

# CANCER GENOMICS

## Lecture 3:

# Probabilistic Methods for Profiling Copy Number Alterations

**GENOME 541**  
Spring 2020



**Gavin Ha, Ph.D.**  
Public Health Sciences Division  
Human Biology Division



@GavinHa



gha@fredhutch.org



<https://github.com/GavinHaLab>

[GavinHaLab.org](http://GavinHaLab.org)

# Outline

---

## 1. Detecting Copy Number Alterations in Cancer Genomes

- Predicting copy number features from sequence data
- Copy number analysis workflow
- Data normalization

## 2. Continuous Hidden Markov Model (HMM)

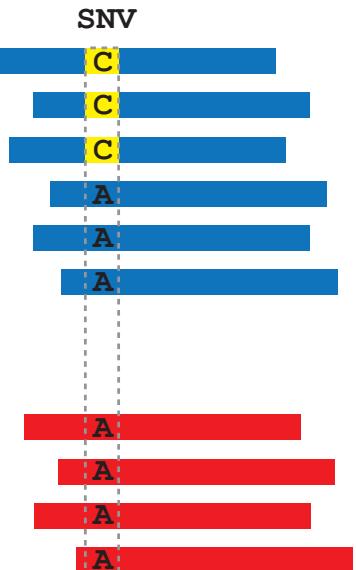
- Graphical model representation
- Components of a continuous HMM
- Inference & parameter estimation using expectation-maximization (EM)

## 3. Copy Number Profiling using a Hidden Markov Model

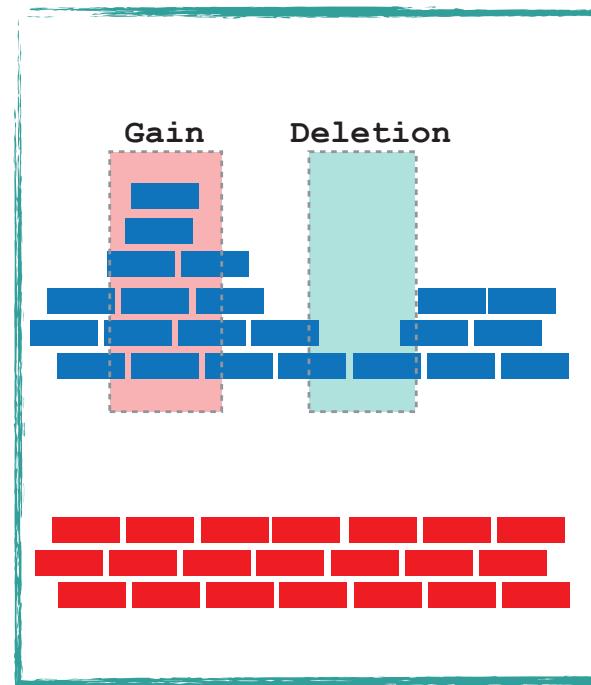
- Probabilistic model for copy number analysis
- Predicting copy number segments using the Viterbi algorithm

# 1. Detecting Copy Number Alterations in Cancer Genomes

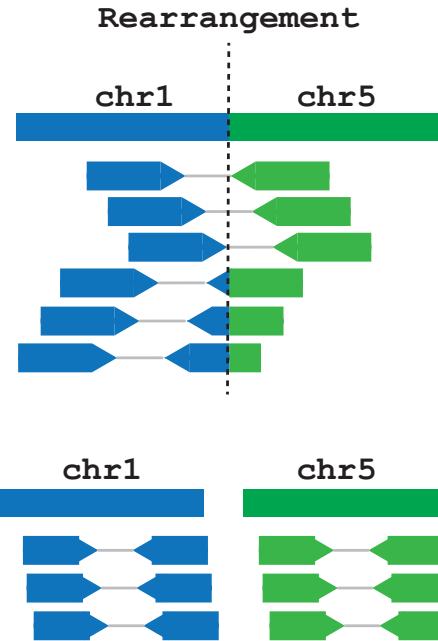
## Mutations (SNV, INDEL)



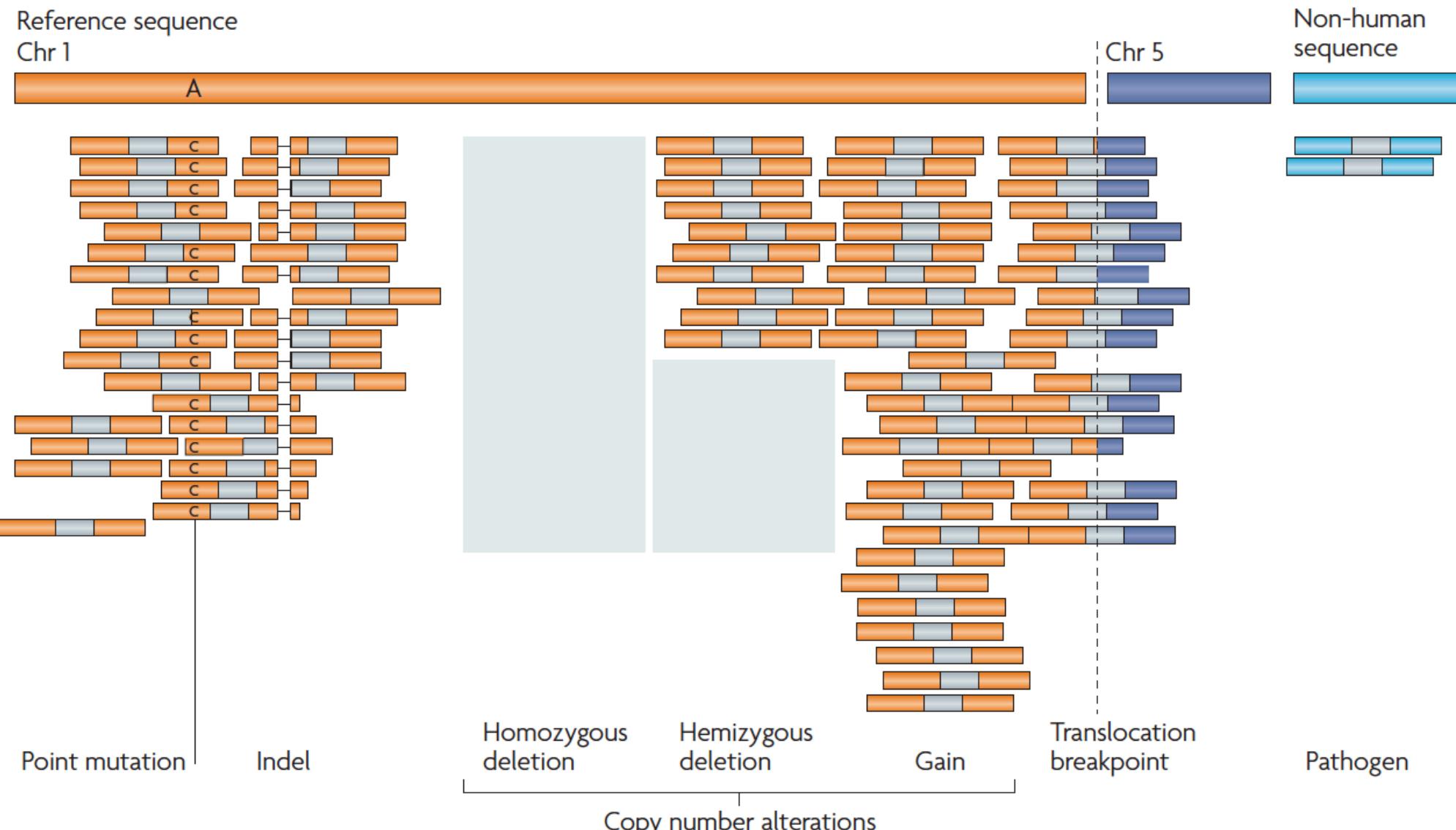
## Copy Number Alterations



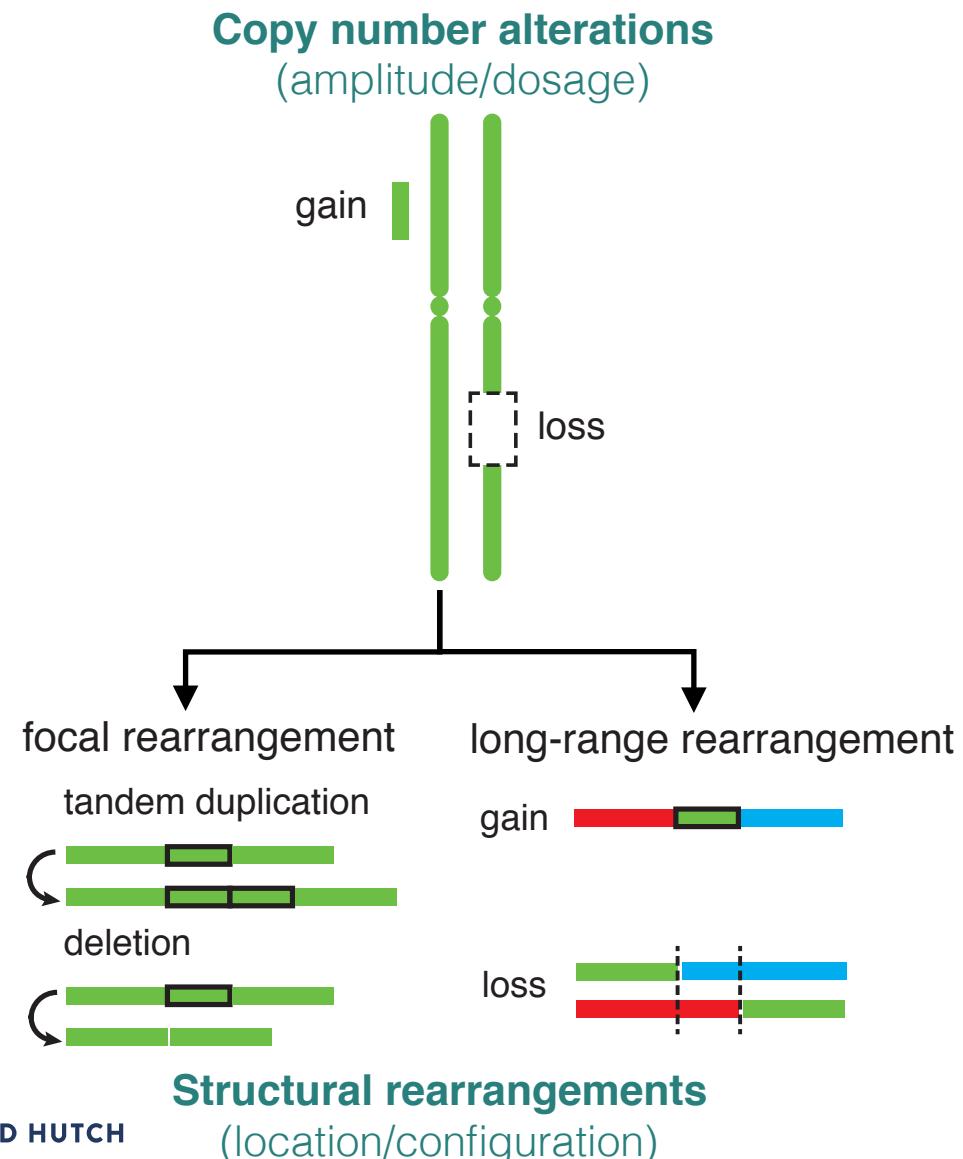
## Structural Variants



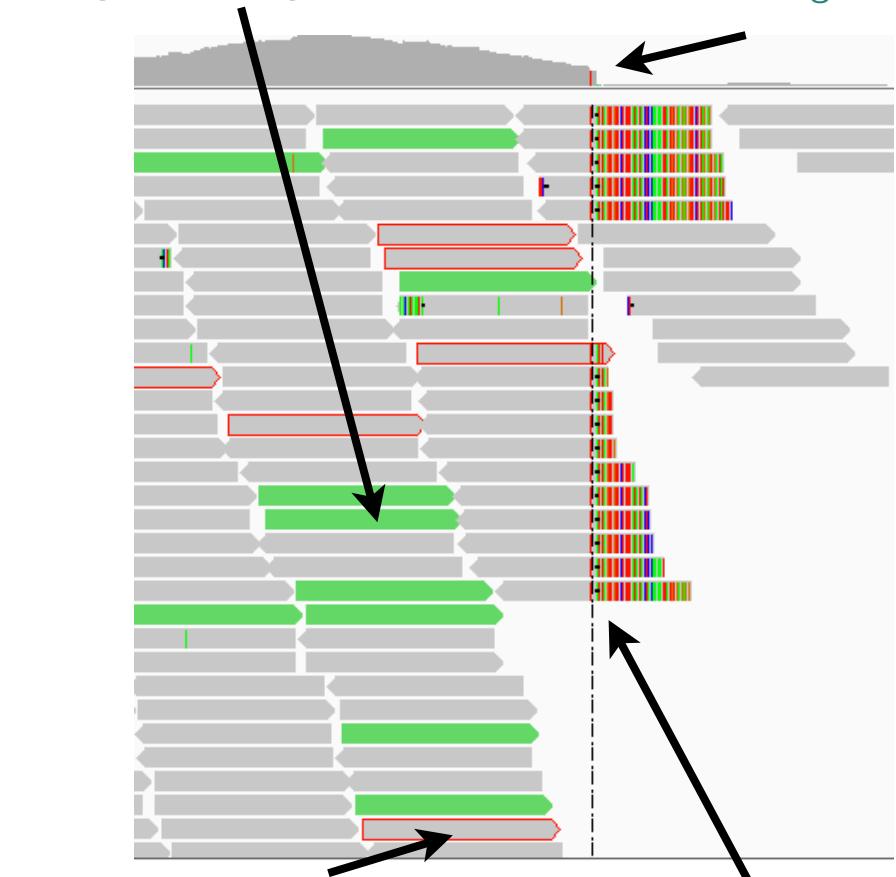
# Predicting genomic alterations from sequence data



# Predicting genomic alterations from sequence data



**"discordant read pair"**  
read pairs with aberrant inferred fragment length

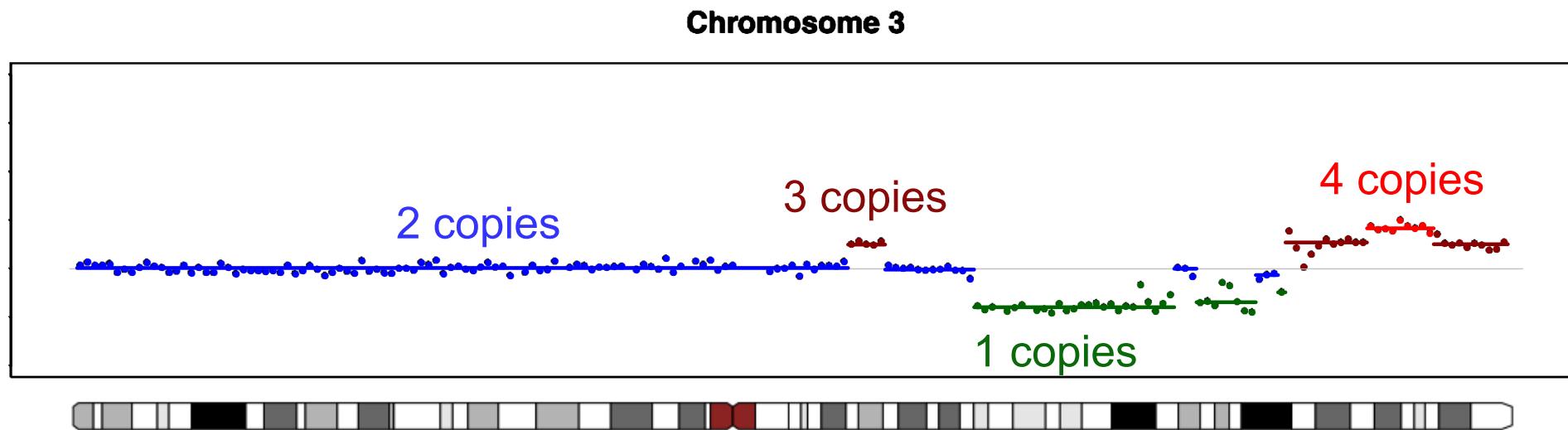


**"copy number change"**  
abrupt change in read coverage

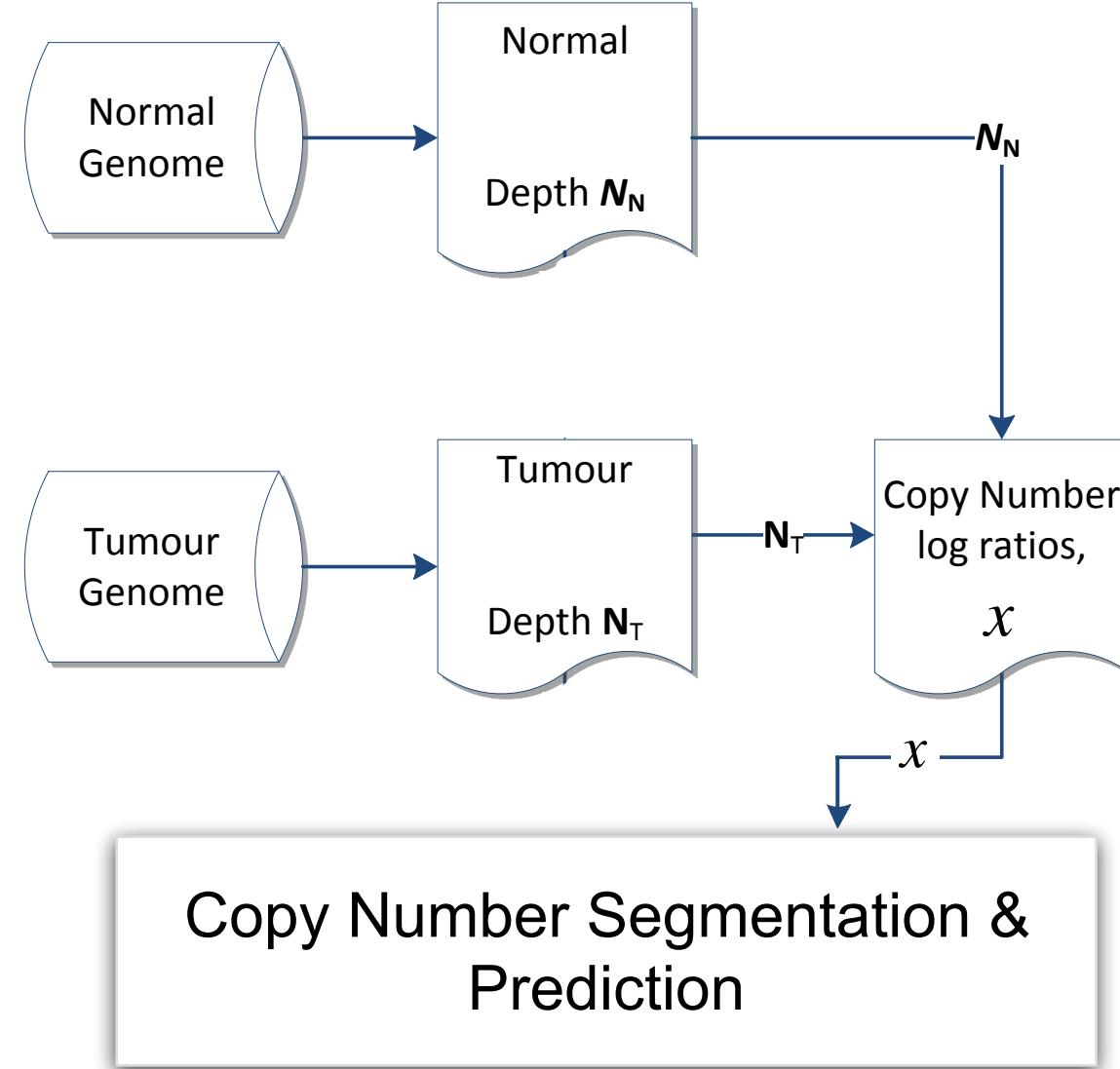
# Tumor DNA Copy Number Analysis Strategy

---

1. Using sequencing read coverage as a measure for DNA copy number
2. Identifying segments of coverage changes
3. Predicting the number of copies for each segment

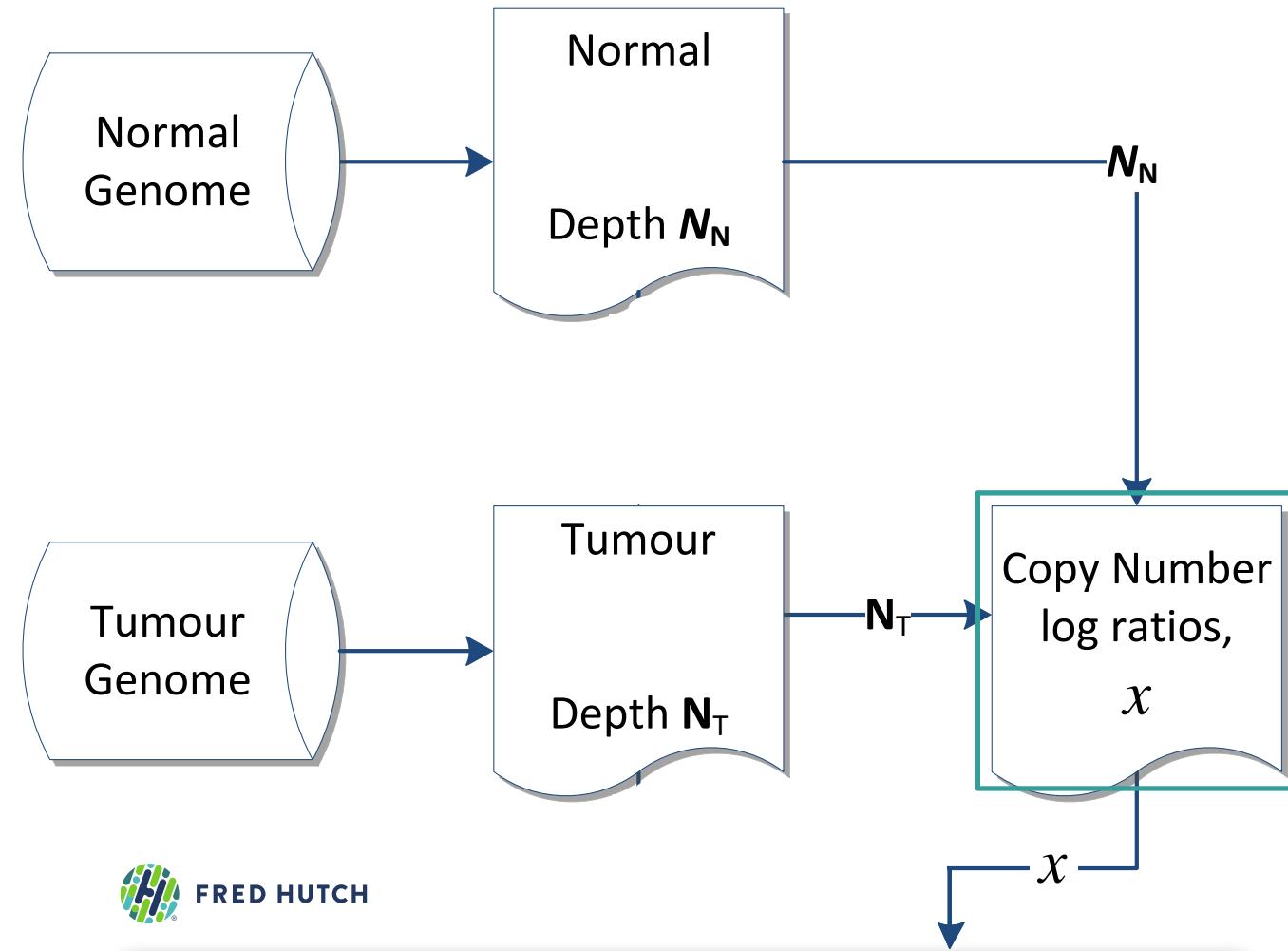


# Cancer Genome Copy Number Analysis Workflow



# Copy Number Analysis Workflow: Normalization

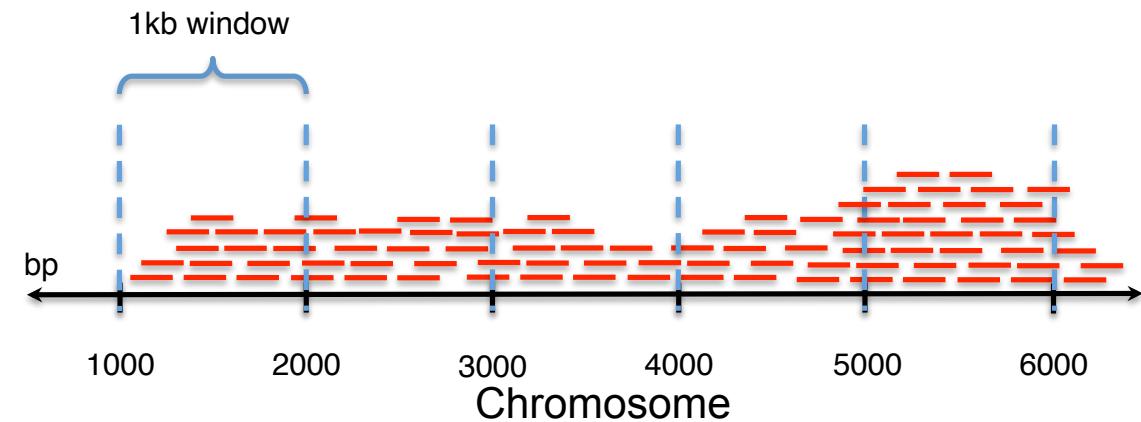
## 1. Correct GC/mappability biases for tumor read depth



$N^{normal}$  = normal read depth

$N^{tumor}$  = tumor read depth

$$\frac{N^{tumor}}{N^{normal}} = \text{copyratio}$$



# Copy Number Analysis Workflow: GC content bias

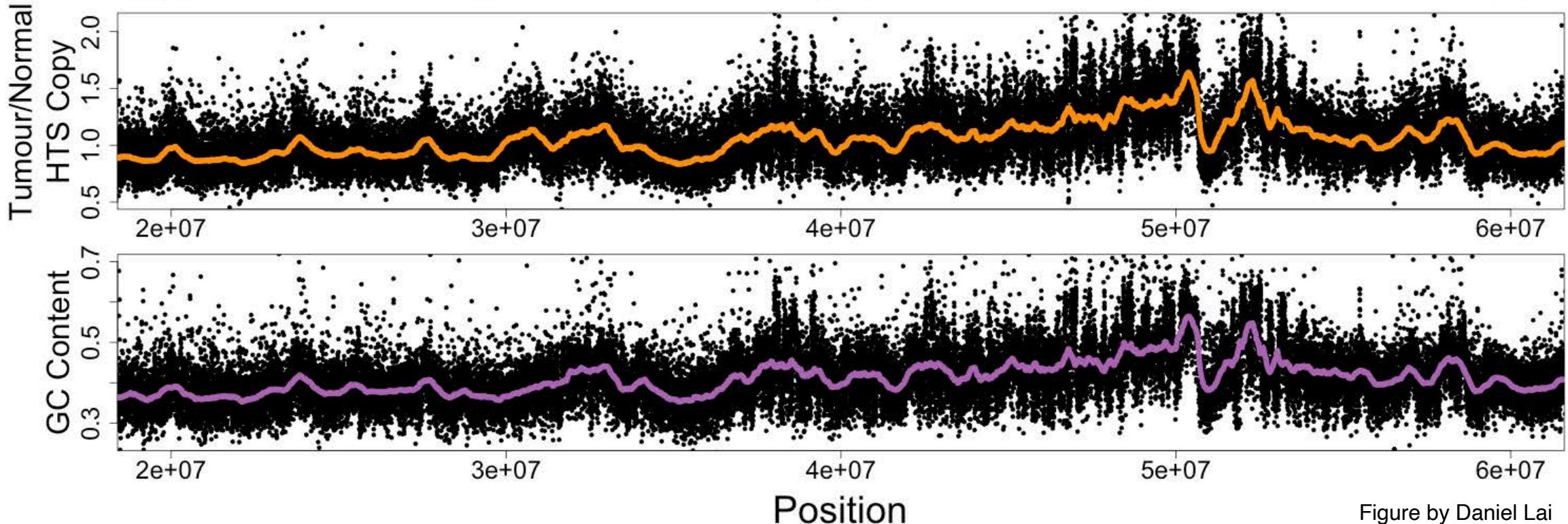


Figure by Daniel Lai

Benjamini and Speed. *Nucleic Acids Research* **40**:e72-86 (2012)

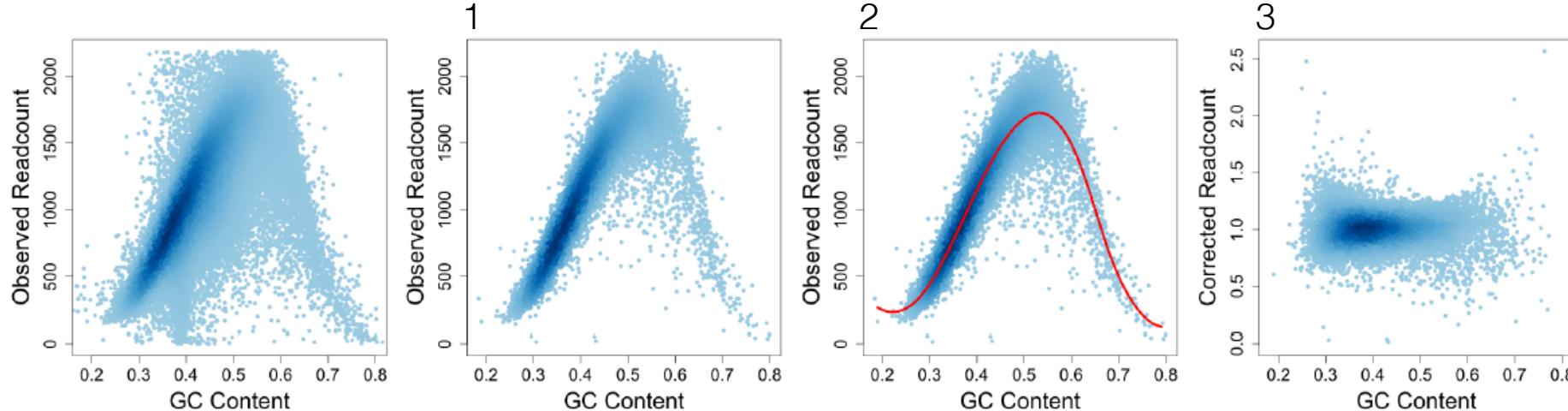
Boeva et al. *Bioinformatics* **29**(3):423-5 (2012)

Ha et al. *Genome Research* **22**:1995-2007 (2012).

Adalsteinsson\*, Ha\* Freeman\* et al. *Nature Communications* **8**:1324 (2017)

# Copy Number Analysis Workflow: GC correction (1)

1. Randomly select 50k bins and filter outliers (bottom & top 1%)
2. Fit `loess()` curve
  - local nonlinear regression
  - smoothing parameter (bandwidth): amount of local data to fit
3.  $\text{corrected read count} = \frac{\text{observed read count (blue dot)}}{\text{expected read count (red line)}}$ 
  - relative differences between observed and predicted read counts



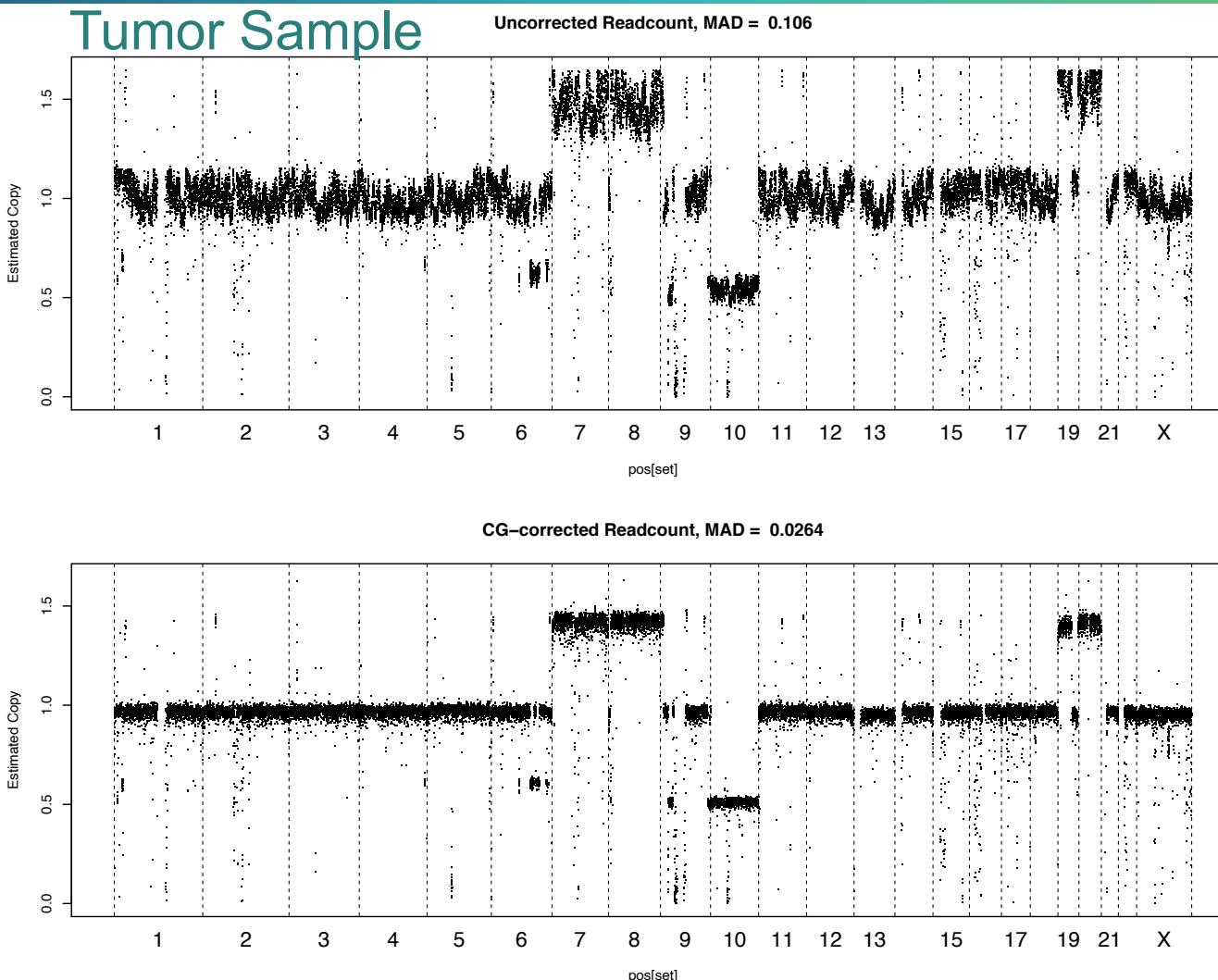
Benjamini and Speed. *Nucleic Acids Research* **40**:e72-86 (2012)

Boeva et al. *Bioinformatics* **29**(3):423-5 (2012)

Ha et al. *Genome Research* **22**:1995-2007 (2012).

Adalsteinsson\*, Ha\* Freeman\* et al. *Nature Communications* **8**:1324 (2017) 10

# Copy Number Analysis Workflow: GC correction (2)



Un-corrected  
read counts



GC-corrected  
read counts

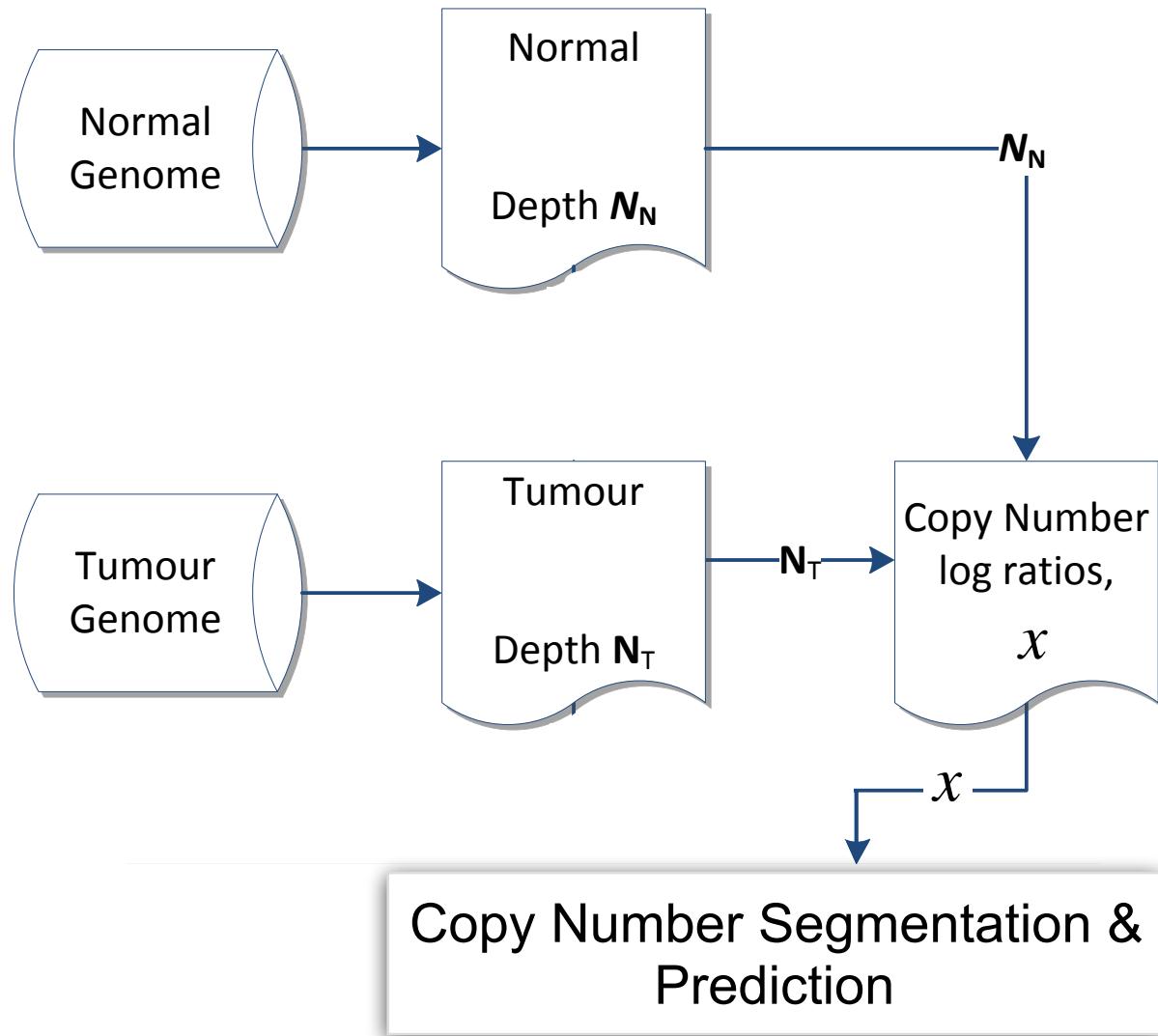
Benjamini and Speed. *Nucleic Acids Research* **40**:e72-86 (2012)

Boeva et al. *Bioinformatics* **29**(3):423-5 (2012)

Ha et al. *Genome Research* **22**:1995-2007 (2012).

Adalsteinsson\*, Ha\* Freeman\* et al. *Nature Communications* **8**:1324 (2017) 11

# Copy Number Analysis Workflow: Normalization



## 1. Correct GC/mappability biases for tumor read depth

$N^{normal}$  = normal read depth

$N^{tumor}$  = tumor read depth

$\hat{N}^{normal}$  = corrected normal read depth

$\hat{N}^{tumor}$  = corrected tumor read depth

$$\log_2 \left( \frac{\hat{N}^{tumor}}{\hat{N}^{normal}} \right) = \text{corrected log ratio}$$

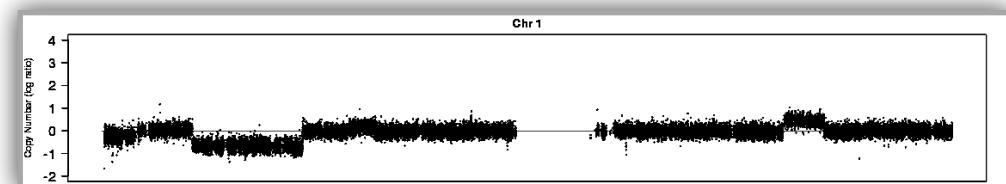
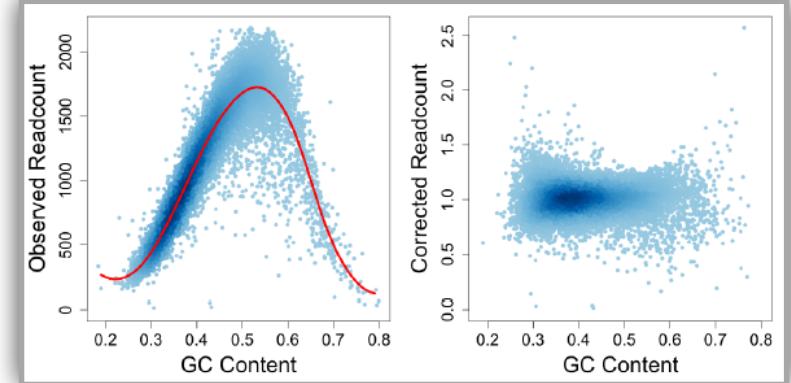
## 2. Perform segmentation and copy number prediction

# Input Sequencing Data for Copy Number Analysis

## Input Data After Normalization

- GC-content bias correction applied to separately for
  - tumor sample reads  $N_{1:T}^{Tumor}$
  - normal sample reads  $N_{1:T}^{Normal}$
- Normalize tumor corrected read counts  $\hat{N}_i^{Tumor}$  with normal corrected read counts  $\hat{N}_i^{Normal}$  to obtain the log ratio for bin  $t \in \{1, \dots, T\}$

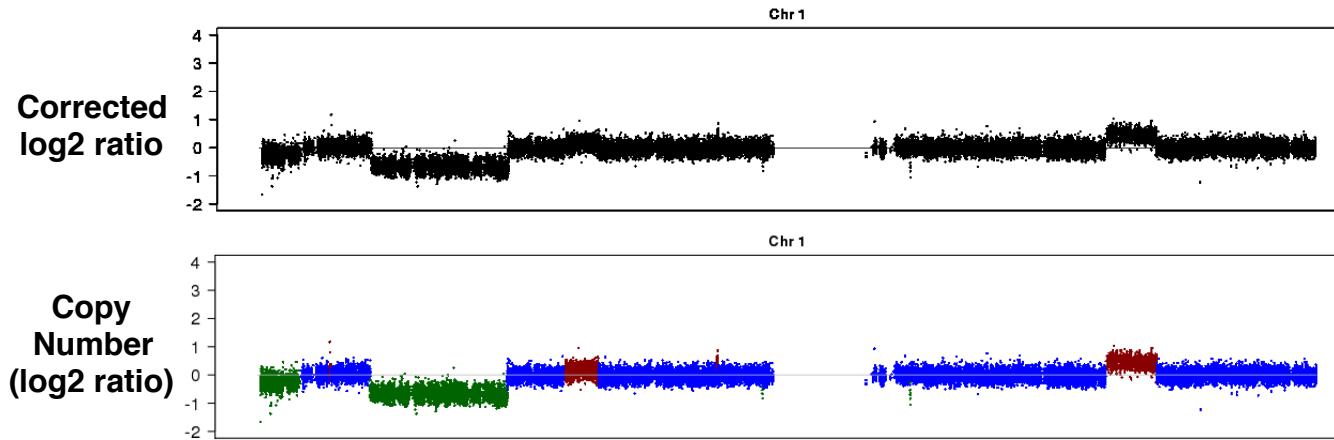
$$x_t = \log_2 \left( \frac{\hat{N}_t^{Tumor}}{\hat{N}_t^{Normal}} \right)$$



Benjamini and Speed. *Nucleic Acids Research* **40**:e72-86 (2012)  
Boeva et al. *Bioinformatics* **29**(3):423-5 (2012)  
Ha et al. *Genome Research* **22**:1995-2007 (2012).  
Adalsteinsson\*, Ha\* Freeman\* et al. *Nature Communications* **8**:1324 (2017)

# Copy Number Segmentation and Prediction

---



Data normalization

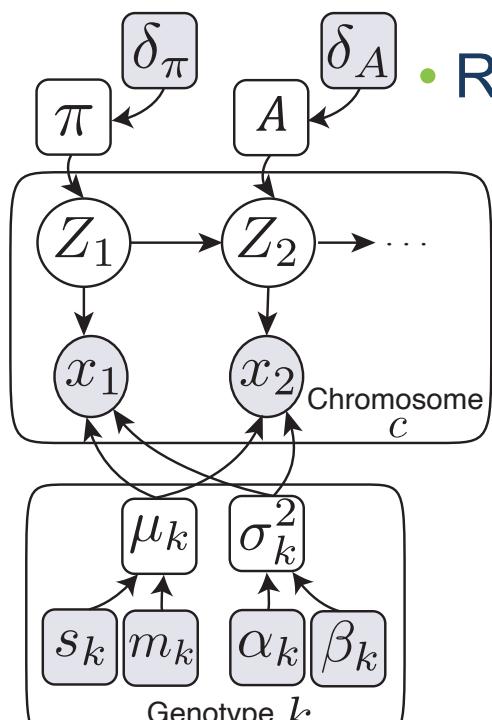
Copy Number Segmentation

- What are the genomic segments of copy number alterations?
- What is the copy number value for each segment?
- How do we account for variability/noise in the data?

Continuous hidden Markov model (HMM)

## 2. Continuous hidden Markov model

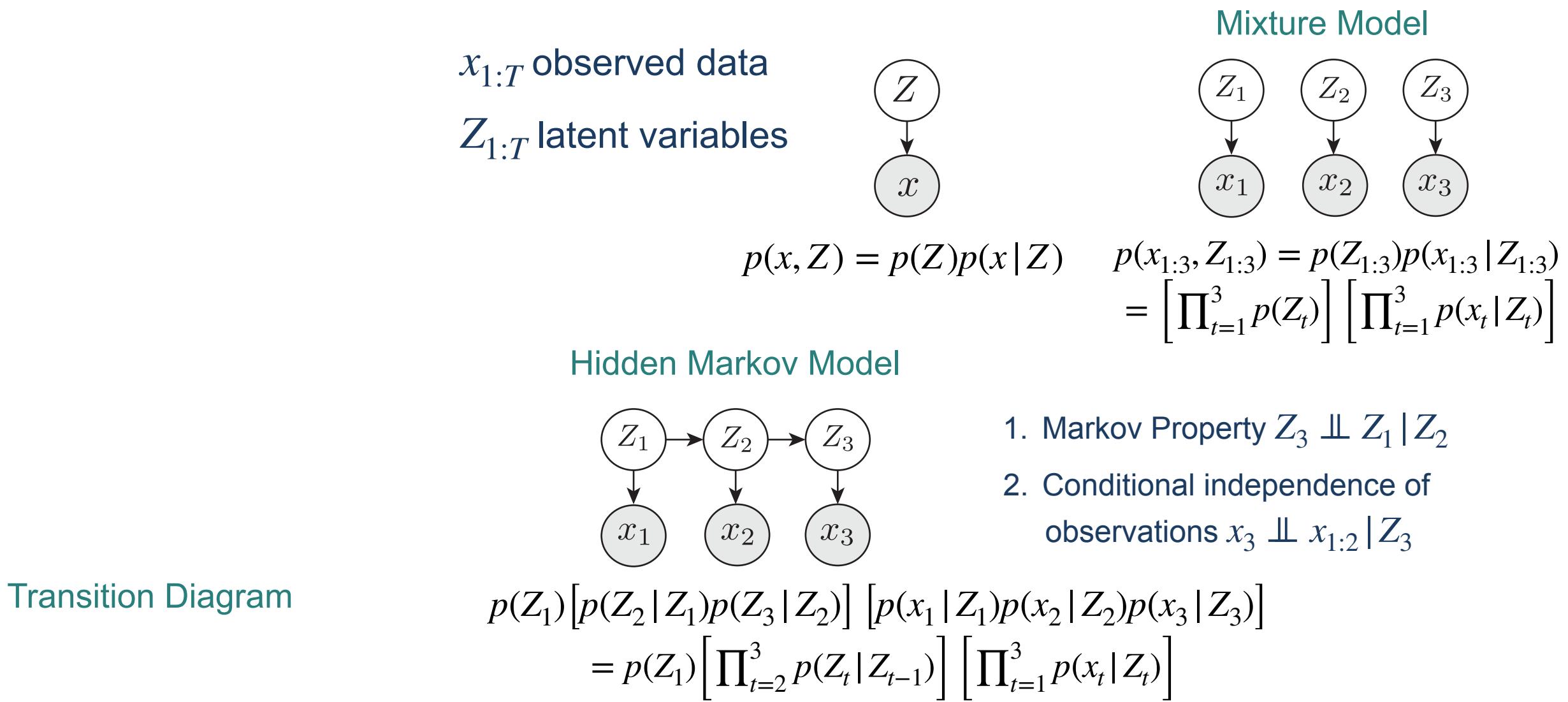
- Hidden Markov Models vs Mixture Models
- Components of a Continuous HMM
- Inference and Parameter Learning using EM



- References:

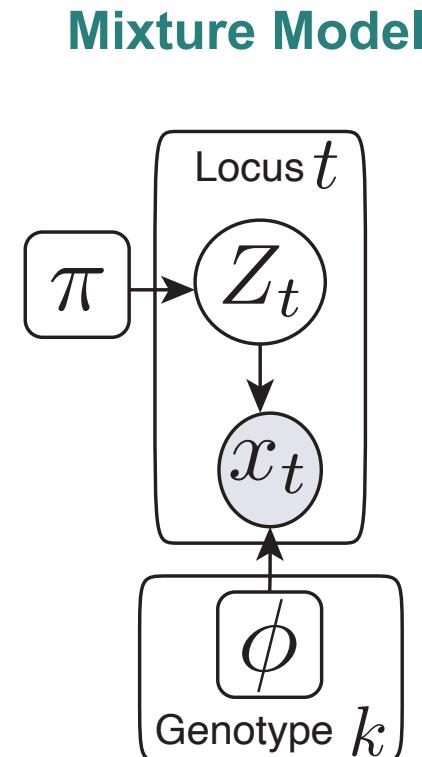
- **HMMcopy** - Ha et al. *Genome Research* **22**:1995-2007 (2012).
- **ichorCNA** - Adalsteinsson\*, Ha\* Freeman\* et al. *Nature Communications* **8**:1324 (2017).
- **TitanCNA** - Ha et al. TITAN: inference of copy number architectures in clonal cell populations from tumor whole-genome sequencing data. *Genome Research* **24**:1881-1893 (2014).
- Murphy, K. (2012). Machine Learning: A Probabilistic Perspective. MIT Press. ISBN: 9780262018029
- Bishop, C. M. (2006). Pattern Recognition and Machine Learning (Information Science and Statistics). Springer. ISBN: 0387310738

# Probabilistic Graphical Model for HMMs



# From Mixture Models to Hidden Markov Models

- Mixture model for iid data is a special case of the HMM



$$p(x_{1:T}, Z_{1:T}) = p(Z_{1:T})p(x_{1:T} | Z_{1:T})$$

**Joint Probability Distribution  
(Complete data likelihood)**

$x_{1:T}$  observed data

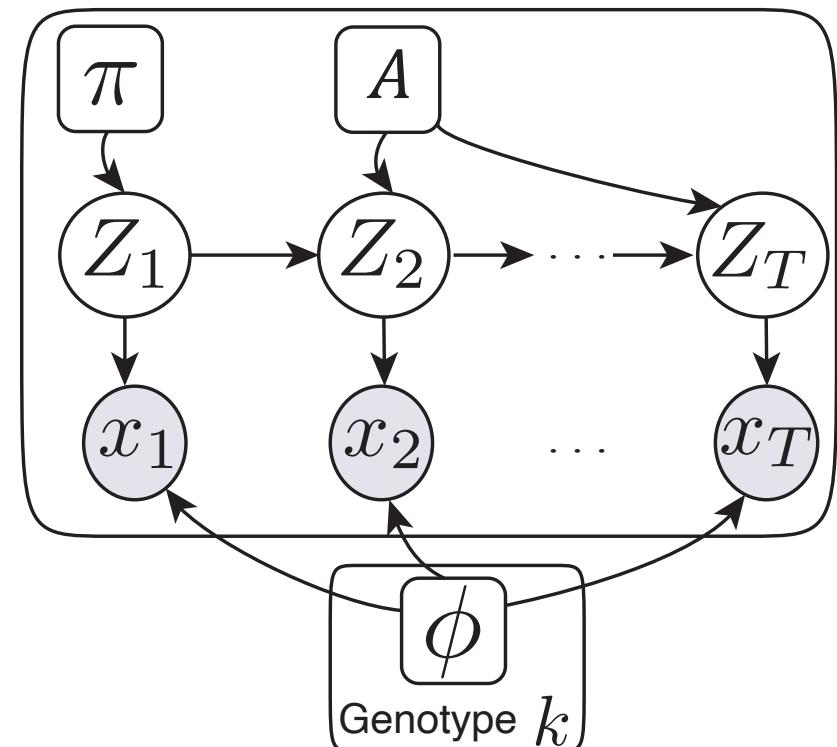
$Z_{1:T}$  latent variables

$\pi$  mixture weights

$\phi$  observation parameters

$$p(x_{1:T}, Z_{1:T} | \theta) = \left[ \prod_{t=1}^T p(Z_t | \pi) \right] \prod_{t=1}^T p(x_t | Z_t, \phi)$$

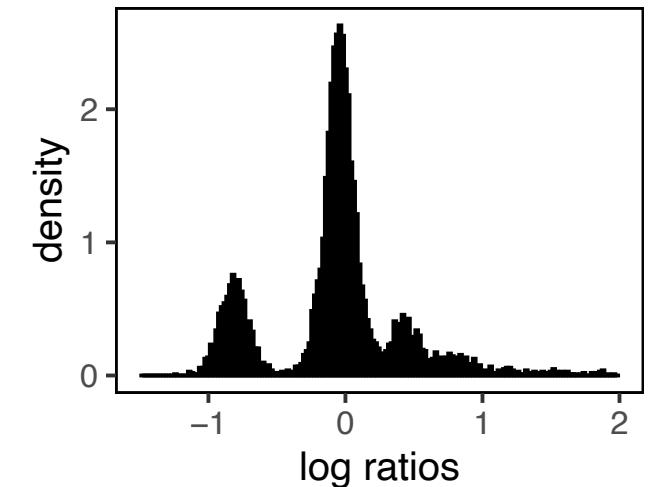
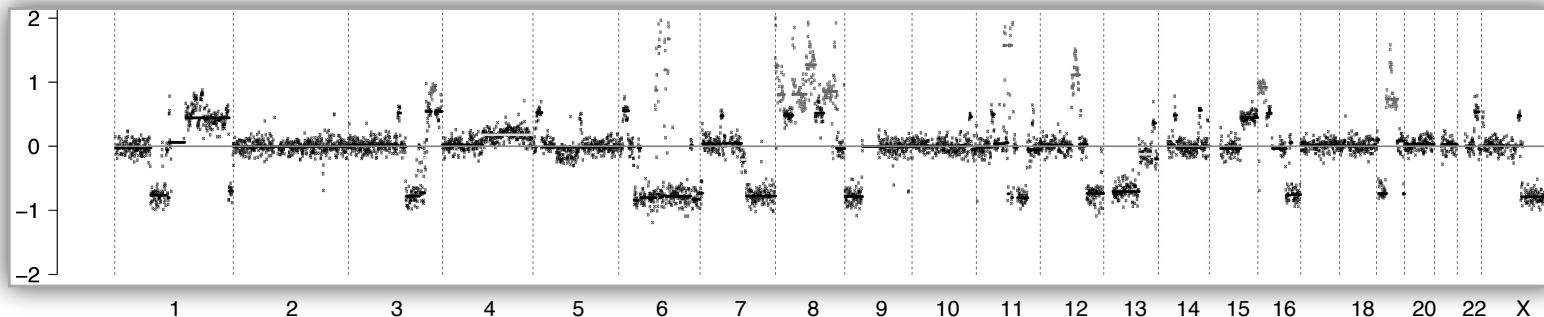
**Hidden Markov Model**



$$p(x_{1:T}, Z_{1:T} | \theta) = \left[ p(Z_1 | \pi) \prod_{t=2}^T p(Z_t | Z_{t-1}, A) \right] \prod_{t=1}^T p(x_t | Z_t, \phi)$$

# Gaussian Mixture Model for Log Ratio Data

Corrected  
(log2 ratio)



The ratios  $\frac{\hat{r}_t^{Tumor}}{\hat{r}_t^{Normal}}$ , for all  $t$  loci are log-normal distributed, so the log ratios  $x_{1:T}$  follow a normal distribution.

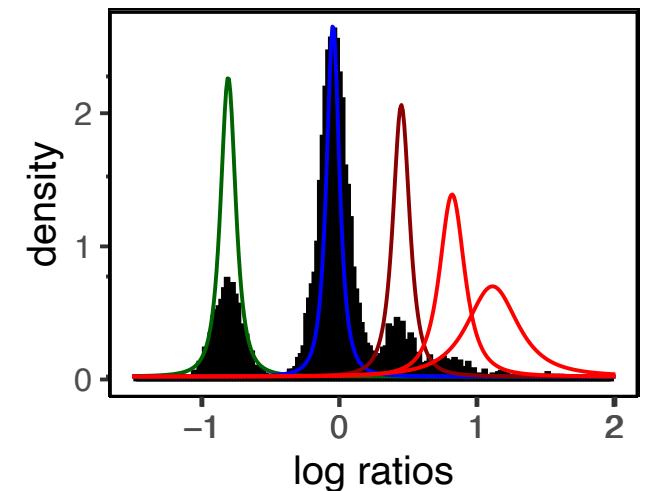
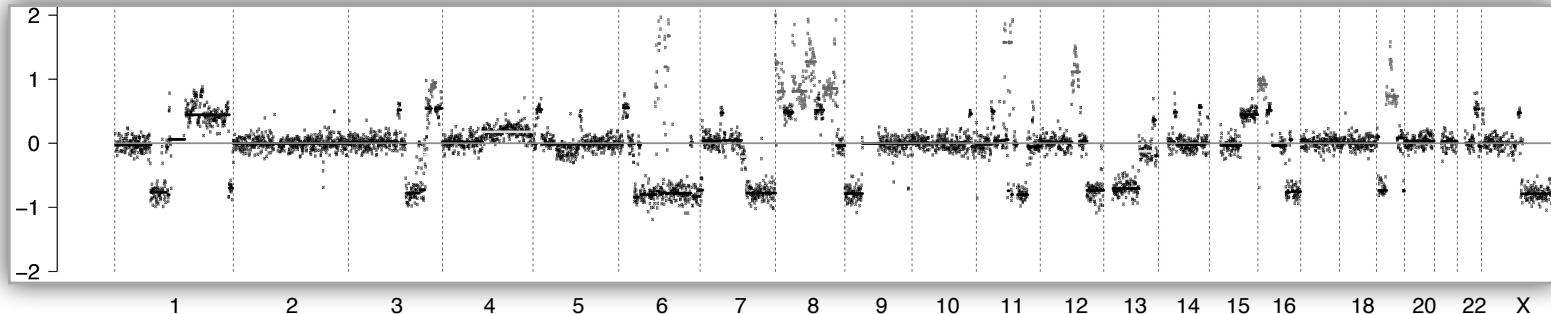
## The Gaussian Distribution

Let  $X$  be a continuous measurement with mean  $\mu$  and variance  $\sigma^2$ , then  $X$  has a Gaussian distribution,  
 $X \sim \mathcal{N}(\mu, \sigma^2)$  or  $p(X = x) = \mathcal{N}(x | \mu, \sigma^2)$  where

$$\mathcal{N}(x | \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(x-\mu)^2}$$

# Gaussian Mixture Model for Log Ratio Data

Corrected  
(log2 ratio)



The ratios  $\frac{\hat{r}_t^{Tumor}}{\hat{r}_t^{Normal}}$ , for all  $t$  loci are log-normal distributed, so the log ratios  $x_{1:T}$  follow a normal distribution.

## The Gaussian Mixture Model

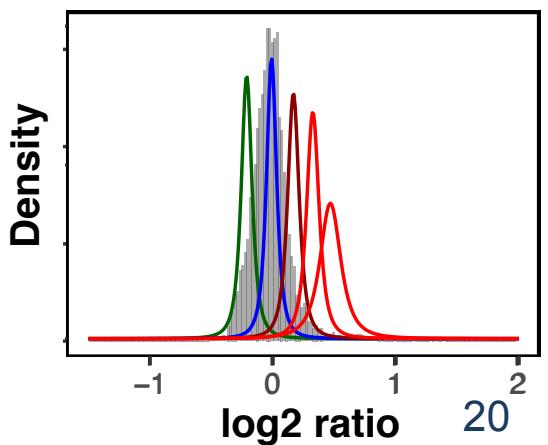
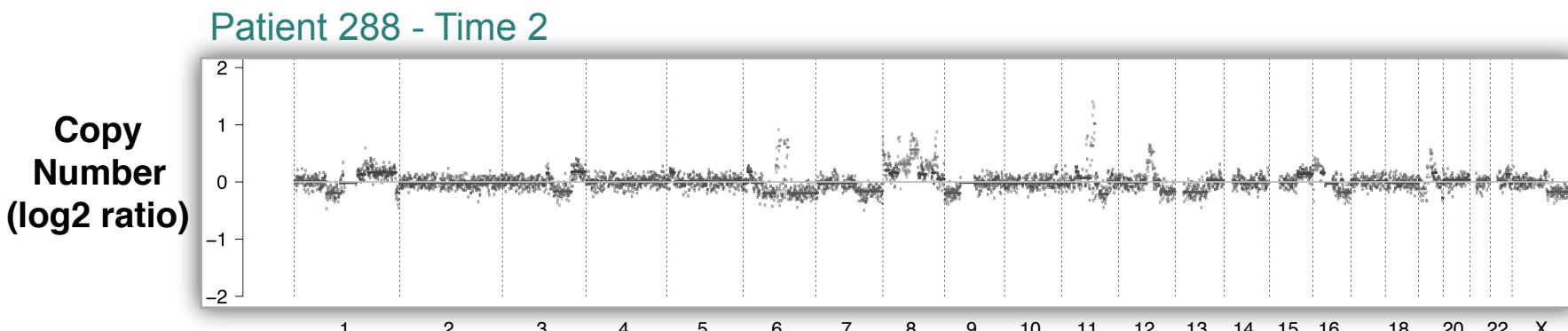
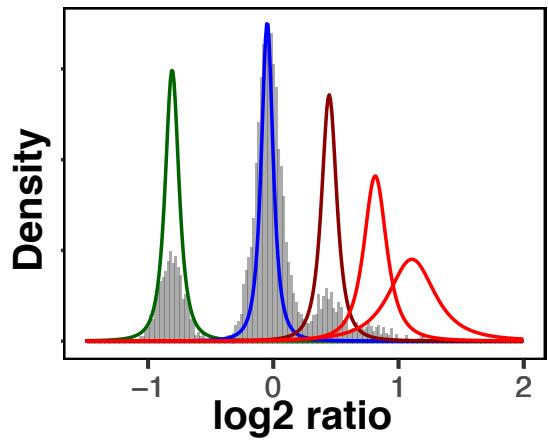
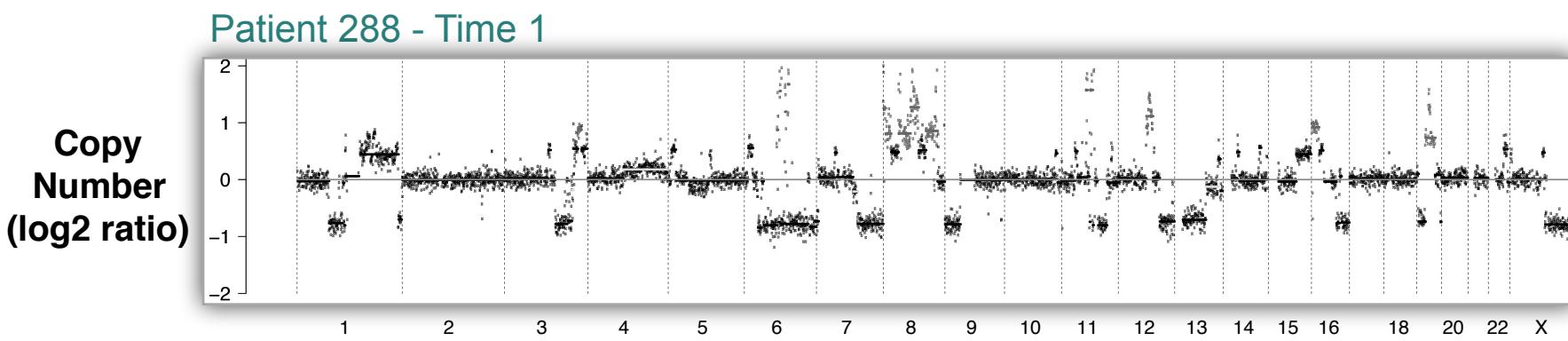
Define a likelihood for a ***K*-component mixture of Gaussians** with means  $\mu = \{\mu_1, \dots, \mu_K\}$  and variance  $\sigma^2 = \{\sigma_1^2, \dots, \sigma_K^2\}$ , where the observation model is a conditional Gaussian

$$p(x_t | Z_i = k, \mu, \sigma^2) = \mathcal{N}(x_t | \mu_k, \sigma_k^2)$$

# Rationale for Estimating Likelihood Parameters

Why are the data multi-modal?

Why should we estimate the mixture distribution parameters?



# Components of a continuous HMM

## Input Data: log ratios

There are  $T$  different data points with continuous values  $x = \{x_1, \dots, x_T\}$ .

## Latent State Model

- The latent variables  $Z = \{Z_1, \dots, Z_T\}$  can be assigned values from a set of  $K$  discrete states with probability

## Initial state distribution

- The probabilities of the states for the first latent variable  $Z_1$  is the parameter  $\pi = \{\pi_1, \dots, \pi_K\}$
- $\pi$  follows a prior distribution  $p(\pi_k | \delta_k) = Dir(\pi_k | \delta_k)$

## Transition Model (homogenous HMM)

- The conditional distribution between adjacent data  $i$  and  $j$  corresponds to a table  $A$  of transition probabilities

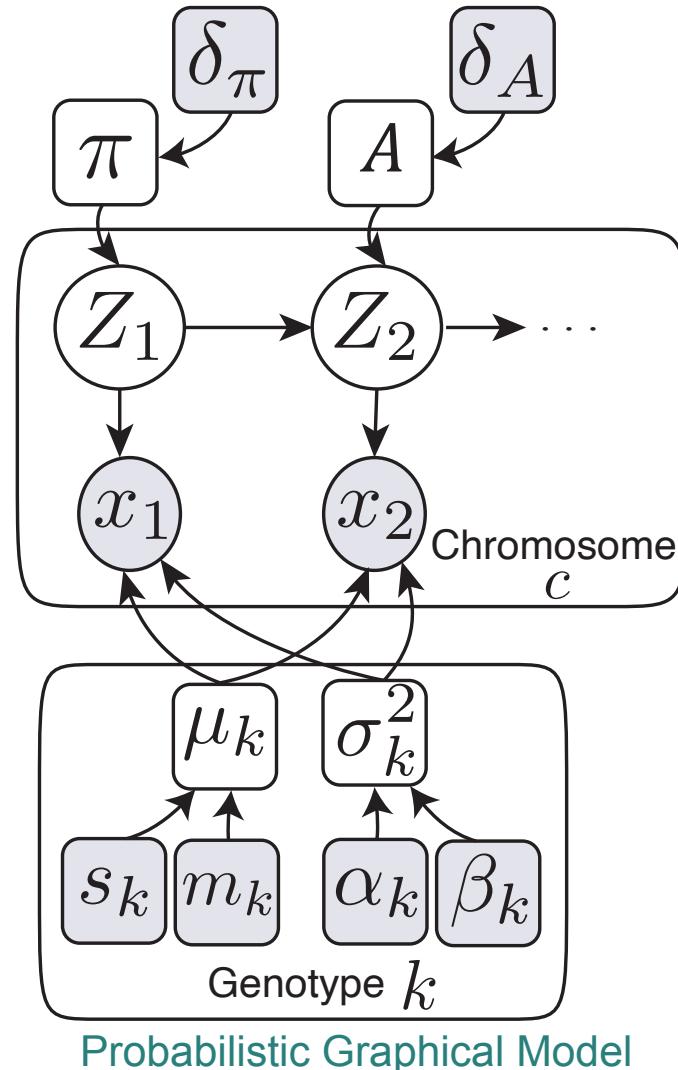
$$p(Z_t = j | Z_{t-1} = i) = A_{ij}$$

## Emission Model (Continuous HMM)

- The emission is modeled using a mixture of Gaussians with the likelihood model

$$p(x_t | Z_t = k, \mu, \sigma^2) = \mathcal{N}(x_t | \mu_k, \sigma_k^2)$$

- $\mu$  is modeled with a prior  $p(\mu_k | m_k, s_k) = \mathcal{N}(\mu_k | m_k, s_k)$
- $\sigma^2$  is modeled with prior  $p(\sigma_k^2 | \alpha_k, \beta_k) = InvGamma(\sigma_k^2 | \alpha_k, \beta_k)$



# Inference & parameter estimation using EM

---

**Expectation-Maximization: Inference and parameter training**

**Initialize parameters:**  $\pi, A, \mu$

**E-Step: Inference using Forwards-Backwards Algorithm (Baum-Welch)**

1. Compute “responsibilities” (Posterior of the latent states  $\gamma(Z_{1:T})$ )
  - State  $Z_t = k$  is “responsible for generating observation  $x_t$ ”
2. Compute “2-slice marginals” (Posterior of state transitions  $\xi(Z_{t-1}, Z_t)$ )
  - Expected number of transitions from state  $k$  to  $j$

**M-Step: Update parameters (learning)**

1. Initial state distribution,  $\pi$
2. Transition probabilities,  $A$
3. Emission likelihood parameters,  $\mu$

**Iterate** between E-Step and M-Step, check when log posterior likelihood,  $\log \mathbb{P}$ , stops increasing.

# Inference & parameter estimation using EM (E-Step)

---

## E-Step: Forwards-backwards Algorithm (Baum-Welch; Sum-Product)

- Forward,  $\alpha(Z_t)$ : joint prob. of observing all *past* data up to time  $t$  when given  $Z_t$
- Backward,  $\beta(Z_t)$ : conditional prob. of all *future* data from time  $t + 1$  to  $T$  when given  $Z_t$

### Forward Probabilities ( $T \times K$ ) - Past

$$\alpha(Z_t = k) = \mathcal{N}(x_t | \mu_k, \sigma_k^2) \sum_{j=1}^K \left\{ A_{jk} \alpha(Z_{t-1} = j) \right\}$$

### Backward Probabilities ( $T \times K$ ) - Future

$$\beta(Z_t = k) = \sum_{j=1}^K \left\{ \mathcal{N}(x_{t+1} | \mu_j, \sigma_j^2) A_{kj} \beta(Z_{t+1} = j) \right\}$$

# Inference & parameter estimation using EM (E-Step)

## E-Step: Compute Responsibilities & 2-Slice Marginals

- Responsibilities,  $\gamma(Z_t = k)$ : is the posterior on the latent states

$$\gamma(Z_t = k) = \frac{\alpha(Z_t = k)\beta(Z_t = k)}{p(\mathbf{x})}$$

**Responsibilities**  
Matrix  $K \times T$

- 2-Slice Marginals,  $\xi(Z_{t-1} = k, Z_t = j)$ : is the expected number of transitions between  $k$  to  $j$

$$\xi(Z_{t-1} = k, Z_t = j) = \frac{\alpha(Z_{t-1} = k)A_{kj}\mathcal{N}(x_t | \mu_j, \sigma_j^2)\beta(Z_t = j)}{p(\mathbf{x})}$$

**2 Slice Marginals**  
Matrix  $K \times K \times (T - 1)$

- The likelihood  $p(\mathbf{x}) = p(\mathbf{x} | \boldsymbol{\mu}, \boldsymbol{\sigma}^2, \boldsymbol{\pi})$  is computed in the forwards recursion

$$\ell = \log p(\mathbf{x}) = \sum_{t=1}^T \log \left( \sum_{k=1}^K \alpha(Z_t = k) \right)$$

**Log likelihood**

# Inference & parameter estimation using EM (M-Step)

Expected complete data log likelihood

Initial State Dist

Transition

Emission

$$Q = \sum_{k=1}^K \gamma(Z_1 = k) \log \pi_k + \sum_{t=2}^T \sum_{j=1}^K \sum_{k=1}^K \xi(Z_{t-1} = k, Z_t = j) \log A_{kj} + \sum_{t=1}^T \sum_{k=1}^K \gamma(Z_t = k) \log \mathcal{N}(x_t | \mu_k, \sigma_k^2)$$

M-Step: update parameters,  $\pi, \mu, \sigma^2$

$$\hat{\pi}_k = \frac{\gamma(Z_1 = k) + \delta^\pi(k) - 1}{\sum_{j=1}^K \{\gamma(Z_1 = j) + \delta^\pi(j) - 1\}}$$

$$\hat{\mu}_k = \frac{s_k \sum_{t=1}^T \gamma(Z_t = k) x_t + m \sigma_k^2}{s_k \sum_{t=1}^T \gamma(Z_t = k) + \sigma_k^2}$$

$$\hat{\sigma}_k^2 = \frac{\sum_{t=1}^T \gamma(Z_t = k) (x_t - \bar{x}_k)^2 + 2\beta_k}{\sum_{t=1}^T \gamma(Z_t = k) + 2(\alpha_k + 1)}$$

$$\text{Where } \bar{x} = \frac{\sum_{t=1}^T \gamma(Z_t = k) x_t}{\sum_{t=1}^T \gamma(Z_t = k)}$$

MAP for initial state distribution

MAP for Gaussian means

MAP for Gaussian variance

# Inference & parameter estimation using EM (M-Step)

## M-Step: Update transition matrix, $A$

$$\hat{A}_{kj} = \frac{\sum_{t=2}^T \xi(Z_{t-1} = k, Z_t = j) + \delta_j^A(k)}{\sum_{l=1}^K \left\{ \sum_{t=2}^T \xi(Z_{t-1} = k, Z_t = l) + \delta_j^A(l) \right\}}$$

Expected number of transitions from  $k$  to  $j$

Prior counts

“Pseudo-counts”

Expected number of transitions from  $k$  to any other state

## Evaluate the log posterior

$$\log \mathbb{P} = \ell + \log Dir(\hat{\pi} | \delta) + \sum_{k=1}^K \left\{ \log \mathcal{N}(\hat{\mu}_k | m_k, s_k) + \log InvGamma(\hat{\sigma}_k^2 | \alpha_k, \beta_k) + \log Dir(A_{k,1:K}^{(0)} | \hat{A}_{k,1:K}) \right\}$$

**Iterate between E-Step and M-Step:** stop when  $\log \mathbb{P}$  changes less than  $\epsilon$  compared to previous EM iteration.

---

**Algorithm 1** HMM Parameter Learning using EM

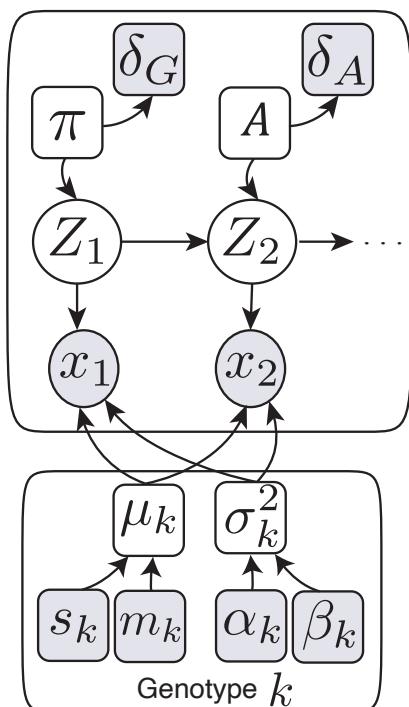
---

- 1: **Inputs:**  
    Data:  $x_{1:T}$   
    Initial parameters:  $\pi^{(0)}, \mu_{1:K}^{(0)}, (\sigma_{1:K}^2)^{(0)}, A^{(0)}$   
    Hyperparameters:  $\delta^\pi, m_{1:K}, s_{1:K}, \alpha_{1:K}, \beta_{1:K}, \delta^A$
- 2: **Initialize:**  
     $\pi \leftarrow \pi^{(0)}, \mu_{1:K} \leftarrow \mu_{1:K}^{(0)}, \sigma_{1:K}^2 \leftarrow (\sigma_{1:K}^2)^{(0)}, A \leftarrow A^{(0)}$
- 3: Compute observed likelihood using initial parameters:  
4:     `obs.lik`  $\leftarrow$  `compute.gauss.lik()`
- 5: **while** converged = false **do**
- 6:     **E-Step:** Compute responsibilities using current parameters:  
7:          $(\gamma(Z_{1:T}), \text{loglik}) \leftarrow .\text{Call}(\text{"forward\_backward"})$
- 8:     **M-Step:** Update parameters:  
9:          $\hat{\pi} \leftarrow \text{update.pi}()$   
10:          $\hat{\mu}_{1:K} \leftarrow \text{update.mu}()$   
11:          $\hat{\sigma}_{1:K}^2 \leftarrow \text{update.var}()$   
12:          $\hat{A} \leftarrow \text{update.A}()$
- 13:     Assign updated parameters:  
14:          $\pi \leftarrow \hat{\pi}, \mu_{1:K} \leftarrow \hat{\mu}_{1:K}, \sigma_{1:K}^2 \leftarrow \hat{\sigma}_{1:K}^2, A \leftarrow \hat{A}$
- 15:     Re-compute observed likelihood using updated parameters:  
16:         `obs.lik`  $\leftarrow$  `compute.gauss.lik()`
- 17:     Compute log Posterior:  
18:         `logP[curr.iter]`  $\leftarrow$  `compute.log.posterior(loglik, ...)`
- 19:     **if** ( `logP[curr.iter]` - `logP[prev.iter]` <  $\epsilon$  ) **then**
- 20:         converged = true
- 21:     **end if**
- 22:     `logP[prev.iter]`  $\leftarrow$  `logP[curr.iter]`
- 23: **end while**
- 24: **return** Converged parameters  $\hat{\pi}, \hat{\mu}_{1:K}, \hat{\sigma}_{1:K}^2, \hat{A}$

---

### 3. Copy Number Profiling using a HMM

- Defining the HMM for copy number analysis
- Copy number segmentation using Viterbi
- References:
  - **HMMcopy** - Ha et al. *Genome Research* **22**:1995-2007 (2012).
  - **ichorCNA** - Adalsteinsson\*, Ha\* Freeman\* et al. *Nature Communications* **8**:1324 (2017).
  - **TitanCNA** - Ha et al. TITAN: inference of copy number architectures in clonal cell populations from tumor whole-genome sequencing data. *Genome Research* **24**:1881-1893 (2014).
  - Murphy, K. (2012). Machine Learning: A Probabilistic Perspective. MIT Press. ISBN: 9780262018029
  - Bishop, C. M. (2006). Pattern Recognition and Machine Learning (Information Science and Statistics). Springer. ISBN: 0387310738



# Probabilistic Model for Copy Number Analysis

## Input Data: log ratios

There are  $T$  different genomic bins with log ratio data  $x = \{x_1, \dots, x_T\}$ .

## Latent State Model: copy number states

There are 5 different possible copy number states (genotypes),  $K = \{1, 2, 3, 4, 5\}$

1. A specific genotype  $k \in K$  can be assigned to each of the **latent states**  $Z = \{Z_1, \dots, Z_T\}$
2. The **initial state distribution**  $\pi = \{\pi_1, \dots, \pi_5\}$  is used for the first latent state  $Z_1$

## Transition Model

3. The probabilities for transitioning to copy number state  $j$  in bin  $t$  from state  $i$  in bin  $t - 1$  are contained in matrix  $A \in \mathbb{R}^{K \times K}$

$$p(Z_t = j | Z_{t-1} = i) = A_{ij}$$

## Emission Model: likelihood for log ratio data

For each copy number state, the log ratio means are  $\mu = \{\mu_1, \dots, \mu_5\}$  and variance  $\sigma^2 = \{\sigma_1^2, \dots, \sigma_5^2\}$

4. The **emission model** is a mixture of Gaussians with *unknown* parameters,  $\mu$  and  $\sigma^2$ ,

$$p(x_i | Z_i = k, \mu, \sigma^2) = \mathcal{N}(x_i | \mu_k, \sigma_k^2)$$

## Prior Model

5. The **priors** in the model have hyper-parameters  $\delta^\pi$ ,  $m_{1:K}$ ,  $s_{1:K}$ ,  $\alpha_{1:K}$ ,  $\beta_{1:K}$ ,  $\delta_{1:K}^A$

$$p(\pi | \delta^\pi) = \text{Dirichlet}(\pi | \delta^\pi)$$

$$p(\mu_k | m_k, s_k) = \mathcal{N}(\mu_k | m_k, s_k)$$

$$p(\sigma_k^2 | \alpha_k, \beta_k) = \text{InvGamma}(\sigma_k^2 | \alpha_k, \beta_k)$$

$$p(A_{k,1:K} | \delta^A) = \text{Dirichlet}(A_{k,1:K} | \delta_k^A)$$

	$j$				
$i$	0	...	5		
0					
...					
5					

$\sum_{j=1}^K A_{ij} = 1$

# Probabilistic Model for Copy Number Analysis

## Input Data: log ratios

There are  $T$  different genomic bins with log ratio data  $x = \{x_1, \dots, x_T\}$ .

## Latent State Model: copy number states

There are 5 different possible copy number states (genotypes),  $K = \{1, 2, 3, 4, 5\}$

1. A specific genotype  $k \in K$  can be assigned to each of the latent states  $Z = \{Z_1, \dots, Z_T\}$
2. The initial state distribution  $\pi = \{\pi_0, \dots, \pi_5\}$  is used for the first latent state  $Z_1$

## Transition Model

3. The probabilities for transitioning to copy number state  $j$  in bin  $t$  from state  $i$  in bin  $t - 1$  are contained in matrix  $A \in \mathbb{R}^{K \times K}$
- $$p(Z_t = j | Z_{t-1} = i) = A_{ij}$$

## Emission Model: likelihood for log ratio data

For each copy number state, the log ratio means are  $\mu = \{\mu_0, \dots, \mu_5\}$  and variance  $\sigma^2 = \{\sigma_0^2, \dots, \sigma_5^2\}$

4. The emission model is a mixture of Gaussians with unknown parameters,  $\mu$  and  $\sigma^2$ ,

$$p(x_i | Z_i = k, \mu, \sigma^2) = \mathcal{N}(x_i | \mu_k, \sigma_k^2)$$

## Prior Model

5. The priors in the model have hyper-parameters  $\delta^\pi$ ,  $m_{1:K}$ ,  $s_{1:K}$ ,  $\alpha_{1:K}$ ,  $\beta_{1:K}$ ,  $\delta_{1:K}^A$

$$p(\pi | \delta^\pi) = \text{Dirichlet}(\pi | \delta^\pi)$$

$$p(\mu_k | m_k, s_k) = \mathcal{N}(\mu_k | m_k, s_k)$$

$$p(\sigma_k^2 | \alpha_k, \beta_k) = \text{InvGamma}(\sigma_k^2 | \alpha_k, \beta_k)$$

$$p(A_{k,1:K} | \delta^A) = \text{Dirichlet}(A_{k,1:K} | \delta_k^A)$$



	$j$				
$i$	0	...	5		
0					
...					
5					

$\sum_{j=1}^K A_{ij} = 1$

# Copy number segmentation using Viterbi

---

## Viterbi algorithm (Max-Sum)

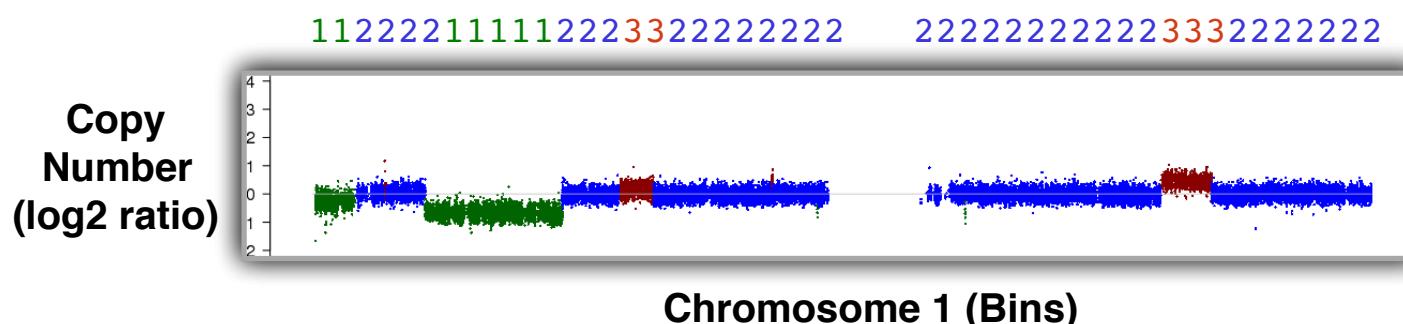
- Find the most probable sequence of copy number states

$$\hat{Z}_{1:T} = \max_{Z_{1:T}} \log p(Z_{1:T} | x_{1:T})$$

- Perform max-sum of probabilities in trellis

$$\omega(Z_{t+1} = k) = \log \mathcal{N}(x_{t+1} | \mu_k, \sigma_k^2) + \max_{Z_t} \left\{ \log A_{Z_t, k} + \log \omega(Z_t) \right\}$$

- Back trace from  $\omega(Z_T)$  to find overall most probable path



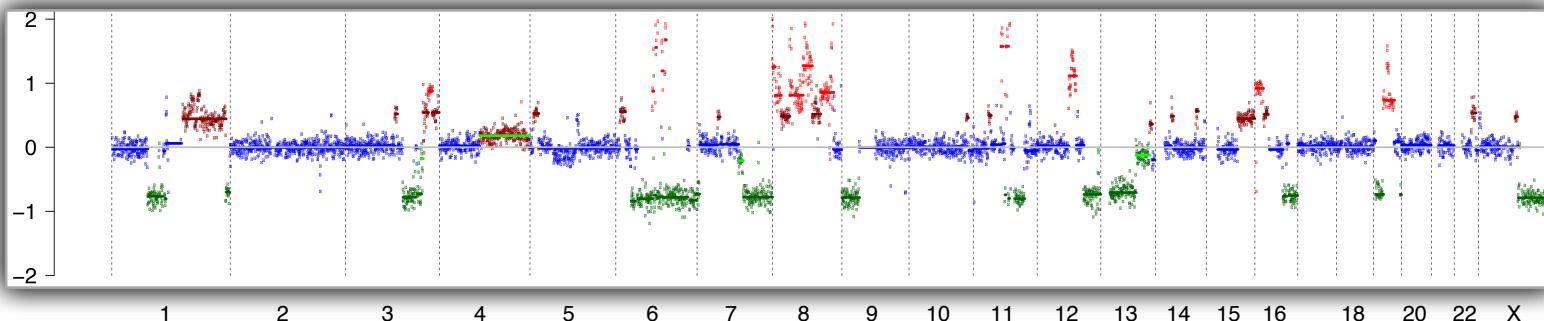
# Rationale for Estimating Likelihood Parameters

Why should we estimate the mixture distribution parameters?

- Can account for technical and biological “noise” by estimating model parameters  
 $\mu = \{\mu_0, \dots, \mu_5\}$  and  $\sigma^2 = \{\sigma_0^2, \dots, \sigma_5^2\}$ ?

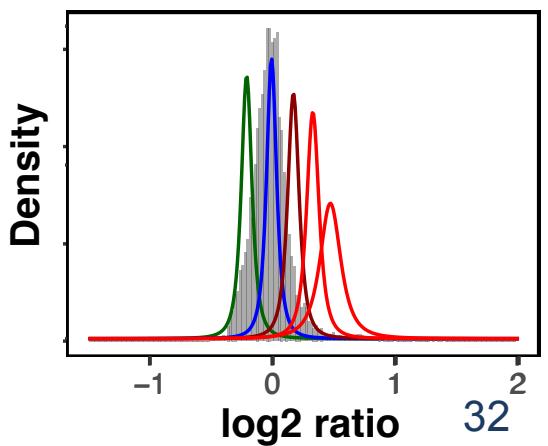
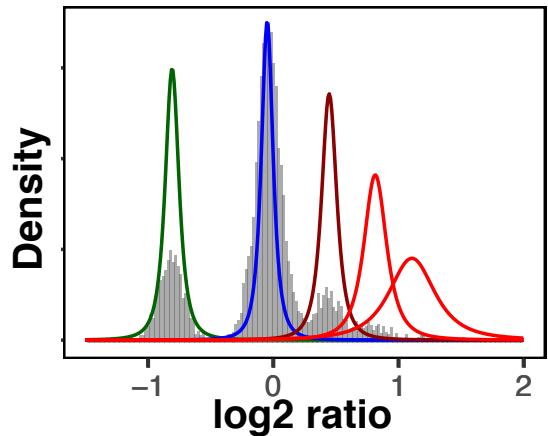
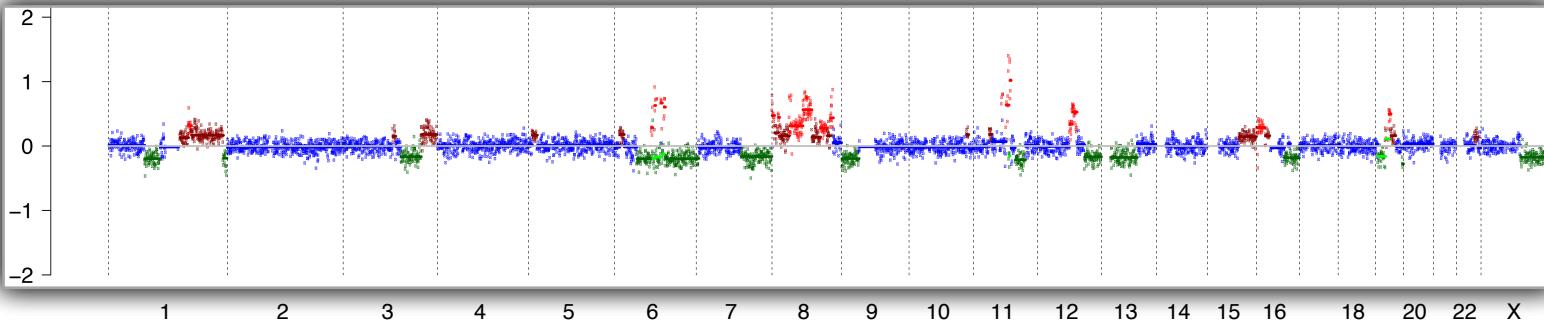
Patient 288 - Time 1

Copy Number (log2 ratio)



Patient 288 - Time 2

Copy Number (log2 ratio)



# Homework #6: Profiling copy number alterations

---

A. Implement a copy number alteration (CNA) caller described in Lecture 3

- Implement components of a continuous HMM in a Bayesian framework
- Learn the parameters and infer the genotypes using EM
- Predict the copy number alteration segments for a chromosome.
- Expected outputs for each question will be provided so that you can check your code.

B. Power calculations for mutation detection described in Lecture 4

**Due: May 15th**

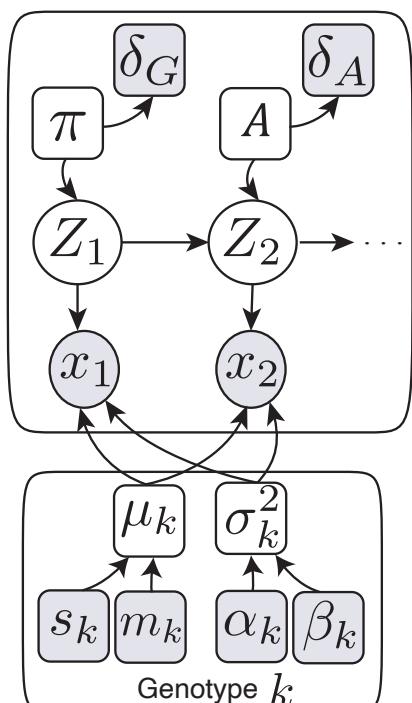
Office Hours with Anna-Lisa Doebley ([adoebley@uw.edu](mailto:adoebley@uw.edu))

Zoom Meeting ID: 446 356 7725      Password: GS541

- Wednesday, May 7, 2-3pm (Homework 5 questions)
- Monday, May 11, 2-3pm
- Wednesday, May 13, 2-3pm

# Extra Slides

- Continuous hidden Markov models (HMMs)
- Parameter inference using EM and copy number segmentation
- References:



- **ichorCNA** - Adalsteinsson\*, Ha\* Freeman\* et al. *Nature Communications* **8**:1324 (2017).
- **HMMcopy** - Ha et al. *Genome Research* **22**:1995-2007 (2012).
- **TitanCNA** - Ha et al. TITAN: inference of copy number architectures in clonal cell populations from tumor whole-genome sequencing data. *Genome Research* **24**:1881-1893 (2014).
- Murphy, K. (2012). Machine Learning: A Probabilistic Perspective. MIT Press. ISBN: 9780262018029
- Bishop, C. M. (2006). Pattern Recognition and Machine Learning (Information Science and Statistics). Springer. ISBN: 0387310738

# ichorCNA: Model inference using EM (extra slide 1)

**Complete data likelihood: joint distribution of latent and observed variables**

$$\begin{aligned} p(x_{1:T}, Z_{1:T} | \boldsymbol{\theta}) &= p(Z_1 | \pi_{1:K}) \left[ \prod_{t=2}^T p(Z_t | Z_{t-1}, A) \right] \prod_{t=1}^T p(x_t | Z_t, \mu, \sigma^2) \\ &= \prod_{k=1}^K \pi_k^{\mathbb{I}(Z_i=k)} \left[ \prod_{t=2}^T \prod_{k=1}^K \prod_{j=1}^K A_{jk}^{\mathbb{I}(Z_{t-1}=j)\mathbb{I}(Z_t=k)} \right] \prod_{t=1}^T \prod_{k=1}^K \mathcal{N}(x_t | \mu_k, \sigma_k^2)^{\mathbb{I}(Z_t=k)} \end{aligned}$$

where  $\boldsymbol{\theta} = \{\pi_{1:K}, \mu_{1:K}, \sigma_{1:K}^2, A\}$

**Complete data log likelihood**

$$\log p(x_{1:T}, Z_{1:T} | \boldsymbol{\theta}) = \sum_{k=1}^K \mathbb{I}(Z_i=k) \log \pi_k + \sum_{t=2}^T \sum_{j=1}^K \sum_{k=1}^K \mathbb{I}(Z_{t-1}=j, Z_t=k) \log A_{jk} + \sum_{t=1}^T \sum_{k=1}^K \mathbb{I}(Z_i=k) \log \mathcal{N}(x_t | \mu_k, \sigma_k^2)$$

**Expected complete data log likelihood**

$$Q = \sum_{k=1}^K \gamma(Z_1 = k) \log \pi_k + \sum_{t=2}^T \sum_{j=1}^K \sum_{k=1}^K \xi(Z_{t-1} = j, Z_t = k) \log A_{jk} + \sum_{t=1}^T \sum_{k=1}^K \gamma(Z_t = k) \log \mathcal{N}(x_t | \mu_k, \sigma_k^2)$$

# ichorCNA: Model inference using EM (extra slide 2)

E-Step: compute responsibilities using the forwards-backwards algorithm (Baum-Welch)

$$\gamma(\mathbf{Z}_t) = p(\mathbf{Z}_t | \mathbf{x}, \theta^{old}) = \frac{p(\mathbf{x} | \mathbf{Z}_t | \theta^{old}) p(\mathbf{Z}_t | \theta^{old})}{p(\mathbf{x} | \theta^{old})}$$

$$\gamma(\mathbf{Z}_t) = \frac{p(x_1, \dots, x_t, \mathbf{Z}_t) p(x_{t+1}, \dots, x_T | \mathbf{Z}_t)}{p(\mathbf{x})}$$

$$\gamma(\mathbf{Z}_t) = \frac{\alpha(\mathbf{Z}_t)\beta(\mathbf{Z}_t)}{p(\mathbf{x})} \quad \begin{matrix} \text{Responsibilities} \\ \text{Matrix } K \times T \end{matrix}$$

Where  $\alpha(Z_t = k) = \mathcal{N}(x_t | Z_t = k) \sum_{j=1}^K \left\{ A_{jk} \alpha(Z_t = j) \right\}$  is the forward recursion probability

Where  $\beta(Z_t = k) = \sum_{j=1}^K \left\{ \mathcal{N}(x_{t+1} | Z_{t+1} = j) A_{kj} \alpha(Z_{t+1} = j) \right\}$  is the backward recursion probability

$$\xi(\mathbf{Z}_{t-1}, \mathbf{Z}_t) = p(\mathbf{x} | \mathbf{Z}_{t-1}, \mathbf{Z}_t) P(\mathbf{Z}_{t-1}, \mathbf{Z}_t)$$
$$\xi(\mathbf{Z}_{t-1}, \mathbf{Z}_t) = \frac{\alpha(\mathbf{Z}_{t-1}) p(x_t | \mathbf{Z}_t) p(\mathbf{Z}_t | \mathbf{Z}_{t-1}) \beta(\mathbf{Z}_t)}{p(\mathbf{x})}$$

**Likelihood function**       $\ell = \log p(\mathbf{x}) = \sum_{t=1}^T \log \left( \sum_{k=1}^K \alpha(Z_t = k) \right)$

**Forward Probabilities**  
Matrix  $K \times T$

**Backward Probabilities**  
Matrix  $K \times T$

**2 Slice Marginals**  
Matrix  $K \times K \times (T - 1)$

Chapter 13 in Bishop (2006).  
Pattern Recognition and Machine  
Learning. Springer

# ichorCNA: Model inference using EM (extra slide 3)

## M-Step: Update the parameters given the responsibilities

$$\text{Prior}(\pi_{1:K}, \mu_{1:K}, \sigma_{1:K}^2, A) = \prod_{k=1}^K \text{Dir}(\pi_k | \delta_k) \text{Dir}(A_k | \delta_A) \mathcal{N}(\mu_k | \alpha, \beta) \text{InvGamma}(\sigma_k^2 | \alpha_k, \beta_k) \quad \text{Priors}$$
$$\mathcal{O} = Q + \log \mathbb{P}(\pi_{1:K}, \mu_{1:K}, \sigma_{1:K}^2, A) \quad \text{Complete data log likelihood} \\ + \log \text{priors}$$

- The object function  $\mathcal{O}$  is used to obtain the update equations for  $\pi_{1:K}$  and  $\mu_{1:K}$

$$\frac{\partial \mathcal{O}}{\partial \pi_k} = 0, \text{ find } \hat{\pi}_k \quad \text{MAP for initial state distribution}$$

$$\frac{\partial \mathcal{O}}{\partial \mu_k} = 0, \text{ find } \hat{\mu}_k \quad \text{MAP for Gaussian means}$$

$$\frac{\partial \mathcal{O}}{\partial \sigma_k^2} = 0, \text{ find } \hat{\sigma}_k^2 \quad \text{MAP for Gaussian variance}$$

$$\frac{\partial \mathcal{O}}{\partial A_{jk}} = 0, \text{ find } \hat{A}_{jk} \quad \text{MAP for transition probabilities}$$

**EM Convergence:** after each iteration, monitor the log posterior

$$\ell = \log p(x) = \sum_{t=1}^T \log \left( \sum_{k=1}^K \alpha(Z_t = k) \right) \quad \text{Incomplete Data Log likelihood}$$

$$\log \mathbb{P} = \ell + \log \text{Prior}(\pi_{1:K}, \mu_{1:K}, \sigma_{1:K}^2, A) \quad \text{Log posterior}$$