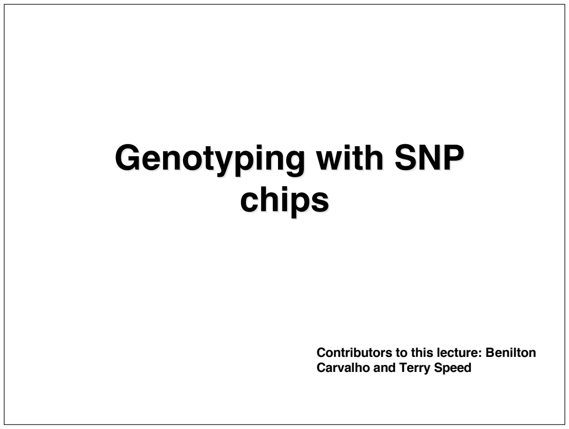
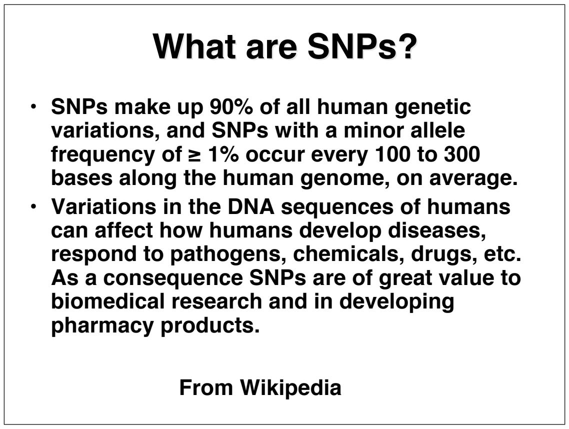


Genotyping with SNP chips

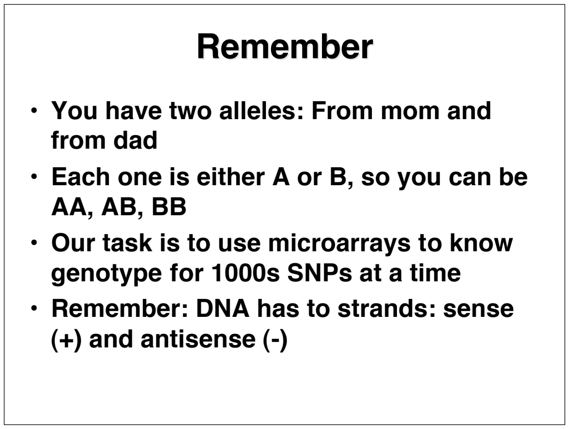
Contributors to this lecture: Benilton Carvalho and Terry Speed



What are SNPs?

- SNPs make up 90% of all human genetic variations, and SNPs with a minor allele frequency of ≥ 1% occur every 100 to 300 bases along the human genome, on average.
- Variations in the DNA sequences of humans can affect how humans develop diseases, respond to pathogens, chemicals, drugs, etc. As a consequence SNPs are of great value to biomedical research and in developing pharmacy products.

From Wikipedia



Remember

- You have two alleles: From mom and from dad
- Each one is either A or B, so you can be AA, AB, BB
- Our task is to use microarrays to know genotype for 1000s SNPs at a time
- Remember: DNA has to strands: sense (+) and antisense (-)

The Affymetrix genotyping microarray

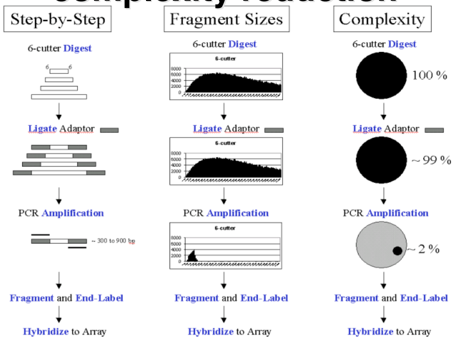
Whole Genome Sampling Assay

1. **Fractionate total genomic DNA with a restriction enzyme (e.g. XBaI)**
2. **Ligate a single generic adaptor to the ends of all fragments**
3. **Use the generic adaptor as primer pair to carry out the PCR, amplifying fragment sizes (250 bp - 2,000 bp) such that the**

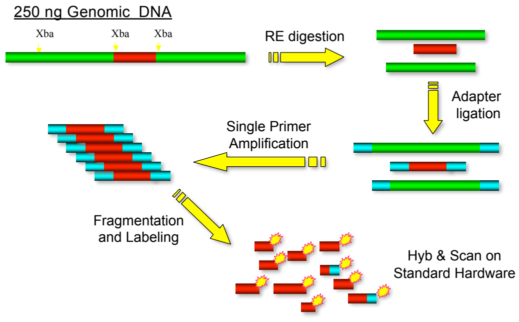
- **PCR is reliable and reproducible, and the**
- **Total PCR product is small enough to hybridize efficiently**

4…. Fragment, label, hybridize, stain, wash, scan, analyse image, then analyse data to call genotypes (our task).

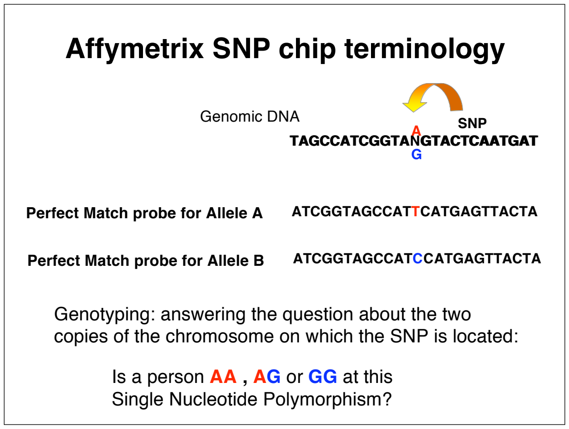
Single primer assay: complexity reduction



Single primer assay: overview



250 ng Genomic DNA
Xba
Xba
Xba
RE digestion
Adapter ligation
Single Primer Amplification
Fragmentation and Labeling
Hyb & Scan on Standard Hardware



Affymetrix SNP chip terminology

Genomic DNA

SNP

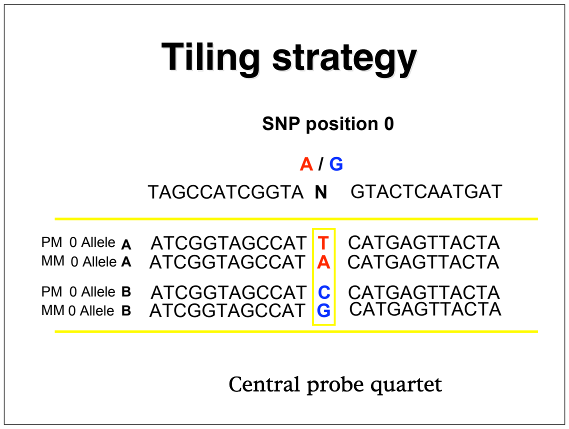
TAGCCATCGGTA$\overset{A}{\underset{G}{N}}$GTACTCAATGAT

Perfect Match probe for Allele A ATCGGTAGCCATTCATGAGTTACTA

Perfect Match probe for Allele B ATCGGTAGCCATCCATGAGTTACTA

Genotyping: answering the question about the two copies of the chromosome on which the SNP is located:

Is a person **AA** , **AG** or **GG** at this
Single Nucleotide Polymorphism?



Tiling strategy

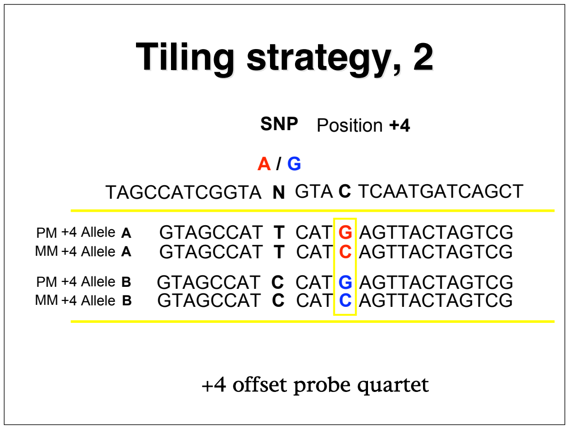
SNP position 0

A / G

TAGCCATCGGTA **N** GTACTCAATGAT

PM 0 Allele **A**	ATCGGTAGCCAT	**T**	CATGAGTTACTA
MM 0 Allele **A**	ATCGGTAGCCAT	**A**	CATGAGTTACTA
PM 0 Allele **B**	ATCGGTAGCCAT	**C**	CATGAGTTACTA
MM 0 Allele **B**	ATCGGTAGCCAT	**G**	CATGAGTTACTA

Central probe quartet



Tiling strategy, 2

SNP Position **+4**

A / G

TAGCCATCGGTA **N** GTA **C** TCAATGATCAGCT

PM +4 Allele **A**	GTAGCCAT	**T**	CAT	**G**	AGTTACTAGTCG
MM +4 Allele **A**	GTAGCCAT	**T**	CAT	**C**	AGTTACTAGTCG
PM +4 Allele **B**	GTAGCCAT	**C**	CAT	**G**	AGTTACTAGTCG
MM +4 Allele **B**	GTAGCCAT	**C**	CAT	**C**	AGTTACTAGTCG

+4 offset probe quartet

In summary: probe level data

- Two alleles
- Two directions
- Two types (PM,MM)
- Up to 7 locations of the SNP in the probe

Affymetrix SNP probe tiling strategy, 3

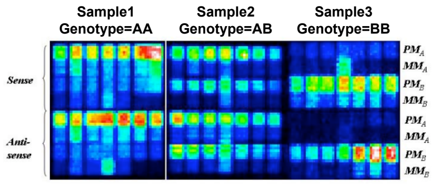
Offset quartets Central quartet Offset quartets

1	2	3	4	5	6	7
PMA	PMA	PMA	PMA	PMA	PMA	PMA
MMA	MMA	MMA	MMA	MMA	MMA	MMA
PMB	PMB	PMB	PMB	PMB	PMB	PMB
MMB	MMB	MMB	MMB	MMB	MMB	MMB

Repeated on the opposite strand: 56 probes for 10K. More recently, 40: just 4 offset quartets instead of 6.

Probe Intensities

Fake (idealized) image for 3 samples on one SNP



Fake, as the probes are not all adjacent on the chip
Idealized, as all the probes are high or low as they should be.

Calling genotypes: A modular approach

MPAM: the first Affymetrix SNP-calling algorithm, used on the 10K SNP chip

Generalities concerning MPAM

- **Derive a reasonable though ad hoc summary statistic, here RAS (feature extraction)**
- **Clusters the statistic in a sensible way, here using MPAM (classification)**
- **Generates new calls by cluster membership, here using elliptical regions, cf. bivariate normal (modelling).**

Ref: Liu, WM *et al*, *Bioinformatics* Dec 2003

MPAM: detection filter

$i \in \{S,T\}$ Sense or anTisense strand
$j \in \{A,B\}$ allele
$k \in \{1,\ldots,7\}$ position of interrogation

$$D_{ijk} = (PM_{ijk} - MM_{ijk}) / (PM_{ijk} + MM_{ijk})$$

$$D_{ij} = \mathrm{median}(D_{ijk})$$

$$D = \max(\min(D_{SA},D_{TA}), \min(D_{SB},D_{TB}))$$

SNPs with low D (<0.03) are not called.

MPAM: feature extraction

i ∈ {S,T} Sense or anTisense strand (also +, - or 1,2)
j ∈ {A,B} allele
k ∈ {1,…,7} position of interrogation

$$MM_{ik} = (MM_{iAk} + MM_{iBk})/2$$

$$s_{ijk} = \max(PM_{ijk} - MM_{ik,}0)$$

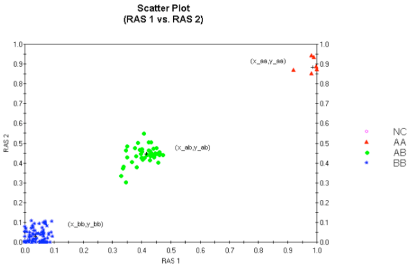
s_{ik} = Relative Allele Signal of kth quartet of strand i

$$s_{ik} = s_{iAk} / (s_{iAk} + s_{iBk})$$

s_i = Relative Allele Signal of strand i

$$s_i = \text{median}(s_{ik})$$

Clustering and modeling



SNP ID: 15749 Individuals Called: 133 Clusters: 3 Silhouette Score: 0.892

MPAM: classification algorithm

- **Partitioning Around Medoids PAM**
Kauffman and Rousseeuw, 1987

- **Work with Relative Allele Signal RAS (s_S, s_T), 2-dim feature space from both forward and reverse strands**

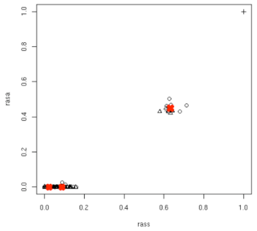
- **n points in feature space: $x_1, x_2, \ldots, x_n$**
Assuming there are k = 2 and 3 groups
minimize $f_{PAM} = \sum_{i=1}^{n}(\min(d(x_i, m_t), t=1:k)$

- **MPAM (modified PAM): minimize**
$f_{MPAM} = f_{PAM} - l \sum_{j=1}^{k}(\min(d(x_a, x_b), x_a \in G_j, x_b \notin G_j)$

Difference between PAM and MPAM

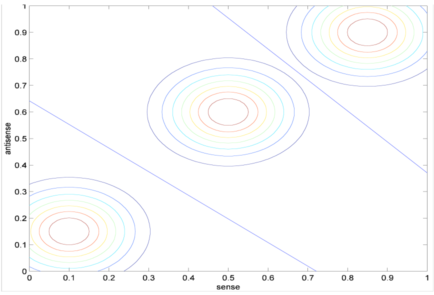
The result of using PAM with 3 groups on the data for one SNP

The penalty used on MPAM is designed to avoid just this situation.



Plot courtesy of Chris Neff

Genotyping using robust models



MPAM Classification quality metrics

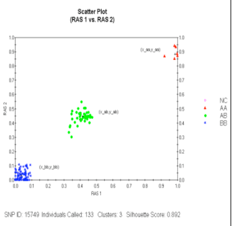
Silhouette width for x_i

a(i) is the av w/i group distance to x_i

b(i) is the av bet group distance to x_i

w(i) = (b(i) – a(i))/max(b(i), a(i))

w = average{w(i): i=1…n}.

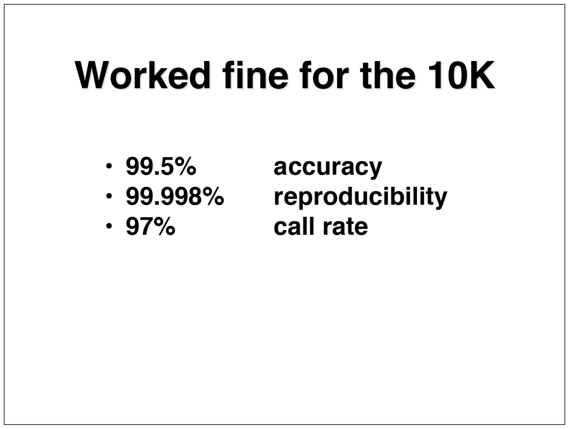


Separation of the groups

sep$_f$=median{lx_aa-x_abl, lx_ab-x_bbl}

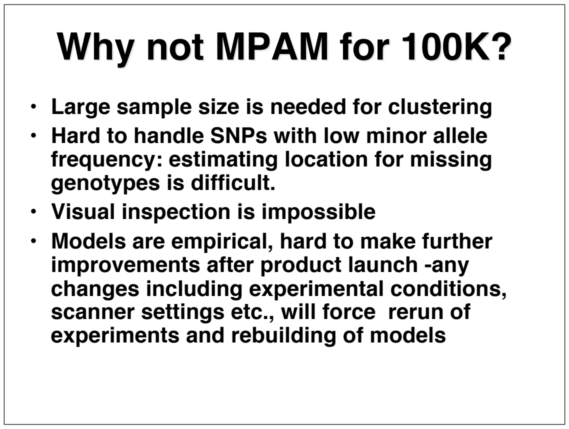
sep$_r$=median{ly_aa-y_abl, ly_ab-y_bbl}

sep = min{sep$_f$,sep$_r$},



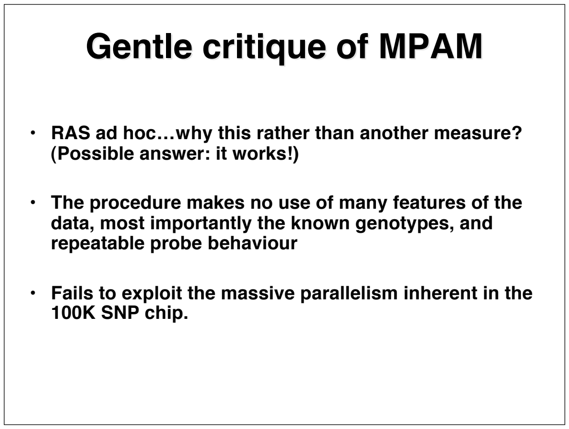
Worked fine for the 10K

• 99.5%	accuracy
• 99.998%	reproducibility
• 97%	call rate



Why not MPAM for 100K?

- Large sample size is needed for clustering
- Hard to handle SNPs with low minor allele frequency: estimating location for missing genotypes is difficult.
- Visual inspection is impossible
- Models are empirical, hard to make further improvements after product launch -any changes including experimental conditions, scanner settings etc., will force rerun of experiments and rebuilding of models



Gentle critique of MPAM

- RAS ad hoc…why this rather than another measure? (Possible answer: it works!)
- The procedure makes no use of many features of the data, most importantly the known genotypes, and repeatable probe behaviour
- Fails to exploit the massive parallelism inherent in the 100K SNP chip.

Unified approach: the Dynamic model-based algorithm, DM

Until recently the vendor-supplied genotype-calling algorithm. Seeks the best fitting pattern of the above kind, including no call (NC). It is a mix of normal likelihood-based model selection and a Wilcoxon test, leading to a final *p*-value which is a form of confidence statement about the call.

There is no training, and it is a single chip procedure.

However, the SNPs on the chip have been selected so that the algorithm works on them.

DM

- **Look at quartets individually and produce a score under normal theory assumptions**
- **Combine scores across quartets to produce a classification into genotypes (resistant to cross-hybridization and model failure)**
- **Provides a "p-value"/goodness of classification metric**

Ref: Di, X. *et al*, *Bioinformatics* May 2005

Likelihood, intensity scale, for each quartet

$$\prod_{i\in\{\mathrm{A,C,G,T}\}}\prod_{j=1}^{n_i}\frac{1}{\sqrt{2\pi\sigma_i^2}}\exp\left\{-\frac{1}{2}\left(\frac{x_{i,j}-\mu_i}{\sigma_i}\right)^2\right\}$$

n_i = number of pixels for feature *i*; $x_{i,j}$ = measured value of pixel *j* μ_i = unknown mean pixel intensity; σ_i = unknown SD of pixel intensities, all for feature $i \in \{A,C,G,T\}$, *x'* denotes reverse strand.

Null model (B for background)

$\mu_A = \mu_C = \mu_G = \mu_T = \mu_B$; $\mu'_A = \mu'_C = \mu'_G = \mu'_T = \mu_B$

Illustrative homozygote model: CC (S for signal)

$\mu_C = \mu_S$; $\mu_A = \mu_G = \mu_T = \mu_B$; $\mu'_C = \mu_S$; $\mu'_A = \mu'_G = \mu'_T = \mu_B$

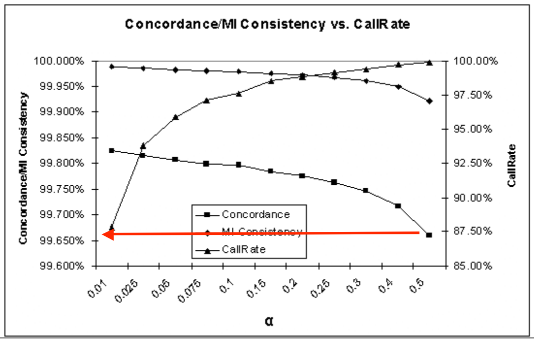
Illustrative heterozygote model: CT

$\mu_C = \mu_T = \mu_S$; $\mu_A = \mu_G = \mu_B$; $\mu'_C = \mu'_T = \mu_S$; $\mu'_A = \mu'_G = \mu_B$

DM: combining quartet-level information

- Start with *N* probe quartets q_i *i=1,...,N,* N typically 10 or 14
- For each probe quartet q_i evaluate log-likelihood *LL* of the 4 possible models:
  - *LL(AA,i), LL(AB,i), LL(BB,i), LL(NC,i), NC=No Call*
- For each probe quartet, transform log-likelihoods to scores:
  - *s(AA,i) = LL(AA,i) – max{LL(m,i), m ≠ AA}*
  - *s(AB,i), s(BB,i), s(N,i)* computed similarly
- Combine quartet-level results to a SNP-level result:
  - for each model *m* ∈ *{AA,AB,BB,NC}* use Wilcoxon signed rank test on *{s(m,i); i=1,…,N}*
  - Yields 4 *p*-values, the call and score for the SNP corresponds to the model with the most significant *p*-value

DM on 30 CEPH trios: HapMap Concordance & Mendelian Inheritance



Why attempt an improvement over DM?

- Perhaps the error rate is too high?
- There is reason to believe it can be improved by
  - a) using the training/test set paradigm;
  - b) carrying out multi-chip analyses, which identify and exploit probe behaviour; and
  - c) exploiting the massive parallelism across SNPs.
- The 100K SNPs were selected from a much larger screening set using DM. For the 500K and >1M SNP chips, a higher yield is desirable, and perhaps a better genotype-calling algorithm could achieve this.

Robust Linear Model with the Mahalanobis distance classifier

- RLMM pronounced “REALM”
- Based on an RMA-like model
  - Uses PM only
  - Linear additive multi-chip model on log scale
  - A- and B-probe and chip effects
  - Robustly estimated parameters
- Classification using Mahalanobis’ distance
- Morphed into BRLMM; CRLMM coming up!

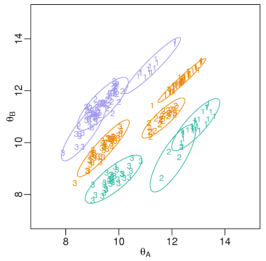
Notation

- Once we are done with first part of preprocessing we have the following:

θ_A and θ_B proportional to log of the amount of fragments from allele A and B respectively

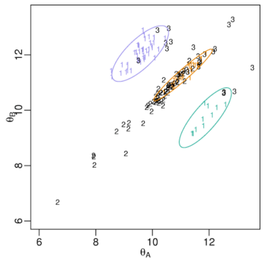
In principal these can only be (log of) 0, x, or 2x, but we know better than to believe this.. In fact we know not to expect the same cut-off to work for all SNPs

It’s not easy



This picture shows that most the information is in the left right diagonal direction, i.e. in the log-ratios

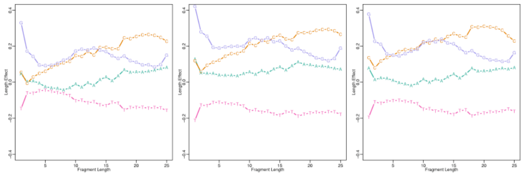
Lab Effect

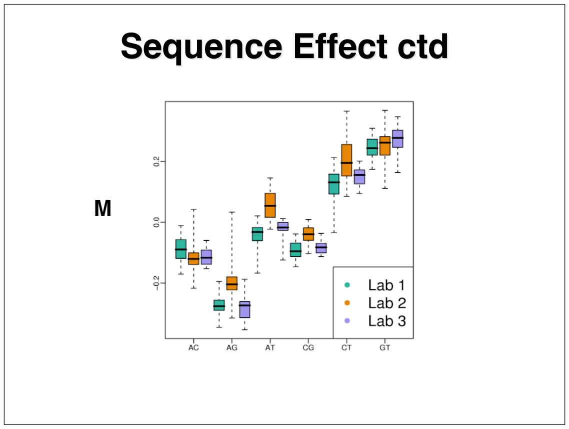


Why is this?

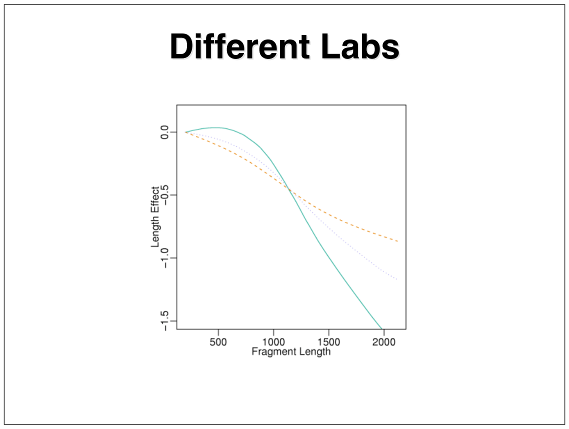
- Our guess is that the PCR step introduces a lot of SNP to SNP variation
- We have proxies for measuring PCR effect: fragment sequence and fragment length
- We can examine the fragment sequence via the probe sequence

Sequence effect

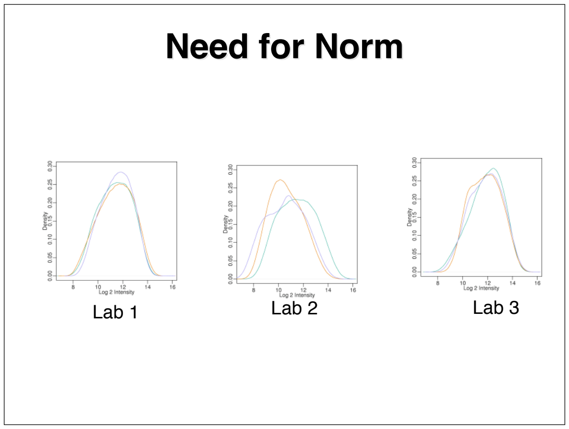




Sequence Effect ctd



Different Labs

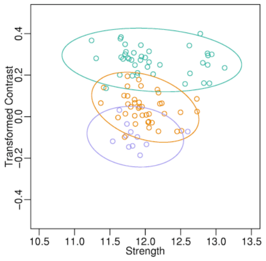


Need for Norm

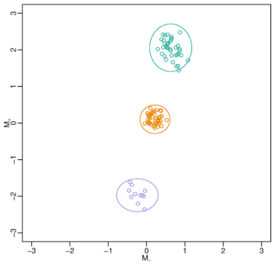
Normalization

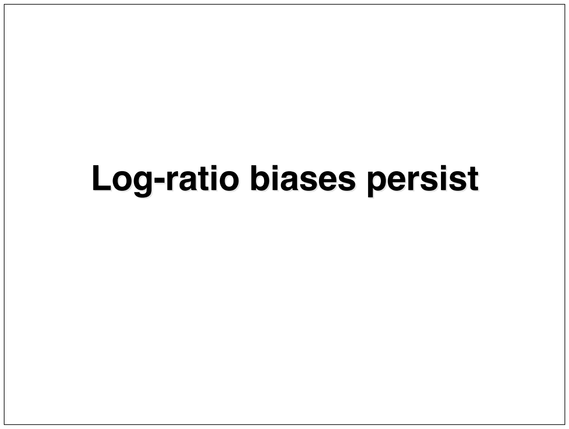
- We normalize/summarize using RMA (no BG correction) after correcting for sequence and length effects on the log intensities
- We then examine log-ratios
- We keep sense and antisense separate

“Broken” probes (BRLMM)

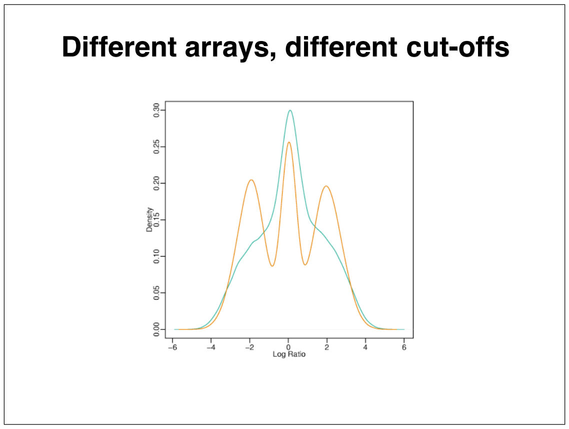


“Broken” probes?

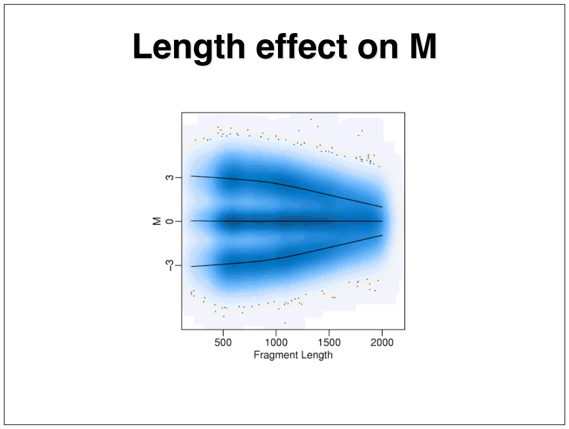




Log-ratio biases persist

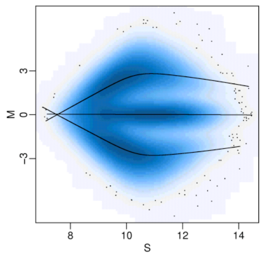


Different arrays, different cut-offs



Length effect on M

Intensity effect on M

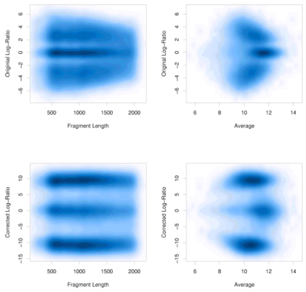


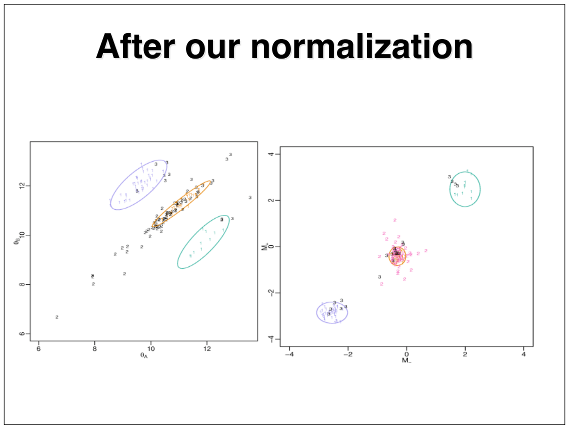
Use mixture model to fix this

$$[M_i|Z_i = k] = f_k(X_i) + \varepsilon_{i,k}$$

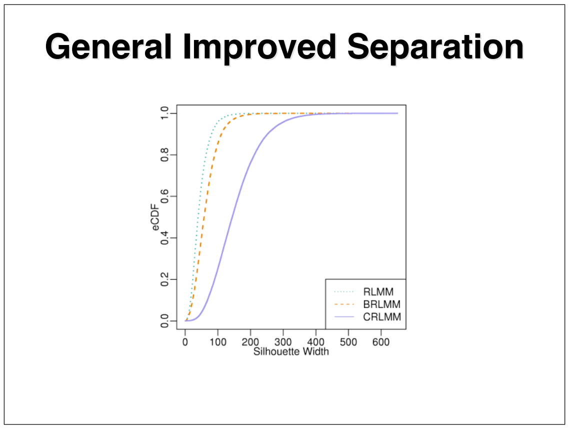
- SNP denoted with I
- Z is true, so k = AA, AB or BB
- X are covariates that cause bias

After fix

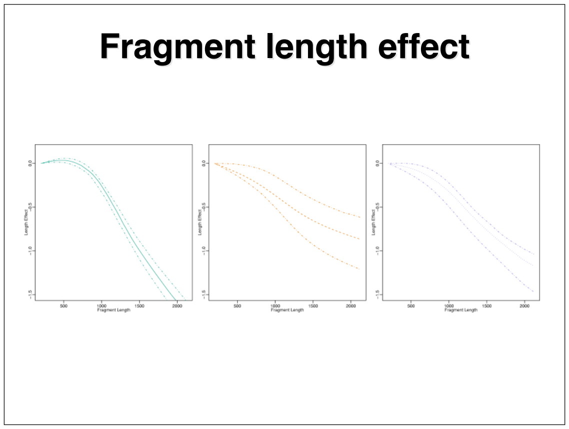




After our normalization

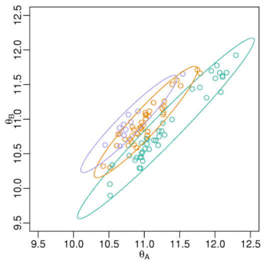


General Improved Separation



Fragment length effect

“Broken” probes (RLMM)



Preprocessing model motivates genotype algorithm

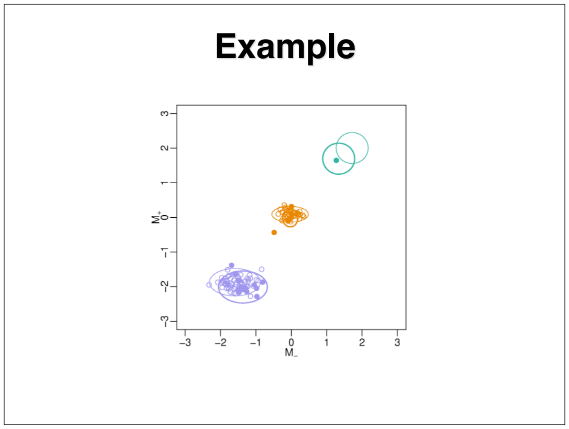
$$[M_{i,j,s}|Z_{i,j} = k, m_{i,k,s}] = f_{j,k}(X_{i,j,s}) + m_{i,k,s} + \varepsilon_{i,j,k,s}.$$

- Array denoted with j
- Shift in cluster center denoted with m
- Assume m are bivairate normal and $\frac{1}{\sigma^2_{i,k,s}} \propto \frac{1}{d_{0,k}s^2_{0,k}}\chi^2_{d_{0,k}}$
- Use training data to estimate
- Use empirical bayes approach for cases with few data points

Predicting regions

$$\tilde{\mathbf{m}}_i = (V^{-1} + \mathbf{N}_i\Sigma^{-1})^{-1}\mathbf{N}_i\Sigma^{-1}\hat{\mathbf{m}}_i$$

$$\tilde{\sigma}^2_{i,k,s} = \frac{(N_{i,k}-1)\hat{\sigma}^2_{i,k,s} + d_{0,k}s^2_{0,k}}{(N_{i,k}-1) + d_{0,k}}, \text{ for } N_{i,k} > 1.$$



Example