# Detection of copy number variation in the mouse genome

Avigail Agam
Life Science Interface Doctoral Training Centre
Department of Statistics
Wellcome Trust Centre for Human Genetics
University of Oxford

May 10, 2007

**Acknowledgements**

I would like to thank Dr Chris Holmes and Dr Richard Mott for their supervision. I would also like to thank Prof Jonathan Flint for his collaborations, and Dr Binnaz Yalzin for her ongoing experimental work to validate the results of this project. Finally I would like to thank Chris Yau and Alex Webb for their helpful discussions.

## Abstract

Copy number variation (CNV) has been implicated in gene expression changes, phenotypic variation and increased risk for complex disease traits, so there is much motivation for the study of CNV and the relationship between it and various phenotypic phenomenon. Presented here is the analysis of an array comparative genomic hybridisation (aCGH) experiment that was carried out using representational oligonucleotide microarray analysis (ROMA) to detect CNV in the mouse genome. The experiment used the inbred mouse strain C57BL/6J as the reference strain, and seven other inbred mouse strains as the test strains.

First, a literature review of existing methods for aCGH data analysis is given. Next, the ROMA data is introduced, and interference in the ROMA data due to single nucleotide polymorphisms (SNPs) is discussed. Then, a nonparametric thresholding method for CNV detection, which searches for runs of probes whose $\log_2$ ratios lie above or below a set threshold (*excursions*), **Excursion Finder** (**EF**), is presented; the method is novel because it integrates known SNP data for the eight strains with the ROMA data. Results are presented: the putative CNVs located by EF are discussed and compared to the results of another nonparametric method, SW-ARRAY (Price et al., 2005); and the relationship between CNVs and quantitative trait loci (QTLs), and expression QTLs (eQTLs), which have been previously mapped to the mouse genome, is analysed. At the end of the report a plan for the completion of this part of the project is given, harder extensions to the work are proposed, and a long term goal for the development of a Bayesian nonparametric segmentation method for high density sequence data is discussed.

# Contents

# Chapter 1

# Introduction

When a cell replicates its DNA several types of chromosomal changes can occur. Feuk et al. (2006) give the following classifications and definitions of these structural changes. Some variants, involving $\sim$ 3 Mb or more, are large enough to be seen under a microscope, and include abnormal numbers of chromosomes *aneuploidies*, chromosomal *rearrangements*, regions of chromosomes that vary in size or morphology *heteromorphisms* and breaks or constrictions in chromosomes *fragile sites*. Other structural variants involving much shorter segments of DNA, typically $< 1$ Kb, are detectable by PCR sequencing and include *insertions*, *deletions*, *duplications* and *inversions*. Finally there are also structural variants, between $\sim$ 1 Kb and 3 Mb in size, which are still submicroscopic but too large for detection by sequencing. The development of array comparative genomic hybridisation (aCGH) technologies have enabled the study of this type of variant. (Briefly, in aCGH probes that map to loci on the genome of interest are printed on to slides and used as targets for hybridisation of fluorescent test and control DNA samples. The dyes emit different wavelengths of light and the ratio of the intensity of the light emitted is measured to detect differential abundance of sequences in the test and control samples. Lastly the ratios are normalised and $\log_2$s transformed for input into downstream analysis of the data. See chapter 2 for further details.)

Useful definitions of structural variants that are $> 1$ Kb are found in Feuk et al. (2006) and Molinaro et al. (2002) and are summarised here. During cell replication, if a region of DNA fails to be replicated then there is a *deletion* or *loss* at that locus. Alternatively a *duplication* or, more generally, a *gain* in copy number can occur if a region is copied more than once (see figure 1.1). Together with *insertions* these types of structural variants are termed copy number variants (CNVs). For brevity, in this remainder of this report, gains in copy number variation will be termed gain CNVs, and losses in CNV will be termed loss CNVs. The detection of loss and gain CNVs in the mouse genome is the primary focus of this project. Examples of other structural variants $> 1$ Kb are: *segmental duplications* - a region of DNA that has two or more 90% sequence identical copies per haploid genome; *inversions* - a segment of DNA with a reverse orientation in comparison to the rest of the chromosome; *translocations* - in which a region of DNA changes position in the genome.

CNVs have for a long time been known to be associated with many human diseases. For example several developmental disorders, such as Down, Prader Willi and Angelman syndromes are caused by a deletion or duplication of a chromosome or part of a chromosome. Additionally, cancer is known to be caused, in part, by mutations in oncogenes and tumour suppressor genes. One such mutation that these genes are susceptible to is CNV, and this contributes to their change in expression levels between normal and cancerous cells.

More recently Redon et al. (2006) conducted an aCGH study of global variation in the human genome using the International HapMap DNA and cell-line collection (Consortium, 2005). As motivation for their work the authors site many publications that have implicated CNVs in gene expression changes, phenotypic variation and increased risk for complex disease traits. In the first part of the paper a genome-wide map of CNV in the human genome is presented (the first of its kind), and CNV regions (CNVR: a region which encompasses overlapping or adjacent CNVs) are reported to cover 12% of the human genome. The authors find that CNVs are associated with segmental duplications, that there are functional categories of genes that are either enriched or underrepresented in CNVs, and that there are "numerous examples of possible relevance [of CNVs] to both Mendelian and complex diseases". Importantly the authors also find that single nucleotide polymorphism (SNP) genotype patterns are perturbed by CNVs. Finally they report marked variation in CNV between populations.

Using the HapMap SNPs and the CNV map reported in Redon et al. (2006), Stranger et al. (2007) have examined the separate and joint effects of SNPs and CNVs on gene expression phenotypes. The authors report that both types of variation have an effect on gene expression, but that these effects are largely mutually exclusive.

Thus there is much motivation for the study of CNV and the relationship between it and various phenotypic phenomenon.

## 1.1 Report summary

Presented here is the analysis of an aCGH experiment that was carried out using representational oligonucleotide microarray analysis (ROMA) to detect CNV in the mouse genome. In ROMA representations of a genome are made using PCR to amplify fragments of the genome previously made with a restriction

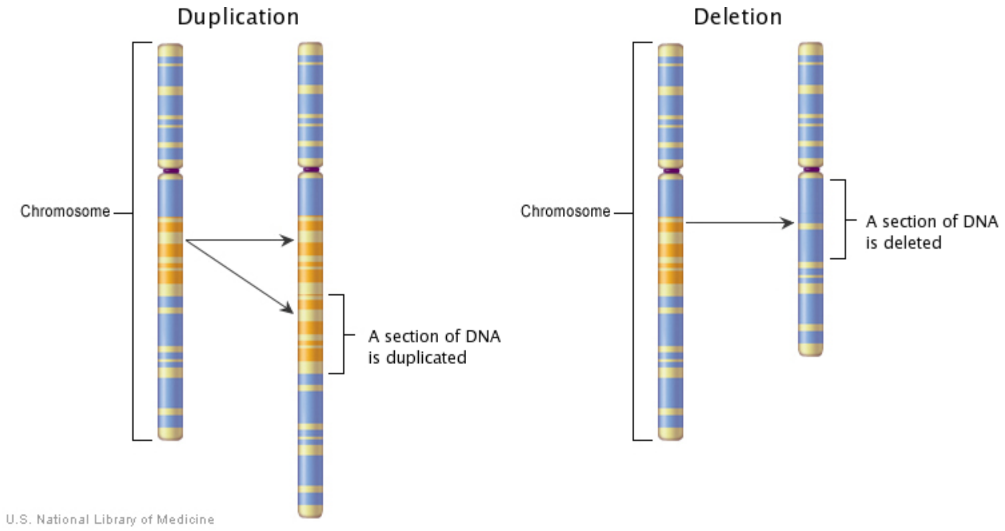
Duplication
Chromosome
A section of DNA is duplicated
Deletion
Chromosome
A section of DNA is deleted
U.S. National Library of Medicine

Figure 1.1: *Gain* and *loss* CNVs. Left: a gain (in this case *duplication*) CNVs depicted (courtesy of the US National Library of Medicine). Right: a loss (*deletion*) CNV is shown (doctored image from the US National Library of Medicine).

endonuclease, and then the representation, rather than the whole genome, is hybridised to an array of long oligonucleotide probes. The ROMA experiment was carried out on seven inbred mouse strains, with an eighth inbred mouse strain as the reference. The seven mouse strains under analysis were A/J, AKR/J, BALB/cJ, C3H/HeJ, CBA/J, DBA/2J and LP/J, and the reference was C57BL/6J.

The goal in the analysis of aCGH data is the automatic detection of CNV in the genome under inspection. Therefore, in chapter 2, aCGH experimental methods are presented, and existing methods for their data analysis are discussed.

As will be discussed in chapter 3, there is a lot of observed variance in the ROMA data, much more so than in previous mouse bacterial artificial chromosomes(BAC) aCGH (Li et al. (2004), Snijders et al. (2004)), human ROMA (Lucito et al., 2003) and mouse ROMA (Lakshmi et al., 2006) experiments. Additionally the ROMA data are of a much higher density than that for which most bioinformatic analytical methods have been designed for, and break the distributional assumption of normality that most methods make. Thus existing techniques for locating CNV are not suitable for this data.

Furthermore, as discussed in section 3.4, a primary cause of the variance in the data is thought to be SNPs. SNPs can cause unwanted variance in one of two ways. First, if there is a SNP in the binding site of the restriction endonuclease then the corresponding fragment will be removed from the genome representation, and will appear as a total deletion. Second, if there is a SNP in an array probe then hybridisation will be reduced, but the amount by which this will occur is hard to predict.

Therefore a nonparametric thresholding method for CNV detection, which searches for runs of probes whose $\log_2$ ratios lie above or below a set threshold (*excursions*), **Excursion Finder**, is presented in chapter 4. The method is novel because it integrates known SNP data for the eight strains with the ROMA data. Due to Wade et al. (2002), the inbred mouse genome is known to have a mosaic structure such that if the genomes of two inbred strains are compared to one another they are found to consist of large regions with *either* very low SNP rates (*SNP matched*) or very high SNP rates (*SNP non-matched*). Therefore in the first part of the algorithm each test strain is compared to the reference strain to find their SNP matched and non-matched regions. Next, with the aim of explaining that proportion of the variance in the ROMA data which is due to SNPs, different thresholds are set for the ROMA data in the SNP matched and non-matched regions, with higher thresholds in the non-matched regions accounting for the extra variance observed in them. Finally, a permutation algorithm is used to assess the significance of putative regions of CNV.

Excursion Finder highlights many CNVs. A selection of regions are currently being verified with other methods (PCR, flourescent in situ hybridisation (FISH), multiplex ligation-dependent probe amplification (MLPA, Schouten et al. (2002))). The results are not yet available, but will eventually enable the assessment of the reliability of the method in terms of, for example, estimates of the false positive rate.

To help assess the reliability of the method before experimental verifications are available, the CNVs found by Excursion Finder on different strains are compared to one another. In section 4.2, using criteria similar to those given in Redon et al. (2006), CNVs are combined across strains to obtain CNV sets (CNVSs) that combine overlapping or adjacent CNVs. The proportion of CNVs that form singleton CNVSs gives initial, if rather strict, estimates of the false positive rates of discovery of loss and gain CNVs.

In section 4.3 the CNVs found by Excursion Finder are compared to those obtained from an SW-ARRAY (Price et al., 2005) analysis. SW-ARRAY is used for comparison firstly because it is the only nonparametric segmentation procedure that also provides a nonparametric test for significance, and secondly because it too has been used to process very high density aCGH data (Redon et al. (2006), Komura et al. (2006)). In terms of loss CNVs detected, SW-ARRAY finds fewer regions than Excursion Finder, but the CNVs detected by SW-ARRAY cover a much larger percentage of the C57BL/6J genome than those detected by Excursion Finder. Both methods find far fewer gain CNVs than loss CNVs, but Excursion Finder finds more, both in number and in terms of percentage of genome covered by them. This analysis is limited and only experimental verification of the extra regions found by Excursion Finder can discern false positives in Excursion Finder from false negatives in SW-ARRAY.

Once CNVSs have been identified the next step, discussed in section 4.4, is to assess the association between them and other regions of the mouse genome that are related to certain phenotypes. (The use of CNVSs over CNVs has two advantages: first, they provide a simpler, coarse grained, input for the downstream analysis; second it becomes easy to remove CNVs for which there is not a lot of evidence, just by removing those which form small CNVSs).Using a genetically heterogeneous stock (HS) of mice descended from the eight strains listed above, small effect quantitative trait loci (QTL) have previously been fine-mapped to the C57BL/6J genome (Solberg et al. (2006), Valdar et al. (2006a), Valdar et al. (2006b)). The phenotypes that were studied are of relevance to human health and target three diseases: anxiety, type II diabetes and asthma. A permutation test has been developed to test the association between regions of CNV and QTL. At the moment, the results show little association between QTLs and CNVs.

Finally, treating gene expression as a phenotype, expression QTLs (eQTLs) have also been identified for the HS mice and mapped to C57BL/6J (data not yet published). In section 4.5 the hypothesis that eQTLs are not enriched in CNVs, with the alternative that they are, is tested. Significant evidence is found to reject the null hypothesis of no enrichment, so there is reason to believe that there is a relationship between CNVs and eQTLS.

In chapter 5 future work is discussed. A plan for the completion of this part of the project is given, harder extensions to the work are proposed, and a long term goal for the development of a Bayesian nonparametric segmentation method for high density, high variance, sequence data is presented.

# Chapter 2

# Background

## 2.1 Chromosomal comparative genomic hybridisation

Comparative Genomic Hybridisation (CGH) is a method for detecting and mapping deletions, duplications and translocations within the genome. Initially fluorescence *in situ* hybridisation (FISH) was used with intact chromosomes as the hybridisation targets to map gains and losses of DNA copy number. Briefly, test and control DNAs labelled with different fluorescent dyes are applied to metaphase chromosomes. Next, laser beams that correspond to the excitation wavelength of the dyes are shone onto the chromosomes. The dyes then emit different wavelengths of light and the ratio of the intensity of the light emitted is measured to detect differential abundance of sequences in the test and control samples. The main shortfalls of this method are its low resolution (it is estimated to have a lower limit of $\sim$ 5Mb), thus only enabling the detection of comparatively large gains or losses, and that it provides little information regarding the locations of the ends of the diversified regions. ((Molinaro et al., 2002), (Lupski and White)).

## 2.2 Array comparative genomic hybridisation

In array comparative genomic hybridisation (aCGH), instead of using chromosomes, probes that map to loci on the genome are printed on to slides and used as targets for the hybridisation of the fluorescent DNA samples. BACs, cDNAs, PCR products and oligonucleotides can all be used as array probes. (See Box 2.1 for an outline of the aCGH method).

This method has afforded several improvements over chromosome CGH, including higher resolution, direct mapping of abnormalities to the genome sequence and, because aCGH enables a genome-wide analysis of DNA sequence copy number in a single experiment, higher throughput. However aCGH cannot highlight ploidy (the number of single sets of chromosomes in a cell or organism) or location of the rearranged sequences that have caused the CNV. Moreover, the resolution of aCGH is dependent upon both the size and the spacing of the probes on the array. ((Molinaro et al., 2002), (Albertson and Pinkel, 2003)).

**Method**

1. A region of the genome is chosen to investigate for CNV. This can be a part of the genome or the whole genome.
2. Probes for loci in the chosen part of the genome are either made and then placed on to glass plates or, as is recently more common, are synthesised in situ with the plates. (This is the microarray).
3. Genomic material from a test and control samples are labelled with different fluorochromes. The fluorochromes are completely distinguishable with no spectral overlap.
4. The samples are hybridised with the array, and the array is washed to remove any DNA that has not hybridised.
5. It is now possible to quantify the signal intensities of the fluorochromes. Typically this is done by adjusting for background signal intensity, averaging ratios for any replicate probes, and normalising signals within and across slides to account for experimental variability.
6. Once the ratios are calculated probes with no ratio are either removed or a signal is imputed for them, and a log transform is taken.
7. The data is then examined to find CNV.

Box 2.1: Outline of the array CGH method (Molinaro et al., 2002)

### 2.2.1 BAC aCGH

BAC aCGH was one of the first implementations of aCGH and is popular because it provides extensive genome coverage, with reliable mapping data and readily available probes (the BACs). BAC arrays usually have approximately 3000 probes, although sometimes they can have an order of magnitude more. The size of the probes on BAC arrays is usually 150 to 200 kb, which is a limiting factor in the resolution of smaller CNVs.

### 2.2.2 Long oligonucleotide aCGH

Long oligonucleotide (60-100 bp) arrays provide a denser coverage of the genome than that achieved by BAC arrays and also improve the detection resolution (30 to 50 kb). Such arrays were first implemented in an assay format known as representational oligonucleotide microarray analysis (ROMA) Lucito et al. (2003).

In ROMA representations of a genome are made using PCR to amplify fragments of the genome previously made with a restriction endonuclease. Because PCR selects short fragments, and because the cleavage site of the restriction enzyme is known, the resulting set of representations are short fragments of DNA that are both predictable from the genome sequence and also reproducible. Since the set of representations are predictable it is possible to design oligonucleotide probes that will hybridise to the representations and that will have a minimal amount of sequence overlap with the rest of the genome. The probes can then be mapped computationally to the genome of interest. Furthermore, because representations reduce the complexity of samples in a repeatable fashion, the signal to noise ratio is increased during hybridisation to array probes. Lastly, by using a representation of genome from which a known subset of fragment representations have been removed, it is possible to assess the non-specific hybridisation to probes and hence to calibrate each of the probes on the oligonucleotide array according to performance. This is much harder to do with BAC arrays.

This report describes the analysis of a ROMA experiment that was carried out to detect novel loci of CNV in inbred mouse strains (figure 2.2 and chapter 3). The experiment, which used 216457 probes, is at least an order of magnitude more dense than all BAC arrays, and is typical of a new generation of very high throughput arrays.

Other non-representational long oligonucleotide arrays have been developed by companies such as NimbleGen and Agilent (Barrett et al. (2004)). A mouse long oligonucleotide whole genome tiling path (WGTP) CGH array, made by NimbleGen and containing $388,852$ probes, has been used by Graubert et al. (2007) to construct a very high resolution map of CNV and segmental duplication in the mouse genome. Integration of this data set with the ROMA data analysed here will be explored in future work.

### 2.2.3 SNP arrays

Another array CGH approach is to use hybridisation signal intensities from SNP arrays. In this method the signal intensities are compared to average values obtained from control experiments, and differences from the average signal indicate a change in copy number. The advantage of using SNP arrays is that they provide genotype data simultaneously with the CNV data. Redon et al. (2006) used both a SNP array (GeneChip 500K Early Access array), and a BAC-WGTP array ($26,574$ probes) to ascertain the extent and effect of global CNV in the human genome.

## 2.3 Analysing aCGH data

The data generated by the high throughput aCGH methods described in the previous section have motivated the development of many algorithms for their automated analysis and subsequent identification of CNV. All of the methods reviewed here assume input aCGH data that consist of normalised $\log_2$ ratios from test vs control samples, indexed by the physical location of the array probes on the genome. Based

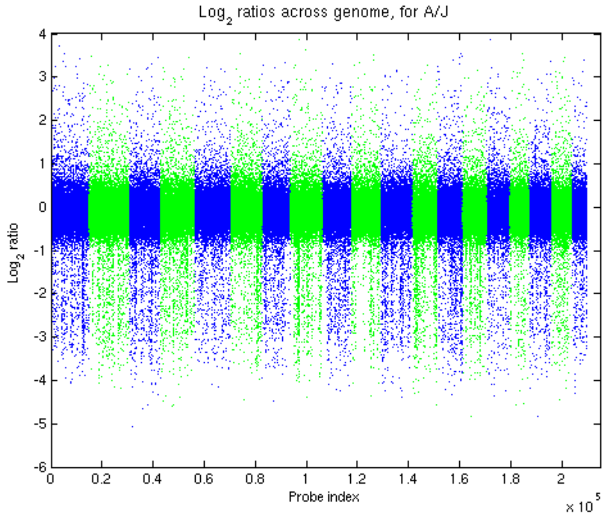

Figure 2.2: $\text{Log}_2$ ratios from the ROMA experiment for mouse strain A/J versus the reference C57BL/6J. There are 209930/216457 probes across the genome for which a signal was obtained in this experiment. $\text{Log}_2$ ratios are plotted against probe indices rather than physical probe locations. Chromosomes are plotted alternately in blue and green for clarity.

on their primary objective most methods can be classified as *smoothing*, *segmentation* or *thresholding.* Another group of methods more informatively grouped by their underlying probabilistic structure are those based on Hidden Markov models (HMMs); importantly, this structure enables HMMs to borrow inferential strength from across the data set in a way that other current methods cannot (see section 2.7 for further discussion). These four categories will be explored here. Lastly a new method, presented by Price et al. (2005), which is better suited to very high throughput aCGH data, and which provides significance ranking for highlighted CNVs (a function not afforded by any other method) will also be discussed.

## 2.4 Smoothing

Smoothing algorithms work on the principal that plots of aCGH data often show regions of constant copy number with abrupt jumps between them, and that they often contain a lot of noise. The primary objective of smoothing algorithms is to provide a visual aid to interpreting the data, and this is achieved by fitting a curve to the data in a process that can handle the sharp transitions. Although these methods do not automatically identify and classify regions of CNV, they do provide an intuitive start to the analytical process. Furthermore, if the smoothing methods are nonparametric then there is no need for the user to provide any input except for the normalised aCGH data. In other words ad hoc pre-processing steps are reduced. Two nonparametric smoothing algorithms are discussed here, quantile smoothing (Eilers and de Menezes, 2004), and wavelet denoising (Hsu et al., 2005).

### 2.4.1 Quantile smoothing of aCGH data

Given a series $y$ of $n$ aCGH data points, Eilers and de Menezes (2004) initially attempt to smooth $y$ by minimising the following objective function, proposed by Whittaker (1923), in which $z$ is the smooth series that approximates $y$:

$$Q_2 = \sum_{i=1}^{n}(y_i - z_i)^2 + \lambda \sum_{i=2}^{n}(z_i - z_{i-1})^2 \tag{2.1}$$

When eq.(2.1) is minimised the first term encourages a close fit of $z$ to $y$, while the second term, tuned by $\lambda$, discourages changes in $z$.

The authors report that this algorithm is not a good choice for aCGH data because “it converts jumps into gradual changes and tends to round plateaus”. Thus they move to the $L_1$ norm and minimise instead the following objective function, with $y$, $n$ and $z$ as above:

$$Q_1 = \sum_{i=1}^{n}|y_i - z_i| + \lambda \sum_{i=2}^{n}|z_i - z_{i-1}| \tag{2.2}$$

Minimisation of eq.(2.2) is reported to yield the desired results, with the smoothed data incorporating the “sudden jumps and flat plateaus” expected to be characteristic of aCGH data. However, as the authors explain, while eq.(2.1) leads to a simple linear system of equations, minimisation of eq.(2.2) is harder; so to do this quantile regression is introduced.

In quantile regression (Koenker and Basset, 1984), given a vector $x$, a regression basis $B$, $m$ regression coefficients $\alpha$ and a parameter $\tau$ that takes values between 0 an 1, the problem is to minimise:

$$S = \sum_{i=1}^{n} \rho_\tau \big( x_i - \sum_{j=1}^{m} b_{ij} \alpha_j \big) \tag{2.3}$$

where

$$\rho_\tau(u) = \begin{cases} \tau u & \text{if } u > 0 \\ (\tau - 1)u & \text{if } u \leq 0 \end{cases} \tag{2.4}$$

The effect of eq.(2.4) in eq.(2.3) is to return weighted absolute values of residuals, with the weight dependent on the original sign of the residuals; positive residuals have weight $\tau$ while negative residuals are weighted $1 - \tau$. However, when $\tau = 0.5$ all absolute residual values receive the same weighting, 0.5, so the weights become independent of sign. This version of the quantile regression problem is termed the median regression problem, and this is the problem solved by Eilers and de Menezes (2004) for smoothing aCGH data.

Looking at the right hand side of eq.(2.2) as a single summation of $2n - 1$ absolute values of terms that are functions of $y$ and $\lambda$, it becomes possible to re-write eq.(2.2) into the median regression problem. This is done by letting:

- $y$, $n$, $z$ and $\lambda$ be defined as above
- $I$ be the $n$ x $m$ identity matrix
- 0 be a vector of $n - 1$ zeros
- $D$ be a matrix such that $Dz = \Delta z$. (So $D$ is the $(n - 1)$ x $n$ matrix that transforms $z$ into the vector of differences of neighbouring elements in $z$.)

then $y*$ and $B$ are constructed as:

$$y* = \begin{pmatrix} y \\ 0 \end{pmatrix} \text{ and } B = \begin{pmatrix} I \\ \lambda D \end{pmatrix} \tag{2.5}$$

Next, replacing $x$ with $y*$ and setting $\tau$ to 0.5 in eq.(2.3), Eilers and de Menezes (2004) use the linear programming methods described in Portnoy and Koenker (1997) to minimise $S = \sum_{i=1}^{2n-1} \rho_{0.5} \big( y *_i - \sum_{j=1}^{n} b_{ij} \alpha_j \big)$, and hence find the smoothed data series $z$ that is equivalent to the regression coefficients $\alpha$.

Eilers and de Menezes (2004) test quantile smoothing on a subset of data from a BAC aCGH experiment presented by Nakao et al. (2004) that tests 125 samples of colon carcinomas for CNV on BAC arrays with 2120 probes spaced at $\sim$ 1.5 Mb; specifically the test data is that obtained from chromosome 1, which involves 133 probes. They show that their method is useful for detecting both large continuous blocks of CNV and also local changes involving only a few probes. The tool is positioned primarily for visualisation and exploration of aCGH data, and as such the user is expected to segment the data by eye once it has been smoothed, or to use the smoothed data in down stream automated analyses. In this regard the method represents a reduction in functionality in comparison to the thresholding methods, but its nonparametric nature, (in median smoothing the only parameter $\tau$ is set to 0.5), makes it easier to use for the less statistical user and also yields a simple interpretation of the output.

### 2.4.2 Denoising aCGH data using wavelets

A wavelet is a function that is compactly supported and has an average value of zero. Wavelet analysis breaks a signal down into a series of wavelets that are scaled and shifted versions of a specified “mother wavelet”. In this sense wavelet analysis is similar to a Fourier transform that breaks a signal down into a series of sine waves of different frequencies, but where as the Fourier transform only provides information about the frequency domain, wavelets enable the simultaneous acquisition of an association between the space (or time) and frequency domains of a signal. Additionally, a mother wavelet can be chosen such that it is good for analysing signals with sharp edges or discontinuity. This is in contrast to a Fourier transform that is restricted to the use of smooth sine waves.

Hsu et al. (2005) propose a wavelet based method to smooth aCGH data prior to any statistical analyses of, and further inferences on, patterns of CNV in the data. They explain that nonparametric techniques are particularly suitable for data smoothing as they do not impose a parametric model for structures in the data. Furthermore, wavelets are cited as particularly useful due to their ability to handle abrupt changes seen in the data.

Denoting $y_i$ to be the observed copy number change at the $i$-th genomic location $x_i$, for $i = 1, ..., n$, the authors use an additive measurement error model to relate the true copy number at $x_i$, $f(x_i)$, and the observed signal $y_i$:

$$y_i = f(x_i) + \epsilon_i \tag{2.6}$$

where $\{\epsilon_i, i = 1, ..., n\}$ are iid $N(0, \sigma^2)$ and $\sigma$ is the standard deviation. A wavelet analysis is proposed to denoise the $y_i$ and hence recover the true signals $f(x_i)$.

The wavelet family used is the one generated by the Haar function:

$$\psi(u) = \begin{cases} \frac{-1}{\sqrt{2}} & \text{if } -1 < u \leq 0 \\ \frac{1}{\sqrt{2}} & \text{if } 0 < u \leq 1 \\ 0 & \text{otherwise} \end{cases} \tag{2.7}$$

Defining $j \in \mathbb{Z}_+$, $t \in \mathbb{R}$, as indices for scale and location respectively, the family of dyadic dilations and translations of the Haar function, $\psi_{j,t}(u) = 2^{j/2}\psi(2^j u - t)$, is the maximal overlap discrete wavelet transform (MODWT). Next let $W$ denote the $n$ x $n$ orthonormal wavelet transformation matrix, with elements defined by the wavelet basis generated by the family of dilations and translation, and $\alpha_{j,t}$ denote the wavelet coefficients for each member of the wavelet family $\psi_{j,t}$.

The Haar wavelet family is used because the wavelet coefficients are simply, for two adjacent segments of probes, the deviation of each of the segments from the mean of the two segments. Thus transforming the aCGH data to the space and frequency domain with the Haar wavelet family directly measures the difference in the means of adjacent segments. Furthermore the Haar wavelet fits the expected structure of the data, with regions of CNV along the chromosome expected to occur in blocks.

The MODWT is employed, rather than the more computationally efficient discrete wavelet transform (DWT), because it is translation invariant. That is, if $f_\tau$ denotes a translation of $f$, $f_\tau(t) = f(t - \tau)$, then $\alpha_\tau(j, u) = \alpha(j, u - \tau)$. The authors explain that this "eliminates alignment artifacts in the wavelet coefficients that arise from a discrete subsampling". This would be a useful feature if one wanted to compare, in the space and frequency domains only, aCGH data with only one aberration per test sample. However, it is unclear as to how this would be helpful when comparing aCGH data from test samples with more than one aberration, since a direct comparison purely in the space and frequency domains would still be difficult in these situations. Thus the only apparent benefit that the MODWT really affords is that it does not require the number of probes on a chromosome to be a power of 2 (a requirement of the DWT).

Having transformed the data into wavelet domain, the denoising process is simply one of setting to zero all $\alpha_{j,t}$ that are close to zero according to some threshold $\lambda$. The new set of coefficients are termed $\hat{\alpha}$, and the denoised signal $\hat{y}$ can be constructed by $\hat{y} = W\hat{\alpha}$.

Hsu et al. (2005) present three test studies: a simulation study; an idealised data study based on a popular "gold standard" data set produced by Snijders et al. (2001) in which 15 fibroblast cell lines were hybridised to BAC arrays with 2276 probes spotted in triplicate; and data from an experiment in which 44 breast cancer tumours were hybridised to BAC arrays with 4762 probes (median spacing of 400 Kb) spotted in triplicate Loo et al. (2004). The authors find that denoising data gives greater power in subsequent statistical analyses than using raw data. However the issue of how to deal with small aberrations remains partly unresolved, so to explore this the authors employ three different methods for choosing $\lambda$. These result in denoised data with varying levels of smoothness. The authors find that using the technique that results in the least smooth denoised function is best for capturing small changes. Unfortunately, even the best thresholding method for small changes still misses some out. Importantly, another shortcoming of wavelet analysis is that it requires data points to be evenly spaced, and as such the authors choose to treat the aCGH data as though this is the case; however this is often an unreasonable assumption, and affects the applicability of this method to at least a subset of aCGH experiments.

## 2.5 Segmentation

Smoothing algorithms are useful for the visualisation or pre-processing of aCGH data prior to further analysis, but they do not address the primary task of *automatically* identifying regions of copy number variance. With this in mind many groups have developed model-based segmentation algorithms with the objective of, rather than smoothing the data, detecting the locations of copy number changes, or *breakpoints*, within the data. Depending on the model chosen, some algorithms also constrain the number of segments to avoid too fine a partition of the data.

As with the smoothing algorithms introduced previously, segmentation methods often view aCGH data as consisting of a series of piecewise constant segments delineated by abrupt jumps. However these methods additionally model the segments as a function of various parameters such as the number of breakpoints, their locations and the mean and variance of the distributions of each segment. Subsequently these methods maximise a function, typically a likelihood function (but not always), to estimate the model parameters from the data. Here three different model-based segmentation approaches are discussed: Circular binary segmentation, proposed by Olshen and Venkatraman (2004), that recursively uses the maximum of a likelihood ratio statistic to detect narrower segments of CNV; a genetic local search algorithm, presented by Jong et al. (2003), that is used to maximise a penalised likelihood function; and an adaptive penalised likelihood method given in Picard et al. (2005). Additionally, to highlight a different approach to the segmentation process, a method based on hierarchical clustering (Wang et al., 2005) which focuses on merging similar segments rather than detecting differences between them, is discussed.

### 2.5.1 Circular binary segmentation

Olshen and Venkatraman (2004) frame the task of breakpoint location in aCGH data as a change-point detection problem.

If $X_1, X_2, ...X_n$ is a sequence of random variables, then $\tau$ is a change-point if $X_1, ..., X_\tau$ share a distribution function $F_0$ and $X_{\tau+1}, ..., X_n$ share a different distribution function $F_1$.

Denoting $y_i$ to be the observed $\log_2$ ratio intensity value at the $i^{th}$ genomic location $x_i$, the additive measurement error model given in eq.(2.6) can be used to relate the true copy number at $x_i$, $f(x_i)$, and the observed signal $y_i$.

Considering the situation where $f(x_1), f(x_2), ..., f(x_\tau)$ are equal, and $f(x_{\tau+1}), ..., f(x_n)$ are equal but different to the $f(x_i)$ at $x_1, ..., x_\tau$, then the signals $y_i$ observed at $x_1, ..., x_\tau$ will come from one distribution function, and the signals at $x_{\tau+1}, ..., x_n$ will come from another. Hence the change-points will be the indices of the probes where the changes in copy number occur.

The *binary segmentation procedure* (Sen and Srivastava, 1975) was an early solution to the change-point detection problem. Let $y_1, ..., y_n$ be the indexed data set and let $S_i = y_1 + \cdots + y_i$, $1 \leq i \leq n$, be the partial sums. Also assume that the $y_i$s are normally distributed with a common known variance. For each $0 < i < n$ calculate the statistic $Z_i$ given by:

$$Z_i = \frac{S_i/i - (S_n - S_i)/(n-i)}{\sqrt{(}1/i + 1/(n-i))} \tag{2.8}$$

and the likelihood ratio statistic for testing the null hypothesis that there is no change against the alternative that there is exactly one change at some location $i$ is then given by:

$$Z_B = max_{1 \leq i < n} |Z_i| \tag{2.9}$$

If the statistic exceeds the upper $\alpha^{th}$ quantile of the null distribution of $Z_B$ the null hypothesis of no change is rejected and the location of the change-point is estimated to be the $i$ for which $Z_B = |Z_i|$. If the variance is unknown then an estimate of the variance derived from the data can be used instead, the $Z_i$ statistics are then replaced by their corresponding t-statistic, and $Z_B$ is replaced by the maximum of the absolute t-statistics.

Binary segmentation proceeds by applying the test to a segment to find a change-point within it, then recursively applying the test to the two resulting segments, until no more changes are detected in any of the segments obtained.

As motivation for a modification of binary segmentation, Olshen and Venkatraman (2004) highlight a problem with binary segmentation originally presented in Venkatraman (1992): that it cannot detect a small changed segment in the middle of a large segment. The problem occurs because the procedure only looks for one change-point at a time. The modification is termed *circular binary segmentation* (CBS) and, considering a segment to be spliced at two ends to form a circle, the likelihood ratio test statistic for testing the hypothesis that the arc from $i+1$ to $j$ and the arc from $j+1$ to $i-1$ have different means is given by:

$$Z_C = max_{1 \leq i < j \leq n} |Z_{ij}| \tag{2.10}$$

where the statistic $Z_{ij}$ is defined for $1 \leq i < j \leq n$ as:

$$Z_{ij} = \frac{(S_j - S_i)/(j-i) - (S_n - S_j + S_i)/(n-j+i)}{\sqrt{1/(j-i) + 1/(n-j+i)}} \tag{2.11}$$

As before, if the statistic exceeds the upper $\alpha^{th}$ quantile of the null distribution of $Z_C$s the null hypothesis of no change is rejected. The possible change-points found via $Z_C$ include those that result in three segments ($j < n$) as well as the single change-points that result in binary segmentation ($j = n$). Once again all change points are found by applying the procedure recursively.

Finally, also incorporated in CBS is a modification that if the data are non-normal then the reference distribution for $Z_C$ is estimated via a permutation method.

Olshen and Venkatraman (2004) have shown that CBS works well by testing the method using the data set from Snijders et al. (2001), an unpublished ROMA data set in which 23 cancer cell lines were hybridised to arrays containing 9820 $70-$mers, and a simulation study. However while the method seems to succeed in the automated segmentation of the aCGH data, it does not provide any such automation for the process of identifying which segments represent regions of significant CNV. In other words no methods for *classification* and subsequent significance ranking of the delineated segments are provided.

### 2.5.2 Genetic local search algorithm

Jong et al. (2003) assume that aCGH data are generated by a Gaussian process and consist of a sequence of piecewise constant segments with sharp changes between them. Furthermore they model the sequence of segments as a function of the locations of the segment boundaries and the mean and variance of the distributions for each segment. Then a likelihood function, penalised by the number of segments, is used to estimate the breakpoint locations. Finally a local search procedure embedded in a genetic algorithm is used to maximise the likelihood function.

More formally, once again denoting $y_i$ to be the observed $\log_2$ ratio intensity value at the $i^{th}$ genomic location $x_i$, the goal stipulated in Jong et al. (2003) is to group the $y_i$ into a small number, $K$, of segments $(y_1, ..., y_{z_1}), (y_{z_1+1}, ..., y_{z_2}), ..., (y_{z_{K-1}}, ..., y_n)$ such that the copy number of the probes in each segment are identical. The indices $z_0 = 0 < z_1 < \cdots < z_{K-1} < n = z_K$ are then the segment breakpoints, delimiting K segments. Additionally, the additive measurement error model given in eq.(2.6) is assumed, and thus the model stipulates that for each segment $k$, $z_{k-1} < i \leq z_k$, the observed signals $y_i$ can be considered as independent and drawn from a normal distribution with mean $\mu_k$ and variance $\sigma_k^2$ that are particular to the $k^{th}$ segment. This leads to a log-likelihood function that can be decomposed into a sum of local log-likelihoods, calculated on each of the segments:

$$L_K = \sum_{k=1}^{K} l_k \tag{2.12}$$

where

$$l_k = -\frac{1}{2} \sum_{i=z_{k-1}+1}^{z_k} \{ \log(2\pi.\sigma_k^2) + \left[ \frac{y_i - \mu_k}{\sigma_k} \right]^2 \} \tag{2.13}$$

and, using maximum likelihood, $\mu_k$ and $\sigma_k$ for the $k^{th}$ segment are estimated as:

$$\hat{\mu_k} = \frac{1}{z_k - z_{k-1}} \sum_{i=z_{k-1}+1}^{z_k} y_i \tag{2.14}$$

$$\hat{\sigma_k^2} = \frac{1}{z_k - z_{k-1}} \sum_{i=z_{k-1}+1}^{z_k} [y_i - \hat{\mu_k}]^2 \tag{2.15}$$

To find the breakpoints this log-likelihood needs to be maximised relative to $z_1, ...z_{K-1}$. However the maximum log-likelihood will be obtained with the largest possible number of breakpoints, so to find a more parsimonious model a penalty term of $\lambda(K-1)$ is subtracted to discourage a large number of segments. Thus the final penalised log-likelihood to be maximised is:

$$\tilde{L}_K = \sum_{k=1}^{K} l_k - \lambda(K-1) \tag{2.16}$$

Jong et al. (2003) report that in their experiments the choice $\lambda = 10$ was appropriate.

### Searching for the minimising set of breakpoints

Jong et al. (2003) have developed several search algorithms to find the maximising set of breakpoints for eq.(2.16). The first presented is a local search algorithm that uses the log-likelihood as a scoring function for any set of breakpoints $z_1, ..., z_{K-1}$. It takes as input the $y_i$ data for one chromosome, and $K-1$ randomly generated breakpoints that segment the data into K segments. At every iteration each breakpoint, selected in a random order, is moved either left or right, also selected randomly. If the move increases the log-likelihood then it is kept, otherwise a move in the other direction is tested, and only kept if it reduces the score. The algorithm finishes when moving each breakpoint does not improve the scoring. This local search algorithm is used in a multi-start local search algorithm, a simulated annealing multi-start local search, and two genetic algorithms, the most successful of which (as reported in the paper) is described next.

The genetic local search algorithm begins by generating an initial population of *individuals* that each represent a random segmentation of the data. The population is built by creating, for each $K$ in a fixed range, a number $m$ of individuals with that many segments. To complete the initialisation step, the local search algorithm is applied to each individual.

Until a termination criterion that disjunctively combines a maximum number of iterations with the fitness of the best individual is reached, the genetic algorithm proceeds as follows:

- randomly select two parents from the population
- generate two new offspring through a blind uniform crossover
- apply a mutation to each of the offspring. Either:
  - **remove**: remove the breakpoint whose removal gives the best score
  - **add**: find the $k^{th}$ segment for which $k = \text{maxarg}(\sigma_k^2)$ and place a breakpoint in the middle of the region
- apply the local search algorithm to each of the offspring
- replace the two worst individuals of the population with the offspring

To test their method the authors use a data set from the archives of the Department of Pathology at VU University Medical Centre. It is comprised of aCGH measurements for 9 gastric tumours from experiments carried out on BAC arrays with approximately 2275 probes, spotted in triplicate and spread along the genome at a spacing of $\sim 1.4$ Mb. They find that the genetic algorithms have good convergence behaviour and that their outcome is robust to the initialisation and other random operators used. Finally they post process the smoothings and breakpoints of the best genetic search algorithm by joining together smoothing levels that are 'close' to one another ("reflecting the observation that few copy number values are present in chromosomes"), and compare the results to a manual smoothing produced by an expert. The outcomes are comparable, but the genetic search algorithm is more susceptible to outliers than the manual smoothing.

Finally, a shortcoming of this method not discussed in the paper is that it too, like CBS, does not provide a mechanism by which to automatically classify and rank segments according to the significance of the evidence that they are copy number variant.

### 2.5.3 Adaptive penalised likelihood model to determine breakpoints

A more statistical approach to maximising the log-likelihood given in eq.(2.12) and eq.(2.13) has been presented in Picard et al. (2005). Rather than choosing an arbitrary penalty constant to lower the number of segments selected in a profile, the authors introduce a new procedure that chooses the penalty constant adaptively to the data. Furthermore they use dynamic programming to find the global maximum, in contrast to genetic local search algorithms that are not guaranteed a globally optimal result.

As explained previously, the log-likelihood will be maximal when each point is in its own segment, so a penalty against too many segments is required for a more parsimonious model. For a given number of segments, $K$, the maximisation of the log-likelihood, $\hat{L}_K$, gives the best segmentation with K segments. In general, a penalised version of the log-likelihood is then given by:

$$\tilde{L}_K = \hat{L}_K - \beta pen(K) \tag{2.17}$$

where $pen(K)$ is a penalty function that increases with the number of segments and $\beta$ is a multiplicative penalisation parameter.

In the more general context of model selection several choices for $pen(K)$ and $\beta$ have been presented, but the authors explain that criteria such as the Akaike Information Criterion (AIC, $\beta = 1$, $pen(K) = 2K$) and the Bayes Information Criterion (BIC, $\beta = \frac{1}{2}\log(n)$, $pen(K) = 2K$) are not suitable in the case of breakpoint detection because they overestimate the number of segments, and hence segment the data into regions that do not necessarily have any biological meaning. The authors also explain that an arbitrary $\beta$ and penalty function can be set so that segmentation is coarser, such as in Jong et al. (2003), but that such a penalty has no firm reasoning behind it.

Instead of applying penalties that are either not suited to the task of breakpoint detection, or have been picked on an ad hoc basis, the authors use the idea of choosing $\beta$ adaptively to the data (Lebarbier (2005) and Lavielle (2005)). Thus in this adaptive penalised likelihood model $\beta$ is defined to change with $\hat{L}_K$ for each segmentation size $K$. Additionally, also based on Lavielle (2005), the authors suggest the use of the penalty function $pen(K) = 2K$.

To calculate the $\beta$s adaptively to the data the $\hat{L}_K$ need to be calculated first. Thus a two step algorithm is suggested for maximising this penalised log-likelihood.

### A segmentation algorithm when the number of segments is known

When the number of segments, $K$, is known the problem of maximising the log-likelihood is simply to find the best partition into $K$ segments. Thus in this part of the algorithm $K$ takes a range of values, $K_1 = 1, ..., K_{max}$, and for each value of $K$ eq.(2.12) is maximised to find $\hat{L}_K$. To do this the segmentation problem is framed as a shortest path problem, and a dynamic programming solution is proposed.

The graph, through which the shortest path must be found, consists of the set of $n$ probes at genomic locations $x_i$, $1 \leq i \leq n$, with all possible segments represented as directed edges $(x_a, x_b)$ for all $x_a$ and $x_b$ such that $1 \leq x_a \leq n-1$ and $x_b > x_a$, and with the weights of the individual edges proportional to the negative log-likelihood:

$$W_1(x_a, x_b) = \sum_{i=x_a+1}^{x_b} \left\{ \log(2\pi.\sigma_k^2) + \left[\frac{y_i - \mu_k}{\sigma_k}\right]^2 \right\} \tag{2.18}$$

Due to the additivity of the log-likelihoods, the weight of a path of K edges from $x_a$ to $x_b$ is just the sum of the weights of the individual edges in the path:

$$W_K(x_a, x_b) = \sum_{k=1}^{K} W_1(start(k), end(k)) \tag{2.19}$$

Where *start* and *end* are functions that return the start node and end node of a directed edge respectively.

Finding the best segmentation of the aCGH data into K segments is then equivalent to finding the minimum weight K-step path, $\hat{W}_K$, through the graph from $x_1$ to $x_n$:

$$\hat{W}_K(x_1, x_n) = min_{x_h}\{\hat{W}_{K-1}(x_1, x_h) + W_1(x_h, x_n)\} \tag{2.20}$$

And this can be solved in $O(n^2)$ time (the algorithm only requires the storage of an upper $nxn$ matrix) using a standard dynamic programming solution to the shortest path problem. At the end of this procedure the quantities $\hat{W}_{K_1}(x_1, x_n), ..., \hat{W}_{K_{max}}(x_1, x_n)$ are stored and used in the second part of the algorithm.

### Estimating the multiplicative penalty parameter $\beta$

Since the maximum likelihood $\hat{L}_K$ measures the fit of the model with K segments to the the data, the aim is to choose a segmentation size $K$ for which $\hat{L}_K$ does not increase significantly. Thus the $\beta$s are defined as a decreasing sequence such that $\beta_0 = \inf$ and:

$$\forall i \geq 1 \beta_i = \hat{K}_{i+1} - \hat{L}_{K_i} \tag{2.21}$$

Thus on the curve $(K, \hat{L}_K)$, the sequence of $\beta_i$ are the slopes between the points $(K_{i+1}, \hat{L}_{K_{i+1}})$ and $(K_i, \hat{L}_{K_i})$, and the procedure to estimate the number of segments is just to calculate the second derivative of this curve:

$$\forall K \in K_1, ..., K_{max} D_K = \hat{L}_{K-1} - 2\hat{L}_K + \hat{L}_{K+1} \tag{2.22}$$

and then to select the highest number of segments $K$ such that the second derivative is lower than a given threshold. The choice of the threshold is arbitrary and the results of the procedure are dependent upon it. Nonetheless the authors explain that "despite this thresholding the procedure remains adaptive, since the penalty constant is estimated according to the data ".

The results given in Picard et al. (2005), based on tests on data from Snijders et al. (2001) and Nakao et al. (2004) show that, in comparison to BIC, AIC and Jong et al. (2003) this segmentation algorithm provides an improved method for estimating the number of segments of differing CNV in a given set of aCGH data. Although the discussion is not made in the paper, the same improvement is made in comparison to Olshen and Venkatraman (2004) because circular binary segmentation continues until all segments are found, and an ad hoc method for pruning the results is used afterwards. However this method still does not provide a method by which to assess the significance of highlighted regions of copy number change, and the user must still do this by eye.

## 2.5.4 Cluster along chromosomes

A different approach to segmenting aCGH data, based on hierarchical clustering along chromosomes (CLAC), is presented in Wang et al. (2005).

### Cluster formation

The standard agglomerative clustering algorithm is a bottom-up procedure that builds a binary tree representing similarities in the data. The algorithm proceeds as follows:

- Start with each data point in a singleton cluster. *These are the leaves of the hierarchical tree.*

- While there is more than one cluster:
  - Find the two closest clusters.
  - Merge the clusters into one. *On the hierarchical tree a new node is created with one branch for each cluster.*

When clustering aCGH data the data points are $\log_2$ ratio intensity values corresponding to probes that are ordered along the genome. Thus the order of the leaves on the hierarchical tree are fixed. Hence only adjacent clusters can be merged and the algorithm is reduced in complexity from $O(n^2)$ to $O(n)$.

The similarity of two clusters depends on the similarity of the $\log_2$ ratio intensity values in the two clusters. A statistic called *relative distance* (rd) is used to measure this similarity. Once again denoting $y_i$ to be the observed $\log_2$ ratio intensity at the $i^th$ genomic location $x_i$ the $rd$ for two contiguous probes is denoted as:

$$rd(y_i, y_{i+1}) = \frac{|y_i - y_{i+1}|}{|y_i| + |y_{i+1}| + |y_i + y_{i+1}|} \tag{2.23}$$

The denominator in eq.(2.23) gives an advantage to pairs of genes that have large absolute values whilst also sharing signs.

Then there are two possible definitions of the distance between two contiguous clusters of probes, $C_i = i_1, i_2, ..., i_k$ and $C_j = j_1, j_2, ..., j_k$ where $j_1 = i_k + 1$:

$$rd_{nearby}(C_i, C_j) = rd(y_{i_k}, y_{j_1}) \tag{2.24}$$

$$rd_{max}(C_i, C_j) = max\{rd(y_{i_t}, y_{j_s}) | i_t \epsilon C_i, j_s \epsilon C_j\} \tag{2.25}$$

### Cluster selection

After the tree is built the next step is to choose which clusters represent segments of probes that are CNV. Three properties are examined to identify such regions:

1. $rd$: The $rd$ of this node in the tree.
2. $size$: The number of probes in this cluster. This is transformed monotonically into $[0, 1]$ by defining $lsize_i = \frac{\log(size_i)}{max\{\log(size_i)\}}$.
3. $meanvalue$: The mean value of the probes in this cluster.

Two kinds of regions are selected. The first are characterised by a big spike, corresponding to small values for $lsize$ and very large values of $|meanvalue|$. The second kind are regions of consistent gain or loss, where ratios may not deviate from 0 very much, but tend to stay positive or negative in the whole region. These regions correspond to nodes with bigger $lsize$, smaller $rd$, and with not too small $|meanvalue|$.

To formalise these rules Wang et al. (2005) use output data from two cDNA aCGH experiments carried out under the same conditions: reference human vs reference human, and human XY vs human XX. The first experiment is used to provide an empirical joint distribution of $rd$ and $lsize$ when there is no CNV, while the second experiment gives an empirical joint distribution of the variables when most probes are not in CNV, but a known subset are in CNV (in this instance the subset of probes that are on chromosome X but not on chromosome Y). This data is from an unpublished larger study of lung cancer carried out by Young Kim and Jonathan Pollack in which 48 lung cancer cell lines were profiled on cDNA microarrays containing cDNAs representing 25736 genes, with an average spacing of 60 Kb.

The authors plot scatter plots of $(lsize, rd)$ and $(lsize, meanvalue)$ for the reference vs reference array and for the XY vs XX experiment. Selection rules for clusters of probes that represent regions of CNV are then chosen based on lines that segment these plots such that all clusters that cannot be CNV (because they are reference vs reference) are in one set, while those that are most likely CNV (they appear in the XY vs XX plots as a group of points distinct from those seen in the reference vs reference experiment), are in another. For any new data sets these segmenting lines, and hence the selection rules, are calibrated dependent upon noise levels in the data, and tuned to achieve desired false discovery rates (see next).

### Controlling the false discovery rate

In addition to the novel approach to segmentation presented in Wang et al. (2005), the false discovery rate (FDR, Benjamini and Hochberg (1995)) is used to provide quantitative statistics about the putative regions of CNV; a functionality not provided by any of the methods described so far.

| *Hypothesis* | *Accept* | *Reject* | *Total* |
|---|---|---|---|
| Null True | U | V | $m_0$ |
| Alternative true | T | S | $m_1$ |
| | W | R | $m$ |

Table 2.1: Outcomes when testing $m$ hypotheses

The possible outcomes when testing $m$ hypotheses are given in table 2.1. The FDR is defined in Benjamini and Hochberg (1995) as the expected proportion of rejected $m$ that are actually true. Referring to table 2.1 this is more formally stated as:

$$FDR = E\left(\frac{V}{R}.1_{\{R>0\}}\right) = E\left(\frac{V}{R}\middle| R > 0\right)P(R > 0) \tag{2.26}$$

Here the null hypothesis for each probe $x_i$ is that it does not belong to a region of CNV. $R$ is then the number of probes selected via the cluster selection process, and $V$ is the number of probes that are selected but are really from $H_0$. Using reference vs reference hybridisations produced under the same experimental conditions as the test vs reference experiments it is possible to estimate the FDR as:

$$\widehat{FDR} = \frac{\text{number of probes picked in the reference array (under the same criteria)}}{\text{number of probes picked in the test array}} \tag{2.27}$$

Parameters for the selection rules are chosen which make the $\widehat{FDR}$ first cross a certain level, say 1%. In this case, if there are $m$ probes selected with this calibration of the rules, then it is possible to say that more than $0.99m$ probes among the $m$ selected are truly significant.

Wang et al. (2005) first assess the performance of CLAC using the cDNA microarray lung cancer study mentioned previously. For a corresponding FDR of 0.009 the results are good; both localised amplifications and contiguous regions of potential gain or loss are delineated, and noise spikes do not cause the procedure to fail. However, because this method does not perform any smoothing prior to clustering, data sets with more variance than the lung cancer set may cause problems for the algorithm.

Finally the method is tested on the BAC aCGH data from Snijders et al. (2001). There is no reference vs reference array in this data set, so 'pseudo reference' arrays have been produced by removing measurements outside the 97% quantiles. The results compare well to the true known regions of CNV, but the FDR is overestimated due to the use of the 'pseudo reference' arrays.

### 2.5.5 CGH-Explorer

One other segmentation method, CGH-Explorer, presented by Lingjaerde et al. (2005), uses the positive false discovery rate (pFDR, Storey (2002)) to select interesting regions of CNV. The pFDR is a version of the FDR defined as $pFDR = E\big(\frac{V}{R}\big|R > 0\big)$. This conditions on the event that positive findings have occurred. pFDR is identically 1 when all null hypotheses are true, so to control the pFDR it has to be estimated for a particular rejection region.

Briefly, a binary classification of the probes is defined based on the signs of probe neighbours; a probe's classification is the sign of its neighbourhood mean (the mean of the probe's $\log_2$ ratio and the $\log_2$ ratios of its four neighbours (two on each side)), unless all of the four probe neighbours have corresponding neighbourhood means that are opposite in sign, in which case the probe's classification is the opposite of its neighbourhood mean sign. This binary classification performs implicit smoothing, (via use of the neighbourhood mean), and causes a partitioning of the genes into sets of consecutive probes. The partitioning is such that in each set all the probes have the same classification, and such that for any two neighbouring sets of probes the sets have probes of opposite classification. For each segment the pair $(L, H)$ is computed, where $L$ is the number of probes in the segment and $H$ is the absolute value of the average of the $\log_2$ ratios of the probes in the segment. The joint null distribution of $(L, H)$ is then found by Monte Carlo simulations (this is parametrised by the observed variance of the noise in the data). A rejection region is defined for a series of rejection region thresholds $\lambda$, and the proportion of probes belonging to segments that fall into the rejection region under the simulated null is calculated, $\alpha(\lambda)$, as are the number of observed probes whose segments fall into the rejection region, $S_\lambda$. Finally these values are used to calculate an estimate of the pFDR, $\widehat{pFDR}(\lambda)$, and the set of probes $S_\lambda$ whose $\lambda$ gives the desired $\widehat{pFDR}(\lambda)$ is returned.

CLAC and CGH-Explorer represent an improvement over previous smoothing and segmenting methods because they output regions of CNV that are likely to be statistically significant. Furthermore the FDR framework means that the end user can explore as many interesting regions of putative CNV as their risk levels for false discoveries allow. However although the data is available for ranking the highlighted regions, this is not implemented by either Wang et al. (2005) or Lingjaerde et al. (2005).

## 2.6 Thresholding

Although thresholding algorithms were among the first analytical methods for aCGH data they are discussed here, after smoothing and segmentation methods, because despite their simplicity they provide automatic classification of regions of CNV, and have the potential to provide quantitative statistics about the highlighted regions.

Thresholding methods establish $\log_2$ ratio thresholds that can be used to classify probes or regions of probes as CNV. For example Pollack (2002) developed the following method for an experiment in which 44 breast tumours and 10 breast cancer cell lines were profiled for CNV on cDNA arrays containing probes mapped to 6691 human genes:

1. Smooth $\log_2$ ratios using a moving average window.
2. For $i = 1, ..., n$ ($n$ is the number of probes), using a 'reference vs reference' experiment, find a window size $k = \hat{k}(i)$ for the $i^{th}$ probe such that it gives the highest positive probe neighbourhood mean $val_{0\_high}(i)$, $k = 1, 3, 5, ..., n/2$. Similarly and a window is found that gives the lowest negative neighbourhood mean $val_{0\_low}(i)$, $k = 1, 3, 5, ..., n/2$.
3. Find upper and lower thresholds for $val_{0\_high}(i)$ and $val_{0\_low}(i)$ respectively, so that the overall proportion of false positives in the reference sample is $\alpha/2$ in each of the upper and lower tails of the empirical null distribution.
4. For the test data , and once again for $i = 1, ..., n$, find a window size $k = \hat{k}(i)$ for the $i^{th}$ probe such that it gives the highest(lowest) positive(negative) probe neighbourhood mean $val_{high}(i)$ ($val_{low}(i)$).
5. All probes for which $val_{high}(i)$ ($val_{low}(i)$) exceeds the upper (lower) threshold are marked as significant and given the corresponding classification. If the number of probes is $n$ and the number of probes called significant is $s$, then the estimated FDR is $\alpha n/s$. $\alpha$ is chosen to give the desired FDR.

As discussed above, classification and the FDR are useful tools in the process of choosing putative regions of CNV for further exploration. However there are some problems with the thresholding method described in Pollack (2002). First, there is no motivation given for the smoothing of the $\log_2$ ratios, nor is there one given for the summary statistics $val_{high}(i)$ and $val_{low}(i)$ calculated for the $i^{th}$ probe. Second, because the summary statistic is calculated for individual probes rather than for regions of probes, it is possible to rank the probes according to their likelihood of being implicated in a region of CNV, but there is no analogous method provided for ranking the regions themselves. Therefore this method has the potential to be very useful for end users, but some statistical grounding and development is required before the potential can be fulfilled.

As an alternative to using a control experiment, thresholds can be set using mixture models. Mixture models assume that the $\log_2$ ratios are independent samples from an underlying distribution that consists of multiple components, with each component corresponding to a different type of CNV. Algorithms such as the EM algorithm are used to estimate, from the data, the parameters of each component of the model (for example in a Gaussian mixture model the mean and variance of each component would be estimated). Finally the estimated parameters are used as thresholds, on the $\log_2$ ratios, with which to classify probes. One such method has been presented in Hodgson et al. (2001) where a mixture of three Gaussian distributions was fitted to aCGH data, (from an experiment using 85 BAC arrays each with $\sim 380$ probes), using a maximum likelihood method.

This method is simple and has been shown to be effective. However it requires the user to make rather subjective and ad hoc decisions regarding good thresholds for the different types of CNV and is therefore susceptible to error. Furthermore, although the mixture model method could provide significance rankings for the regions found, Hodgson et al. (2001) have not implemented this. These shortcomings render this method far less useful than its potential allows.

## 2.7 Hidden Markov models

None of the smoothing and segmentation methods presented so far have an explicit, and biologically meaningful, underlying model of the possible types of CNV. CGH-Explorer and the thresholding methods provide a limited model which explicitly assumes that positive/high (negative/low) $\log_2$ ratios are associated with gain (loss) in copy number, but nothing is said of the relationship between states of CNV, or of the relationship between probes with the same CNV. Hidden Markov Models (HMMs) provide a probabilistic framework in which to describe a CNV state dependency structure and to relate observed signals to it. This framework affords several improvements over the previous methods.

Without a state dependency structure the simplifying assumption in the methods discussed is almost always that the observed signal intensities are independent of one another conditional on an underlying, but unspecified, state of CNV. Although this is reasonable when the underlying probes are probes, short oligonucleotides that are much closer to one another are unlikely to act independently. Furthermore it is useful to incorporate the dependence of signals from neighbouring probes, via explicit relationships between their underlying states of CNV, in the segmentation process. HMMs account for the inherent dependencies between neighbouring probes by probabilistically relating their hidden true states of CNV.

Additionally the previous methods do not combine data from probes that are in the same class of region but are far apart on the genome. Conversely, if a state dependency is made explicit, information can be borrowed globally across the data because each probe, regardless of its physical location, will contribute information to one of the explicit states of CNV. In this way strengths are borrowed across the genome when an HMM is fitted to data.

Furthermore, in the smoothing and segmentation methods all classifications of the delineated regions are carried out in an ad hoc fashion after the segmentation process. However the problems of segmentation and classification are intertwined, therefore an improved approach would solve the problems jointly. HMMs provide an explicit underlying state dependency structure; so fitting the model automatically generates classifications for, and hence segmentation of, the data.

Finally, many of the methods discussed so far do not estimate the statistical significance of the detected copy number changes, and among those that do (Wang et al. (2005), Lingjaerde et al. (2005), Pollack (2002), Hodgson et al. (2001)), implementations are either somewhat insufficient or ad hoc. HMMs provide a thorough statistical framework for detecting CNVs and enable the detection of such regions based on statistical significance.

In summary, these attributes make the use of HMMs in aCGH data analysis an important advancement in the field.

### 2.7.1 Elements of an HMM

The following description of an HMM is presented here primarily for reference, and is based on the explanation given in Rabiner (1989).

A discrete time HMM with continuous output is characterised by the following:

- $N$, the number of states in the model. The states are hidden but for many practical applications there is some real-world significance attached to them. In the case of aCGH data analysis the states represent various types of CNV. The individual states are denoted as $S = \{S_1, S_2, \cdots, S_N\}$, and the state at time(position or point) $i$ as $q_i$, where $1 \leq i \leq n$ and $n$ is the number of points (in the case of aCGH data $n$ is the number of probes).
- The state transition probability distribution $A = \{a_{jk}\}$ where, $\forall i$,

  $$a_{jk} = P[q_{i+1} = S_k | q_i = S_j], 1 \leq j, k \leq N \tag{2.28}$$

  This transition matrix is the explicit representation of the relationship between states of CNV.
- The observation probability density in state $j$, the most general representation of which is a finite mixture of the form

  $$b_j(\mathbf{O}) = \sum_{m=1}^{M} c_{jm} \Pi[\mathbf{O}, \mu_{jm}, \mathbf{U}_{jm}], 1 \leq j \leq N \tag{2.29}$$

  where $\mathbf{O}$ is the vector being modelled, $c_{jm}$ is the mixture coefficient for the $m^{th}$ mixture in state $j$ and $\Pi$ is any log-concave or elliptically symmetric density (eg Gaussian), with mean vector $\mu_{jm}$ and covariance matrix $\mathbf{U}_{jm}$ for the $m^{th}$ mixture component in state $j$. Usually a Gaussian density is used for $\Pi$, as is the case in the methods presented next. $B$ is then the vector $b_j(\mathbf{O})$, and relates the underlying states of the HMM to the observed data. In the case of aCGH data analysis the observation probability density relates the true CNV of a probe to the $\log_2$ ratio observed for that probe.
- The initial state distribution $\pi = \{\pi_j\}$ where

  $$\pi_j = P[q_1 = S_j], 1 \leq j \leq N \tag{2.30}$$

$\lambda$ is used to compactly note the parameters $A$, $B$ and $\pi$ of an HMM.

### 2.7.2 Choosing an optimal state sequence

When using HMMs to segment and classify aCGH data, the primary task is to find the underlying sequence of CNV states. Thus a brief explanation of a method for choosing the optimal state sequence, based on Rabiner (1989) is given here.

It is difficult to choose an appropriate definition of optimality for a state sequence. The criterion used by all of the HMM methods described in the following sections is to choose the states $q_i$ which are *marginally* most likely. These are calculated as follows.

Define the forward variable $\alpha_i(j)$ as:

$$\alpha_i(j) = P(O_1 O_2 \cdots O_i, q_i = S_j | \lambda) \tag{2.31}$$

that is, the probability of the partial observation sequence, $O_1 O_2 \cdots O_i$ until point $i$, and state $S_j$ at point $i$, given the model $\lambda$.

The backward variable $\beta_i(j)$ is defined as:

$$\beta_i(j) = P(O_{i+1}O_{i+2}\cdots O_n|q_i = S_j, \lambda) \tag{2.32}$$

that is the probability of the partial sequence from $i+1$ to the end, given state $S_j$ at point $i$ and the model $\lambda$.

The forward and backward variables are then solved inductively using the $Forward\text{–}Backward\ Procedure$. Solve for $\alpha_i(j)$:

1. Initialisation:
$$\alpha_1(j) = \pi_j b_j(O_1), 1 \leq j \leq N \tag{2.33}$$

2. Induction:
$$\alpha_{i+1}(k) = \big[\sum_{j=1}^{N} \alpha_i(j)a_{jk}\big]b_k(O_{i+1}), 1 \leq i \leq n-1, 1 \leq k \leq N \tag{2.34}$$

Solve for $\beta_i(j)$:

1. Initialisation:
$$\beta_I(j) = 1, 1 \leq j \leq N \tag{2.35}$$

2. Induction:
$$\beta_i(j) = \sum_{k=1}^{N} a_{jk}b_k(O_{i+1})\beta_{i+1}(k) \tag{2.36}$$

Finally define the variable:

$$\gamma_i(j) = P(q_i = S_j|O, \lambda) \tag{2.37}$$

that is, the probability of being in state $S_j$ at point $i$, given the observation sequence $O$. This can be expressed in terms of the forward and backward variables:

$$\gamma_i(j) = \frac{\alpha_i(j)\beta_i(j)}{P(O|\lambda)} = \frac{\alpha_i(j)\beta_i(j)}{\sum_{j=1}^{N} \alpha_i(j)\beta_i(j)} \tag{2.38}$$

Using $\gamma_i(j)$ it is then possible to solve for the most likely state $q_i$ at point $i$ as,

$$q_i = argmax_{1 \leq j \leq N}[\gamma_i(j)], 1 \leq i \leq n \tag{2.39}$$

This method maximises the expected number of correct states but it does so without regard to the probability of the occurrence of sequences of states. An alternative optimality criterion is to find the single best state sequence path, that is to maximise $P(Q|O, \lambda)$. A dynamic programming technique for finding the best state sequence is the $Viterbi$ algorithm, also explained in Rabiner (1989). However, this will not be discussed here because none of the current HMM methods for aCGH data use this criterion. At the end of this section the ramifications for aCGH data analysis of maximising eq.(2.39), rather than using the Viterbi algorithm, will be discussed.

### 2.7.3 Initial application of HMMs to aCGH data analysis

The first application of HMMs to BAC aCGH data analysis was presented in Fridlyand et al. (2004). The five possible CNV states that probes could be classified under were specified accordingly:

1. **Focal aberrations** are localised regions (one or two probes) of altered copy number. These are sub-classified as:
   - **Low level** gains or losses
   - **High level focal amplifications**
   - **Outliers**. Note that these do not represent a CNV type but rather "an auxiliary quantity used in finding amplifications and detecting array problems"
2. **Transition points** are inter-probe spaces that border two large regions associated with different copy number states.
3. **Whole chromosomal changes** occur when an entire chromosome is gained or lost.

The process for fitting the HMM and classifying probes is carried out by two algorithms.

#### Segmenting probes into sets with the same underlying copy number

For each chromosome, the following steps are carried out:

1. For each $K = 1 ... K_{max}$ an HMM of size $K$ is fitted to the data. $\pi$ is initialised by assigning the majority of probability to the "normal" state, and the remaining probability uniformly among all other states. The transition matrix $A$ is initialised with a high probability assigned to staying in the same state and low non-zero probabilities assigned to the transitions between states. $B$ is initialised by segmenting the normalised signals in to $K$ states using partitioning among medoids (Kaufman and Rousseeuw, 1990) and then, for each state, using the median of the normalised signals in the state as an estimate for the mean of the state, and similarly estimating the variance of the state. Next the EM algorithm is used to maximise $Lik(\lambda|\mathbf{O})$: in the estimation step the Forward-Backward algorithm is employed to solve equation 2.39, and to identify an optimal state sequence such that at each $t$ the most likely $q_t$ is chosen; in the maximisation step the parameters $\lambda$ are estimated, based in the optimal state sequence, to increase $Lik(\lambda|\mathbf{O})$. Once the maximum likelihood estimates for $\lambda$ are obtained, they can be used in the Forward-Backward algorithm to find the maximum marginal state occupancy. Finally, a penalised negative log-likelihood (penalising for an increased complexity or number of states) is calculated for the $K$-state HMM.
2. Choose the model that minimises the negative log-likelihood.
3. Merge states with close medians until either there is only one state left, or all the states have medians that are separated from all other states by at least some threshold $d$. (Note that this step is problematic because it undermines any meaning assigned to the original states.)

#### Assigning CNV types to genomic regions and individual probes

Having segmented the probes on each chromosome the probes are classified as follows:

First the sample standard deviation is estimated: Compute the median absolute deviation (MAD) of the probes in the states containing at least 20 probes located on chromosomes partitioned into no more than 3 states. The standard deviation is estimated as median of the MADs for all such states.

Finally probes are classified, according to the scheme listed above, depending on the distance of their $\log_2$ ratios from the median of their allocated state, and also on the states of neighbouring probes. Transition points are placed between two regions whose state differs, and whole chromosomal changes are identified when three heuristic rules pertaining to quantity, mean and median of $\log_2$ ratio values are obeyed.

This method, tested on the data from Snijders et al. (2001), affords some improvement over previous methods because it provides automatic classification of probes and segments. However the power of HMMs is not employed because the states of the HMM have no intrinsic meaning. Instead the states represent a discrete number of mean levels and the HMM is employed as an elaborate model fitting step in what transpires to be a segmentation method, penalised for complexity, followed by a merging step. Consequently post-processing is required to determine labels for the segments, and therefore segmentation and classification are still solved separately. Furthermore, there is still no significance associated with the highlighted regions of CNV.

### 2.7.4 Bayesian HMM for aCGH data analysis

An HMM method that addresses the problems of Fridlyand et al. (2004) is presented in Guha et al. (2006). There the authors propose a Bayesian 4-state HMM, in which informative priors, based on knowledge of aCGH data, are assumed for all unknown parameters. Copy number variations are identified using posterior probabilities. Since the posterior distribution cannot be solved analytically, MCMC is used for simulation based inference.

#### Likelihood function

In Guha et al. (2006) it is argued that chromosomes have differing propensities for CNV, and therefore each one should be fitted with its own HMM, and thus has a distinct set of parameters. For a given chromosome the ordered genomic locations are denoted by $i, 1 \leq i \leq n$, and the probes at those positions by $x_i$. Then $O_i$ denotes the normalised $\log_2$ ratio for the probe $x_i$ at position $i$.

For each probe there is a true hidden *copy number state*, $q_i$, which can take a value in the set $S = S_1, S_2, S_3, S_4$. $q_i = S_1$ represents a loss at $x_i$, $q_i = S_2$ represents no change, $q_i = S_3$ denotes a single copy gain and $q_i = S_4$ represents a multiple copy gain. The states $q_1, ..., q_n$ denote the copy number changes along a chromosome. (Note that this definition is not symmetrical for gain and loss because of the $\log_2$ ratios that were originally observed in BAC aCGH.)

For $j = 1, ..., 4$, $\mu_{S_j}$ is defined as the expected $\log_2$ ratio of all probes $x_i$ for which $q_i = S_j$. The $\mu_{S_j}$'s are unknown, but due to their biological interpretations they can be ordered: $\mu_{S_1} < \mu_{S_2} < \mu_{S_3} < \mu_{S_4}$. The normalised observed $\log_2$ ratios are assumed to be independently distributed, conditional on copy number, as $O_i\ N(\mu_{q_i}, \sigma^2_{q_i})$, where $1 \leq i \leq n$.

Finally the elements of transition matrix $A$ are assumed to be strictly positive, and the matrix is assumed to have a unique stationary distribution, denoted by $\pi_A = (\pi_A(1), \pi_A(2), \pi_A(3), \pi_A(4))$, where $\pi_A(j)$ is strictly positive for state $S_j, j = 1, ..., 4$. Then the initial state distribution $\pi = \pi_A$.

The chromosome specific hyper-parameters are $A$, means $\{\mu_{S_1}, \mu_{s_2}, \mu_{s_3}, \mu_{S_4}\}$ and error variances $\{\sigma^2_{S_1}, \sigma^2_{S_2}, \sigma^2_{S_3}, \sigma^2_{S_4}\}$.

### Priors

Let $X\ F.I(c < X < d)$ denote that $X$ has distribution $F$ on the interval $(c, d)$, with the density rescaled to make it a random variable. Then the means, $\mu_{S_j}$ are assumed to have the following priors:

- $\mu_{S_1}\ N(-1, \tau^2_{S_1}).I(\mu_{S_1} < \epsilon)$, where $\epsilon > 0$ and determines the boundaries for the $\mu_{S_j}$
- $\mu_{S_2}\ N(0, \tau^2_{S_2}).I(-\epsilon < \mu_{S_2} < \epsilon)$
- $\mu_{S_3}\ N(0.58, \tau^2_{S_3}).I(\epsilon < \mu_{S_3} < 0.58)$
- $[\mu_{S_4}|\mu_{S_3}, \sigma_{S_3}]\ N(1, \tau^2_{S_4}).I(\epsilon < \mu_{S_4} > \mu_{S_3} + 3\sigma_{S_3})$

For a discussion of the choice of the priors, which are generally based on the theoretical signals obtained from pure samples, and also for an analysis of a robust range of choices for the $\tau$s and $\epsilon$, the reader is referred to Guha et al. (2006).

Priors for the measurement errors are assumed to be $\sigma^{-2}_{S_j}\ gamma(1,1).I(\sigma^{-2}_{S_j} > 6)$ for $j = 1, 2, 3$ and $\sigma^{-2}_{S_4}\ gamma(1,1)$. Finally, with $\mathbf{a}_i$ denoting the $i^{th}$ row of $A$, it is assumed that the $\mathbf{a}_i$ are independently distributed with $\mathbf{a}_i \sim Dirichlet_4(\theta_{S_i,S_1}, \theta_{S_i,S_2}, \theta_{S_i,S_3}, \theta_{S_i,S_4})$, where $i = 1, ..., 4$ and the $\theta$s are positive. Again, a discussion of these choices is given in Guha et al. (2006).

### Posterior inference

The posterior distribution cannot be solved analytically so MCMC is used for simulation based inference. HMM parameters are iteratively sampled in blocks, along with state sequences that are generated by a stochastic version of the Forward-Backward algorithm. For each iteration of the MCMC there is a Bernoulli variable $Z_{ij}$ for each probe and each type of CNV. For a given probe $x_i$, and a given state $S_j$, $q_i = S_j$ $(Z_{ij} = 1)$for some MCMC draws and $q_i \neq S_j$ $(Z_{ij} = 0)$ for the remaining iterations. The probability that $Z_{ij} = 1$ is the posterior probability that probe $q_i = S_j$, and for a large enough sample of MCMC outcomes, the average of the $Z_{ij}$ is a simulation-consistent estimate of the posterior probability. Finally the Bayes decision rule corresponding to a 0-1 loss function is used to declare $q_i = S_j$ if the estimated posterior probability is greater than 0.5. If all $n$ probes on a chromosome have a common Bernoulli outcome, and the simulation consistent posterior probability of a chromosome-wide alteration is greater than 0.5, then a whole chromosomal change is declared.

Having assigned an underlying state of CNV, $q_i$, for all $x_i$, a classification scheme closely modelled on the scheme listed in Fridlyand et al. (2004) is used to give further biological interpretation to the four basic states of CNV, including sub-labelling of some focal aberrations as outliers. Finally transition points, also defined in the original HMM paper, are detected by finding a simulation consistent estimate of the set of change-points that have the highest *joint* posterior probability.

### Modifications to the Bayesian HMM

Shah et al. (2006) have made two modifications to the Bayesian HMM introduced above. Briefly the HMM is extended such that the observation density is a mixture of two Gaussians, one representing probes that belong to one of the four states listed previously (inliers), and the other representing outliers. The outlier distribution is modelled as Gaussian with $\mu_0$ and $\sigma_0^2$. An indicator variable is then used to act as a “switching parent” variable for each probe $x_i$, which selects between the outlier parameters and the inlier parameters. The indicator variables are modelled as conditionally independent, so there are no Markovian dynamics on the outliers. This means that the model can make temporary “excursions” to the outlier state, without incurring any “penalty” that would be caused through the use of a state transition matrix.

The second modification in Shah et al. (2006) is that instead of assuming that chromosomes have differing propensities for CNV, the opposite argument is made and the posterior distributions of the parameters $A, \mu, \sigma$ are expected to be consistent across chromosomes. Thus these parameters can be estimated using pooled data across all the chromosomes in the sample, which is postulated to be advantageous since the estimates are then guided by more data. Only the sampling of the states is estimated individually for each chromosome (because there is no real-world interpretation for the dependency of a probe at the end of one chromosome and the probe at the start of another chromosome).

The primary output of both types of Bayesian HMM is a CNV state, $q_i$, for each probe $x_i$ that has maximum marginal likelihood. Further biological classifications are made based on the scheme listed in Fridlyand et al. (2004), and transition points that border two large regions associated with different CNV states are also highlighted. The Bayesian HMM in Guha et al. (2006) is tested on three data sets: the “gold standard” data set in Snijders et al. (2001); a data set from Aguirre et al. (2004) from an experiment in which 24 pancreatic adenocarcinoma cell lines and 13 primary tumour specimens were hybridised to cDNA arrays comprising 14160 cDNAs; and a final set from Bredel et al. (2005) from an experiment in which 26 samples representing primary glioblastoma multiforme were hybridised to cDNA arrays with 41421 probes. The method is reported to perform successfully, and to have a favourable comparison to the CBS and adaptive penalised likelihood models discussed above. Shah et al. (2006) run their method on data from an experiment reported by de Leeuw et al. (2004), in which 8 mantle cell lymphoma cell lines have been hybridised to Sub Megabase Resolution Tiling arrays comprising 32000 probes. The tests suggest that pooling data and integrating knowledge of outliers into the HMM framework increase the accuracy of results. Therefore both Bayesian HMMs represent significant improvements to the field of aCGH data analysis because they perform segmentation and classification jointly, and because they provide a probabilistic framework in which to assess results.

However there are shortcomings of HMMs. First, there are always many parameters to set. This can either be done using expert knowledge or, as is the case with both the Bayesian HMMs, parameters can be set using prior distributions. The shortcoming of the former approach is that if parameters are not set by an expert, or if indeed there is no knowledge regarding appropriate parameter values, then parameters can lose meaning and results become impossible to interpret. The problem with the latter approach is that it is computationally intensive, if MCMC simulation is employed, and with the increase in resolution of aCGH techniques, this could become significant.

A second problem with the HMMs presented here is the chosen criteria for optimality of a state sequence. All three methods choose the states, $q_i$ that are marginally most likely using the Forward-Backward algorithm. It can be argued that in the case of aCGH data analysis, especially as array resolution increases, the point of interest is not the most likely state of CNV for each probe, but rather the most probable pattern of CNVs in the data as a whole. With this in mind it seems more reasonable to instead employ the Viterbi algorithm to find the single best state sequence.

Last, the output probabilities from the Bayesian HMMs, given for the classification of individual probes, are not the probabilities of interest in the task of ranking highlighted segments of CNV within the aCGH data. Although switching to the Viterbi algorithm will give a probability for the most likely single sequence of CNVs, it will still be necessary to develop methods that provide probabilities and rankings for segments within the global pattern of CNVs.

## 2.8 Smith-Waterman algorithm adapted for aCGH

With the exception of CBS, which has been used to analyse a very high resolution aCGH data set consisting of 388352 probes (Graubert et al., 2007), none of the methods described so far have been tested on a data set larger than the 41421 probe experiment used in a review of aCGH data analysis algorithms conducted by Lai et al. (2005). By analysis it is clear that several of these methods, including Eilers and de Menezes (2004), Picard et al. (2005), Guha et al. (2006) and Shah et al. (2006), either will not scale to, or are too computationally intensive for, aCGH data of the order of $10^5$ probes. Most of the methods discussed make the assumption that the $\log_2$ ratios are independently and normally distributed conditional on copy number. However, referring to figures 2.2 and 3.3 it is clear that with some high resolution data sets this may no longer be a reasonable assumption. Methods that do not assume this are those presented by Olshen and Venkatraman (2002), Pollack (2002), Wang et al. (2005) and Lingjaerde et al. (2005). The last three methods are also those that provide control of the FDR, but the first two require, ideally, a reference vs. reference experiment to do this, and none of them provide a significance ranking for the highlighted regions of CNV.

The Smith-Waterman algorithm adapted for aCGH data (SW-ARRAY), presented by Price et al. (2005) is the only method that offers both a nonparametric segmentation procedure and a nonparametric test of significance.

### 2.8.1 Motivation

In bioinformatics the Smith-Waterman algorithm was originally applied to the problems of DNA and protein sequence local alignment (Smith and Waterman, 1981), and for the identification of protein sequence segments with unusual properties (Karlin and Altschul, 1990). Array CGH data, when analysed in genome order, can be considered as a one dimensional series of continuously distributed scores, in which sub-sequences composed primarily of high values may indicate regions of gain in copy number, and those composed largely of low values might be due to loss in copy number. Thus the problem of finding regions of CNV reduces to that of identifying sub sequences with unusual properties, and the work of Karlin and Altschul (1990) is readily applicable.

### 2.8.2 Statistics of maximal segment scores

Once again denoting $y_i$ to be the observed $\log_2$ ratio at the $i^{th}$ genomic location $x_i$, for $i = 1, ..., n$, each $y_i$ can be considered as a score from a continuous distribution. Two assumptions are made about the data. First that at least one of the $y_i$ is positive, and second that either $\mathrm{E}(y_i) < 0$, or the scores can be transformed so that this is the case.

The behaviours of sums of consecutive $\log_2$ ratios is as follows. Starting at $y_i$ define the partial sums as:

$$S_{ik} = \sum_{r=i}^{r=i+k} y_r, \tag{2.40}$$

that is, $S_{ik}$ is the accumulated score of $k$ consecutive probes, starting at probe $x_i$. As $k$ increases the $S_{ik}$ are a random walk with negative drift. When looking for regions of increased CNV the goal is to find the maximum attained by this walk, and hence to find the corresponding sequence of $y_i$ that has the greatest additive score. (To find regions of decreased CNV the signs of the data must first be changed, and then the goal remains the same). This is the *maximal segment* and its score is the *maximal segment score.* The maximal segment score is also a local maximum; you can neither shrink nor expand the segment without reducing the score.

The basic result, stated in Karlin and Altschul (1990) (see also Mott and Tribe (1999)), is as follows. Denote the *maximal segment score* for a sequence of length $n$, as $S(n)$. Then the mean of the distribution of $S(n)$ is of the order $\ln n/\lambda$, where $\lambda$ is the *unique positive* solution to the equation

$$\int f(y) exp\{\lambda y\} = 1, \tag{2.41}$$

and $f(y)$ is the density of $\log_2$ ratios.

Then the random variable $\bar{S}(n) = S(n) - (\ln n)/\lambda$ has the close approximating extreme-value Gumbel distribution

$$Prob \bar{S}(n) > s \approx 1 - \exp\{-K * \exp\{-\lambda x\}\}, \tag{2.42}$$

where $K$ is related to the mean of the distribution by $\mu = \frac{\gamma + \log K}{\lambda}$, where $\gamma$ is Euler's constant (0.577).

This result will be important in future work when the method will be extended to incorporate SNP data (see chapter 5), but for the moment the focus is on the algorithm employed by Price et al. (2005) to identify all the locally high scoring segments in the aCGH data.

### 2.8.3 Composition of the maximal segment

An important consequence of searching for maximal segments is that, given an appropriate choice of scoring scheme, the maximal segment will correspond to the most statistically significant segment in terms of the type of segment of interest. It is easier to understand this by way of an example (presented in Karlin and Altschul (1990)):

As explained previously, one of the first applications of the Smith-Waterman algorithm was to search for regions of a protein characterised by an unusual amino acid composition. In this case, by definition, the frequencies of amino acids in the regions of interest will be different to the frequencies of the amino acids elsewhere in the protein, so a scoring scheme based on these relative frequencies will be best for distinguishing the regions. In particular Karlin and Altschul (1990) show that the optimal score for an amino acid $a$ is the log-likelihood ratio $\log(u/v)$ where $u$ is the frequency with which $a$ appears in the regions of interest and $v$ is the frequency with which it appears in the rest of the protein. In this example, the maximal scoring segment is the maximal sum of log-likelihood ratios, and is hence directly interpretable as the segment most likely to be a region of the type characterised by the amino acid composition of interest.

In terms of aCGH data the scores currently proposed, (the $\log_2$ ratio intensity values), do not have a comparable intrinsic meaning, and hence nor do they imply such a meaning for maximal scoring segments. However it is worth noting that if such a scoring scheme could be devised, then the use of the Smith-Waterman algorithm would automatically confer a statistical meaning on the maximal scoring segment.

### 2.8.4 Algorithm

The Smith-Waterman algorithm is an efficient way to find all locally high scoring segments in the data. This is the one dimensional adaptation for aCGH data, SW-ARRAY, presented in Price et al. (2005):

1. Subtract a threshold $t_0$ from the $\log_2$ ratios so that the mean of the adjusted scores is negative.
2. Let $\check{y}_i$ be the adjusted score for the $i^{th}$ probe $x_i$. The score of the segment from $p$ to $q$ inclusive is then defined as
$$T(p,q) = \sum_{i=p}^{q} \check{y}_i. \tag{2.43}$$
3. $S(q)$ then denotes the adjusted score of the segment ending at coordinate $q$, and $B(q)$ is the coordinate of the beginning of the segment that ends at $q$. The following recursion finds the high scoring segments:

Recalling that the probes are indexed by $i$, where $i = 1, ..., n$, set $S(0) = 0$, $B(0) = 1$, and for $q > 0$:

$$S(q) = \begin{cases} S(q-1) + \check{y}_q & \text{if } S(q-1) + \check{y}_q > 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.44}$$

$$B(q) = \begin{cases} B(q-1) & \text{if } S(q) > 0 \\ q & \text{otherwise} \end{cases} \tag{2.45}$$

Finally the boundaries $\{B(q_{max}), q_{max}\}$ and score $S(q_{max})$ of the maximal segment are returned. To identify all high scoring segments the maximal segment is replaced by zeroes and the algorithm is repeated until no positive scoring segments are found.

### 2.8.5 Permutation test

To estimate the statistical significance of a high scoring segment the adjusted aCGH data are permuted many times (say, 1000), and SW-ARRAY is performed on each of the permuted data sets. The proportion of times that the maximal segment score in the permuted data is larger than that of the high scoring segment is then a permuted $p$-value for the high scoring segment. This method is “based on the premise that successive scores from the permuted data approximate the null distribution of scores” (Price et al., 2005).

### 2.8.6 Application to aCGH data

In Price et al. (2005) an experiment is presented in which a set of DNA samples from patients with accurately mapped known monosomies on the terminal 2Mb region of chromosome 16p are hybridised to DOP-PCR amplified BAC and PAC arrays of the region. SW-ARRAY is tested on the data set and is found to perform very accurately and to be relatively insensitive to the choice of $t_0$.

In terms of performance on a very high throughput data set, Komura et al. (2006) use a modification of SW-ARRAY as part of their analysis of a 500K EA (Early Access) array. These arrays are a pre-commercial version of the GeneChip Human Mapping 500K Array set which contains 534500 SNPs on two genotyping arrays. Each array has unique outliers and the merged error distribution is therefore known to be non-Gaussian, so SW-ARRAY is ideal for this situation because it is nonparametric. Furthermore because the SW-ARRAY algorithm is $\mathcal{O}(nv)$, where $v$ is the number of permutations required for the significance test, it also scales well to data sets of this magnitude.

## 2.9 Summary

Table 2.2 summarises the aCGH analysis methods described in this chapter. Smoothing methods are useful for visualisation and pre-processing prior to segmentation, but they do not automatically segment aCGH data. Segmentation algorithms address this problem but, in general, do not classify or rank putative regions of CNV. CLAC and CGH-Explorer represent some improvement over most segmentation algorithms because they provide control over the FDR, and CGH-Explorer also provides a binary classification of regions. It should be noted that some post processing segment merging schemes have been proposed by Hupe et al. (2004) and Willenbrock and Fridlyand (2005), and that these are intended as precursors for a range of heuristic classification algorithms that have been developed for the classification of putative CNV segments located by segmentation algorithms.

HMMs have the potential to incorporate a biologically meaningful underlying model of CNV into a joint segmentation and classification process. Additionally they borrow inferential strength from across the data set. Finally they provide a statistical framework for detecting CNVs and enable the detection of such regions based on statistical significance. Unfortunately the HMM presented by Fridlyand et al. (2004) does not fulfil any of these potentials and transpires to be a complicated segmentation method followed by a merging step. In contrast the Bayesian HMMs presented by Guha et al. (2006) and Shah et al. (2006) are probabilistically well-founded, perform well, and provide posterior probabilities on the inferences made. However, all HMMs have many parameters that are difficult to set, and the MCMC employed in both Bayesian HMMs to set these parameters could become too computationally intensive for very high throughput data ($\sim 10^5$ probes). Furthermore the Bayesian HMMs give a probability for the classification of individual probes, but the probability of interest is that of the segments of CNVs, and methods still need to be developed for this task.

Of all the methods discussed here the thresholding methods are the simplest, and are also some of the best suited for the task of locating CNV in aCGH data. The methods work by searching for probes that are above or below a certain threshold, so there is an automatic classification of highlighted probes. Furthermore they have the potential to provide quantitative statistics about highlighted runs of probes. Unfortunately neither of the papers discussed in section 2.6 calculate the statistical significance of putative CNVs, but Pollack (2002) do control the FDR.

SW-ARRAY is a nonparametric thresholding and segmentation procedure. It provides implicit smoothing of the data and an automatic classification of putative regions. Importantly, it is the only method presented that provides a nonparametric test of significance for putative CNVs. Additionally it is one of only two methods (the other is CBS) with published use on a very high throughput data set. Therefore SW-ARRAY will provide a good point of reference for the novel aCGH analytical method presented in this project, which aims to locate and rank putative regions of CNV in a very high throughput ROMA data set (see chapters 3 and 4).

| Method | Authors | Category | Extra | Statistical Framework | Data sets |
|---|---|---|---|---|---|
| - | Pollack (2002) | Thresholding using ref vs. ref experiment | Pre-smoothing | Control FDR | + Pollack (2002): cDNA, 6691 |
| - | Hodgson et al. (2001) | Thresholding using mixture models | - | - | + Hodgson et al. (2001): BAC, 380 |
| Quantile smoothing | Eilers and de Menezes (2004) | Smoothing | - | - | + Nakao et al. (2004): BAC, 2120 |
| Wavelet denoising | Hsu et al. (2005) | Smoothing | - | - | + Snijders et al. (2001): BAC, 2276<br>+ Loo et al. (2004): BAC, 4762 |
| Circular binary segmentation | Olshen and Venkatraman (2004) | Segmentation | Pre-smoothing | - | + Snijders et al. (2001)<br>+ ROMA, 9820<br>+ Graubert et al. (2007): long oligo, 388352 |
| Genetic local search | Jong et al. (2003) | Segmentation | - | - | + BAC, 2275 |
| Penalised likelihood model | Picard et al. (2005) | Segmentation | - | - | + Snijders et al. (2001)<br>+ Nakao et al. (2004) |
| Cluster along chromosomes | Wang et al. (2005) | Segmentation | - | Control FDR | + cDNA, 25736 |
| CGH-Explorer | Lingjaerde et al. (2005) | Binary classification | Smoothing | Control FDR | - |
| HMM | Fridlyand et al. (2004) | HMM for segmentation | Post-process classification | - | + Snijders et al. (2001) |
| Bayesian HMM | Guha et al. (2006) & Shah et al. (2006) | Bayesian HMM for segmentation and auto-classification | - | Posterior probability for states of probes | + Snijders et al. (2001)<br>+ Aguirre et al. (2004): cDNA, 14160<br>+ Bredel et al. (2005): cDNA, 41421<br>+ de Leeuw et al. (2004): SMRT, 32000 |
| Smith-Waterman for aCGH | Price et al. (2005) | Nonparametric segment identification | Implicit smoothing | Nonparametric significance test and ranking | + BAC, 2Mb region<br>+ Komura et al. (2006), 534500 |

Table 2.2: Summary of aCGH analytical methods.

# Chapter 3

# Mouse ROMA experiment

## 3.1 Inbred mouse strains

In the following chapters data from a ROMA CGH experiment will be introduced and analysed. The experiment was carried out on seven inbred mouse strains, with an eighth strain as the reference strain. The seven strains were A/J, AKR/J, BALB/cJ, C3H/HeJ, CBA/J, DBA/2J and LP/J, and the reference strain was C57BL/6J. These inbred mouse strains are of particular interest because there is other genotypic data for them in the form of single nucleotide polymorphisms (SNPs). Furthermore they are the progenitors of a genetically heterogeneous stock (HS) which has been used in a study of quantitative trait loci (QTLs) (Solberg et al. (2006), Valdar et al. (2006a), Valdar et al. (2006b)). With SNP, CNV and QTL data for this group of mouse strains it is possible to study the relationships between these phenomena. The study, in turn, requires the development of new methods for the integration of such data sets.

## 3.2 Probe set

Oxford Gene Technology (OGT) designed 216749 60-mer probes to provide a coverage of Build 33 of the C57BL/6J mouse genome. Of these, 216457 probes were successfully re-mapped to Build 36 of C57BL/6J and this is the build that the following analysis is performed upon. It is important to note that probes are designed with reference to C57BL/6J, so deletions in this strain are not detectable with this method (but this is true for the reference strain in any aCGH experiment).

X and Y chromosomes are not considered in this analysis because CNV due to sex confuses the signal from CNVs due to strain on those chromosomes. Additionally, there are problems because of lower probe density and higher mapping uncertainty for these regions. Analysing only the autosomal chromosomes and removing probes for which no hybridisation signal was obtained, the number of probes remaining per strain ranges from 209444 (CBA/J) to 209935 (C3H/HeJ), with five of the seven test strains having $\geq 209930$ probes remaining.

Figure 3.1 is a cumulative histogram of the distance between neighbouring probes. The median distance between probes is 5.2 Kb, with 90% of distances $< 27.4$ Kb, and 99% $< 89.3$ Kb. The maximum distance between neighbouring probes is 3.5 Mb.

## 3.3 Data exploration

### 3.3.1 Original data

In figure 2.2 the $\log_2$ ratios for all probes across the length of the genome, for A/J versus C57BL/6J, are plotted. Probes which had lower hybridisation in A/J compared to C57BL/6J have negative $\log_2$ ratios and those with higher relative hybridisation have positive ratios. The profile of A/J is typical of all of the mouse strains under examination; it appears to be flat , with a 'shell' of probes that approach large absolute ratios across the genome.

Figure 3.2 gives a more detailed view of the $\log_2$ ratio profile for several of the chromosomes in A/J. A range of observed signal variation is depicted, both within and across chromosomes. Some chromosomes, such as chromosome 10, have very little variation in signal whereas others, such as chromosome 1, have a lot. Within chromosomes there are often distinct regions with a much higher density 'shell' of high absolute ratios. For example, this is the case at the end of chromosome 1, from 170 to 180 Mb. These higher density regions give a structured appearance to the signal variance.

A histogram of the $\log_2$ ratios in the A/J experiment, and a QQ plot of the ratios versus the standard normal distribution, are shown in figure 3.3 to help gauge the normality of the distribution of the ratios. The 'shell' of high absolute ratio probes is manifested in the heavy tails of the distribution that is apparent in both plots. From the histogram it is clear that the negative tail is much heavier than the positive one. This attribute is discussed later in section 3.4.

To explore the difficulties incurred by the use of such high throughput methods of aCGH, the task of identifying a region of CNV is discussed briefly. There is a known 480 Kb duplication on chromosome 17 at around 30 Mb. In this example 65 probes are implicated in the duplication in A/J, as compared to the two or so probes that one would expect to be implicated on a BAC array. With the knowledge that

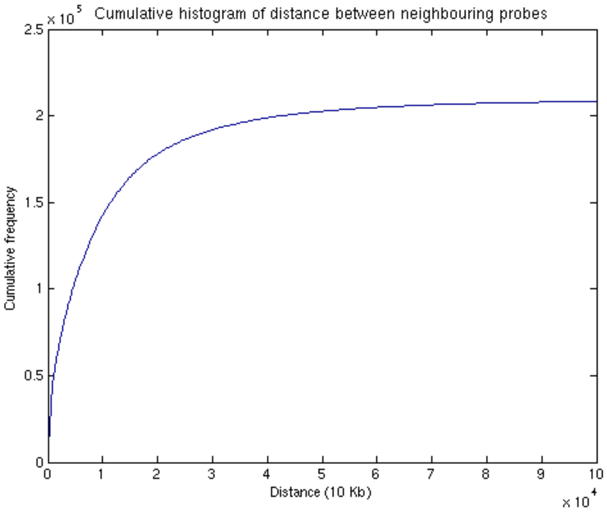

Figure 3.1: Cumulative histogram of the distance, in 10s of Kb, between neighbouring probes.(The line goes through the midpoint of each bar in the histogram.)

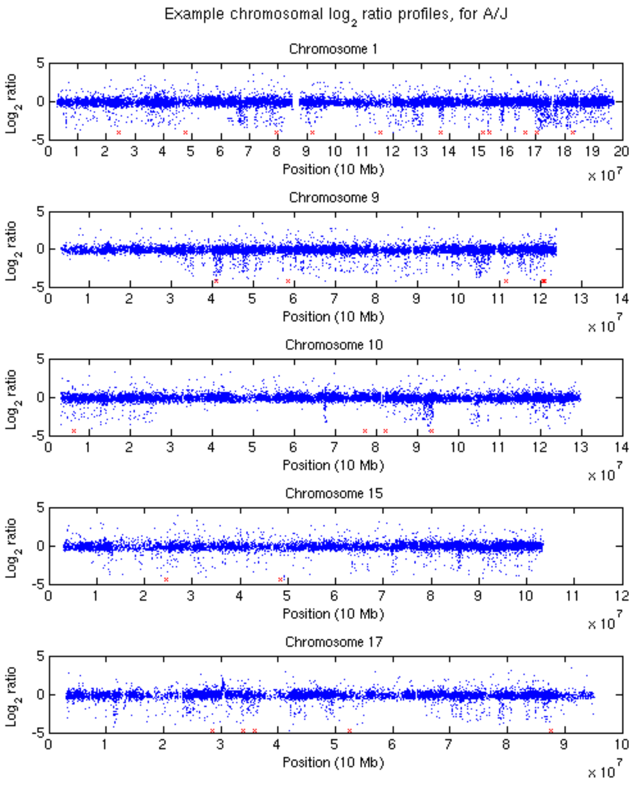

Figure 3.2: Example chromosomal $\log_2$ ratio profiles plotted for A/J. $\mathrm{Log}_2$ ratios are plotted against physical probe locations, given in 10s of Mb. Probes for which there was an observed signal from both A/J and C57BL/6J are plotted in blue. Probes for which a signal was only observed on C57BL/6J are in red. The latter probes are nominally given a $\log_2$ ratio of the minimum ratio observed in the former set. A range of observed variation is shown on different chromosomes. Chromosome 10 has the least variation in signal. Chromosomes 9, 15 and 17 display medium amounts of variation, with chromosome 1 having significantly more from 170 Mb onwards. Sometimes, in regions where large absolute ratios are observed, the large ratios are somewhat sparse in comparison to the density of smaller ratios. This is true of the start of chromosome 1 (start to 40 Mb), the middle of chromosome 9 (50 to 100 Mb), the start of chromosome 10 (up to 25 Mb), and large parts of chromosomes 15 and 17. However, there are often also regions in which there is an increase in the density of the 'shell' of higher absolute ratio probes. This is true, for example, of the region from 170 Mb to 180 Mb on chromosome 1. These regions yield a more structured appearance, that is to say there are distinct regions which seem to have a much higher propensity for large absolute ratios than the rest of the genome.

this large duplication exists, it is easy to locate it by eye in figure 3.2. However, with 10000 probes on chromosome 17 alone, locating an unknown CNV as large as this would be hard, and it would be near impossible to detect smaller unknown CNVs by eye. Thus it is clear that an automated process for the detection of CNVs is especially necessary with very high throughput aCGH data.

In the remainder of this chapter the need for an additional pre-processing step to normalise between the arrays used within one strain versus C57BL/6J experiment is assessed, and the role of SNPs in the variation of the signal is explored.

### 3.3.2 Normalising the data

Each strain versus C57BL/6J ROMA experiment used either 9 or 10 arrays of $\sim$ 20000 probes to obtain an overall coverage of $\sim$ 200000, and the raw signals from each array were normalised within arrays, by OGT, using locally weighted scatter plot smoothing(*lowess*). Such normalisation corrects for differences in labelling and detection efficiencies for the fluorescent labels, and also for differences in the quantities of genomic DNA from the samples. These factors can cause a shift in the mean ratio of the intensities from the two labels so the intensities must be re-scaled to account for this before the data is analysed. While this normalisation has rendered the data within slides comparable, no normalisation has been performed between slides, so there is no guarantee that $\log_2$ ratios *from different slides within one experiment* are comparable. Here the requirement for a between slides normalisation is assessed, and a simple normalisation procedure is proposed.

Observing the bar charts in figure 3.4 it can be seen that each slide contains probes from across the genome, and that the probes from each chromosome are distributed approximately equally across all slides. Assuming that most probes in the genome do not lie in regions of CNV (a reasonable assumption given the distribution of probes seen in figure 2.2 and in the histogram of figure 3.3), this assignment of probes to slides should mean that the distribution of $\log_2$ ratios obtained from all of the slides should

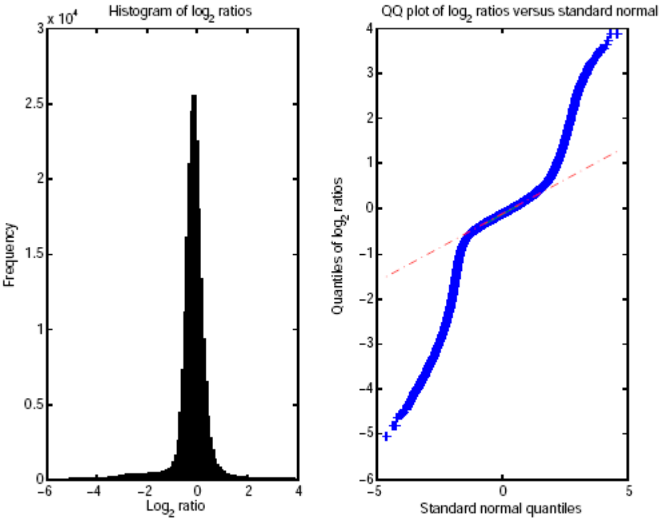

Figure 3.3: Left: Histogram of $\log_2$ ratios from the A/J versus C57BL/6J experiment. Right: QQ plot of $\log_2$ ratios, from the A/J versus C57BL/6J experiment, versus standard normal (dotted red line). Both plots clearly depict the non-normal distribution of the ratios, with heavy tails on both sides of the distribution, but especially so (as is apparent in the histogram), on the negative side.

not be biased by chromosomal origin. However it is possible that there may be slide specific effects.

To assess whether this is the case, box and whisker plots of the $\log_2$ ratios from each slide of the A/J ROMA experiment are shown in figure 3.5. For several of the pairwise comparisons between box and whisker plots for slides, the medians of the two slides differ at the 5% significance level. Additionally the whiskers (1.5 times the interquartile range) show that there is sometimes a difference in the size of the range of values obtained on the slides too. Furthermore a Kruskal-Wallis test, (the rank-randomisation analogue of the ANOVA), carried out at the 5% level, rejects the null hypothesis that all of the $\log_2$

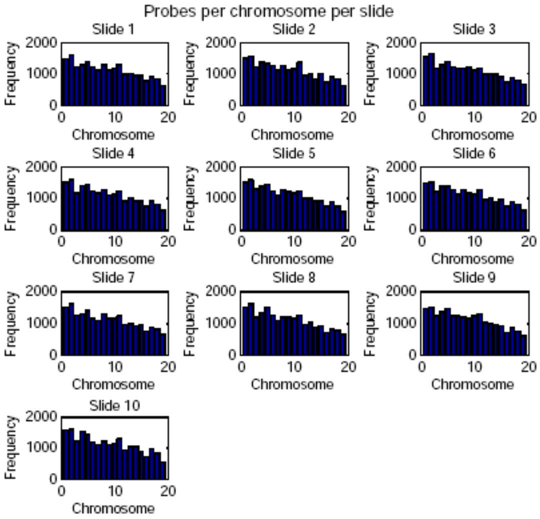

Figure 3.4: Bar charts of the number of probes from each chromosome on each slide. It can be seen that each slide has a selection of probes from across the genome, and that the probes from each chromosome are distributed approximately equally across all slides.

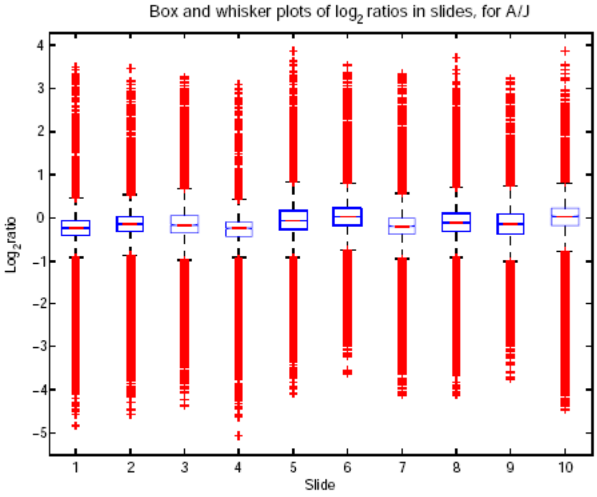

Figure 3.5: Box and whisker plots of the $\log_2$ ratios of probes in each slide of the A/J versus C57BL/6J experiment. The lines on the boxes are notches which represent a robust estimate of the uncertainty about the medians for box-to-box comparison. Boxes whose notches do not overlap indicate that the medians of the two groups differ at the 5% significance level. This is the case for many of the pairwise comparisons between slides. The whiskers represent 1.5 times the inter-quartile range, and it can be seen that the size of the range of values obtained in slides is sometimes different (compare slides 4 and 5, for example)

ratios are drawn from the same distribution (see table 3.1).

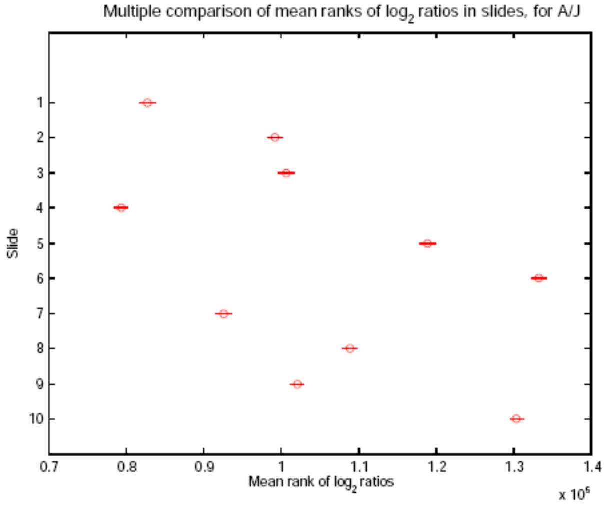

Figure 3.6: Graph of the multiple pairwise comparisons, between slides, of the $\log_2$ ratios from the A/J versus C57BL/6J experiment. The tests are carried out using the ranks of the probes, rather than the numeric values, as calculated for the Kruskal-Wallis test described above. Each slide mean rank is depicted with a circle, and the 95% confidence interval with a line. Two means are significantly different if their intervals are disjoint, and are not significantly different if their intervals overlap. All of the pairwise comparisons, except those between slides 2 and 3, and slides 3 and 9, reject the null hypothesis at the 5% significance level, using Tukey's honestly significant difference criterion, that the $\log_2$ ratios from the pair of slides come from the same distribution.

Lastly the results of a multiple comparison test, used to show which pairs of slides are significantly different from one another, are shown in figure 3.6. All of the pairwise comparisons, except those between slides 2 and 3, and slides 3 and 9, reject the null hypothesis at the 5% significance level, using Tukey's honestly significant difference criterion, that the $\log_2$ ratios from the pair of slides come from the same distribution.

It seems desirable, therefore, to normalise slides such that their $\log_2$ ratios come from the same distribution. Since lowess normalisation has already been performed to account for variability within slides, and since the slides have $\log_2$ ratios that come from scaled and shifted versions of the same distribution (see pairwise QQ plots in figure 3.7), it is sensible to simply scale and shift the data to one chosen distribution. Due to the heavy tails of the distribution, the $\log_2$ ratios from each slide are normalised by

| **Source** | **SS** | **df** | **MS** | **Chi-sq** | **Prob>Chi-sq (p)** |
|---|---|---|---|---|---|
| Slides | 6.3431e+13 | 9 | 7.0479e+12 | 1.7291e+04 | 0 |
| Error | 7.0626e+14 | 209803 | 3.3663e+09 | | |
| Total | 7.6969e+14 | 209812 | | | |

Table 3.1: Kruskal-Wallis table. Standard ANOVA table calculated using the ranks of the data rather than the numeric values. (Ranks are found by ordering the $\log_2$ ratios from smallest to largest across all slides. The rank for a tied $\log_2$ ratio is equal to the average rank for all ratios tied with it.) The first column is the source of variability, the second is the sum of squares due to each source, the third is the degrees of freedom associated with the source, the fourth is the chi-squared statistic (replacing the F statistic in an ANOVA), and the p-value measures the significance of the chi-square statistic. The Kruskal-Wallis test rejects the null hypothesis, at the 5% level, that all of the $\log_2$ ratios are drawn from the same distribution.

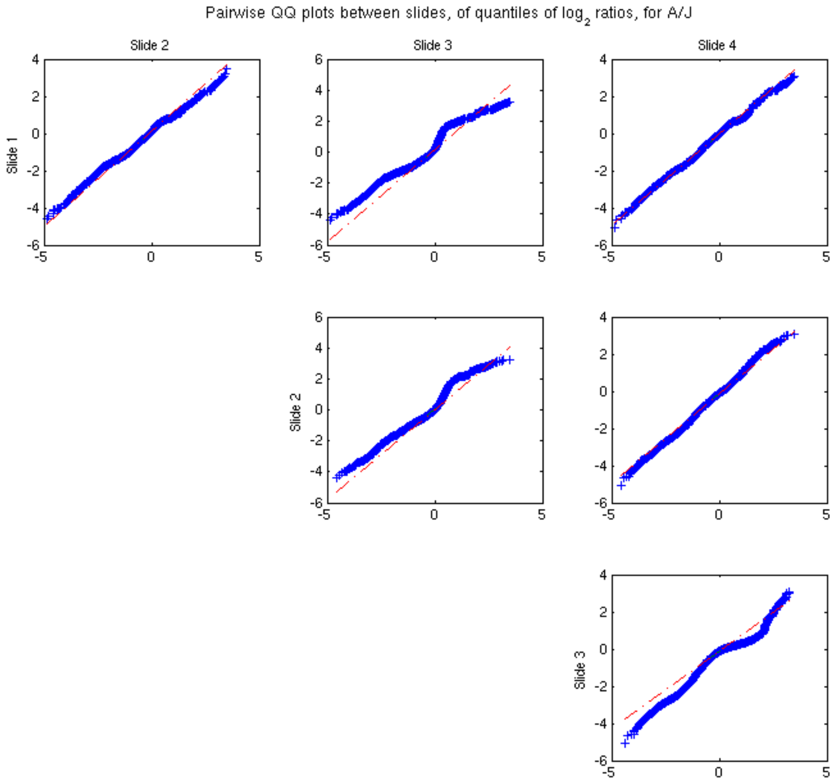

Figure 3.7: Pairwise QQ plots, from slides 1 to 4, of the $\log_2$ ratios in the A/J versus C57BL/6J experiment. The straight red line is where quantile-quantile pairs from the two sampled distributions would lie if the distributions were scaled and shifted versions of one another. The actual quantile-quantile pairs are plotted as blue crosses. These pairwise plots are representative of all of the pairwise plots between slides for A/J. In all cases the blue crosses lie very close to the red line, so it is fair to assume that the $\log_2$ ratios from different slides come from scaled and shifted versions of the same distribution.

subtracting a robust estimator of the location, and dividing by a robust estimator of the spread; namely the median and median absolute deviation (MAD) respectively. (Note that quantile normalisation can be used for between slide normalisation but, since it renders the entire distribution (i.e. not only the inter-quartile range) of every slide identical, such a normalisation is not appropriate because it could have the detrimental effect of removing signal from the tails of the distribution, and hence of real CNVs.)

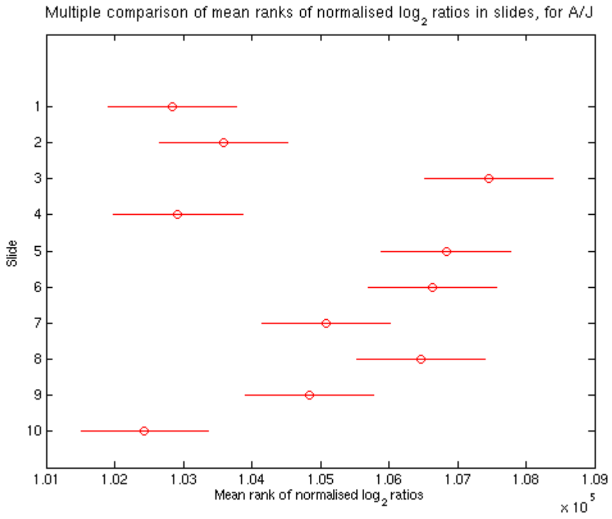

Figure 3.8: Graph of the multiple pairwise comparisons, between slides, of the normalised $\log_2$ ratios from the A/J versus C57BL/6J experiment. Each slide mean rank is depicted with a circle, and the 95% confidence interval with a line. Two means are significantly different if their intervals are disjoint, and are not significantly different if their intervals overlap. Far fewer pairwise tests between slides reject the null hypothesis that the $\log_2$ ratios on those slides are from the same distribution.

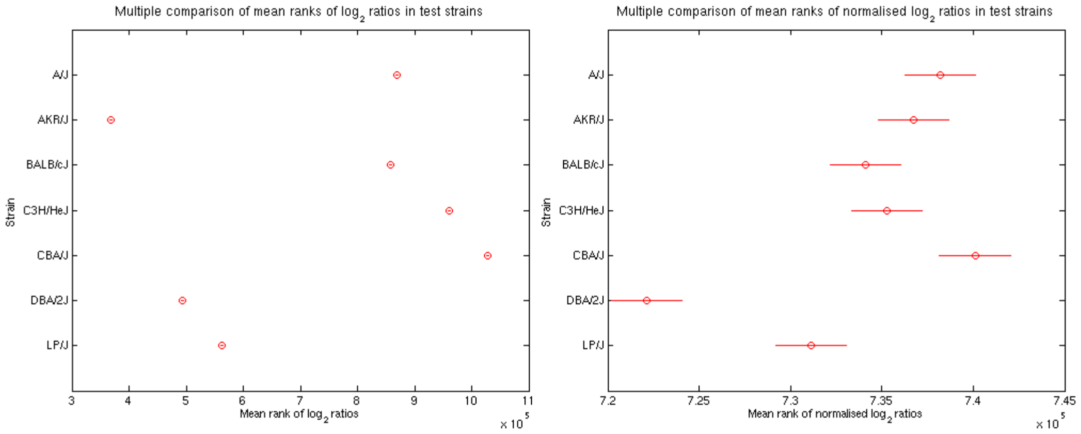

Figure 3.9: Left: Graph of the multiple pairwise comparisons, between mouse strain experiments, of the $\log_2$ ratios. All tests reject the null hypothesis that $\log_2$ ratios from the compared strains come from the same distribution. Right: Graph of the multiple pairwise comparisons, between mouse strain experiments, of the normalised $\log_2$ ratios. Far fewer pairwise tests between strains reject the null hypothesis that the $\log_2$ ratios for those strains are from the same distribution. However all tests involving DBA/2J still reject the null hypothesis.

A multiple comparison of the normalised $\log_2$ ratios is shown in figure 3.8. Far fewer pairwise hypothesis tests between slides reject the null hypothesis that the $\log_2$ ratios on those slides are from the same distribution. Additionally, this normalisation makes it easier to directly compare the $\log_2$ ratios obtained from the different mouse strain ROMA experiments. Figure 3.9 shows the result of the multiple comparison test between strains, before and after normalisation. Once again far fewer pairwise tests reject the null hypothesis. However all the pairwise tests involving DBA/2J still reject the null hypothesis, so the development of further normalisation techniques, or the use of existing ones, for example those implemented by Agilent in Genespring, might be required to make the strain data more comparable. Nonetheless, for current needs this normalisation method is sufficient, and for all further analysis $\log_2$ ratios normalised in this way will be used exclusively.

## 3.4 Analysing the relationship between SNPs and ROMA data

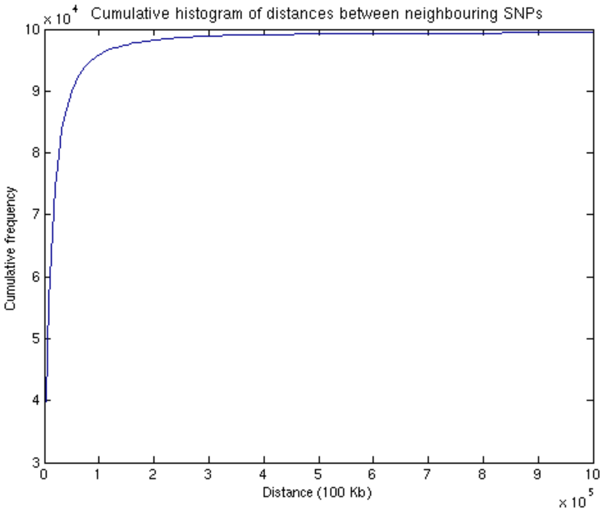

Figure 3.10: Cumulative histogram of the distance, in 100s of Kb, between neighbouring SNPs.(The line goes through the midpoint of each bar in the histogram.)

A SNP data set is available for mouse strains A/J, AKR/J, BALB/cJ, C3H/HeJ, CBA/J, DBA/2J, LP/J and C57BL/6J. The data is a combination of 13800 SNPs from the Wellcome-CTC mouse inbred strain SNP genotype set, and ∼ 140000 SNPs from the BROAD institute, kindly provided by Mark Daly (http://www.well.ox.ac.uk/mouse/INBREDS). Figure 3.10 shows the histogram of distances between neighbouring SNPs, mapped to C57BL/6J. The mean distance between SNPs in the data set is ∼ 24 Kb, the median is ∼ 9.2 Kb, with 90% of SNPs less than ∼ 50.9 Kb apart.

### 3.4.1 Previous discussions on the effect of SNPs on ROMA

In the original paper describing ROMA (Lucito et al., 2003), and in the mouse ROMA paper written by the same group (Lakshmi et al., 2006), the effect of SNPs within and around probe sequences are suggested as the main cause of the “shell” of higher $\log_2$ ratios observed over the whole genome.

As explained in section 4, in ROMA representations of a genome are made using PCR to amplify fragments of DNA previously made with a restriction endonuclease. PCR selects short fragments, and the cleavage sight of the restriction enzyme is known, so the resulting set of representations are short fragments of DNA that are predictable from the genome sequence and also reproducible. However, as discussed in both Lucito et al. (2003) and Lakshmi et al. (2006), SNPs in the restriction sight of the restriction endonuclease will cause a failure in the fragmentation process such that the corresponding fragment will not be created, thus appearing as a total deletion; this is one mechanism by which SNPs can interfere in the ROMA hybridisation signal. A second mechanism by which SNPs interfere with ROMA hybridisation occurs when there is a SNP within an array probe; this will cause reduced hybridisation, but the amount by which this will happen is hard to predict. Therefore, since SNP data is available for the inbred mouse strains, it is necessary to ascertain whether SNPs cause variation in the ROMA experiment via one of these two mechanisms.

Important work by Wade et al. (2002) has revealed that there is a mosaic structure in the inbred mouse genome; when two inbred mouse strains are compared to one another "long segments of either extremely high ($\sim$ 40 SNPs per 10 Kb) or extremely low ($\sim$ 0.5 SNPs per 10 Kb) [SNP] rates" are observed. Furthermore, there is evidence to suggest that the transition between these regions is sharp. (This structure is due to the breeding history of inbred mice. In segments where the SNP rate is low "the two strains share a very recent common subspecies origin". Conversely, in segments where the SNP rate is high "the two strains inherited the region from different subspecies".)

The mosaic SNP segmentation of inbred mouse strains helps elucidate the effect of SNPs on the ROMA data; by partitioning the ROMA data from two inbred mouse strains, into $\log_2$ ratios from probes in high SNP rate segments and $\log_2$ ratios from probes in low SNP rate segments, it is possible to observe the effect of SNPs (or lack thereof) on the ROMA data. Furthermore it is interesting to assess whether SNPs, in addition to causing unwanted variance in the hybridisation signal, are also associated with loss CNVs such that SNPs and loss CNVs occur in the same regions of the genome. These analyses are presented in the remainder of this chapter.

For simplicity, from here in, segments of the genome between a pair of inbred mouse strains where the SNP rate is low are termed *SNP matched*, and segments in which the SNP rate is high are termed *SNP non-matched*. Furthermore if one of the pair is the *reference* strain then the *test* strain is termed *reference SNP matched* in segments of the genome where the test and reference strain are SNP matched. Similarly, the test strain is termed *reference SNP non-matched* where the strains are SNP non-matched. Furthermore, to avoid confusion with terminology already used for aCGH analytical methods, all such *segments* are referred to as *regions*.

### 3.4.2 Comparison of ROMA data from SNP matched and non-matched regions between pairs of strains

The correlation between ROMA data from the SNP matched regions of a pair of strains is compared here to the correlation between the ROMA data from their SNP non-matched regions. This comparison is carried out between all pairs of strains. Figure 3.11 is a schematic of the analysis carried out between two test strains. First, using an adaptation of SW-array described in section 4.1, box 4.2, SNP matched and non-matched regions are identified between two strains (in figure 3.11 W and X are SNP matched between A and B, and Y and Z are SNP non-matched). Next, all of the ROMA data that lie in SNP matched regions for the two strains (in the diagram these are the $\log_2$ ratios corresponding to the first eight probes), are compared for correlation, and then the ROMA data in SNP non-matched regions (the last eight probes) are compared. Thus a correlation coefficient between ROMA data from SNP matched regions between the two strains is obtained, and a corresponding one is obtained for their SNP non-matched regions.

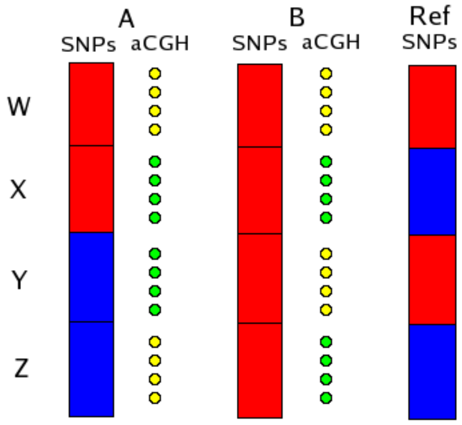

Figure 3.11: Schematic of the analysis between one pair of test strains, A and B, for which there is SNP data (red and blue columns), and ROMA data (yellow and green circles). Also shown is the data available for the ROMA reference strain, Ref, for which there is only SNP data. All SNPs are bi-allelic, and therefore the SNP data can be depicted by two colours. If two strains have matching SNPs over a region of the genome they share the same colour in that segment of the SNP column. Thus, for example, A and B are seen to have matching SNP values in regions W and X, and non-matching SNP regions in Y and Z. Conversely, ROMA data can take a continuous range of values, but to simplify the diagram they have been depicted in only two colours; two probes with the same colour have similar values, and two probes with different colours have dissimilar values. The ROMA data are given in genome order and can therefore be assigned to one of the genomic regions according to their location. Finally, the correlation between ROMA probes from A and B in SNP matched regions W and X is compared to the correlation between ROMA probes in SNP non-matched regions Y and Z. (Note that the specific relationship between SNPs and CNVs depicted in the diagram, namely that there is an association, is only intended as an example of a possible relationship, and at this stage of the discussion does not represent any conclusions about the actual relationship observed.)

A scatter plot of the correlation coefficients obtained for ROMA data in SNP matched and non-matched regions between pairs of test strains is shown in figure 3.12. By observation it is clear that there is a higher correlation between ROMA data obtained from two strains in their SNP-matched regions than there is between the ROMA data in their SNP non-matched regions. Furthermore the *t*-test for difference of means yields a p-value of $4.8643e^{-24}$.

This confirms a relationship between SNPs and ROMA data, but it is not yet possible to discern how much of the relationship is due to SNPs and CNVs occurring in the same regions of the genome, and how much of the relationship is due to SNPs causing hybridisation problems in ROMA. Clearly, if SNPs are associated with CNV then when two test strains are SNP matched they will also share CNV structure relative to the reference strain, so their ROMA data (which is measuring CNV) will be better correlated. However, increased correlation between ROMA data from SNP matched regions, as compared to correlation between data from SNP non-matched regions, can also occur as follows:

In regions where a pair of test strains are SNP matched to one another they must also, (because SNPs are bi-allelic), share the same *reference SNP match* and *reference SNP non-match* regions. This means that if SNPs interfere with hybridisation in ROMA via the two mechanisms described previously, then in the regions where test strains are SNP matched they will be subject to the same pattern of hybridisation problems. In other words the test strains will both have little hybridisation problems when they are both reference SNP matched, and the same hybridisation problems when they are both reference SNP non-matched. Conversely, in regions where test strains are SNP non-matched they will not share the same reference SNP match and non-match structure, and will therefore also have different patterns of ROMA hybridisation problems. So ROMA data from SNP matched regions will be better correlated than ROMA data from SNP non-matched regions even if SNPs are not associated with CNV.

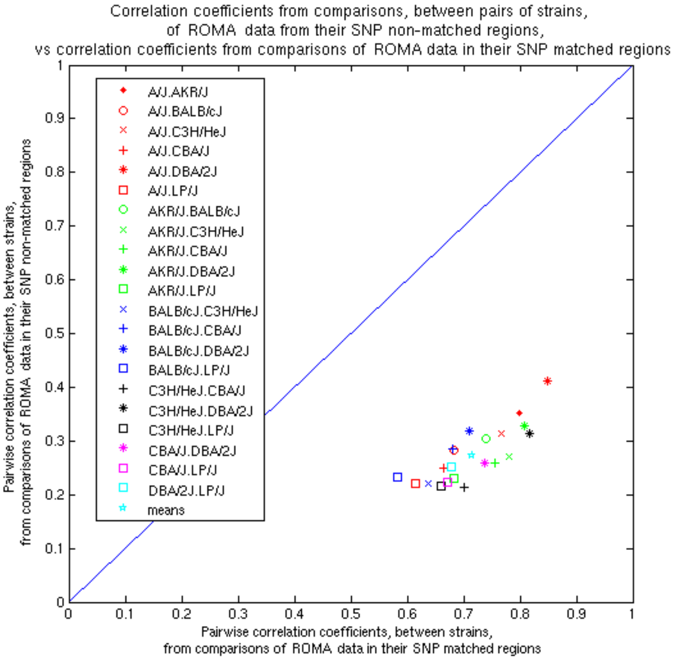

Figure 3.12: Scatter plot of the correlation coefficients obtained for ROMA data in SNP-matched and non-matching regions, between pairs of test strains. The mean correlation of ROMA data from SNP matched regions is 0.71771, and for SNP non-matched regions is 0.27888. Performing a *t*-test for difference of means yields a p-value of $4.8643e^{-24}$.

### 3.4.3 Further examining the relationship between SNPs and ROMA data

To further explore the nature of the relationship between SNPs and ROMA data another analysis is carried out here which *only uses the ROMA data from SNP matched regions between pairs of test strains.* (Referring to figure 3.11, the combined segment of W and X would be one such region of the genome for strains A and B.) These regions are divided into regions where both test strains are reference SNP matched (in the figure, W), and into regions where both test strains are reference SNP non-matched (X). Next the relationship between ROMA data from reference SNP matched regions is compared to the relationship between ROMA data from reference SNP non-matched regions. Such a comparison helps elucidate the nature of the relationship between SNPs and ROMA data.

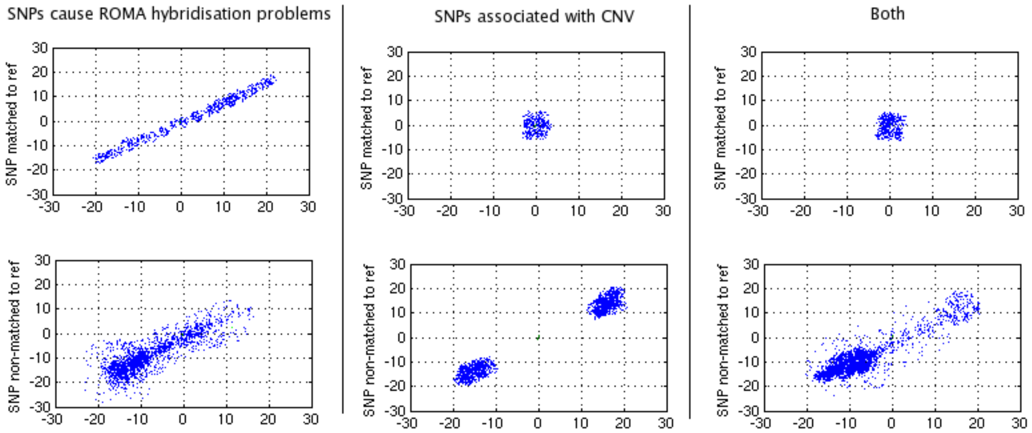

Figure 3.13: Predicted scatter plots between ROMA data from the SNP matched regions between a pair of test strains. Three scenarios are shown: Left: SNPs cause hybridisation problems in ROMA but are not otherwise associated with CNVs. Middle: SNPs are associated with CNVs such that they occur in the same regions of the genome. Right: SNPs cause hybridisation problems in ROMA *and* occur in the same regions of the genome as CNVs. Each scenario has two predicted scatter plots: the top plot is for ROMA data from regions where both the test strains are reference SNP matched; and the bottom plot is for ROMA data from regions where both the test strains are reference SNP non-matched.

Three types of relationship between SNPs and ROMA data are possible: SNPs cause hybridisation problems in ROMA; SNPs are associated with CNVs such that they occur in the same regions of the genome; SNPs cause hybridisation problems in ROMA *and* occur in the same regions of the genome as CNVs.

Expected scatter plots of ROMA data, from two test strains, from regions of the genome where the test strains are SNP matched to one another are shown in figure 3.13. Two scatter plots are predicted for each of the above scenarios; one for ROMA data from regions of the genome where both test strains are reference SNP matched, and one for ROMA data from regions of the genome where both test strains are reference SNP non-matched.

If SNPs only cause hybridisation problems in ROMA and are not associated with CNVs such that they lie in the same regions of the genome (figure 3.13, left), then a spread of $\log_2$ ratios, from very low to very high, is expected regardless of whether or not the test strains are reference SNP matched or reference SNP non-matched. A cloud of $\log_2$ ratios whose values are not well correlated is predicted in the latter plot because, as explained previously, SNPs within probes will cause a reduction in hybridisation, but the amount by which this will occur is hard to predict and not necessarily systematic. Also note that because SNPs can only cause a decrease in hybridisation, an increase in the density of probes with low $\log_2$ ratios is also predicted in the latter plot, with a corresponding reduction in the density of positive $\log_2$ ratios.

If SNPs are associated with CNVs such that they lie in the same regions of the genome, but they do not cause hybridisation problems in ROMA (middle), then when the test strains are reference SNP matched they should both have $\log_2$ ratios around zero, and when they are both reference SNP non-matched they should both have low and high $\log_2$ ratios, and very few ratios around zero. Note that these plots are based on the scenario in which SNPs and CNVs only occur in the same regions. Clearly if this were not the case then there would be some high and low $\log_2$ ratios in the SNP match plot, and some near zero $\log_2$ ratios in the SNP non-match plot.

Finally, if SNPs cause hybridisation problems and are also associated with CNV (right), then when both strains are reference SNP matched they should have $\log_2$ ratios clustered around zero, and when they are both reference SNP non-matched they will both have low and high $\log_2$ ratios. Once again, as for the first scenario (left), a cloud of $\log_2$ ratios whose values are not well correlated is predicted in the reference SNP non-matched plot. Also in this plot, as in the first scenario, an increase in the density of probes with low $\log_2$ ratios is predicted, with a corresponding reduction in the density of positive $\log_2$ ratios. Furthermore, some $\log_2$ ratios around zero are expected when probes in gain CNV regions have some, but not all, of their hybridisation reduced.

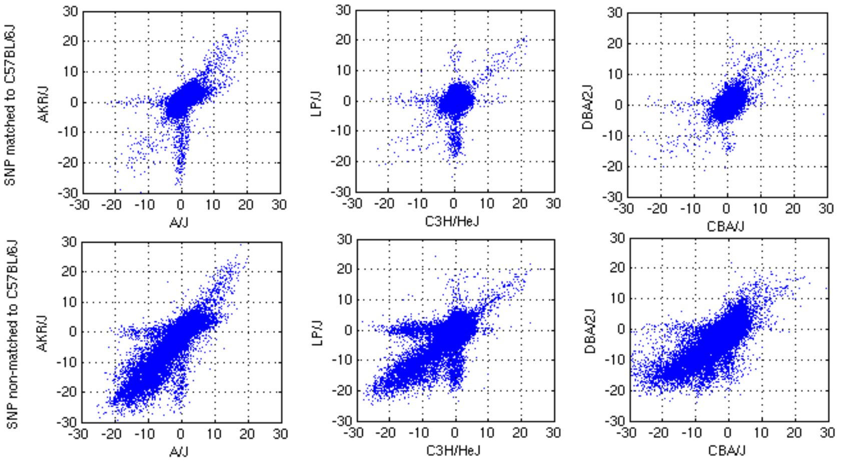

Figure 3.14: Three pairs of scatter plots observed in the real pairwise test strain comparisons that are typical of the types of plots seen over all of the pairwise comparisons. Left to right: AKR/J vs A/J, LP/J vs C3H/HeJ and DBA/2J vs CBA/J. Top: ROMA data from regions where the two test strains are SNP matched to each other and are both reference (C57BL/6J) SNP matched. Bottom: ROMA data from regions where the two test strains are SNP matched to each other and are both reference SNP non-matched.

Figure 3.14 shows three pairs of scatter plots, observed in the real pairwise test strain comparisons, which are representative of the types of plots seen over all of the pairwise comparisons. All the plots show a clustering of $\log_2$ ratios around zero when test strains are reference SNP matched, and a noticeable increase in the spread of values when the test strains are reference SNP non-matched. However while all pairwise comparisons show a small increase, in the reference SNP non-matched regions, of probes with a high positive $\log_2$ ratio, the majority of the increase is in the negative direction. Referring to the discussion of figure 3.13, these plots are indicative of both an association between SNPs and CNVs, and also SNPs as a source of hybridisation failure in the ROMA experiment.

In all three of the reference SNP matched region plots, a small number of probes have well correlated large absolute $\log_2$ ratios. In all three of the reference SNP non-matched region plots there are many probes with near zero $\log_2$ ratios (perhaps more than would be expected due to gain CNV probes containing SNPs). These observations suggest that although there is an association between SNPs and CNVs, the two phenomena do not always occur in the same regions of the genome.

The pairwise scatter plots between A/J and AKR/J, and C3H/HeJ and LP/J show clusters of $\log_2$ ratios that have large negative values in one of the test strains but not in the other. Such clusters occur mainly in the regions where the test strains are reference SNP non-matched, but they also occur in the reference SNP matched regions. This pattern was not depicted in the predicted scatter plots shown in figure 3.13, and is most likely due to another source of hybridisation failure, or a mis-classification of regions in the test strains as SNP matched or SNP non-matched to each other or to the reference strain. Such sets of probes also exist in the pairwise comparison between CBA/J and DBA/2J, but they are not as distinct due to the additional cloud of probes whose values are not as well correlated between the test strains. This cloud of $\log_2$ ratios probably occurs because SNPs within probes cause a reduction in hybridisation, the magnitude of which is hard to predict and not necessarily systematic.

These findings motivate a threshold based CNV discovery method, discussed in the next chapter, which sets thresholds for ROMA data dependent on whether they come from reference SNP matched or non-matched regions; higher thresholds are set in the reference SNP non-matched regions, (compared to those set in the reference SNP matched regions), to account for the proportion of the variance seen in them that is due to SNPs. The results also indicate that search strategies which use data about known SNPs and CNVs in one strain to inform the search for CNVs in another are feasible and potentially powerful. Such ideas, discussed further in future work, will form an extension to methods discussed here.

# Chapter 4

# ROMA data analysis

## 4.1 Excursion Finder

A nonparametric thresholding method for CNV detection, which searches for runs of probes whose $\log_2$ ratios lie above or below a set threshold (*excursions*), **Excursion Finder**, is presented here.

As explained in section 3.4, a SNP data set is available for the mouse strains tested in the ROMA experiment described in chapter 3. Excursion Finder (EF) is novel because it integrates this SNP data with the ROMA data. In the first part of the algorithm each test strain is compared to the reference strain to find their SNP matched and non-matched regions. Next, with the aim of explaining that proportion of the variance in the ROMA data which is due to SNPs, different thresholds are set for the ROMA data in the SNP matched and non-matched regions; higher thresholds are set in the non-matched regions to account for the extra SNP-caused variance observed in them. The algorithm is detailed in boxes 4.1 and 4.2 and a schematic diagram of the algorithm is shown in figure 4.3. Finally a permutation algorithm, explained in box 4.4 and figure 4.5, is used to estimate an empirical null distribution of excursion lengths. (This method is based on a similar premise to that of the permutation test used in Price et al. (2005); that successive excursions lengths from the permuted data approximate the null distribution of excursion lengths.) The null distribution is then used to assess the significance of excursions located by EF.

### 4.1.1 EF results

EF is used here to analyse the mouse ROMA data in conjunction with the available SNP data. All of the located putative CNVs are available on-line at http://gscan.well.ox.ac.uk/gscan/wwwqtl.cgi, where they can be viewed simultaneously with SNP distribution patterns and QTLs in the mouse strains.

Using an empirical null distribution estimated by the permutation algorithm in box 4.4 with the “guess threshold” for the length of real excursions set to 4, EF yields putative loss CNVs with probe lengths ranging from 3 to 14, and gain CNVs with probe lengths ranging from 3 to 45. The median probe length for both types of CNV is 3. The $75^{th}$ percentile is 4 and 3 for loss and gain CNVs respectively. The $95^{th}$ percentile of probe lengths is 5 for both, and the $99^{th}$ is 8 for both. Minima, maxima, means and percentiles of the lengths of loss and gain CNVs found by EF are given in table 4.1.

**EF algorithm**

For each test strain:

1. Find all reference SNP matched and non-matched regions using the SNP matching algorithm in box 4.2.
2. Find the $R^{th}$ percentile, where $0 < R < 50$, and the $Q^{th}$ percentile, where $Q = 100 - R$, of all ROMA data in all reference SNP matched regions across the whole genome. $tm_{lower}$ and $tm_{upper}$ are assigned these values in order, and are respectively the lower and upper thresholds for loss and gain CNV in regions of the test strain that are reference SNP matched to C57BL/6J.
3. Find the $R^{th}$ percentile, where $0 < R < 50$, and the $Q^{th}$ percentile, where $Q = 100 - R$, of all ROMA data in all reference SNP non-matched regions across the whole genome. $tn_{lower}$ and $tn_{upper}$ are assigned these values in order, and are respectively the lower and upper thresholds for loss and gain CNV in regions of the test strain that are reference SNP non-matched.
4. For each set of ROMA data in a reference SNP matched region, search for all excursions of probes in which all probes have a $\log_2$ ratio$< tm_{lower}$. Keep all excursions whose length is significant at the 5% level according to the permuted null distribution of lengths of excursions (see box 4.4). These excursions are then the putative regions of *loss* CNV in reference SNP matched regions.
5. For each set of ROMA data in a reference SNP matched region, search for all excursions of probes in which all probes have a $\log_2$ ratio$> tm_{upper}$. Keep all excursions whose length is significant at the 5% level according to the permuted null distribution of lengths of excursions. These excursions are then the putative regions of *gain* CNV in reference SNP matched regions.
6. Repeat steps 4 and 5 for the reference SNP non-matched regions.

Box 4.1: EF: a threshold based CNV detection algorithm that integrates SNP data into the threshold setting process.

| | **EF** | | **SW-ARRAY** | |
|---|---|---|---|---|
| | **Loss CNV** | **Gain CNV** | **Loss CNV** | **Gain CNV** |
| **min** | 0.37 | 0.51 | 9 | 3 |
| **max** | 4107 | 544 | 15086 | 452 |
| **mean** | 34 | 36 | 1808 | 175 |
| **median** | 15 | 22 | 899 | 111 |
| **10** | 2 | 5 | 141 | 20 |
| **25** | 6 | 10 | 322 | 28 |
| **75** | 31 | 47 | 1938 | 307 |
| **90** | 67 | 84 | 4103 | 451563 |
| **95** | 106 | 113 | 8318 | “” |
| **99** | 304 | 192 | 13807 | “” |

Table 4.1: Minima, maxima, means and percentiles of lengths (in Kb) of loss and gain CNVs found by EF (left) and SW-ARRAY (right).

**SNP matching algorithm**

For any two strains, $A$ and $B$, with arrays of SNP values on each chromosome $A_{snp}$ and $B_{snp}$ indexed by $i$:
For each chromosome:

1. Calculate a comparison array, $comp_{AB}$, where:

$$comp_{AB}(i) = (A_{snp}(i) == B_{snp}(i)), \tag{4.1}$$

2. Subtract a threshold $t_0$, where $0 < t_0 < 1$, from each element in $comp_{AB}$:

$$adj\_comp_{AB}(i) = comp_{AB}(i) - t_0 \tag{4.2}$$

   Thus matches are rewarded with a score of $1 - t_0$ and the non-matches are penalised with a score of $0 - t_0$. The mean of the adjusted scores in $adj\_comp_{AB}$ must be negative.[1]
3. Find all high-scoring islands in $adj\_comp_{AB}$ using the one dimensional Smith-Waterman algorithm described in section 2.8.
4. Choose all high scoring islands whose length is $\geq$ the number of matches required to overcome one non-match in an island of matches[2]. These are the SNP matched regions.
5. The SNP non-matched regions are then all regions of the genome in between matched regions, plus the region between the start of the chromosome and the first matched region, and the region between the end of the last matched region and the end of the chromosome.

**Notes**

1. Because there are only two values that an element of $adj\_comp$ can take, $t_0$ gives direct control over the minimum length of a contiguous run of matches required to overcome exactly one non-match. For example if $t_0 = 0.9$ the reward for a match is 0.1 and the penalty for a non-match is $-0.9$. This means that 10 matches are required to overcome exactly one non-match. With 9 matches required for every non-match thereafter. Choosing $t_0$ is therefore an heuristic process dependent on the effect of non-matches in a "matched" region versus the requirement for a smooth segregation of the genome that does not switch too rapidly from matched to non-matched.
2. In Price et al. (2005) the statistical significance of islands is estimated by permutation. However since the desired required length for a contiguous run of matches to overcome a non-match is already determined and implemented through $t_0$, it is sensible to use this same length as the threshold for accepted matched regions.

Box 4.2: SNP matching algorithm: an adaptation of the one dimensional Smith-Waterman algorithm (Price et al. (2005)) for the location of SNP matched and non-matched regions between two strains.

EF locates 5266 CNV loss regions across all seven test strains, and 2450 gain regions. The number of loss regions per strain ranges from 668 (BALB/cJ) to 845 (AKR/J). The number of gain regions per strain varies from 258 (A/J) to 493 (LP/J). The percentage of the C57BL/6J genome that is CNV with each of the test strains is found to vary from 0.75% (LP/J) to 1.42% (C3H/HeJ) in the loss regions, and from 0.29% (C3H/HeJ) to 0.83% (LP/J) in the gain regions. (Note that although C3H/HeJ and LP/J suggest a trend for negative correlation between percentage of genome in CNV loss and percentage in CNV gain, in general over all the strains no such correlation is seen (correlation coefficient = −0.3993)). See table 4.2 for all such statistics for each strain.

### EF locates more loss CNVs than gain CNVs

EF locates more loss CNVs than gain CNVs. This is more likely due to the nature of the ROMA data, which has a much lower density of positive $\log_2$ ratios than near zero or negative values, than to the method itself. As explained in section 3.4, the low density of positive signal, (and hence the high density of negative signal), is due to SNPs in the restriction endonuclease site causing the removal of fragments from the genome representation, and also due to SNPs in the probes causing reduced hybridisation. The signals from true gain CNVs might well be removed by this SNP interference process, thus causing an increase in the false negative rate of gain CNV calls. Comparisons to SW-ARRAY in section 4.3 show that SW-ARRAY also has problems detecting gain CNVs, hence corroborating the theory that it is the data, rather than the methods, that is the limiting factor in locating gain CNVs.

### EF detects a small percentage of pairwise CNV between strains

The results also suggest that there is only a small amount of pairwise CNV between inbred mouse strains. However it may be the case that larger proportions of the mouse genome are in pairwise CNV, but that while EF is good at locating many small regions of CNV, it is not capable of overcoming the variation in

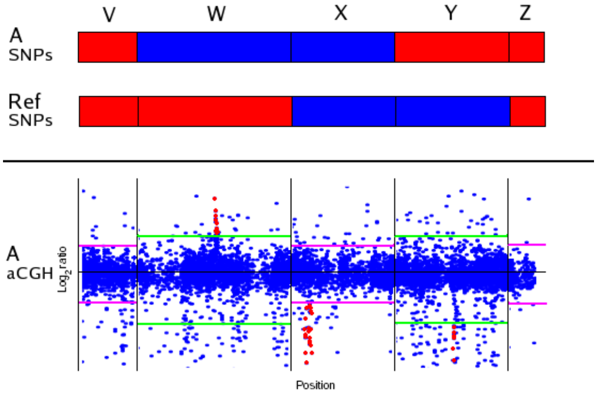

Figure 4.3: Schematic diagram of the EF algorithm. Top: Find reference SNP matched and non-matched regions between the test strain, A, and the reference strain. Here V, X and Z are reference SNP matched, and W and Y are reference SNP non-matched. Bottom: Split the ROMA data for A accordingly. Find upper and lower thresholds in reference SNP matched and non-matched regions (pink in SNP matched, green in SNP non-matched). Locate excursions of probes that exceed the threshold and whose length is significant with reference to a permutation based null distribution of excursion lengths (highlighted in red). (Note that the ROMA data shown is a segment of chromosome 17 from the A/J versus C57BL/6J experiment. However the thresholds and excursions depicted are not real.)

**Permutation algorithm**

For each strain:

1. Find reference SNP matched and non-matched regions.
2. Calculate $(tm_{lower}, tm_{upper})$ and $(tn_{lower}, tn_{upper})$ for the ROMA data from reference SNP matched and non-matched regions respectively.
3. Find all excursions of probes, below the lower thresholds and above the upper thresholds, in ROMA data from reference SNP matched and non-matched regions.
4. Remove from the ROMA data all probes that are members of excursions with length > a "guess threshold".[1]
5. Recalculate $(tm_{lower}, tm_{upper})$ and $(tn_{lower}, tn_{upper})$ for the *remaining* ROMA data from the reference SNP matched and non-matched regions respectively.
6. Repeat the following many times:
    (a) Combine all ROMA data from matched regions.
    (b) Permute
    (c) Split the permuted data into null matched regions that are the same in size and number as the original matched regions (after removal of excursions)
    (d) Repeat steps a, b and c for non-matched regions
    (e) Find all excursions of probes, below the lower thresholds and above the upper thresholds, in ROMA data from null SNP matched and null non-matched regions.
    (f) Store the frequencies of the lengths of excursions found under all four conditions (high and low excursions in reference SNP matched regions and high and low excursions in reference SNP non-matched regions). The total of all of these across all permutations will constitute the four null distributions of excursion lengths for this strain.

**Notes**

1. The "guess threshold" is a very rough estimate of the likely length of an unbroken excursion when there is an underlying CNV. This step is used to remove all $\log_2$ ratios that are actually due to CNVs before permuting for a null distribution of excursion lengths when there are no CNVs.

Box 4.4: Permutation algorithm: an algorithm that estimates an empirical null distribution of excursion lengths.

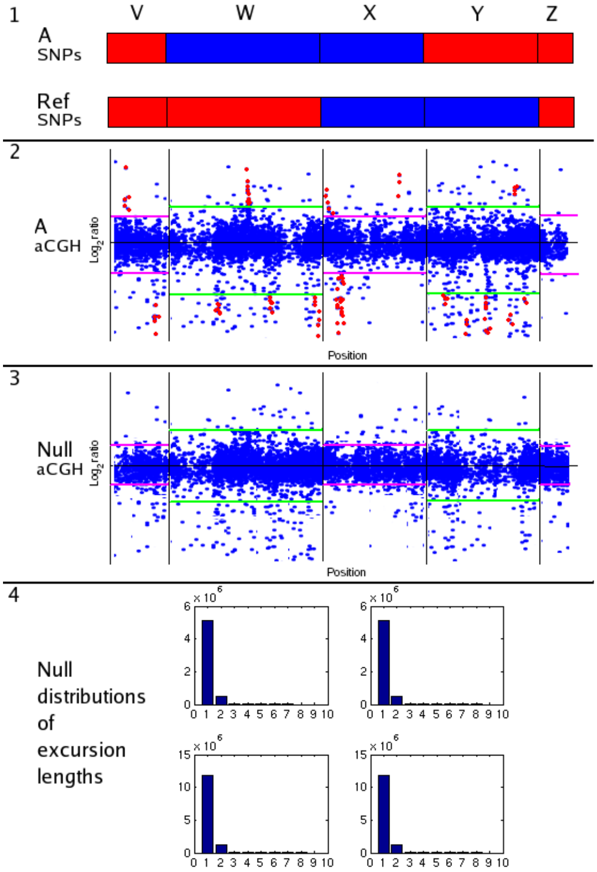

Figure 4.5: Schematic diagram of the permutation algorithm for the null distribution of excursion lengths. 1: Find reference SNP matched and non-matched regions between the test strain, A, and the reference strain. Here V, X and Z are reference SNP matched, and W and Y are reference SNP non-matched. 2: Split the ROMA data for A accordingly. Find upper and lower thresholds in reference SNP matched and non-matched regions (pink in SNP matched, green in SNP non-matched). Locate excursions of probes that exceed the guess threshold (highlighted in red). 3: Remove the excursions from step 2. Permute the ROMA data; this is the null ROMA data. Recalculate thresholds. 4: For both types of region and both types of threshold count the number of excursions of length $1, 2, ..., \max$ where max is the maximum excursion length observed. Repeat the last step for each permutation to estimate the null distribution of excursion lengths.

ROMA data to locate large regions of CNV whose signal is disrupted by outliers. Indeed, EF can only detect large disrupted CNVs as a series of nearby small CNVs with gaps in between them, and these gaps might well represent a large quantity of lost CNV signal. In section 4.3 EF is compared to SW-ARRAY to assess the relative strengths of the two methods in terms of locating small and large regions of CNV.

### Are loss CNVs more commonly found in reference SNP non-matched regions?

| | **EF** | | | | **SW-ARRAY** | | | |
|---|---|---|---|---|---|---|---|---|
| **Strain** | **Loss CNV** | | **Gain CNV** | | **Loss CNV** | | **Gain CNV** | |
| | **Freq** | **% CNV** | **Freq** | **% CNV** | **Freq** | **% CNV** | **Freq** | **% CNV** |
| **A/J** | 772 | 1.02% | 258 | 0.31% | 83 | 4.74% | 4 | 0.03% |
| **AKR/J** | 845 | 1.17% | 662 | 0.77% | 100 | 8.52% | 8 | 0.05% |
| **BALB/cJ** | 668 | 0.93% | 511 | 0.54% | 95 | 4.04% | 2 | 0.02% |
| **C3H/HeJ** | 789 | 1.42% | 680 | 0.29% | 111 | 8.18% | 3 | 0.03% |
| **CBA/J** | 809 | 1.11% | 645 | 0.56% | 75 | 7.27% | 5 | 0.03% |
| **DBA/2J** | 683 | 0.96% | 606 | 0.30% | 114 | 8.90% | 4 | 0.03% |
| **LP/J** | 700 | 0.75% | 588 | 0.79% | 100 | 8.13% | 5 | 0.03% |

Table 4.2: Total number of loss and gain CNVs found (relative to C57BL/6J) in each test strain, by EF (left) and SW-ARRAY (right). The percentage of the C57BL/6J genome in CNV with the test strains is also shown.

The proportion of each test strain genome that is SNP matched to C57BL/6J is given in table 4.3. To assess whether loss CNVs in test strains are more commonly found in reference SNP non-matched regions than in reference SNP matched regions, the following proportions are calculated for each test strain:

- The proportion of the SNP matched reference genome identified as loss CNV in the test strain.
- The proportion of the SNP non-matched reference genome identified as loss CNV in the test strain.

(Where *SNP matched reference genome* corresponds to that part of the reference genome that is SNP matched to the test strain in question. Similarly for the *SNP non-matched reference genome*.)

Table 4.4 gives the resultant proportions. A one sided $t$-test rejects the null hypothesis ($p = 4.6e^{-5}$) that the mean proportions are the same, in favour of the alternative that *the mean proportion of the* ***SNP non-matched*** *reference genome identified as loss CNV is greater than the mean proportion of the* ***SNP matched*** *reference genome identified as loss CNV.*

| **Strain** | **% SNP matched to C57BL/6J** |
|---|---|
| **A/J** | 30.41 |
| **AKR/J** | 39.58 |
| **BALB/cJ** | 38.65 |
| **C3H/HeJ** | 33.73 |
| **CBA/J** | 32.29 |
| **DBA/2J** | 30.55 |
| **LP/J** | 31.28 |

Table 4.3: Proportion of each test strain genome SNP matched to C57BL/6J.

The corresponding proportions for gain CNVs are given in table 4.5. A two sided $t$-test does not reject the hypothesis that the mean proportions are the same ($p$ value of 0.54). So there is no evidence to suggest that gain CNVs are more commonly found in reference SNP matched regions than in reference SNP non-matched regions.

| **Strain** | **% of SNP match identified as loss CNV** | **% of SNP non-match identified as loss CNV** |
|---|---|---|
| **A/J** | 0.41% | 1.29% |
| **AKR/J** | 0.93% | 1.32% |
| **BALB/cJ** | 0.67% | 1.10% |
| **C3H/HeJ** | 0.64% | 1.82% |
| **CBA/J** | 0.41% | 1.45% |
| **DBA/2J** | 0.32% | 1.24% |
| **LP/J** | 0.37% | 0.93% |

Table 4.4: Proportion of the SNP matched reference genome (left), and SNP non-matched reference genome (right), identified as loss CNV in each test strain.

| **Strain** | **% of SNP match identified as gain CNV** | **% of SNP non-match identified as gain CNV** |
|---|---|---|
| **A/J** | 0.24% | 0.33% |
| **AKR/J** | 0.93% | 0.66% |
| **BALB/cJ** | 0.59% | 0.51% |
| **C3H/HeJ** | 0.30% | 0.28% |
| **CBA/J** | 0.62% | 0.54% |
| **DBA/2J** | 0.25% | 0.31% |
| **LP/J** | 1.00% | 0.70% |

Table 4.5: Proportion of the SNP matched reference genome (left), and SNP non-matched reference genome (right), identified as gain CNV in each test strain.

The results for loss CNVs might be caused by a positional association of SNPs and loss CNVs (that is, that they occur on the same parts of the genome). Alternatively, if SNPs are interfering with the ROMA method by one of the mechanisms described in section 3.4, the results might be due to a, potentially, increased false positive rate in the reference SNP non-matched regions.

## 4.2 Combining regions of CNV across strains

The false positive rate (FPR) of EF needs to be assessed. In the absence of a "gold standard" genome and/or experimental verification of putative regions of CNV this poses a hard problem. At the time of writing, FISH, PCR and MLPA experiments are under way to assess the method, but no results are yet available. Therefore the following is carried out here as a first, somewhat coarse, approach to quantifying the FPR.

By combining putative regions of CNVs into CNV sets (CNVSs), which are composed of CNVs located in similar positions on different strains, it is possible to find CNVs that are only found on one strain (*singleton* CNVSs), and highlight these as more likely false positives. The main premise behind this approach is that the more strains on which a CNV appears, the more evidence there is for it, and the more likely it is to be real; so singleton CNVSs are the most likely candidates for false positives because there is less evidence for them.

The combing algorithm developed for this task is depicted schematically in figure 4.6 and outlined in box 4.7.

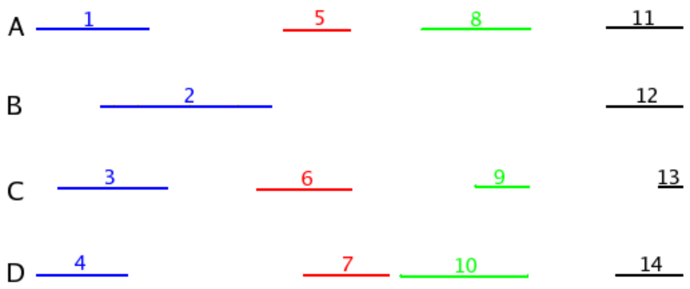

Figure 4.6: Schematic of the CNV combining algorithm. First find a set of CNVSs such that in every CNVS each member CNV overlaps all other member CNVs, and such that all CNVs appear in at least one CNVS. Here there are four such CNVSs, blue (made of CNVs 1, 2, 3 and 4 in strains A, B, C and D respectively), red (5, 6, 7, on A, C, D), green (8, 9 , 10 on A, C, D) and black (11, 12, 13, 14 on A, B, C, D). The CNVSs are depicted here in order of their start points on the genome. The merging procedure will merge the blue CNVS and the red CNVS because CNVs 2 and 6 overlap. Then the green CNVS will be merged with the new merged blue/red CNVS because strain D has CNVs 7 and 10 very close to one another. The black CNVS will not be merged with the new blue/red/green CNVS because neither of the merging requirements are satisfied.

### 4.2.1 CNVS Results

To find the largest possible set of singleton CNVSs the combining algorithm has been run with a *close* threshold of 0. Loss and gain CNVs have been analysed and grouped separately. The frequency of singleton, double, triple, and so forth, CNVSs are shown in table 4.6.

| **Loss CNVSs** | | | **Gain CNVSs** | | |
|---|---|---|---|---|---|
| **Size** | **Freq** | **# CNVs** | **Size** | **Freq** | **# CNVs** |
| 1 | 1634 | 1634 | 1 | 1310 | 1310 |
| 2 | 435 | 870 | 2 | 290 | 580 |
| 3 | 288 | 864 | 3 | 108 | 324 |
| 4 | 140 | 560 | 4 | 37 | 148 |
| 5 | 99 | 495 | 5 | 6 | 30 |
| 6 | 75 | 450 | 6 | 2 | 12 |
| 7 | 44 | 308 | 8 | 1 | 8 |
| 8 | 2 | 16 | 13 | 1 | 13 |
| 9 | 3 | 27 | 25 | 1 | 25 |
| 10 | 2 | 20 | - | - | - |
| 11 | 2 | 22 | - | - | - |
| **Total** | **2724** | **5266** | | **1756** | **2450** |

Table 4.6: Frequency table of CNVS sizes for loss CNVs (left), and for gain CNVs (right). The number of CNVs included in each CNVS size category is also given (this is just $size * freq$).

1634/5266 (31%) of loss CNVs are not overlapped by CNVs on other strains, and therefore form singleton CNVSs. The remaining 3632 (69%) loss CNVs can be grouped into CNVSs of at least two CNVs. 1310/2450 (53%) of gain CNVs form singleton CNVSs, leaving 1140 (47%) CNVs that can be grouped into bigger CNVSs.

**CNV combining algorithm**

```
For each chromosome:
  1. Initialise an empty list, sets, to hold all sets of CNVs that overlap each other.[1]

  2. For each CNV on the chromosome, found on any strain:
     (a) Add CNV to all sets of CNVs in which all member CNVs overlap the current CNV.
     (b) If no such set exists find all non-empty subsets in which all member CNVs overlap the current CNV. Add these subsets to sets
     (c) If no such subsets exist create a new singleton set consisting of the CNV.


  3. Sort sets in ascending order of the start attribute. This gives the ordered list ordered_sets
  4. Merge the sets in ordered_sets from left to right as follows:
  5. For each set in ordered_sets (except the last):
     (a) Call the current set $X$ and the next set in ordered_sets $Y$.
     (b) If end of $X \geq$ start of $Y$ then merge the sets.[2]
     (c) Else if any strain has a CNV in both $X$ and $Y$, and the CNVs are less than a threshold close apart, then merge the sets.[3]
     (d) Otherwise leave $X$ and $Y$ as separate sets.
```

**Notes**

1. Each CNV in each set has three attributes: *strain*, *cnv_start* and *cnv_end*. Each set of CNVs has three attributes: *start*, *end*, and *cnv_list*. The *start* and *end* of a set are respectively defined as the minimum of all *cnv_start*s in the set, and the maximum of all *cnv_end*s in the set.
2. Merge two sets in *ordered_sets* by removing both the sets from the list and replacing them with one set that has as its *cnv_list* the union of the two original *cnv_list*s. Set *start* and *end* of the set accordingly.
3. The threshold *close* is set heuristically and depends on the desired granularity of the sets. It is useful to think of this threshold in terms of both the distance between probes (and hence the number of probes with no extreme signal that can be overcome to combine two CNVs), and also in terms of the proportion of a CNV set likely to be composed of actual putative CNV, versus the proportion of "glue" regions of genome that have not been found to be implicated in CNV.

Box 4.7: CNV combining algorithm: an algorithm that groups overlapping regions of CNV into CNV sets (CNVSs).

The number of singleton CNVSs, and hence potential false positives, is large. However it is worth noting that, for the loss CNVSs at least, particularly with the strict setting of *close* to 0, the assumption that all singleton CNVSs represent false positives gives a very cautious estimate of the FPR. Thus the proportion of singleton CNVSs that cannot be verified experimentally will provide an improved estimate.

Unfortunately the number of singleton gain CNVSs is extremely large, so it is likely that there are many false positive gain CNVs. As will be shown in the next section, SW-ARRAY also has many problems locating gain CNVs, so either a novel method is required to robustly estimate regions of gain CNVs in this ROMA data set, or a new experimental approach is necessary.

## 4.3 Comparison to SW-ARRAY

SW-ARRAY is used here to analyse the mouse ROMA data. All of the located putative CNVs are available on-line at http://gscan.well.ox.ac.uk/gscan/wwwqtl.cgi, alongside the results from EF.

In Price et al. (2005) a pre-processing step to remove outliers from the data is suggested. However due to the non-normality of the ROMA data, especially in the negative tail of the distribution, a well reasoned and appropriate threshold for outliers is not clear, so in this initial SW-ARRAY analysis outliers have not been removed, and instead the input for the algorithm is just the ROMA data normalised as discussed in section 3.3.2.

Because SW-ARRAY is designed to locate contiguous sequences of predominantly high (low) values, while allowing for small numbers of probes in those sequences that are not high (low), the algorithm has identified runs of probes that are longer than those located by EF. SW-ARRAY finds loss CNVs that are between 4 and 1500 probes long, and gain CNVs that are between 2 and 66 probes. The median probe lengths are 79 and 10 for loss and gain CNVs respectively, and the $95^{th}$ percentiles are 539 and 65 respectively.

Minima, maxima, means and percentiles of the lengths of loss and gain CNVs found by SW-ARRAY are given alongside those located by EF in table 4.1. Observing the mean and percentiles of the distribution of excursion lengths found by the two methods, it can be seen that those located by SW-ARRAY are a lot longer than those found by EF. Again, this is as expected due to SW-ARRAY's ability to overcome more variance in the ROMA data than EF can.

Turning next to the number of excursions found by SW-ARRAY, and the percentage of the genome that they cover (see table 4.2) it becomes apparent that although SW-ARRAY finds less loss CNVs than EF (607 compared to 5266), the percentage of the genome covered by loss CNVs is found to be much higher; so SW-ARRAY finds fewer but longer loss CNVs than EF. Once more, this is as expected since a large CNV reported by SW-ARRAY will most likely be reported as many small nearby CNVs by EF. Interestingly the same behaviour is not seen for gain CNVs; although it is still the case that far fewer regions are found by SW-ARRAY than by EF (only 31 are found across all test strains), the percentage of the C57BL/6J genome found to be covered by them, (0.02% to 0.05% across test strains), is approximately an order of magnitude less than that found to be covered by the EF gain CNVs. It is not clear whether the discrepancy between the two methods is due to false negatives in SW-ARRAY or false positives in EF. However it seems likely that both methods are being affected by the nature of the ROMA data which, as discussed previously, has a low density of positive signal which is bound to cause an increase in the false negative rate in gain CNV calls.

In a final comparison of EF and SW-ARRAY, CNVs have been combined across strains and methods to form new CNVSs (see table 4.7). Out of 607 loss CNVs located by SW-ARRAY only 10 are found to form singleton CNVSs when combined with the EF CNVs. Additionally, only one double and one triple of SW-ARRAY loss CNVs are not overlapped by EF CNVs. Therefore most of the SW-ARRAY loss CNVs are corroborated by EF. Similarly, out of 31 gain CNVs found by SW-ARRAY only 2 form singletons; the rest are overlapped by EF gain CNVs.

Combing SW-ARRAY and EF also corroborates more of the EF loss CNVSs. The number of singleton EF loss CNVSs is reduced from 1634 to 887, thus lowering the putative FPR for EF loss CNVs from 31% to 17% (887/5266). Validation of these 887 CNVs will provide a better estimate of the FPR. It will also be interesting to assess the nature of the 1023 EF loss CNVs that overlap one another in CNVSs of size $2-7$, but are not identified by SW-ARRAY. Such a study will help characterise the relative strengths and weaknesses of the two methods.

Unfortunately the gain CNVs found by SW-ARRAY provide almost no extra evidence for the EF gain CNVs (the singleton EF gain CNVSs are only reduced from 1310 to 1298). Once again it is impossible to determine which of the methods is correct (ie. whether the problem is one of EF false positives or SW-ARRAY false negatives), therefore further analysis is required or the characterisation of gain CNVs in the mouse genome.

| **Loss CNVSs** | | | | | **Gain CNVSs** | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **Size** | **Freq** | **EF** | **SW-ARRAY** | **Both** | **Size** | **Freq** | **EF** | **SW-ARRAY** | **Both** |
| 1 | 897 | 887 | 10 | 0 | 1 | 1300 | 1298 | 2 | 0 |
| 2 | 203 | 186 | 1 | 16 | 2 | 289 | 288 | 0 | 1 |
| 3 | 115 | 97 | 1 | 17 | 3 | 105 | 104 | 0 | 1 |
| 4 | 57 | 36 | 0 | 21 | 4 | 35 | 33 | 0 | 2 |
| 5 | 36 | 19 | 0 | 17 | 5 | 10 | 6 | 0 | 4 |
| 6 | 28 | 12 | 0 | 16 | 6 | 3 | 1 | 0 | 2 |
| 7 | 21 | 7 | 0 | 14 | 8 | 2 | 0 | 0 | 2 |
| 8 | 12 | 0 | 0 | 12 | 13 | 1 | 0 | 0 | 1 |
| 9 | 7 | 0 | 0 | 7 | 51 | 1 | 0 | 0 | 1 |
| 10 | 9 | 0 | 0 | 9 | - | - | - | - | - |
| 11-20 | 56 | 0 | 0 | 56 | - | - | - | - | - |
| 21-30 | 21 | 0 | 0 | 21 | - | - | - | - | - |
| 31-40 | 8 | 0 | 0 | 8 | - | - | - | - | - |
| 41-50 | 5 | 0 | 0 | 5 | - | - | - | - | - |
| $> 51$ | 17 | 0 | 0 | 17 | - | - | - | - | - |
| **Total** | **5944** | **1244** | **12** | **236** | | **2481** | **1730** | **2** | **14** |

Table 4.7: Frequency table of CNVS sizes for loss CNVs (left), and for gain CNVs (right), after combining CNVs across strains and methods. Within each category of CNV (loss and gain), the total frequency for each CNVS size is given, and this is broken down into subtotals for the frequency of that size of CNVS: composed only of CNVs found by EF; composed only of CNVs found by SW-ARRAY; and composed of CNVs found by both methods.

## 4.4 Analysing the relationship between CNVs and QTLs

Using a genetically heterogeneous stock (HS) of mice descended from C57BL/6J and the seven test strains, small effect quantitative trait loci (QTL) have previously been fine-mapped to the C57BL/6J genome (Solberg et al. (2006), Valdar et al. (2006a), Valdar et al. (2006b)). The phenotypes that were studied target three diseases: anxiety, type II diabetes and asthma. Of interest is the relationship between CNVs and QTLs. Therefore a permutation test for the significance of the overlap between QTLs and putative CNVSs (from the combination of EF and SW-ARRAY outputs) is developed here. The algorithm is described in box 4.8 and a schematic of the algorithm is shown in figure 4.9.

### 4.4.1 QTL permutation test results

To reduce the chance of using false positive CNVs in this part of the analysis, the QTL permutation test is carried out using CNVSs that are either:

**QTL permutation test**

```
For each phenotype:
  1. For each QTL:
     (a) Calculate the proportion of the 95% confidence interval that is overlapped by a CNVS.
  2. Calculate the mean CNV coverage across all QTLs for this phenotype.
  3. Initialise $extreme\_overlap = 0$ to store the number of times that bootstrapped QTLs for this phenotype have more overlap by CNVs than the real QTLs.
  4. Repeat N times (where N is large):
     (a) Permute the QTL 95% confidence intervals. No overlaps permitted.
     (b) Go through each bootstrap QTL:
           i. Calculate the proportion of the bootstrap 95% confidence interval that is overlapped by a CNVS.
     (c) Find the mean CNV coverage across all bootstrapped QTLs for this phenotype.
     (d) If the mean is more extreme than that of the real mean overlap then $extreme\_overlap += 1$
  5. Divide $extreme\_overlap$ by N to give the permutation based $p$-value for this phenotype's QTLs.
```

Box 4.8: QTL permutation test: an algorithm for calculating the amount of QTL overlapped by CNVs, and for calculating the significance of this overlap.

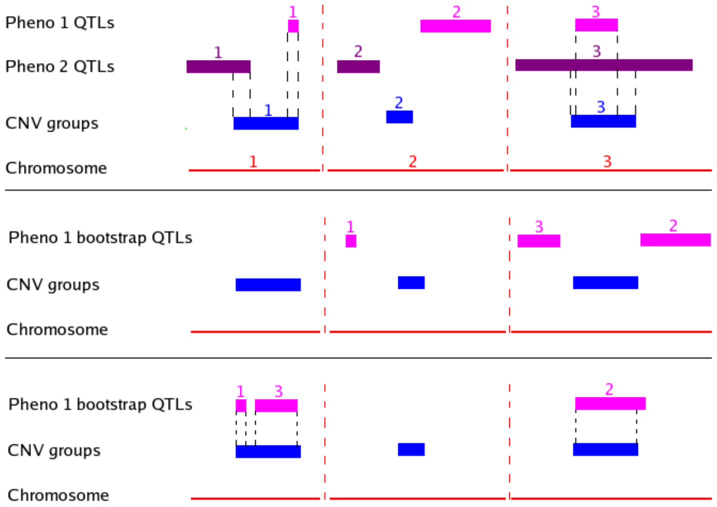

Figure 4.9: Schematic of the QTL permutation test. Top: QTLs for phenotype 1 (pink), QTLs for phenotype 2 (purple), CNVSs (blue), spread over 3 chromosomes (red). Overlap of QTLs by CNVSs are highlighted with black dashed lines. Some overlap by CNVSs is seen for both sets of QTLs. Middle: the first iteration of the permutation test for Phenotype 1 QTLs. Note that all three QTLs have been repositioned, and that the CNVSs have remained in their original position. In this iteration there is less overlap by CNVSs for the bootstrap QTLs than there was for the real QTLs, so the *extreme_overlap* variable is not incremented. Bottom: the second iteration of the permutation test for Phenotype 1 QTLs. This time more overlap is seen for the bootstrapped QTLs than for the real QTLs, so *extreme_overlaps* is incremented by 1. When all the permutations are finished for phenotype 1 the permutation test will be repeated for phenotype 2, etc.

- composed of CNVs found in at least two different strains or methods;
- or have at least three member CNVs.

Since the median distance between probes is $\sim 5$ Kb, 25% of the time 10 Kb will be enough to merge two CNVSs that are separated by one probe (because there is a probability of 0.5 that one inter CNV space is $\geq 5$ Kb, and a separation by one probe counts as two inter-CNV spaces on either side of it). Furthermore, looking at the distribution of lengths of CNVs found by EF and SW-ARRAY, it is clear that an inter-CNV space of 10 Kb in a CNVS is very unlikely to be larger than the true CNVs being merged. Therefore the merging threshold *close* has been set to a conservative 10 Kb.

| | **Phenotype** | **% Mean overlap of QTL** | **Related to Phenotypes** |
|---|---|---|---|
| 1 | Biochem: Creatinine | 49% | 2, 5 |
| 2 | Biochem: Urea | 40% | 1 |
| 3 | PM: Open Arm Distance | 37% | 4, 6 |
| 4 | PM: Open Arm Entries | 42% | 3, 6, 7, 10 |
| 5 | EPM: Open Arm Latency | 38% | 6 |
| 6 | EPM: Open Arm Time | 41% | 3, 4, 6, 7 |
| 7 | Haem: MCV | 40% | 4, 6 |
| 8 | Haem: MPV | 49% | 9, 10 |
| 9 | Haem: WBC | 38% | 8 |
| 10 | Imm: PctCD3 | 35% | 4, 8 |

Table 4.8: Phenotypes with QTLs which are significantly overlapped by CNVs at the 5% level.

Out of 96 phenotypes the QTL permutation test finds 10 with QTLs which are significantly overlapped by CNVs at the 5% level (see table 4.8). This is a little more than expected by chance. However, some of the phenotypes are related to one another and have overlapping QTLs, thus reducing the set to $\sim 5$ truly differing phenotypes. Therefore, thus far there is no evidence for an association between CNVs and QTLs.

## 4.5 Analysing the relationship between CNVs and eQTLs

Treating gene expression as a phenotype, expression QTLs (eQTLs) have also been identified for the HS mice and mapped to C57BL/6J (work not published). In a final analysis, the positional relationship between CNVs and eQTLs is examined here.

A total of 3295 eQTLs have been mapped to the mouse genome. The eQTLs are grouped according to their log $p$ values, and within each group the number of eQTLs in CNVSs versus the number outside CNVSs is recorded. The contingency table is given in table 4.9.

| log $\boldsymbol{p}$ | **eQTLs in CNVSs** | **eQTLs outside CNVSs** | **Totals** |
|---|---|---|---|
| 0-49 | 756 | 1971 | **2727** |
| 50-99 | 133 | 240 | **373** |
| 100-149 | 45 | 86 | **131** |
| $> 150$ | 28 | 36 | **64** |
| **Totals** | **962** | **2333** | **3295** |

Table 4.9: Contingency table of eQTLs observed in CNVSs and eQTLs observed outside CNVSs.

Using Pearson's $\chi$-square test, with 3 degrees of freedom, the null hypothesis that the eQTLs are not positionally related to CNVs is tested. A $\chi$-squared statistic of 18.6371 is obtained, giving a a $p$-value of 0.0003249. Therefore there is strong evidence to reject the null hypothesis in favour of the alternative that **eQTLs are positionally related to CNVs**.

# Chapter 5

# Future Work

To bring this section of the project to completion three data sets must be incorporated into the analysis of the putative CNVs located by EF and SW-ARRAY. The data sets are:

- A high density array-based SNP sequencing data set, produced by Perlegen (http://www.perlegen.com), for fifteen mouse strains, five of which are in the ROMA experiment set of eight. The arrays cover the non-repetitive fraction of the C57BL/6J genome as a series of 25-mers (each assayed base position has a 25-mer centered on it). First, the genome of each strain was amplified using long-range PCR in 10kb sections. These were then hybridised to the arrays, and allele-calling software was used to call the SNPs. Identifying runs of PCR sequencing failures in this data will enable the location of putative deletions **relative to C57BL/6J** which are not visible in the ROMA data.

- An HMM analysis of the genotyping errors in the eight inbred mouse strains, kindly conducted by Gil McVean in the Department of Statistics, University of Oxford. Briefly, an HMM was used to infer strain mosaic structure for the eight mouse strains from the HS mice (for which the eight inbred strains are the founding strains) (Mott et al., 2000). Next, places on the genome where the genotype observed was different to that predicted by the inferred strain structure were identified (allowing for errors in the inference of the strain structure). All such 'errors' were then pulled together across mice, and SNPs where there were a large number of errors were identified. Such errors may be due to CNVs and might be useful for corroboration of the putative CNVs located by EF and SW-ARRAY.

- The mouse CNV set published by Graubert et al. (2007). This work represents the most recent, and only other very high throughput, analysis of CNV in the mouse genome. The experiments use C57BL/6J as the reference strain, and five of the seven test strains from the ROMA experiment are analysed in the paper, so it will be possible to make a direct comparison. In addition, since CBS was used in the aCGH analysis carried out in Graubert et al. (2007), it will be useful to use CBS to analyse the ROMA data, and to compare the results to those published in the paper. It will also be informative to analyse their raw data using EF and SW-ARRAY and to compare the highlighted CNVs to the published results. Finally, it will be useful to integrate the data from Graubert et al. (2007) with the ROMA data set. This will produce a mouse aCGH data set with higher density than any other mouse aCGH data currently available. The data will be analysed for CNVs, and the results compared to those obtained from the two data sets individually. Importantly, the technique could potentially be generalised to a wide variety of high throughput data sets.

Clearly, it will also be necessary to assess the putative CNVs using the results of verification experiments currently being conducted by Binnaz Yalcin at the Wellcome Trust Centre for Human Genetics (see sections 4.2, 4.2.1 and 4.3, for discussions of how the experimental data will be used). The verification techniques used for putative CNVs depend on their size and whether they are gain or loss CNVs, and include PCR, fluorescent in situ hybridisation (FISH) and multiplex ligation-dependent probe amplification (MLPA Schouten et al. (2002)).

As a corollary to this work it will be constructive to review the nature of the data produced by the very high throughput aCGH methods now emerging. In particular it will be helpful to compare the distribution of the aCGH data from Graubert et al. (2007) to that of the ROMA data analysed here. Of primary interest is a comparison of the tails of the distributions, which are extremely imbalanced in the ROMA data. A review, based on the literature review given in chapter 2, of the applicability of current aCGH analytical methods to such high throughput methods, will also be informative.

In the next phase of the project modifications and extensions of the methods described in this report will be explored:

- SW-ARRAY will be extended to support the integration of SNP data. Referring to section 2.8, it is possible to solve eq.(2.41), by numerical analysis, separately for ROMA data from reference SNP matched and non-matched regions. Furthermore it is possible to choose a parameter $\tau$, again by numerical analysis, to subtract from the ROMA data in the SNP matched regions, such that the solution to eq.(2.41) is the same for this adjusted ROMA data as it is for the ROMA data in the SNP non-matched regions. Then, with reference to eq.(2.42), the mean maximal segment score, $\bar{S}(n)$, will have very similar distributions in ROMA data from both types of region (except for the factor $K$, which is harder to account for). Thus, after adjusting the ROMA data in the SNP matched regions by $\tau$, it will be possible to use the SW-ARRAY algorithm across the whole data set, as described in section 2.8.4.
- EF will be modified to incorporate the SNP data in a different way, such that its power to detect CNVs will increase by combining data across mouse strains. The SNP genotypes will be used to infer phylogenetic trees of the mouse strains which, due to the mosaic nature of the mouse strains, change along the length of the genome. For each phylogeny the mouse strains will be grouped according to their similarity to C57BL/6J. Then the average signal across all strains that are different to C57BL/6J will be calculated, and thresholds set accordingly, before searching for runs of average $\log_2$ ratios that exceed the thresholds. There are many open questions in this process. Two hard ones are: How should the phylogenetic trees be inferred?; and how should the trees be divided into C57BL/6J similar/dissimilar strains?

In the field of bioinformatics high throughput sequence analysis problems are commonplace, and often involve data which are from a continuous but otherwise uncharacterised distribution, have a high variance and, importantly, may have related data or knowledge that can explain some of the variance in the signal. Therefore it is desirable to complete this project with a Bayesian nonparametric approach to segmentation of high density, high variance sequence data. A nonparametric method would be advantageous because it would not require a functional form for the distribution of the data. Furthermore, it would render the method readily generalisable to any sequence data, even if the underlying distribution of the data was known. The Bayesian aspect of the approach would enable the coherent introduction of external data sets. So for example, in the case of the ROMA data set, a prior probability of CNV in a particular mouse strain could be derived from the SNP based phylogenies discussed above. Alternatively the prior probability could be based on the evidence for CNV in other test strains. Finally, such an approach would provide a coherent method for placing probabilities on detected events, so in the case of the ROMA data it would provide posterior probabilities for detected CNVs.

# Bibliography

A J Aguirre, C Brennan, G Bailey, R Sinha, B Feng, C Leo, Y Zhang, J D Gans, N Bardeesy, C Cauwels, C Cordon-Cardo, M S Redston, R A DePinho, and L Chin. High-resolution characterisation of the pancreatic adenocarcinoma genome. *Proceedings of the National Academy of Sciences*, 101:9067–9072, 2004.

D Albertson and D Pinkel. Genomic microarrays in human genetic disease and cancer. *Human Molecular Genetics*, 12:R145–R152, 2003.

M T Barrett, A Scheffer, A Ben-Dor, N Sampas, D Lipson, R Kincaid, P Tsang, B Curry, K Baird, P S Meltzer, Z Yakhini, L Bruhn, and S Laderman. Comparative genomic hybridization using oligonucleotide microarrays and total genomic DNA. *Proceedings of the National Academy of Science*, 101: 17765–17770, 2004.

Y Benjamini and Y Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society B*, 57:289–300, 1995.

M Bredel, C Bredel, D Juric, G R Harsh, H Vogel, L D Recht, and B I Sikic. High-resolution genome-wide mapping of genetic alterations in human glial brain tumors. *Cancer Research*, 65:4088–4096, 2005.

The International HapMap Consortium. A haplotype map of the human genome. *Nature*, 437:1299–1320, 2005.

R J de Leeuw, J J Davies, A Rosenwald, G Bebb abd R D Gascoyne, M J Dyer, L M Staudt, J A Martinez-Climent, and W L Lam. Comprehensive whole genome array CGH profiling of mantle cell lynphoma model genomes. *Human Molecular Genetics*, 13:1827–1837, 2004.

P H C Eilers and R X de Menezes. Quantile smoothing of array CGH data. *Bioinformatics*, 21:1146–1153, 2004.

L Feuk, AR Carsin, and S W Scherer. Structural variation in the human genome. *Nature Reviews*, 7: 85–97, 2006.

J Fridlyand, A M Snijders, D Pinkel, D G Albertson, and A N Jain. Hidden markov model approach to the analysis of array CGH data. *Journal of Multivariate Analysis*, 90:132–153, 2004.

T A Graubert, P Cahan, D Edwin, R R Selzer, T A Richmond, P S Eis, W D Shannon, X Li, H L McLeod, J M Cheverud, and T J Ley. A high resolution map of segmental DNA copy number variation in the mouse genome. *Public Library of Science Genetics*, 3:21–29, 2007.

S Guha, Y Li, and D Neuberg. Bayesian Hidden Markov Modeling of array CGH data. Technical Report 24, Harvard School of Public Health, 2006.

G Hodgson, J F hager, S Volik, S Hariono, M Wernick, D Moore, D G Albertson, D Pinkel, C Collins, D Hanahan, and J W Gray. Genome scanning with array CGH delineates regional alterations in mouse islet carcinomas. *Nature Genetics*, 29:459–464, 2001.

L Hsu, S G Self, D Grove, T Randolph, K Wang, J J Delrow, L Loo, and P Porter. Denoising array-based comparative genomic hybridization data using wavelets. *Biostatistics*, 6:211–226, 2005.

P Hupe, N Stransky, J Thiery, FRadvanyi, and E Barillot. Analysis of array cgh data: from signal ratio to gain and loss of dna regions. *Bioinformatics*, 20:3413–3422, 2004.

K Jong, E Marchiori, A van der Vaart, B Ylstra, M Weiss, and G Meijer. Chromosomal breakpoint detection in human cancer. *Lecture Notes in Computer Science*, 2611:54–65, 2003.

S Karlin and S F Altschul. Methods for assessing the statistical significance of molecular subsequence. *Journal of Molecular Biology*, 147:195–197, 1990.

L Kaufman and P J Rousseeuw. *Finding groups in data: an introduction to cluster analysis*. Wiley, 1990.

R W Koenker and G W Basset. 4 (pathological) examples in asymptotic statistics. *American Statistician*, 38:209–212, 1984.

D Komura, F Shen, S Ishikawa, K R Fitch, W Chen, J Zhang, G Liu, S Ihara, H Nakamura, M E Hurles, C LOee, S W Scherer, K W Jones, M H Shapero, J Huang, and H Aburatani. Genome-wide detection of human copy number variations using high-density DNA oligonucleotide arrays. *Genome Research*, 16:1575–1584, 2006.

W R Lai, M D Johnson, R Kucherlapati, and P J Park. Comparative analysis of algorithms for identifying amplifications and deletions in array CGH data. *Bioinformatics*, 21:3763–3770, 2005.

B Lakshmi, I M Hall, C Egan, J Alexander, A Leotta, J Healy, L Zender, M S Spector, W Xue, S W Lowe, M Wigler, and R Lucito. Mouse genomic representational oligoneucleotide microarray analysis: Detection of copy number variations in normal and tumor specimens. *Proceedings of the National Academy of Science*, 103:11234–11239, 2006.

M Lavielle. Using penalized contrasts for the change-point problem. *Signal Processing*, 85:1501–1510, 2005.

E Lebarbier. Detecting multiple change-points in the mean of gaussian process by model selection. *Signal Processing*, 85:717–736, 2005.

J Li, T Jiang, J Mao, A Balmain, L Peterson, C Harris, P Rao, P Havlak, R Gibbs, and W Cai. Genomic segmental polymorohisms in inbred mouse strains. *Nature Genetics*, 36:952–954, 2004.

O C Lingjaerde, L O Baunbusch, K Liestol, I K Glad, and A-L Borresen-Dale. CGH-explorer: a program for analysis of array-CGH data. *Bioinformatics*, 21:821–822, 2005.

L W M Loo, D I Grove, C L Neal, L A Cousens, E L Schubert, E M Williams, I N Holcomb, J J Delrow, B J Trask, L Hsu, and P L Porter. Array-CGH analysis of genomic alterations in breast cancer subtypes. *Cancer Research*, 64:8541–8549, 2004.

R Lucito, J Healy, J Alexander, A Reiner, D Esposito, MChi, L Rodgers, A Brady, J Sebat, J Troge, J A West, S Rostan, K C Q Nguyen, S Powers, K Q Ye, A Olshen, E Venkatraman, L Norton, and M Wigler. Representational oligonucleotide microarray analysis: A high resolution method to detect genome copy number variation. *Genome Research*, 13:2291–2305, 2003.

J Lupski and L White. Genome based arrays. Website: http://mrrc.pedi.bcm.tmc.edu/cores/genomearrays.html.

A Molinaro, M van der Laan, and D Moore. Comparative genomic hybridization array analysis. Technical report, Division of Biostatistics, University of California, Berkeley, 2002.

R Mott and R Tribe. Approximate statistics of gapped alignments. *Journal of Computational Biology*, 6:91–112, 1999.

R Mott, C J Talbot, M G Turri, A C Collins, and J Flint. A method for fine mapping quantitative trait loci in outbred animal stocks. *Proceedings of the National Academy of Science*, 97(23):12649–12654, 2000.

K Nakao, K R Mehta, J Fridlyand, D H Moore, A N Jain, A Lafuente, J W Wiencke, J P Terdiman, and F M Waldman. High resolution analysis of DNA copy number alterations in colorectal cancer by array-based comparative genomic hybridsation. *Carcinogenesis*, 25:1345–1357, 2004.

A B Olshen and E S Venkatraman. Change-point analysis of array-based comparative genomic hybridization data. *American Statistical Association Proceedings of the Joint Statistical Meetings*, pages 2530–2535, 2002.

A B Olshen and E S Venkatraman. Circular binary segmentation for the analysis of array-based DNA copy number data. *Biostatistics*, 5:557–572, 2004.

F Picard, S Robin, M Lavielle, C Vaisse, and J-J Daudin. Chromosomal breakpoint detection in human cancer. *BMC Bioinformatics*, 6:27, 2005.

J R Pollack. Microarray analysis reveals a major and direct role of DNA copy number alteration in the transcriptional program of human breast tumors. *Proceedings of the National Academy of Science, USA*, 99:12963–12968, 2002.

S Portnoy and R W Koenker. The gaussian hare and the laplacian tortoise: computability of squared-error versus absolute-error estimators. *Statistical Science*, 12:279–296, 1997.

T S Price, R Regan, R Mott, A Hedman, B Honey, R J Daniels, L Smith, A Greenfield, A Tiganescu, V Buckle, N Ventress, H Ayyub, A Salhan, S Pedraza-Diaz, J Broxholme, J Ragoussis, D R Higgs, J Flint, and J L Knight. SW-ARRAY: a dynamic programming solution for the identification of copy-number changes in genomic DNA using array comparative genome and hybridization data. *Nucleic Acids Research*, 33:3455–3464, 2005.

L R Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77:257–286, 1989.

R Redon, S Ishikawa, K R Fitch, L Feuk, G H Perry, T D Andrews, H Fiegler, M H Shapero, A R Carson, W Chen, E K Cho, S Dallaire, J L Freeman, J R Gonzalez abd M Gratacos, J Huang, D Kalaitzopoulos, D Komura, J R MacDonald, C R Marshall, R Mei, L Montgomery, K Nishimura, K Okamura, F Shen, M J Somerville, J Tchinda, A Valsesia, C Woodwark, F Yang, J Zhang, T Zerjal, J Zhang, L Armengol, D F Conrad, X Estivill, C Tyler-Smith, N P Carter, H Aburatani, C Lee, K W Jones, S W Scherer, and M E Hurles. Global variation in copy number in the human genome. *Genome Research*, 7118:444–454, 2006.

J P Schouten, C J McElgunn, R Waaijer, D Zwijnenburg, F Diepvens, and G Pals. Relative quantification of 40 nucleic acid sequences by multiplex ligation-dependent probe amplification. *Nucleic Acids Research*, 30, 2002.

A Sen and M S Srivastava. On the tests for detecting a change in mean. *Anals of Statistics*, 3:98–108, 1975.

S P Shah, X Xuan, R J de Leeuw, M Khojasteh, W L Lam, R Ng, and K P Murphy. Integrating copy number polymorphisms into array CGH analysis using a robust HMM. *Bioinformatics*, 22:e431–e439, 2006.

T F Smith and M S Waterman. Identification of molecular subsequences. *Journal of molecular biology*, 147:195–197, 1981.

A Snijders, N Nowak, B Huey, J Fridyland, S Law, J Conroy, T Tokuyasu, K Demir, R Chiu, J Mao, and D Albertson. Genomic segmental polymorohisms in inbred mouse strains. *Nature Genetics*, 36: 952–954, 2004.

A M Snijders, N Nowak, R Segraves, S Blackwood, N Brown, J Conroy, G Hamilton, A K Hindle, B Huey, and K Kimura. Assembly of microarrays for genome-wide measurement of DNA copy number. *Nature Genetics*, 29:263–264, 2001.

L C Solberg, W Valdar, D Gauguier, G Nunez, A Taylor, S Burnett, C Arboledas-Hita, PHernandez-Pliego, S Davidson, P Burns, S Bhattacharya amd T Hough, D Higgs, P Klenerman, W O Cookson, Y Zhang, R M Deacon, J N Rawlins, R Mott, and J Flint. A protocol for high-throughput phenotyping, suitable for quantitative trait analysis in mice. *Mammalian Genome*, 17:129–146, 2006.

J D Storey. A direct approach to false discovery rates. *Journal of the Royal Statistical Society B*, 64: 479–498, 2002.

B E Stranger, M S Forrest, M Dunning, C E Ingle, C Beazley, N Thorne, R Redon, C P Bird, A de Grassi, C Lee, C Tyler-Smith, N Carter, S W Scherer, S Tavare, P Deloukas abd M E Hurles, and E T Dermitzakis. Relative impact of nucleotide and copy number variation on gene expression phenotypes. *Science*, 315:848–853, 2007.

W Valdar, L C Solberg, D Gauguier, S Burnett, P Klenerman, W O Cookson, M S Taylor, J N Rawlins, R Mott, and J Flint. Genome-wide genetic association of complex traits in heterogeneous stock mice. *Nature Genetics*, 38:879–887, 2006a.

W Valdar, L C Solberg, D Gauguier, W O Cookson, J N Rawlins, R Mott, and J Flint. Genetic and environmental effects on complex traits in mice. *Genetics*, 174:959–984, 2006b.

E S Venkatraman. Consistency results in multiple change-point situations. Technical report, Department of Statistics, Stanford University, 1992.

C M Wade, E J Kulbokas III, A W Kirby, M C Zody, J C Mullikin, E S Lander, K Lindblad-Toh, and M J Daly. The mosaic structure of variation in the laboratory mouse genome. *Nature*, 420:574–578, 2002.

P Wang, Y Kim, J Pollack, B Narasimhan, and R Tibshirani. A method for calling gains and losses in array CGH data. *Biostatistics*, 6:45–58, 2005.

E Whittaker. On a new method of graduation. *Proceedings Edinburgh Mathematical Society*, 41:63–75, 1923.

H Willenbrock and J Fridlyand. A comparison study: applying segmentation to array CGH data for downstream analyses. *Bioinformatics*, 21:4084–4091, 2005.