

CANCER GENOMICS

Lecture 2:

Probabilistic Methods for Mutation Detection

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

Outline

1. Detecting Mutations in Cancer Genomes

- Visualizing somatic vs germline SNVs
- Sequencing read count data

2. Primer on statistical modeling (cont'd)

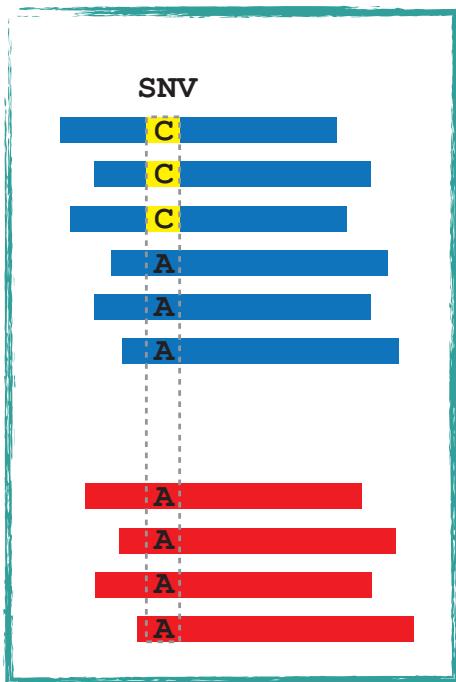
- Mixture models, inference and parameter estimation using the EM algorithm

3. Mixture Models for SNV Detection

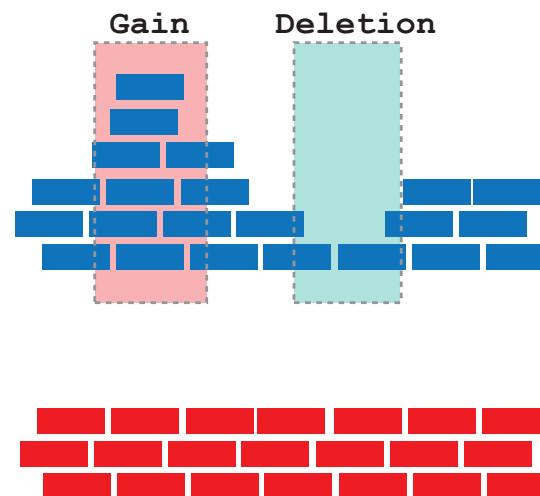
- SNV genotyping strategy
- SNVMix probabilistic model and EM inference
- Predicting somatic SNVs in cancer

1. Detecting Mutations in Cancer Genomes

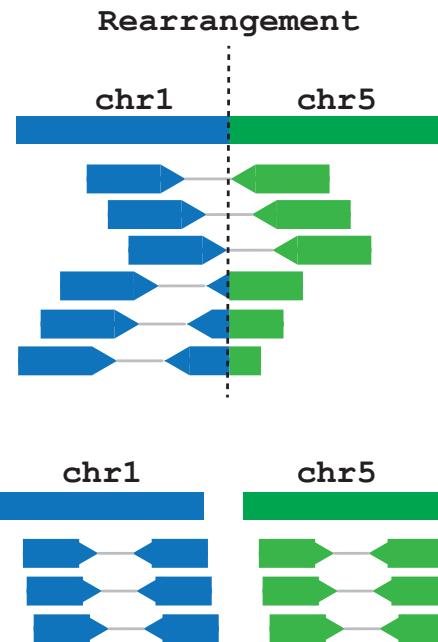
Mutations (SNV, INDEL)



Copy Number Alterations



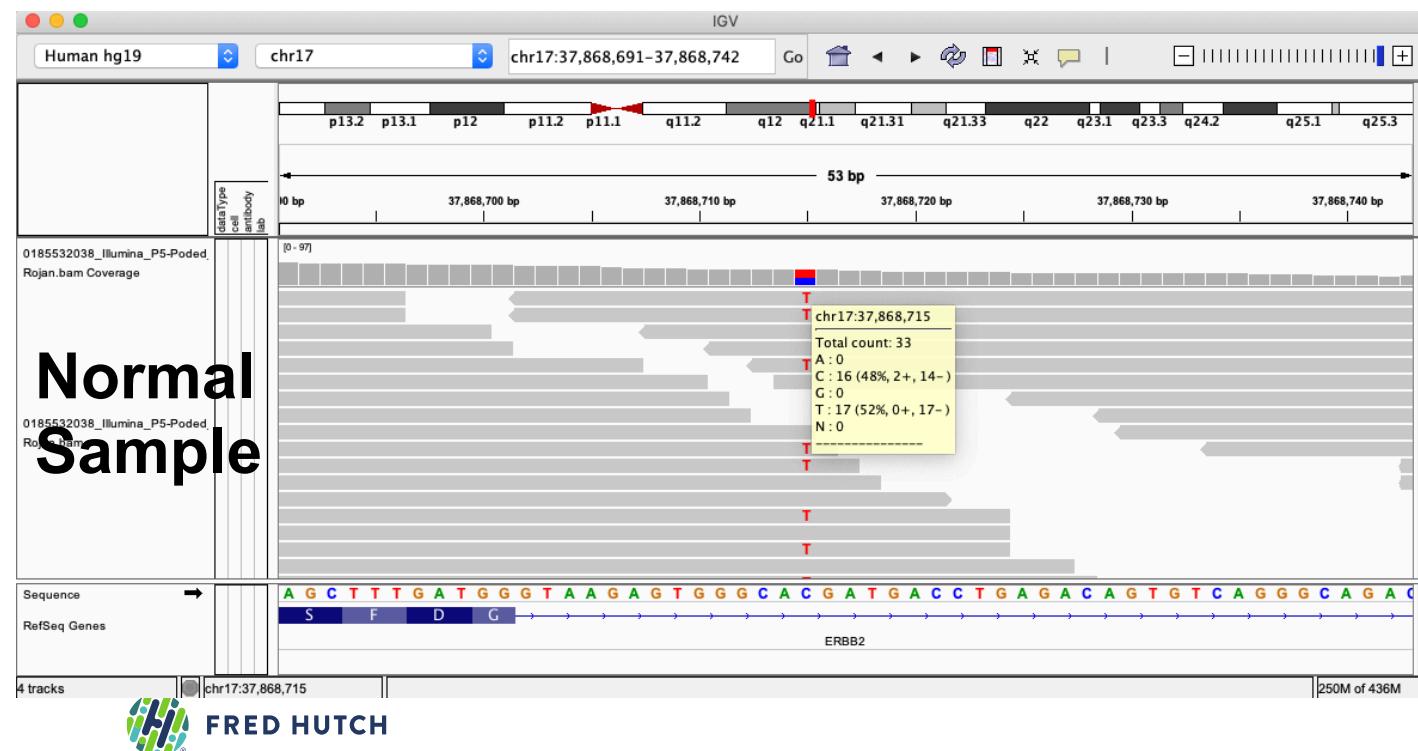
Structural Variants



Visual inspection using IGV: Germline SNVs

Integrative Genomics Viewer (<https://software.broadinstitute.org/software/igv>)

- ~1.5 to 2 million **SNPs** per individual
- Identify SNPs from normal peripheral blood mononuclear cells (PBMC)



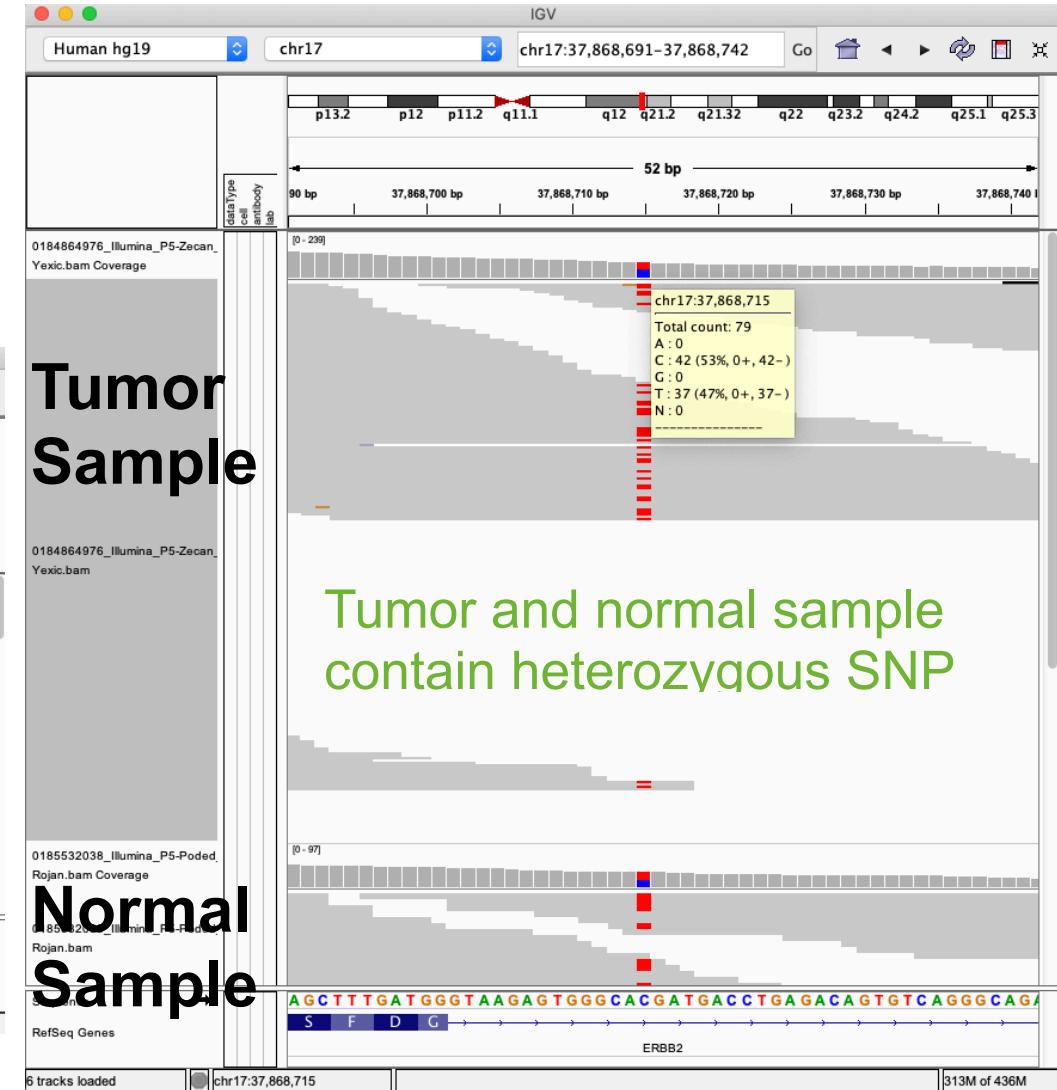
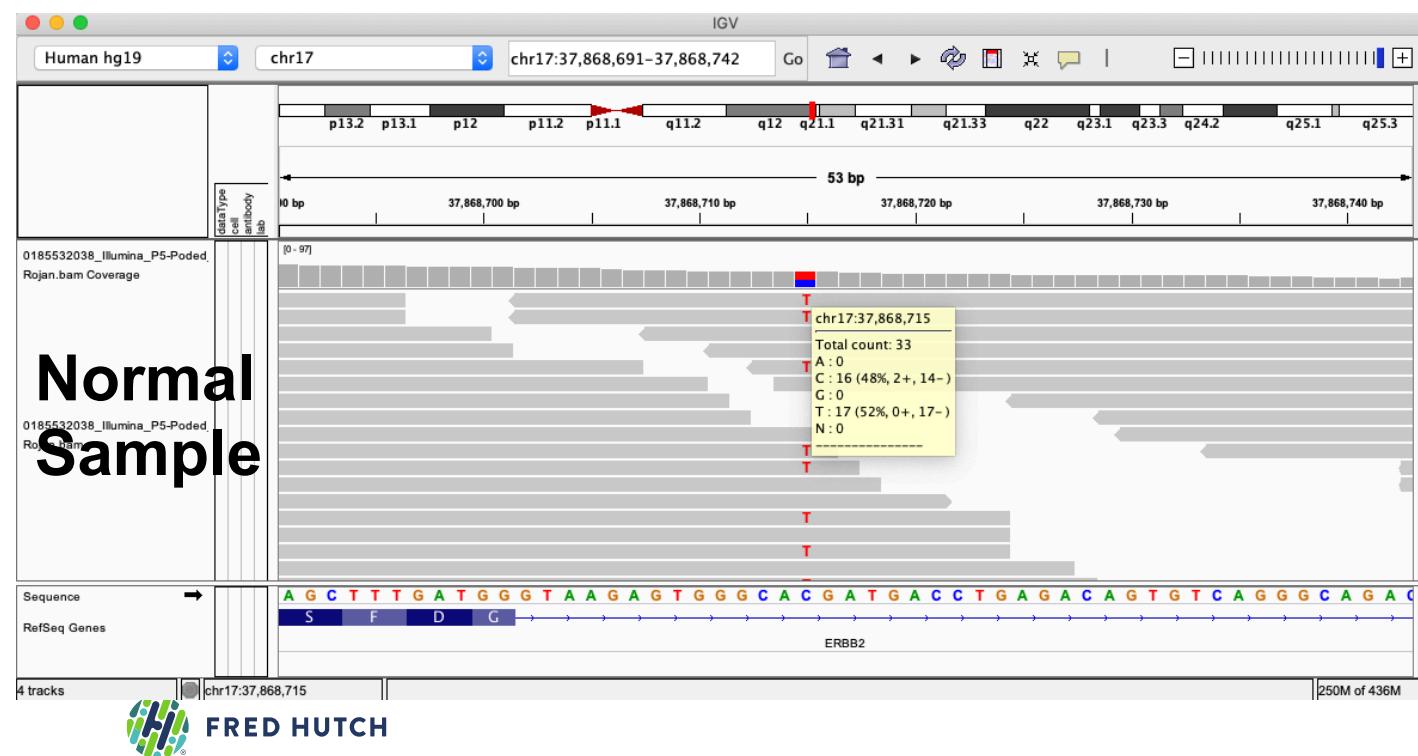
Heterozygous SNP with 37 reads containing the variant and having depth 79 reads

37/79 (47%) variant allele fraction (VAF)

Visual inspection using IGV: Germline SNVs

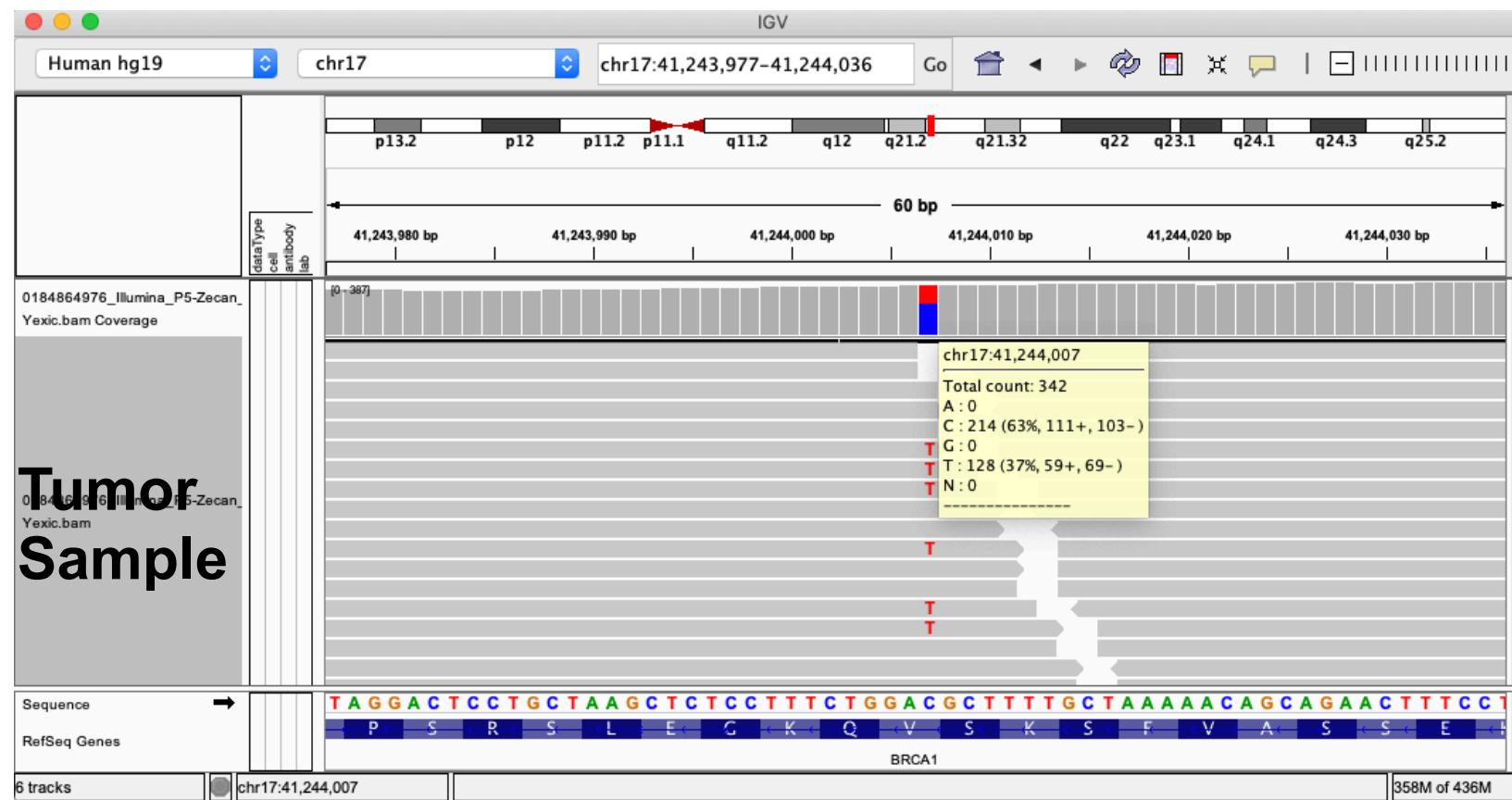
Integrative Genomics Viewer (<https://software.broadinstitute.org/software/igv>)

- ~1.5 to 2 million **SNPs** per individual
- Identify SNPs from normal peripheral blood mononuclear cells (PBMC)



Visual inspection using IGV: Somatic SNVs

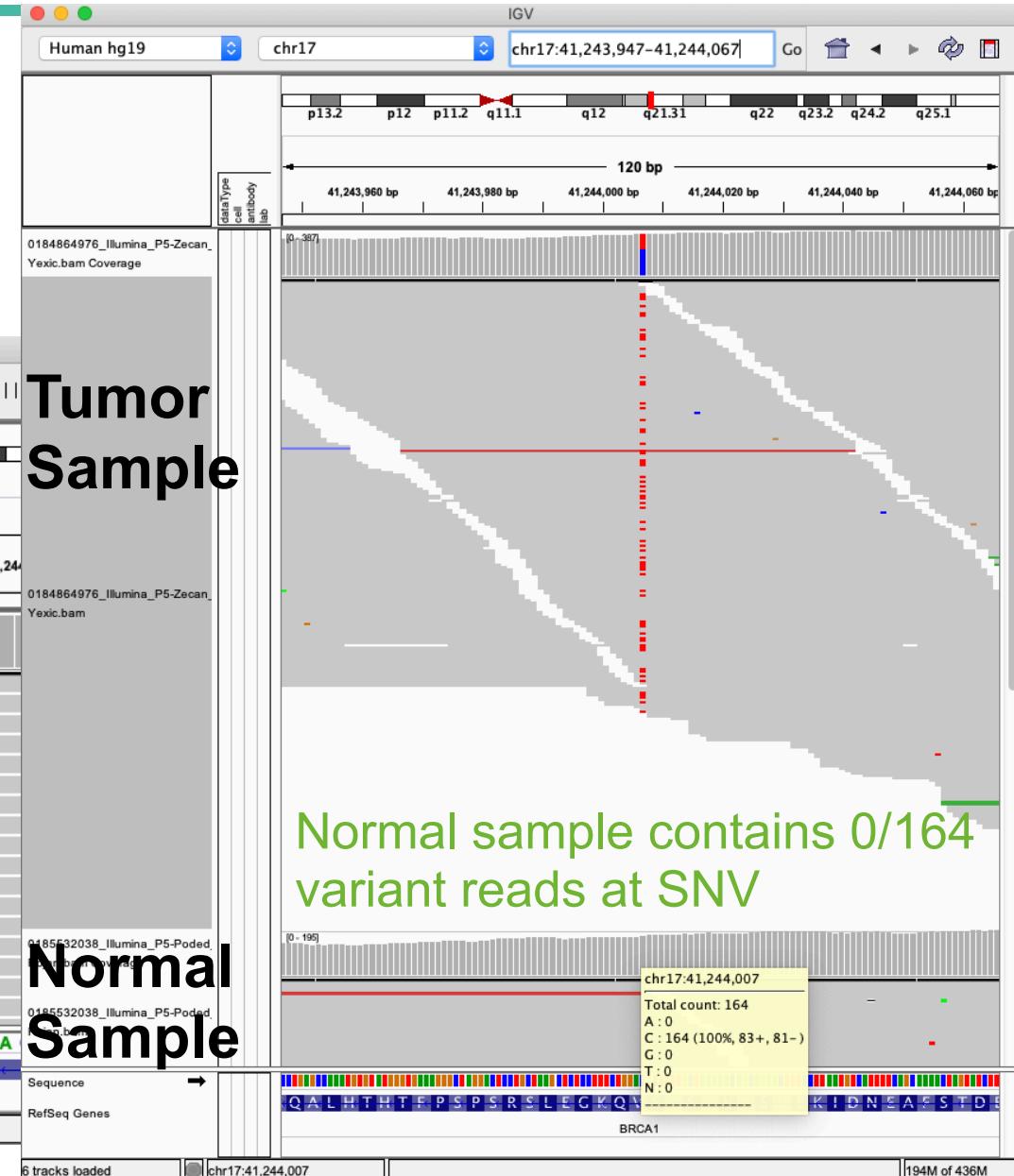
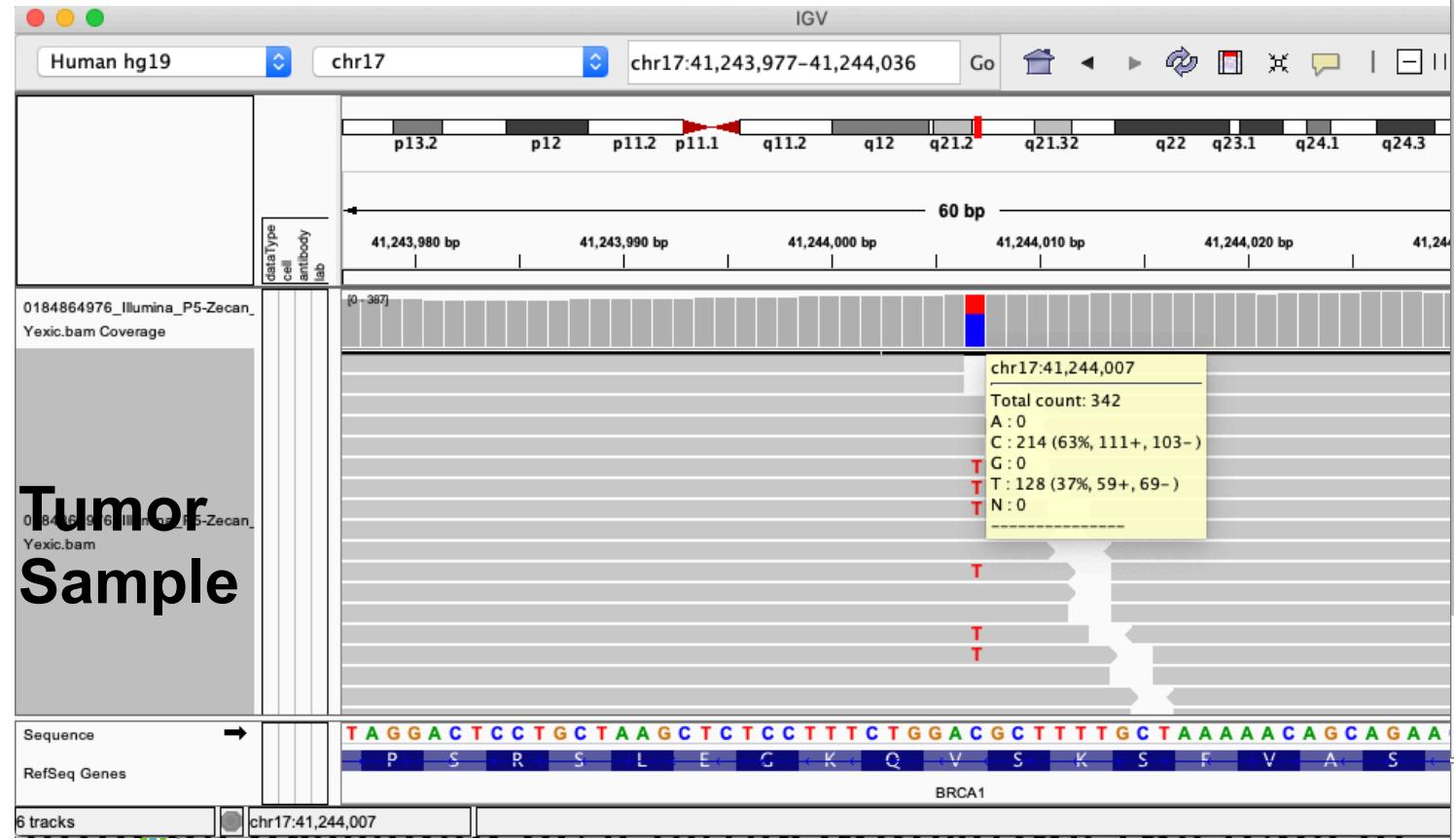
- Somatic **SNV** requires comparing case (tumor) with control (PBMC)
- On the order of 10 to 10^4 number of mutations



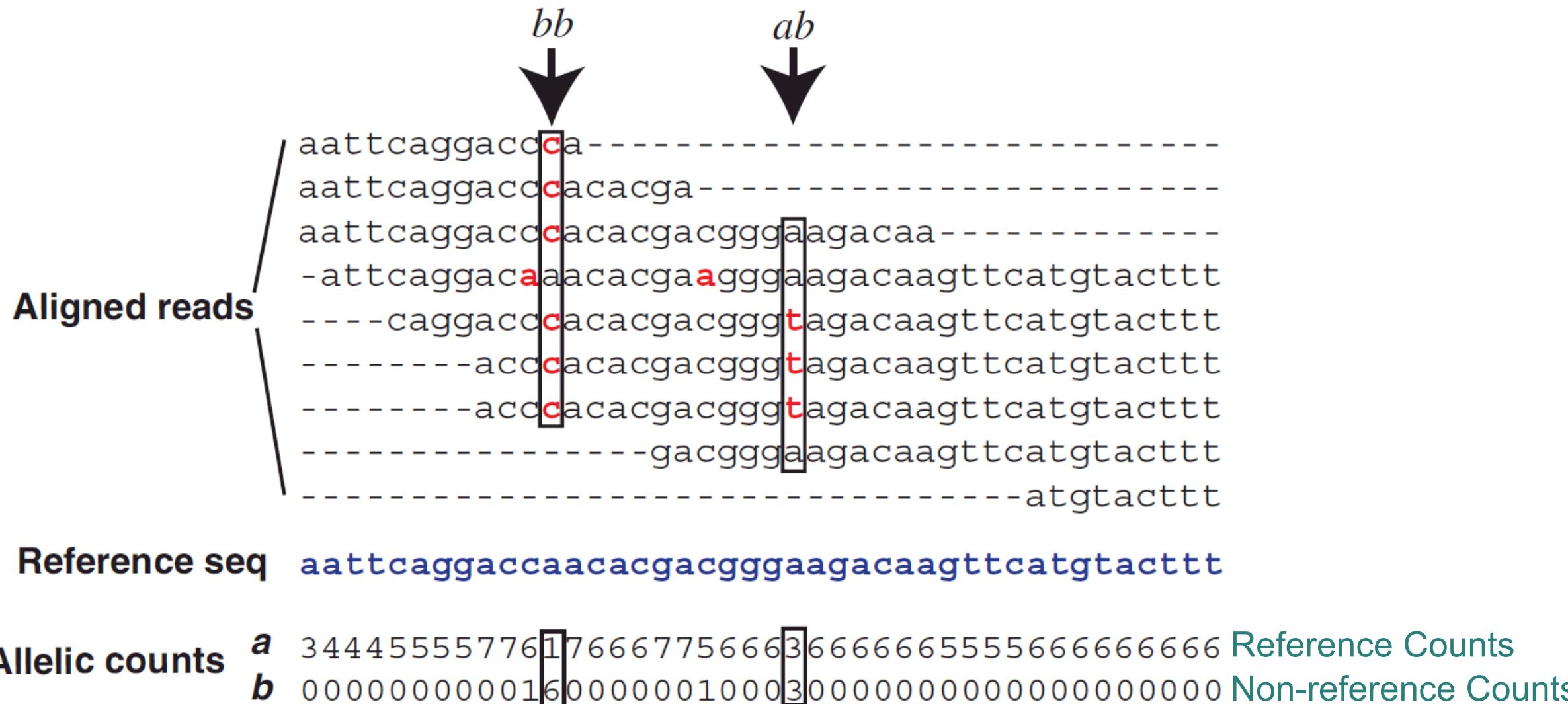
Potential SNV with
128/342 (37%) VAF
p.V118I

Visual inspection using IGV: Somatic SNVs

- Somatic **SNV** requires comparing case (tumor) with control (PBMC)
- On the order of 10 to 10^4 number of mutations



Single Nucleotide Variant (SNV) Calling: Single Sample

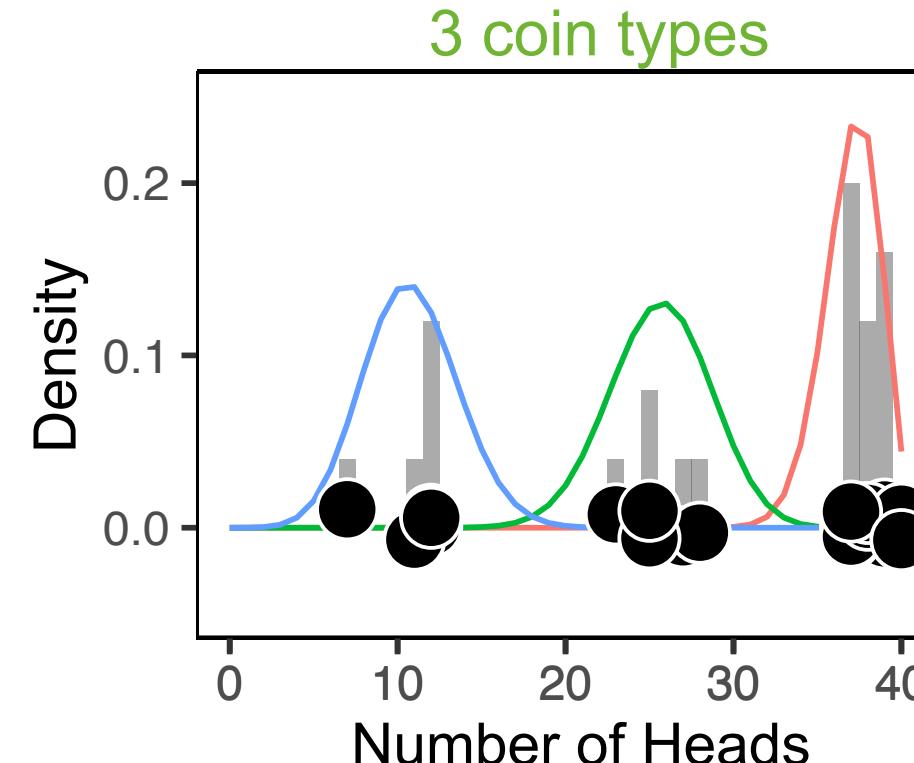
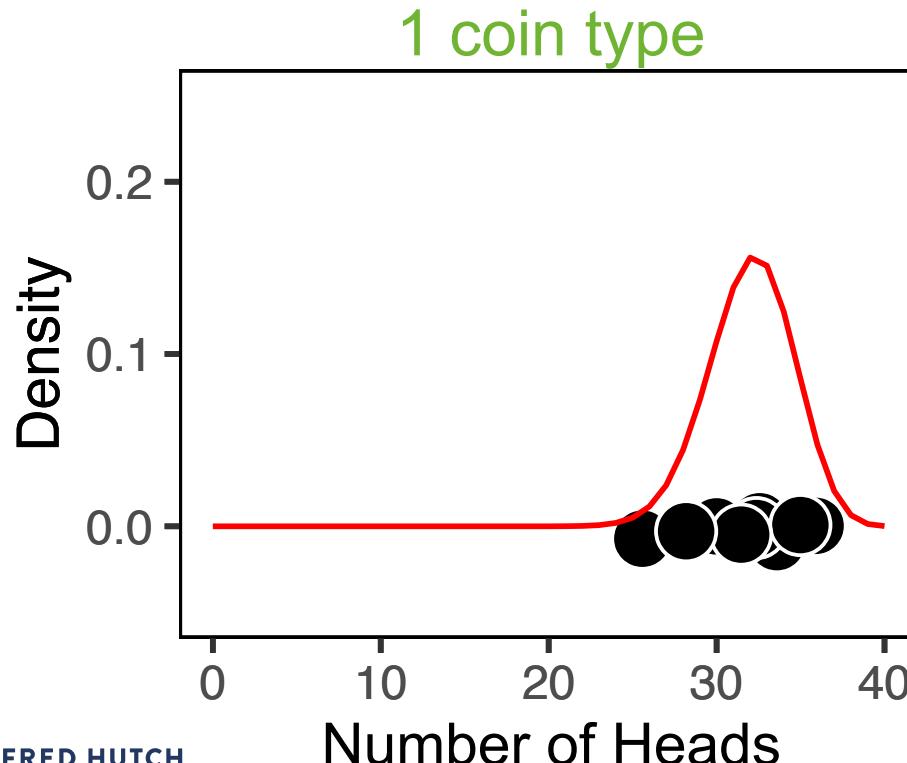


2. Primer on statistical modeling (cont'd)

- Probability
 - Unsupervised learning, probability rules & Bayes' theorem
 - Binomial distribution, Bayesian statistics
 - Beta-binomial model example
- **Mixture models, EM inference & parameter learning**
- References:
 - 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

Mixture Model: Referee example with multiple coins

- Recall: There are T different referees who tossed the same coin $N = \{1, \dots, N_T\}$ times and came up with counts of heads $x = \{1, \dots, x_T\}$.
- Now suppose there are **3 types of coins**: (1) probably fair, (2) unfairly favors heads, (3) unfairly favors tails denoted as $\{\text{fair}, \text{heads}, \text{tails}\}$.
- Each referee **draws one coin** (with replacement) from a hat containing these coin types mixed together.



Mixture Model: Referee example with multiple coins

- Recall: There are $T = 8$ different referees who tossed the *same* coin $N = \{1, \dots, N_T\}$ times and came up with counts of heads $x = \{1, \dots, x_T\}$.
- Now suppose there are **3 types of coins**: (1) probably fair, (2) unfairly favors heads, (3) unfairly favors tails denoted as $\{\text{fair}, \text{heads}, \text{tails}\}$.
- Each referee **draws one coin from a hat** that contains a bunch of these coin types mixed together.
 - We don't know the proportion of each coin type in the hat.
 - We don't know which coin each referee drew from the hat.
 - We don't know the fairness (probability of heads) for each type of coin.

Referee	# of tosses (N)	# of heads (x)	Prop. of heads	Type of coin used?
Referee 1	40	25	0.63	?
Referee 2	42	35	0.83	?
Referee 3	39	27	0.69	?
Referee 4	x_T	N_T	x_T/N_T	?

Coin Type	Proportion in hat	Prob. of heads
“Fair”	?	?
“Heads”	?	?
“Tails”	?	?

Mixture Model: Latent state model

1. What is the proportion of each coin type in the hat?

Find the probability for using each coin type.

- π_k is the probability of drawing coin type k
- $\boldsymbol{\pi} = (\pi_{fair}, \pi_{heads}, \pi_{tails})$ where $\sum_{k=1}^K \pi_k = 1$

2. Which coin did each referee draw?

Define the latent variables.

- Let $Z_i = k$ be the type of coin, $k \in \{fair, heads, tails\}$, that referee i draws
- Z_i is called a **latent variable** and follows a *Categorical* distribution with parameter $\boldsymbol{\pi}$

$$p(Z_i = k | \boldsymbol{\pi}_{1:K}) = \text{Cat}(Z_i = k | \boldsymbol{\pi}_{1:K}) \\ = \begin{cases} \pi_{fair} & \text{if } k = fair \\ \pi_{heads} & \text{if } k = heads \\ \pi_{tails} & \text{if } k = tails \end{cases}$$

- The proportions $\boldsymbol{\pi}_{1:K}$ of the coin types follows a Dirichlet distribution (conjugate prior)

$$p(\boldsymbol{\pi}_{1:K} | \boldsymbol{\delta}_{1:K}) = \text{Dirichlet}(\boldsymbol{\pi}_{1:K} | \boldsymbol{\delta}_{1:K})$$

Coin Type	Proportion in hat	Prob. of heads
“Fair”	π_{fair}	?
“Heads”	π_{heads}	?
“Tails”	π_{tails}	?

Referee	Type of coin used?
Referee 1	Z_1
Referee 2	Z_2
Referee 3	Z_3
Referee T	Z_T

Mixture Model: Likelihood as a mixture of binomials

3. What is the fairness (prob. of heads) for each type of coin?

Find the probability of heads for each coin type.

- Recall: for a single coin, $p(x_i | N_i, \mu) = \text{Bin}(x_i | N_i, \mu)$
- Define the likelihood for a **3-component mixture of binomials** with 3 parameters, $\mu_{\text{fair}}, \mu_{\text{heads}}, \mu_{\text{tails}}$, one for each type of coin

Coin Type	Proportion in hat	Prob. of heads
“Fair”	π_{fair}	μ_{fair}
“Heads”	π_{heads}	μ_{heads}
“Tails”	π_{tails}	μ_{tails}

$$p(x_i | Z_i = k, N_i) = \text{Bin}(x_i | N_i, \mu_k)$$

Observed likelihood

$$p(x_i | N_i, \mu_{1:K}, \pi_{1:K}) = \sum_{k=1}^K \pi_k \text{Bin}(x_i | N_i, \mu_k)$$

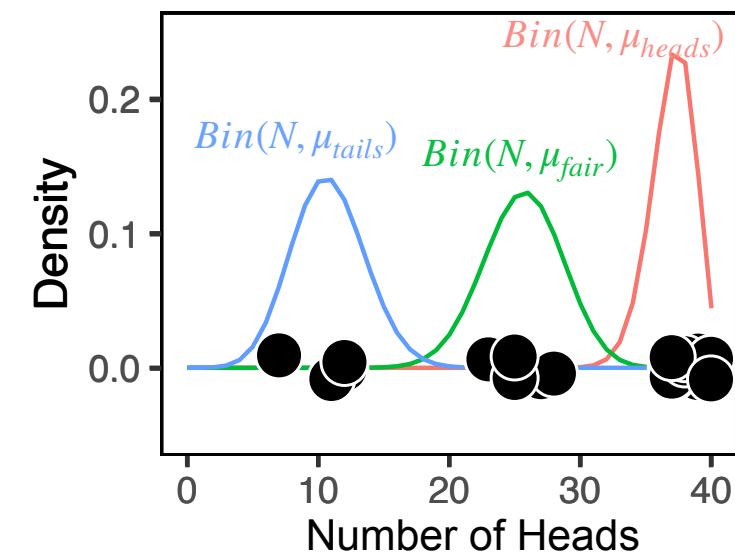
Mixture model

$$L(x_{1:T}, N_{1:T} | \mu_{1:K}, \pi_{1:K}) = \prod_{i=1}^T \sum_{k=1}^K \pi_k \text{Bin}(x_i | N_i, \mu_k)$$

Likelihood function

$$\ell = \sum_{i=1}^T \log \left(\sum_{k=1}^K \pi_k \text{Bin}(x_i | N_i, \mu_k) \right)$$

Log likelihood



Mixture Model: Inference & parameter estimation using EM (1)

Expectation-Maximization: Inference and parameter training

Initialize parameters: $\pi_{1:K}$ and $\mu_{1:K}$

E-Step: compute “responsibilities” (inference)

1. Which coin did each referee draw? (Posterior of the latent states $\gamma(Z_{1:T})$)

- Soft-clustering: Referee i has a probability for using each of the coins.
- responsibilities: “coin that is responsible for generating observation x_i ”

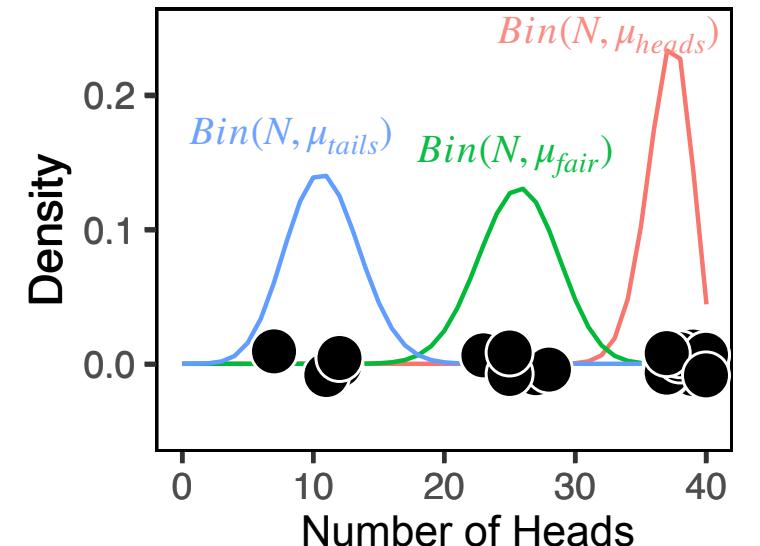
M-Step: Update parameters (learning)

2. What is the proportion of each coin type in the hat? $\pi_{1:K}$

3. What is the fairness (prob. of heads) for each coin type? $\mu_{1:K}$

Iterate between E-Step and M-Step, check when log-likelihood ℓ stops increasing.

Responsibilities			
Referee	Fair Coin	Heads Coin	Tails Type Coin
1	$\gamma(Z_1 = F)$	$\gamma(Z_1 = H)$	$\gamma(Z_1 = T)$
2	$\gamma(Z_2 = F)$	$\gamma(Z_2 = H)$	$\gamma(Z_2 = T)$
3	$\gamma(Z_3 = F)$	$\gamma(Z_3 = H)$	$\gamma(Z_3 = T)$
T	$\gamma(Z_T = F)$	$\gamma(Z_T = H)$	$\gamma(Z_T = T)$



Chapter 9 in Bishop (2006). Pattern Recognition and Machine Learning.
Springer

Section 3.3, 3.4, 11.2 in Murphy (2012).
Machine Learning: A Probabilistic Perspective. MIT Press

Mixture Model: Inference & parameter estimation using EM (2)

E-Step: compute responsibilities (inference)

1. What is the probability for a referee to draw each coin type? (Posterior of the latent states $Z_{1:T}$)

- Find the responsibilities given the current parameters

$$\begin{aligned} p(Z_i = k | x_i, N_i, \pi_{1:K}, \mu_{1:K}) &= \frac{p(x_i | Z_i = k)p(Z_i = k)}{p(x_i)} \\ &= \frac{\pi_k \text{Bin}(x_i | N_i, \mu_k)}{\sum_{k'=1}^K \pi_{k'} \text{Bin}(x_i | N_i, \mu_{k'})} \\ &= \gamma(Z_i = k) \end{aligned}$$

Bayes' Rule
Posterior distribution
of the latent variables

Responsibilities
Matrix $T \times K$

- Responsibilities = “coin that is responsible for generating observation x_i ”
- Soft-clustering: Referee i has a probability for using each of the coins.
- $\gamma(Z_{1:T})$ is a matrix of probabilities with dimensions $T \times K$

Mixture Model: Inference & parameter estimation using EM (3)

M-Step: Update parameters (learning)

2. What is the proportion of each coin type in the hat?

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

MAP for π

3. What is the fairness (prob. of heads) for each coin type?

$$\hat{\mu}_k = \frac{\sum_{i=1}^T \gamma(Z_i = k)x_i + \alpha_k - 1}{\sum_{i=1}^T \gamma(Z_i = k)N_i + \alpha_k + \beta_k - 2}$$

MAP for μ

Evaluate the log likelihood and log posterior: use updated parameters

$$\text{Log posterior} \quad \log \mathbb{P} = \sum_{i=1}^T \log \left(\sum_{k=1}^K \hat{\pi}_k \text{Bin}(x_i | N_i, \hat{\mu}_k) \right) + \log \text{Dir}(\hat{\pi} | \delta) + \sum_{k=1}^K \log \text{Beta}(\hat{\mu}_k | \alpha_k, \beta_k)$$

Log likelihood Log priors

Iterate between E-Step and M-Step:

- Stop EM when new $\log \mathbb{P}$ changes less than ϵ compared to previous EM iteration.

Algorithm 1 Binomial Mixture Model Inference and Learning using EM

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

Mixture Model: Inference & parameter estimation using EM (extra slide 1)

Incomplete data log likelihood

$$L(x_{1:T}, N_{1:T} | \mu_{1:K}, \pi_{1:K}) = \prod_{i=1}^T \sum_{k=1}^K \pi_k \text{Bin}(x_i | N_i, \mu_k)$$

- The incomplete data log likelihood (plus the priors) is used to monitor EM convergence

Expected complete data log likelihood

Complete data likelihood

$$L(\mu_{1:K}, \pi_{1:K} | x_{1:T}, Z_{1:T}, N_{1:T}) = \prod_{i=1}^T \prod_{k=1}^K \pi_k \text{Bin}(x_i | N_i, \mu_k)^{\mathbb{I}(Z_i=k)}$$

Complete data log likelihood

$$\ell(\mu_{1:K}, \pi_{1:K} | x_{1:T}, Z_{1:T}, N_{1:T}) = \sum_{i=1}^T \sum_{k=1}^K \mathbb{I}(Z_i=k) \{ \log \pi_k + \log \text{Bin}(x_i | N_i, \mu_k) \}$$

Expected complete data log likelihood

$$\begin{aligned} Q &= \mathbb{E} [\ell(\mu_{1:K}, \pi_{1:K} | x_{1:T}, Z_{1:T}, N_{1:T})] = \sum_{i=1}^T \sum_{k=1}^K \mathbb{E} [\mathbb{I}(Z_i=k)] \{ \log \pi_k + \log \text{Bin}(x_i | N_i, \mu_k) \} \\ &= \sum_{i=1}^T \sum_{k=1}^K \gamma(Z_i=k) \{ \log \pi_k + \log \text{Bin}(x_i | N_i, \mu_k) \} \end{aligned}$$

- The expected complete data log likelihood is in the M-Step for updating parameters.

Mixture Model: Inference & parameter estimation using EM (extra slide 2)

M-Step: Update the parameters given the responsibilities

$$\mathbb{P}(\pi_{1:K}, \mu_{1:K}) = Dir(\boldsymbol{\pi} | \boldsymbol{\delta}) \prod_{k=1}^K Beta(\mu_k | \alpha, \beta) \quad \text{Priors}$$

$$\mathcal{O} = Q + \log \mathbb{P}(\pi_{1:K}, \mu_{1:K}) \quad \text{Complete data log likelihood} \\ + \log \text{priors}$$

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

$$\frac{\partial \mathcal{O}}{\partial \mu_k} = 0, \text{ find } \hat{\mu}_k \quad \text{and} \quad \frac{\partial \mathcal{O}}{\partial \pi_k} = 0, \text{ find } \hat{\pi}_k$$

EM Convergence: after each iteration, monitor the log posterior

$$\ell = \sum_{i=1}^T \log \left(\sum_{k=1}^K \pi_k Bin(x_i | \mu_k, N_i) \right) \quad \text{Incomplete Data Log likelihood}$$

$$\log \mathbb{P}(\pi_{1:K}, \mu_{1:K} | x_{1:T}) = \ell + \log \mathbb{P}(\pi_{1:K}, \mu_{1:K}) \quad \text{Log posterior}$$

- If the log posterior, $\log \mathbb{P}(\pi_{1:K}, \mu_{1:K} | x_{1:T})$, stops increasing by ϵ , then EM is converged.
- If not using a Bayesian framework, then use the log likelihood, ℓ , to monitor convergence.

Mixture Models: Online Tutorial and Resource

fiveMinuteStats (<https://stephens999.github.io/fiveMinuteStats/>)

by **Dr. Matthew Stephens**, Professor in Statistics & Human Genetics at University of Chicago

1. Introduction to mixture models with probabilistic derivations and R code

- Examples with Bernoulli and Gaussian models
- https://stephens999.github.io/fiveMinuteStats/intro_to_mixture_models.html

2. Introduction to EM with Gaussian Mixture Model example and R code

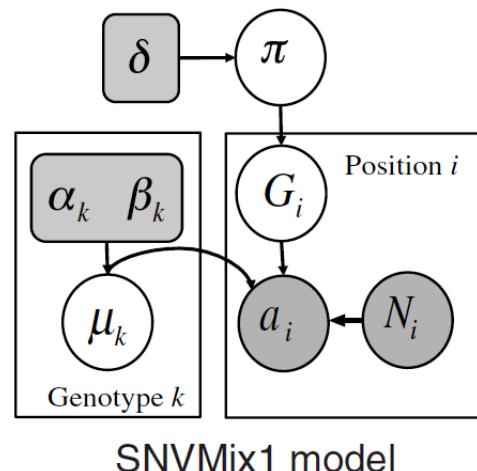
- https://stephens999.github.io/fiveMinuteStats/intro_to_em.html

3. Mixture Model for SNV Detection

- SNV genotyping strategy
- SNVMix probabilistic model and EM inference
- Predicting somatic SNVs in cancer

References:

- Goya et al. **SNVMix**: predicting single nucleotide variants from next-generation sequencing of tumors. *Bioinformatics* **26**:730-36 (2010)
- Roth et al. **JointSNVMix**: a probabilistic model for accurate detection of somatic mutations in normal/tumour paired next-generation sequencing data. *Bioinformatics* **28**:907-13 (2012)



SNV Genotyping

Variant Allele Fraction Analysis

- Single sample

Genotypes: AA , AB , BH

The diagram illustrates the relationship between three states: Homozygous Reference (not SNV), Heterozygous Variant (Het SNV), and Homozygous Variant (Homd SNV). A curved green arrow points from the Homozygous Reference state to the Heterozygous Variant state. Another curved green arrow points from the Heterozygous Variant state back to the Homozygous Reference state. A third curved green arrow points from the Heterozygous Variant state to the Homozygous Variant state. Additionally, a vertical green arrow points upwards from the Homozygous Variant state to a blank space above it.

Homozygous Reference (not SNV)	Heterozygous Variant (Het SNV)	Homozygous Variant (Homd SNV)
-----------------------------------	-----------------------------------	----------------------------------

Genotype	AA	AB	BB
Allelic Fraction	~1.0	~0.5	~0

Aligned reads

bb

ab

aattcaggacc c a-----	-----
aattcaggacc c acacg-----	-----
aattcaggacc c acacgacggg a agacaa-----	-----
-attcaggac a aacacg a gggaagacaagttcatgtacttt	-----
-----caggacc c acacgacggg t agacaagttcatgtacttt	-----
-----acc c acacgacggg t agacaagttcatgtacttt	-----
-----acc c acacgacggg t agacaagttcatgtacttt	-----
-----gacggg a agacaagttcatgtacttt	-----atgtacttt

Reference seq aattcaggaccaacacgacggaaagacaagttcatgtacttt

Allelic counts **a** 3444555577617666775666366666655556666666666
b 00000000001600000010003000000000000000000000000

SNVMix. Goya et al. *Bioinformatics* **26**:703-36 (2010)

- Allelic Fraction is defined as the fraction of reference reads, $\frac{A}{N}$, where depth $N = A + B$
 - Allelic Fractions in the table are the expected values for each genotype and used as ***initial values*** for the probability of observing a ***reference base*** μ_{AA} , μ_{AB} , μ_{BB}
 - Why estimate $\mu_{1..K}$? Why might the observed allelic fractions be different than the expected values?

Mapping the Referee Example to Mutation Calling

Referee Coin Toss Example

Data

Referees $1, \dots, T$

For each Referee i

- Coin Tosses: N_i
- Count of heads: x_i
- Count of tails: $N_i - x_i$

Parameters

Probability to draw coins: π_{fair} , π_{heads} , π_{tails}

Probability of heads for 3 types of coins

$$\mu_{fair}, \mu_{heads}, \mu_{tails}$$

Responsibilities

Probability that Referee i used coin k : $\gamma(Z_i = k)$

Mutation Calling from Sequencing Data

Data

Genomic loci $1, \dots, T$

For each locus i

- Depth (total reads): N_i
- Count of reference base: x_i
- Count of variant base: $N_i - x_i$

Parameters

Probability of genotypes: π_{AA} , π_{AB} , π_{BB}

Probability of reference base for 3 genotypes:

$$\mu_{AA}, \mu_{AB}, \mu_{BB}$$

Responsibilities

Probability that locus i has genotype k : $\gamma(Z_i = k)$

SNVMix: Probabilistic Model

Sequence Data

There are T different genomic loci with read depths $N = \{1, \dots, N_T\}$ and reference base counts $x = \{1, \dots, x_T\}$.

There are $K = 3$ different possible genotypes AA , AB , BB

Mixture Model Setup

1. The probabilities for the genotypes are π_{AA} , π_{AB} , π_{BB}
2. Thus, a specific genotype $k \in AA, AB, BB$ can be assigned to the latent state Z_i at locus i with these probabilities

$$p(Z_i = k | \pi_{1:K}) = \begin{cases} \pi_{AA} & \text{if } k = AA \\ \pi_{AB} & \text{if } k = AB \\ \pi_{BB} & \text{if } k = BB \end{cases}$$

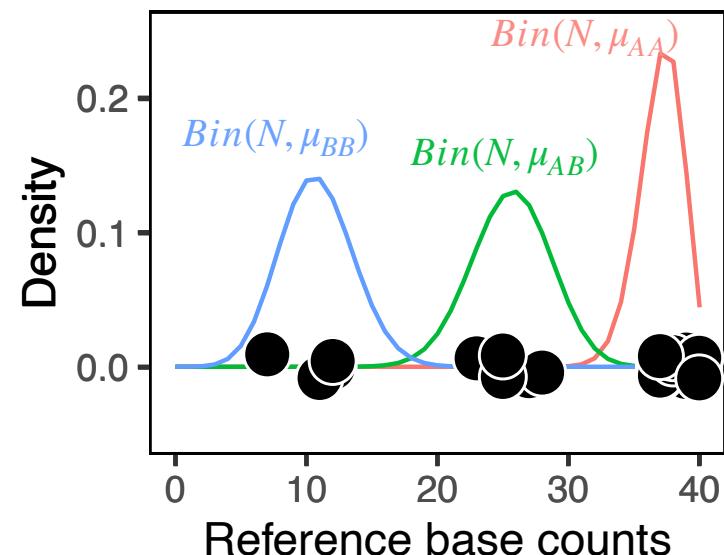
3. The probability of observing a reference base for each of the 3 genotypes are μ_{aa} , μ_{ab} , μ_{bb}
4. The likelihood is a **3-component mixture of binomials** with the 3 parameters, μ_{aa} , μ_{ab} , μ_{bb} , one for each genotype

$$p(x_i | N_i, \mu_{1:K}, \pi_{1:K}) = \sum_{k=1}^K \pi_k \text{Bin}(x_i | N_i, \mu_k)$$

5. The priors for genotype $k \in \{aa, ab, bb\}$ in the model are

$$p(\pi_{1:K} | \delta_{1:K}) = \text{Dirichlet}(\pi_{1:K} | \delta_{1:K})$$

$$p(\mu_k | \alpha_k, \beta_k) = \text{Beta}(\mu_k | \alpha_k, \beta_k)$$



SNVMix: Inference & parameter estimation using EM (revisited)

E-Step: compute responsibilities

1. What is the probability of locus i having genotype k ?

$$\gamma(Z_i = k) = \frac{\pi_k \text{Bin}(x_i | N_i, \mu_k)}{\sum_{j=1}^K \pi_j \text{Bin}(x_i | N_i, \mu_j)}$$

M-Step: maximize parameters

2. What is the probability of genotype k ?

$$\hat{\pi}_k = \frac{\sum_{i=1}^T \gamma(Z_i = k) + \delta(k) - 1}{\sum_{j=1}^K \left\{ \sum_{i=1}^T \gamma(Z_i = j) + \delta(j) - 1 \right\}}$$

Responsibilities
Matrix $T \times K$

MAP for π

3. What is the probability of observing a reference base for genotype k ?

$$\hat{\mu}_k = \frac{\sum_{i=1}^T \gamma(Z_i = k) x_i + \alpha_k - 1}{\sum_{i=1}^T \gamma(Z_i = k) N_i + \alpha_k + \beta_k - 2}$$

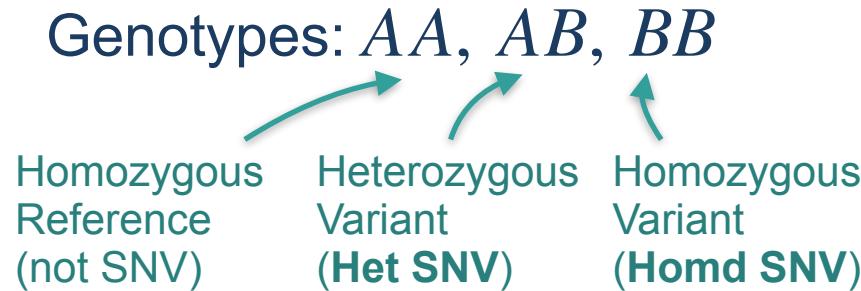
MAP for μ

Evaluate the log likelihood and log posterior: use updated parameters

$$\log \mathbb{P} = \sum_{i=1}^T \log \left(\sum_{k=1}^K \hat{\pi}_k \text{Bin}(x_i | \hat{\mu}_k, N_i) \right) + \log \text{Dir}(\hat{\pi}_k | \delta_k) + \sum_{k=1}^K \log \text{Beta}(\hat{\mu}_k | \alpha_k, \beta_k) \quad \text{Log posterior}$$

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

SNVMix: Calling somatic SNVs from genotype inference



- To call a variant for each locus i , we can apply a threshold on the responsibilities $\gamma(Z_i)$
- We can sum $\gamma(Z_i = AB)$ and $\gamma(Z_i = BB)$ to get the overall probability (either genotype AB or BB) that locus i is a variant containing the non-reference allele (B)
- Additional steps required for filtering and determining if variant is somatic vs germline
 - Minimum 3 variant reads ($N_i - x_i$) is typically required
 - Account for mapping and base qualities of sequenced reads (i.e. SNVMix2)
 - Compare locus i in tumor sample to (1) matched normal sample, (2) germline databases

Responsibilities			
Referee	AA	AB	BB
1	$\gamma(Z_1 = AA)$	$\gamma(Z_1 = AB)$	$\gamma(Z_1 = BB)$
2	$\gamma(Z_2 = AA)$	$\gamma(Z_2 = AB)$	$\gamma(Z_2 = BB)$
3	$\gamma(Z_3 = AA)$	$\gamma(Z_3 = AB)$	$\gamma(Z_3 = BB)$
T	$\gamma(Z_T = AA)$	$\gamma(Z_T = AB)$	$\gamma(Z_T = BB)$

SNV Genotyping Callers

Variant Allele Fraction Analysis

- Single sample

Genotypes: AA , AB , BB

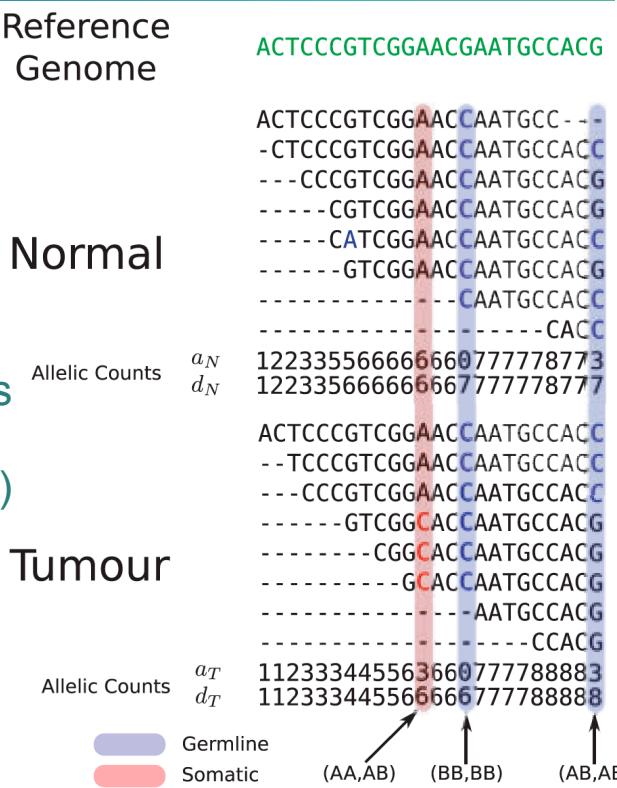


- Joint tumor-normal

Joint Genotypes:

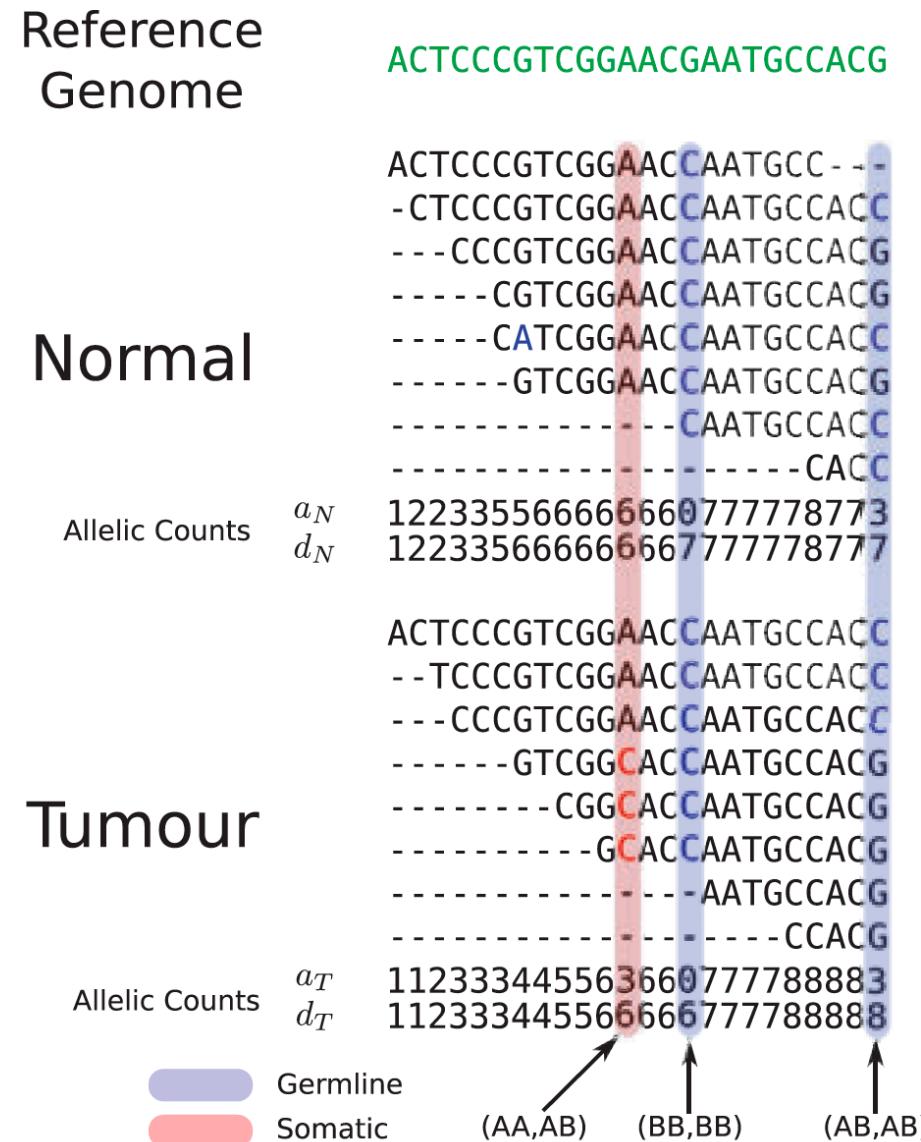
$g_N \setminus g_T$	AA	AB	BB
AA	0.01	0.95	0.00
AB	0.00	0.04	0.00
BB	0.00	0.00	0.00

- Cohort level or panel: Machine Learning (supervised)



Variant caller	Type of variant	Single-sample mode	Type of core algorithm
BAYSIC [48]	SNV	No	Machine learning (ensemble caller)
CaVEMan [34]	SNV	No	Joint genotype analysis
deepSNV [38]	SNV	No	Allele frequency analysis
EBCall [37]	SNV, indel	No	Allele frequency analysis
FaSD-somatic [31]	SNV	Yes	Joint genotype analysis
FreeBayes [44]	SNV, indel	Yes	Haplotype analysis
HapMuC [42]	SNV, indel	Yes	Haplotype analysis
JointSNVMix2 [30]	SNV	No	Joint genotype analysis
LocHap [43]	SNV, indel	No	Haplotype analysis
LoFreq [36]	SNV, indel	Yes	Allele frequency analysis
LoLoPicker [39]	SNV	No	Allele frequency analysis
MutationSeq [45]	SNV	No	Machine learning
MuSE [40]	SNV	No	Markov chain model
MuTect [35]	SNV	Yes	Allele frequency analysis
SAMtools [8]	SNV, indel	Yes	Joint genotype analysis
Platypus [41]	SNV, indel, SV	Yes	Haplotype analysis
qSNP [24]	SNV	No	Heuristic threshold
RADIA [26]	SNV	No	Heuristic threshold
Seurat [33]	SNV, indel, SV	No	Joint genotype analysis
Shimmer [25]	SNV, indel	No	Heuristic threshold
SNooPer [47]	SNV, indel	Yes	Machine learning
SNVSniffer [32]	SNV, indel	Yes	Joint genotype analysis
SOAPsnv [27]	SNV	No	Heuristic threshold
SomaticSeq [46]	SNV	No	Machine learning (ensemble caller)
SomaticSniper [28]	SNV	No	Joint genotype analysis
Strelka [17]	SNV, indel	No	Allele frequency analysis
TVC [97]	SNV, indel, SV	Yes	Ion Torrent specific
VarDict [18]	SNV, indel, SV	Yes	Heuristic threshold
VarScan2 [9]	SNV, indel	Yes	Heuristic threshold
Virmid [29]	SNV	No	Joint genotype analysis

Somatic SNV Detection using Joint Inference from Tumor-Normal Pairs



Joint Genotype Probabilities

$$P(G_{(g_N, g_T)}^i = 1)$$

$g_N \setminus g_T$	AA	AB	BB
AA	0.01	0.95	0.00
AB	0.00	0.04	0.00
BB	0.00	0.00	0.00

Homework #5: Single-nucleotide Genotype Caller

Implement a standard binomial mixture model described in Lecture 2.

- Learn the parameters and infer the genotypes
- Annotate the mutation status for a set of genomic loci.
- Expected outputs for each question will be provided so that you can check your code.
- RStudio Markdown and Python Jupyter Notebook templates provided.

Due: May 8th

Office Hours with Anna-Lisa Doebley (adoebley@uw.edu)

Zoom Meeting ID: 446 356 7725 Password: GS541

- Monday, May 4, 2-3pm
- Wednesday, May 6, 2-3pm