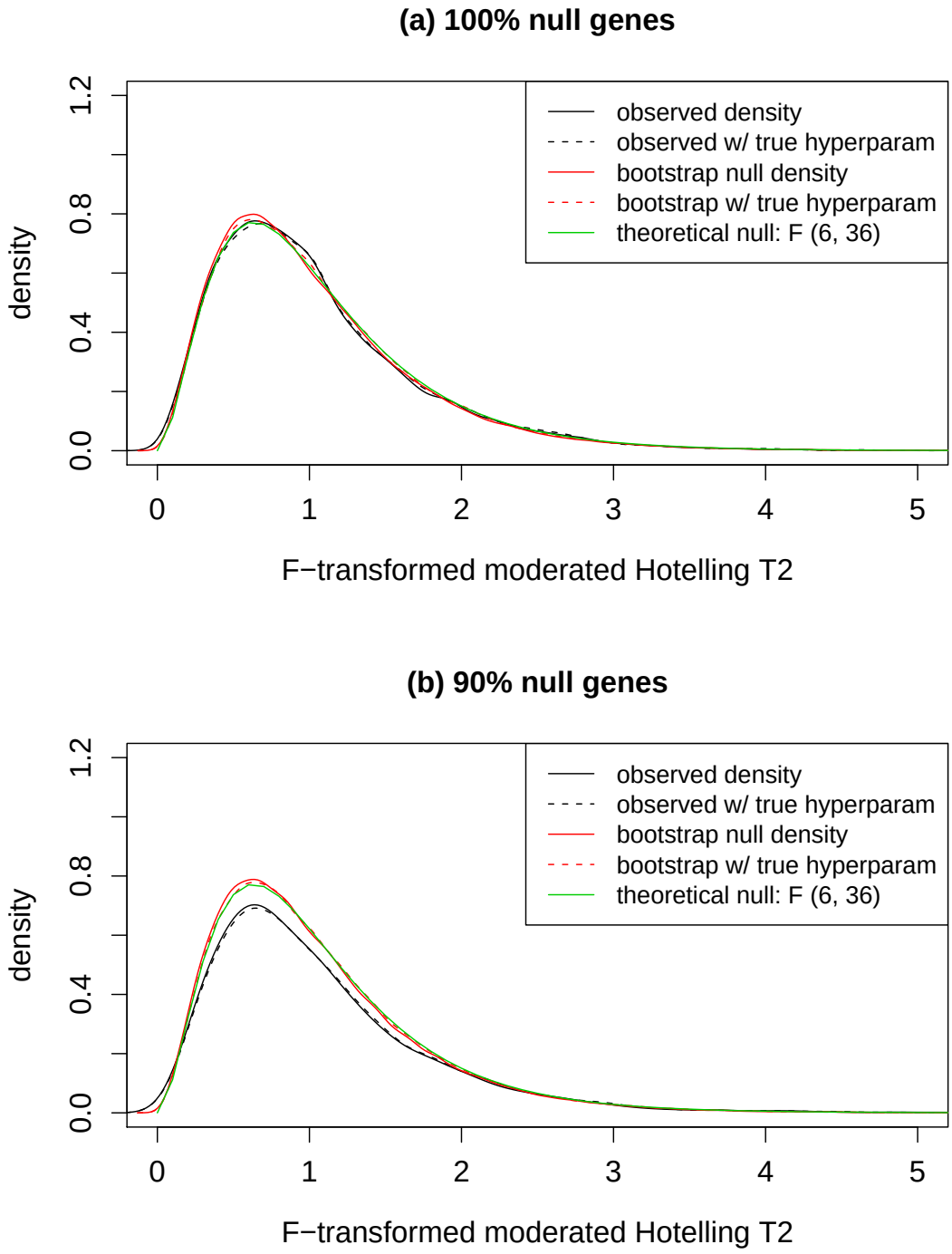

Figure 2.3: Distributions of the F-transformed $\widetilde{T}^2$ in large samples. The dimension of the multivariate normal is $k = 6$ and the sample size is $n = 30$. The true prior degrees of freedom used to simulate the data is $\nu = 12$. The estimate obtained from the simulated data set is $\hat{\nu} = 7$. (a) 100% null genes; (b) 90% null genes.

anti-conservative, because the bootstrap null distribution had a downward bias when the sample size was small. Both the frequentist and the empirical Bayes FDR procedures had improved performances when the sample size was increased.

2.5.2 Effects of moderation

Moderation plays important roles in the FDR procedures for two reasons. The first reason is that it affects both the theoretical and the empirical null distributions, as mentioned earlier. The second reason is related to the assumption of gene-wise independence. Both the empirical Bayes and the frequentist FDR-controlling procedures assume that the gene-specific test statistics are independent. Since moderation induces dependencies among the test statistics, it is important to investigate the effects of moderation on the FDR procedures. Recall that the amount of moderation is determined by the prior degrees of freedom ν. We performed some simulations with extreme values of ν, while holding all the other parameters fixed. Figures 2.5 and 2.6 compare the distributions of the F-transformed $\widetilde{T}^2$-statistics under different scenarios. In both figures, the left panels represent the small moderation scenario and the right panels represent the large moderation scenario. The only difference between these two figures is that, whereas the hyperparameters were estimated from the data in Figure 2.5, the known true hyperparameters were plugged-in to compute $\widetilde{T}^2$ in Figure 2.6.

When ν was set to the low-end value of 6, estimates of the hyperparameters were far from the truth. The observed distribution was strikingly different from the theoretical null distribution, but matched closely with the empirical null distribution (see Figure 2.5). When we controlled for the errors in hyperparameter estimation by plugging in the true value, the discrepancy between the observed and the theoretical null distributions was resolved (see Figure 2.6). Of course, in reality we never know the true hyperparameters. Therefore, it is important to be cautious about the variability of hyperparameter estimation. When ν was set to the high-end value of 60, the observed distribution of $\widetilde{T}^2$ matched quite well with the theoretical null distribution. The right-hand sides of Figures 2.5 & 2.6 appear similar, because hyperparameter estimation was fairly reliable when ν was large.

Figure 2.7 compares the two FDR-controlling procedures, in situations with varying amounts of moderation. When the degree of moderation was small ($\nu = 6$), the empirical Bayes method was preferable because the theoretical null distribution of the moderated Hotelling T^2 statistics was invalid. When the degree of moderation was large ($\nu = 60$), the frequentist method was preferable because hyperparameter estimation was quite reliable. The empirical Bayes method was slightly anti-conservative, because of the small sample bias in the bootstrap estimate of the null distribution.

2.6 Application to real data

The ability of the plant *Arabidopsis thaliana* to acquire resistance against pathogens depends on the production of salicylic acid. Around 14 days after infection by the powdery mildew *Golovinomyces orontii*, mutant plants that are defective for making salicylic acid develop some grayish white powdery blotches on their leaves and stems. On the other

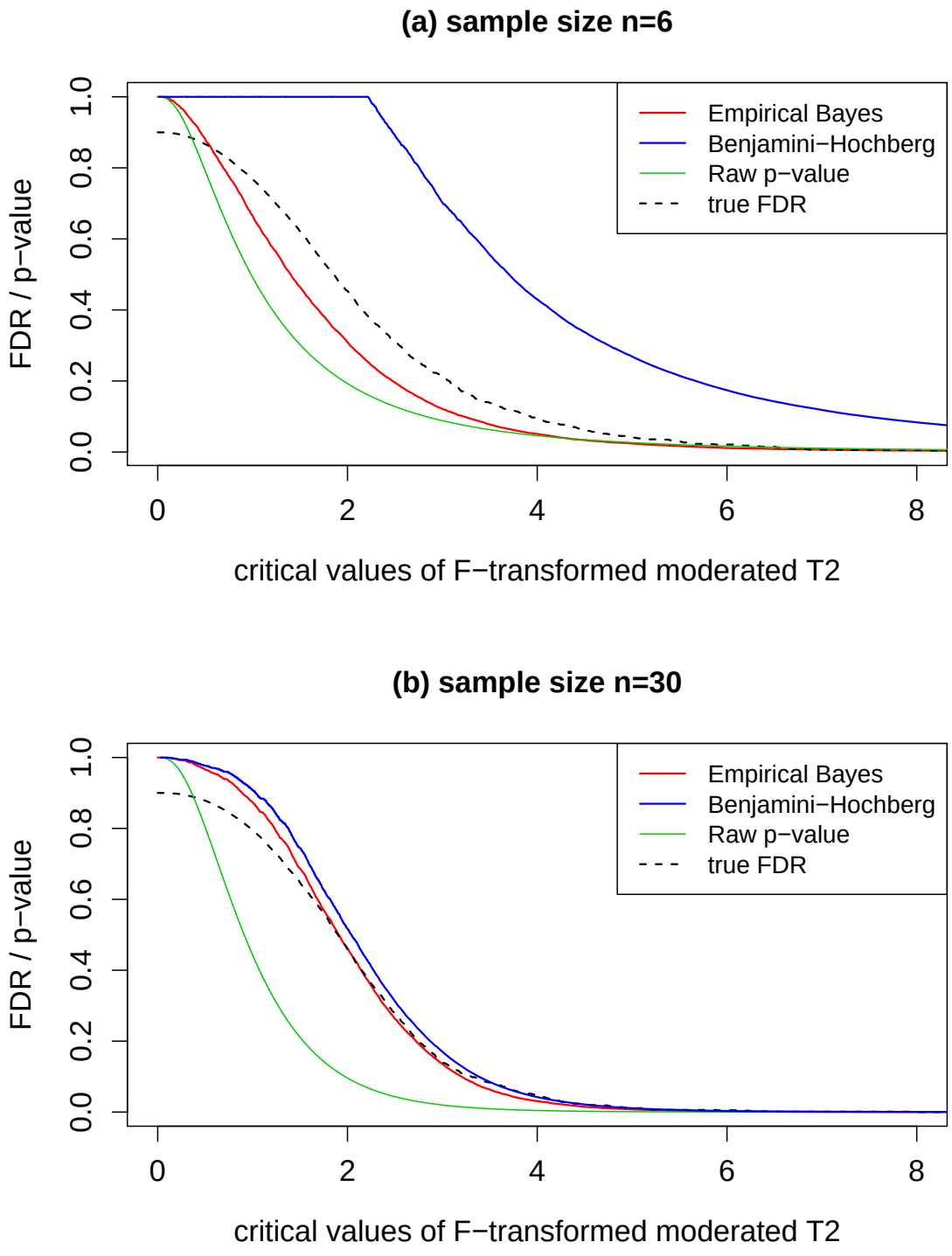

Figure 2.4: Comparison of FDR procedures applied to the moderated Hotelling T^2 statistics. The dimension of the multivariate normal is $k = 6$. (a) sample size $n = 6$; (b) sample size $n = 30$.

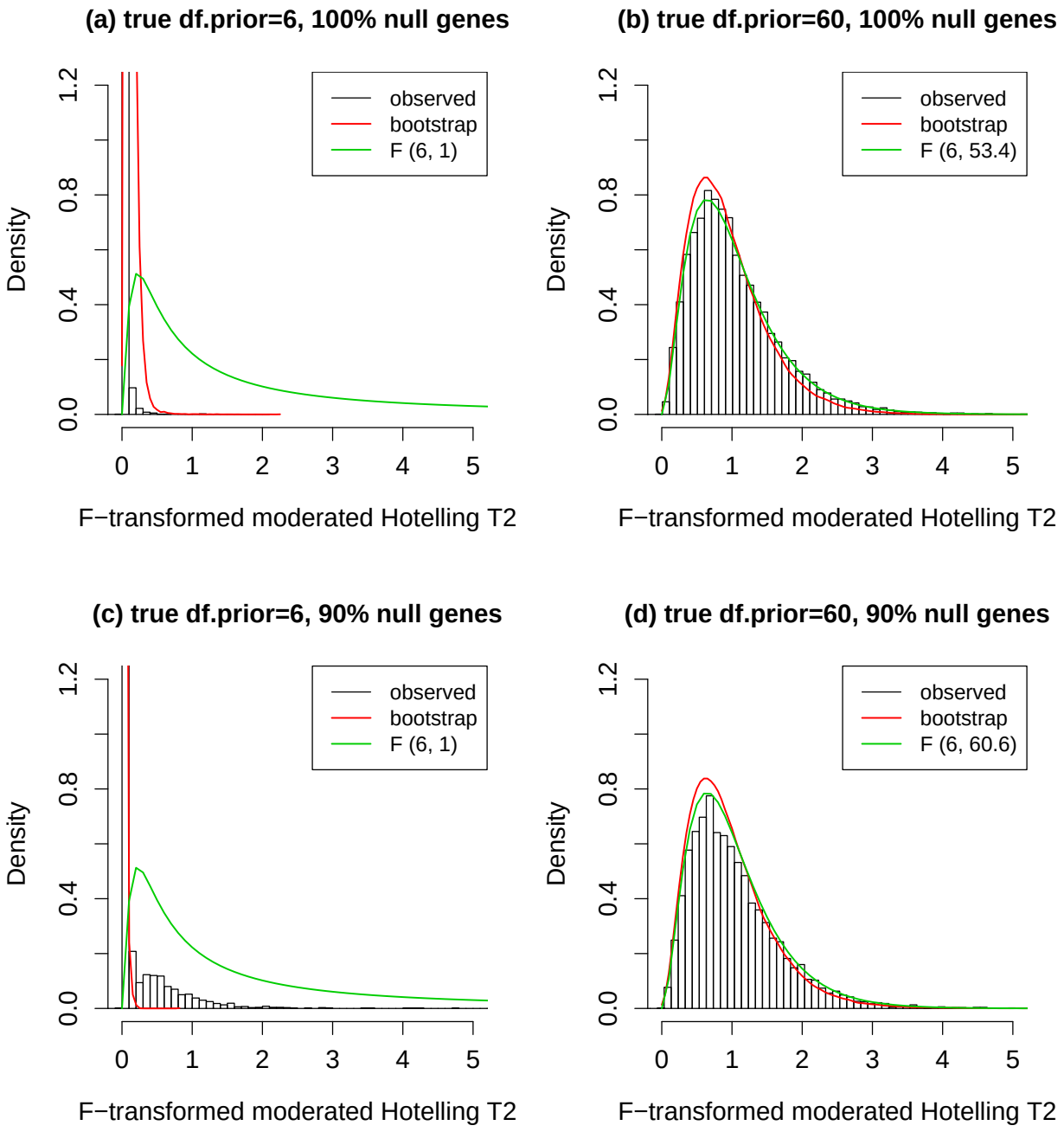

Figure 2.5: Effects of moderation on the distribution of the moderated Hotelling T^2 statistics. Each condition was examined by simulating two data sets. For each data set, the hyperparameters were estimated and the $\widetilde{T}^2$ statistics were computed using the *timecourse* package. The dimension of the multivariate normal is $k = 6$ and the sample size is $n = 6$. (a) true $\nu = 6$, estimate $\hat{\nu} = 1$, 100% null genes. (b) true $\nu = 60$, estimate $\hat{\nu} = 53$, 100% null genes. (c) true $\nu = 6$, estimate $\hat{\nu} = 1$, 90% null genes. (d) true $\nu = 60$, estimate $\hat{\nu} = 61$, 90% null genes.

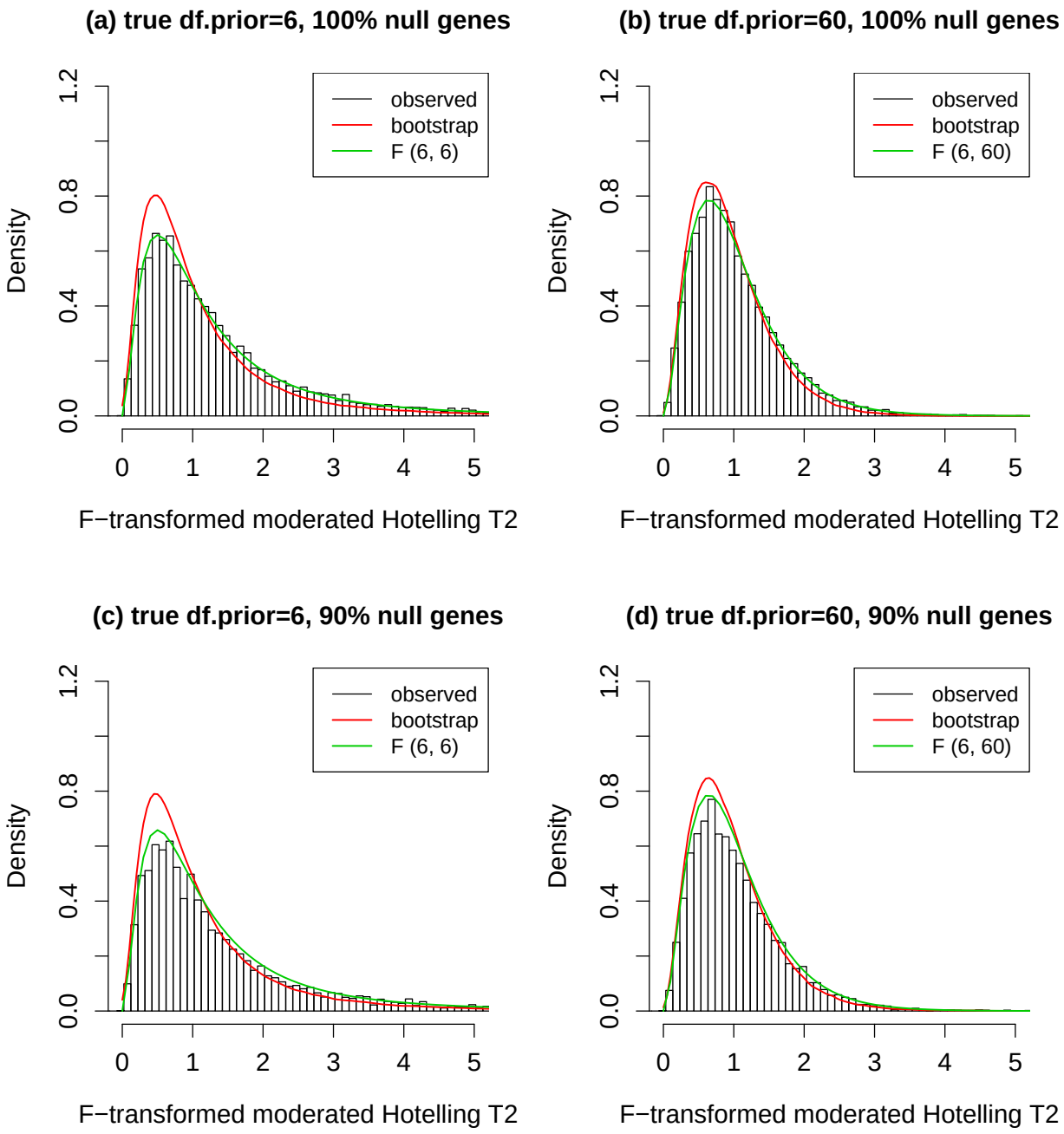

Figure 2.6: Effects of moderation on the distribution of the moderated Hotelling T^2 statistics. The known true hyperparameters were plugged-in. The dimension of the multivariate normal is $k = 6$ and the sample size is $n = 6$. (a) true $\nu = 6$, 100% null genes. (b) true $\nu = 60$, 100% null genes. (c) true $\nu = 6$, 90% null genes. (d) true $\nu = 60$, 90% null genes.

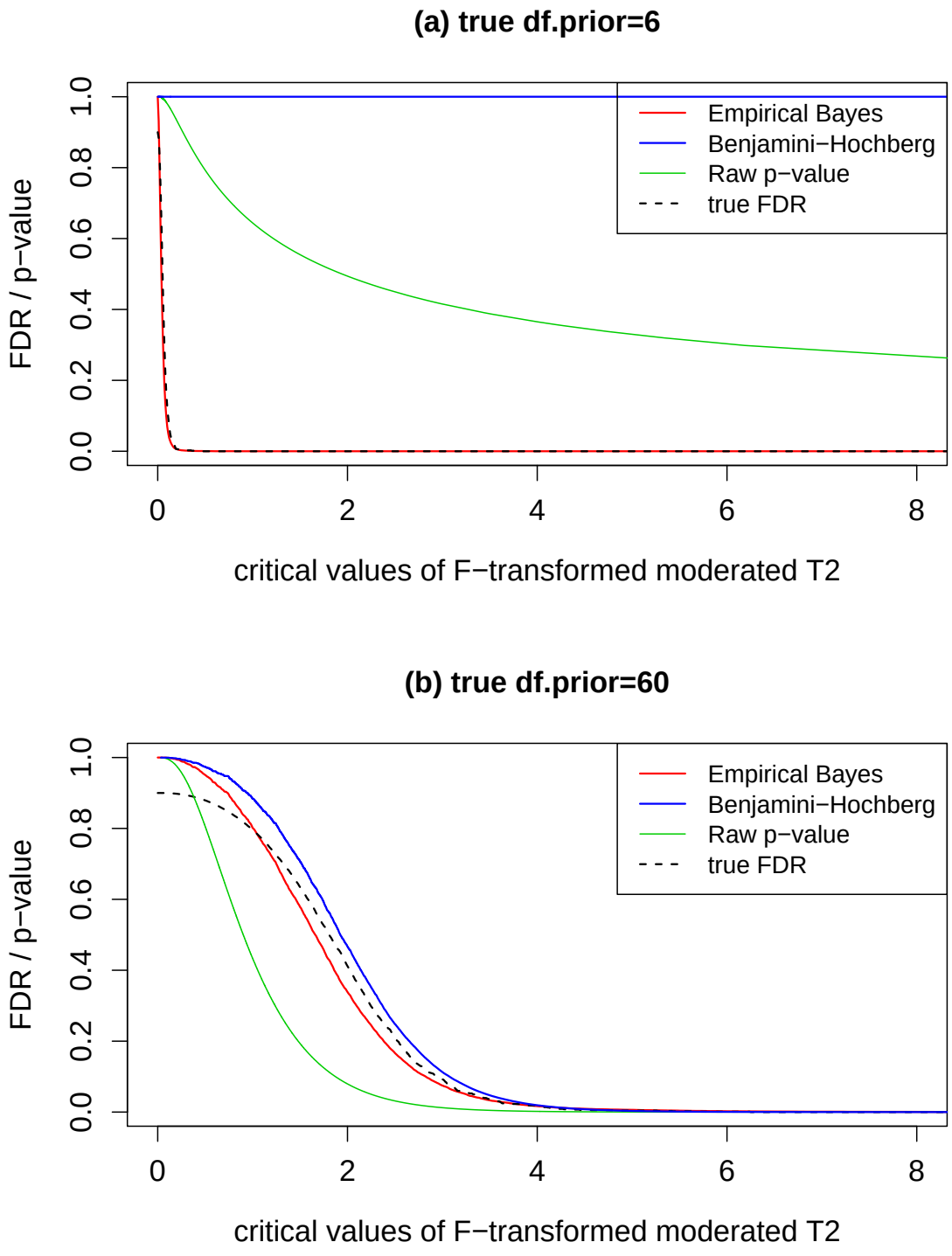

Figure 2.7: Effects of moderation on the FDR-controlling procedures for the moderated Hotelling T^2 statistics. The dimension of the multivariate normal is $k = 6$ and the sample size is $n = 6$. (a) true $\nu = 6$; (b) true $\nu = 60$.

hand, wild type plants that produce normal levels of salicylic acid remain healthy. Mary Wildermuth's group was interested in identifying genes whose expression profiles are different in these two types of plants. They performed a microarray time course study consisting of 6 time-points and 4 longitudinal replicates. The wild type and mutant samples were paired at each collection point [15]. This problem can be formulated as testing the null hypothesis that the paired differences, between gene expression measurements in the mutant and wild type plants, are expected to be the 6×1 vector of zeros. After some quality controls and preprocessing, the microarray data were converted into normalized log ratios of the fluorescence intensities for the mutant versus wild type samples. We performed a paired two-sample longitudinal analysis using the *timecourse* package. Visual examination of the time series plots and the knowledge of particular genes suggested that the top 870 genes have changed patterns of expression between the wild type and mutant plants [65].

A challenge in the analysis of this data set is that the sample size (4 replicates) is smaller than the dimension (6 time-points). Neither of the two FDR procedures produces satisfactory results in this situation. So we decided to circumvent the problem by splitting the data into two subsets. The full data set contained time points taken at 0, 0.25, 1, 3, 5, 7 hours post-infection. The 0, 1, 5 hr time points were assigned to Subset A; the 0.25, 3, 7 hr time points were assigned to Subset B. This arrangement aimed at capturing the long-range changes in the gene expression profiles.

We used simulations to benchmark the comparison of the two FDR procedures. Multivariate Gaussian data were simulated using parameters that were estimated from the real data. Because the hyperparameters η and ν tend to be under-estimated, we chose input parameters such that the final estimates from the simulated data matched well with the estimates from the real data. Table 2.1 summarizes the parameters used in the simulations. Figure 2.8 shows the distributions of the F-transformed moderated Hotelling T^2 statistics computed from the two subsets individually, for either the real data (left panels) or the simulated data (right panels). The heavier tails in the left panels suggest that the multivariate Gaussian model does not provide a perfect fit for the real data. However, this model is still useful for identifying the genes with changed expression profiles. Figure 2.9 compares the two FDR-controlling procedures, when applied to each subset individually. These figures illustrate that, under the conditions which emulate the *A. thaliana* data set, the frequentist procedure is too conservative. Although the empirical Bayes procedure is slightly anti-conservative, it yields FDR estimates that are much closer to the true false discovery rates.

Results from analyzing the individual subsets can be combined using an Intersection-Union-Test (IUT). The IUT is useful when the null hypothesis set can be expressed as a union of two sets, and the alternative hypothesis set is the intersection of their complement sets. The overall null hypothesis can be rejected only if each of the individual null hypotheses can be rejected. If each individual test has level-α, then the overall IUT also has level-α [5, 4]. For the *A. thaliana* data set, the overall null hypothesis states that the gene expression profile is unchanged, with respect to either Subset A or Subset B. The alternative hypothesis states that the gene expression profile is changed, with respect to both Subset A and Subset B. In other words, both the vector consisting of the 0, 1, 5 hour time-points, and the vector consisting of the 0.25, 3, 7 hour time-points are non-zero. This is more stringent than the

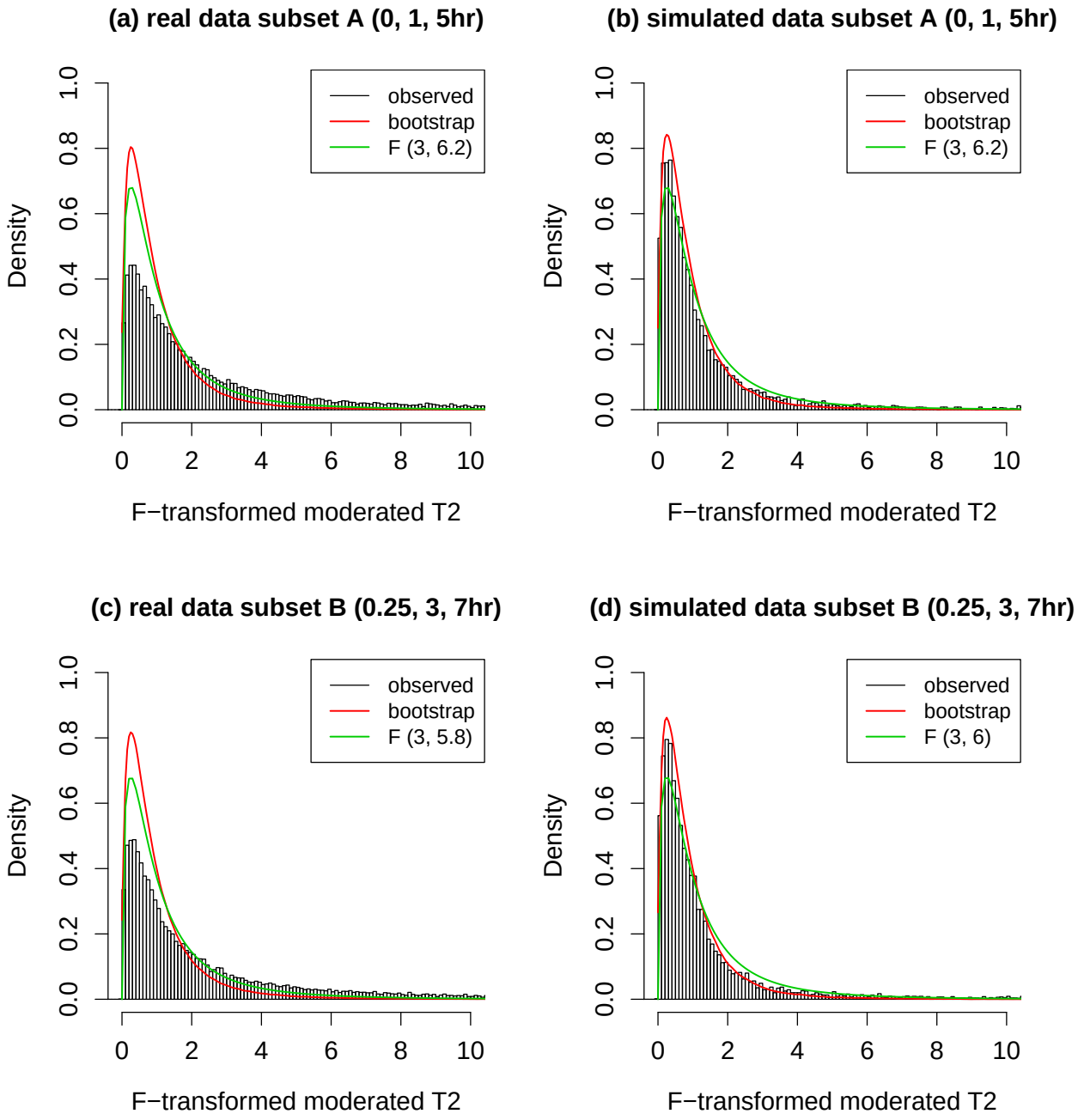

Figure 2.8: Distribution of the F-transformed moderated Hotelling T^2 statistics for the two subsets of the real data, and simulations based on the parameters estimated from the real data. The dimension of the multivariate normal is $k = 3$ and the sample size is $n = 4$ in each subset. Top panels: Subset A contains the 0, 1, 5 hours time points. Bottom panels: Subset B contains the 0.25, 3, 7 hours time points. Left panels: real data. Right panels: simulated data.

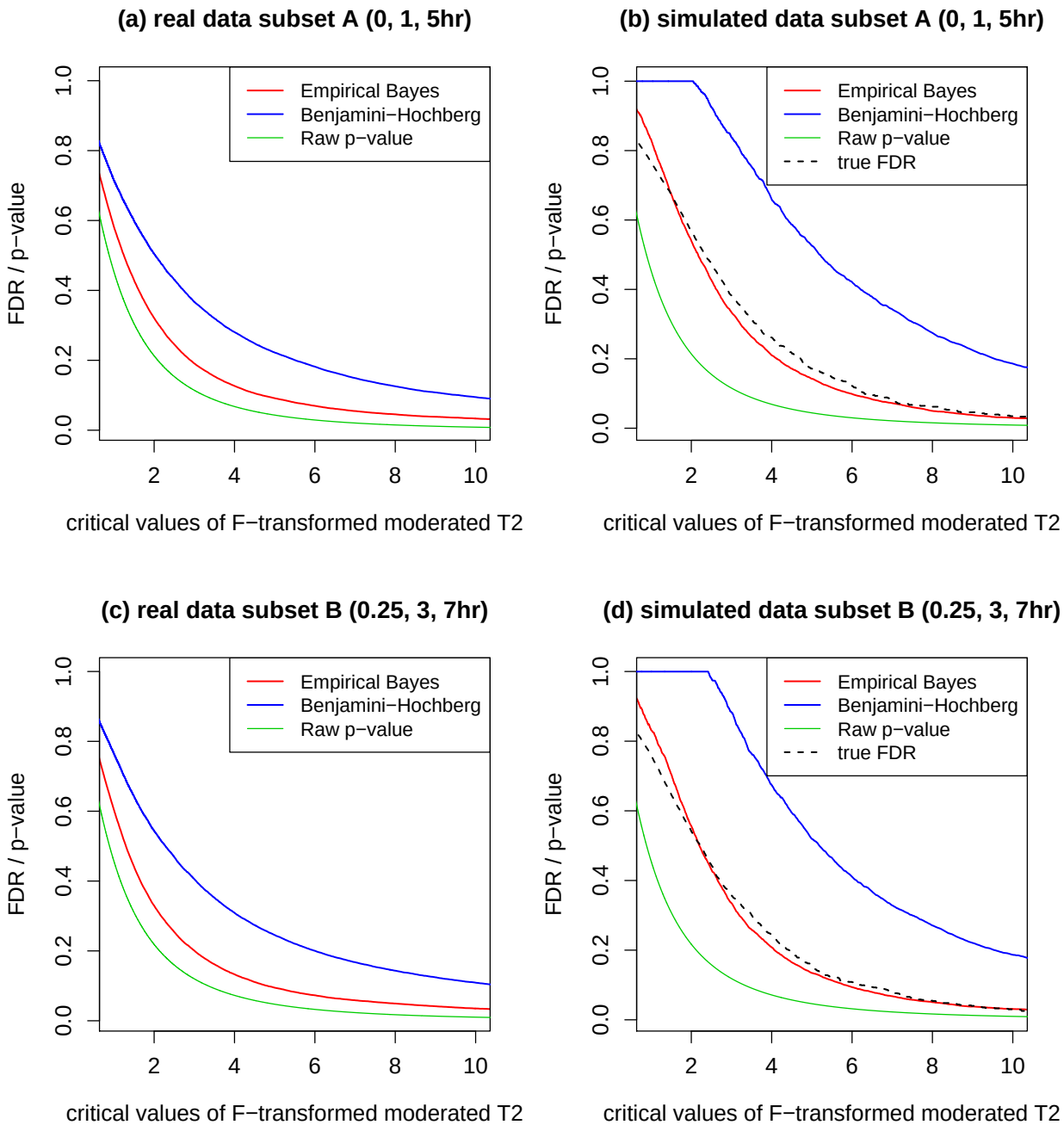

Figure 2.9: Comparison of the two FDR procedures applied to the moderated Hotelling T^2 statistics computed from the individual subsets. The dimension of the multivariate normal is $k = 3$ and the sample size is $n = 4$ in each subset. Top panels: Subset A contains the 0, 1, 5 hours time points. Bottom panels: Subset B contains the 0.25, 3, 7 hours time points. Left panels: real data. Right panels: simulated data.

Hyperparameters	p_1	η	ν	$\mathbf{\Lambda}$
Input for Simulating Subset A	0.10	0.34	7.2	0.025 0.000 -0.003 0.000 0.029 0.000 -0.003 0.000 0.036
Estimates from Simulated Subset A	0.02	0.08	5.2	0.030 0.000 -0.003 0.000 0.037 0.000 -0.003 0.000 0.045
Estimates from Real Subset A	0.02	0.09	5.2	0.025 0.000 -0.003 0.000 0.029 0.000 -0.003 0.000 0.036
Input for Simulating Subset B	0.10	0.28	6.8	0.037 0.002 0.002 0.002 0.023 0.004 0.002 0.004 0.035
Estimates from Simulated Subset B	0.02	0.07	5.0	0.051 0.002 0.002 0.002 0.032 0.005 0.002 0.005 0.047
Estimates from Real Subset B	002	0.07	4.8	0.037 0.002 0.002 0.002 0.023 0.004 0.002 0.004 0.035

Table 2.1: Hyperparameters used in the simulations aimed at emulating the setting of the real data.

alternative hypothesis for analyzing the full data set, which states that the vector consisting of all 6 time-points is non-zero.

The frequentist approach for controlling the FDR of the genome-wide IUTs involves the following steps. First, convert the F-transformed moderated Hotelling T^2 statistics into nominal p-values, based on the theoretical null distributions $\mathcal{F}(3, 6.2)$ and $\mathcal{F}(3, 5.8)$ for Subsets A and B, respectively. Second, determine the p-value of the IUT for each gene, by selecting the larger of the two p-values from either Subset A or Subset B. Third, apply the Benjamini-Hochberg FDR-controlling procedure to the IUT p-values, to adjust for genome-wide multiple testing. At the 5% level, the frequentist FDR procedure selected 936 genes, whose expression profiles changed in both Subset A and Subset B. Among these genes, 665 were also found in the top 870 genes obtained from analyzing the full time course.

The empirical Bayes FDR procedure for the genome-wide IUTs involves comparing the observed test statistics to an empirical null distribution. For each gene, the test statistic is the minimum of the F-transformed moderated Hotelling T^2 statistics, computed from either Subset A or Subset B alone. The empirical null distribution can be constructed by pooling together the bootstrap null statistics, obtained from resampling both Subset A and Subset B individually. We used 20 iterations of bootstrap resampling for each subset. This empirical estimate covers the true null distribution at the right-tail, because the test statistic is a minimum. Thus the level of the IUTs is conserved with this choice of the empirical null distribution. The method described in Section 2.4 is applied subsequently. At the 5% level,

the empirical Bayes FDR procedure selected 1017 genes, whose expression profiles changed in both Subset A and Subset B. Among these genes, 689 were also found in the top 870 genes obtained from analyzing the full time course.

The gene sets selected by the two different FDR procedures overlapped by 698 genes. Nonexpressor of pathogenesis-related genes 1 (NPR1) regulates systemic acquired resistance (SAR) in *A. thaliana*, and is induced after treatment with salicylic acid. This gene, along with 3 other genes related to NPR1, were among the overlapped set of 698 genes. The temporal expression profile of NPR1 is shown in Figure 2.10. The vertical axis represents the differences between expression measurements in the mutant and wild type plants. The horizontal axis represents days post-infection. Shown at the top of the plot is the rank according to the moderated Hotelling T^2 statistics obtained from the full time course. Figure 2.11 shows the temporal expression profiles of four genes that were selected by the empirical Bayes IUT, but not by the frequentist IUT. These genes appear to have non-zero expression patterns across time, regardless of their ranks according to the full time course. Figure 2.12 shows the temporal expression profiles of four genes that were selected by the frequentist IUT, but not by the empirical Bayes IUT. These genes were chosen from the top of the list ranked by the analysis of the full time course. Despite their high ranks, the temporal profiles of these genes do not appear to have any striking patterns other than random fluctuations. These results suggest that the empirical Bayes FDR-controlling procedure is more powerful for identifying genes with changed patterns of expression in the *A. thaliana* data set.

2.7 Summary of results

We investigated two FDR-controlling procedures for multiple-testing with the moderated Hotelling T^2 statistics. The frequentist approach relies on a theoretical null distribution, whereas the empirical Bayes approach relies on an empirical estimate of the null distribution. A recent simulation study by Bradley Efron [24] demonstrated that the theoretical null distribution may be far from reality, in the presence of high correlations among the statistics. This is an argument in support of the empirical null distribution.

We proposed an empirical Bayes FDR procedure for multiple testing using the moderated Hotelling T^2 statistic. The null distribution is estimated using the parametric bootstrap. The bootstrap distribution has a downward bias when the sample size is small, but converges to the theoretical null distribution when the sample size is sufficiently large, $n = 5k$. The small sample bias in the empirical null distribution leads to slightly anti-conservative estimates of the false discovery rates.

Another key factor that affects the performances of the FDR procedures is the amount of moderation. We simulated some data with varying prior degrees of freedom ν, to investigate the effects of moderation. When ν is small, hyperparameter estimation is prone to large errors. In such situations, the theoretical null distribution of $\widetilde{T}^2$ is invalid, thus the frequentist FDR procedure fails. However, the parametric bootstrap remains to be an adequate estimator for the true null distribution. When ν is large, hyperparameter estimation is reliable, thus the frequentist FDR procedure is optimal. Hence the frequentist FDR procedure is highly sensitive to hyperparameter estimation errors, which may be disastrous

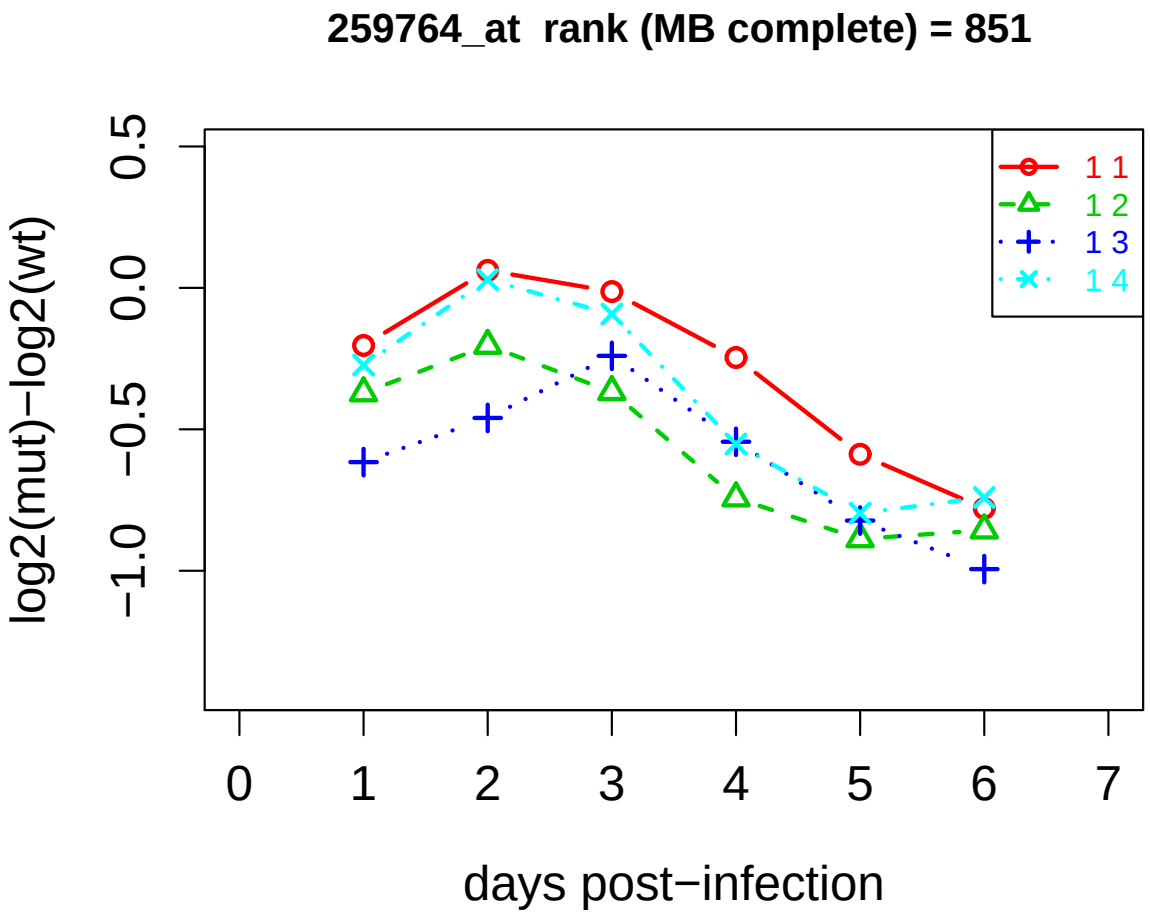

Figure 2.10: Temporal expression profile of Nonexpressor of pathogenesis-related genes 1 (NPR1). Shown at the top of this plot is the rank of the moderated Hotelling T^2 statistic obtained from the analysis of the full data set, containing 6 time points. Each curve represents an independent longitudinal replicate of the time course, indexed by 11 to 14.

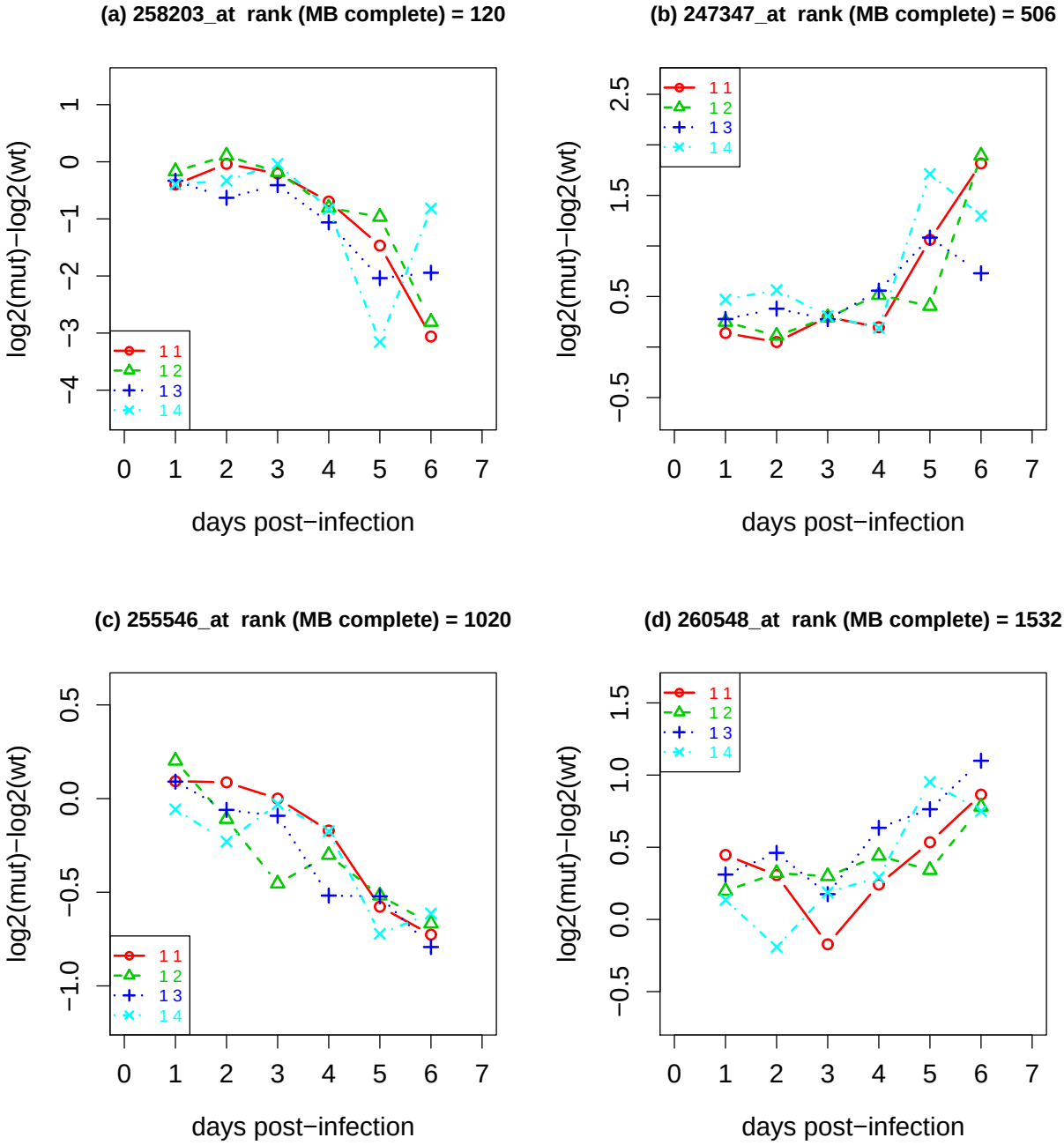

Figure 2.11: Temporal expression profiles of 4 genes that were selected by the empirical Bayes IUT, but not by the frequentist IUT, at 5% FDR. Shown at the top of each plot is the rank of the moderated Hotelling T^2 statistic obtained from the analysis of the full data set, containing 6 time points. Each curve represents an independent longitudinal replicate of the time course, indexed by 11 to 14.

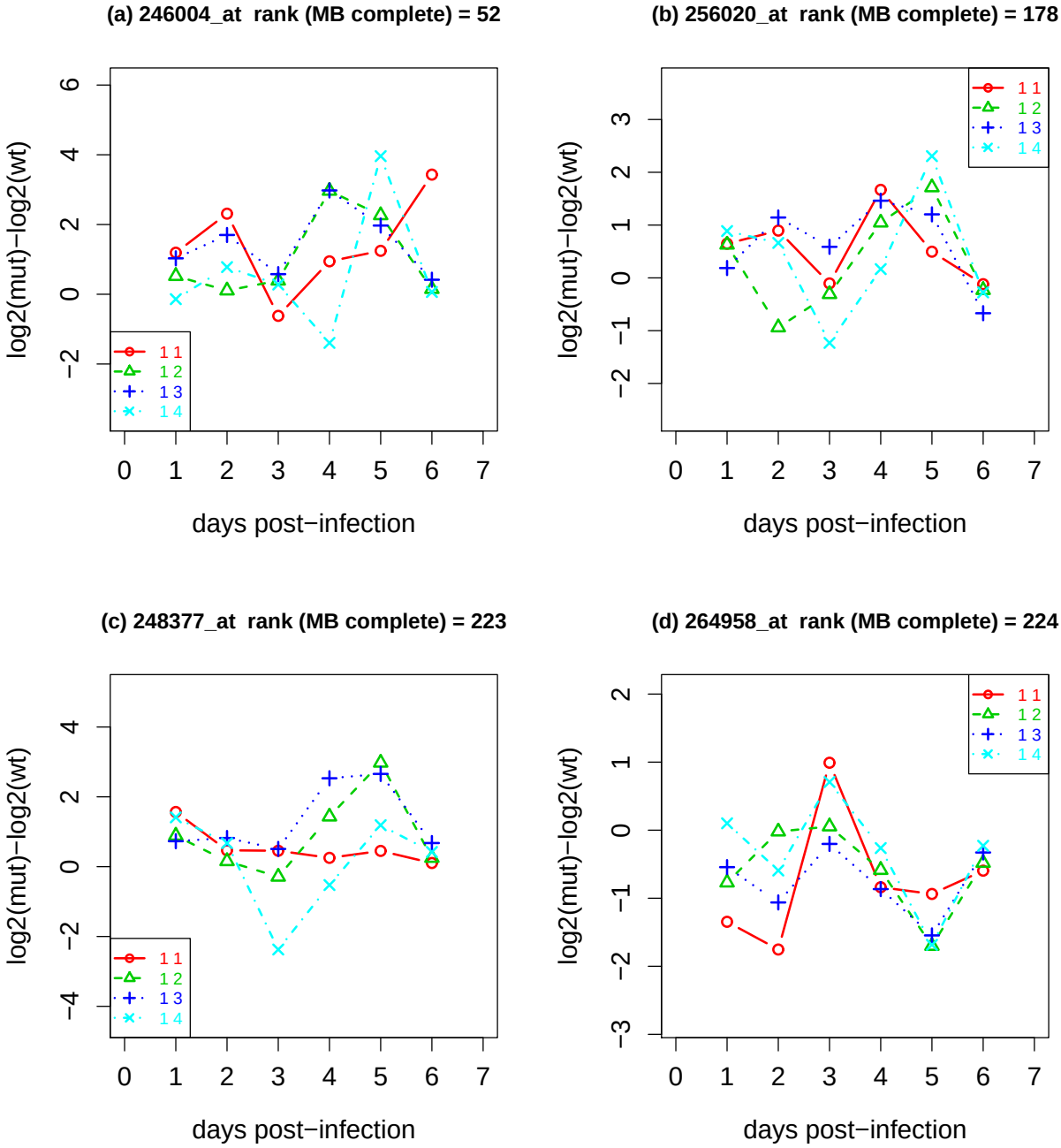

Figure 2.12: Temporal expression profiles of 4 genes that were selected by the frequentist IUT, but not by the empirical Bayes IUT, at 5% FDR. Shown at the top of each plot is the rank of the moderated Hotelling T^2 statistic obtained from the analysis of the full data set, containing 6 time points. Each curve represents an independent longitudinal replicate of the time course, indexed by 11 to 14.

when the true value of ν is small. The empirical Bayes FDR procedure, albeit its slightly anti-conservative nature in small samples, is more broadly useful because of its robustness.

The two FDR-controlling procedures are feasible only when the sample size n (the number of replicates) is at least as large as the dimension k (the number of time points). In real microarray experiments, the number of replicates is often limited. This problem can be circumvented by splitting the full time course into subsets of fewer time-points. Results from analyzing the individual subsets can be combined using intersection-union tests (IUTs). The FDR-controlling procedures can be adapted for the IUTs, as discussed in Section 2.6. An application to the *A. thaliana* data demonstrated that the empirical Bayes FDR procedure is more powerful for identifying genes with changed patterns of expression.

Chapter 3

A nonhomogeneous hidden Markov model for integrating ChIP-chip data from multiple tiling array designs

3.1 Motivation

A key step in the regulation of gene expression is the localization of DNA binding proteins to specific sites on the chromosomes. Biologists often perform ChIP-chip experiments to identify the chromosomal targets of these DNA binding proteins. Barbara Meyer's group at UC Berkeley is interested in a process called "dosage compensation" that ensures a balanced expression of sex-linked genes. In most higher organisms, the female individual carries two copies of the X-chromosome, whereas the male individual carries only one. If left unregulated, the female individual would express twice the amount of X-linked gene products as the male individual, leading to some serious consequences. Different species have evolved different mechanisms of ensuring that equal levels of X-linked gene products are expressed in the two genders. In mammals one copy of the female X-chromosome is completely inactivated, in the form of Barr bodies. The *Caenorhabditis elegans* genome contains five autosomes and one sex chromosome. The female-equivalent worms are called hermaphrodites, because they can self-fertilize. Whereas the hermaphrodite worms carry two copies of the X-chromosome (XX), the male worms carry only one copy (XO). The hermaphrodites reduce the expression of X-linked genes by one-half, through a process called dosage compensation. This process involves the binding of a protein complex, called the dosage compensation complex (DCC), to both copies of Chromosome X in the hermaphrodites. A review of this subject can be found in WormBook [46].

The Meyer group is interested in understanding how the dosage compensation complex regulates the expression of X-linked genes. Thus they performed some ChIP-chip experiment to identify the DNA binding targets of this protein complex, using the two-color NimbleGen tiling arrays. Due to the scale and duration of this project, the experiments were performed using three different designs of tiling arrays, as listed in Table 3.1. All of the "replicates" in this study are biological replicates, i.e. the RNA samples were obtained from independent

Design	Label	Description
1	WS170-50	WS170 genome release, isothermal probes, tiled at roughly every 50 base pairs
2	WS180-40-norep	WS180 genome release, masked out repeat regions, tiled at roughly every 40 base pairs
3	WS180-50	WS180 genome release, isothermal probes, tiled at roughly every 50 base pairs

Table 3.1: Summary of tiling array designs

Condition	Protein	Design 1	Design 2	Design 3	Total # "rep"
wild type	Dpy-27	1	1	1	3
wild type	Sdc-3	0	1	2	3
wild type	IgG control	0	0	2	2
smo-1 mutant	Dpy-27	1	1	1	3
smo-1 mutant	Sdc-3	0	2	0	2
smo-1 mutant	IgG control	0	0	1	1

Table 3.2: Summary of ChIP-chip experiments

chromatin immunoprecipitation experiments.

Dpy-27 is a chromosome condensation protein homolog that regulates dosage compensation through association with Chromosome X [16]. Sdc-3 is a protein that coordinately controls both sex determination and dosage compensation. It contains two zinc finger domains that are required for association with the hermaphrodite X-chromosomes. Other components of the DCC, including Dpy-27, are required for the synthesis, stability and localization of Sdc-3 to the X-chromosome [18]. Both Dpy-27 and Sdc-3 were analyzed by ChIP-chip experiments under two different conditions. In wild type worms, the DCC is assembled properly, thus Dpy-27 and Sdc-3 bind jointly to Chromosome X to mediate dosage compensation. There is some unpublished evidence that the assembly of the DCC requires a type of histone-modification called sumoylation. Thus the DNA-binding profiles of Dpy-27 and Sdc-3 are likely to change in mutant worms with the *smo-1* gene knocked-out. A summary of the experiments is given in the Table 3.2. This chapter is devoted to the analysis of a single protein at a time. The next chapter is devoted to the joint analysis of multiple proteins. A separate analysis of the IgG control data will be discussed in Section 4.6.3.

When replicate ChIP-chip experiments are performed on tiling arrays with different probe designs, integration of the data is a challenge. None of the existing methods for analyzing ChIP-chip data address this challenge. One obvious way of integrating the different tiling array designs is to create a pseudo-design by breaking the genome into contiguous bins. Each bin should be large enough to cover at least a few probes from the original design.

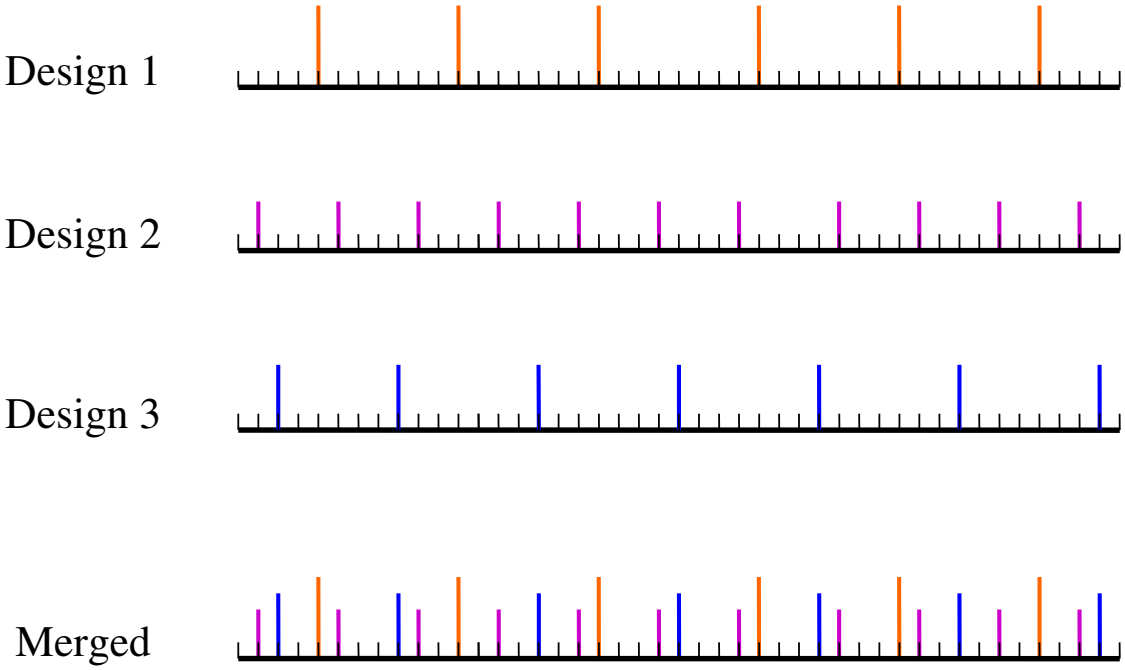

Figure 3.1: Data integration approach

The original data can be transformed by taking weighted averages of the probes mapped to each bin in the pseudo-design. Then, the conventional methods for ChIP-chip analysis can be applied to the binned data. However, a major drawback of this approach is the loss of resolution. Ideally, the different designs should be integrated while preserving the highest resolution in the original data. Figure 3.1 illustrates our approach to the data integration problem. Probes from the different designs are mapped onto a merged design at the single-base resolution. As a convention, the center position of each probe is used for this mapping. The origin of each probe is indicated in this figure by the color of the vertical strip. We developed a nonhomogeneous hidden Markov model to realize the single-base level of data integration.

3.2 Nonhomogeneous hidden Markov model for one protein

Hidden Markov models (HMMs) have been used extensively in genomics for over a decade. Array CGH studies that aim at identifying segments of chromosomal copy-number aberrations often take the HMM approach to data analysis [29, 44]. A few transcription factor

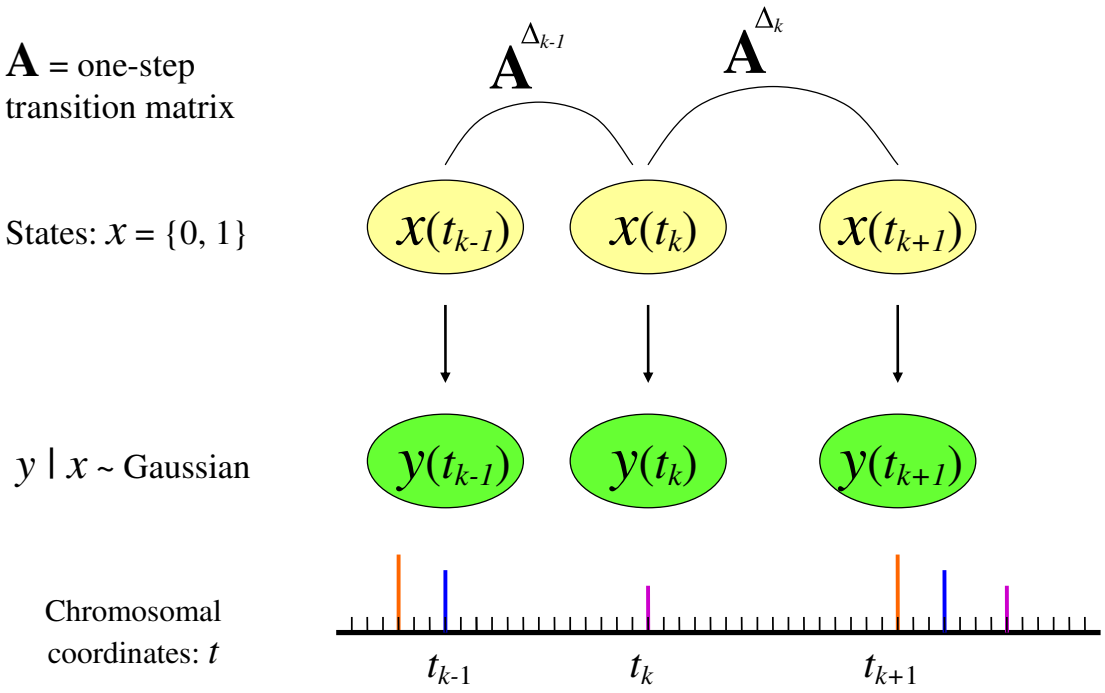

Figure 3.2: Nonhomogeneous hidden Markov model

mapping studies also incorporated HMMs in the analysis of tiling array data [41, 35, 20]. The most common type of HMM assumes that the intervals between adjacent observations are equidistant, a property known as "time homogeneous." In order to achieve data integration at the single-base level, we need a model that can accommodate intervals of variable lengths. Thus we propose a nonhomogeneous version of the hidden Markov model stated as follows. At each base position along the genome, the binding state of a particular protein is binary (0=unbound, 1=bound). Transitions between the states follow an unobserved Markov chain. At some (but not all) base positions, observations are emitted according to some Gaussian distributions, conditional on the hidden states. The emission distributions are protein-specific and design-specific.

Figure 3.2 illustrates some notation that is necessary for specifying the model. Let t_k, for $k \in \{1, ..., T\}$, denote the genomic positions, in base pairs, at which observations exist. The initial position $t_0 = 0$ is unobserved in general. Let $\Delta_k = t_{k+1} - t_k$ denote the number of single-base steps (i.e. base pairs) between adjacent observations. Let $x(t_k) \in \{0, 1\}$ denote the hidden state at position t_k. Let $y(t_k)$ denote the observation at position t_k. Multiple observations occurring at the same position are treated as conditionally independent.

$$y(t_k)|x(t_k) = 0 \sim \mathcal{N}(\mu_0, \sigma_0^2)$$
$$y(t_k)|x(t_k) = 1 \sim \mathcal{N}(\mu_1, \sigma_1^2)$$

Transitions between the states are dictated by a Markov chain. Let π denote the vector of initial probabilities. Let $\mathbf{A}$ denote the one-step transition matrix, with entries a_{ij}. The size of $\mathbf{A}$ is 2×2. Write $a(n; i, j)$ for entries of the n-step transition matrix. The ultimate goal of fitting this model is to infer about the hidden states at all observed positions. In order to do so, we need to estimate the parameters $\mu_0, \sigma_0^2, \mu_1, \sigma_1^2, \pi, \mathbf{A}$.

3.2.1 Approximation of the transition matrix

Since the transition matrix is often concentrated along the diagonal, we can make the following approximation. Let $\mathbf{Q}$ denote the difference between $\mathbf{A}$ and the identity matrix. If n is small, then we can approximate the matrix exponential linearly.

$$\mathbf{A} = \mathbf{I} + \mathbf{Q}$$
$$\mathbf{A}^n \approx \mathbf{I} + n\mathbf{Q}$$

The algorithm for estimating the transition matrix will be described later in this section. Let us fast forward and write down the estimated one-step transition matrix $\hat{\mathbf{A}}$ obtained by applying this algorithm to the wild type Dpy27 data for a 1 MB region of Chromosome X. The majority of the gaps between adjacent probes are less than 40 base pairs, after integration of the different array designs. The following equations demonstrate that the linear approximation holds sufficiently for $n \leq 40$.

$$\hat{\mathbf{A}} = \begin{pmatrix} 0.9996 & 0.0004 \\ 0.0021 & 0.9979 \end{pmatrix}$$

$$\hat{\mathbf{Q}} = \hat{\mathbf{A}} - \mathbf{I} = \begin{pmatrix} -0.0004 & 0.0004 \\ 0.0021 & -0.0021 \end{pmatrix}$$

$$40\hat{\mathbf{Q}} = \begin{pmatrix} -0.0175 & 0.0176 \\ 0.0835 & -0.0835 \end{pmatrix}$$

$$\mathbf{I} + 40\hat{\mathbf{Q}} = \begin{pmatrix} 0.9824 & 0.0176 \\ 0.0835 & 0.9165 \end{pmatrix}$$

$$\hat{\mathbf{A}}^{40} = \begin{pmatrix} 0.9829 & 0.0171 \\ 0.0814 & 0.9186 \end{pmatrix}$$

3.2.2 Modified forward-backward algorithm

We provide the forward-backward algorithm for the nonhomogeneous HMM, without writing out its derivation. The only distinction from the conventional forward-backward algorithm is that the one-step transition probabilities are replaced by the Δ_k-step transition probabilities.

The forward variable is defined as:

$$\alpha(x(t_k)) = p(y(t_1), ..., y(t_k), x(t_k)).$$

For the nonhomogeneous case, its recursion equations are given below.

$$\begin{aligned}\alpha(x(t_1)) &= \pi p(y(t_1)|x(t_1)) \\ \alpha(x(t_{k+1})) &= \sum_{x(t_k)} \alpha(x(t_k)) a(\Delta_k; x(t_k), x(t_{k+1})) p(y(t_{k+1})|x(t_{k+1}))\end{aligned}$$

The backward variable is defined as:

$$\beta(x(t_k)) = p(y(t_{k+1}), ..., y(t_T)|x(t_k)).$$

For the nonhomogeneous case, its recursion equations are given below.

$$\begin{aligned}\beta(x(t_T)) &= \mathbf{1} \\ \beta(x(t_{k-1})) &= \sum_{x(t_k)} \beta(x(t_k)) a(\Delta_{k-1}; x(t_{k-1}), x(t_k)) p(y(t_k)|x(t_k))\end{aligned}$$

The gamma variable is defined as the posterior probability: $\gamma(x(t_k)) = p(x(t_k)|\mathbf{y})$. It can be obtained from the forward and backward variables.

$$\gamma(x(t_k)) = \frac{\alpha(x(t_k))\beta(x(t_k))}{\sum_{x(t_k)} \alpha(x(t_k))\beta(x(t_k))}$$

To estimate the transition matrix, we also need the co-occurrence probabilities:

$$\xi(x(t_k), x(t_{k+1})) = p(x(t_k), x(t_{k+1})|\mathbf{y}).$$

For the nonhomogeneous case, their recursion equations are given below.

$$\xi(x(t_k), x(t_{k+1})) = \frac{\alpha(x(t_k)) a(\Delta_k; x(t_k), x(t_{k+1})) \beta(x(t_{k+1}) p(y(t_{k+1})|x(t_{k+1}))}{\sum_{x(t_k)} \alpha(x(t_k))\beta(x(t_k))}$$

3.2.3 Modified Baum-Welch algorithm

The Baum-Welch algorithm [2] is often used for estimating the parameters in hidden Markov models. A review by Rabiner provides a derivation of this algorithm for the homogeneous HMM [53]. Our nonhomogeneous model requires some modifications to this algorithm. We summarize below the modified Baum-Welch update equations for the nonhomogeneous HMM. A detailed derivation will be presented later in this subsection.

- Initial distribution:
$$\hat{\pi}_i = \frac{\sum_{k=1}^{T-1} \gamma(x(t_k) = i))}{T-1}$$

- Transition probabilities:
$$\begin{aligned}
\hat{a}_{ij} =& \frac{1}{\lambda_i} \sum_{k=1}^{T-1} \xi(x(t_k) = i, x(t_{k+1}) = j) \\
\hat{a}_{ii} =& 1 - \sum_{j:j \neq i} \hat{a}_{ij} \\
\hat{\lambda}_i =& \sum_{k=1}^{T-1} \left[\frac{\Delta_k}{1 + \Delta_k(\hat{a}_{ii} - 1)} \right] \xi(x(t_k) = i, x(t_{k+1}) = i) \\
& \text{where } \hat{a}_{ii} \text{ is estimated from the previous iteration.}
\end{aligned}$$

- Emission distribution means:
$$\hat{\mu}_i = \frac{\sum_{k=1}^{T-1} \gamma(x(t_k) = i) y(t_k)}{\sum_{k=1}^{T} \gamma(x(t_k) = i)}$$

- Emission distribution variances:
$$\hat{\sigma}_i^2 = \frac{\sum_{k=1}^{T-1} \gamma(x(t_k) = i)[y(t_k) - \hat{\mu}_i]^2}{\sum_{k=1}^{T} \gamma(x(t_k) = i)}$$

We now derive the Baum-Welch update equations for the nonhomogeneous HMM. The Baum-Welch algorithm is an expectation-maximization (EM) algorithm, which aims at finding the maximum likelihood estimates of the parameters in the HMM. Let θ denote the vector of parameters, including the emission parameters, the stationary probabilities and the one-step transition matrix. The complete data consist of the hidden states and the observations at positions t_k for $k \in \{1, ..., T\}$. We would like to maximize the likelihood of the parameter given the observed data: $L(\theta|\mathbf{y})$. But this is very hard, because the sampling density of $\mathbf{y}$ is a marginal density. EM is an iterative method that alternates between an expectation (E) step and a maximization (M) step. Let the superscript (r) index the iterations. In the E-step, compute the expectation of the complete log likelihood, with respect to the unobserved data, conditional on the observed data $\mathbf{y}$ and the current parameter estimate $\theta^{(r)}$. In the M-step, choose an updated parameter estimate $\theta^{(r+1)}$ to maximize the conditional expectation of the complete log likelihood. An EM update never decreases $L(\theta|\mathbf{y})$, thus convergence to a local maximum is guaranteed [19].

We now derive the update equations for the nonhomogeneous HMM. The complete likelihood function for this model is:

$$L(\theta|\mathbf{x}, \mathbf{y}) = \pi(x(t_0)) \prod_{k=0}^{T-1} a(\Delta_k; x(t_k), x(t_{k+1})) p(y(t_{k+1})|x(t_{k+1})).$$

After taking the logarithm, we obtain:

$$\log L = \log \pi(x(t_0)) + \sum_{k=0}^{T-1} \log a(\Delta_k; x(t_k), x(t_{k+1})) + \sum_{k=0}^{T-1} \log p(y(t_{k+1})|x(t_{k+1})).$$

In the E-step, we compute the expectation of the complete log likelihood conditional on the observed data. This can be written as the sum of the conditional expectations of the individual terms in the complete log likelihood function. The parameters are fixed at the values of the current estimates. The superscript (r), indicating the iteration number, is dropped to simplify the notations. Because t_0 is unobserved in general, we can drop it from the last two terms.

$$\begin{aligned}
&\mathbf{E}[\log \pi(x(t_0))|\mathbf{y}] = \sum_i \gamma(x(t_0) = i) \log \pi(x(t_0) = i) \\
&\mathbf{E}\left[\sum_{k=0}^{T-1} \log a(\Delta_k; x(t_k), x(t_{k+1}))|\mathbf{y}\right] = \frac{T}{T-1}\mathbf{E}\left[\sum_{k=1}^{T-1} \log a(\Delta_k; x(t_k), x(t_{k+1}))|\mathbf{y}\right] \\
=&\frac{T}{T-1}\sum_{k=1}^{T-1}\left\{\sum_i \sum_j \xi(x(t_k) = i, x(t_{k+1}) = j)\ \log a(\Delta_k; i, j))\right\} \\
&\mathbf{E}\left[\sum_{k=0}^{T-1} \log p(y(t_{k+1})|x(t_{k+1}))|\mathbf{y}\right] = \frac{T}{T-1}\mathbf{E}\left[\sum_{k=1}^{T-1} \log p(y(t_{k+1})|x(t_{k+1}))|\mathbf{y}\right] \\
=&\frac{T}{T-1}\sum_{k=1}^{T-1}\left\{\sum_i \gamma(x(t_{k+1}) = i)\ \log p(y(t_{k+1})|x(t_{k+1}) = i)\right\}
\end{aligned}$$

In the M-step, we choose $\theta^{(r+1)}$ to maximize the conditional expectation of the complete log likelihood. Since all of the terms in this function are non-negative, maximizing each term individually achieves maximization of the sum.

1. **Initial Distribution of the Markov chain**

   Maximization of $\mathbf{E}[\log \pi(x(t_0))|\mathbf{y}]$ with respect to π_i, and subject to the constraints:

   $$\sum_i \pi(i) = 1$$

   yields the estimates of the initial probabilities:

   $$\hat{\pi}_i = \gamma(x(t_0) = i).$$

However, since t_0 is not observed, there is no fixed value for $\gamma(x(t_0) = i)$. Thus we take the expectation of $\gamma(x(t_0) = i)$ over all the observations to obtain:

$$\hat{\pi}_i = \frac{\sum_{k=1}^{T-1} \gamma(x(t_k) = i)}{T-1}.$$

2. **One-step Transition Matrix**

Maximization of $\mathbf{E}[\sum_{k=1}^{T-1} \log a(\Delta_k; x(t_k), x(t_{k+1}))|\mathbf{y}]$ with respect to a_{ij}, subject to the constraints:

$$\sum_j a_{ij} = 1$$

yields the updates for the transition probabilities. Let λ_i denote the Lagrange multipliers for the constraints:

$$-\sum_i \lambda_i \left(\sum_j a_{ij} - 1 \right).$$

Recall the notation $\mathbf{Q} = \mathbf{A} - \mathbf{I}$. For small $\mathbf{Q}$, $\mathbf{A}^n \approx \mathbf{I} + n\mathbf{Q}$. Thus,

$$a(\Delta_k; i, j) = \begin{cases} \Delta_k a_{ij} & \text{if } i \neq j; \\ 1 + \Delta_k(a_{ii} - 1) & \text{if } i = j. \end{cases}$$

The goal is to maximize:

$$\begin{aligned}
&\sum_{k=1}^{T-1} \left[\sum_i \sum_j \xi(x(t_k) = i, x(t_{k+1}) = j) \log a(\Delta_k; i, j) \right] - \sum_i \lambda_i \left(\sum_j a_{ij} - 1 \right) \\
=&\sum_{k=1}^{T-1} \left[\sum_i \sum_{j:i \neq j} \xi(x(t_k) = i, x(t_{k+1}) = j) \log(\Delta_k a_{ij}) \right] \\
&+ \sum_{k=1}^{T-1} \left[\sum_i \xi(x(t_k) = i, x(t_{k+1}) = i) \log(1 + \Delta_k(a_{ii} - 1)) \right] \\
&- \sum_i \lambda_i \left(\sum_j a_{ij} - 1 \right)
\end{aligned}$$

To obtain the MLEs of a_{ij}, set the partial derivatives to zero, while considering the cases of $i \neq j$ and $i = j$ separately. For the cases of $i \neq j$,

$$\frac{\partial}{\partial a_{ij}} : \sum_{k=1}^{T-1} \left[\sum_i \sum_j \xi(x(t_k) = i, x(t_{k+1}) = j) \frac{\Delta_k}{\Delta_k a_{ij}} \right] - \lambda_i = 0;$$

$$\therefore \hat{a}_{ij} = \frac{1}{\lambda_i} \sum_{k=1}^{T-1} \xi(x(t_k) = i, x(t_{k+1}) = j).$$

For the cases of $i = j$,

$$\frac{\partial}{\partial a_{ii}} : \sum_{k=1}^{T-1} \left[\sum_i \xi(x(t_k) = i, x(t_{k+1}) = i) \frac{\Delta_k}{1 + \Delta_k(a_{ii} - 1)} \right] - \lambda_i = 0;$$

$$\therefore \hat{\lambda}_i = \sum_{k=1}^{T-1} \xi(x(t_k) = i, x(t_{k+1}) = i) \frac{\Delta_k}{1 + \Delta_k(\hat{a}_{ii} - 1)},$$

where $\hat{a}_{ii}$ is estimated from the previous iteration.

The estimates of a_{ii} can be obtained from the probability constraints, as given below.

$$\hat{a}_{ii} = 1 - \sum_{j:i \neq j} \hat{a}_{ij}$$

The conventional homogeneous HMM corresponds to the special case of $\Delta_k = 1$ for $k \in \{1, ..., T\}$. The estimate of a_{ii} in this special case is derived below.

$$\frac{\Delta_k}{1 + \Delta_k(a_{ii} - 1)} = \frac{1}{a_{ii}};$$

$$\therefore \hat{a}_{ii} = \frac{1}{\lambda_i} \sum_{k=1}^{T-1} \xi(x(t_k) = i, x(t_{k+1}) = i).$$

The expression for $\hat{\lambda}_i$ can be further simplified, starting from the constraints $\sum_j a_{ij} = 1$.

$$\sum_j a_{ij} = \sum_{j:i \neq j} \frac{1}{\lambda_i} \sum_{k=1}^{T-1} \xi(x(t_k) = i, x(t_{k+1}) = j) + \frac{1}{\lambda_i} \sum_{k=1}^{T-1} \xi(x(t_k) = i, x(t_{k+1}) = i);$$

$$1 = \frac{1}{\lambda_i} \sum_j \sum_{k=1}^{T-1} \xi(x(t_k) = i, x(t_{k+1}) = j);$$

$$\therefore \hat{\lambda}_i = \sum_{k=1}^{T-1} \sum_j \xi(x(t_k) = i, x(t_{k+1}) = j) = \sum_{k=1}^{T-1} \gamma(x(t_k) = i).$$

This expression agrees with the result of the homogeneous case described by Rabiner [53].

3. **Emission Parameters**
   Maximization of $\mathbf{E}[\sum_{k=1}^{T-1} \log p(y(t_{k+1})|x(t_{k+1}))|\mathbf{y}]$ with respect to (μ_i, σ_i^2) yields the estimates of the emission parameters. Assuming σ_i^2 is constant and non-zero, differentiation with respect to μ_i leads to the following equation.

$$\sum_{k=1}^{T-1} \gamma(x(t_k) = i) \frac{y(t_k) - \mu_i}{\sigma_i^2} = 0$$

   Thus,

$$\hat{\mu}_i = \frac{\sum_{k=1}^{T-1} \gamma(x(t_k) = i) y(t_k)}{\sum_{k=1}^{T} \gamma(x(t_k) = i)}.$$

   Differentiation with respect to σ_i leads to the following equation.

$$\sigma_i \sum_{k=1}^{T-1} \gamma(x(t_k) = i) \left\{ \frac{[y(t_k) - \hat{\mu}_i]^2}{\sigma_i^2} - 1 \right\}$$

   Thus,

$$\hat{\sigma}_i^2 = \frac{\sum_{k=1}^{T-1} \gamma(x(t_k) = i)[y(t_k) - \hat{\mu}_i]^2}{\sum_{k=1}^{T} \gamma(x(t_k) = i)}.$$

Notice that $\hat{\mu}_i$ and $\hat{\sigma}_i^2$ are simply weighted versions of the sample mean and the sample variance, respectively. The weighting depends on the posterior probabilities of the hidden states at the observed base positions.

3.2.4 Initialization of the parameter estimates

The Baum-Welch algorithm requires reasonable estimates of the parameters as the initial values. A rather long-winded approach is to create a set of binned data by converting the different tiling array designs into a common pseudo-design. Then, the initial estimates could be obtained by running a conventional peak-calling algorithm on the binned data. In practice, we found that the following procedure leads to reasonable initial values, and requires minimal computation.

1. Estimate μ_1 and σ_1^2 using observations in the 90th to 100th percentile of the each experiment. Estimate μ_0 and σ_0^2 using the remaining observations. If two or more replicate experiments were performed using the same tiling array design, then the estimates obtained from the different replicates are averaged.

2. Assign each position to either State 0 or State 1, depending on whether the observation exceeds the 90th percentile of all the observations collected using same tiling array design.

3. Estimate the initial probabilities based on the abundance of each state.

$$\hat{\pi}_1 = \frac{\text{number of positions assigned to State 1}}{\text{total number of positions in the integrated data set}}$$

4. Consider all the positions assigned to State 1. If the gap between two neighboring positions is within 50 bases, then the two positions are joined together in the same interval. Thus a set of State 1 intervals are obtained. Similarly, obtain a set of State 0 intervals by joining neighboring positions that are assigned to State 0.

5. Estimate the transitions probabilities of the two-state HMM as follows.

$$\hat{a}_{11} = 1 - \frac{1}{\text{average length of the State 1 intervals}}$$

$$\hat{a}_{00} = 1 - \frac{1}{\text{average length of the State 0 intervals}}$$

3.3 Simulation study for one protein

A two-state nonhomogeneous HMM is used to model the ChIP-chip data of a single protein. At any given chromosomal position, the hidden states are either bound or unbound to DNA. The algorithm for fitting this model, described in Section 3.2, involves a linear approximation of the one-step transition matrix. We did a simulation study to check the conditions under which the algorithm performs sufficiently. The goal of the simulations was not to validate the assumptions of the nonhomogeneous HMM. Instead, the goal of the simulations was to examine how the linear approximation might affect inferences about the hidden states. Thus we used the classification rate for each state as the ultimate metric for assessing the performance of our algorithm.

We selected a 1 MB region on Chromosome X for the simulation study. The chromosomal coordinates of the probes in this region were recorded for the three tiling array designs. These coordinates dictate where the observations are emitted. The hidden states of the protein were generated for every base position in this region, according to a Markov chain. The length of the Markov chain is 1 million bases. Following the notations introduced in Section 3.2, let π and $\mathbf{A}$ denote the initial probabilities and the one-step transition matrix of the Markov chain, respectively. Tiling array data were simulated according to the state-conditional Gaussian distributions. Let μ_0 and σ_0^2 denote the mean and variance parameters for emissions in the unbound state. These parameters have fixed values for all probes in the unbound state. Let μ_1 and σ_1^2 denote the mean and variance parameters for emissions in the bound state. To simulate peaks of variable lengths, a different value of the mean parameter μ_1 was chosen for each peak. The variance parameter σ_1^2 was fixed at the same value for all peaks. Let μ_1^* denote another parameter with a fixed value. To determine the emission parameters at each position, the following rules were applied.

1. If the state of the current position is 0 (unbound), then use the emission parameters μ_0 and σ_0^2.

2. If the state of the previous position is 0 (unbound), but the state of the current position is 1 (bound), then choose a new value for the mean parameter of the new peak.

$$\mu_1 \sim \text{Uniform}(\mu_1^* - 0.5 \times \sigma_1, \mu_1^* + 0.5 \times \sigma_1)$$

   Notice that the expected value of μ_1 is μ_1^*. Choose the variance parameter σ_1^2, which has a fixed value for all peaks.

3. If the state of the previous position is 1 (bound), and the state of the current position is 1 (bound), then continue to generate observations using the same emission parameters as those used for the previous position.

The values of μ_0, σ_0^2, μ_1^* and σ_1^2 depend on the tiling array design. The probe design also dictates whether an observation is emitted at any particular base position. To emulate the setting of the wild type Dpy-27 data, we generated 3 replicates in each simulated data set, with one replicate coming from each design. Under this setting, we simulated 100 data sets using the following parameters. These parameters were obtained by fitting the nonhomogeneous HMM to the wild type Dpy-27 data for the selected 1 MB region on Chromosome X.

- Emission Parameters for Design 1

$$\begin{aligned} \mu_0 &= 0.124 & \sigma_0 &= 0.152 \\ \mu_1 &= 0.657 & \sigma_1 &= 0.195 \end{aligned}$$

- Emission Parameters for Design 2

$$\begin{aligned} \mu_0 &= 0.170 & \sigma_0 &= 0.128 \\ \mu_1 &= 0.675 & \sigma_1 &= 0.178 \end{aligned}$$

- Emission Parameters for Design 3

$$\begin{aligned} \mu_0 &= 0.053 & \sigma_0 &= 0.121 \\ \mu_1 &= 0.357 & \sigma_1 &= 0.161 \end{aligned}$$

- Markov Chain Parameters

$$\pi = (0.82, 0.18); \ \mathbf{A} = \begin{pmatrix} 0.9993 & 0.0007 \\ 0.0032 & 0.9968 \end{pmatrix}$$

Each simulated data set was analyzed using the algorithm described in Section 3.2. The parameter estimates obtained from the 100 simulations are summarized as boxplots in Figures 3.3 to 3.7. In each panel of these figures, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates. The variability of the parameter estimates results from simulation errors. Figures 3.3 to 3.5 show the emission parameters for each tiling array design separately. The estimates of μ_0, μ_1 and σ_0 fluctuate around their true values, as expected. The estimates of σ_1 are consistently larger than the true value of σ_1. This is because the simulation model is slightly more complex than the assumptions of the nonhomogeneous HMM. To generate peaks of variable heights, a different value of μ_1 was chosen randomly for each peak to simulate the tiling array data. Whereas a mixture of Gaussians was generated in the simulations, the nonhomogeneous HMM assumes only a single Gaussian distribution for the bound state. Thus the estimates of σ_1 were inflated. Figure 3.6 shows the initial distribution of the hidden states. The estimates of π_0 and π_1 fluctuate around their true values. Figure 3.7 shows the transition probabilities of the hidden Markov chain. There appears to be a bias in the estimation of the transition matrix, which deserves a closer examination.

Our algorithm uses a linear approximation of the matrix exponential, described in 3.2.1, to achieve a closed-form solution for the maximum likelihood estimate of the transition matrix. This approximation works better for smaller step sizes than larger step sizes. If the bias in the estimation of the transition matrix was due to errors in the linear approximation, then we should see a reduction in the bias when smaller step sizes are used in the simulations. Thus we repeated the simulations with progressively smaller spacing between the probes. Since the purpose is to investigate the effects of step sizes, we used a hypothetical design with uniform spacing between the probes in the next set of simulation experiments. Each simulated data set contained only one set of observations, instead of 3 replicates. For the first experiment, we set the spacing between adjacent probes to 20 bp. For the second experiment, we set the spacing to 10 bp. For the third experiment, we set the spacing to 2 bp. The computational complexity of the forward-backward algorithm is linear in the number of observations. In order to keep the running time within reasonable limits, we varied the length of the Markov chain to achieve the same number (50,000) of observations in each experiment. Figure 3.8 shows the results of the first experiment, with observations emitted at every 20 base pairs. Figure 3.9 shows the results of the second experiment, with observations emitted at every 10 base pairs. Figure 3.10 shows the results of the third experiment, with observations emitted at every other base pair. Indeed, the bias in the estimates was progressively reduced when we decreased the step sizes between the adjacent observations. The careful reader may also notice a progressive increase in the variance. This was merely a side-effect of simulating shorter Markov chains in the experiments with more densely placed probes. Since shorter chains contain fewer events of transitions, the transition probabilities estimated from shorter chains have higher variances.

Given that our algorithm produces slightly-biased estimates of the transition matrix, we then looked at how the bias might affect our inferences about the hidden states. For each

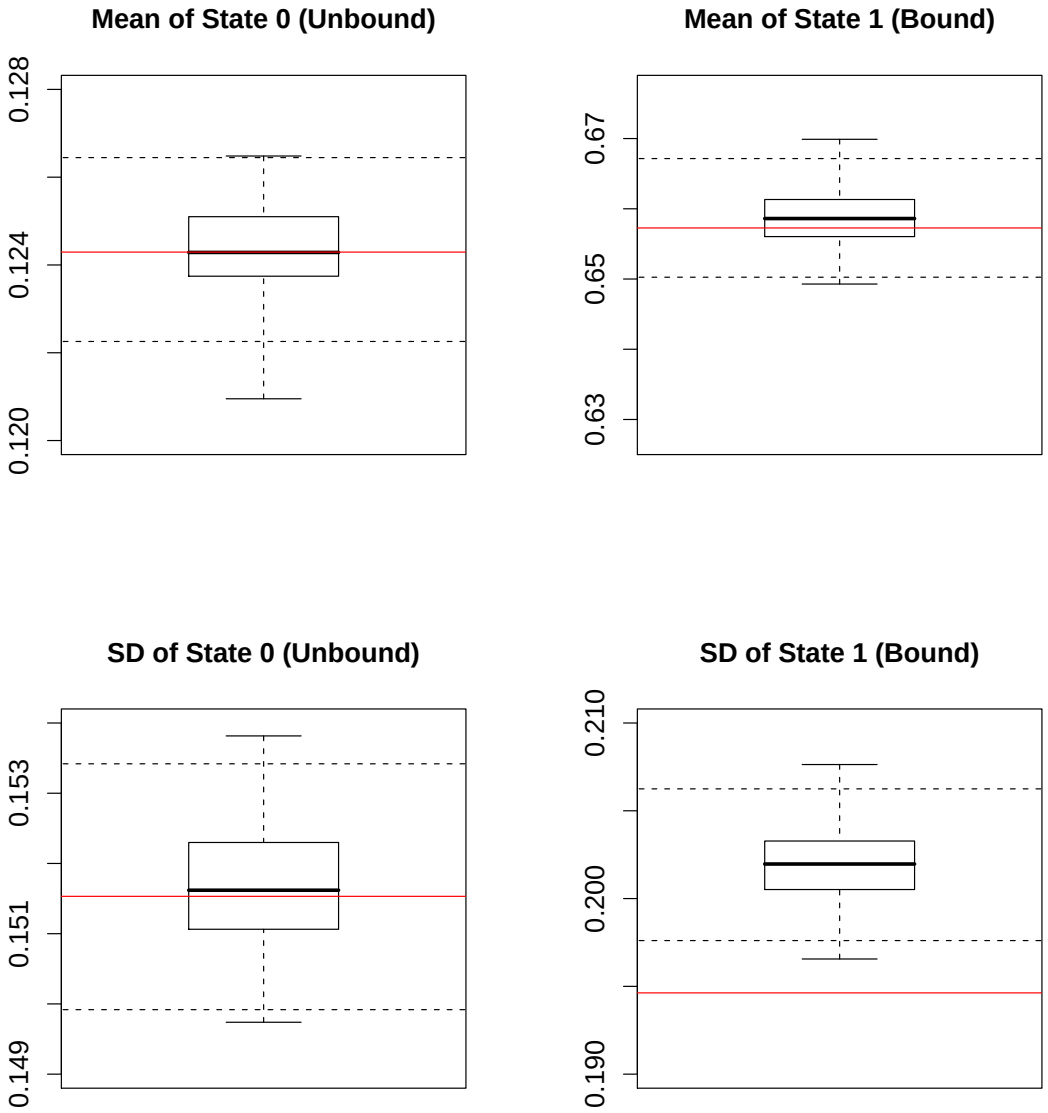

Figure 3.3: Emission Parameters (Design 1): 100 simulations of tiling array data were generated according to the chromosomal coordinates of the probes in a 1 MB region of Chromosome X. Each simulated data set contained one array of each design, summing up to 3 "replicates". Estimates of the emission parameters for Design 1 are summarized as boxplots. In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the estimates. The black dashed lines represent Mean +/− 2 SDs of the estimates.

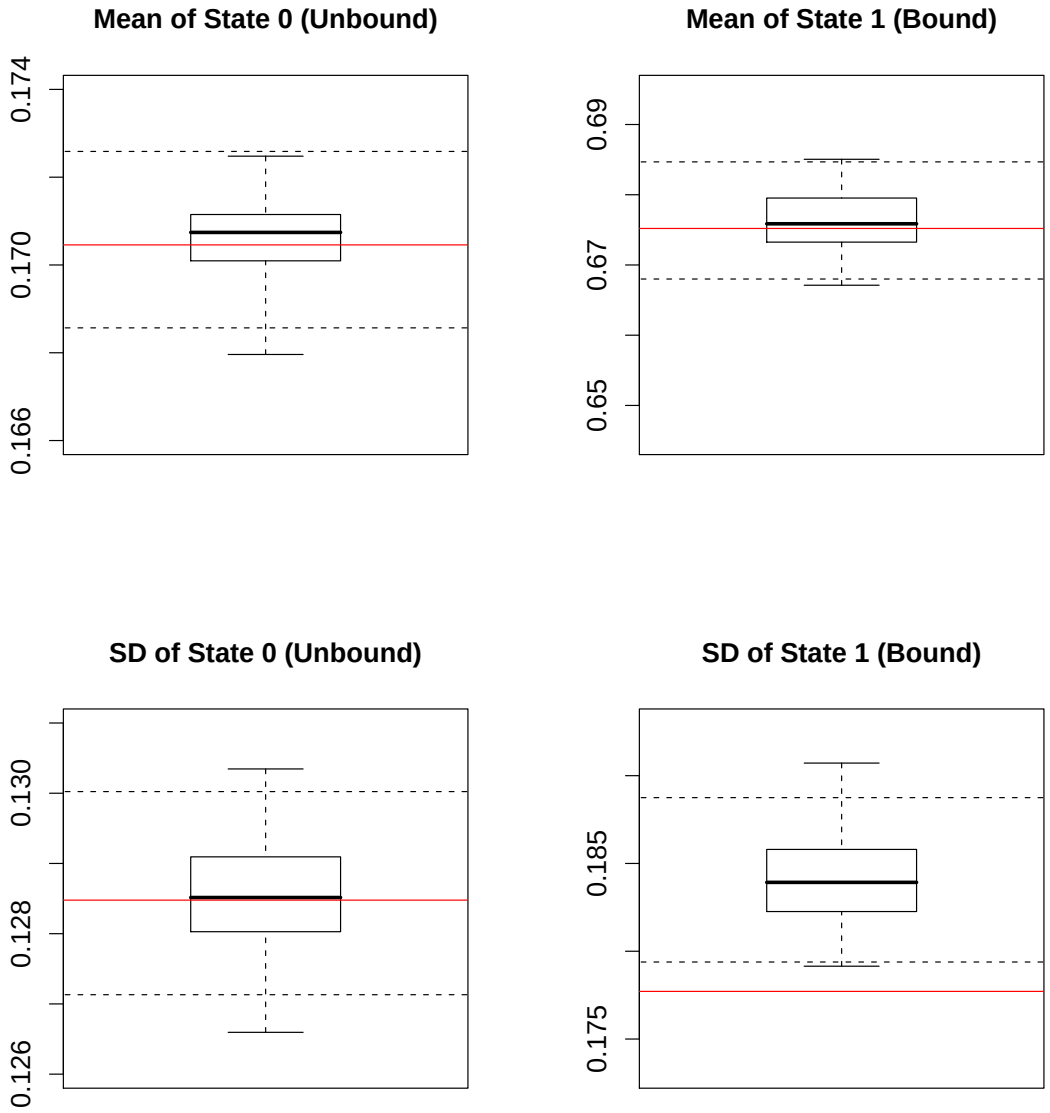

Figure 3.4: Emission Parameters (Design 2): 100 simulations of tiling array data were generated according to the chromosomal coordinates of the probes in a 1 MB region of Chromosome X. Each simulated data set contained one array of each design, summing up to 3 "replicates". Estimates of the emission parameters for Design 2 are summarized as boxplots. In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the estimates. The black dashed lines represent Mean +/− 2 SDs of the estimates.

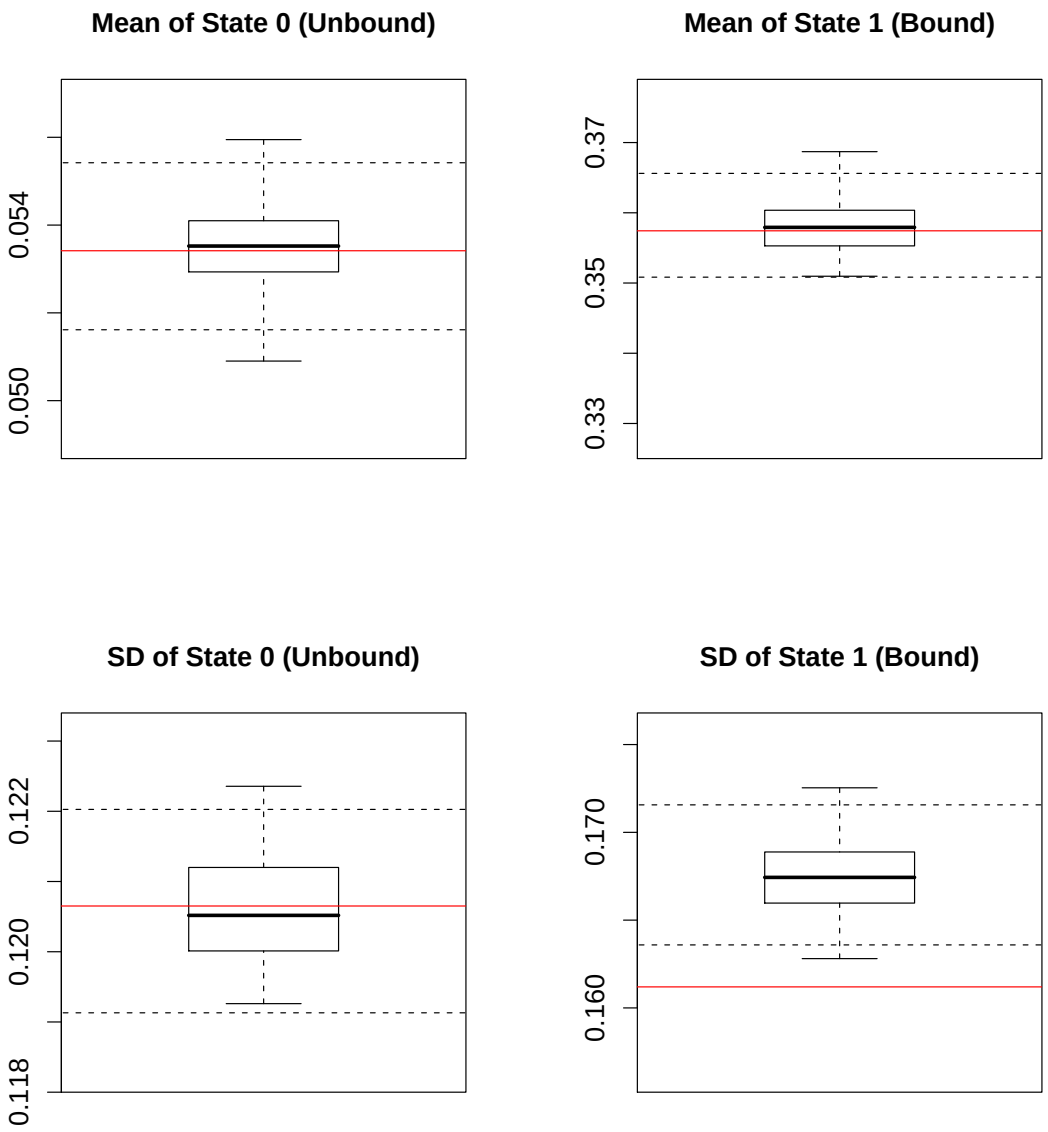

Figure 3.5: Emission Parameters (Design 3): 100 simulations of tiling array data were generated according to the chromosomal coordinates of the probes in a 1 MB region of Chromosome X. Each simulated data set contained one array of each design, summing up to 3 "replicates". Estimates of the emission parameters for Design 3 are summarized as boxplots. In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the estimates. The black dashed lines represent Mean +/− 2 SDs of the estimates.

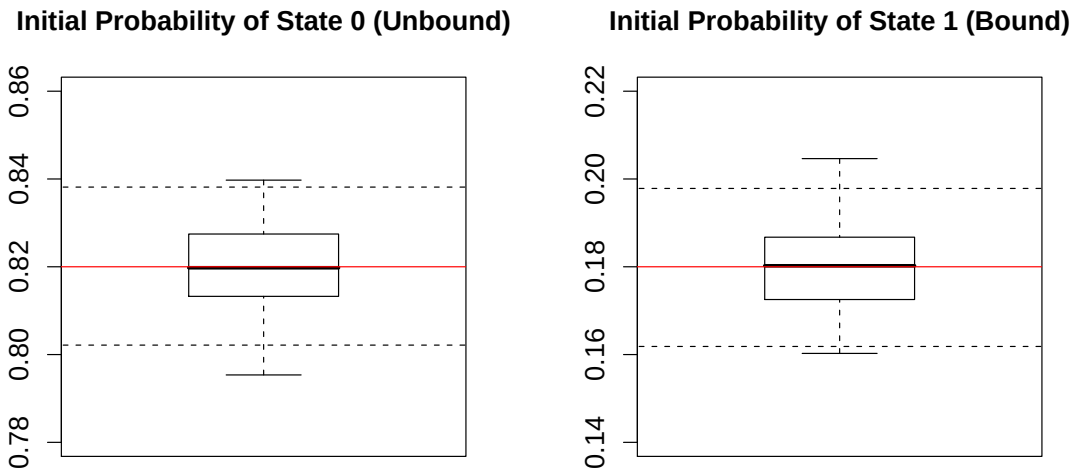

Figure 3.6: Stationary Distribution: 100 simulations of tiling array data were generated according to the chromosomal coordinates of the probes in a 1 MB region of Chromosome X. Each simulated data set contained 3 replicates, with one from each design. Estimates of the initial probabilities are summarized as boxplots. In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates.

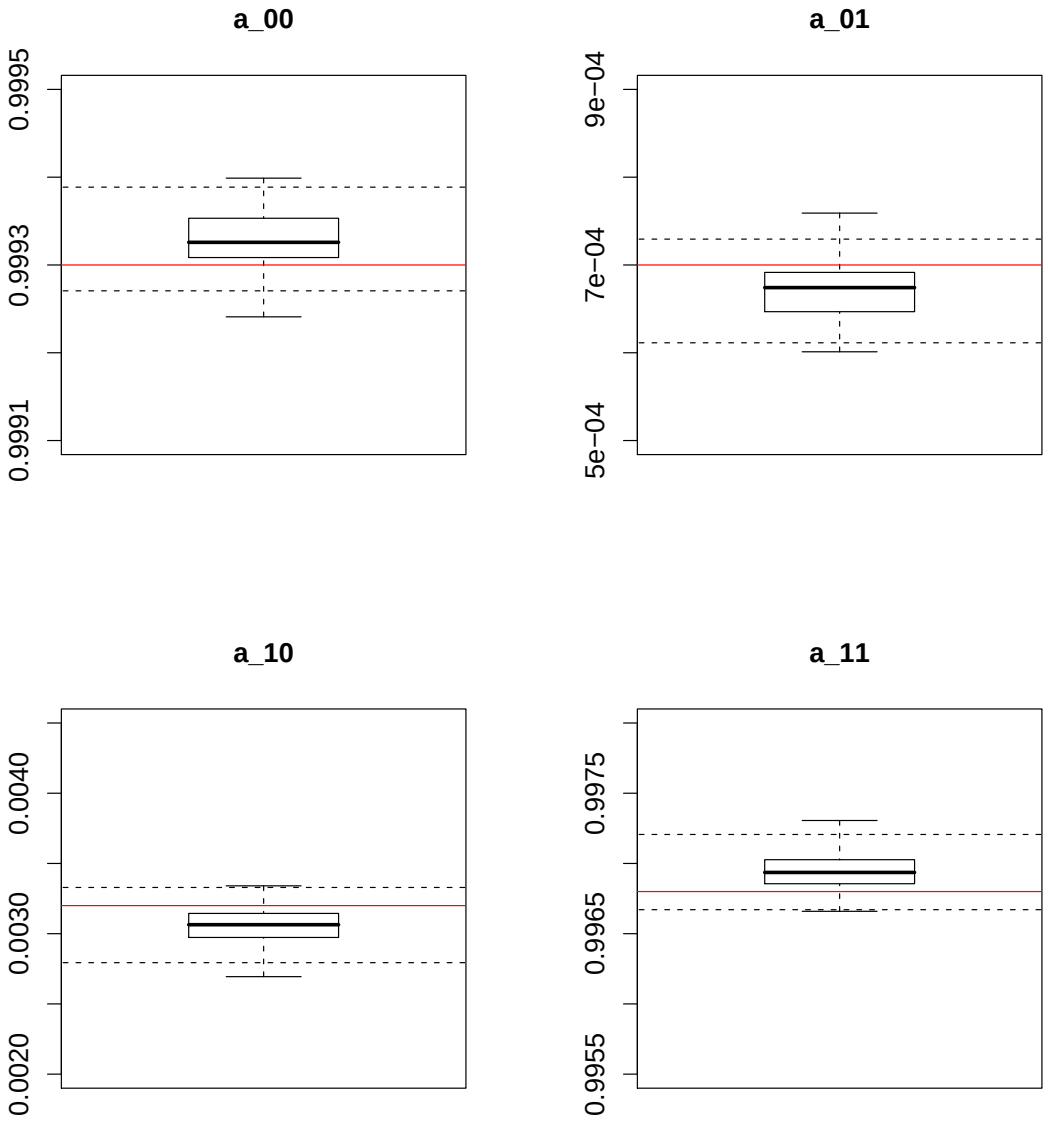

Figure 3.7: Transition Matrix (variable Δ_k): 100 simulations of tiling array data were generated according to the chromosomal coordinates of the probes in a 1 MB region of Chromosome X. Each simulated data set contained one array of each design, summing up to 3 "replicates". Estimates of the transition probabilities are summarized as boxplots. In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the estimates. The black dashed lines represent Mean +/− 2 SDs of the estimates.

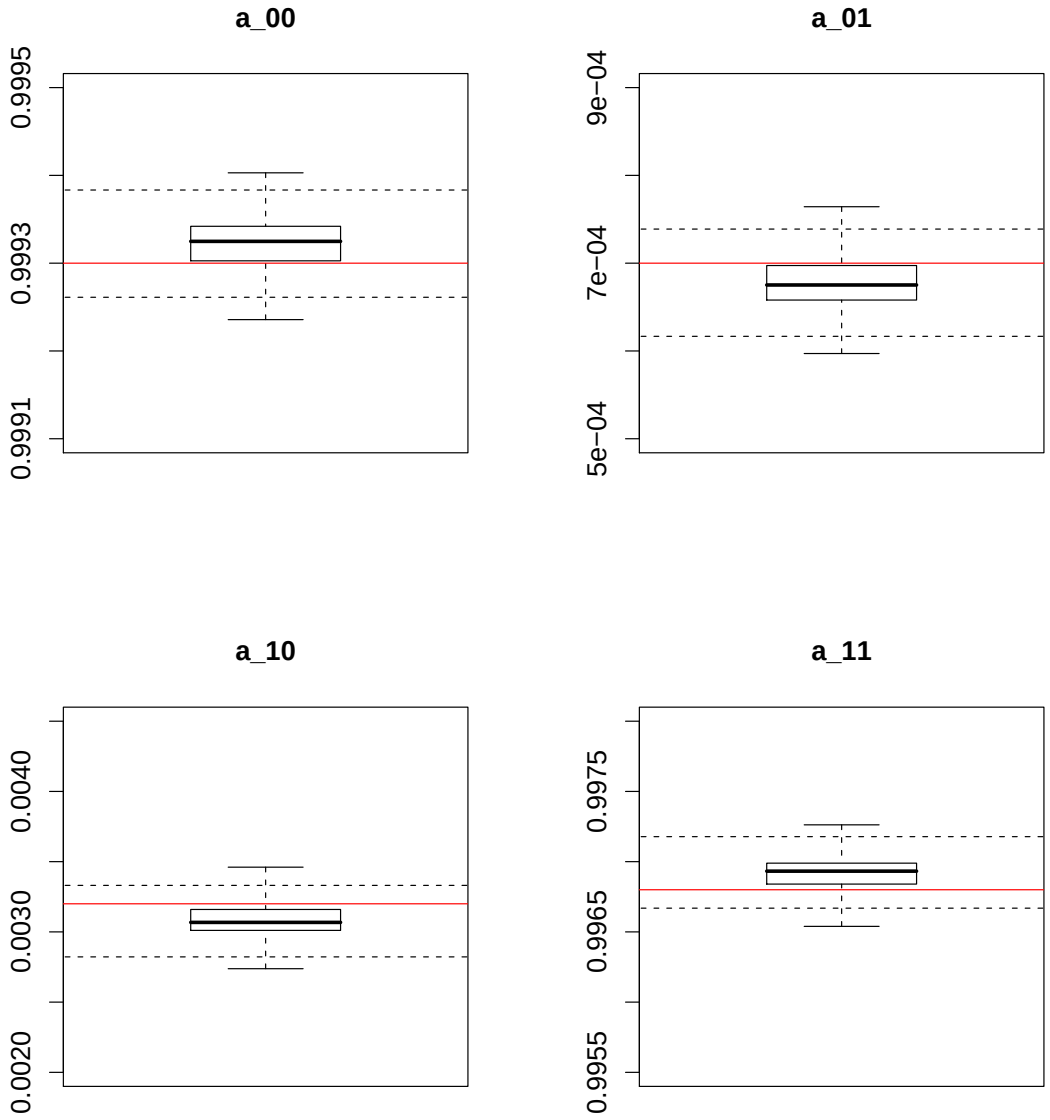

Figure 3.8: Transition Matrix ($\Delta_k = 20$): 100 simulations of tiling array data were generated according to a hypothetical design with probes placed at every 20 base pairs. Each simulated data set contained one array. Estimates of the transition probabilities are summarized as boxplots. In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates.

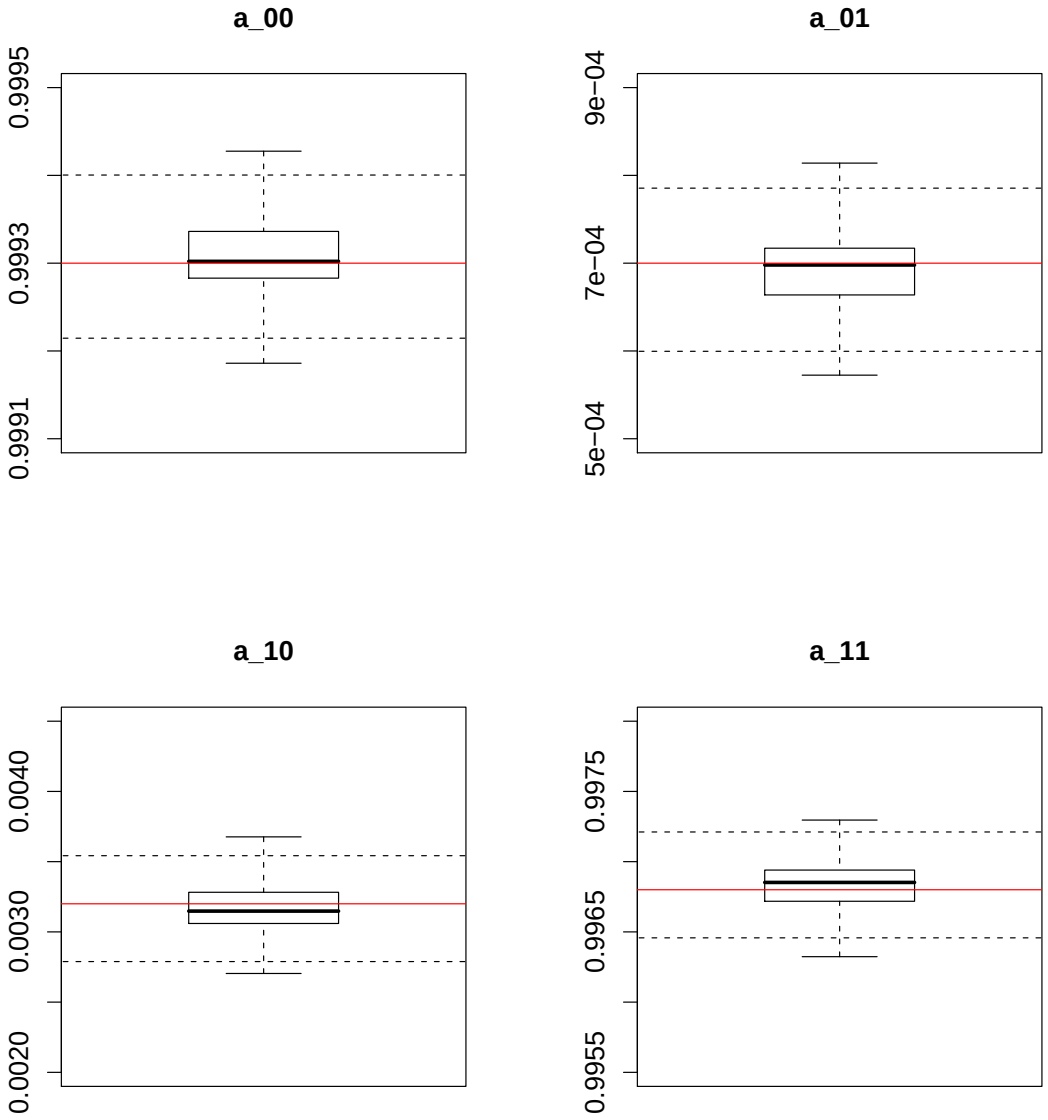

Figure 3.9: Transition Matrix ($\Delta_k = 10$): 100 simulations of tiling array data were generated according to a hypothetical design with probes placed at every 10 base pairs. Each simulated data set contained one array. Estimates of the transition probabilities are summarized as boxplots. In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates.

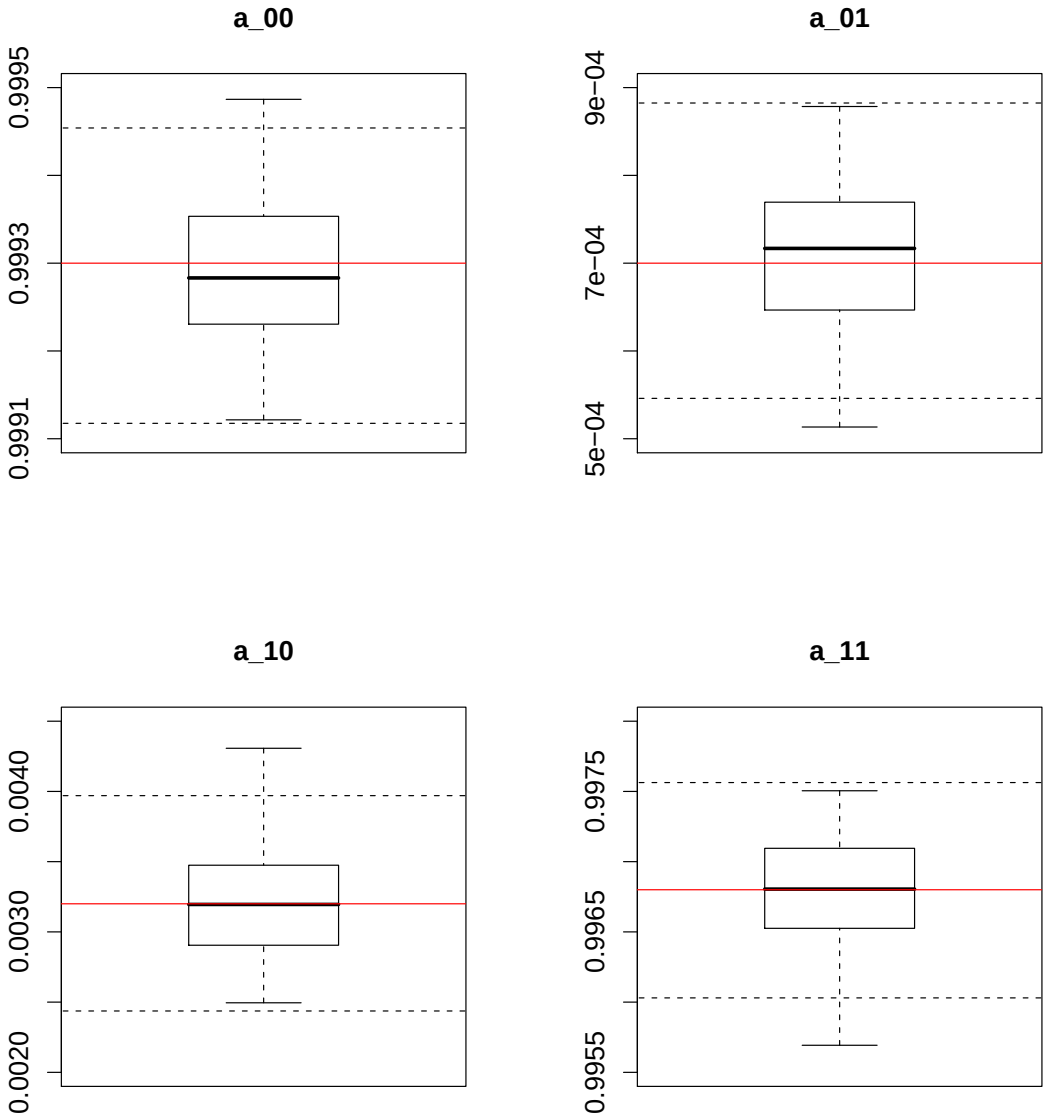

Figure 3.10: Transition Matrix ($\Delta_k = 2$): 100 simulations of tiling array data were generated according to a hypothetical design with probes placed at every other base pair. Each simulated data set contained one array. Estimates of the transition probabilities are summarized as boxplots. In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates.

simulated data set, we compared the true hidden states with the inferred states from our algorithm. For each simulated data set, we tabulated the number of correctly inferred positions and the number of incorrectly inferred positions in a confusion matrix. To account for differences in the state prevalences, we reported each entry as a percentage of the number of observations in each true state. To summarize across 100 simulations, we drew one boxplot for each entry in the confusion matrix. A good algorithm should lead to confusion matrices with the diagonal entries close to 100%, and the off-diagonal entries nearly zero. Figure 3.11 shows the results of the experiment with observations emitted at the same chromosomal coordinates as the real data. Figure 3.12 shows the results of the experiment with observations emitted at every 20 base pairs. Figure 3.13 shows the results of the experiment with observations emitted at every 10 base pairs. Figure 3.14 shows the results of the experiment with observations emitted at every other base pair. As the spacing between the observations was decreased, the error rates also decreased. These results suggest that the errors in state inferences are associated with the bias in the estimation of the transition matrix. Under the setting of the real data, the error rates were below 2% in all of the 100 simulations. Thus we conclude that the nominal bias in the estimation of the transition matrix is tolerable.

3.4 From candidate probes to binding peaks

The ultimate goal of a ChIP-chip analysis is to identify chromosomal regions that are directly targeted by DNA binding proteins. When the probe level signals are plotted against their chromosomal positions, a binding target should exhibit the characteristic shape of a triangular peak. The signals are the strongest at the center of the peak, and taper off near the ends. Although the nonhomogeneous hidden Markov Model provides a way to estimate the probability that a probe lies within a bound region, it is insufficient to conclude the analysis at the probe level. In order to draw inferences about the binding sites, some postprocessing of the probe level information is required. A number of the existing methods including TAS, TiMAT and MA2C take the following approach. First, the candidate probes are identified by applying a threshold to the probe level summary statistics. Then, the candidate probes are joined into intervals according to some criteria. A couple of parameters called *max gap* and *min run* modulate the formation of the intervals. Enriched probe positions separated by a distance of up to *max gap* are joined together to form an interval. Any interval with a length of less than *min run* is rejected. The remaining intervals are called peaks by the other methods. We adopted the following criteria for joining probes into intervals, but we refrain from calling the intervals peaks at this point.

1. A probe is considered as a candidate probe if its posterior probability of being in the bound state is at least 0.8.

2. The maximum gap allowed between adjacent candidate probes within any interval is 300 bp.

3. The minimum length required for any interval 500 bp.

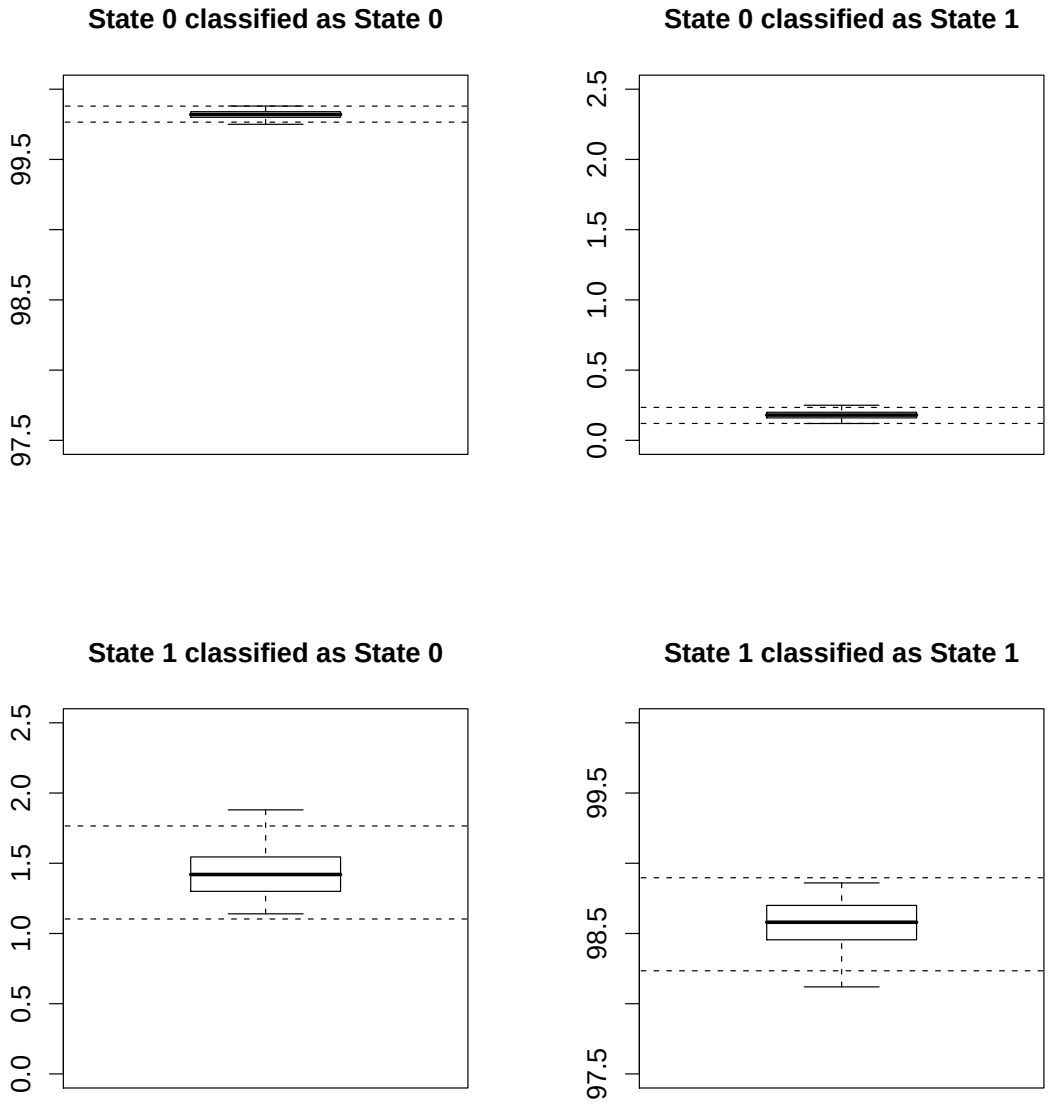

Figure 3.11: Confusion Matrix (variable Δ_k): 100 simulations of tiling array data were generated according to the chromosomal coordinates of the probes in a 1 MB region of Chromosome X. Each simulated data set contained one array of each design, summing up to 3 "replicates". The inferred states of the observed positions were compared against the true states. A confusion matrix was tabulated for each simulated data set, and its entries were normalized into percentages of the true states. Each panel summarizes one entry in the confusion matrix across 100 simulations. The inner-quartile range is delineated by the borders of the box, with a thick line drawn at the median. The whiskers extend to the extreme values. The black dashed lines represent Mean +/− 2 SDs.

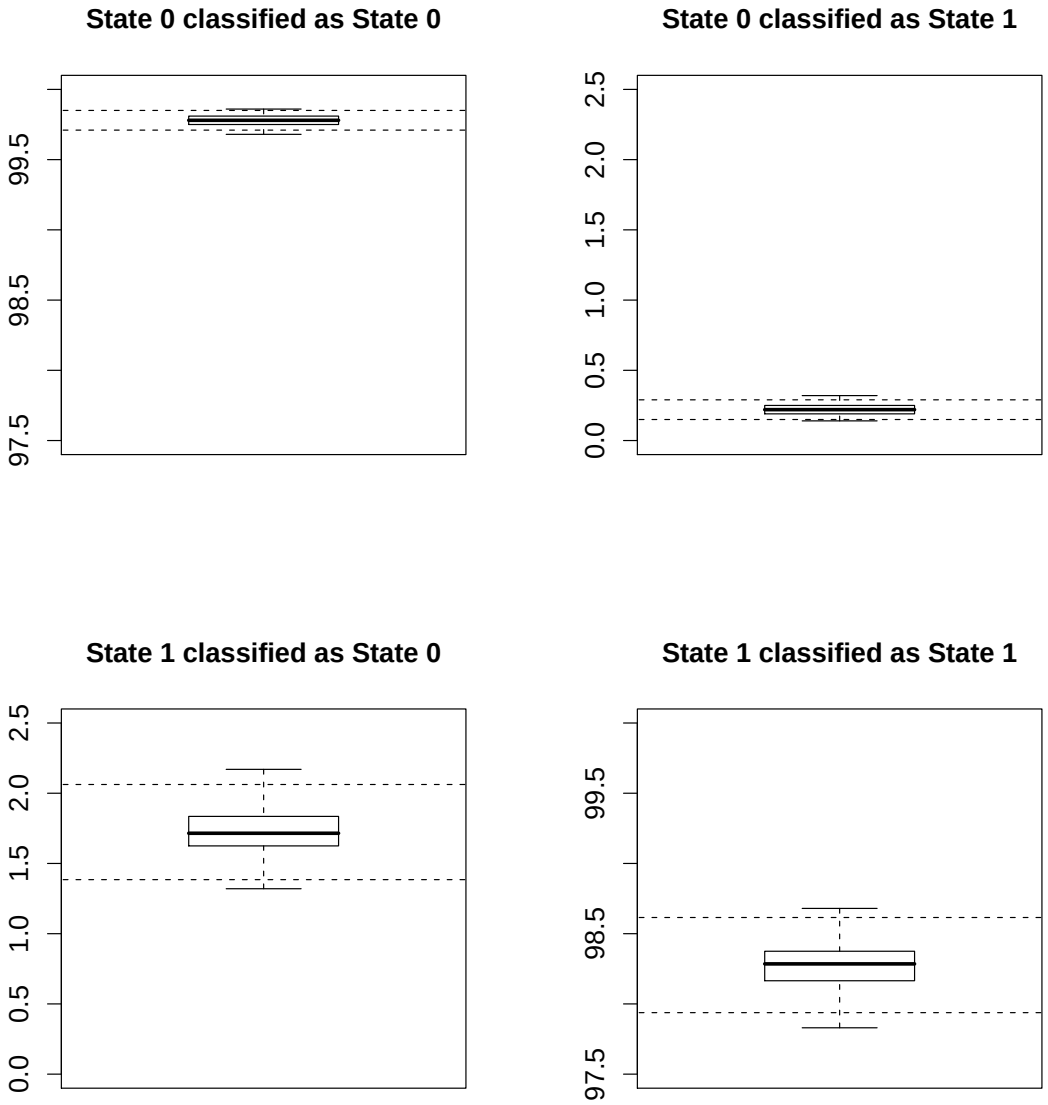

Figure 3.12: Confusion Matrix ($\Delta_k = 20$): 100 simulations of tiling array data were generated according to a hypothetical design with probes placed at every 20 base pairs. Each simulated data set contained one array. The inferred states of the observed positions were compared against the true states. A confusion matrix was tabulated for each simulated data set, and its entries were normalized into percentages of the true states. Each panel summarizes one entry in the confusion matrix across 100 simulations. The inner-quartile range is delineated by the borders of the box, with a thick line drawn at the median. The whiskers extend to the extreme values. The black dashed lines represent Mean $+/-$ 2 SDs.

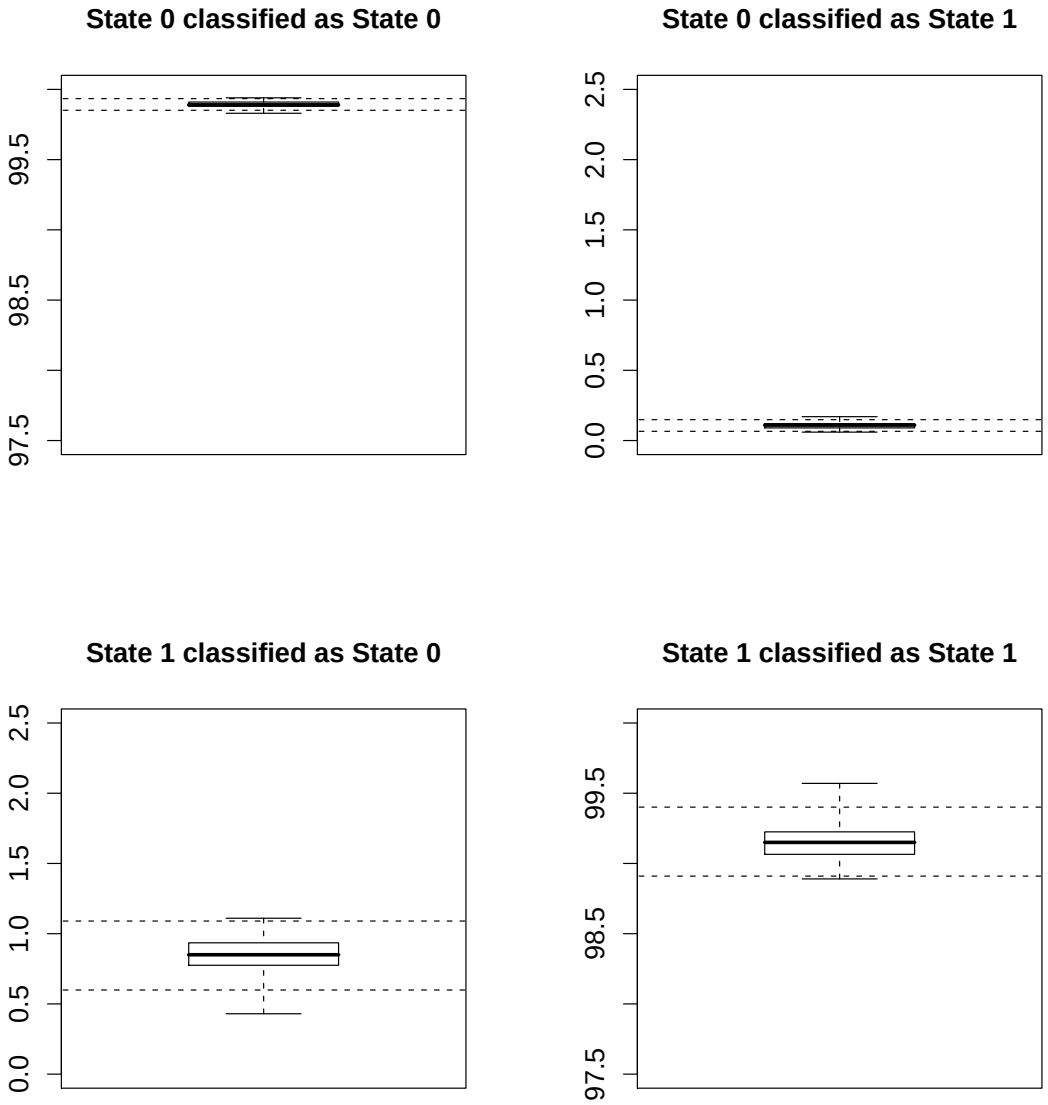

Figure 3.13: Confusion Matrix ($\Delta_k = 10$): 100 simulations of tiling array data were generated according to a hypothetical design with probes placed at every 10 base pairs. Each simulated data set contained one array. The inferred states of the observed positions were compared against the true states. A confusion matrix was tabulated for each simulated data set, and its entries were normalized into percentages of the true states. Each panel summarizes one entry in the confusion matrix across 100 simulations. The inner-quartile range is delineated by the borders of the box, with a thick line drawn at the median. The whiskers extend to the extreme values. The black dashed lines represent Mean +/− 2 SDs.

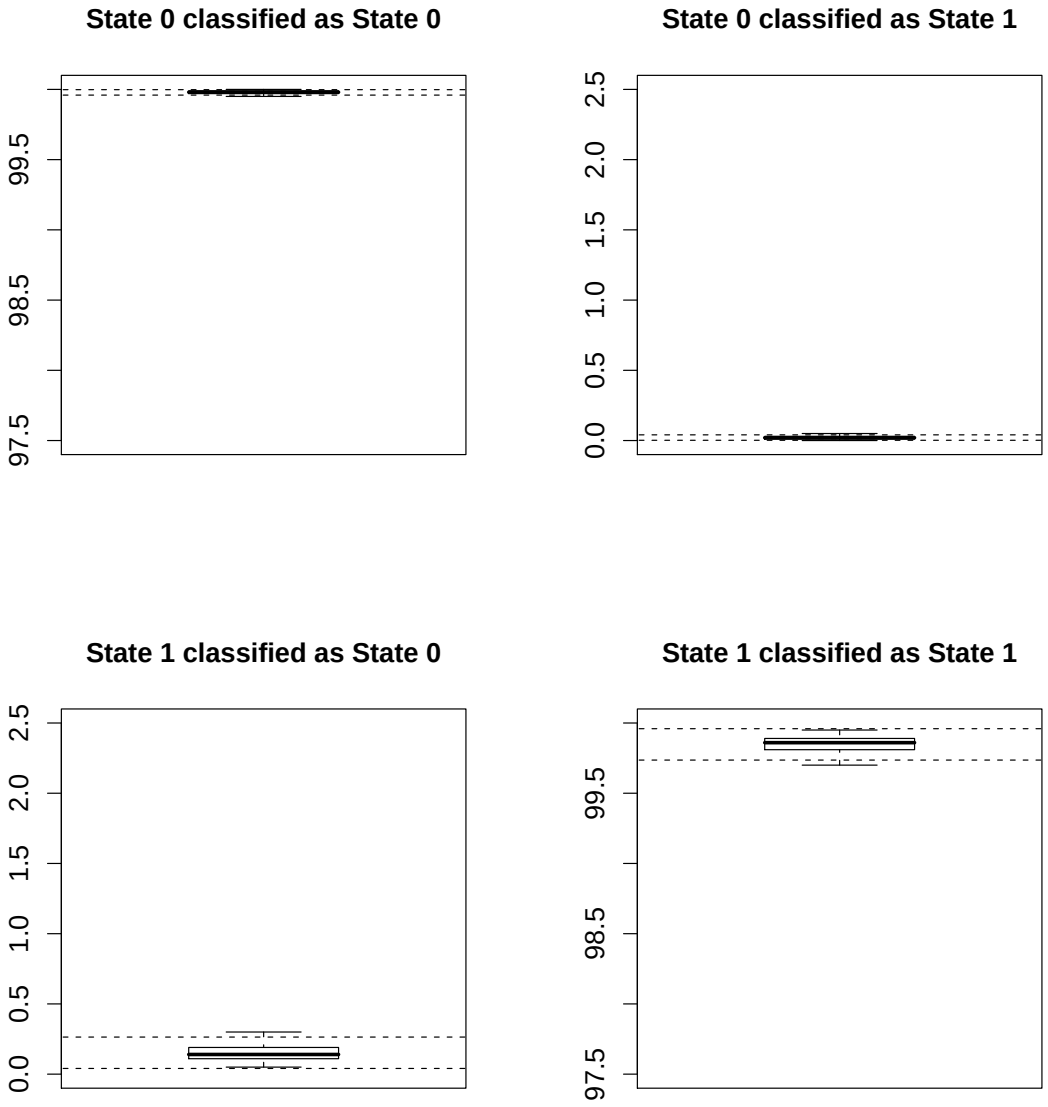

Figure 3.14: Confusion Matrix ($\Delta_k = 2$): 100 simulations of tiling array data were generated according to a hypothetical design with probes placed at every other base pair. Each simulated data set contained one array. The inferred states of the observed positions were compared against the true states. A confusion matrix was tabulated for each simulated data set, and its entries were normalized into percentages of the true states. Each panel summarizes one entry in the confusion matrix across 100 simulations. The inner-quartile range is delineated by the borders of the box, with a thick line drawn at the median. The whiskers extend to the extreme values. The black dashed lines represent Mean +/− 2 SDs.

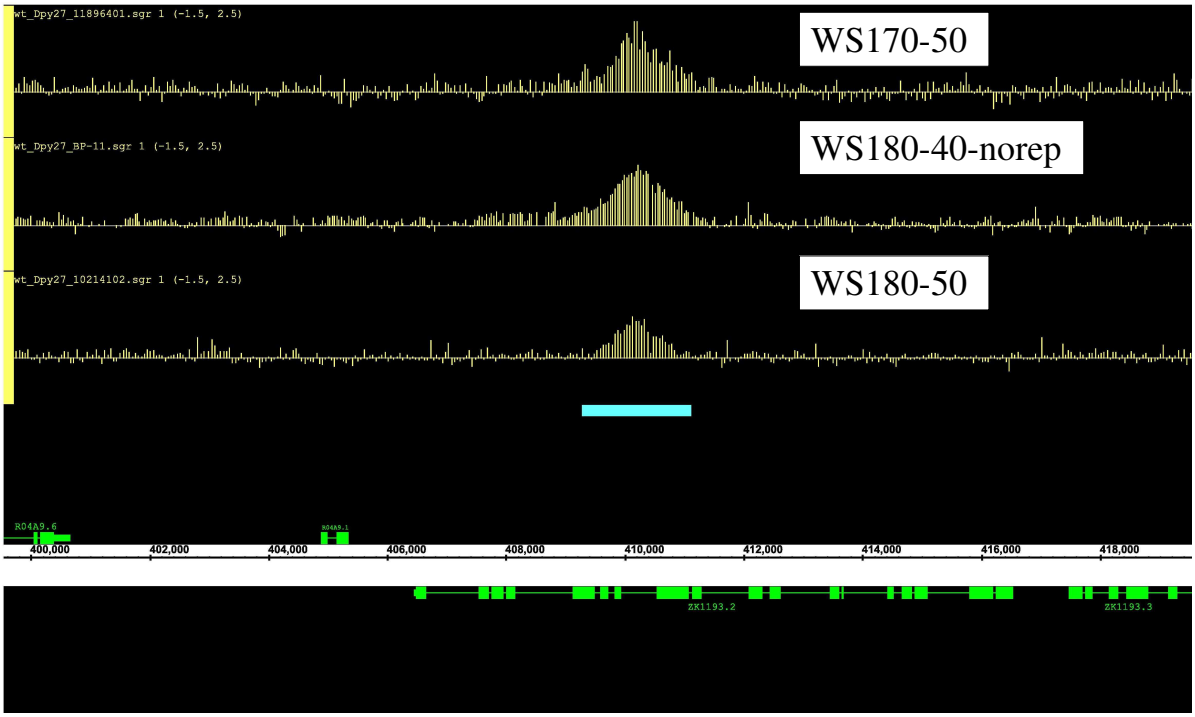

Figure 3.15: An ideal symmetric triangular peak.

4. The minimum density of probes within any interval is 1 probe per 100 bp.

The last criterion was added to avoid calling peaks in regions that have very sparse observations. Visual inspection of the data on a genome browser reveals that, while some intervals appear to be real binding sites, others are likely to be false positives. For example, taller peaks are more likely to be real than shorter ones, because signal strength is related to binding affinity. Peaks with the characteristic triangular shape are more likely to be real than regions that exhibit the rectangular shape. Figures 3.15 and 3.16 illustrate the distinction between triangular and rectangular "peaks". Rectangular intervals occur as the result of promiscuous associations between the protein complex and various DNA segments during chromatin immunoprecipitation. This phenomenon is particularly prevalent in the DCC experiments, because the dosage compensation complex is both large and abundant. Moreover, not every real binding site has the ideal shape of a symmetric triangle. Sometimes, the peak may be off-centered with a long tail. In other cases, two or more nearby binding sites may be merged into a multi-modal peak. Figure 3.17 illustrates these types of peaks, which are also very likely to contain real binding sites. Clearly, there is a need to rank the intervals according to the properties of real binding sites. Intervals that pass a filter on the ranking criterion can then be called peaks, which represent the putative binding sites.

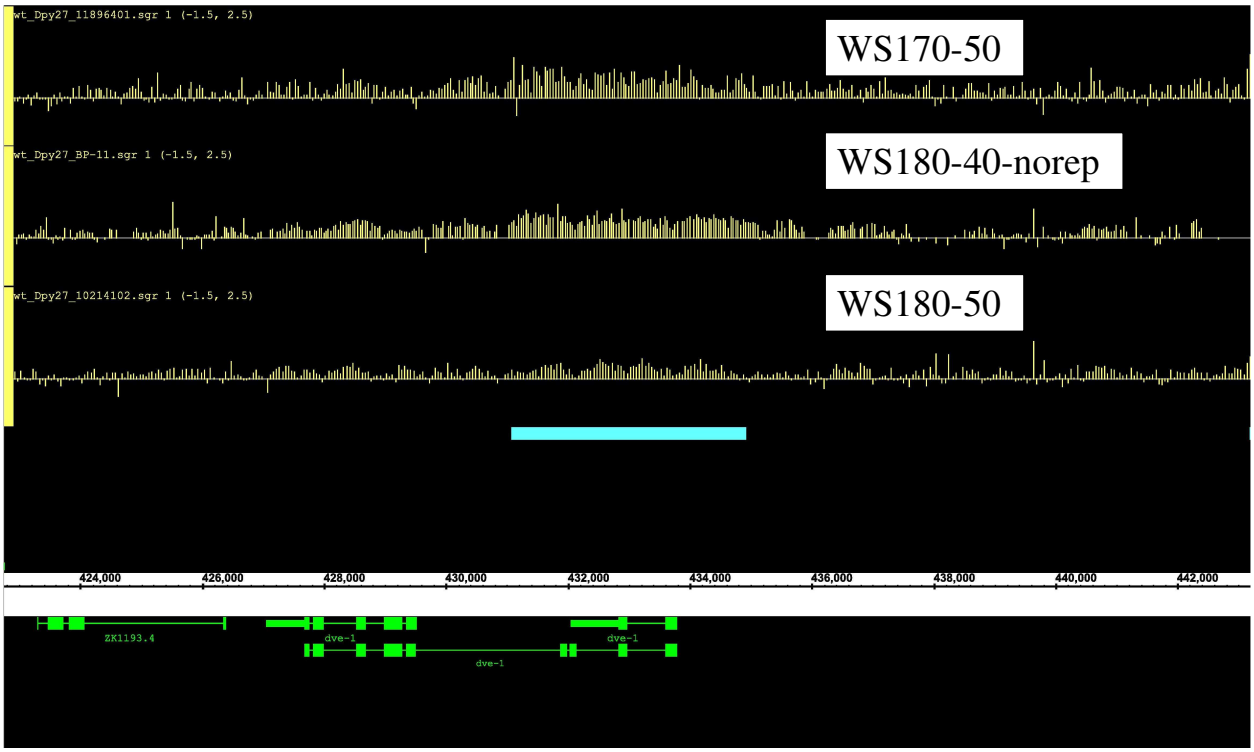

Figure 3.16: A false positive rectangular interval.

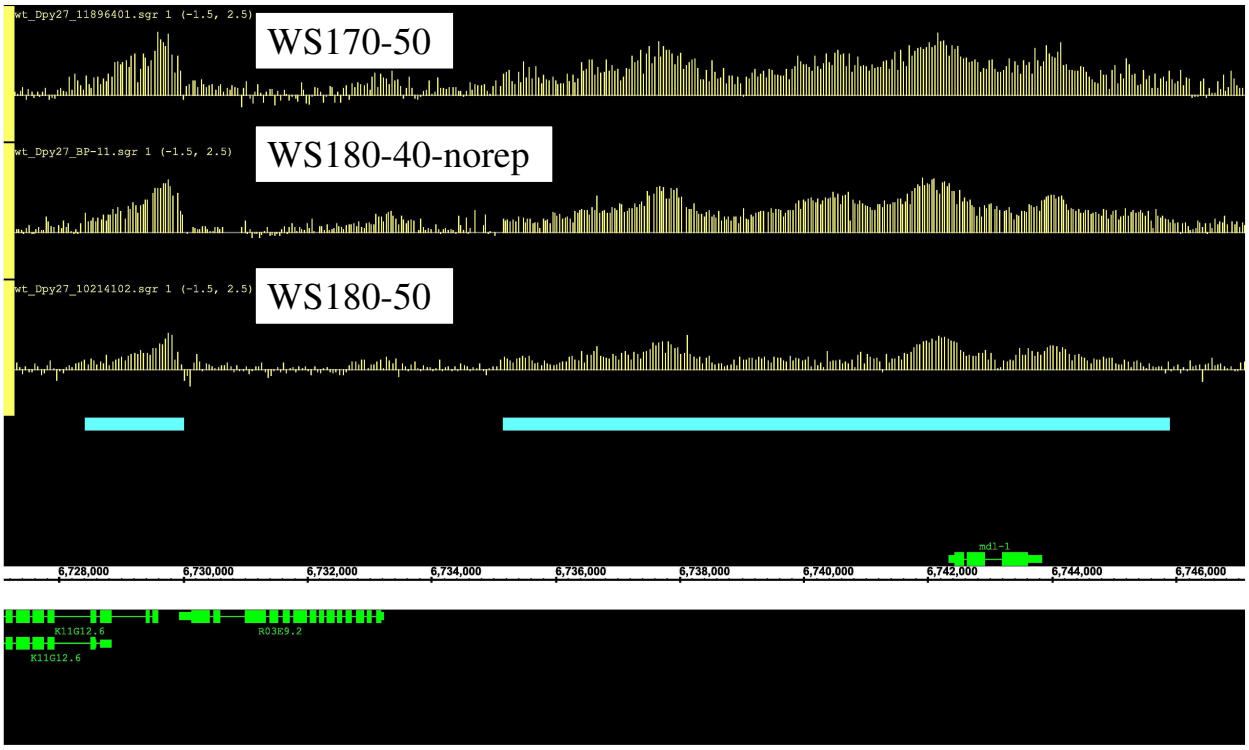

Figure 3.17: Non-ideal peaks. The first peak has the shape of an asymmetric triangle. The second peak is the product of multiple nearby binding sites being merged together. Both peaks are probably real binding sites.

Among the existing ChIP-chip analysis methods, the only one that uses the shape information is MPeak by Zheng et al. [70]. This method begins by identifying local maximum probes in the data. It then fits a truncated triangle shape regression model within a window centered at each local maximum probe. The slope of the left and right legs of the triangle are represented by two separate coefficients, which are allowed to be different for each peak. The start and end positions of the window can be chosen from a range of values, so a number of models are fitted using different combinations of these values. The model that has the smallest residual variance determines the best fitted start and end positions of the peak centered at this probe. This process of fitting truncated triangles is repeated for the neighbors of the local maximum probe. Among the local maximum probe and its neighbors, the probe with the smallest residual variance is chosen to identify the best-fitted triangle. MPeak was developed for a promoter study involving the ChIP-chip of some transcription factors and RNA polymerase II [40]. The truncated triangle model was suitable because this type of data generally have well separated peaks. However, the dosage compensation protein complex binds to the X-chromosome in a much higher density, to achieve a chromosome-wide attenuation of gene expression. So the ChIP-chip data of the dosage compensation proteins contain a substantial number of merged peaks, as seen in Figure 3.17. This special feature of the DCC data makes it difficult to separate the real peaks from the false positives solely based on the shape of isolated triangles. Thus we needed a more flexible method for ranking the peaks.

In order to develop a method that separates the real binding sites from the experimental artifacts, a set of pre-defined positives and negatives is required. Since the true binding sites are unknown, we curated a set of "standards" through visual inspection of the tiling array data on a genome browser. For the wild type data, 300 positions were selected randomly from Chromosome X and 600 positions were selected randomly from the other five autosomes. For the mutant data, 900 positions were selected randomly from the entire genome. We manually inspected the 10 kb window centered at each selected position, and recorded the start and end positions of any positive or negative regions. A positive region has two distinctive characteristics: 1) most of the probes in the region have higher signals than the surrounding; 2) the shape of the region resembles either a single triangle or a few merged triangles. A negative region also has some high signal probes, but the shape of the region does not exhibit any peak-like features. The probes in a negative region could be of either similar heights or variable heights. In either case, the distribution of the high signal probes is fairly random. False positive peaks are often observed in the repetitive regions of the genome, which generally have high affinities for non-specific binding. Figure 3.18 shows one such example.

Table 3.3 summarizes the number of standard regions curated for each data set. The term "Joint" refers to the shared DNA binding sites of Dpy-27 and Sdc-3. The standard "Joint" regions were defined by visual inspection of the tiling array data. In Chapter 4, we will describe a generalization of the nonhomogeneous HMM that enables the joint analysis of multiple proteins. The "Joint" peaks were obtained by running this algorithm on the Dpy-27 and Sdc-3 data simultaneously. The number of curated regions is small, relative to the number of positions we had examined. This is because the 10 kb windows surrounding a

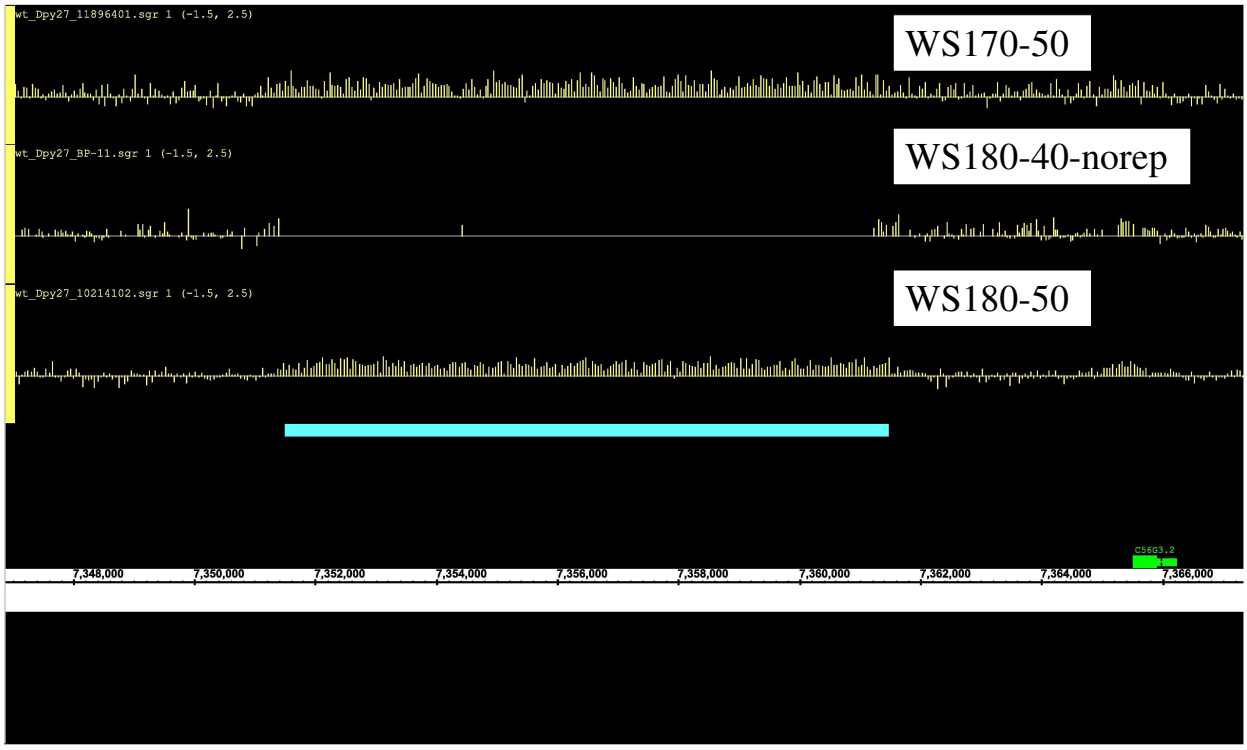

Figure 3.18: A repetitive region that has a false positive interval due to non-specific binding. The middle track has missing data in this region, because the repetitive regions were masked out on Design 2.

Data sets	No. Positives	No. Negatives
wild type Dpy-27	201	225
wild type Sdc-3	225	221
wild type Joint	187	192
mutant Dpy-27	198	303
mutant Sdc-3	328	286
mutant Joint	305	279

Table 3.3: Curated sets of positive and negative regions

large fraction of the positions did not contain any consistent signals that could be classified as either positives or negatives. Among the 900 positions selected for wild type data, roughly 400 did not have any signals in their vicinities, and about 20 positions were surrounded by some irregular signals that could not be classified. Among the 900 positions selected for mutant data, roughly 350 did not have any signals in their vicinities, and about 80 positions were surrounded by some irregular signals that could not be classified. Each set of curated standards was randomly split into two halves. The first half was designated as the training set and the second half was designated as the testing set. We used the training set exclusively in the development of a ranking and filtering procedure. The testing set was reserved for comparing our peak-calling method with some other existing methods, to be described in the next section.

Our ranking procedure involves the following three steps.

1. Compute a measure of signal strength called *peak height*.
2. Compute a measure of shape called *non-uniformity*.
3. Define the final score as a linear combination of *peak height* and *non-uniformity*.

For each interval, consider the probes whose signals are in the top quartile of all the probes within the interval. Let H denote the average height of these top quartile probes. Then, consider the probes which fall within either the 250 bp window upstream of the start position or the 250 bp window downstream of the end position. Let B denote the average height of these background probes. The *peak height* measure is defined as:

$$\textit{peak height} = H - B.$$

Each interval is partitioned into 5 contiguous segments. Let N_i denote the number of probes in the i-th segment, for $i = 1, ..., 5$. Let X_i denote the number of probes in the i-th segment that are greater than the median of all the probes within the interval. For a false positive interval, the high signal probes are randomly distributed along the chromosomal coordinate. So the expected value of $\frac{X_i}{N_i}$ is $\frac{1}{2}$. The *non-uniformity* measure is defined as:

$$\textit{non-uniformity} = \sum_{i=1}^{5} |\frac{X_i}{N_i} - 0.5|.$$

Since each interval results from the integrative analysis of replicate experiments, let us take a moment to discuss how the replicates are handled. For each experiment, we computed the density of probes within a given interval. If the density is at least one probe per 100 bp, then we computed the *peak height* and *non-uniformity* measures using the log ratio data for this experiment. The overall *peak height* and *non-uniformity* measures for the interval were obtained by averaging the measures computed for the individual experiments. When processing the joint peaks of different proteins, the same approach was taken with the experiments of different proteins treated as if they were replicates.

In general, peaks that represent real binding sites should have strong signals and non-rectangular shapes. So the final score could be a linear combination of *peak height* and *non-uniformity*, in which both coefficients are positive. To determine the coefficients, we performed a principal component analysis on these two measures. Figure 3.19 shows scatter plots of all the intervals along the two principal components. The left-hand column represents the wild type condition. The right-hand column represents the *smo-1* mutant condition. The top row shows the results of analyzing Dpy-27 alone. The center row shows the results of analyzing Sdc-3 alone. The bottom row shows the results of analyzing Dpy-27 and Sdc-3 jointly. In each panel, intervals that overlap with the curated positive regions in the training set are shown in red; intervals that overlap with the curated negative regions in the training set are shown in green. The separation of positive and negative intervals is much better for the wild type data than the mutant data. This is because the mutant data are plagued with far more noise. The *smo-1* Sdc-3 data set is the worst of all cases, because it contains only two replicate experiments, both of which were performed using the tiling array Design 2. The average length of probes in Design 2 is about 40 bp, as opposed to 50 for the other two designs. The repeat masking of Design 2 led to many gaps in the tiling array data. Our goal is to develop a ranking procedure that is generally applicable to future experiments, provided they have decent quality. To avoid compromising the performance of our ranking procedure by including poor quality data, we decided to choose the coefficients purely based on the wild type data. The same coefficients were then used to compute the peak scores for the *smo-1* mutant condition.

In each of the wild type scatter plots (Figure 3.19, panels a-c), the best separation of the positive and negative peaks appears to be a line that forms a 45° angle with the x-axis. The y-intercept of the line is arbitrary, as it depends on the trade-off between false positives and false negatives. The optimal scoring function is perpendicular to the line of best separation. Thus we can estimate the coefficients of *peak height* and *non-uniformity* by setting (PC1 − PC2) equal to an arbitrary constant. The coefficients obtained from each plot separately are listed below.

- wild type Dpy-27: **Score** = 1.36 × *peak height* + 0.40 × *non-uniformity*
- wild type Sdc-3: **Score** = 1.38 × *peak height* + 0.30 × *non-uniformity*

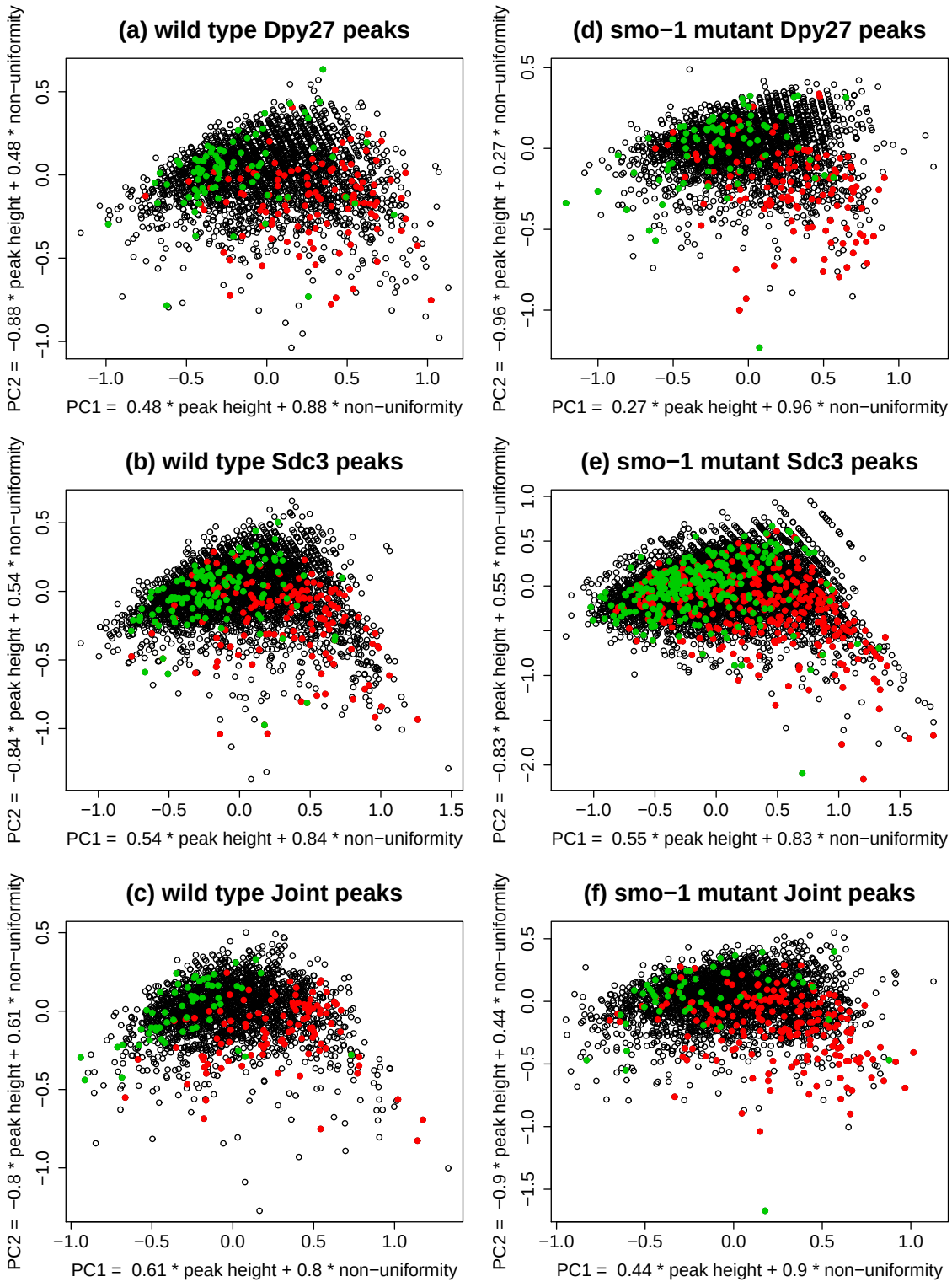

Figure 3.19: Scatter plots of all peaks along the first two principal components. Peaks that overlap with the curated positive regions in the training set are colored in red. Peaks that overlap with the curated negative regions in the training set are colored in green.

- wild type Joint: **Score** $= 1.41 \times$ *peak height* $+ 0.19 \times$ *non-uniformity*

The goal is to find a scoring function that works sufficiently well in general, rather than optimizing for any particular experiment. So we took averages of the coefficient estimates across the three data sets and rounded to the first decimal places. In practice, the precise values of these coefficients are not critical. The final scoring function is given below.

$$\textbf{Score} = 1.4 \times \textit{peak height} + 0.3 \times \textit{non-uniformity}$$

We then performed a receiver operating characteristic (ROC) analysis of the ranking procedure. For each data set, a peak that overlaped with any of the curated positive regions in the training set was defined as a positive peak; a peak that overlaped with any of the curated negative regions in the training set was defined as a negative peak. At any given threshold, false positive and false negative rates were calculated according to the following definitions.

$$\text{true positive rate} = \frac{\text{number of positive intervals that pass the threshold}}{\text{total number of positive intervals}}$$

$$\text{false positive rate} = \frac{\text{number of negative intervals that pass the threshold}}{\text{total number of negative intervals}}$$

Figure 3.20 shows the plots of sensitivity (true positive rate) vs. specificity (1 - false positive rate) obtained by filtering the intervals at various thresholds of the peak scores. The left-hand column represents the wild type condition. The right-hand column represents the *smo-1* mutant condition. The top row shows the results of analyzing Dpy-27 alone. The center row shows the results of analyzing Sdc-3 alone. The bottom row shows the results of analyzing Dpy-27 and Sdc-3 jointly. As discussed earlier, the *smo-1* Sdc-3 data set has the most noise, thus the highest error rates. In general, the joint analysis of two proteins has improved performance over the separate analyses of single proteins. Please note that these sensitivity and specificity measures only reflect the performance of filtering by peak scores, but not the nonhomogeneous HMM that leads to the candidate probes. The entire package of probe level analysis by nonhomogeneous HMM and postprocessing of the peaks will be evaluated in the next section.

3.5 Comparisons with existing methods for ChIP-chip data analysis

A systematic evaluation of the ChIP-chip technology using spike-in DNA samples was reported in 2008 [36]. In this study, a pool of 100 randomly selected cloned genomic DNA sequences were mixed at various concentrations, and spiked into a commercial preparation of human genomic DNA. One spike-in mixture was prepared at a high concentration for direct labeling and hybridization to tiling arrays. Another spike-in mixture was prepared at a low concentration, and required DNA amplification before hybridization to tiling arrays.

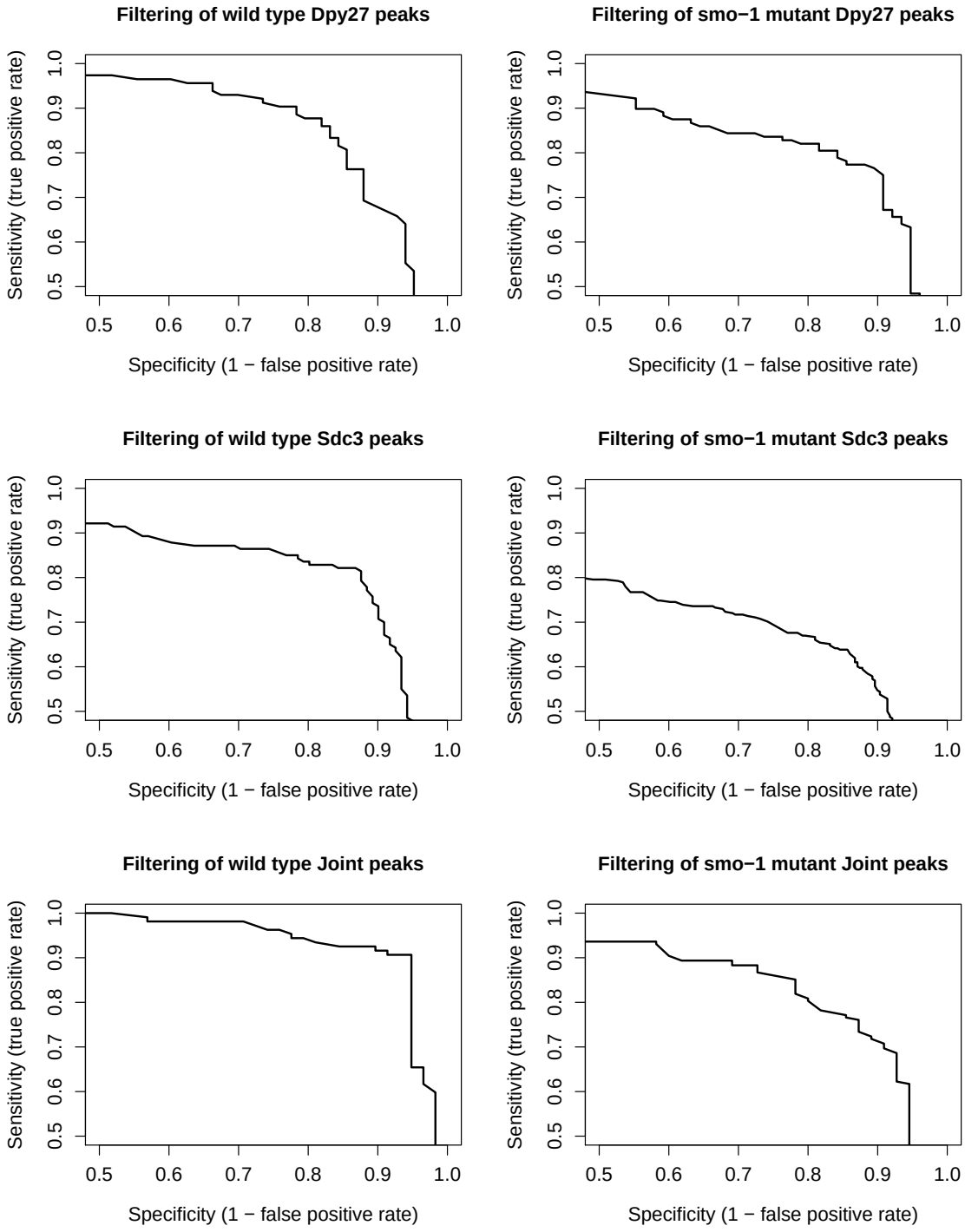

Figure 3.20: ROC analysis of the peak filtering procedure for various data sets. True positive and false positive rates are defined in terms of the intervals that overlap with the curated positive and negative regions in the training set.

The two mixtures were distributed to 8 independent research groups, all of which were blind to the true spike-in regions. Each laboratory hybridized the spike-in DNA samples to one of four different types of tiling arrays: Affymetrix, Agilent, NimbleGen, and a PCR tiling array. The tiling array data were analyzed using 13 different algorithms, and the results were reported as rank-ordered lists of predicted spike-in regions. The authors used an ROC-like curve analysis to compare the different peak-calling algorithms for each type of tiling arrays. In the case of NimbleGen tiling arrays, TAMALPAIS [6] was the best algorithm for analyzing both the concentrated sample and the diluted sample. The runner-ups were NimbleGen's peak-calling algorithm [43] for the concentrated sample, and MA2C [60] for the diluted sample. We decided to compare our method with these three "best" existing peak-calling algorithms for analyzing NimbleGen tiling arrays.

3.5.1 Review of the three best existing methods

TAMALPAIS begins by applying a percentile threshold on the log ratios of IP versus input for each array. For a given threshold (i.e. 95th percentile or 98th percentile), each probe position is recoded as either 0 (below the threshold) or 1 (above the threshold). Positions that have missing data due to repeat masking are recoded as X. The result is a sequence of 0's and 1's, intercepted with X's that are ignored in the analysis. Real binding sites should be represented by long runs of 1's. However, short runs of 1's may occur simply by chance. Let n denote the length of the sequence, which is about 2 million probes for the *C. elegans* tiling arrays. Let p denote the probability of getting a 1 at each position, which depends on the percentile threshold. Let R_n denote the length of the longest consecutive run of 1's that occurs by chance. Erdös and Rényi [28] proved that

$$\frac{R_n}{\log(n)} \rightarrow \frac{1}{\log(1/p)}.$$

Gordon, Schilling and Waterman [30] proved that

$$\text{mean}(R_n) = \log_{1/p}(n) + \frac{0.577}{\theta} - \frac{1}{2} + r_1(n)$$

and

$$\text{variance}(R_n) = \frac{\pi^2}{6\theta^2} + \frac{1}{12} + r_2(n)$$

where 0.577 is the Euler-Mascheroni constant, $\theta = ln(1/p)$, and $r_1(n)$ and $r_2(n)$ are negligible for large n.

For each observed run of 1's of length W, TAMALPAIS computes a z-score according to

$$z = \frac{W - \text{mean}(R_n)}{\sqrt{\text{variance}(R_n)}}.$$

Under the extreme-value distribution, the p-value of observing W is

$$P(Z > z) = 1 - exp[-exp(-1.2825z - 0.577)].$$

TAMALPAIS reports the results of this algorithm using 4 different combinations of the percentile threshold and the p-value cutoff, leading to 4 levels of stringency named as L1-L4. L1 uses the 98th percentile with p-value < 0.0001; L2 uses the 95th percentile with p-value < 0.0001; L3 uses the 98th percentile with p-value < 0.05; L4 uses the 95th percentile with p-value < 0.05 [6]. TAMALPAIS is known to be fairly conservative. Because we are interested in comparing the receiver operating characteristics of various algorithms, it is important that we start with the most inclusive list of peaks calls from each algorithm. So we chose to use the L4 output from TAMALPAIS in the ROC analysis.

The peak-calling algorithm implemented in NimbleGen's analysis software, called NimbleScan, is based on a paper by Lucas et al. in 2007 [43]. A sliding window of user defined width is moved across the data track of each experiment. If the log ratio of a probe exceeds the probe-height cutoff, then this probe is a potential peak probe. If the number of potential peak probes in the window is equal to or greater than the user defined *in-window-number* cutoff, then a peak is recognized. As the sliding window moves along the data track, the length of a recognized peak can be extended. This peak detection procedure is repeated with decreasing values of the probe-height cutoff. A parameter called p is used to represent the probe-height cutoff, with larger values meaning higher stringency. Under the default settings, peak detection is done 76 times, starting with $p = 90$, decreasing p by 1 in each step, and finishing with $p = 15$. A peak that is recognized in the first scan is assigned to the category of $p = 90$. A peak that is not recognized in the first scan, but emerges in the second scan is assigned to the category of $p = 89$. Thus the peaks can be ranked by stringency according to the parameter p. FDRs are calculated by applying the same peak detection procedure to permuted data that simulate background noise. For a given value of p, the ratio of the average number of peaks recognized across 20 permuted data tracks to the number of peaks recognized in the original data is considered as the FDR. For the 50-mer tiling arrays used in Lucas et al., the authors found a window size of 625 bp to be optimal. So we chose a window size of 600 bp to analyze the *C. elegans* tiling arrays with NimbleScan. The *in-window-number* cutoff is allowed to be different depending on whether there are probes in the window that are below the probe-height cutoff. When the window includes probes that are below the probe-height cutoff, we set the *in-window-number* cutoff to 6 probes. When all the probes in the window are above the probe-height cutoff, we set the *in-window-number* cutoff to 3 probes.

MA2C preprocesses the hybridization intensities of each array using a GC-specific normalization. Probes are assigned to different bins depending on their GC-contents. Within each GC-bin, probes are standardized using the same set of estimates for the means and variances of the background intensities, as well as the covariance between the two channels. The parameters can be estimated robustly using a two-dimensional generalization of Tukey's bi-weight estimation. Each probe is summarized by a normalized and correlation weighted log ratio, which is then rescaled globally to ensure that the normalized score has variance 1. Peak detection is based on the sliding window approach. An MA2C score is computed for a window of user defined length, centered at each probe position. The program provides a number of options for computing this window-level MA2C score. The default option is to use the median of the normalized probe scores within each window. Windows with MA2C scores

exceeding a certain threshold are then joined into peaks. The threshold for the MA2C scores can be determined using either the p-value approach or the FDR approach. The p-values are assigned based on the normal probability distribution. The FDR is estimated empirically as follows. For a given cutoff value M, peaks with MA2C scores greater than M are considered as positive MA2C score peaks; peaks with MA2C scores less than $-M$ are considered as negative MA2C score peaks. FDR is estimated as

$$\text{FDR} = \frac{\text{number of negative MA2Cscore peaks}}{\text{number of positive MA2Cscore peaks}}.$$

The authors of MA2C applied their method to another set of *C. elegans* ChIP-chip experiments of Dpy-27 and Sdc-3 [27]. They used a window size of 600 bp, with the MA2C score cutoffs based on p-values of 10^{-4} and 10^{-5} [60]. We also adopted the window size of 600 bp when running MA2C on the Meyer Lab data. Since we were interested in analyzing the receiver operating characteristics of MA2C, we lowered the MA2C score cutoff to p-value $< 10^{-3}$, to obtain a more inclusive list of peaks.

3.5.2 Results of comparisons by ROC

TAMALPAIS and NimbleScan were designed to analyze one experiment at a time. Although previous studies had tried various ways of combining the peak calls from replicate experiments, they all involved making arbitrary decisions at various steps. MA2C provides the option of averaging the probe level scores from replicate experiments before peak detection. However, this requires the same tiling array design to be used in all of the replicate experiments. Since our study involves replicate experiments performed on tiling arrays with different designs, the replicate analysis option provided in MA2C cannot be utilized. Thus we decided to perform the ROC analysis of these three existing methods at the level of single experiments. The testing set of curated standards, which was described in the Section 3.4, was used as the benchmark. For a given method, 100 different sets of peak calls were produced by thresholding the peaks scores at different values. A curated positive region that overlaped with any of the called peaks was considered a positive hit. A curated negative region that overlaped with any of the called peaks was considered a negative hit. Sensitivity and specificity were calculated according to the following definitions.

$$\text{sensitivity} = \frac{\text{number of positive hits}}{\text{number of curated positive regions}}.$$

$$\text{specificity} = 1 - \frac{\text{number of negative hits}}{\text{number of curated negative regions}}.$$

Figures 3.21-3.24 show the ROC curves for four different peak calling algorithms when applied to each experiment separately. Since the nonhomogeneous HMM is proposed specifically for the integration of tiling array designs, we also ran this method on the combined dataset of all replicates for each protein. The combined analysis of replicated data by nonhomogeneous HMM is represented by solid red curves. The single experiment version of our method is represented by dashed red curves. TAMALPAIS is represented by dashed green

curves; NimbleScan is represented by dashed cyan curves; MA2C is represented by dashed blue curves. In each figure, different replicates of the same protein are plotted in different rows, with the experiment IDs shown in the legends. The right-hand side panels are zoomed-in versions of the left-hand side panels. When applied to each experiment separately, our method performed similarly as the best existing methods. When the replicate experiments were combined, our method performed with pronounced improvements.

Please note that the sensitivity and specificity values shown in the ROC curves are only meant to be comparative. Since we do not know the true binding sites of these dosage compensation proteins, we cannot make any statements about the absolute sensitivities and specificities of any particular method. The curated sets of positive and negative regions were selected for the purpose of comparing different methods, when we don't know the truth. Sensitivity and specificity defined in terms of these curated standards would invariably depend on the overall difficulty of classifying these regions. For example, when there is a good signal to noise ratio in the raw data, the human curator may be able to select more weak positive peaks by visual inspection, as in the case of the wild type data. However, when the quality of the raw data is poor, the human curator may refrain from making judgement calls about borderline cases, as in the case of the mutant data. When the curated set of standards contains more borderline cases, the overall sensitivities and specificities of all the methods would appear lower. At the same time, we would also expect to see a better separation of the different methods by ROC analysis. On the other hand, when the curated set of standards contains fewer borderline cases, all of the different methods would appear to perform better. But these sensitivity and specificity values should not be interpreted as absolute, and can only be used for making comparisons within the same data set. The differences between the wild type and mutant ROC results can be explained by this phenomenon.

Table 3.4 summarizes the number of peaks called by each method, when specificity is controlled at 90% or above. In most cases, the peak counts of our method are similar to the peak counts of the best existing methods. However, our method called fewer peaks for *smo-1* Dpy-27 when applied to each replicate separately. We looked through the regions that were called *smo-1* Dpy-27 peaks by the other methods but not ours, and found the majority of them to be in one of two categories. In the first category, the peaks have very weak signals. When each replicate experiment was analyzed separately, our method considered the weak signals insignificant. When the replicates are combined, our method recognized the peaks due to the consistency of the weak signals. A couple of examples are shown in Figures 3.25 and 3.26. In the second category, the peaks have strong signals. But they were filtered out because of their roughly rectangular shapes. A couple of examples are shown in Figures 3.27 and 3.28. These results suggest that the nonhomogeneous HMM does not perform worse than the best existing methods, when applied to each experiment separately. The advantage of the nonhomogeneous HMM is its capacity for integrating replicate experiments performed using tiling arrays with different probe designs. Indeed, the combined analysis by nonhomogeneous HMM is far more powerful than the existing methods.

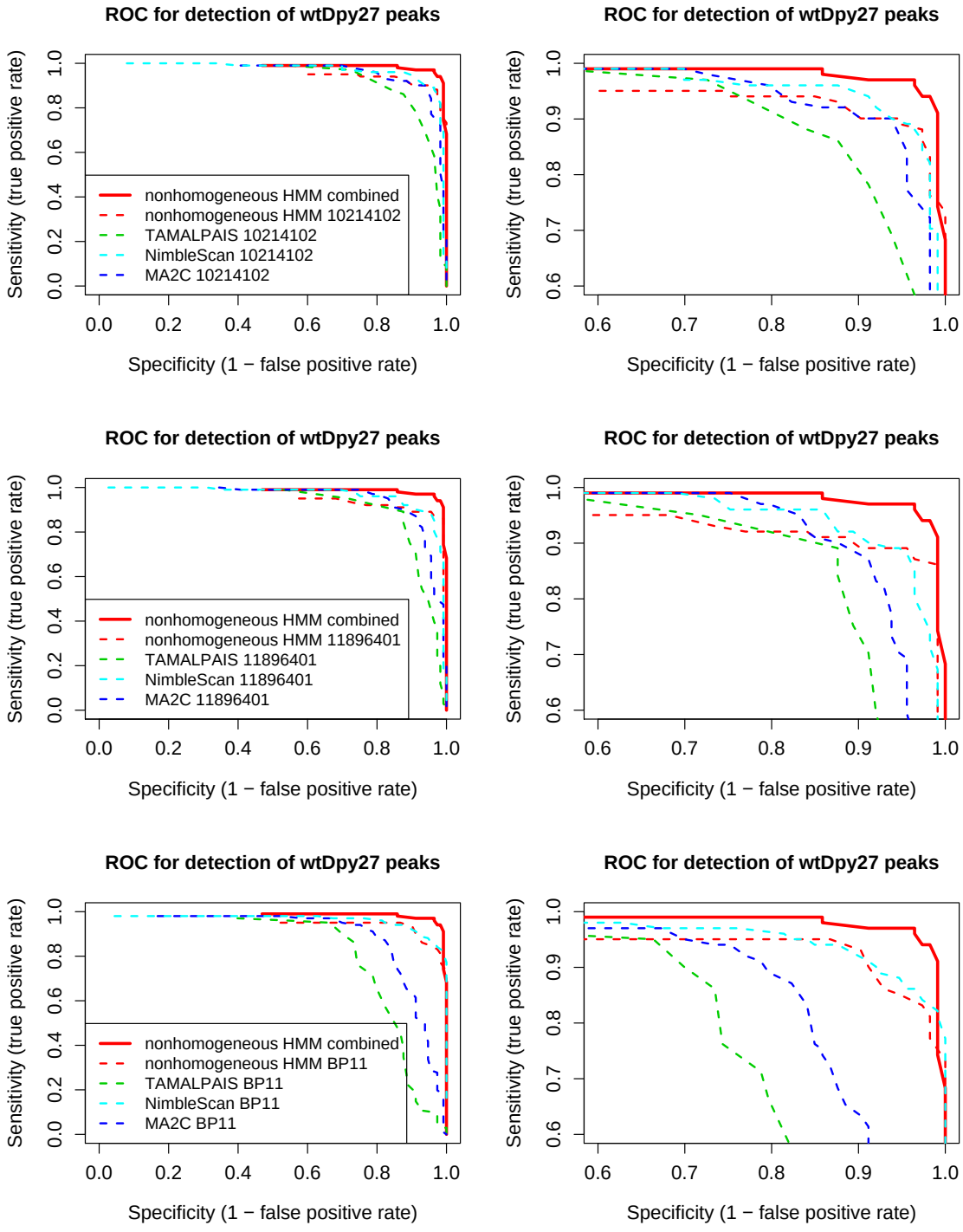

Figure 3.21: ROC curves for wild type Dpy-27. Each experiment was analyzed separately by: HMM (red, dashed), TAMALPAIS (green, dashed), NimbleScan (cyan, dashed), MA2C (blue, dashed). The combined set of 3 replicates was analyzed by HMM (red, solid). Right-hand side panels are zoomed in versions of left-hand side panels.

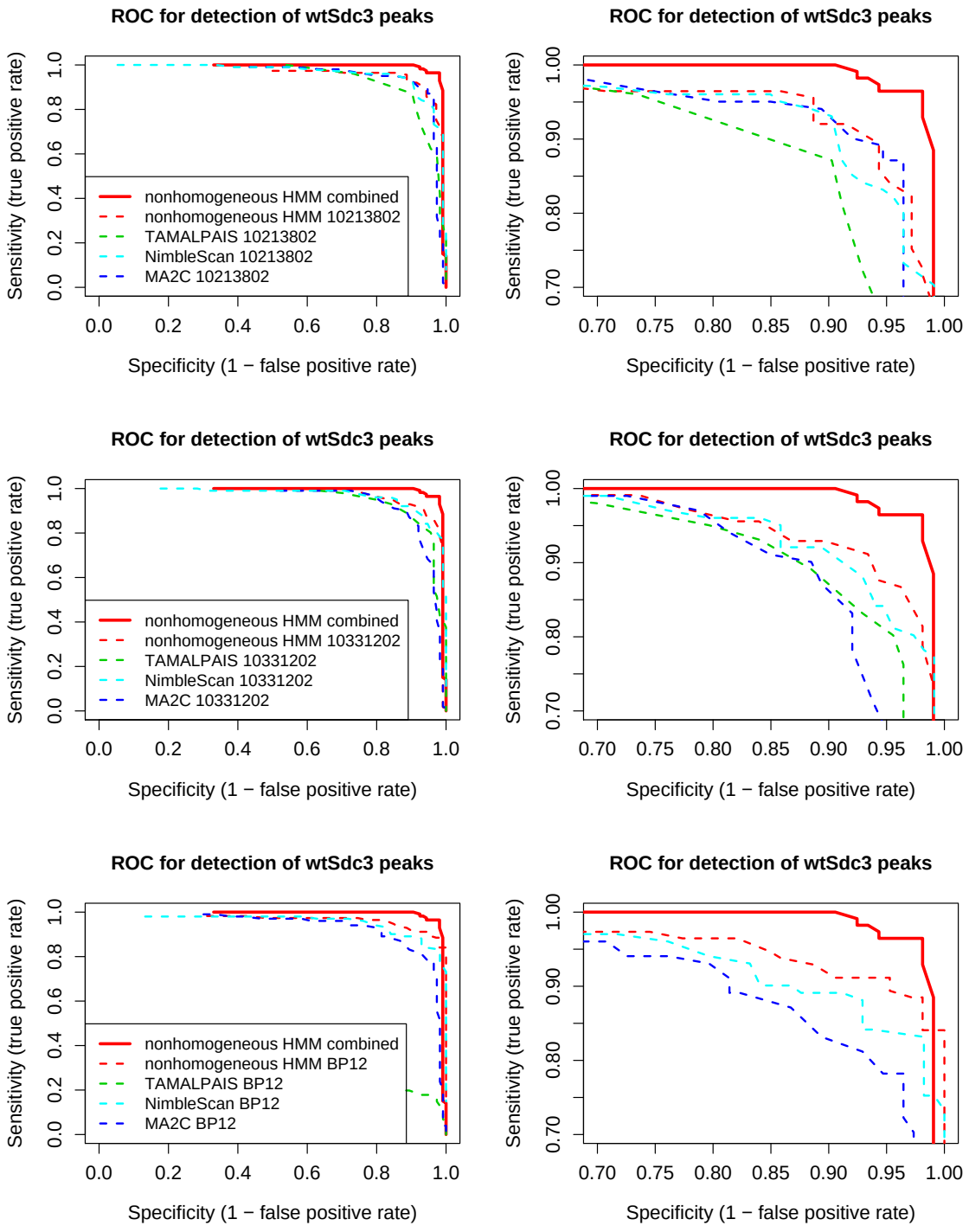

Figure 3.22: ROC curves for wild type Sdc-3. Each experiment was analyzed separately by: HMM (red, dashed), TAMALPAIS (green, dashed), NimbleScan (cyan, dashed), MA2C (blue, dashed). The combined set of 3 replicates was analyzed by HMM (red, solid). Right-hand side panels are zoomed in versions of left-hand side panels.

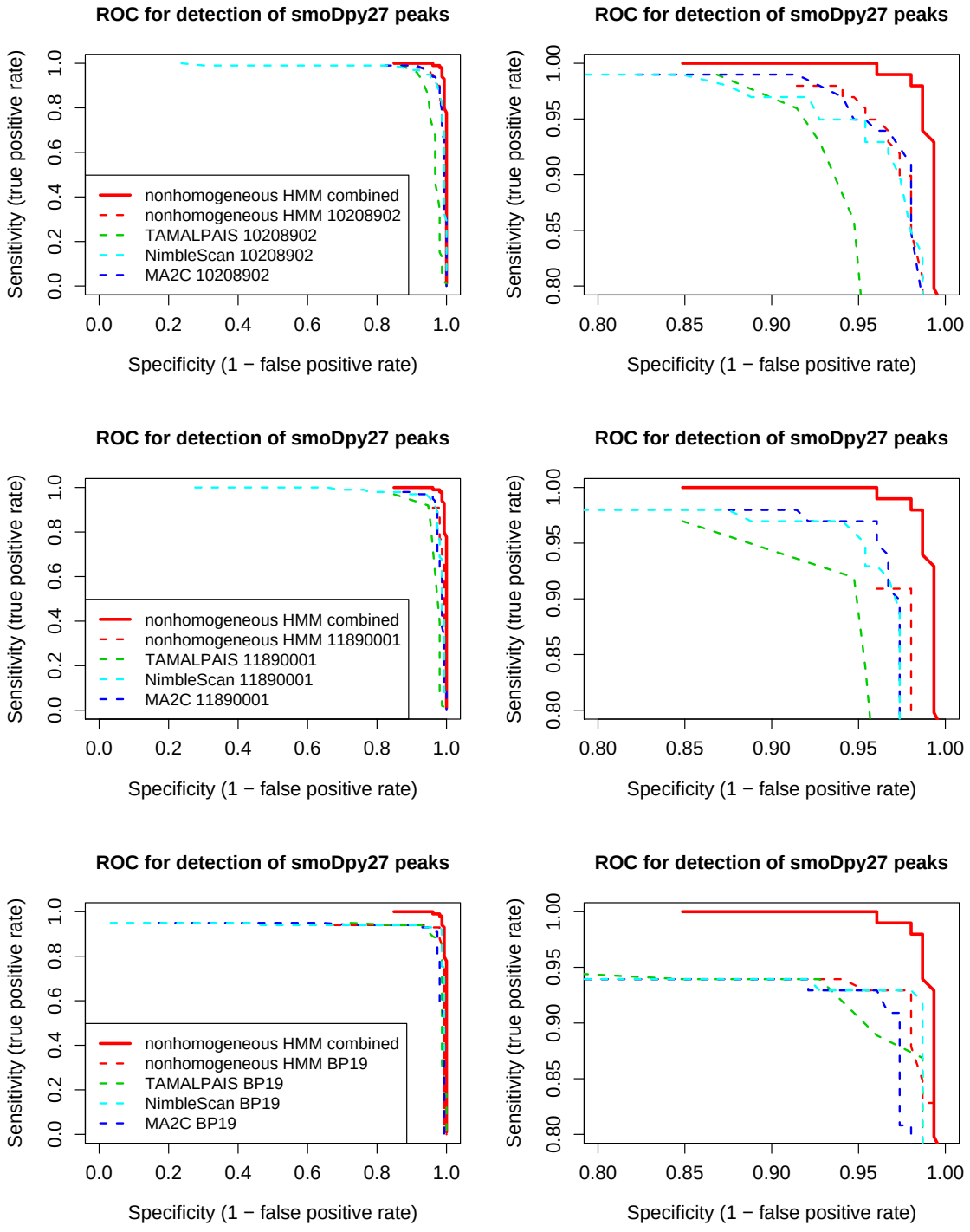

Figure 3.23: ROC curves for *smo-1* mutant Dpy-27. Each experiment was analyzed separately by: HMM (red, dashed), TAMALPAIS (green, dashed), NimbleScan (cyan, dashed), MA2C (blue, dashed). The combined set of 3 replicates was analyzed by HMM (red, solid). Right-hand side panels are zoomed in versions of left-hand side panels.

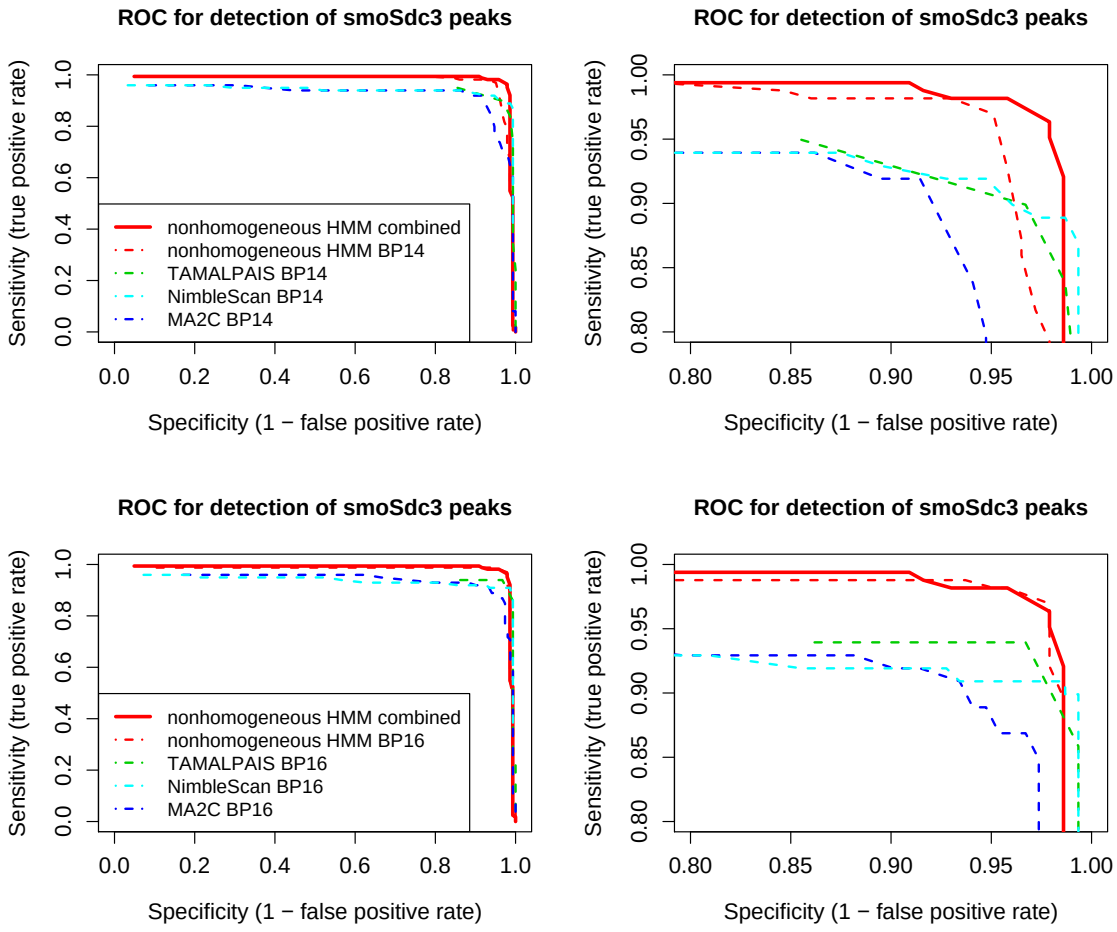

Figure 3.24: ROC curves for *smo-1* mutant Sdc-3. Each experiment was analyzed separately by: HMM (red, dashed), TAMALPAIS (green, dashed), NimbleScan (cyan, dashed), MA2C (blue, dashed). The combined set of 2 replicates was analyzed by HMM (red, solid). Right-hand side panels are zoomed in versions of left-hand side panels.

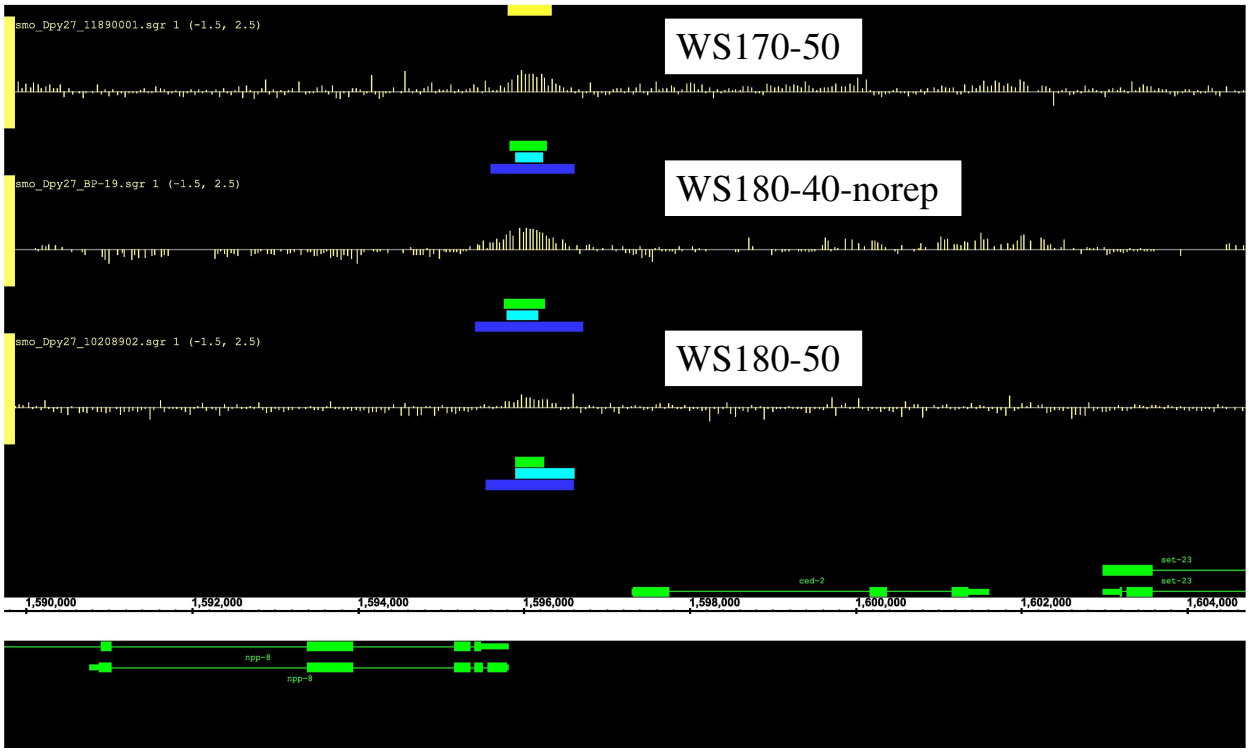

Figure 3.25: Weak peaks that were called by the other methods but not by the nonhomogeneous HMM, when applied to each replicate separately. Yellow intervals represent peaks called by the combined version of the nonhomogeneous HMM. Red intervals represent peaks called by the single experiment version of the nonhomogeneous HMM. Here, red intervals are missing because the single experiment analysis did not recognize any peaks. But the combined analysis recognized a peak. Green intervals represent peaks called by TAMALPAIS. Cyan intervals represent peaks called by NimbleScan. Blue intervals represent peaks called by MA2C.

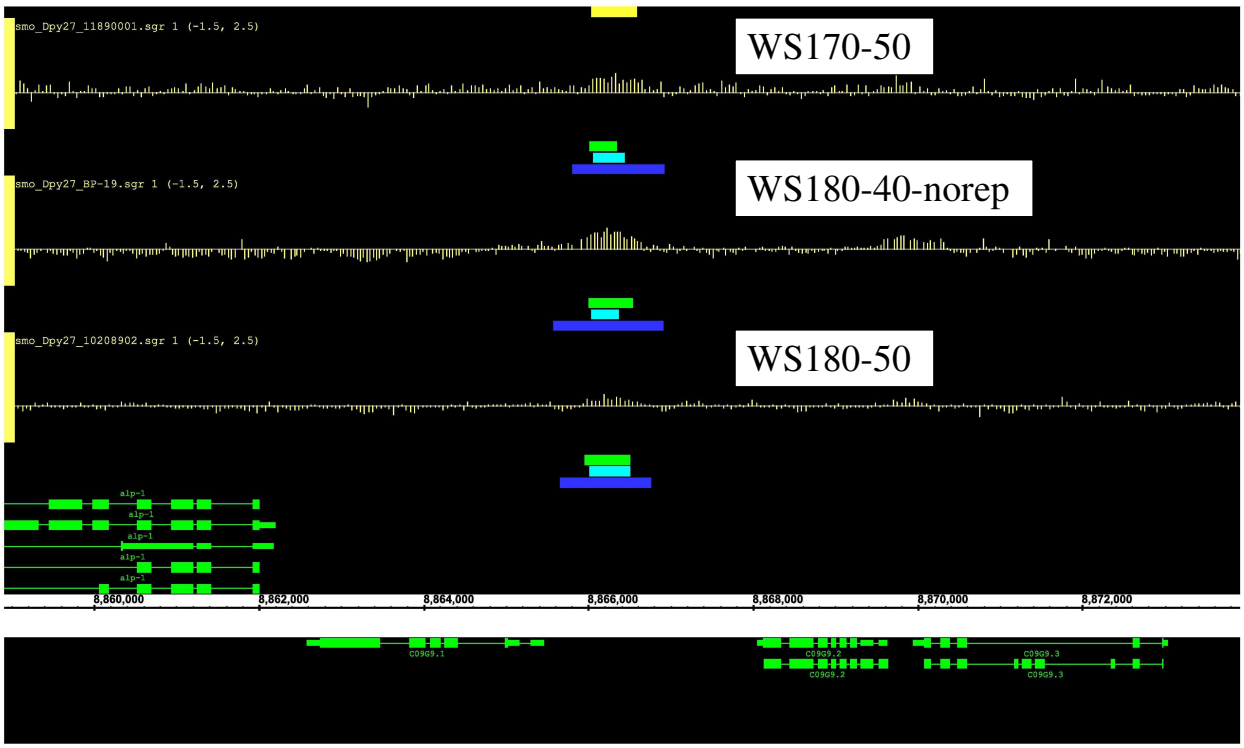

Figure 3.26: Weak peaks that were called by the other methods but not by the nonhomogeneous HMM, when applied to each replicate separately. Yellow intervals represent peaks called by the combined version of the nonhomogeneous HMM. Red intervals represent peaks called by the single experiment version of the nonhomogeneous HMM. Here, red intervals are missing because the single experiment analysis did not recognize any peaks. But the combined analysis recognized a peak. Green intervals represent peaks called by TAMALPAIS. Cyan intervals represent peaks called by NimbleScan. Blue intervals represent peaks called by MA2C.

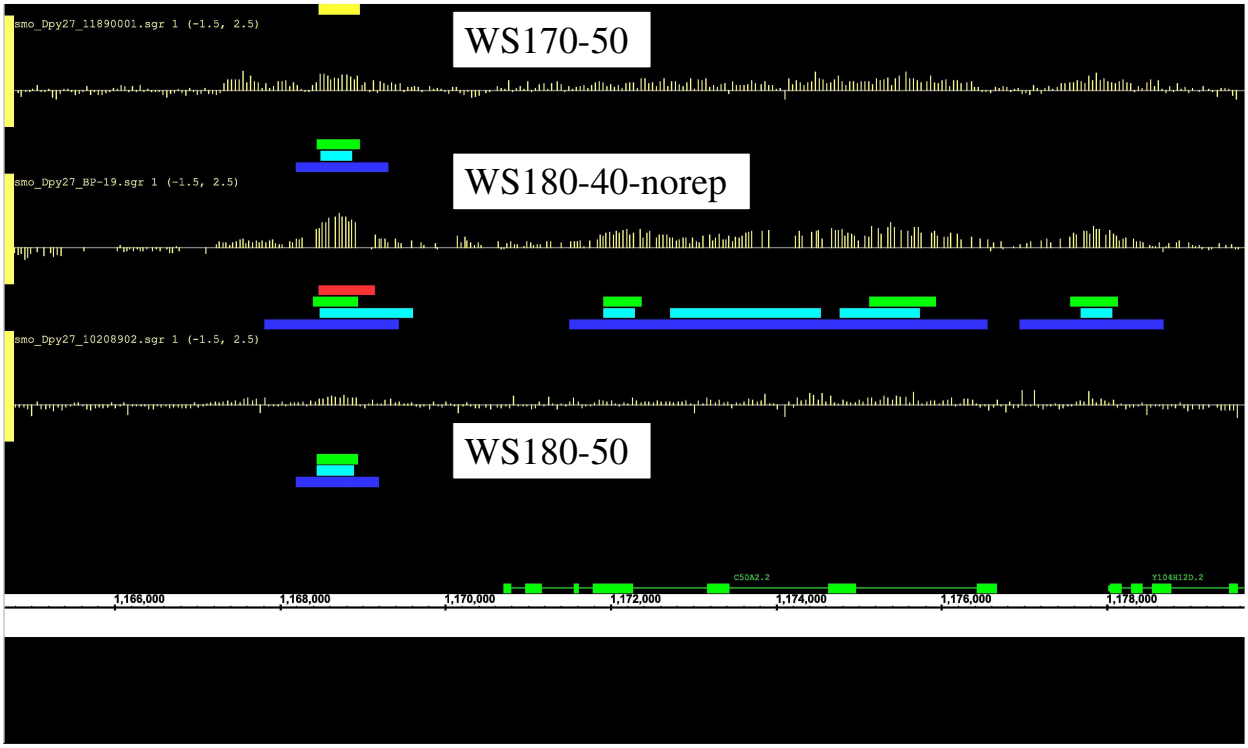

Figure 3.27: Peaks that were called by the other methods but not by our method due to postprocessing. Yellow intervals represent peaks called by the combined version of the nonhomogeneous HMM. Red intervals represent peaks called by the single experiment version of the nonhomogeneous HMM. Green intervals represent peaks called by TAMALPAIS. Cyan intervals represent peaks called by NimbleScan. Blue intervals represent peaks called by MA2C. The intervals in the region between 1,172,000 and 1,176,000 were filtered out by our method due to their roughly rectangular shapes.

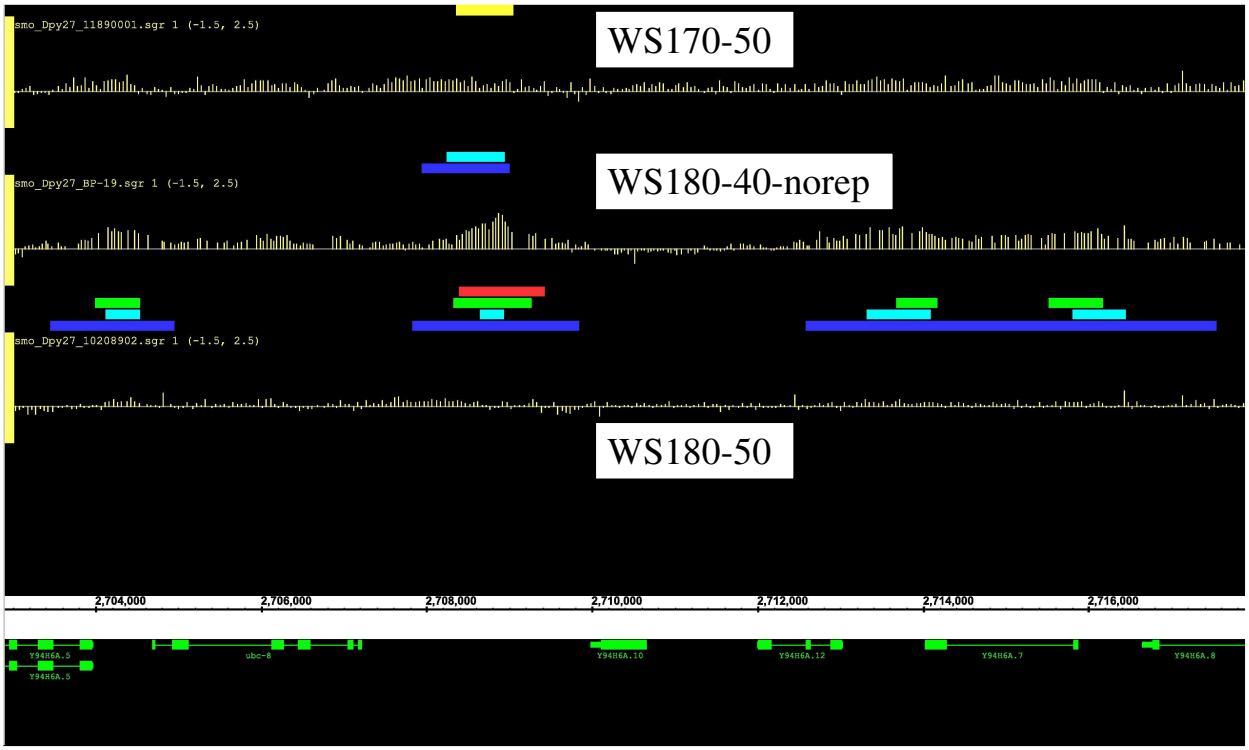

Figure 3.28: Peaks that were called by the other methods but not by our method due to postprocessing. Yellow intervals represent peaks called by the combined version of the nonhomogeneous HMM. Red intervals represent peaks called by the single experiment version of the nonhomogeneous HMM. Green intervals represent peaks called by TAMALPAIS. Cyan intervals represent peaks called by NimbleScan. Blue intervals represent peaks called by MA2C. The intervals in the region between 2,712,500 and 2,716,500 were filtered out by our method due to their roughly rectangular shapes.

Experiments	nh HMM	TAMALPAIS	NimbleScan	MA2C
wild type Dpy-27 combined	1559	–	–	–
wild type Dpy-27 10214102	1327	1052	1605	1052
wild type Dpy-27 11896401	1380	1406	1473	1006
wild type Dpy-27 BP11	1308	294	1479	565
wild type Sdc-3 combined	2518	–	–	–
wild type Sdc-3 10213802	1485	1727	1950	1636
wild type Sdc-3 10331202	1504	1657	1755	1253
wild type Sdc-3 BP12	1964	623	1776	1438
smo-1 Dpy-27 combined	2555	–	–	–
smo-1 Dpy-27 10208902	1822	2330	2576	1953
smo-1 Dpy-27 11890001	1038	2166	2527	2385
smo-1 Dpy-27 BP19	2407	3070	3351	2694
smo-1 Sdc-3 combined	3611	–	–	–
smo-1 Sdc-3 BP14	3118	4051	4231	3340
smo-1 Sdc-3 BP16	3249	4540	4420	3393

Table 3.4: Summary of peak calls by various methods

Chapter 4

A peak-detection method for the joint analysis of ChIP-chip data from multiple DNA binding proteins

4.1 Motivation

Many processes in gene regulation are carried out by multiple-protein complexes. A well-known example is the Chromatin Structure Remodeling Complex (RSC), which contains 17 subunits. Recently, an increasing number of such examples have been found in histone modifications [63]. The mammalian MLL3/4 Set1-H3K4 methyltransferase complex coordinates the removal of a repressive methyl mark with the formation of an activating methyl mark on histone H3. The Polycomb group (PcG) of transcriptional silencing complexes consist of three separate protein complexes (PRC1, PRC2 and PhoRC) that assemble on chromatin and coordinate H2A lysine 119 ubiquitination and H3 lysine methylation. The recruitment of these protein complexes to DNA is hierarchical. The process that motivated our study is dosage compensation. In mammals and flies, dosage compensation is associated with specific histone post-translational modifications and histone variants replacements. A recent study reported a new implication of histone modification in *C. elegans* dosage compensation [50]. Barbara Meyer's group is interested in the genome wide localization of the dosage compensation complex (DCC) under various conditions. They performed ChIP-chip experiments in both wild type worms and mutant worms that are deficient for histone sumoylation. Components of the DCC that they investigated include Dpy-27, a condensin homolog, and Sdc-3, a zinc-finger protein. Replicate experiments were performed using NimbleGen two-color tiling arrays with three different designs. Table 3.2 provides a summary of the ChIP-chip experiments. The main goal of these experiments is to identify chromosomal regions that are jointly bound by Dpy-27 and Sdc-3, which constitute the DCC binding targets.

The existing methods for ChIP-chip data analysis are designed to analyze only one protein at a time. Ad-hoc approaches have been taken when there is a need to integrate information about two or more proteins. When all of the tiling arrays have the same probe design, the probe level summary statistics can be averaged, and the single-protein analysis methods can

be applied to the averages. Alternatively, each experiment can be analyzed separately, and the peaks calls from individual experiments can be cross-listed to obtain the jointly bound sites. However, none of these approaches are satisfying either in theory or in practice. With more and more ChIP-chip studies of protein complexes underway in the scientific community, there is a compelling need for an algorithm that enables the joint analysis of two or more proteins. In this chapter, we describe a generalization of the nonhomogeneous HMM that enables the joint analysis of multiple proteins.

4.2 Conditional independence of the observations

Since the experiments were performed independently, it is reasonable to assume that the observations of different proteins are independent conditional on the hidden states. This assumption reduces the dimensionality of the parameter space dramatically. However, this assumption needs to be validated by checking for potential correlations between the proteins.

It is well known that the design of the probes can affect the signals on microarrays. Thus it is worthwhile to consider the correlations separately for each design of tiling arrays. Figure 4.1 shows some scatter-plots of the data collected using tiling array Design 2, for a 1 MB region on Chromosome X. Figure 4.2 shows some scatter-plots of the data collected using Design 3, for the same chromosomal region. In each panel, the wild type Dpy-27 data are plotted along the horizontal axis, and the wild type Sdc-3 data are plotted along the vertical axis. Because the correlations are part of the state-dependent emission parameters, the data points were stratified by the states. Ideally, we would like to stratify the data points by the true hidden states. Since the true hidden states were not available, we decided to use the inferred states. Details of the hidden state inference method will be discussed in the next section. The focus of this section is to establish the premises for building the method. The inferred states are color-coded as follows. State 00 (not bound by either protein) is colored in black; State 01 (bound by Sdc-3 but not bound by Dpy-27) is colored in red; State 10 (bound by Dpy-27 but not bound by Sdc-3) is colored in green; State 11 (jointly bound by Dpy-27 and Sdc-3) is colored in blue. The correlations are nearly zero in States 00, 01 and 01. However, the correlations in State 11 are 0.71 for Design 2 and 0.62 for Design 3.

Although correlations exist in the jointly bound state, it may still be advantageous to ignore the correlations to make the inference problem feasible in general. To explore this possibility, we simulated some bivariate Gaussian data as follows. A Markov chain was generated along every base position in the same 1 MB region of Chromosome X, according to the initial distribution π and the transition matrix $\mathbf{P}$.

$$\pi = (0.772, 0.035, 0.058, 0.135)$$

$$\mathbf{P} = \begin{pmatrix} 0.9988 & 0.0006 & 0.0004 & 0.0003 \\ 0.0126 & 0.9857 & 0.0007 & 0.0010 \\ 0.0048 & 0.0005 & 0.9929 & 0.0018 \\ 0.0017 & 0.0002 & 0.0007 & 0.9974 \end{pmatrix}$$

A total of two datasets were generated according to each tiling array design, with one

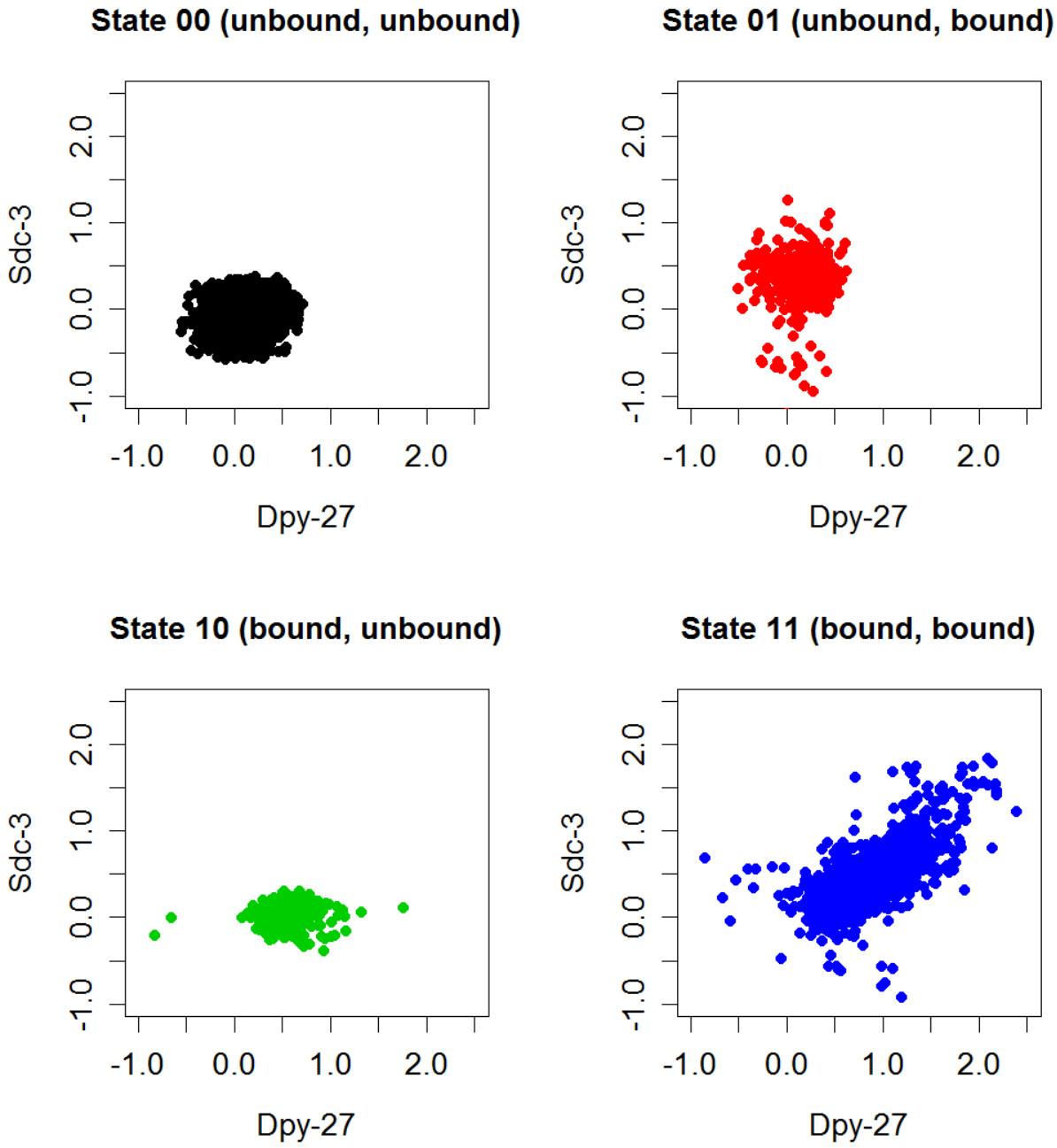

Figure 4.1: Correlation plots for tiling arrays of Design 2. Wild type data for a pair of ChIP-chip experiments are stratified by the inferred states in the two-protein model. Each data point represents a probe on tiling array Design 2. There appears to be a positive correlation between Dpy-27 and Sdc-3 in State 11.

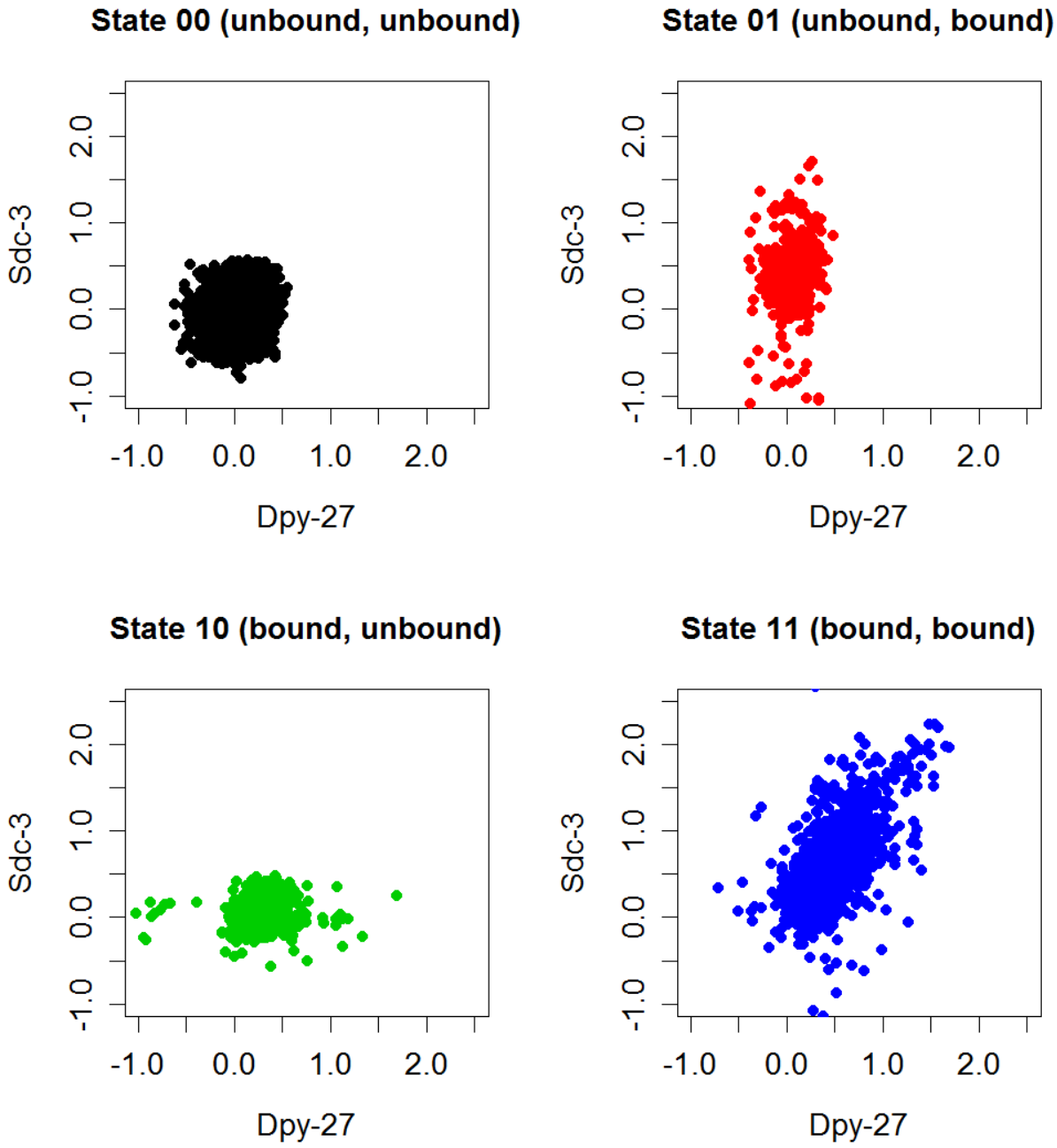

Figure 4.2: Correlation plots for tiling arrays of Design 3. Wild type data for a pair of ChIP-chip experiments are stratified by the inferred states in the two-protein model. Each data point represents a probe on tiling array Design 3. There appears to be a positive correlation between Dpy-27 and Sdc-3 in State 11.

Design	Label	unbound μ	unbound σ	bound μ	bound σ
1	WS170-50	0.124	0.151	0.6577	0.195
2	WS180-40-norep	0.170	0.128	0.675	0.178
3	WS180-50	0.053	0.121	0.357	0.161

Table 4.1: Emission parameters for simulating Dpy-27 data

Design	Label	unbound μ	unbound σ	bound μ	bound σ
1	WS170-50	-0.001	0.153	0.927	0.254
2	WS180-40-norep	-0.049	0.115	0.299	0.179
3	WS180-50	-0.005	0.132	0.421	0.189

Table 4.2: Emission parameters for simulating Sdc-3 data

representing Dpy-27 and one representing Sdc-3. The actual center positions of the probes in each design dictated where the observations occurred. At each observed position, bivariate Gaussian data were generated according to the design-specific parameters given in Table 4.1 and Table 4.2. Details of the data simulation procedure can be found in Section 4.4. The only difference here is that correlations were added to the simulation model. In States 00, 01 and 10, the correlations between Dpy-27 and Sdc-3 were set to zero. In State 11, the correlation was set to either 0.6, 0.7 or 0.8, depending on the tiling array design.

This simulated data set was subsequently fitted to two models. The first model assumes conditional independence between the two proteins. The second model assumes dependence between the two proteins in State 11. Thus the second model contains three additional parameters, which are the design-specific correlations between Dpy-27 and Sdc-3 in State 11. After model fitting, each observed position was assigned to a state according to the posterior probabilities, i.e. the state which has the largest value for the γ variable. A confusion matrix was tabulated for each model, to compare the inferred states with the true states. The results of one simulation experiment are shown in Table 4.3 and Table 4.4.

To compare Table 4.3 and Table 4.4, let us focus on the diagonal elements. Differences in the off-diagonal elements are only complementary to the differences in the diagonal elements.

	State 00	State 01	State 10	State 11
True State 00	48142	41	46	6
True State 01	84	2042	1	19
True State 10	104	2	3587	24
True State 11	45	16	20	7939

Table 4.3: States reconstruction by the conditional independence model

	State 00	State 01	State 10	State 11
True State 00	48133	42	47	13
True State 01	83	2059	1	3
True State 10	100	2	3601	14
True State 11	27	5	14	7974

Table 4.4: States reconstruction by the bivariate model

	State 00	State 01	State 10	State 11
True State 00	-0.014 %	–	–	–
True State 01	–	+0.365 %	–	–
True State 10	–	–	+0.223 %	–
True State 11	–	–	–	+0.377 %

Table 4.5: Percent differences between the diagonal elements in the confusion matrices of the two models

A higher count in a diagonal element means that more probes of a particular state are correctly classified. There appears to be a slight improvement in the classification rates when the conditional independence model is extended into the bivariate model, by estimating design-specific the correlations in State 11. To assess whether the difference is real, 100 cycles of data simulation and model fitting were performed. In each cycle, the diagonal elements in the confusion matrix of the conditional independence model were subtracted from those of the bivariate model, to obtain the differences. These differences were then divided by the respective average counts and converted into percentages. Table 4.5 summarizes the percent differences when averaged over the 100 cycles.

Figure 4.3 shows the box plots of the percent differences for the diagonal elements of the confusion matrices. In each panel, the solid line inside the box represents the median over 100 simulations, and the dashed lines represent 2 SDs away from the mean. Positive values indicate improvements of the bivariate model over the conditional independence model, in classification rates. For each of the binding states (01, 10 and 11), the bivariate model exhibits a slight improvement of less than 1%. Thus we conclude that the benefits of estimating the correlation parameters are minimal.

This simulation study was carried out in a vanilla setting. Each pair of experiments was performed using tiling arrays with the same design. The three different designs were covered by exactly three pairs of experiments. In practical situations, such a balanced pairing of experiments is uncommon. Thus the estimation of correlations is likely to be unfeasible in general. Since the conditional independence model is quite robust even in the presence of correlations, we decided to proceed with it for its feasibility merits.

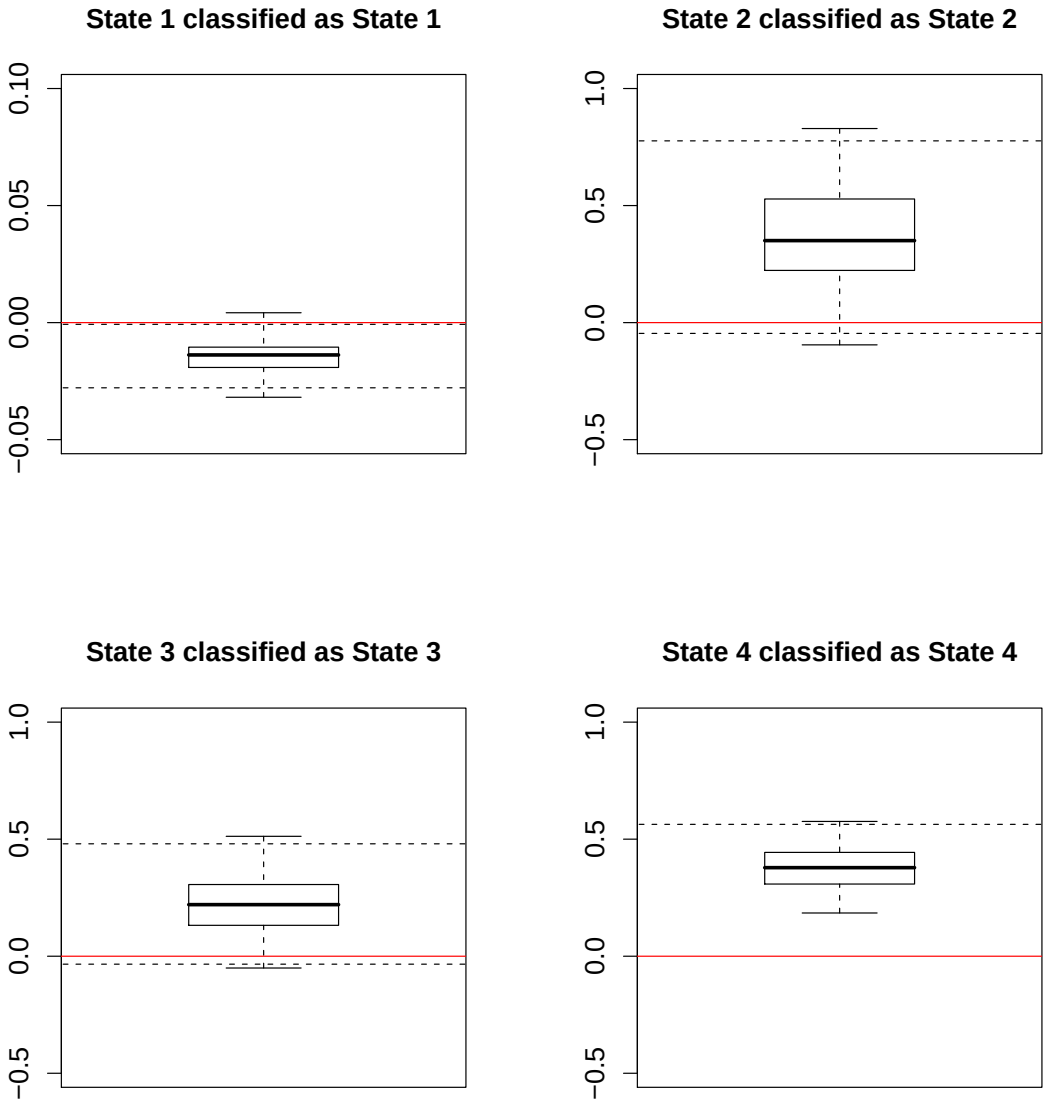

Figure 4.3: Confusion matrices of true states versus inferred states were computed for 100 simulations. The differences between the diagonal elements in the confusion matrices of the two models were divided by their average counts, to obtain the percent differences. Solid lines represent the medians. Dashed lines represent 2 SDs away from the means.

4.3 Nonhomogeneous hidden Markov model for multiple proteins

We now present a generalized version of the nonhomogeneous HMM for analyzing multiple proteins. Consider a situation in which P DNA-binding proteins form the complex of interest. Suppose that at least one ChIP-chip experiment was performed for each protein. Let $p \in \{1, ..., P\}$ index the proteins. Let M_p denote the number of replicate experiments performed for protein p, and let $m \in \{1, ..., M_p\}$ index the replicates. We extend the notations used in Section 3.2 to describe the multiple-protein model. Let T denote the total number of observed positions after integration of the tiling array designs. Let t_k, for $k \in \{1, ..., T\}$, denote the genomic positions of the observations. Let $\Delta_k = t_{k+1} - t_k$ denote the number of single-base steps (i.e. base pairs) between two adjacent observations. Let $x_p(t_k) \in \{0, 1\}$ denote the hidden state of protein p at position t_k, in the single-protein model. Let $y_{pm}(t_k)$ denote the m-th observation of protein p at position t_k. We assume that each protein emits observations independently of the other proteins, conditional on the hidden states. Thus we have one set of univariate Gaussian emission parameters for each protein and each tiling array design. The index for the tiling array designs is omitted with the understanding that the emission parameters are design-specific.

$$y_{pm}(t_k)|x_p(t_k) = 0 \sim \mathcal{N}(\mu_{p0}, \sigma^2_{p0})$$
$$y_{pm}(t_k)|x_p(t_k) = 1 \sim \mathcal{N}(\mu_{p1}, \sigma^2_{p1})$$

A basic assumption of our model is that the binding status of each protein follows a two-state Markov chain, with the state being either 0 if unbound or 1 if bound. The combined binding status of the protein complex follows a Markov chain with 2^P states. These states can be represented by strings of 0's and 1's, which indicate the binding statuses of the individual components. Thus the vector of initial probabilities, denoted by π, has length 2^P. The one-step transition matrix, denoted by $\mathbf{A}$, has dimension 2^P by 2^P. Figure 4.4 illustrates the four hidden states of a complex with two proteins. In this case, the initial distribution and the transition matrix can be written as:

- Initial distribution:
$$\pi = (\pi_{00}, \pi_{01}, \pi_{10}, \pi_{11})$$
- Transition matrix:
$$\mathbf{A} = \begin{pmatrix} a_{00,00} & a_{00,01} & a_{00,10} & a_{00,11} \\ a_{01,00} & a_{01,01} & a_{01,10} & a_{01,11} \\ a_{10,00} & a_{10,01} & a_{10,10} & a_{10,11} \\ a_{11,00} & a_{11,01} & a_{11,10} & a_{11,11} \end{pmatrix}$$

In order to obtain initial estimates for the parameters in the multiple-protein model, we first need to fit one single-protein model for each component of the protein complex. The emission parameters estimated by fitting the individual model of each protein can be

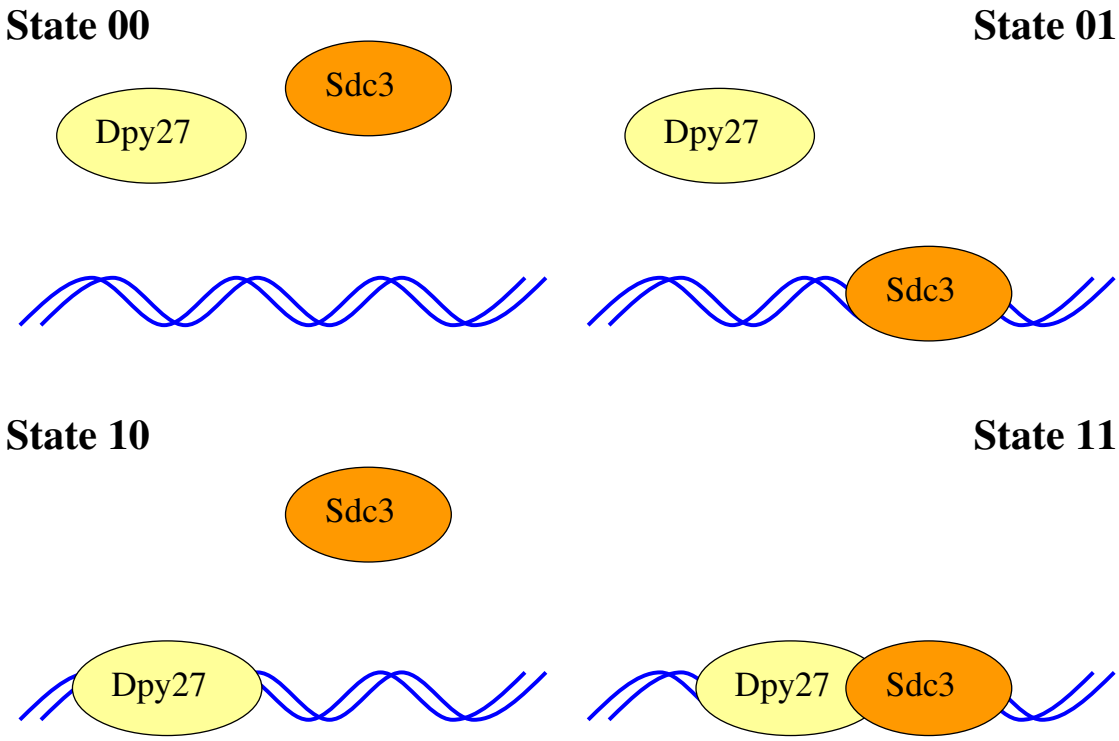

Figure 4.4: Four Hidden States of the Two-Protein Model

carried over as the emission parameters in the multiple-protein model. Since the emission distribution of each protein is independent of the other proteins, no further updates of the emission parameters are needed when fitting the multi-protein model. For a given protein, each observed position can be assigned to either State 0 or State 1 by thresholding the posterior probabilities. To obtain an initial estimate of π, we can simply count the positions where the binding statuses of the individual proteins satisfy the combination specified for each state. For example, the initial probability of State 01 can be estimated as the number of positions with Protein 1 in the unbound state and Protein 2 in the bound state, divided by the total number of observed positions. To obtain an initial estimate of $\mathbf{A}$, we make a simplifying assumption that the Markov chains of the individual proteins are independent of each other. Then the transition matrix for the protein complex can be written as the Kronecker product of the transition matrices of the individual proteins. Again, let us consider the two-protein example for illustration. Let $\mathbf{B}$ denote the transition matrix of Protein 1, and let $\mathbf{C}$ denote the transition matrix of Protein 2. The transition matrix of the complex can be estimated as $\mathbf{A} = \mathbf{B} \bigotimes \mathbf{C}$.

$$\mathbf{B} = \begin{pmatrix} b_{00} & b_{01} \\ b_{10} & b_{11} \end{pmatrix}$$

$$\mathbf{C} = \begin{pmatrix} c_{00} & c_{01} \\ c_{10} & c_{11} \end{pmatrix}$$

$$\mathbf{A} = \mathbf{B} \bigotimes \mathbf{C} = \begin{pmatrix} b_{00}c_{00} & b_{00}c_{01} & b_{01}c_{00} & b_{01}c_{01} \\ b_{00}c_{10} & b_{00}c_{11} & b_{01}c_{10} & b_{01}c_{11} \\ b_{10}c_{00} & b_{10}c_{01} & b_{11}c_{00} & b_{11}c_{01} \\ b_{10}c_{10} & b_{10}c_{11} & b_{11}c_{10} & b_{11}c_{11} \end{pmatrix}$$

Please note that we do not expect the Markov chains of the individual proteins to be independent of each other. This is because proteins that function in the same complex are more likely to bind jointly at the same location than separately at different locations. Nevertheless, the estimates obtained based on the independence assumption are adequate for initializing the modified Baum-Welch algorithm. We use the Kronecker product only to get the initial values of the transition probabilities. We then run the modified Baum-Welch algorithm described in Section 3.2 to refit the parameters. Recall that the emission parameters for any given experiment are determined by 1) the binary state of the protein and 2) the tiling array design. So the only distinction between the single-protein and multi-protein models is that the latter involves a higher dimensional Markov chain. We also found that the estimates of the Markov parameters generally converge within 10 iterations of the Baum-Welch updates.

4.4 Simulation study for two proteins

In Section 3.3, we described a simulation study of the two-state nonhomogeneous HMM for one protein. A four-state model, as illustrated in 4.4, is required to analyze the joint binding sites of two proteins. The hidden states are: (00) not bound by either one of the two

proteins; (01) not bound by the first protein but bound by the second protein; (10) bound by the first protein but not bound by the second protein; (11) bound by both of the two proteins. The algorithm for fitting this four-state model also involves a linear approximation to the exponentials of the one-step transition matrix. Although we saw in Section 3.3 that the approximation errors are tolerable for the two-state model, we cannot be certain about the four-state model because of its increased complexity. Thus we repeated the simulation study for the two-protein model.

We selected the same 1 MB region of Chromosome X as the one used in 3.3 for this simulation study. The chromosomal coordinates of the probes in the selected region were recorded for the three tiling array designs. These coordinates dictate where the observations are emitted. The hidden states of the two proteins were generated for every base position in this region, according to a four-state Markov chain. The length of the Markov chain is 1 million bases. Following the notations in Section 4.3, let π and $\mathbf{A}$ denote the initial probabilities and the one-step transition matrix of the Markov chain, respectively. Tiling array data were simulated according to the state-conditional Gaussian distributions. We assumed that the two proteins emit observations independently of each other, conditional on the hidden states. This assumption was shown to be acceptable in Section 4.2. Let μ_{p0} and σ^2_{p0} denote the mean and variance parameters for observations of the p-th protein in the unbound state. Let μ_{p1} and σ^2_{p1} denote the mean and variance parameters for observations of the p-th protein in the bound state. To simulate peaks of variable lengths, a different value of the mean parameter μ_{p1} was chosen for each peak. The variance parameter σ^2_{p1} was fixed at the same value for all peaks of the same protein. Let μ^*_{p1} denote another parameter with a fixed value, which is unique for each protein. The following list summarizes how the emission parameters were determined at each position.

1. If the current position is in State 00 (not bound by either protein), then use the emission parameters μ_{p0} and σ^2_{p0} for $p \in \{1, 2\}$.

2. If the current position is not in State 00, and it is different from the state of the previous position, then choose a new mean parameter for each protein that has a new peak. If it is in State 01 (not bound by the first protein but bound by the second protein), choose a new mean parameter for the second protein ($p = 2$). If it is in State 10 (bound by the first protein but not by the second protein), choose a new mean parameter for the first protein ($p = 1$). If it is in State 11 (bound by both of the two proteins), then choose a new mean parameter for each of the two proteins separately.

$$\mu_{p1} \sim \text{Uniform}(\mu^*_{p1} - 0.5 \times \sigma_{p1}, \mu^*_{p1} + 0.5 \times \sigma_{p1})$$

   Notice that the expected value of μ_{p1} is μ^*_{p1}. Choose the variance parameter σ^2_{p1}, which has a fixed value for all peaks.

3. If the current position is not in State 00, and it is the same as the state of the previous position, then continue to generate observations using the same emission parameters as those used for the previous position.

Design	Unbound μ_{p0}	Unbound σ_{p0}	Bound μ_{p1}	Bound σ_{p1}
1	0.124	0.151	0.658	0.195
2	0.170	0.128	0.675	0.178
3	0.053	0.121	0.357	0.161

Table 4.6: Emission parameters for Protein 1

Design	Unbound μ_{p0}	Unbound σ_{p0}	Bound μ_{p1}	Bound σ_{p1}
1	-0.001	0.153	0.927	0.254
2	-0.049	0.115	0.299	0.179
3	-0.005	0.132	0.421	0.189

Table 4.7: Emission parameters for Protein 2

The values of μ_{p0}, σ^2_{p0}, μ^*_{p1} and σ^2_{p1} depend on the tiling array design. The probe design also dictates whether an observation is emitted at any particular base position. To emulate the setting of the wild type Dpy-27 data, we generated 3 replicates in each simulated data set, with one replicate coming from each design. Under this setting, we simulated 100 data sets using the following parameters. These parameters were obtained by fitting the four-state nonhomogeneous HMM to the wild type Dpy-27 and Sdc-3 data for the 1 MB region selected from Chromosome X.

- Markov Chain Stationary Distribution

$$\pi = (0.772, 0.035, 0.058, 0.135)$$

- Markov Chain Transition Matrix

$$\mathbf{A} = \begin{pmatrix} 0.9988 & 0.0006 & 0.0004 & 0.0003 \\ 0.0126 & 0.9857 & 0.0007 & 0.0010 \\ 0.0048 & 0.0005 & 0.9929 & 0.0018 \\ 0.0017 & 0.0002 & 0.0007 & 0.9974 \end{pmatrix}$$

Each simulated data set was analyzed using the algorithm described in Section 4.3. The parameter estimates obtained from the 100 simulations are summarized as boxplots in Figures 4.5 to 4.12. In each panel of these figures, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates. Figures 4.5 to 4.10 show the emission parameters for each tiling array design separately. The estimates of μ_{p0}, μ_{p1} and σ_{p0} fluctuate around their true values, as expected.

The estimates of σ_{p1} are consistently larger than the true value of σ_{p1}. This is because the simulation model is slightly more complex than the assumptions of the nonhomogeneous HMM. To generate peaks of variable heights, a different value of μ_{p1} was chosen randomly to simulate the tiling array data. Whereas a mixture of Gaussians was generated in the simulations, the nonhomogeneous HMM assumes only a single Gaussian distribution for the bound state. Thus the estimates of σ_{p1} were inflated. Figure 4.11 shows the initial distribution of the hidden states. The true values of the initial probabilities are within the Mean +/− 2 SDs window of the estimates. Figure 4.12 shows the transition probabilities of the hidden Markov chain. There appears to be a bias in the estimation of the transition matrix, which deserves a closer examination.

We saw in Section 3.3 that the bias in the estimation of the two-state transition matrix is related to the linear approximation of matrix exponentials. This approximation works better for smaller step sizes than larger step sizes. When we used smaller step sizes in the simulations, we saw a reduction in the bias. To investigate this for the four-state model, we also repeated the simulations of the four-state model with progressively smaller spacing between the probes. Since the purpose is to investigate the effects of step sizes, we used a hypothetical design with uniform spacing between the probes in the next set of simulation experiments. Each simulated data set contained only one set of observations for each protein. For the first experiment, we set the spacing between adjacent probes to 20 bp. For the second experiment, we set the spacing to 10 bp. For the third experiment, we set the spacing to 2 bp. The computational complexity of the forward-backward algorithm is linear in the number of observations. In order to keep the running time within reasonable limits, we varied the length of the Markov chain to achieve the same number (50,000) of observations in each experiment. Figure 4.13 shows the results of the first experiment, with observations emitted at every 20 base pairs. Figure 4.14 shows the results of the second experiment, with observations emitted at every 10 base pairs. Figure 4.15 shows the results of the third experiment, with observations emitted at every other base pair. Again, bias in the estimates was progressively reduced when we decreased the step size between the adjacent observations. Thus we conclude that the bias in the estimation of the four-state transition matrix is also related to the linear approximation of matrix exponentials.

We then looked at how the bias in the estimation of the transition matrix might affect our inferences about the hidden states. For each simulated data set, we compared the true hidden states with the inferred states from our algorithm. For each simulated data set, we tabulated the number of correctly inferred positions and the number of incorrectly inferred positions in a confusion matrix. To account for differences in the state prevalences, we reported each entry as a percentage of the number of observation in each true state. To summarize across 100 simulations, we drew one boxplot for each entry in the confusion matrix. A good algorithm should lead to confusion matrices with the diagonal entries close to 100%, and the off-diagonal entries nearly zero. Figure 4.16 shows the results of the experiment with observations emitted at the same chromosomal coordinates as the real data. The error rates were below 7% in all of the 100 simulations. Figure 4.17 shows the results of the experiment with observations emitted at every 20 base pairs. At $\Delta_k = 20$, the error rates were below 5%. Figure 4.18 shows the results of the experiment with observations emitted at every 10

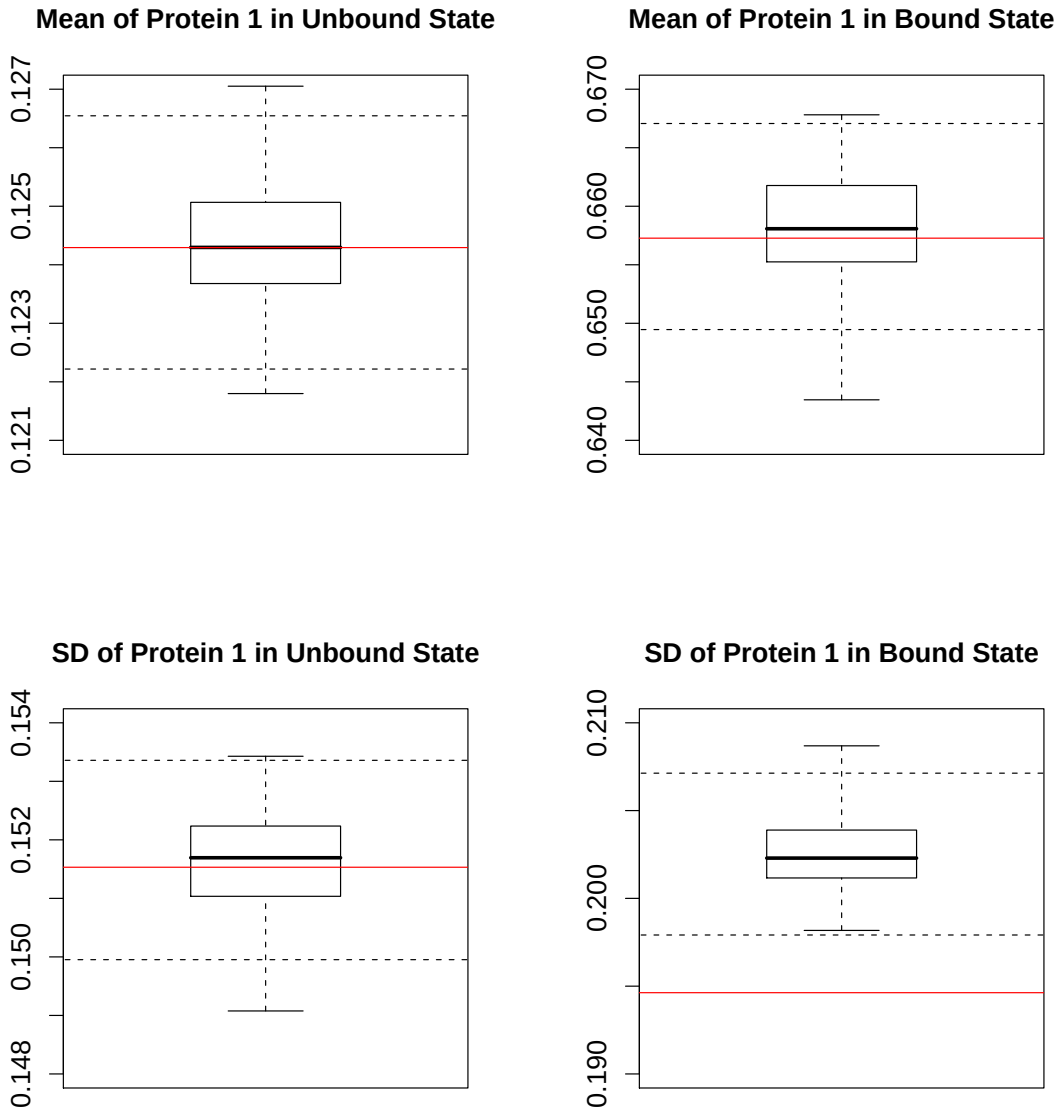

Figure 4.5: Emission Parameters (Protein 1, Design 1): 100 simulations of tiling array data were generated according to the chromosomal coordinates of the probes in a 1 MB region of Chromosome X. Each simulated data set contained 3 replicates for each protein, with one replicate from each design. Estimates of the emission parameters for Protein 1 on Design 1 are summarized as boxplots. In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates.

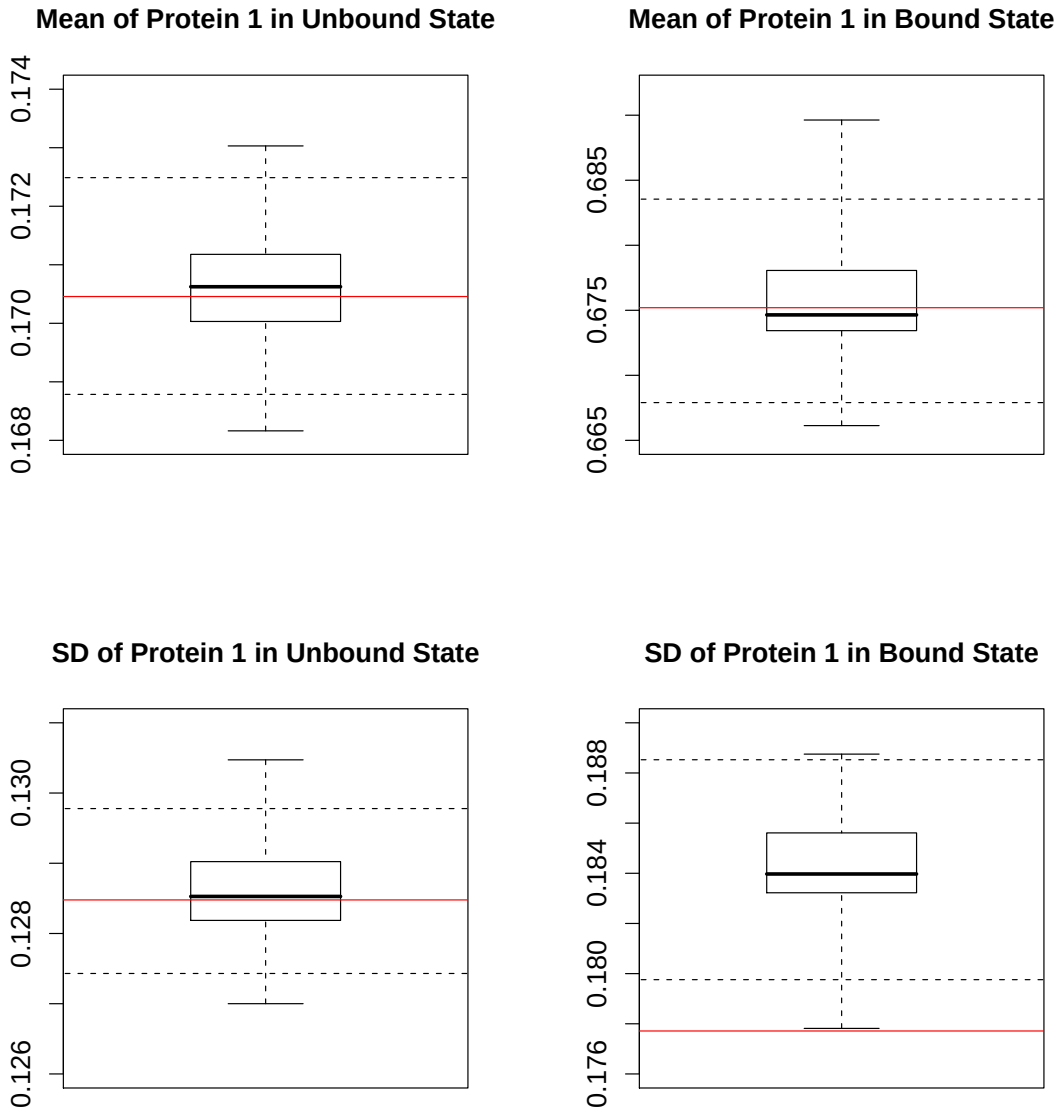

Figure 4.6: Emission Parameters (Protein 1, Design 2): 100 simulations of tiling array data were generated according to the chromosomal coordinates of the probes in a 1 MB region of Chromosome X. Each simulated data set contained 3 replicates for each protein, with one replicate from each design. Estimates of the emission parameters for Protein 1 on Design 2 are summarized as boxplots. In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates.

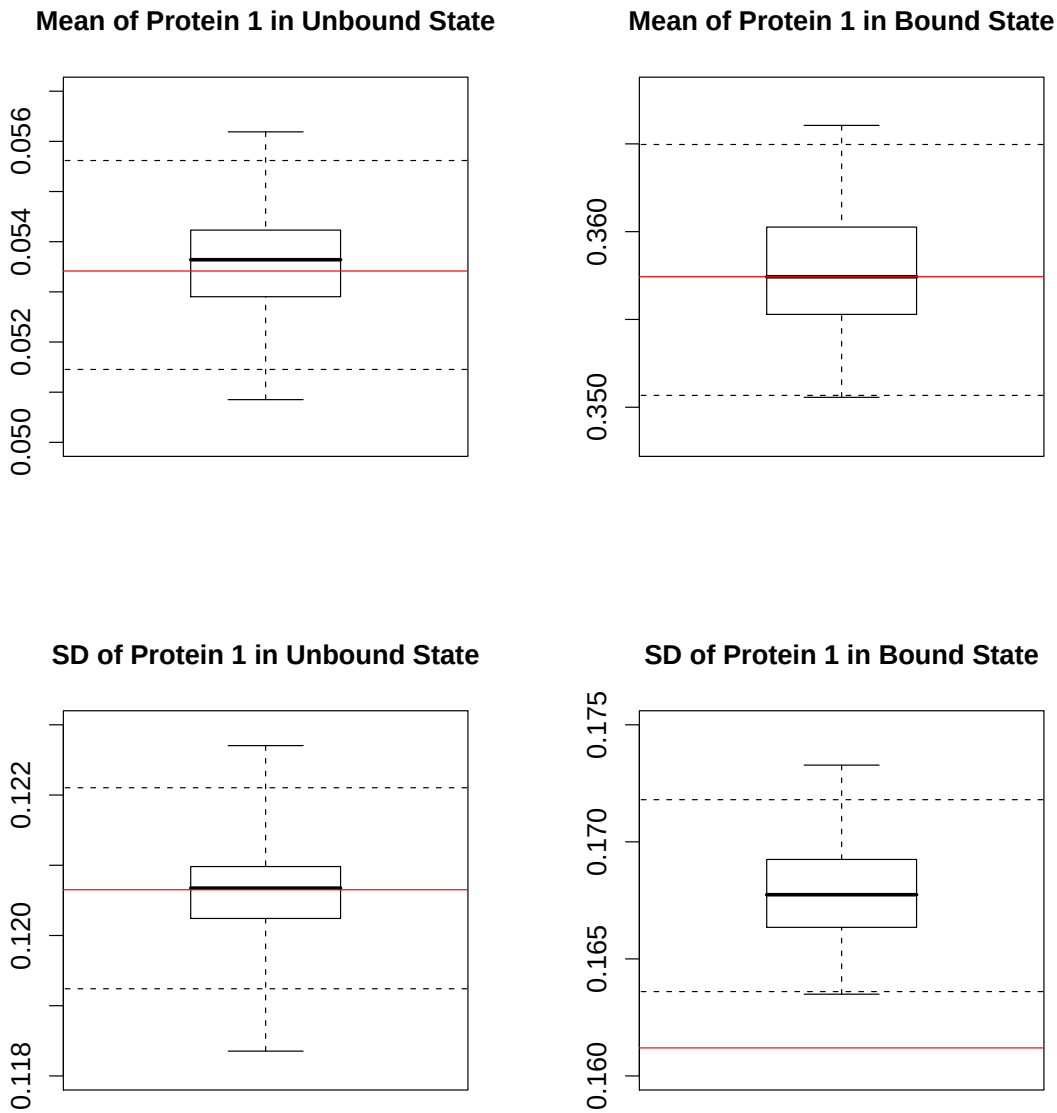

Figure 4.7: Emission Parameters (Protein 1, Design 3): 100 simulations of tiling array data were generated according to the chromosomal coordinates of the probes in a 1 MB region of Chromosome X. Each simulated data set contained 3 replicates for each protein, with one replicate from each design. Estimates of the emission parameters for Protein 1 on Design 3 are summarized as boxplots. In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates.

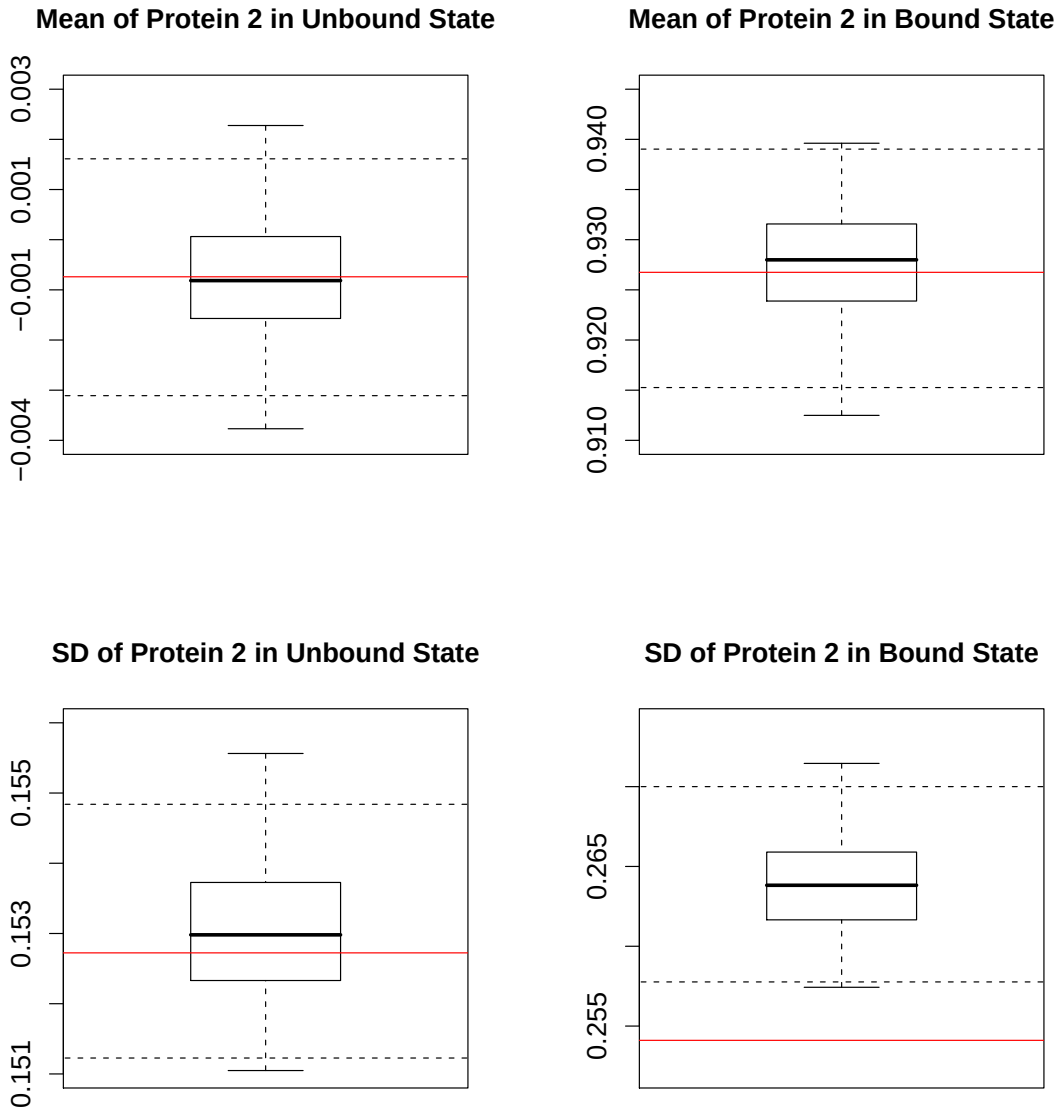

Figure 4.8: Emission Parameters (Protein 2, Design 1): 100 simulations of tiling array data were generated according to the chromosomal coordinates of the probes in a 1 MB region of Chromosome X. Each simulated data set contained 3 replicates for each protein, with one replicate from each design. Estimates of the emission parameters for Protein 2 on Design 1 are summarized as boxplots. In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates.

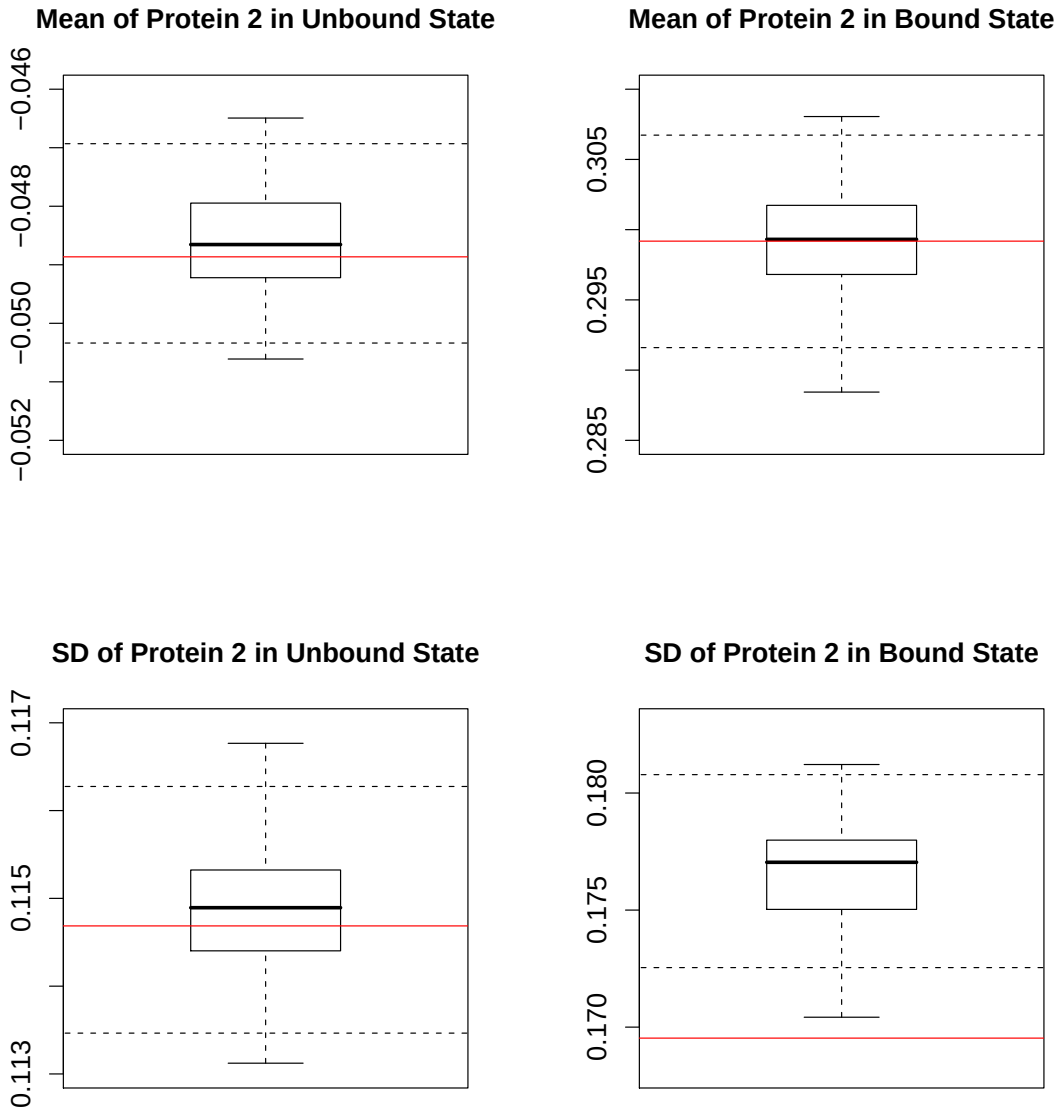

Figure 4.9: Emission Parameters (Protein 2, Design 2): 100 simulations of tiling array data were generated according to the chromosomal coordinates of the probes in a 1 MB region of Chromosome X. Each simulated data set contained 3 replicates for each protein, with one replicate from each design. Estimates of the emission parameters for Protein 2 on Design 2 are summarized as boxplots. In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates.

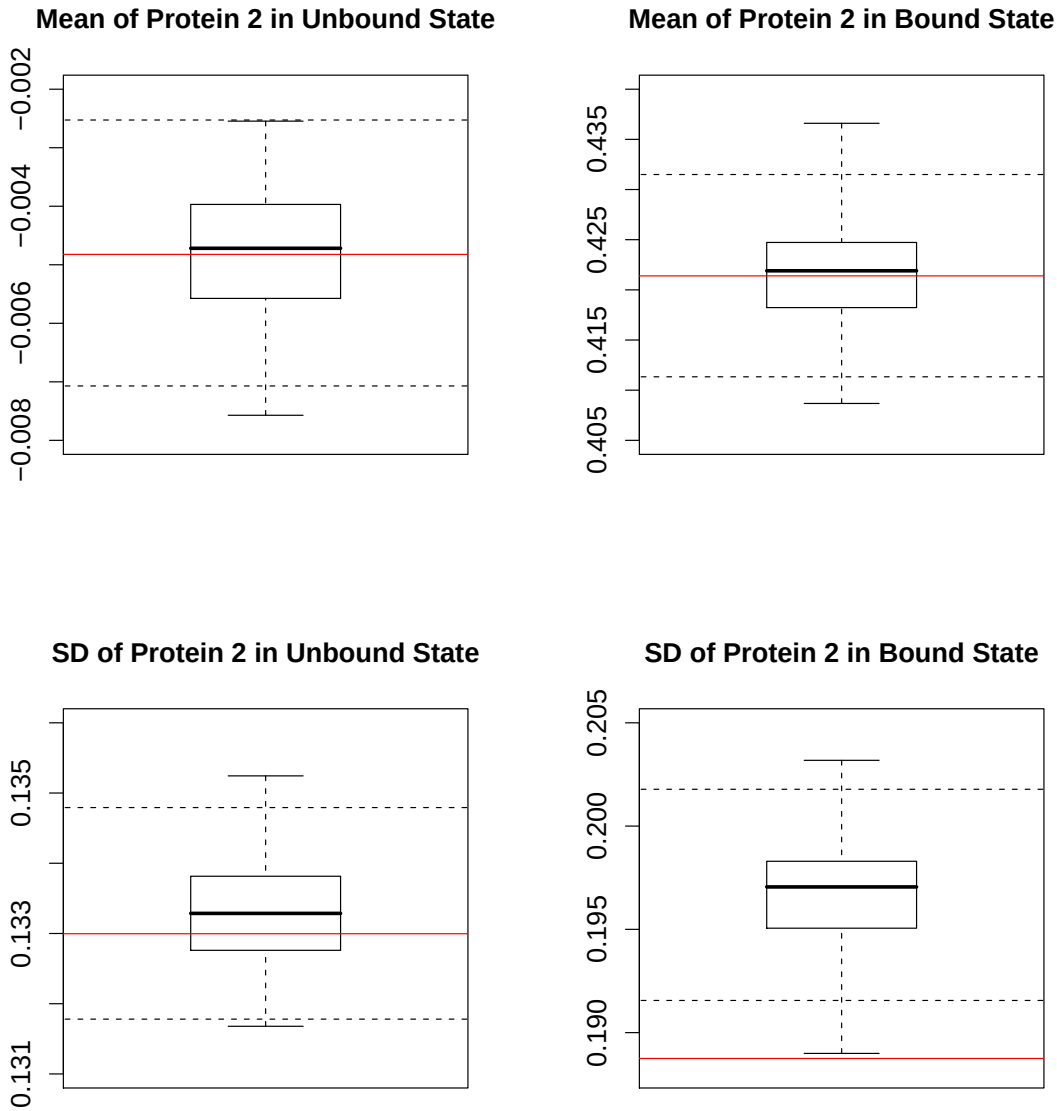

Figure 4.10: Emission Parameters (Protein 2, Design 3): 100 simulations of tiling array data were generated according to the chromosomal coordinates of the probes in a 1 MB region of Chromosome X. Each simulated data set contained 3 replicates for each protein, with one replicate from each design. Estimates of the emission parameters for Protein 2 on Design 3 are summarized as boxplots. In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates.

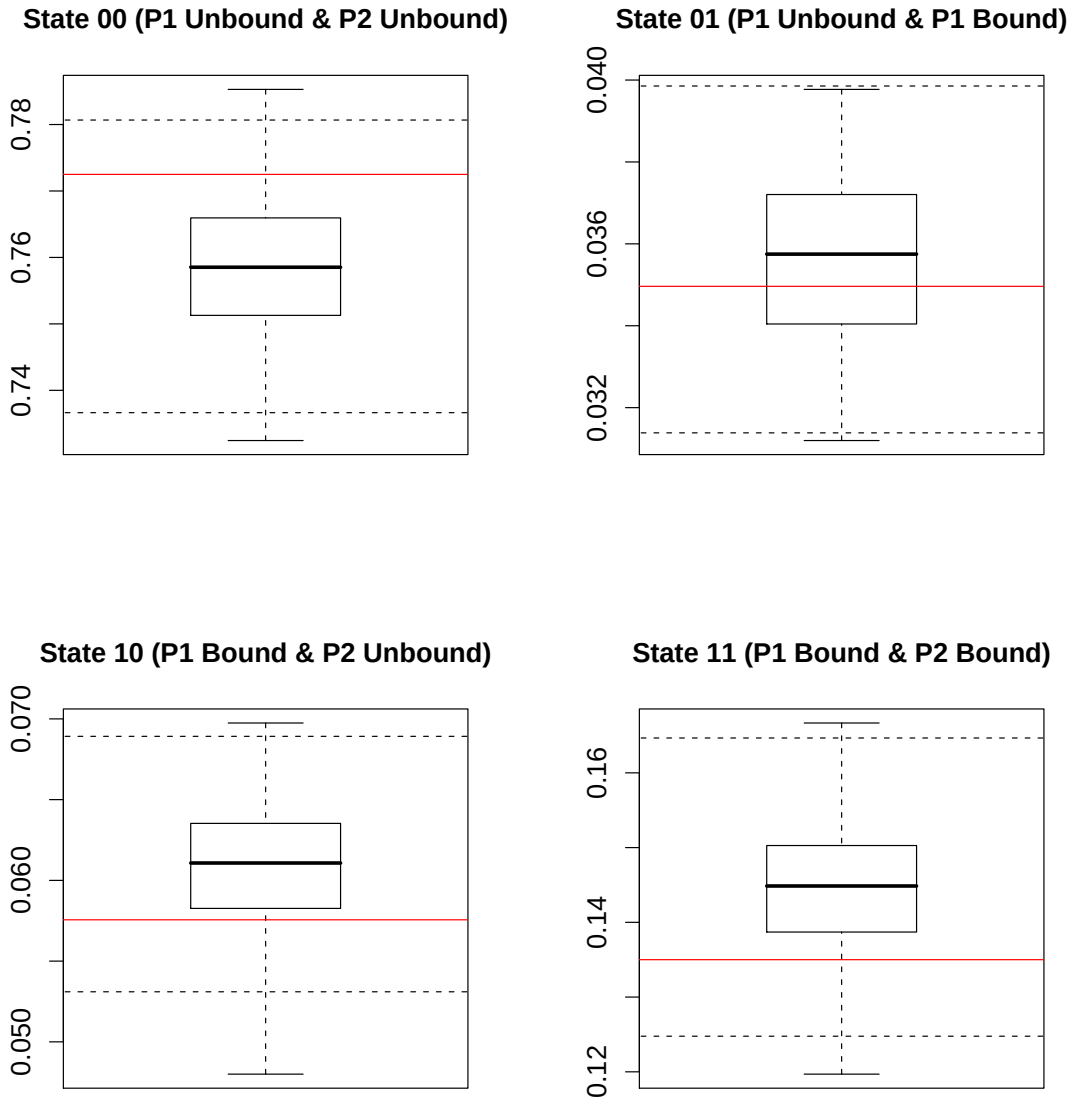

Figure 4.11: Stationary Distribution: 100 simulations of tiling array data were generated according to the chromosomal coordinates of the probes in a 1 MB region of Chromosome X. Each simulated data set contained 3 replicates for each protein, with one replicate from each design. Estimates of the initial probabilities are summarized as boxplots. In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates.

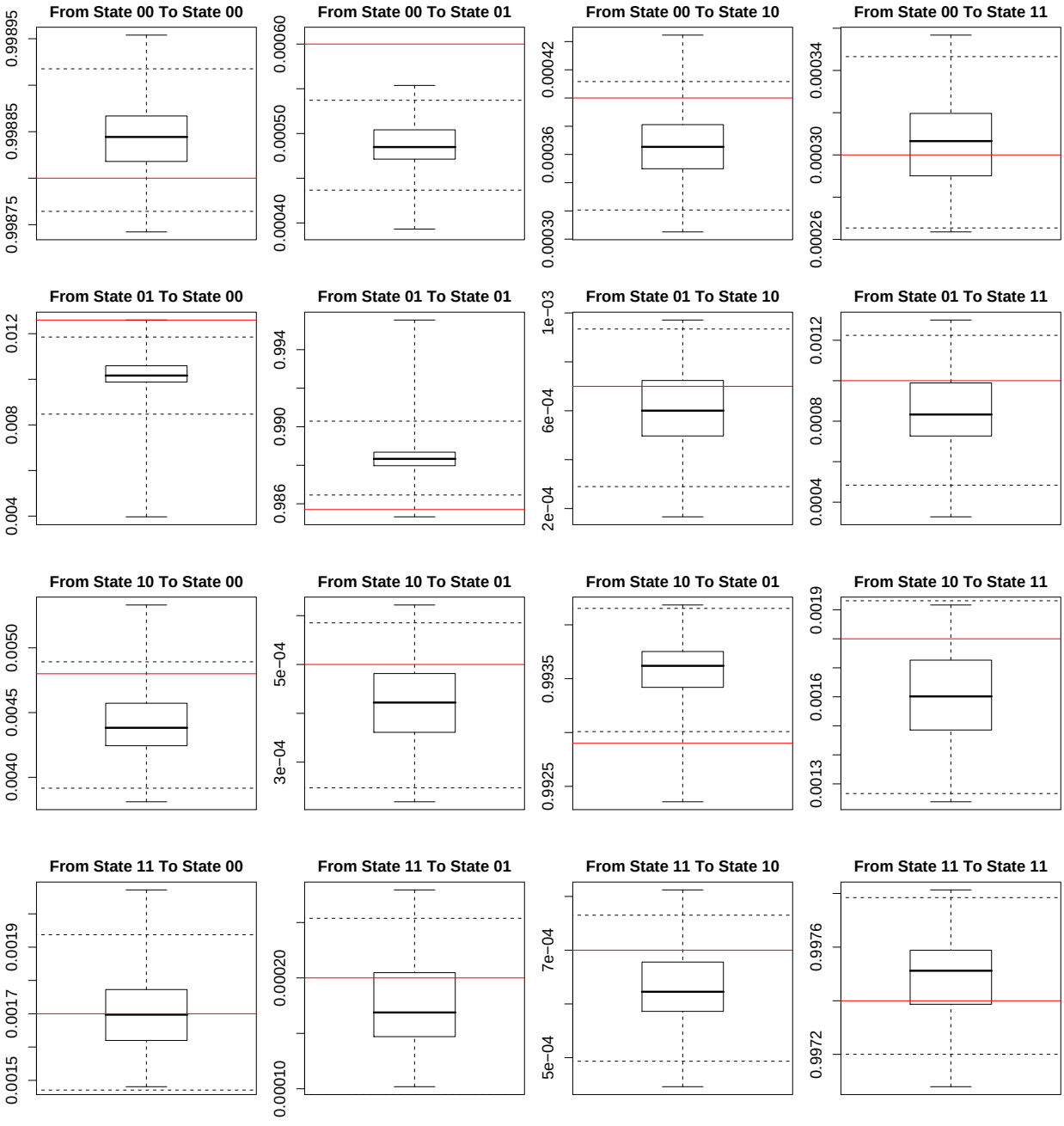

Figure 4.12: Transition Matrix (variable Δ_k): In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates.

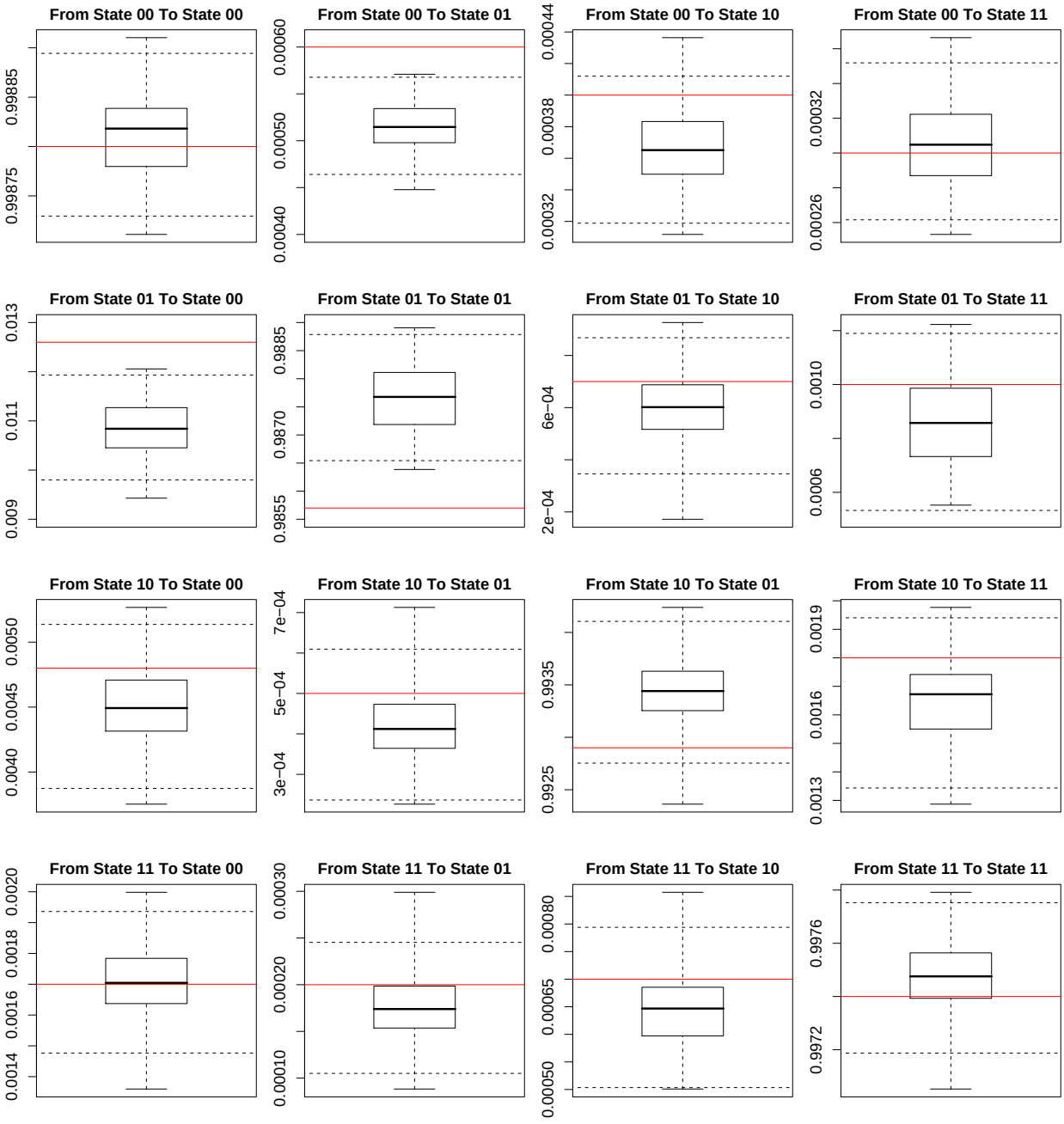

Figure 4.13: Transition Matrix ($\Delta_k = 20$): In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates.

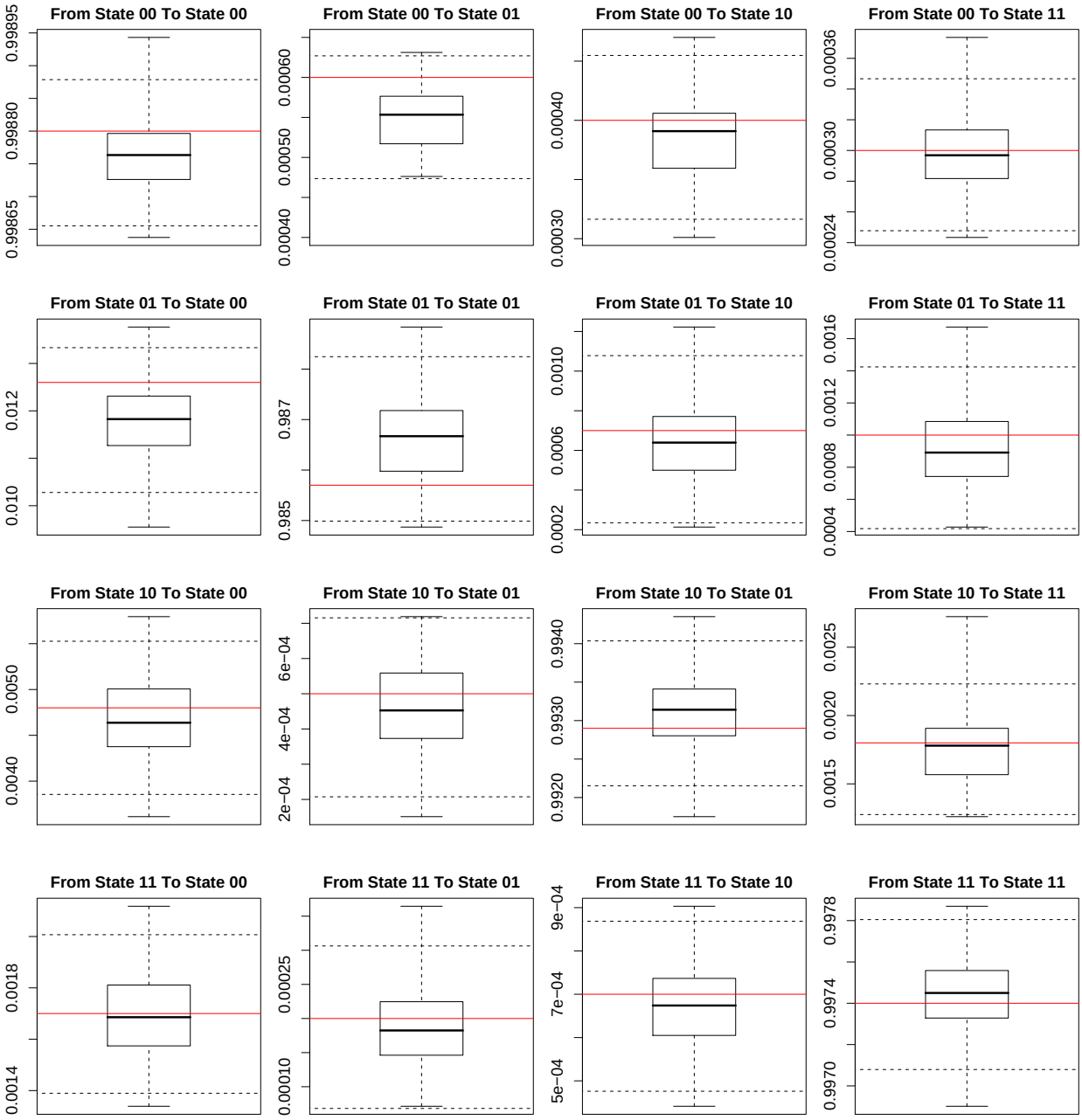

Figure 4.14: Transition Matrix ($\Delta_k = 10$): In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates.

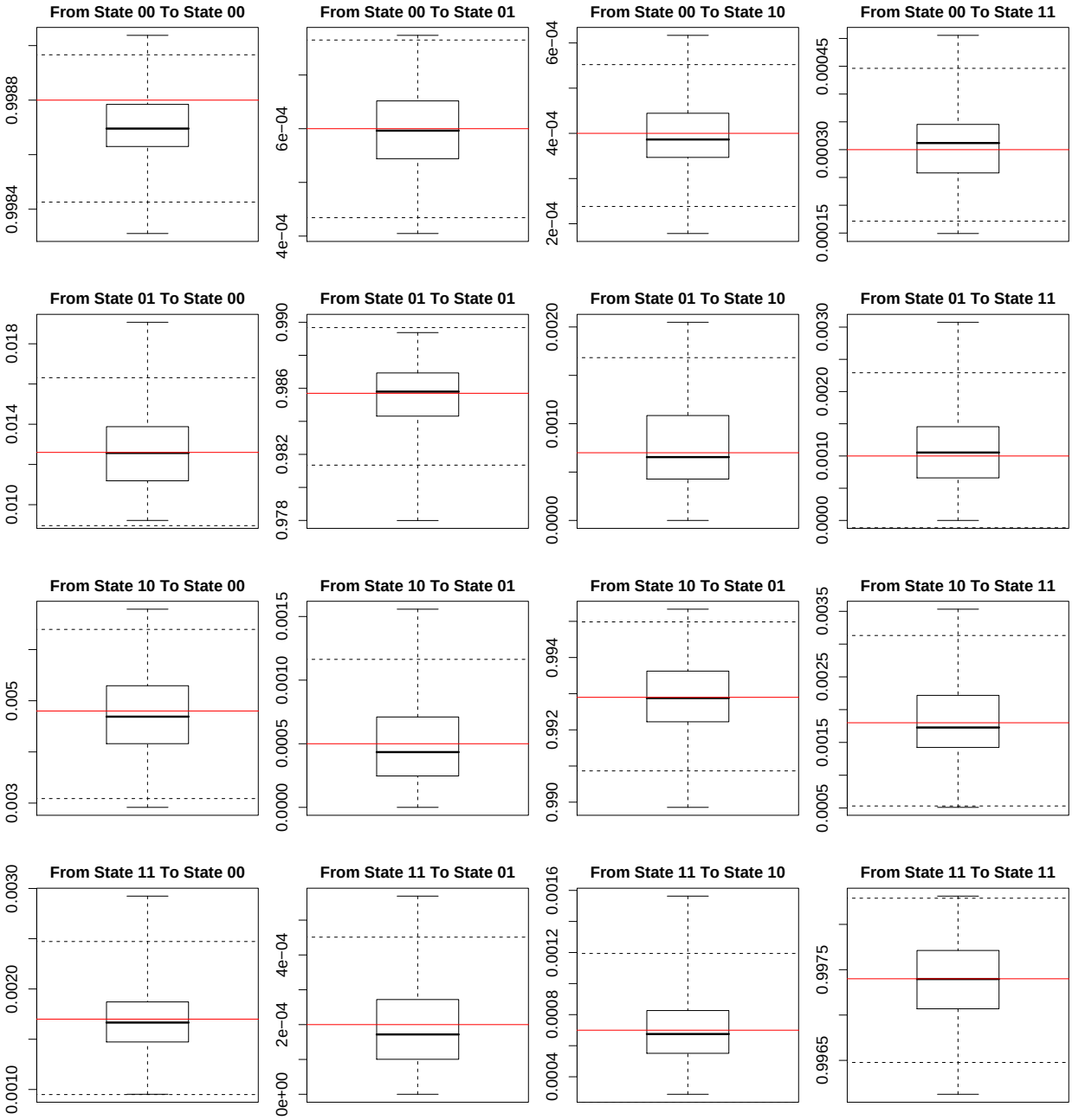

Figure 4.15: Transition Matrix ($\Delta_k = 2$): In each panel, the true parameter value is represented by a red horizontal line. The inner-quartile range of the 100 parameter estimates is represented by a box, with a thick line drawn at the median. The whiskers extend to the extreme values of the parameter estimates. The black dashed lines represent Mean +/− 2 SDs of the parameter estimates.

base pairs. At $\Delta_k = 10$, the error rates were below 3%. Figure 4.19 shows the results of the experiment with observations emitted at every other base pair. At $\Delta_k = 2$, the error rates were below 1%. As the spacing between the observations was decreased, the error rates also decreased. Again, we found that the errors in state inferences are associated with the bias in the estimation of the transition matrix. The linear approximation works better when the transition matrix is closer to the identify matrix. The majority of the errors occur in State 01 and State 10. This is because the 2nd and 3rd diagonal elements of the transition matrix are smaller (less close to one) than the other two diagonal elements. Since the goal of the two-protein analysis is to identify the joint binding sites, we are mainly concerned with State 11. Under the setting of the real data, the error rates for inferences about State 11 were below 2% in all of the 100 simulations. Thus we conclude that the state inference errors associated with the biased estimation of the transition matrix are tolerable.

4.5 Comparison with the alternative approach

None of the existing peak-calling algorithms have the capacity for analyzing the ChIP-chip data of multiple proteins simultaneously. When the biologists need to identify the shared targets of two or more DNA binding proteins, they take the approach of cross-listing the binding sites identified from the individual proteins. In promoter studies that involve two or more transcription factors, the investigators often summarize the binding sites of each protein as a list of target genes. Then, the overlaps between the gene lists are illustrated using Venn Diagrams [39]. This approach has the limitation that the binding sites that cannot be mapped to genes are excluded from the analysis. A more general approach is to cross-list the genomic coordinates of the peaks identified for each protein. Rada-Iglesias et al. took this approach to compare the binding sites of the human upstream stimulatory factors USF1 and USF2 [54]. They first obtained a set of binding targets for each protein using a sliding window peak caller. They then calculated the center positions of all the peaks in the first set. An overlap was reported if any base within a fixed window around the center position was shared with some peaks in the second set. The window size was either 1 kb or 2 kb in each direction around the center position. A problem with this approach is that two peaks with marginal overlaps could easily be reported, when the experimental evidence for a joint binding site is weak. Hollenhorst et al. used a modified version of this approach to study three members of the ETS transcription factor family [33]. To identify regions that are dual-bound by a pair of proteins, they began by locating the center positions of all the peaks for the first protein. They then examined the p-values of all the probes from the second protein that lie within 1 kb from each center position of the first set. If at least one of the probes had a p-value of less than 0.001, then the region represented by this center position was reported as dual-bound. A more recent study by Boros et al. also adopted the approach of cross-listing the peaks from the first protein with the probe level statistics from the second proteins [10]. The advantage of cross-listing with the probe level statistics, as opposed to cross-listing with the peaks, is that regions with marginal overlaps are less likely to be reported. Among the three "best" existing algorithms for analyzing NimbleGen data, MA2C is the only one that provides the probe level statistics in its outputs. So we chose to

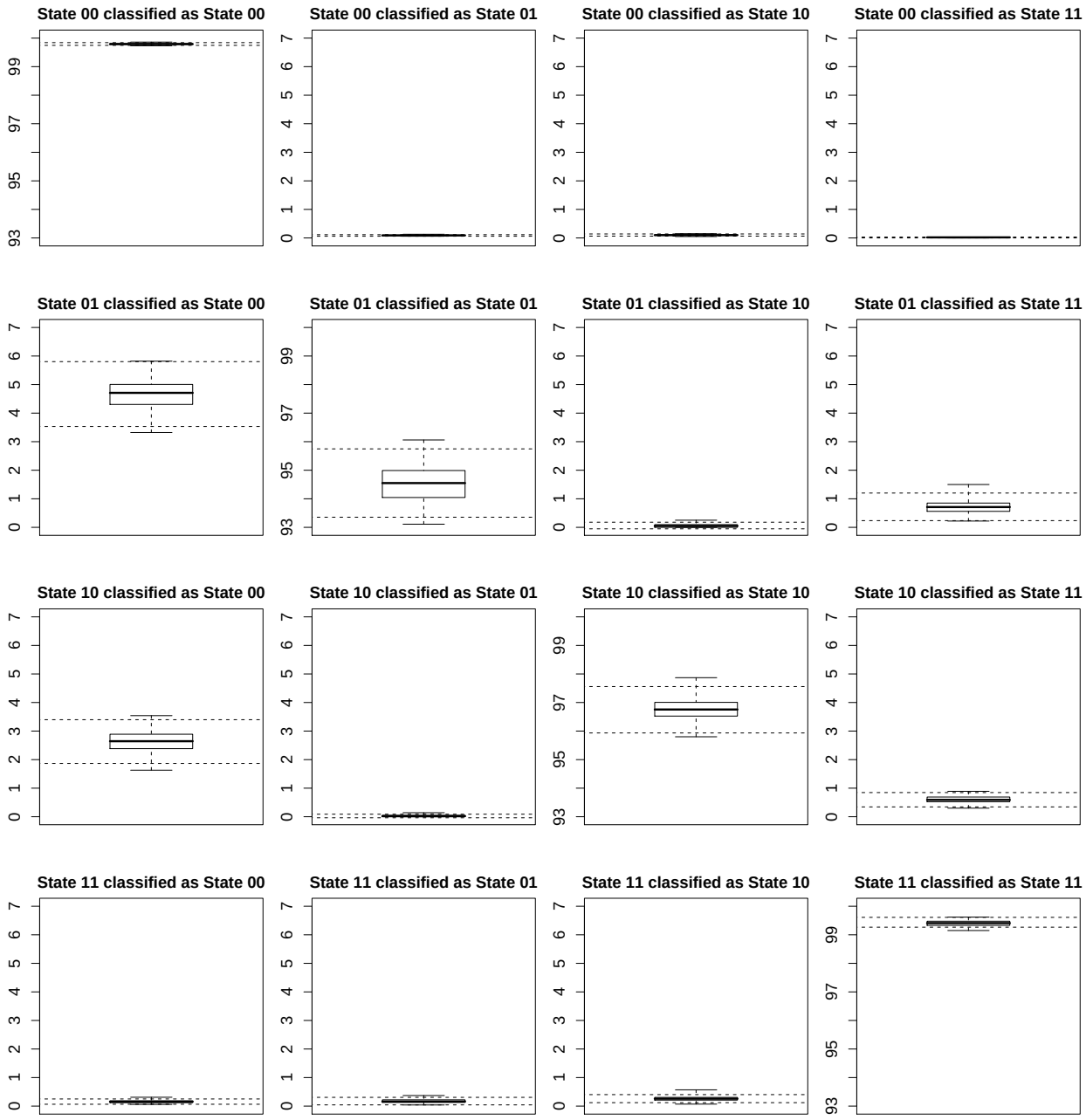

Figure 4.16: Confusion Matrix (variable Δ_k): A confusion matrix was tabulated for each simulated data set, and its entries were normalized into percentages of the true states. Each panel summarizes one entry in the confusion matrix across 100 simulations. The inner-quartile range is delineated by the borders of the box, with a thick line drawn at the median. The whiskers extend to the extreme values. The black dashed lines represent Mean +/− 2 SDs.

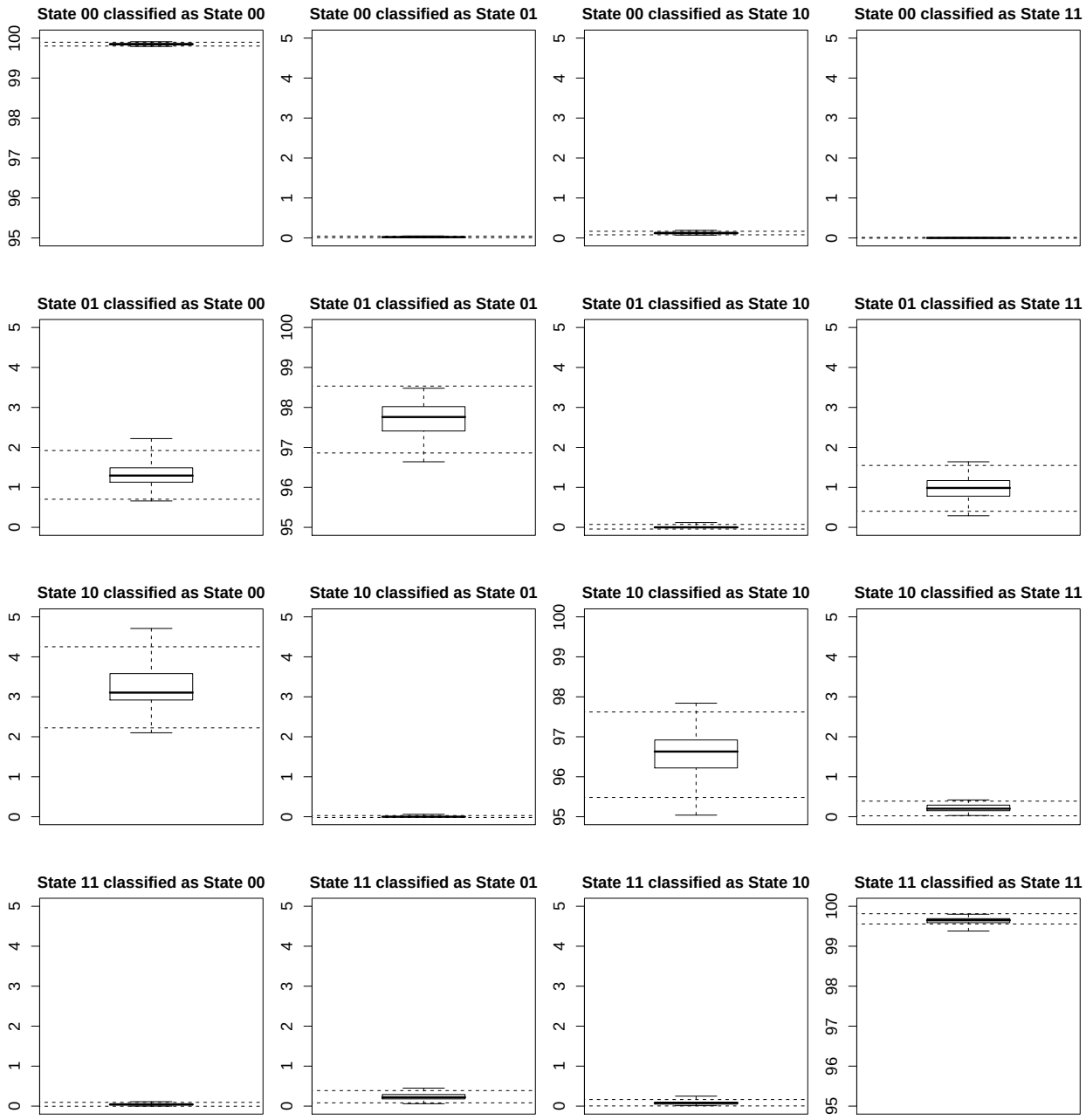

Figure 4.17: Confusion Matrix ($\Delta_k = 20$): A confusion matrix was tabulated for each simulated data set, and its entries were normalized into percentages of the true states. Each panel summarizes one entry in the confusion matrix across 100 simulations. The inner-quartile range is delineated by the borders of the box, with a thick line drawn at the median. The whiskers extend to the extreme values. The black dashed lines represent Mean +/− 2 SDs.

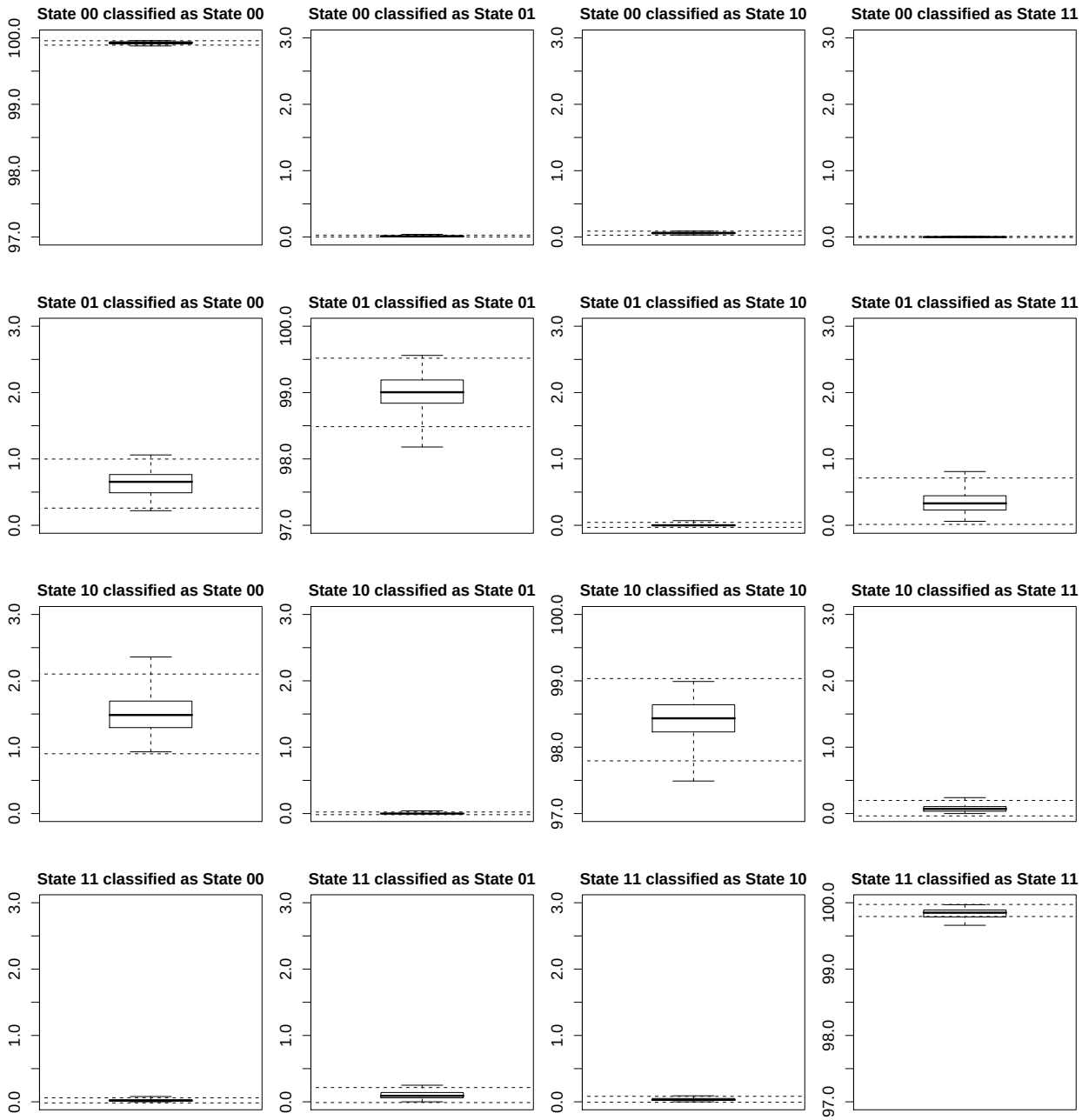

Figure 4.18: Confusion Matrix ($\Delta_k = 10$): A confusion matrix was tabulated for each simulated data set, and its entries were normalized into percentages of the true states. Each panel summarizes one entry in the confusion matrix across 100 simulations. The inner-quartile range is delineated by the borders of the box, with a thick line drawn at the median. The whiskers extend to the extreme values. The black dashed lines represent Mean +/− 2 SDs.

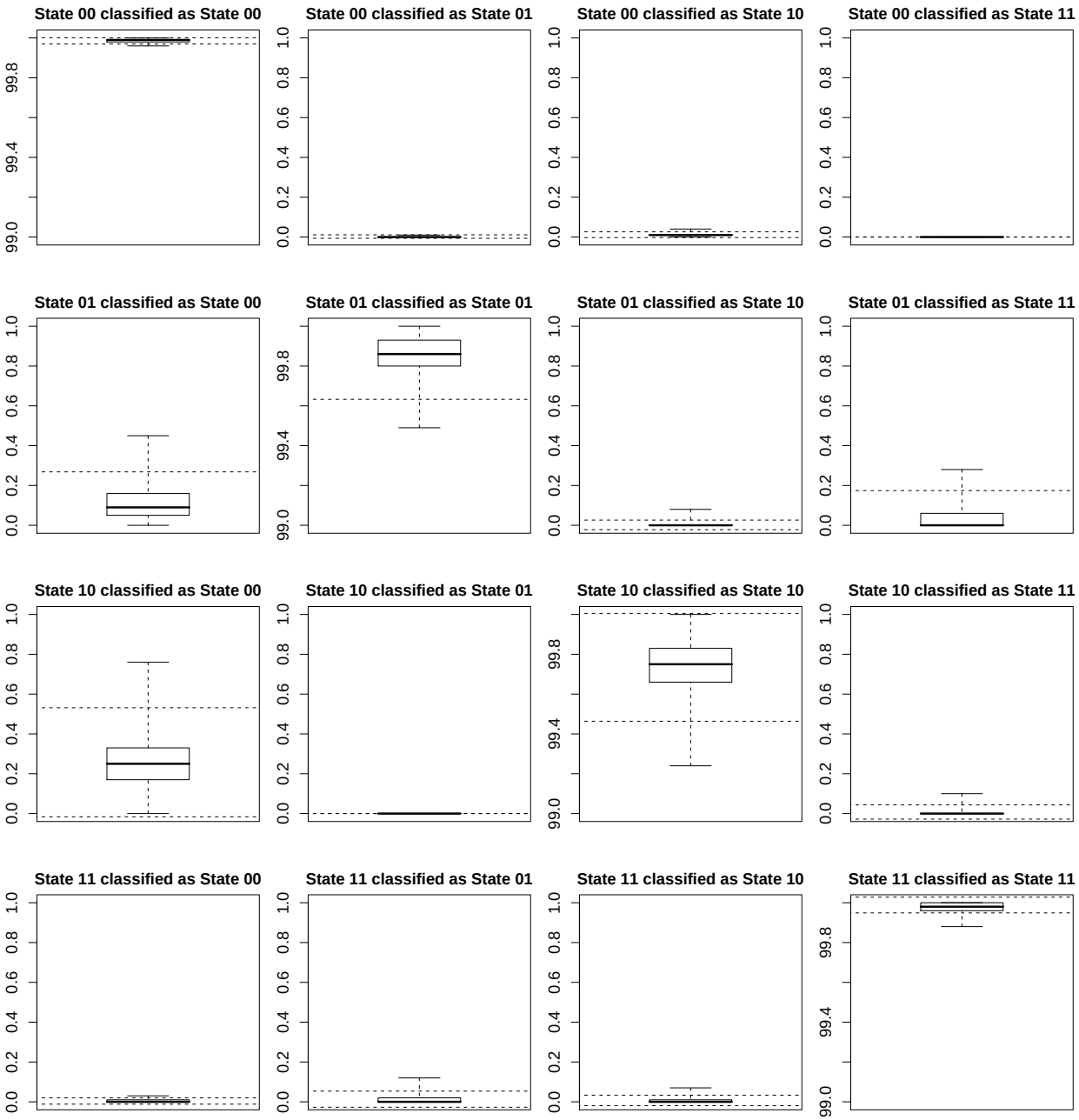

Figure 4.19: Confusion Matrix ($\Delta_k = 2$): A confusion matrix was tabulated for each simulated data set, and its entries were normalized into percentages of the true states. Each panel summarizes one entry in the confusion matrix across 100 simulations. The inner-quartile range is delineated by the borders of the box, with a thick line drawn at the median. The whiskers extend to the extreme values. The black dashed lines represent Mean +/− 2 SDs.

emulate this currently popular cross-listing approach using MA2C.

We devised the following two-protein analysis using MA2C, which involves 3 steps. In the first step, we merged all the peaks from replicate experiments to produce one set of peaks for each protein, under either the wild type or mutant condition. When merging the replicates, all of the non-overlapping peaks were kept, along with their original coordinates and MA2C scores. The overlapping peaks were trimmed such that their chromosomal coordinates in the merged set represent only the merged regions. The decision to trim the ends of overlapping peaks was based on the observation that MA2C peaks tend to be long, and that the ends often include background noise. The MA2C score of each merged peak was defined as the maximum of MA2C scores of the replicate experiments. This update of the MA2C scores reflected the thought that replication provides stronger experimental evidence than single experiments. In the second step, the merged set of peaks from the first protein was cross-listed with the probe level statistics of the second protein. We calculated the center positions of all the peaks in the merged set of the first protein. For each peak in the first set, we looked at the probe level statistics from all replicates of the second protein, which lie within 1 kb from the center position. If any of these probe level statistics had a p-value of less than 0.001, then the peak was considered a joint binding site. We performed cross-listing on Dpy-27 and Sdc-3 reciprocally, so two sets of cross-listed peaks were produced. In the third step, we merged the two sets of cross-listed peaks: 1) using Dpy-27 as the first protein and 2) using Sdc-3 as the first protein. The same criteria used in Step 1 were applied to merge the cross-listed peaks in Step 3. The result was one set of Dpy-27 and Sdc-3 joint peaks for each condition, with an MA2C score assigned to each peak.

We performed an ROC analysis to compare the MA2C cross-listing method with our proposed nonhomogeneous HMM method. The testing set of curated standards described in Section 3.4 was used to benchmark the comparisons. The joint peaks detected by the MA2C cross-listing method were ranked by the MA2C scores. The joint peaks detected by the nonhomogeneous HMM method were ranked by our shape-based peak scores, as described in Section 3.4. For each method, 100 different sets of peak calls were produced by thresholding the peaks scores at different values. A curated positive region that overlaped with any of the called peaks was considered a positive hit. A curated negative region that overlaped with any of the called peaks was considered a negative hit. Sensitivity and specificity were calculated according to the following equations.

$$\text{sensitivity} = \frac{\text{number of positive hits}}{\text{number of curated positive regions}}.$$

$$\text{specificity} = 1 - \frac{\text{number of negative hits}}{\text{number of curated negative regions}}.$$

Figure 4.20 shows some ROC curves comparing our nonhomogeneous HMM method with the MA2C cross-listing method described above. Because sensitivity and specificity were calculated based on the curated sets of positive and negative regions, they should only be used for making comparisons within each data set. The apparently higher values of sensitivity and specificity shown for the *smo-1* mutant condition should not be interpreted as better performances of the peak detection methods on this data set. They are merely

the consequence of having fewer borderline cases in the curated set of standard regions, as discussed in Section 3.5. Because the wild type data appear cleaner to the human eye, more borderline cases were included in curated standards. Thus the wild type curated standards have more discriminative power for comparing different methods. According to the ROC analysis based on wild type data, our nonhomogeneous HMM method has better performance than the currently popular cross-listing approach for analyzing two proteins. More importantly, the nonhomogeneous HMM method has a solid theoretical foundation and is easily scalable to more than two proteins. The cross-listing approach, on the other hand, quickly becomes intractable when the number of proteins gets large.

4.6 Issues in the application to real data

When applying the nonhomogeneous HMM to the *C. elegans* ChIP-chip data, we encountered a number of issues that are beyond the scope of the core methodology. In this section, we will discuss how we handled the issues that arose from: 1) preprocessing of the tiling array data; 2) estimation of the emission parameters on the genome wide scale; 3) combining replicate data in the postprocessing of peaks; 4) analysis of IgG control data. Finally, we will summarize the results of our analysis.

4.6.1 Preprocessing of tiling array data

Due to the complex nature of the microarray technology, experimental variations may easily introduce non-biological biases in the data. These biases include the different labeling efficiencies of the dyes used in two-color microarrays, variations between the spatial positions of the probes on a slide or between slides, non-uniformity in the hybridization within a slide, variations in the hybridization conditions between slides [59]. So preprocessing is an important first step in microarray data analysis. Quantile normalization, developed by Bolstad and Speed [8], has been the preferred method for preprocessing gene expression microarray data. A number of recently published tiling array analysis packages also chose quantile normalization as the preferred method [69, 42, 34]. So we decided to use quantile normalization in the preprocessing of the *C. elegans* ChIP-chip data. The traditional implementation of quantile normalization assumes that all the arrays in a given experimental set have the same design of probes. However, our data set contains tiling arrays of different probe designs. Thus we implemented a modified version of quantile normalization that can accommodate multiple designs. The traditional implementation builds a target distribution where the i-th ranked intensity is the average of all the i-th ranked intensities of all the arrays, with each array sorted in ascending order. A rank is computed for each probe on any given array. The normalized intensity of a particular probe is the intensity of its rank-equivalent probe on the target distribution. After normalization, each array should have the same distribution of probe level intensities as the target distribution. Our modified implementation differs from the traditional implementation in the way that the target distribution is built, which is explained below.

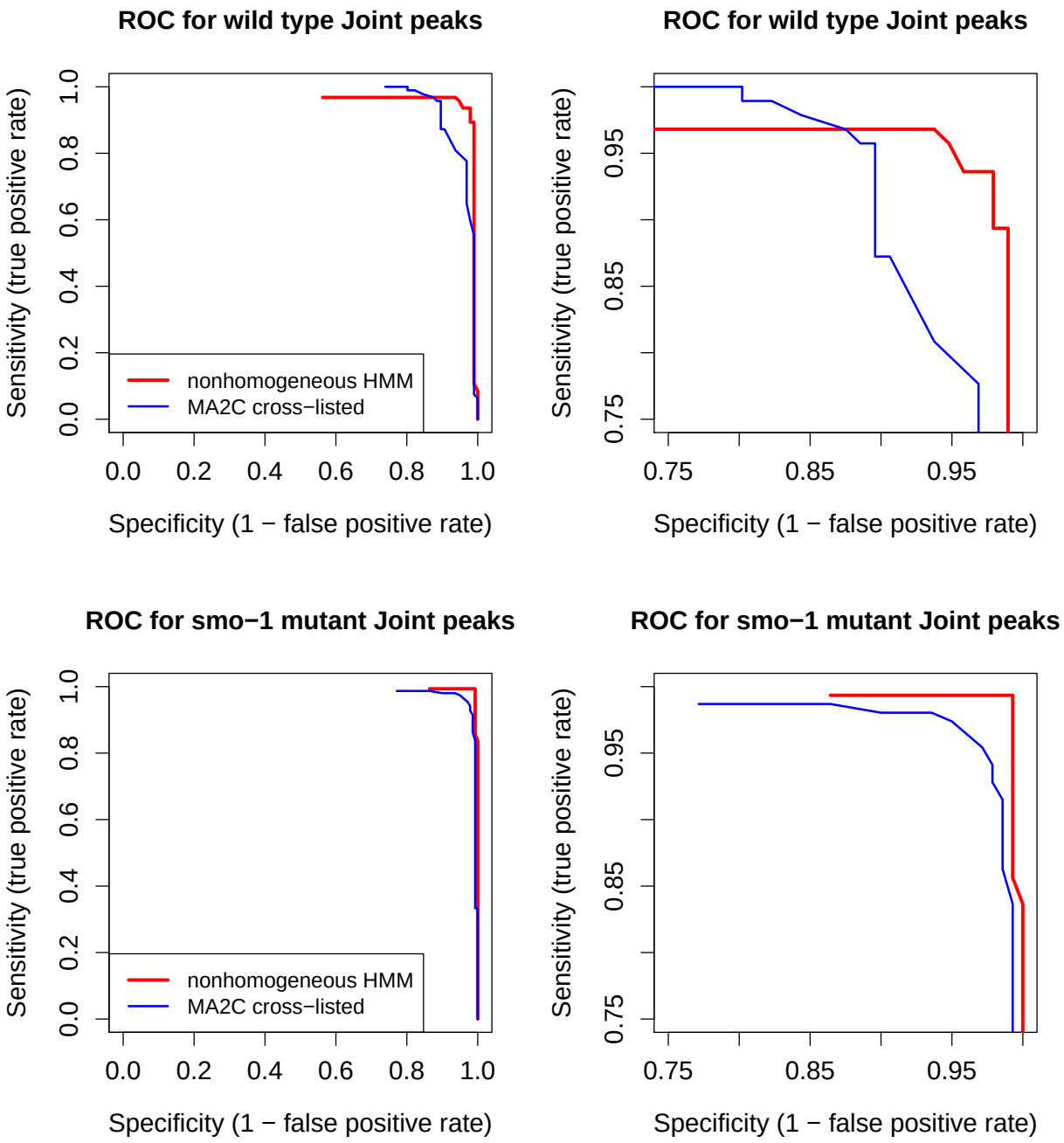

Figure 4.20: ROC curves comparing the nonhomogeneous HMM with the MA2C cross-listing method, when used to identify the shared binding sites of Dpy-27 and Sdc-3. The nonhomogeneous HMM is shown in red; MA2C is shown in blue. The right-hand side panels are zoomed-in versions of the left-hand side panels.

Our data set consists of experiments performed on tiling arrays with three different probe designs. Let N_1 denote the number of probes in Design 1; Let N_2 denote the number of probes in Design 2; Let N_3 denote the number of probes in Design 3. Let $N = max(N_1, N_2, N_3)$. The target distribution contains N probes. After sorting the probe level intensities of each array in ascending order, each probe i is assigned an integer x_i that indicates its relative position on the target distribution.

$$x_i = \text{floor}[rank(i) \times \frac{N}{N_1}] \qquad \text{if the array belongs to Design 1}$$

$$x_i = \text{floor}[rank(i) \times \frac{N}{N_2}] \qquad \text{if the array belongs to Design 2}$$

$$x_i = \text{floor}[rank(i) \times \frac{N}{N_3}] \qquad \text{if the array belongs to Design 3}$$

The intensity of the j-th ranked probe in the target distribution is computed as the average of all the probes with $x_i = j$ among all the arrays. For a small fraction of the probes in the target distribution, the number of experiments used in computing the target intensity is less than the total number of experiments in the set. Nevertheless, we decided to construct the target distribution using the maximum value of N to maintain the highest possible resolution.

We quantile normalized the single-channel intensities of each array in the complete data set. Then, two channels were combined as log base-2 ratios of IP versus input. To verify that our modified implementation works properly, we looked at the 3 replicates of wild type Dpy-27 before and after quantile normalization. Figure 4.21 shows QQ-plots of the log ratios for the replicate experiments performed with Design 1 and Design 2, before and after quantile normalization. Figure 4.22 shows QQ-plots of the log ratios for the replicate experiments performed with Design 2 and Design 3, before and after quantile normalization. Figure 4.23 shows QQ-plots of the log ratios for the replicate experiments performed with Design 3 and Design 1, before and after quantile normalization. The QQ-plot of Design 1 versus Design 2 is substantially closer to the 45° line after quantile normalization. This difference is not as obvious in the QQ-plots of Design 2 versus Design 3 and Design 3 versus Design 1.

We would like to look at the pair-wise differences between replicate data before and after quantile normalization. In order to achieve a one-to-one correspondence between the probes of different designs, we created a pseudo-design that contains probes of length 200 bp. Several probes of length 40 to 50 bp in the original design were binned together according to the pseudo-design. The intensity of a binned probe on the pseudo-design was computed as a weighted average of the probes in the original design, with weights proportional to the number of base overlaps. Mapping to the pseudo-design was performed at the level of the single-channel intensities. The two channels were then combined as log base-2 ratios between the binned IP intensities and the binned input intensities. For each binned probe, we computed pair-wise differences between the log ratios of replicate experiments. When there is no experimental bias in the data, the expected value of the differences between replicates is zero. Figure 4.24 shows boxplots of these differences before and after quantile normalization. Whereas the median of each box deviates from zero before normalization; the

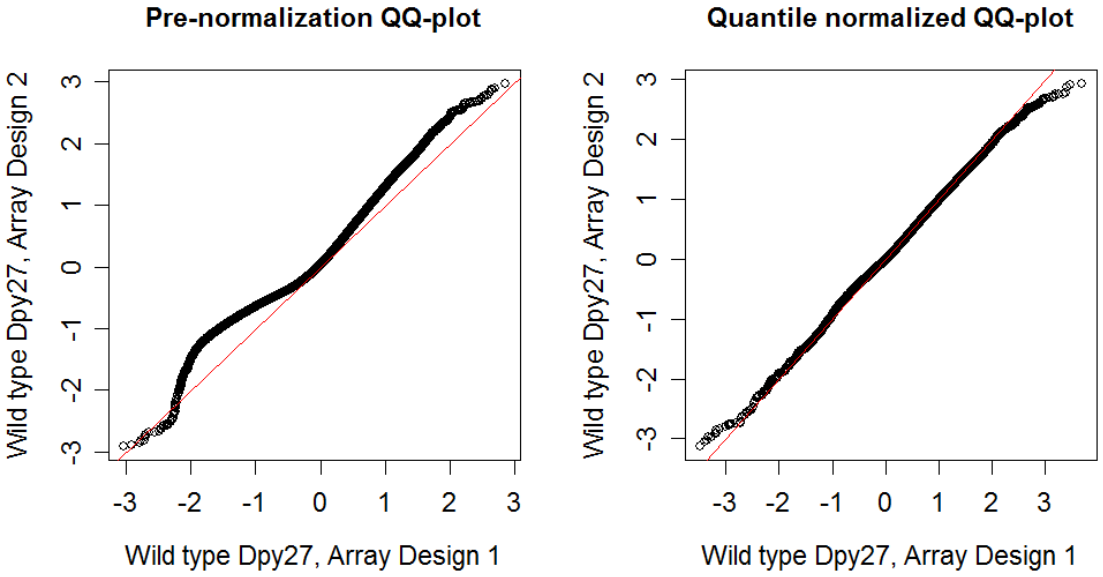

Figure 4.21: QQ-plots of wild type Dpy-27 replicate data on Designs 1 and 2 before and after quantile normalization. The 45° line is drawn in red.

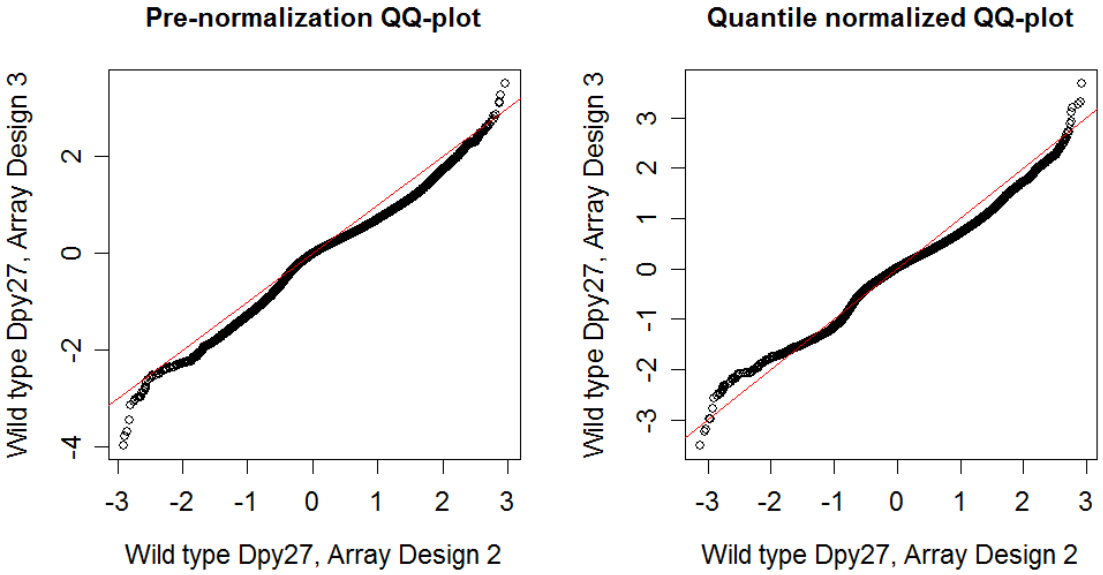

Figure 4.22: QQ-plots of wild type Dpy-27 replicate data on Designs 2 and 3 before and after quantile normalization. The 45° line is drawn in red.

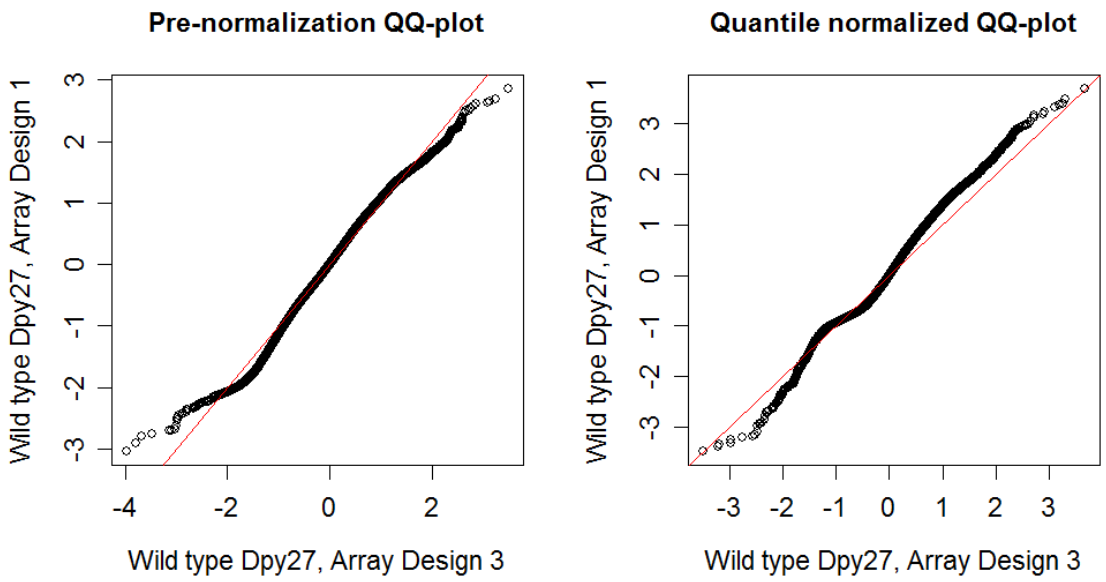

Figure 4.23: QQ-plots of wild type Dpy-27 replicate data on Designs 3 and 1 before and after quantile normalization. The 45° line is drawn in red.

median of each box hits right at zero after normalization. Clearly, quantile normalization was useful for removing the experimental biases in this data set.

4.6.2 Estimation of emission parameters

When fitting the emission parameters on the genome wide scale, we noticed that the Baum-Welch updates tend to push the mean of the bound state closer to the unbound state after every iteration. This is probably related to the fact that a large proportion of the positions with non-zero posterior probabilities for the bound state are actually background noise. Although the posterior probability of each position is small, the sum of all of them is substantial. Since the true binding sites make up for only a small proportion of the positions with non-zero posterior probabilities, the weighting in the emission parameter estimates is likely to be unfavorable for the binding sites. In other words, the non-binding sites have a diluting effect on the emission parameter estimates for the binding sites. In the analysis of wild type data, this issue was resolved by restricting the emission parameter updates to be solely based on Chromosome X. Since Chromosome X has a much higher density of true binding sites than the rest of the genome, the diluting effect of the non-binding sites becomes negligible when the restriction is in place.

When analyzing the *smo-1* mutant data, we first tried estimating the emission parameters from the whole genome, and the same problem described above occurred. When we restricted emission parameter estimation to Chromosome X, the final estimates of the mean parameters were generally smaller for the mutant data than the wild type data. A closer inspection of

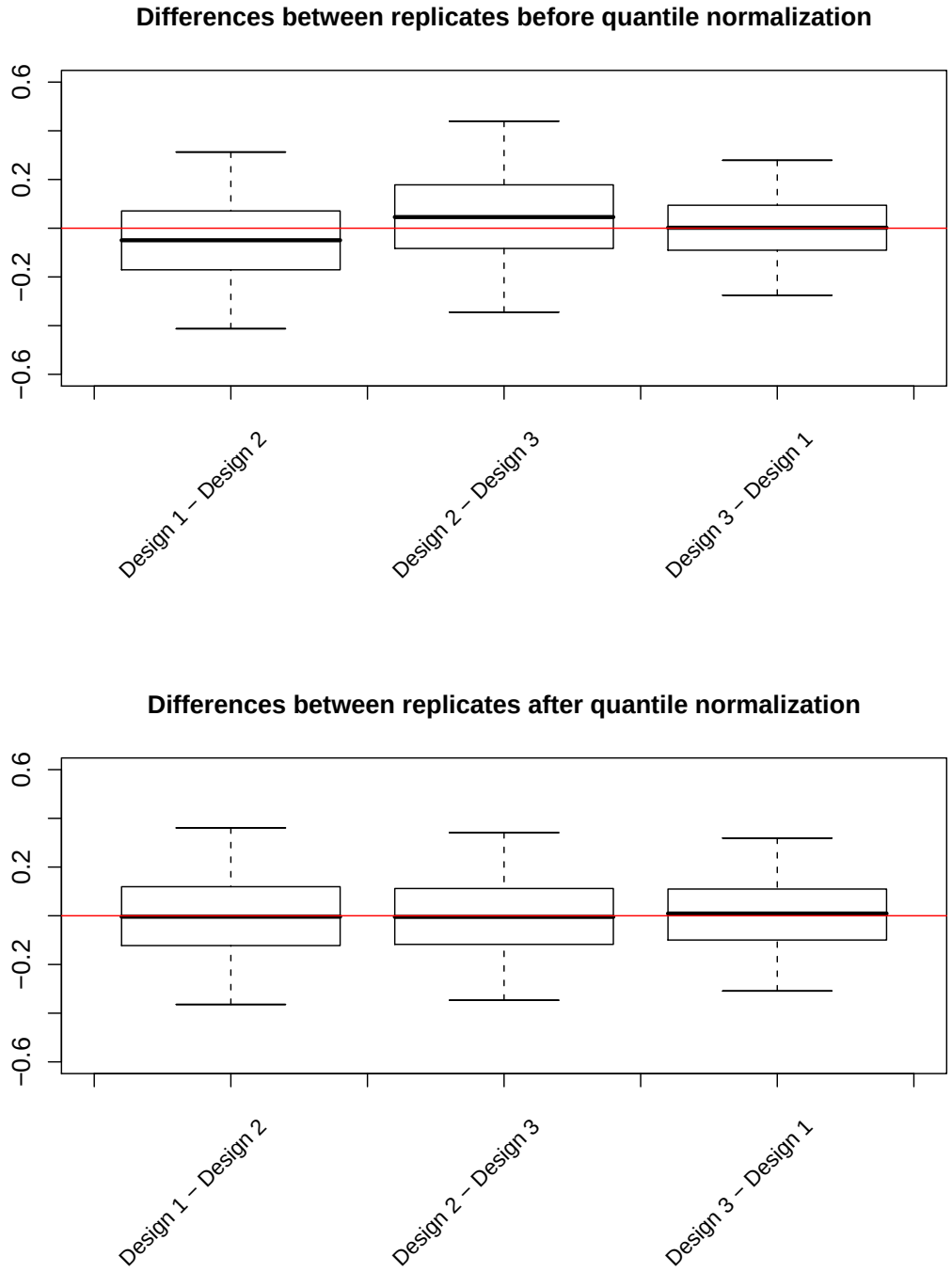

Figure 4.24: Boxplots of pair-wise differences between replicates of wild type Dpy-27, before and after quantile normalization. The 25th and 75th quantiles are represented by the boundaries of each box. The median is drawn as a thick black horizontal line. The whiskers extend to the most extreme data points within one interquartile range from the box.

the mutant data on a genome browser revealed that the autosomal peaks tend to be stronger than the X-Chromosomal peaks. So, using the X-chromosomal observations to estimate the emission parameters for the whole genome is not such a good idea with the mutant data. When a lower value is used as the emission mean of the bound state, more peaks will be detected due to the overall inflating effect on the posterior probability of the bound state. Indeed, we saw many noisy regions being called peaks in the mutant data, when the emission parameters were estimated from Chromosome X.

Another issue that needs to be considered is the comparison between wild type and mutant peaks. Ideally, the emission parameters of the mutant data should have the same values as those of the wild type data, because the definition of a peak should be held constant as we move across the two conditions. The consistency in what defines a peak would then form the basis for comparing the localization of peaks under the wild type and mutant conditions. If the emission parameters were dramatically different, then it would be very difficult to compare the mutant peaks with the wild type peaks. So we would like to check how similar or different the emission distributions are under the wild type and mutant condition. In order to do so, we need to know where the mutant peaks are located. The next thing that we tried was fixing the emission parameters of the mutant data at the same values as the wild type emission parameters. This approach gave peak calls that appeared reasonable according to visual inspection. For both the wild type data and the mutant data, we defined some intervals using the criteria listed below.

1. A probe is considered as a candidate probe if its posterior probability of being in the binding state is at least 0.8.

2. The maximum gap allowed between candidate probes within any interval is 300 bp.

3. The minimum length required for any interval is 500 bp.

Probes that fall within these intervals are likely to be in the bound state. Probes that fall outside of these intervals are likely to be in the unbound state. The distribution of the observations that fall within these intervals can be used as a proxy for the emission distribution of the bound state. Similarly, the distribution of the observations that fall outside of these intervals can be used as a proxy for the emission distribution of the unbound state. Figure 4.25 compares the distributions of wild type and *smo-1* mutant Dpy-27 observations collected using tiling array Design 1. Figure 4.6.2 and Figure 4.27 compare the distributions of wild type and *smo-1* mutant Dpy-27 observations collected using Design 2 and Design 3, respectively. The top row of each figure shows histograms of the observations that are likely to be in the unbound state, labeled as "probes in non-peak regions." The bottom row of each figure shows histograms of the observations that are likely to be in the bound state, labeled as "probes in peak regions." The left column of each figure shows the histograms of the wild type data. The right column of each figure shows the histograms of the *smo-1* mutant data. In general, the wild type histogram has a wider spread than the mutant histogram, indicating that the wild type distribution has a larger variance. For observations in the non-peak regions, the mode of each histogram is near zero. For the observations in the peak regions of Design 1 and Design 2, the modes of the wild type histograms are right-shifted by 0.1 in

comparison to the modes of the mutant histograms. For the observations in the peak regions of Design 3, the modes of both the wild type and mutant histograms are the same.

To compare the wild type and mutant distributions more closely, we made QQ-plots using the observations in the 5th to 95th percentiles. The top and bottom 5% of the observations in each data set were excluded to avoid distractions by outliers. Figure 4.28 shows the QQ-plots of the log ratios for Design 1. Figure 4.29 shows the QQ-plots of the log ratios for Design 2. Figure 4.30 shows the QQ-plots of the log ratios for Design 3. The 45° line is drawn in red on each QQ-plot. For observations in the unbound state, the wild type and mutant conditions have very similar distributions. For observations in the bound state, the wild type and mutant distributions deviate from each other slightly. These plots suggest that the State 1 emission distributions are slightly different for the wild type and mutant conditions. Nevertheless, the approach of fixing the emission parameters at the wild type values is still useful for analyzing the mutant data, because it allows the wild type and mutant conditions to be compared on the same scale.

4.6.3 Analysis of IgG control data

Chromatin immunoprecipitation experiments involve the binding of antibodies to the protein-DNA complex being analyzed. However, non-specific interactions are also likely to occur between the antibodies and the other molecules in the cell extract. Certain regions in the genome are more prone to non-specific binding than others. Thus a standard protocol in the field is to perform some control experiments using non-specific IgG, also known as the mock IP's. The goal of the IgG control experiments is to identify regions of the genome that are prone to non-specific binding. If a peak overlaps with any of these regions, then it should be flagged as a false positive.

The Meyer group performed three IgG control experiments using tiling arrays of Design 3: two under the wild type condition and one under the *smo-1* mutant condition. Visual inspection of the data revealed two types of non-specific binding by IgG. The first type of non-specific intervals consist of long runs of probes with relatively low signals. The second type of non-specific intervals consist of short runs of probes with very high signals. Thus we designed a three-state nonhomogeneous HMM to accommodate these different types of non-specific binding in the IgG data. The hidden states of this model are: 0) non-binding, 1) weak binding with long duration, 2) strong binding with short duration. Because the binding regions make up for a very small fraction of the genome, it is not practical to obtain initial estimates of the parameters through an automated procedure. So we manually selected regions that are characteristic of the two binding states.

We estimated the emission parameters for Design 3 using the observations in the characteristic regions. Just like in the analysis of the mutant data, the IgG emission parameters were also fixed at initialization and never updated. At each iteration of the modified Baum-Welch algorithm, only the Markov chain parameters were updated. Table 4.8 shows the estimates of the emission parameters for tiling array Design 3.

Based on the average length of the characteristic binding regions of each type, we obtained initial estimates for the diagonal entries in the one-step transition matrix. The transition

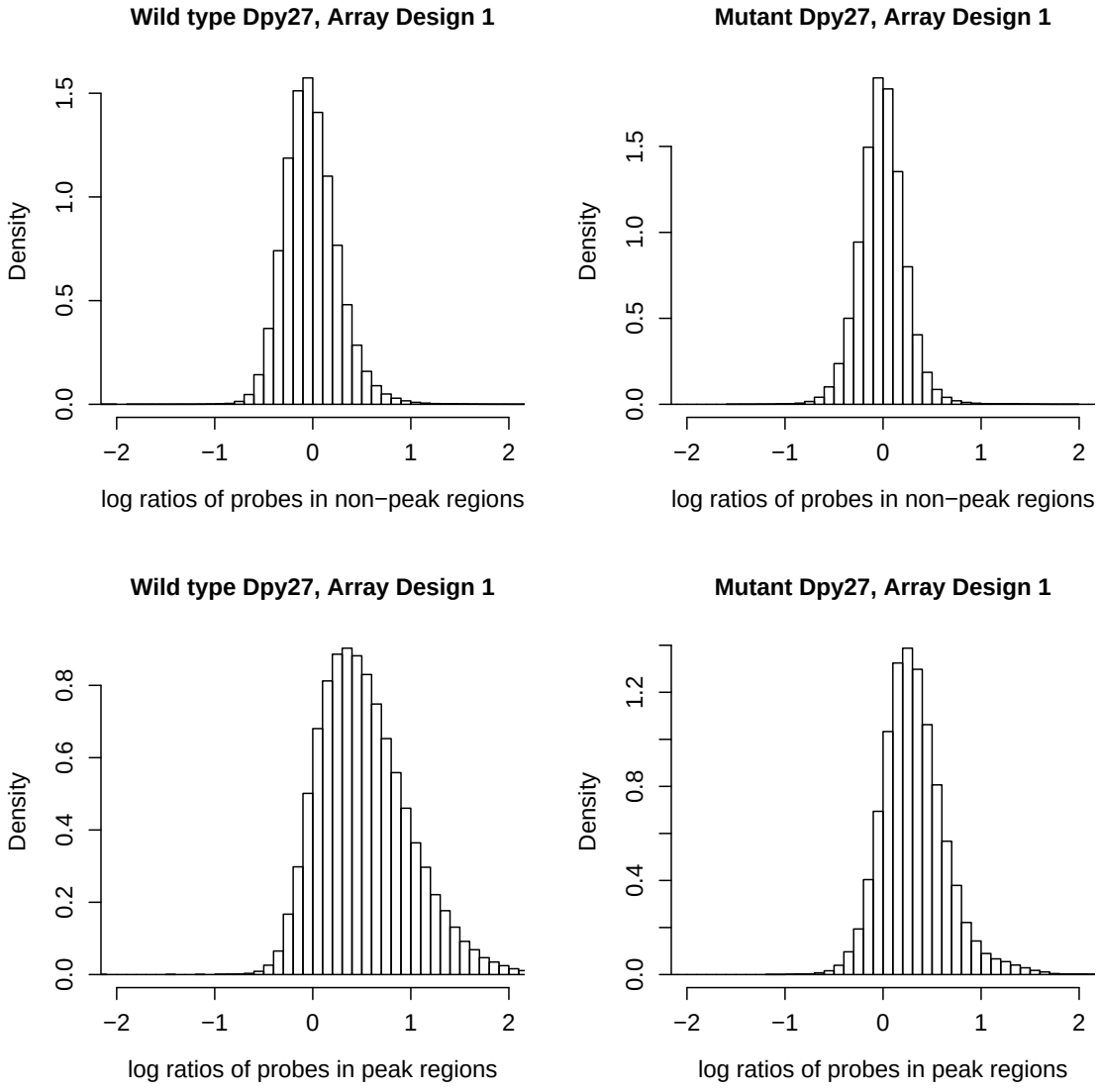

Figure 4.25: Histograms of Dpy-27 observations on tiling arrays of Design 1

State	Mean (μ)	SD σ
0	0.000	0.191
1	0.227	0.164
2	1.777	0.562

Table 4.8: Emission parameters for IgG on tiling arrays of Design 3

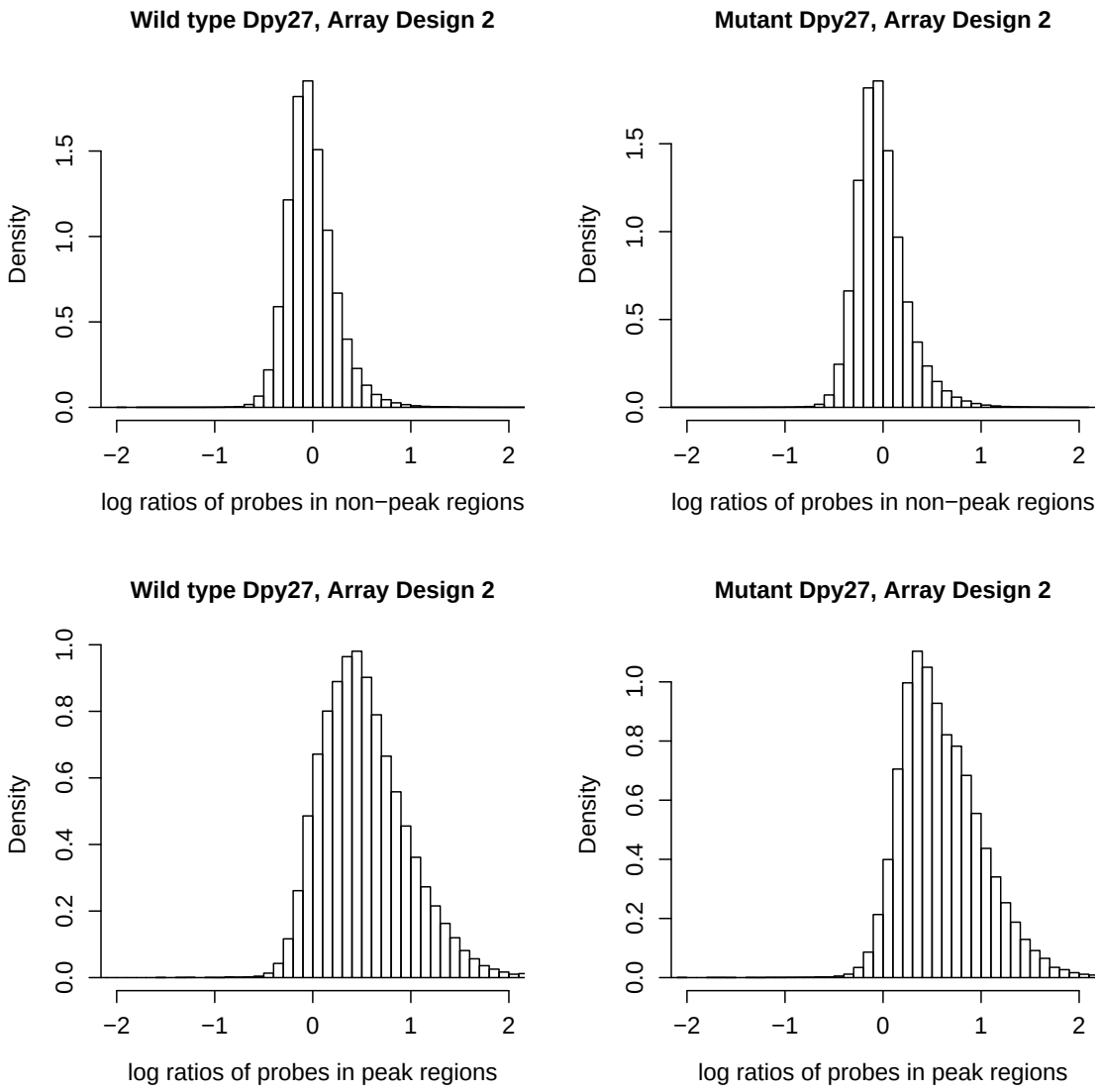

Figure 4.26: Histograms of Dpy-27 observations on tiling arrays of Design 2

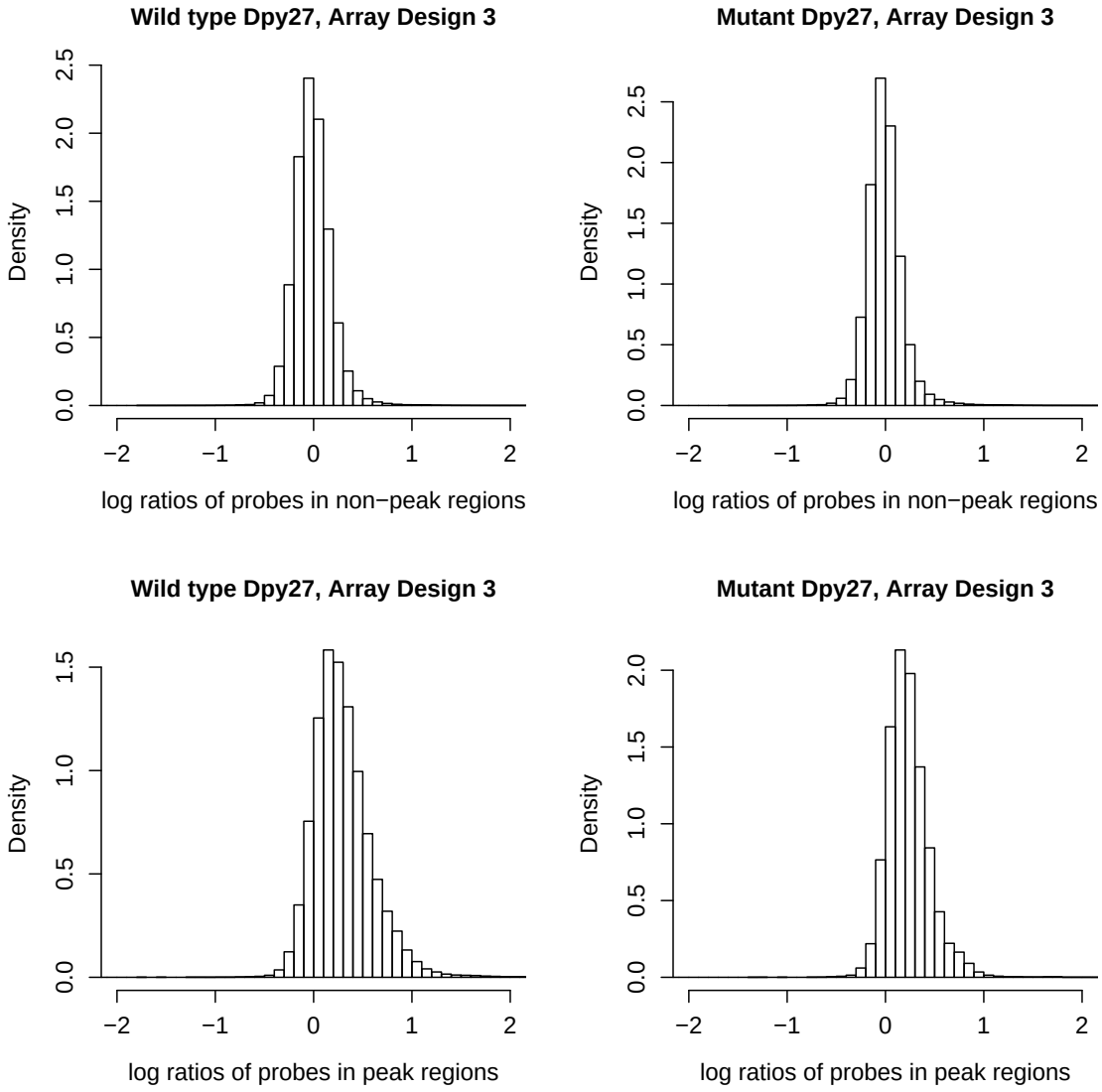

Figure 4.27: Histograms of Dpy-27 observations on tiling arrays of Design 3

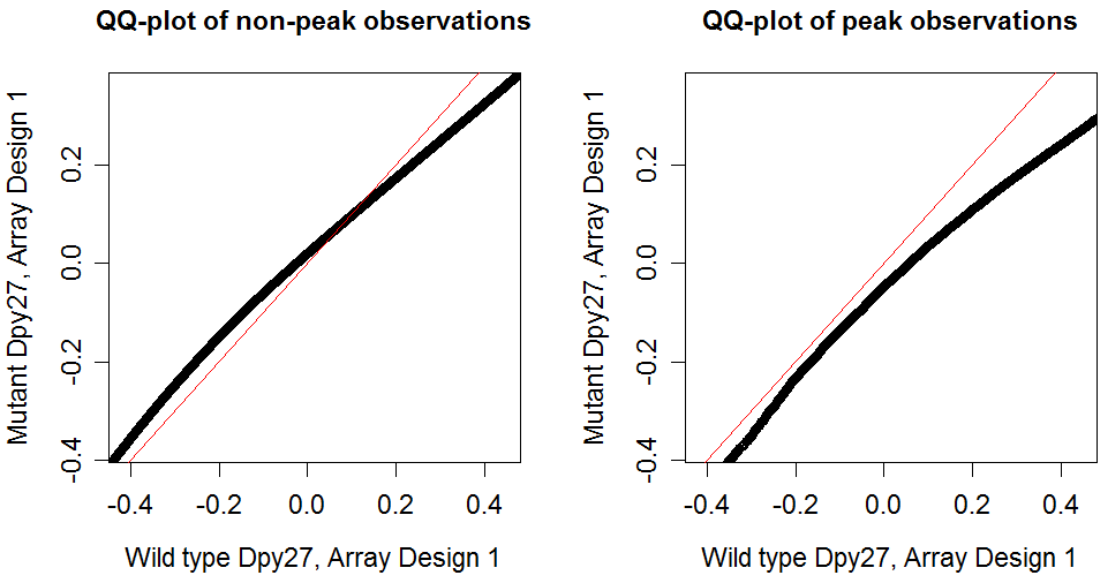

Figure 4.28: QQ-plots of Dpy-27 observations on tiling arrays of Design 1

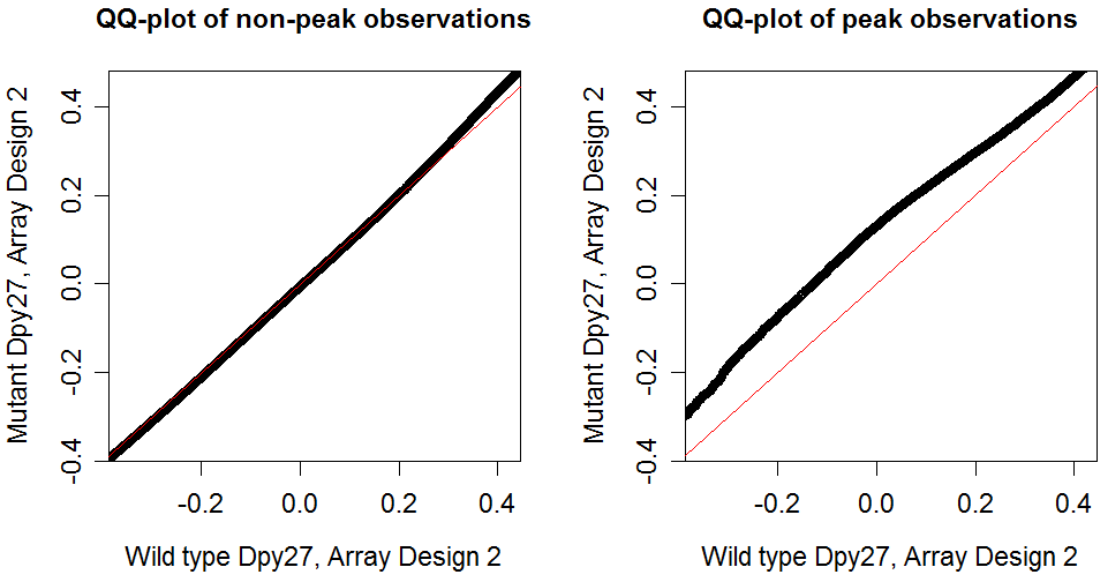

Figure 4.29: QQ-plots of Dpy-27 observations on tiling arrays of Design 2

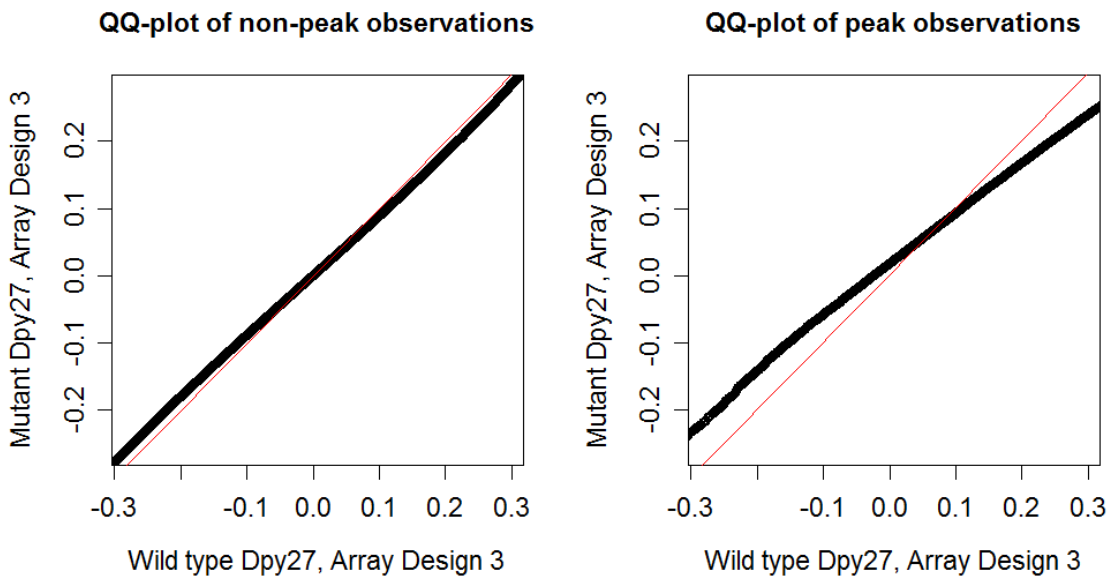

Figure 4.30: QQ-plots of Dpy-27 observations on tiling arrays of Design 3

probabilities from State 0 to State 1 and from State 0 to State 2 were estimated based on rough estimates for the relative abundances of State 1 and State 2. Transition probabilities between State 1 and State 2 were initialized to zero. We then used the transition matrix to compute the stationary distribution. Although the estimates obtained this way are likely to be quite off, they are nevertheless useful for initializing the Baum-Welch algorithm. Shown below are the initial values of the stationary distribution and the transition matrix.

- Initial Estimate of the Stationary Distribution for IgG

$$\pi = (0.978, 0.0212, 0.0004)$$

- Initial Estimate of the Transition Matrix for IgG

$$\mathbf{A} = \begin{pmatrix} 0.9999851 & 0.0000142 & 0.000000642 \\ 0.000656 & 0.999344 & 0.00 \\ 0.00155 & 0.00 & 0.9984466 \end{pmatrix}$$

In the process of fitting the IgG data, we noticed many single probe spikes of artificial signals that can lead to noisy peak calls. This type of spikes also existed in the IP data of bona-fide DNA binding proteins. Because the real peaks were far more abundant than the artificial spikes in the protein IP data, these spikes did not cause any problems there. However, the artificial spikes are problematic in the IgG data, because there are far fewer peaks here. Fortunately, the noise can be reduced substantially by applying the filtering criterion below.

1. For every probe, compute the absolute differences between the log ratio of the given probe and log ratios of its immediate neighbors.

2. If either of the two absolute differences exceeds 4 times the current estimate of the SD for the unbound state, then the probe is filtered out.

At each iteration of the Baum-Welch algorithm, the filtering criterion is applied before the forward-backward variables are computed. Because the spikes generally occur in the background of the non-binding probes, the threshold is chosen based on the standard deviation of the unbound state. It is conceivable that the spikes may also occur among the binding probes, but they would be essentially indistinguishable from the real signals. Thus the filtering criterion aims at removing the single probe spikes in the unbound state.

After running the modified Baum-Welch algorithm for 10 iterations, we observed convergence of the Markov chain parameter estimates, which are given below.

- Final Estimate of the Stationary Distribution for IgG

$$\pi = (0.943, 0.057, 0.000)$$

- Final Estimate of the Transition Matrix for IgG

$$\mathbf{A} = \begin{pmatrix} 0.9998 & 0.0002 & 0.0000 \\ 0.0034 & 0.9966 & 0.0000 \\ 0.0003 & 0.0016 & 0.9981 \end{pmatrix}$$

Figures 4.31 and 4.32 show a couple examples of the intervals in State 1 (weak binding with long duration). Figures 4.33 and 4.34 show a couple examples of the intervals in State 2 (strong binding with short duration). In each figure, the log ratios of three IgG control experiments are displayed in cyan. The posterior probabilities of State 1 are shown in orange. The posterior probabilities of State 2 are shown in pink. To illustrate the effects of filtering out single probe spikes, let us take a look at Figure 4.33. There is a spike around position 5,921,000 with a value that far exceeds any of its neighboring probes. If left unfiltered, this spike would lead to a high posterior probability for one of the binding states, but its duration would be the length of a single probe. Because we incorporated the filtering criterion in the model fitting procedure, the posterior probabilities of the two binding states are nearly zero at this position.

Finally, the intervals of State 1 and State 2 were defined by joining neighboring probes according to the following rules.

1. Consider probes with posterior probabilities of at least 0.8, and call them candidate probes for that particular binding state.

2. The maximum gap allowed between candidate probes within any interval is 300 bp.

3. The minimum length required for any interval is 500 bp.

If a peak identified from a set of real protein IP experiments contains any overlap with the IgG intervals, then this peak should be flagged as a possible false positive due to non-specific binding.

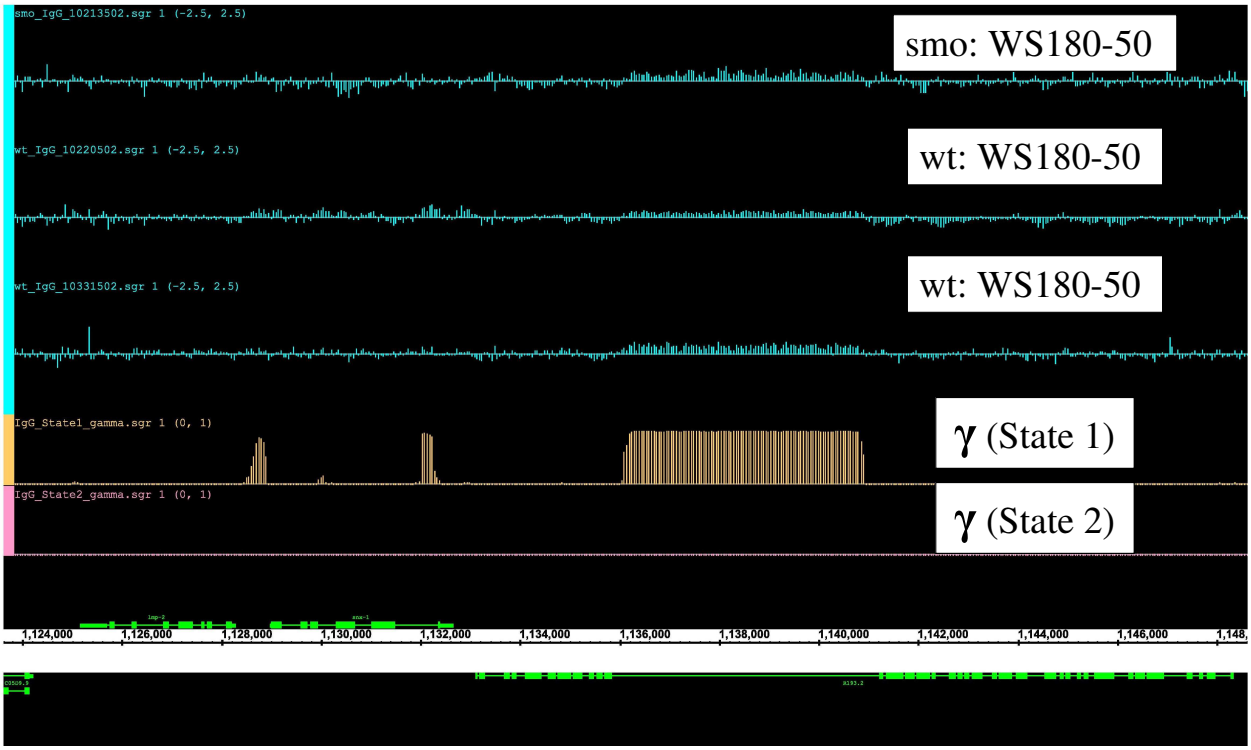

Figure 4.31: An IgG interval of State 1: The log ratios of three IgG control experiments are displayed in cyan. The posterior probabilities of State 1 are shown in orange. The posterior probabilities of State 2 are shown in pink.

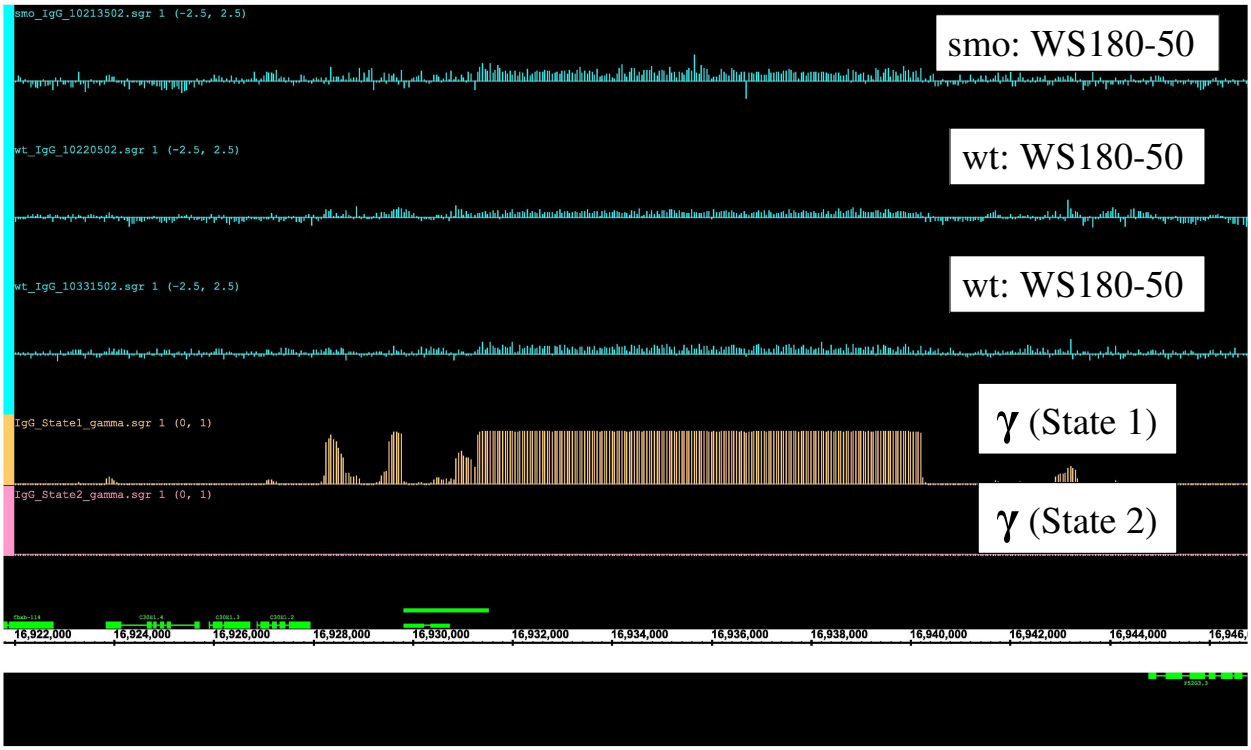

Figure 4.32: An IgG interval of State 1: The log ratios of three IgG control experiments are displayed in cyan. The posterior probabilities of State 1 are shown in orange. The posterior probabilities of State 2 are shown in pink.

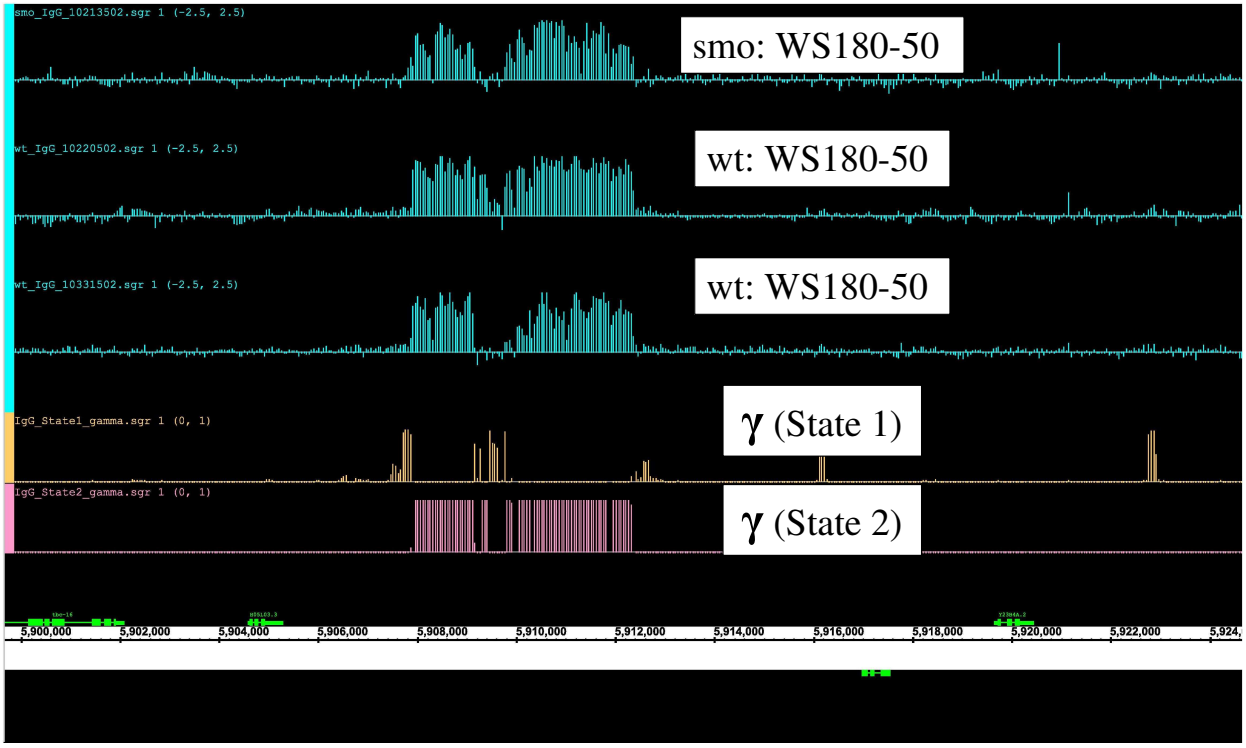

Figure 4.33: An IgG interval of State 2: The log ratios of three IgG control experiments are displayed in cyan. The posterior probabilities of State 1 are shown in orange. The posterior probabilities of State 2 are shown in pink.

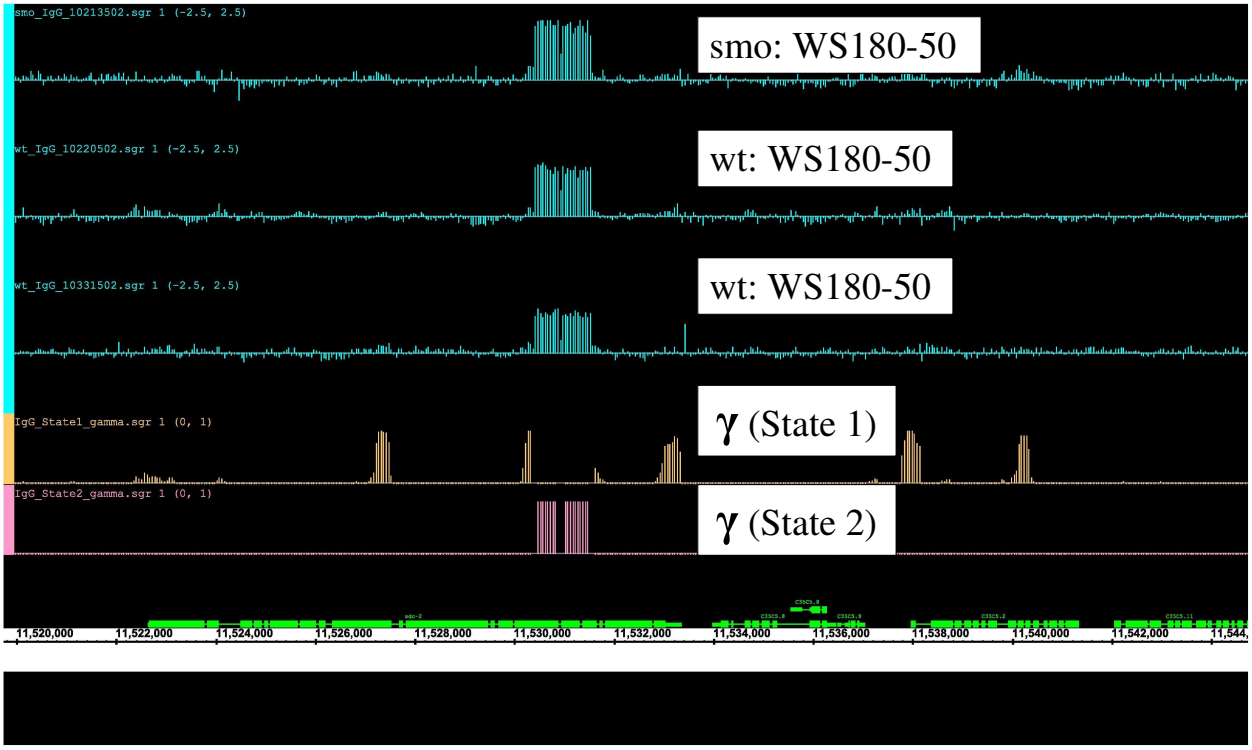

Figure 4.34: An IgG interval of State 2: The log ratios of three IgG control experiments are displayed in cyan. The posterior probabilities of State 1 are shown in orange. The posterior probabilities of State 2 are shown in pink.

Design	unbound μ	unbound σ	bound μ	bound σ
1	0.125	0.231	0.680	0.380
2	0.171	0.168	0.690	0.317
3	0.059	0.146	0.379	0.241

Table 4.9: Emission parameters for Dpy-27

Design	unbound μ	unbound σ	bound μ	bound σ
1	N/A	N/A	N/A	N/A
2	-0.034	0.133	0.361	0.299
3	0.025	0.193	0.524	0.388

Table 4.10: Emission parameters for Sdc-3

4.6.4 Summary of results

We fitted a two-state nonhomogeneous HMM for each protein under each condition, according to the method described in Section 3.2. Convergence of the parameter estimates occurred within 10 iterations of the modified Baum-Welch updates. Table 4.9 shows the final estimates of the emission parameters obtained from fitting the wild type Dpy-27 data. Table 4.10 shows the final estimates of the emission parameters obtained from fitting the wild type Sdc-3 data. The same emission parameters were used to analyze the *smo-1* mutant data. The final estimates of the Markov chain parameters for the single-protein models are given below.

- Markov chain parameters for wild type Dpy-27

$$\pi = (0.952, 0.048);\ \mathbf{A} = \begin{pmatrix} 0.9999 & 0.0001 \\ 0.0019 & 0.9981 \end{pmatrix}$$

- Markov chain parameters for wild type Sdc-3

$$\pi = (0.922, 0.078);\ \mathbf{A} = \begin{pmatrix} 0.9998 & 0.0002 \\ 0.0023 & 0.9977 \end{pmatrix}$$

- Markov chain parameters for *smo-1* mutant Dpy-27

$$\pi = (0.953, 0.047);\ \mathbf{A} = \begin{pmatrix} 0.9999 & 0.0001 \\ 0.0019 & 0.9981 \end{pmatrix}$$

- Markov chain parameters for *smo-1* mutant Sdc-3

$$\pi = (0.800, 0.200);\ \mathbf{A} = \begin{pmatrix} 0.9997 & 0.0003 \\ 0.0014 & 0.9986 \end{pmatrix}$$

We postprocessed the probe level summaries into peaks using the method described in Section 3.4. The peak score cutoffs were selected to control for the false positive rate of filtering below 15%, according to the training set of curated standards described in Section 3.4. Under the wild type condition, the cutoffs were 1.42 and 1.34 for Dpy-27 and Sdc-3 peaks, respectively. Under the mutant condition, the cutoffs were 1.28 and 1.56 for Dpy-27 and Sdc-3 peaks, respectively. The results were 930 wild type Dpy-27 peaks, 1349 wild type Sdc-3 peaks, 784 *smo-1* mutant Dpy-27 peaks, 2163 *smo-1* mutant Sdc-3 peaks.

We then fitted a four-state nonhomogeneous HMM to identify the jointly bound sites of Dpy-27 and Sdc-3 using the method described in Section 4.3. The emission parameters shown in Table 4.9 and 4.10 were used in the joint analyses. The final estimates of the Markov chain parameters are given below.

- Stationary distribution for wild type data

$$\pi = (0.918, 0.036, 0.009, 0.037)$$

- Transition matrix for wild type data

$$\mathbf{A} = \begin{pmatrix} 0.9998 & 0.0001 & 0.0000 & 0.0000 \\ 0.0039 & 0.9951 & 0.0000 & 0.0010 \\ 0.0021 & 0.0000 & 0.9966 & 0.0013 \\ 0.0006 & 0.0010 & 0.0003 & 0.9980 \end{pmatrix}$$

- Stationary distribution for *smo-1* mutant data

$$\pi = (0.780, 0.164, 0.006, 0.050)$$

- Transition matrix for *smo-1* mutant data

$$\mathbf{A} = \begin{pmatrix} 0.9996 & 0.0004 & 0.0000 & 0.0000 \\ 0.0018 & 0.9976 & 0.0000 & 0.0005 \\ 0.0009 & 0.0000 & 0.9988 & 0.0002 \\ 0.0001 & 0.0019 & 0.0000 & 0.9980 \end{pmatrix}$$

Notice that the stationary distribution of State 01 is far larger than State 10, suggesting that Sdc-3 binding sites are more abundant than Dpy-27 binding sites. This is consistent with the hypothesis that Sdc-3 recruits the dosage compensation complex to discrete X-recognition elements on Chromosome X, thus Dpy-27 binding depends on Sdc-3 [17]. Besides functioning as part of the DCC, Sdc-3 also plays other roles in sex-determination. The abundance of binding sites that are unique to Sdc-3 may also be explained by these roles.

The probe level summaries obtained from the two-protein nonhomogeneous HMM were also postprocessed using the method described in Section 3.4. We focused on State 11 to identify the joint binding sites of Dpy-27 and Sdc-3. The peak score cutoffs were selected to control for the false positive rate of filtering below 15%, according to the training set of curated standards described in Section 3.4. The cutoffs for the Dpy-27 and Sdc-3 jointly

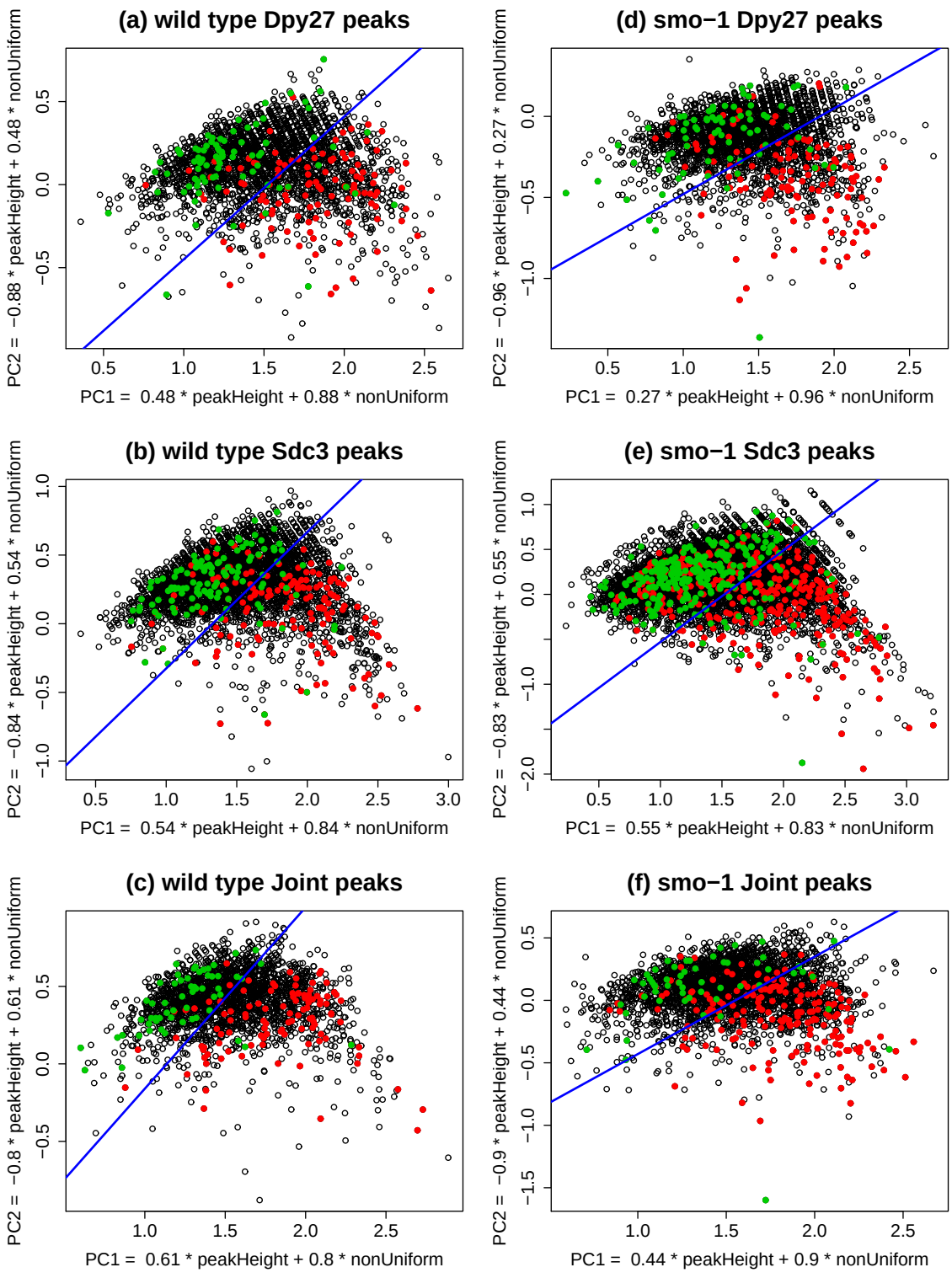

Figure 4.35: Peak filtering results (training set): Scatter plots of the peaks along the principal components. Peaks that overlap with the curated positive (negative) regions in the training set are colored in red (green). The peak score cutoffs are represented by the solid blue lines.

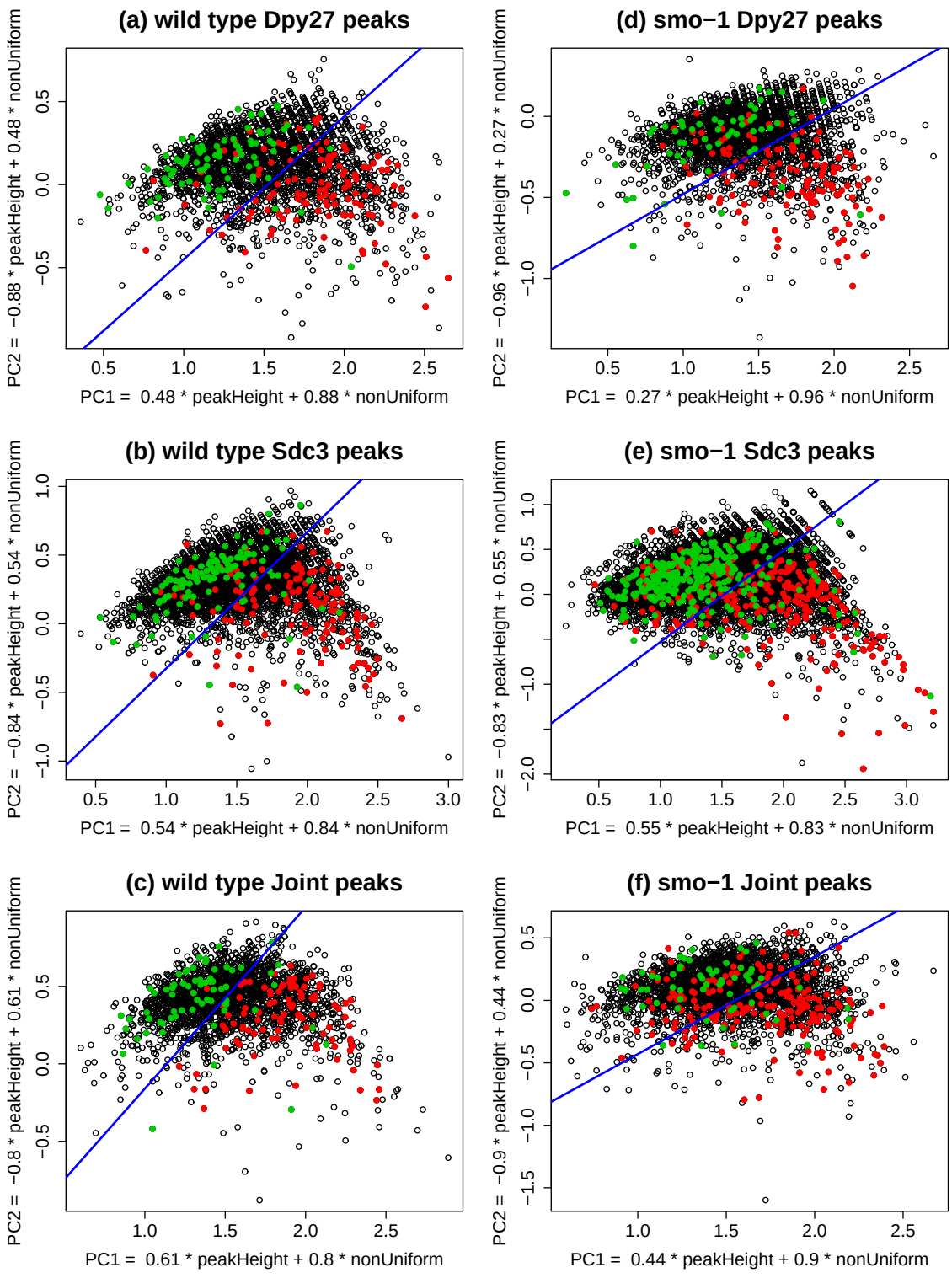

Figure 4.36: Peak filtering results (testing set): Scatter plots of the peaks along the principal components. Peaks that overlap with the curated positive (negative) regions in the testing set are colored in red (green). The peak score cutoffs are represented by the solid blue lines.

Condition	Protein	False Positive Rate	False Negative Rate
wild type	Dpy-27	1%	12%
wild type	Sdc-3	2%	6%
wild type	Joint	4%	7%
smo-1 mutant	Dpy-27	1%	7%
smo-1 mutant	Sdc-3	2%	5%
smo-1 mutant	Joint	1%	18%

Table 4.11: Overall error rates of the final peak calls

bound peaks were 1.28 under the wild type condition and 1.36 under the *smo-1* mutant condition. The results were 1060 joint peaks under the wild type condition and 1093 joint peaks under the mutant condition.

Figure 4.35 shows the results of filtering by peak scores on the training set of curated standards. Figure 4.36 shows the results of filtering by peak scores on the testing set of curated standards. Each panel is a scatter plot of the intervals along the principal components of *peak height* and *non-uniformity*. The training set of curated standards are shown in Figure 4.35. The testing set of curated standards are shown in Figure 4.36. Peaks that overlap with the curated positive regions are colored in red. Peaks that overlap with the curated negative regions are colored in green. The cutoff for the final peak scores is represented by the solid blue line in each plot. Peaks that lie above the blue line were filtered out. The cutoffs were chosen to control for the training set false positive rates of peak filtering within 15%. The results of peak filtering look fairly similar between the training set and the testing set, according to these scatter plots.

To estimate the overall error rates of our final peak calls, we used the ROC results of the entire method on the testing set of curated standards. The false positive and false negative rates are summarized for each set of peak calls in Table 4.11. Because these values were estimated using a relatively small set of curated positive and negative regions, they are only meant to be suggestive.

Under the wild type condition, the majority of the binding peaks occurred on Chromosome X. Under the *smo-1* mutant condition, a large fraction of those X-chromosomal peaks moved off to the autosomes. Among the 930 wild type Dpy-27 peaks, 613 were on Chromosome X. In contrast, among the 784 mutant Dpy-27 peaks, only 198 were on Chromosome X. Among the 1349 wild type Sdc-3 peaks, 445 were on Chromosome X. In contrast, among the 2163 mutant Sdc-3 peaks, only 327 were on Chromosome X. Among the 1060 wild type joint peaks, 593 were on Chromosome X. In contrast, among the 1093 mutant joint peaks, only 229 were on Chromosome X. Clearly, sumoylation plays an important role in the localization of the dosage compensation complex to Chromosome X.

Chapter 5

Conclusions and discussion

5.1 FDR procedure for microarray time course data

DNA microarray time course experiments are often used to monitor gene expression changes during a biological process. Unlike periodic time courses with cyclic patterns, developmental time courses may have arbitrary temporal patterns. Developmental microarray time course experiments are often performed with few time points and few replications. When the sampling scheme is longitudinal, each replicate consists of a series of samples collected from the same experimental unit in a temporal order. Thus there may be some autocorrelations in the time course gene expression data. However, traditional methods of time series analysis are not applicable because of the small number of time points. The multivariate empirical Bayes model proposed by Tai and Speed [67, 68] provides a solution to the analysis of longitudinally replicated microarray time course data. The time series of each gene is modeled after a multivariate Gaussian vector, which captures the autocorrelations in time. The empirical Bayes approach stabilizes the estimation of the covariance matrix in small samples. The *MB*-statistic derived from this model was shown quite useful for ranking the genes according to changes in their temporal expression profiles.

In Chapter 2 of this thesis, we developed an empirical Bayes FDR-controlling procedure for multiple hypothesis testing using the *MB*-statistic. Under the null hypothesis of no "differential expression", the distribution of the test statistic is related to the F-distribution by a constant factor. The frequentist FDR procedure applies the Benjamini-Hochberg [3] step-up adjustment to the nominal p-values obtained under the F-distribution. Because specification of the null distribution involves the prior degrees of freedom in the hierarchical Bayesian model, it is sensitive to the variability in hyperparameter estimation. We proposed to estimate the null distribution using the parametric bootstrap. This involves first estimating the Gaussian parameters using the observed data, and then generating "resampled data" from the Gaussian model. Critical values of the test statistic are determined according to the empirical Bayes FDR procedure due to Efron et al. [26]. We performed some simulations to compare the frequentist and the empirical Bayes FDR procedures under various circumstances. When the sample size is small, the bootstrap null distribution is slightly anti-conservative. On the other hand, when the prior degrees of freedom is small, the F-distribution is far off from

the true null distribution because of hyperparameter estimation errors. We found that that the empirical Bayes FDR procedure is more robust than its frequentist counterpart to the variability in hyperparameter estimation.

When applying the FDR procedures to a fungal infection time course in *A. thaliana*, we encountered a challenge of the sample size being smaller than the dimension of the multivariate Gaussian. To overcome this challenge, we split the time course into two subsets, each with half as many time points carefully selected to capture the long range changes in gene expression. After analyzing each subset individually, the results were combined using Intersection Union Tests (IUTs). We adapted both the frequentist and the empirical Bayes FDR-controlling procedures for the genome-wide IUTs. The top ranked genes selected by each method were compared by visual inspection. Our results suggest that the empirical Bayes FDR-controlling procedure is more powerful when the sample size is small.

The main limitation of this method is that hyperparameter estimation is unreliable when the number of replicates is small, i.e. less than or equal to the number of time points. In the univariate version of the hierarchical Bayesian model, the prior distribution on σ^2 is an inverse-gamma, with the shape parameter equal to one-half of ν. Smyth [58] developed an empirical method for estimating ν by taking linear approximations to the log of the sample variance. This method was adopted for the multivariate model, to obtain an estimate of ν based on each diagonal element of the sample covariance matrix. The final estimate of ν is an average of the estimates obtained from the individual diagonal elements [67]. We observed through some simulations that the prior degrees of freedom tend to be under-estimated, possibly because the off-diagonal elements are ignored. Improvements to the estimation of ν, especially in small samples, would be desirable in future research.

5.2 Nonhomogeneous HMM for ChIP-chip data

Chromatin immunoprecipitation on chip (ChIP-chip) experiments are often performed to detect the genome-wide localization sites of DNA-binding proteins. Tiling arrays containing probes that are densely placed along the chromosomes are often used in ChIP-chip studies. The existing methods for analyzing ChIP-chip data cannot integrate experiments performed on tiling arrays with different probe designs. In Chapter 3 of this thesis, we proposed a nonhomogeneous hidden Markov model to integrate the tiling array data of different designs. At the center position of each probe, the binding status of the protein is represented by a hidden state. Emission of the observations depends on which design of tiling arrays was used in the experiment. We derived a modified Baum-Welch algorithm for fitting the nonhomogeneous HMM. This algorithm involves a linear approximation to the matrix exponential. Simulation results suggested that the algorithm is well-behaved under the setting relevant to our study, and that the inference errors associated with the linear approximation are negligible.

The posterior probabilities of the hidden states are used to identify the candidate probes. We developed a two-step procedure that converts the candidate probes into peaks. In the first step, the candidate probes are joined into intervals according to some conventional criteria. In the second step, a score is computed for each interval based on both the signal strength and the signal pattern. Intervals that pass a filter on the score are called peaks,

which are the putative binding sites. We curated a set of positive and negative regions from a *Caenorhabditis elegans* data set by visual inspection. We then performed some ROC analyses to compare our method with the three best existing methods, using the curated "standards" as the benchmark. When applied to each experiment individually, our method performs similarly as the best existing methods. When applied to the combined data set, which consists of replicate experiments done on tiling arrays of different designs, our method has a pronounced improvement in performance.

Many important biological processes require the concerted functions of multiple proteins in a complex. Transcriptional regulation by DNA-binding proteins is no exception. A major challenge in ChIP-chip data analysis is that the existing methods do not provide the option of analyzing multiple proteins simultaneously. This makes the determination of joint binding sites very difficult. The current practice is to analyze each protein separately, and then cross-list the peak calls of the individual proteins. In Chapter 4, we proposed a generalization of the nonhomogeneous hidden Markov model that enables the joint analysis of ChIP-chip data for multiple proteins. This model assumes that each protein emits observations independently of the other proteins, conditional on the hidden states. Simulation results suggested that the conditional independence assumption is applicable, and that our algorithm is useful for identifying the shared binding sites of different proteins. We illustrated the usage of this method through a concrete example of two proteins. ROC curves produced using the curated "standards" demonstrated that our joint analysis method out-performs the alternative cross-listing approach.

A new technology called ChIP-seq has emerged due to the recent advancements in next-generation sequencing technologies. Instead of hybridizing the immunoprecipitated DNA fragments to microarrays, the samples are analyzed by multiplex short-read DNA sequencing. The advantages of ChIP-seq over ChIP-chip include higher resolution up to the single base level, more comprehensive coverage of the genome, and lower requirement for the amount of immunoprecipitated DNA. Currently, ChIP-seq has limited accessibility due to high costs. It is conceivable that ChIP-seq might replace ChIP-chip when it becomes more accessible in the future [49]. Because the two technologies address essentially the same biological questions, the need for joint analyses of multiple proteins also exists in studies done with ChIP-seq. Since ChIP-seq generates short reads that could be mapped to arbitrary locations in the genome, each experiment constitutes a unique "probe design." The nonhomogeneous hidden Markov model proposed in this thesis may be adapted for ChIP-seq data in future research. This requires replacing the Gaussian emission distribution with another distribution that is compatible with ChIP-seq data. Since the post-alignment data are counts, the Poisson distribution and the negative binomial distribution are often used to model ChIP-seq data. When choosing a proper emission distribution for the nonhomogeneous hidden Markov model, feasibility of the parameter estimation algorithm needs to be considered.

Bibliography

[1] Z. Bar-Joseph. Analyzing time series gene expression data. *Bioinformatics*, 20(16):2493–2503, 2004.

[2] L. E. Baum, T. Petrie, G. Soules, and N. Weiss. A Maximization Technique Occurring in the Statistical Analysis of Probabilistic Functions of Markov Chains. *The Annals of Mathematical Statistics*, 41(1):164–171, 1970.

[3] Y. Benjamini and Y. Hochberg. Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing. *Journal of the Royal Statistical Society. Series B (Methodological)*, 57(1):289–300, 1995.

[4] R. L. Berger. Likelihood Ratio Tests and Intersection-Union Tests. In S. Panchapakesan and N. Balakrishnan, editors, *Advances in Statistical Decision Theory and Applications*, pages 225–237. Birkhauser Verlag, 1997.

[5] R. L. Berger and D. F. Sinclair. Testing Hypotheses Concerning Unions of Linear Subspaces. *Journal of the American Statistical Association*, 79(385):158–163, 1984.

[6] M. Bieda, X. Xu, M. A. Singer, R. Green, and P. J. Farnham. Unbiased location analysis of E2F1-binding sites suggests a widespread role for E2F1 in the human genome. *Genome Research*, 16(5):595–605, 2006.

[7] B. Bolstad, F. Collin, J. Brettschneider, K. Simpson, L. Cope, R. Irizarry, and T. Speed. Quality Assessment of Affymetrix GeneChip Data. In R. Gentleman, V. J. Carey, W. Huber, R. A. Irizarry, and S. Dudoit, editors, *Bioinformatics and Computational Biology Solutions Using R and Bioconductor*, Statistics for Biology and Health, pages 33–47. Springer New York, 2005.

[8] B. Bolstad, R. Irizarry, M. Astrand, and T. Speed. A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. *Bioinformatics*, 19(2):185–193, 2003.

[9] B. Bolstad, R. Irizarry, L. Gautier, and Z. Wu. Preprocessing High-density Oligonucleotide Arrays. In R. Gentleman, V. J. Carey, W. Huber, R. A. Irizarry, and S. Dudoit, editors, *Bioinformatics and Computational Biology Solutions Using R and Bioconductor*, Statistics for Biology and Health, pages 13–32. Springer New York, 2005.

[10] J. Boros, I. J. Donaldson, A. O'Donnell, Z. A. Odrowaz, L. Zeef, M. Lupien, C. A. Meyer, X. S. Liu, M. Brown, and A. D. Sharrocks. Elucidation of the ELK1 target gene network reveals a role in the coordinate regulation of core components of the gene regulation machinery. *Genome Research*, 19(11):1963–1973, 2009.

[11] M. Buck, A. Nobel, and J. Lieb. ChIPOTle: a user-friendly tool for the analysis of ChIP-chip data. *Genome Biology*, 6(11):R97, 2005.

[12] M. J. Buck and J. D. Lieb. ChIP-chip: considerations for the design, analysis, and application of genome-wide chromatin immunoprecipitation experiments. *Genomics*, 83(3):349 – 360, 2004.

[13] M. L. Bulyk. DNA microarray technologies for measuring protein-DNA interactions. *Current Opinion in Biotechnology*, 17(4):422 – 430, 2006. Protein technologies.

[14] S. Cawley, S. Bekiranov, H. H. Ng, P. Kapranov, E. A. Sekinger, D. Kampa, A. Piccolboni, V. Sementchenko, J. Cheng, A. J. Williams, R. Wheeler, B. Wong, J. Drenkow, M. Yamanaka, S. Patel, S. Brubaker, H. Tammana, G. Helt, K. Struhl, and T. R. Gingeras. Unbiased Mapping of Transcription Factor Binding Sites along Human Chromosomes 21 and 22 Points to Widespread Regulation of Noncoding RNAs. *Cell*, 116(4):499 – 509, 2004.

[15] D. Chandran, Y. C. Tai, G. Hather, J. Dewdney, C. Denoux, D. G. Burgess, F. M. Ausubel, T. P. Speed, and M. C. Wildermuth. Temporal Global Expression Data Reveal Known and Novel Salicylate-Impacted Processes and Regulators Mediating powdery Mildew Growth and Reproduction on Arabidopsis. *Plant Physiol.*, 149(3):1435–1451, 2009.

[16] P.-T. Chuang, D. G. Albertson, and B. J. Meyer. DPY-27: A chromosome condensation protein homolog that regulates *C. elegans* dosage compensation through association with the X chromosome. *Cell*, 79(3):459 – 474, 1994.

[17] G. Csankovszki, P. McDonel, and B. J. Meyer. Recruitment and Spreading of the *C. elegans* Dosage Compensation Complex Along X Chromosomes. *Science*, 303(5661):1182–1185, 2004.

[18] T. Davis and B. Meyer. SDC-3 coordinates the assembly of a dosage compensation complex on the nematode X chromosome. *Development*, 124(5):1019–1031, 1997.

[19] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977.

[20] J. Du, J. S. Rozowsky, J. O. Korbel, Z. D. Zhang, T. E. Royce, M. H. Schultz, M. Snyder, and M. Gerstein. A supervised hidden markov model framework for efficiently segmenting tiling array data in transcriptional and ChIP-chip experiments: systematically incorporating validated biological knowledge. *Bioinformatics*, 22(24):3016–3024, 2006.

[21] S. Dudoit, M. van der Laan, and K. Pollard. Multiple Testing. Part I. Single-Step Procedures for Control of General Type I Error Rates. *Statistical Applications in Genetics and Molecular Biology*, 3, 2004.

[22] S. Dudoit, M. van der Laan, and K. Pollard. Multiple Testing. Part III. Procedures for Control of the Generalized Family-Wise Error Rate and Proportion of False Positives. *Statistical Applications in Genetics and Molecular Biology*, 3, 2004.

[23] S. Dudoit and M. J. van der Laan. *Multiple Testing Procedures with Applications to Genomics*. Springer Series in Statistics. Springer, 2008.

[24] B. Efron. Correlation and Large-Scale Simultaneous Significance Testing. *Journal of the American Statistical Association*, 102(477):93–103, 2007.

[25] B. Efron and R. Tibshirani. Empirical Bayes Methods and False Discovery Rates for Microarrays. *Genetic Epidemiology*, 23(1):70–86, 2002.

[26] B. Efron, R. Tibshirani, J. D. Storey, and V. Tusher. Empirical Bayes Analysis of a Microarray Experiment. *Journal of the American Statistical Association*, 96(456):1151–1160, 2001.

[27] S. Ercan, P. G. Giresi, C. M. Whittle, X. Zhang, R. D. Green, and J. D. Lieb. X chromosome repression by localization of the *C. elegans* dosage compensation machinery to sites of transcription initiation. *Nature Genetics*, 39:403–408, 2007.

[28] P. Erdös and A. Rényi. On a new law of large numbers. *Journal d'Analyse Mathématique*, 23:103–111, 12 1970.

[29] J. Fridlyand, A. M. Snijders, D. Pinkel, D. G. Albertson, and A. N. Jain. Hidden Markov models approach to the analysis of array CGH data. *Journal of Multivariate Analysis*, 90(1):132 – 153, 2004. Special Issue on Multivariate Methods in Genomic Data Analysis.

[30] L. Gordon, M. F. Schilling, and M. S. Waterman. An extreme value theory for long head runs. *Probability Theory and Related Fields*, 72(2):279–297, 6 1986.

[31] R. Gottardo, W. Li, W. E. Johnson, and X. S. Liu. A Flexible and Powerful Bayesian Hierarchical Model for ChIP-chip Experiments. *Biometrics*, 64(2):468–478, 2008.

[32] A. K. Gupta and D. K. Nagar. *Matrix Variate Distributions*. Chapman and Hall/CRC, 2000.

[33] P. C. Hollenhorst, A. A. Shah, C. Hopkins, and B. J. Graves. Genome-wide analyses reveal properties of redundant and specific promoter occupancy within the ETS gene family. *Genes and Development*, 21(15):1882–1894, 2007.

[34] H. Ji, H. Jiang, W. Ma, D. S. Johnson, R. M. Myers, and W. H. Wong. An integrated software system for analyzing ChIP-chip and ChIP-seq data. *Nature Biotechnology*, 26(11):1293–1300, November 2008.

[35] H. Ji and W. H. Wong. TileMap: create chromosomal map of tiling array hybridizations. *Bioinformatics*, 21(18):3629–3636, 2005.

[36] D. S. Johnson, W. Li, D. B. Gordon, A. Bhattacharjee, B. Curry, J. Ghosh, L. Brizuela, J. S. Carroll, M. Brown, P. Flicek, C. M. Koch, I. Dunham, M. Bieda, X. Xu, P. J. Farnham, P. Kapranov, D. A. Nix, T. R. Gingeras, X. Zhang, H. Holster, N. Jiang, R. D. Green, J. S. Song, S. A. McCuine, E. Anton, L. Nguyen, N. D. Trinklein, Z. Ye, K. Ching, D. Hawkins, B. Ren, P. C. Scacheri, J. Rozowsky, A. Karpikov, G. Euskirchen, S. Weissman, M. Gerstein, M. Snyder, A. Yang, Z. Moqtaderi, H. Hirsch, H. P. Shulha, Y. Fu, Z. Weng, K. Struhl, R. M. Myers, J. D. Lieb, and X. S. Liu. Systematic evaluation of variability in ChIP-chip experiments using predefined DNA targets. *Genome Research*, 18(3):393–403, 2008.

[37] W. E. Johnson, W. Li, C. A. Meyer, R. Gottardo, J. S. Carroll, M. Brown, and X. S. Liu. Model-based analysis of tiling-arrays for ChIP-chip. *Proceedings of the National Academy of Sciences*, 103(33):12457–12462, 2006.

[38] S. Keles. Mixture Modeling for Genome-Wide Localization of Transcription Factors. *Biometrics*, 63(1):10–21, 2007.

[39] B. L. Kidder, J. Yang, and S. Palmer. Stat3 and c-Myc Genome-Wide Promoter Occupancy in Embryonic Stem Cells. *PLoS ONE*, 3(12):e3932, 12 2008.

[40] T. H. Kim, L. O. Barrera, M. Zheng, C. Qu, M. A. Singer, T. A. Richmond, Y. Wu, R. D. Green, and B. Ren. A high-resolution map of active promoters in the human genome. *Nature*, 436:876–880, 2005.

[41] W. Li, C. A. Meyer, and X. S. Liu. A hidden Markov model for analyzing ChIP-chip experiments on genome tiling arrays and its application to p53 binding sequences. *Bioinformatics*, 21(suppl-1):i274–282, 2005.

[42] X.-y. Li, S. MacArthur, R. Bourgon, D. Nix, D. A. Pollard, V. N. Iyer, A. Hechmer, L. Simirenko, M. Stapleton, C. L. L. Hendriks, H. C. Chu, N. Ogawa, W. Inwood, V. Sementchenko, A. Beaton, R. Weiszmann, S. E. Celniker, D. W. Knowles, T. Gingeras, T. P. Speed, M. B. Eisen, and M. D. Biggin. Transcription Factors Bind Thousands of Active and Inactive Regions in the *Drosophila* Blastoderm. *PLoS Biol*, 6(2):e27, 02 2008.

[43] I. Lucas, A. Palakodeti, Y. Jiang, D. J. Young, N. Jiang, A. A. Fernald, and M. M. Le Beau. High-throughput mapping of origins of replication in human cells. *EMBO Reports*, 8:770–777, 2007.

[44] J. C. Marioni, N. P. Thorne, and S. Tavare. BioHMM: a heterogeneous hidden Markov model for segmenting array CGH data. *Bioinformatics*, 22(9):1144–1146, 2006.

[45] R. L. McCaffrey, P. Fawcett, M. O'Riordan, K.-D. Lee, E. A. Havell, P. O. Brown, and D. A. Portnoy. A specific gene expression program triggered by Gram-positive bacteria

in the cytosol. *Proceedings of the National Academy of Sciences of the United States of America*, 101(31):11386–11391, 2004.

[46] B. J. Meyer. X-Chromosome dosage compensation. *WormBook*, June 2005.

[47] T. C. Mockler and J. R. Ecker. Applications of DNA tiling arrays for whole-genome analysis. *Genomics*, 85(1):1 – 15, 2005.

[48] R. J. Muirhead. *Aspects of Multivariate Statistical Theory*. John Wiley & Sons, Inc., 2005.

[49] P. J. Park. ChIP-seq: advantages and challenges of a maturing technology. *Nature Reviews Genetics*, 10:669–680, 2009.

[50] E. L. Petty, K. S. Collette, A. J. Cohen, M. J. Snyder, and G. Csankovszki. Restricting Dosage Compensation Complex Binding to the X Chromosomes by H2A.Z/HTZ-1. *PLoS Genet*, 5(10):e1000699, 10 2009.

[51] K. S. Pollard and M. J. van der Laan. Choice of a null distribution in resampling-based multiple testing. *Journal of Statistical Planning and Inference*, 125(1-2):85 – 100, 2004. The Third International Conference on Multiple Comparisons.

[52] Y. Qi, A. Rolfe, K. D. MacIsaac, G. K. Gerber, D. Pokholok, J. Zeitlinger, T. Danford, R. D. Dowell, E. Fraenkel, T. S. Jaakkola, R. A. Young, and D. K. Gifford. High-resolution computational models of genome binding events. *Nature Biotechnology*, 24:963–970, 2006.

[53] L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, August 1989.

[54] A. Rada-Iglesias, A. Ameur, P. Kapranov, S. Enroth, J. Komorowski, T. R. Gingeras, and C. Wadelius. Whole-genome maps of USF1 and USF2 binding and histone H3 acetylation reveal new aspects of promoter structure and candidate genes for common human disorders. *Genome Research*, 18(3):380–392, 2008.

[55] D. J. Reiss, M. T. Facciotti, and N. S. Baliga. Model-based deconvolution of genome-wide DNA binding. *Bioinformatics*, 24(3):396–403, 2008.

[56] P. C. Scacheri, G. E. Crawford, and S. Davis. Statistics for ChIP-chip and DNase Hypersensitivity Experiments on NimbleGen Arrays. In A. Kimmel and B. Oliver, editors, *DNA Microarrays, Part B: Databases and Statistics*, volume 411 of *Methods in Enzymology*, pages 270 – 282. Academic Press, 2006.

[57] M. Schena, D. Shalon, R. W. Davis, and P. O. Brown. Quantitative Monitoring of Gene Expression Patterns with a Complementary DNA Microarray. *Science*, 270(5235):467–470, 1995.

[58] G. K. Smyth. Linear models and empirical Bayes methods for assessing differential expression in microarray experiments. *Statistical Applications in Genetics and Molecular Biology*, 3(1), February 2004.

[59] G. K. Smyth, Y. H. Yang, and T. Speed. Statistical Issues in cDNA Microarray Data Analysis. In M. J. Brownstein and A. B. Khodursky, editors, *Functional Genomics*, Methods in Molecular Biology, pages 111–136. Humana Press, 2003.

[60] J. Song, W. E. Johnson, X. Zhu, X. Zhang, W. Li, A. Manrai, J. Liu, R. Chen, and X. S. Liu. Model-based analysis of two-color arrays (MA2C). *Genome Biology*, 8(8):R178, 2007.

[61] P. T. Spellman, G. Sherlock, M. Q. Zhang, V. R. Iyer, K. Anders, M. B. Eisen, P. O. Brown, D. Botstein, and B. Futcher. Comprehensive Identification of Cell Cycle-regulated Genes of the Yeast Saccharomyces cerevisiae by Microarray Hybridization. *Mol. Biol. Cell*, 9(12):3273–3297, 1998.

[62] J. Storey. A direct approach to false discovery rates. *Journal of the Royal Statistical Society Series B (Statistical Methodology)*, 64:479–498(20), 2002.

[63] T. Suganuma and J. L. Workman. Crosstalk among Histone Modifications. *Cell*, 135(4):604 – 607, 2008.

[64] W. Sun, M. J. Buck, M. Patel, and I. J. Davis. Improved ChIP-chip analysis by a mixture model approach. *BMC Bioinformatics*, 10(173), 2009.

[65] Y. C. Tai. *Multivariate Empirical Bayes Models for Replicated Microarray Time Course Data*. PhD thesis, University of California, Berkeley, 2005.

[66] Y. C. Tai and T. P. Speed. Statistical analysis of microarray time course data. In *DNA Microarrays*, chapter 20. BIOS Scientific Publishers Limited, Taylor & Francis, 2005.

[67] Y. C. Tai and T. P. Speed. A multivariate empirical Bayes statistic for replicated microarray time course data. *Annals of Statistics*, 34(5):2387–2412, 2006.

[68] Y. C. Tai and T. P. Speed. On Gene Ranking Using Replicated Microarray Time Course Data. *Biometrics*, 65(1):40–51, 2009.

[69] Z. Zhang, J. Rozowsky, H. Lam, J. Du, M. Snyder, and M. Gerstein. Tilescope: online analysis pipeline for high-density tiling microarray data. *Genome Biology*, 8(5):R81, 2007.

[70] M. Zheng, L. O. Barrera, B. Ren, and Y. N. Wu. ChIP-chip: Data, Model, and Analysis. *Biometrics*, 63(3):787–796, 2007.