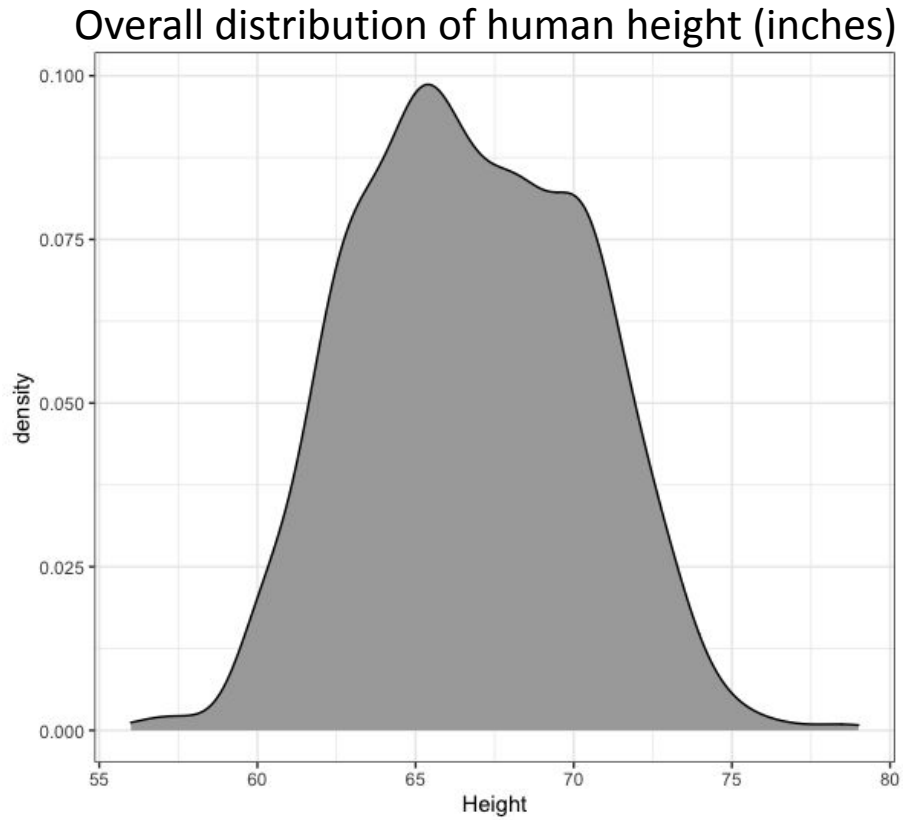
An Introduction to Mixture Models and their application in Biological Sciences

Timothy Daley

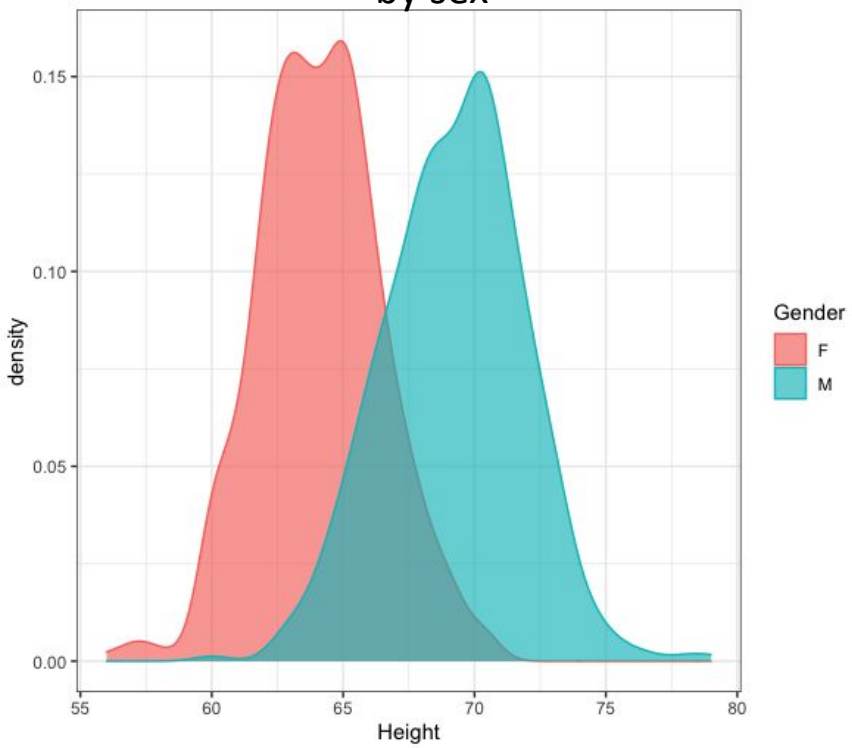
Example: Human height



Data: *Regression towards Mediocrity in Hereditary Stature,* Galton 1886

Example: Human height

Distribution of human height (inches) by sex



Data: *Regression towards Mediocrity in Hereditary Stature,* Galton 1886

Mixture model Likelihood

If we know each person sex:

$$\mathcal{L}(\boldsymbol{x}|\mu_m, \mu_f, \sigma_m, \sigma_f) = \prod_{x \in \text{male}} \mathcal{N}(x|\mu_m, \sigma_m) \cdot \prod_{x \in \text{female}} \mathcal{N}(x|\mu_f, \sigma_f)$$

But we don't, so we assign them to both groups

$$\mathcal{L}(\boldsymbol{x}|\mu_m, \mu_f, \sigma_m, \sigma_f) = \prod_{x} p\mathcal{N}(x|\mu_m, \sigma_m) + (1-p)\mathcal{N}(x|\mu_f, \sigma_f)$$

Posterior group probabilities

Given observation *x* and estimated parameters $\boldsymbol{\theta}$, what is the probability it came from a particular group?

$$\Pr(\text{male}|x, \mu_m, \mu_f, \sigma_m, \sigma_f) = \frac{\text{likelihood of male}}{\text{total likelihood}}$$

$$= \frac{p\mathcal{N}(x|\mu_m, \sigma_m)}{p\mathcal{N}(x|\mu_m, \sigma_m) + (1-p)\mathcal{N}(x|\mu_f, \sigma_f)}$$

EM algorithm

EM stands for Expectation-Maximization

Basic idea:

- Compute expected group probabilities given current guess of parameters (Expectation step)
- Update model parameters given current group probabilities (Maximization step)
- Iterate until convergence
- The EM algorithm is guaranteed to converge to a local optimum (or saddle point 😬)

EM algorithm: example

Initial guess:

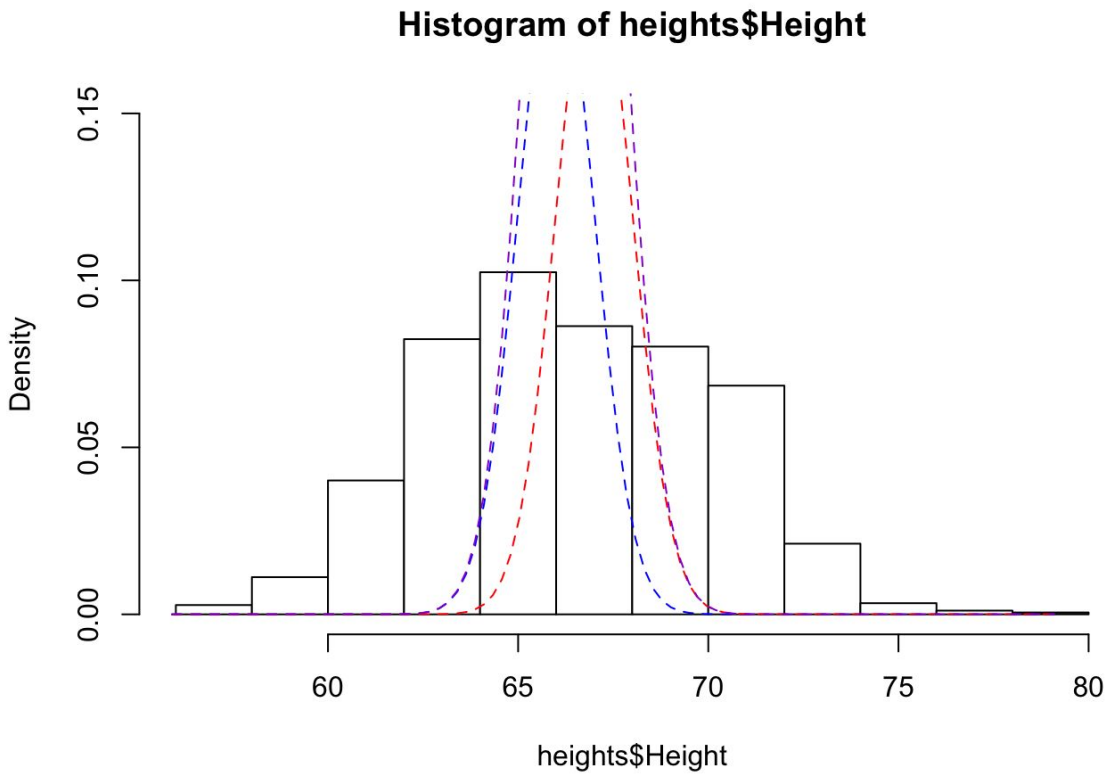
$\mu_1 = 66$

$\mu_2 = 67$

$\sigma_1 = 1$

$\sigma_2 = 1$

$p = 0.5$



EM algorithm: example

2nd round:

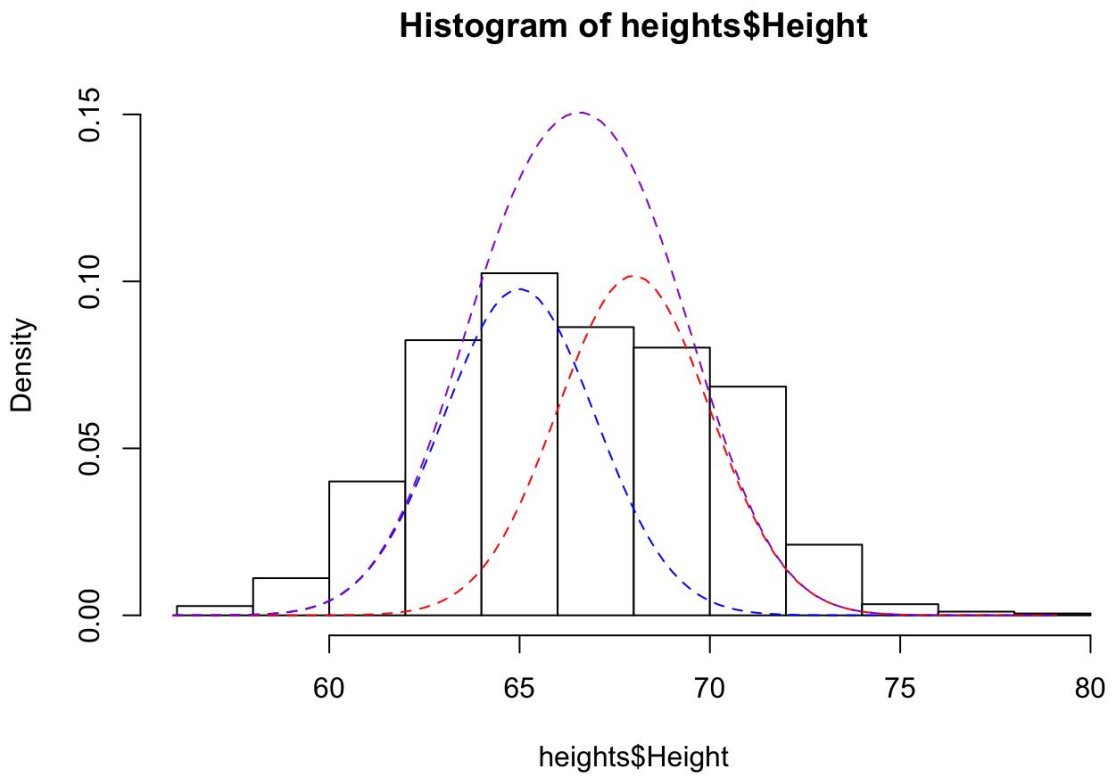
$\mu_1 = 65$

$\mu_2 = 68$

$\sigma_1 = 2$

$\sigma_2 = 2$

$p = 0.49$



EM algorithm: example

Final estimates:

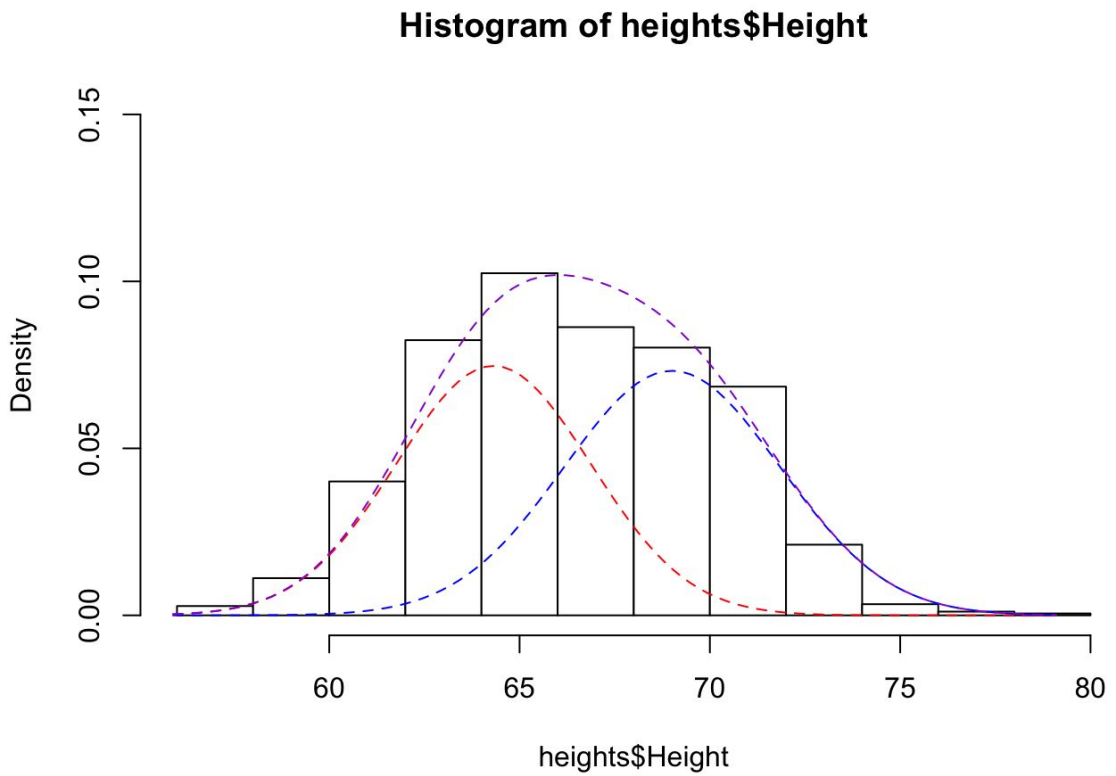
$\mu_1 = 64.32$

$\mu_2 = 69.01$

$\sigma_1 = 2.56$

$\sigma_2 = 2.84$

$p = 0.479$



EM algorithm: what if you change the initialization?

Initial guess:

$\mu_1 = 67$

$\mu_2 = 66$

$\sigma_1 = 1$

$\sigma_2 = 1$

$p = 0.5$

Final estimates:

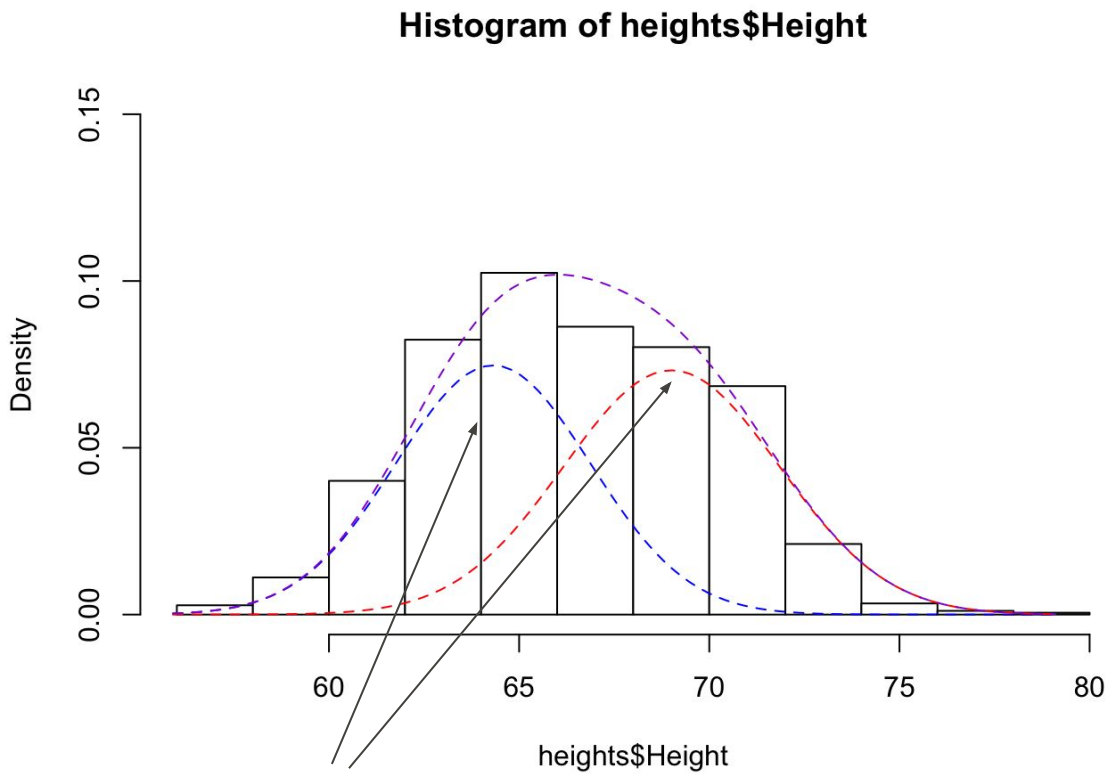
$\mu_1 = 69.01$

$\mu_2 = 64.32$

$\sigma_1 = 2.84$

$\sigma_2 = 2.56$

$p = 0.52$



notice the order switched

Statistical Identifiability

Statistical identifiability means 1 set of observations gives 1 model

Not identifiable:
(effectively same model)

Model 1: $\prod_x 0.48 \cdot \mathcal{N}(x|\mu = 64.32, \sigma = 2.56) + 0.52 \cdot \mathcal{N}(x|\mu = 69.01, \sigma = 2.84)$

Model 2: $\prod_x 0.52 \cdot \mathcal{N}(x|\mu = 69.01, \sigma = 2.84) + 0.48 \cdot \mathcal{N}(x|\mu = 64.32, \sigma = 2.56)$

Fixes:

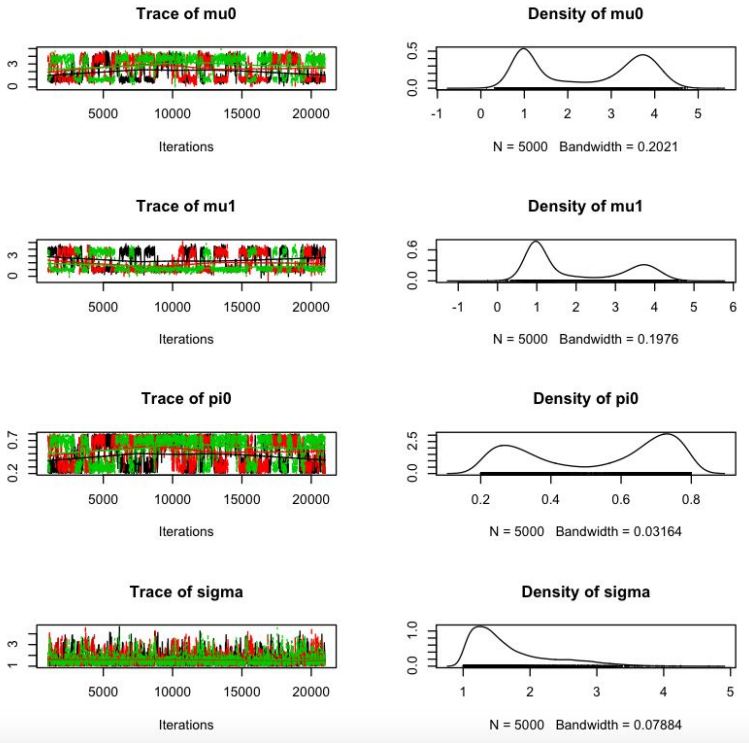
- Restrict set of models, e.g. $\mu_1 < \mu_2$
- Remove problematic part, e.g. Daley et al 2018

Statistical Identifiability and label switching

Common problem in MCMC fitting of mixture models

Lack of statistical identifiability leads to label switching

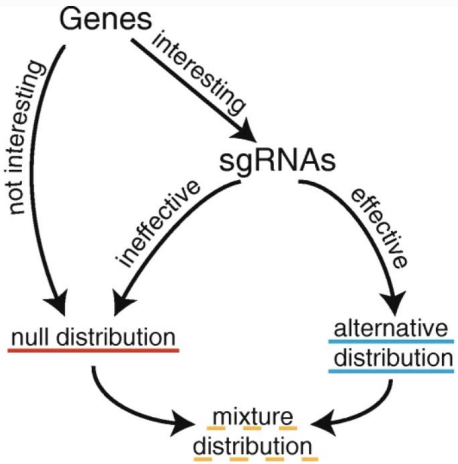
Since 2 models are equally valid, the MCMC sampler will switch between the 2



https://stats.stackexchange.com/questions/178321/my-mcmc-do-not-overlap-mixturemodel-with-jags-and-r

Marginalization to ensure identifiability

Model CRISPR screen results as a hierarchical mixture
Mixture of null and non-null genes
For non-null genes, guides are a mixture of working and not



Full likelihood: $$\mathcal{L}(f_0, f_1, p, \pi | x_{i,j}, Z_g, Y_i; i = 1, \ldots, N, j = 1, \ldots, J, g = 1, \ldots, G) = \prod_{g=1}^{G} \left(\prod_{i:g_i=g} f_0(x_i)\right)^{1-Z_g} \left(\prod_{i:g_i=g} f_0(x_i)^{1-Y_i} f_1(x_i)^{Y_i}\right)^{Z_g}.$$

Can't separate Z_{gi} (gene-level) and Y_i (guide level) $$\prod_{i=1}^{N} f_0^{(1-Y_i Z_{g_i})} f_1^{Y_i Z_{g_i}}.$$

Solution: choose one level for inference (gene),
marginalize (average over) the other level (guide)

How to choose number of components *k*?

Minimize model information criteria (info lost by the choice of model)

- AIC $2K - 2\log$ likelihood, where K is the # of parameters
- BIC $2K \log n - 2\log$ likelihood, where K is the # of parameters

Non-parametric: don't choose, use as many as possible (infinite mixture)

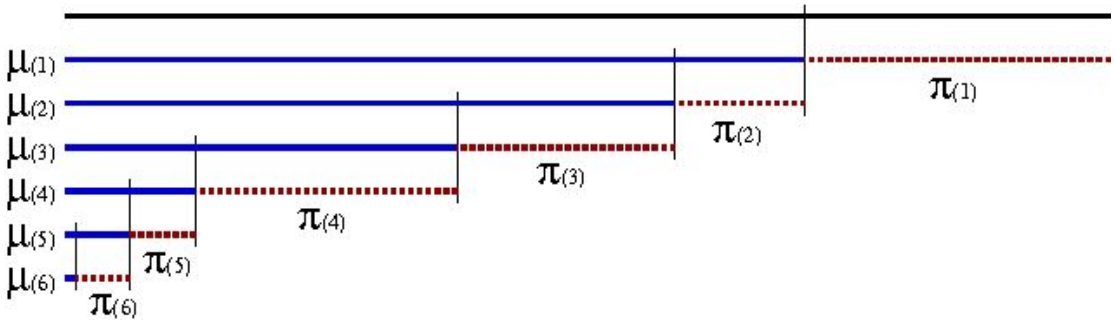
Set a prior on mixture components and their proportions

e.g. stick breaking process

sample π_1 from (0, 1)

sample π_2 from (0, 1 - π_1)

….



Examples of mixture models

- Biological samples are a mixture of cell types
  - Biopsies are a mixture of tumor and normal tissue
  - Directed differentiated cell lines are a mixture of differentiated and undifferentiated cells
- Topic Models
  - Under a bag of words model, documents can be considered a mixture of topics and topics are a mixture of words
- *K*-means clustering is a special case of Gaussian mixture model
- FDR control
  - *q* values, a Bayesian interpretation of FDRs
  - Efron's 2-groups model
    - High-throughput tests (e.g. GWAS, CRISPR screens, RNA-seq differential expression) result in a mixture of nulls and true hits

Cell type mixtures, aka deconvolution

Have:

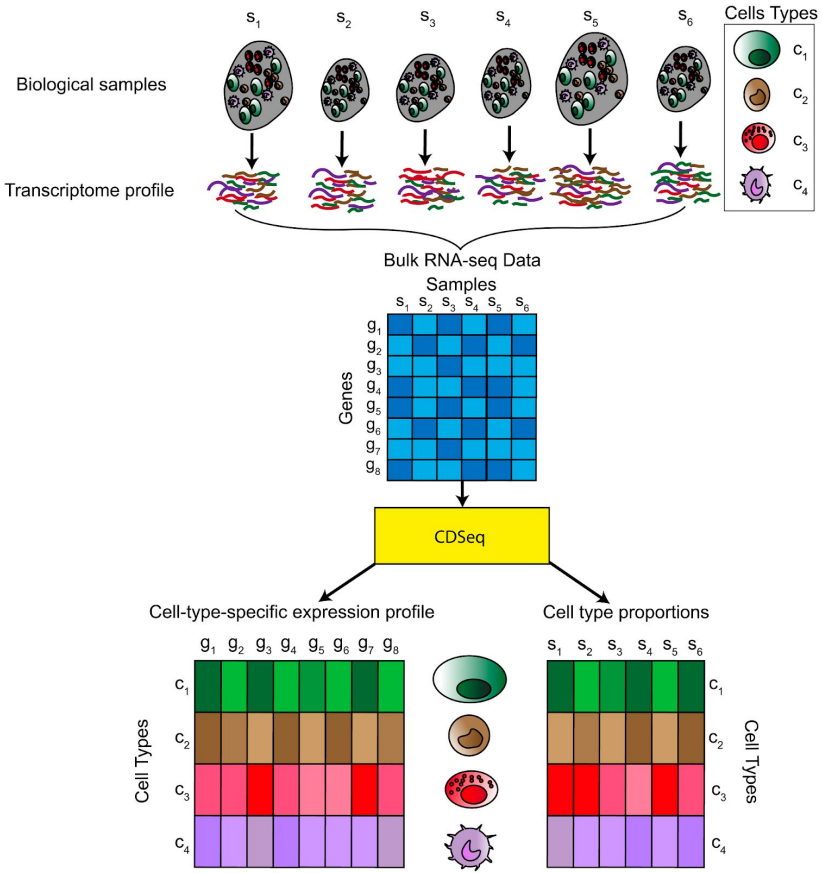
- Data (RNA-seq, ATAC-seq, etc) from mixed tissue
- Reference set of cell-type-specific profiles

Goal:

- Obtain the estimated deconvolution of the sample into estimated cell types

Problem:

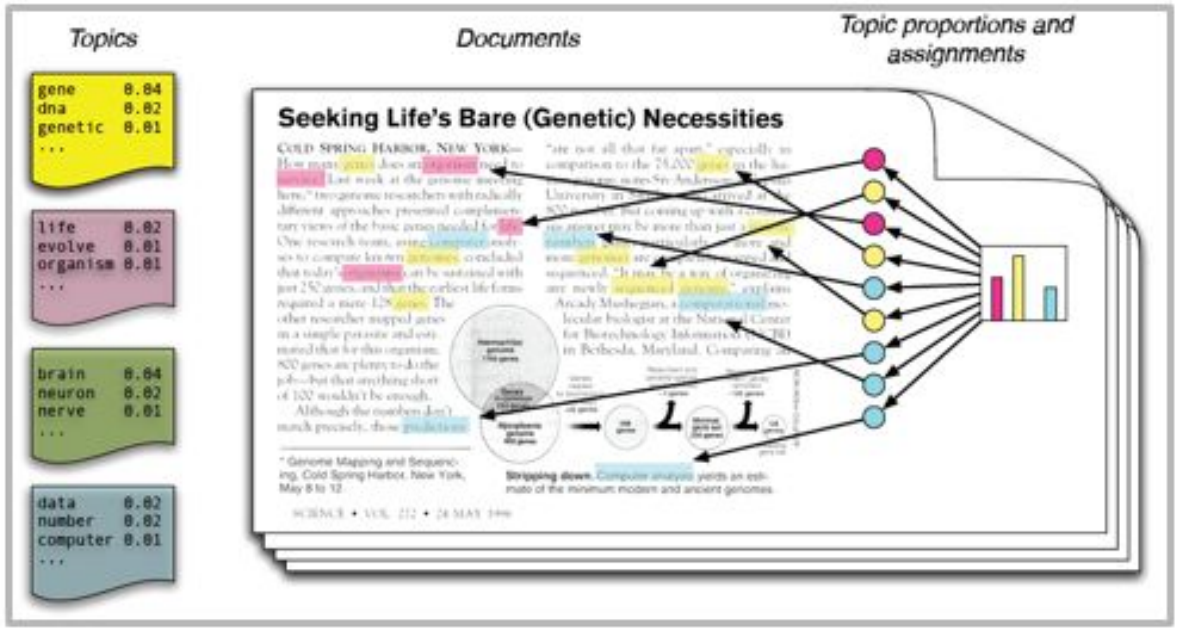
- How to ensure identifiability?



Topic Models

Hierarchical mixture model:

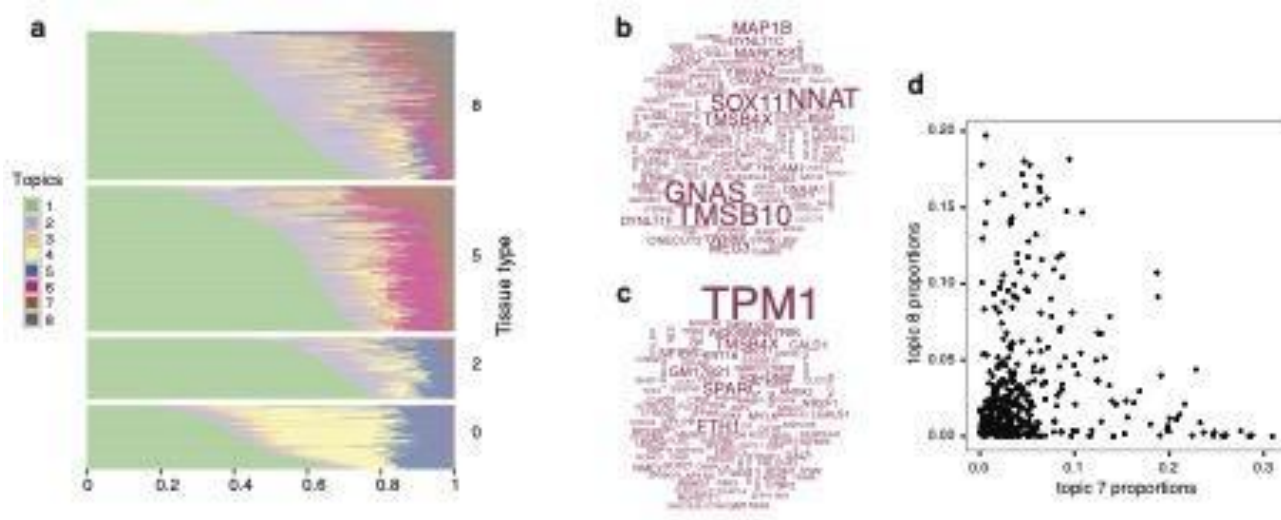
Documents are a mixture of topics
Topics are a mixture of words
We only see the final document i words, missing topics
Topic models estimate the mixtures simultaneously



Topic Models for Genomics

Cells are a mixture of gene regulatory programs

Each gene regulatory program is a mixture of gene expression (or open chromatin, or whatever)

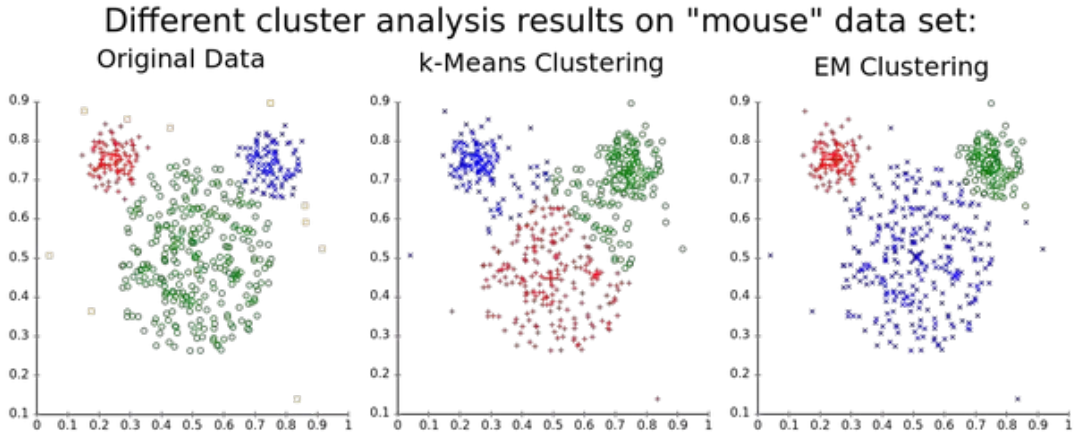


Decomposed single cell RNA-seq data of ASCL1-driven differentiation of ESCs by topic models. The cells appear to be going to either myocyte or neuron and the corresponding regulatory networks are captured in topics 7 & 8. The primary topics capture housekeeping and similar genes.

K-means clustering is a special case of a Gaussian mixture model

K-means clustering: find μ_j to minimize $\sum_x \sum_{j=1}^{k} 1(x \in \text{ group } j)(x - \mu_j)^2$

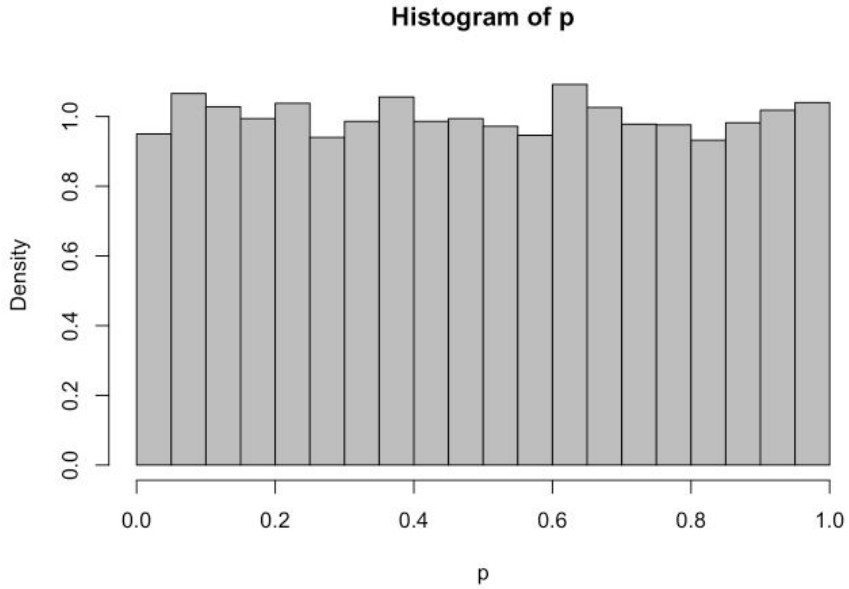
Gaussian mixture: find μ_j, σ_j to minimize $\sum_x \sum_{j=1}^{k} 1(x \in \text{group } j) \frac{(x - \mu_j)^2}{2\sigma_j^2}$



q values

Q: What is the distribution of *p* values under a correctly specified null hypothesis?

```
z = rnorm(10000, mean = 0, sd = 1)
p = pnorm(z, mean = 0, sd = 1, lower.tail = F)
hist(p, breaks = 30, col = "grey", probability = T)
```



q values

Null hypothesis follow a uniform distribution,
Non-null should be a peak near 0

Model as a mixture of uniform and a non-uniform

$$p \sim (1-\lambda)\mathcal{U}(0,1) + \lambda f(p)$$

Estimate FDR from mixture

$$\begin{aligned}\text{FDR}(t) &= \frac{\#\text{ False Positives}}{\#\text{ tests called significant}} \\ &= \frac{\#\text{truly null } p_i \text{ w/ } p_i \leq t}{\#\, p_i \leq t} \\ &\approx \frac{(1-\lambda)Nt}{\#\, p_i \leq t}\end{aligned}$$

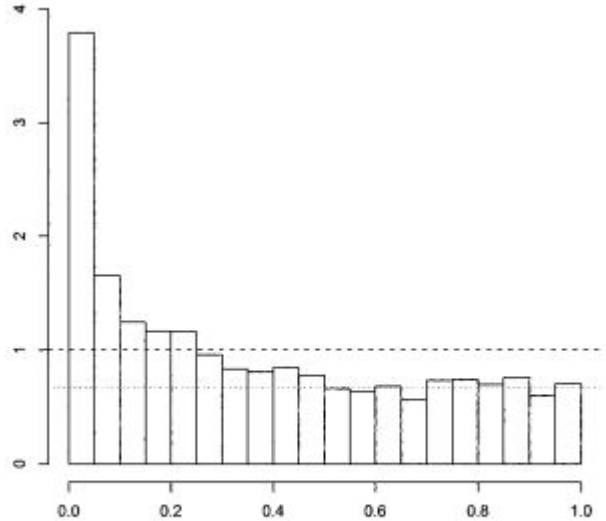


Fig. 1. A density histogram of the 3,170 *p* values from the Hedenfalk *et al.* (14) data. The dashed line is the density histogram we would expect if all genes were null (not differentially expressed). The dotted line is at the height of our estimate of the proportion of null *p* values.

Storey & Tibshirani, 2003, PNAS

Efron's two-groups model

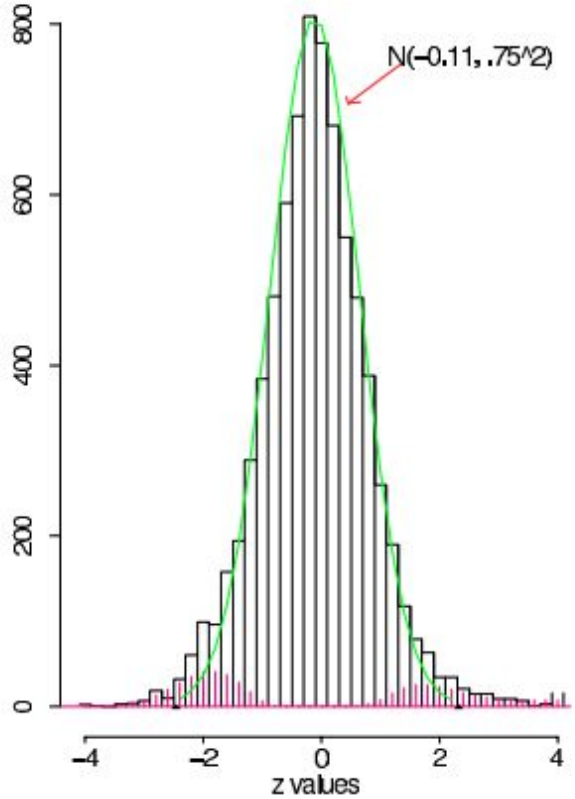
Applicable in high-throughput hypothesis tests

Convert *p* values to *z* values $z = \Phi^{-1}(p)$

Assumption: most genes are null
Consequence: measurements near the peak of the distribution are null, outliers are non-null

What is distribution of *z* under null?

Efron: *z* should be *N*(0,1), but usually isn't.
Idea: fit of main peak should be nulls



histogram of *z*-values from a microarray experiment comparing HIV+ patients to HIV-

The local fdr

FDR at threshold *t* is equal to the fraction of False Positives among test called significant

One interpretation: if we set threshold *t,* then

FDR(*t*) = Pr(a randomly chosen test w/ $p \leq t$ is null)

local fdr(*t*) = Pr(test w/ $p = t$ is null)

Fit a mixture model to z

$$z \sim (1-\lambda) f_0(z) + \lambda f_1(z).$$

Then we can compute $\text{fdr}(z)$ as

$$\text{fdr}(z) = \frac{(1-\lambda) f_0(z)}{(1-\lambda) f_0(z) + \lambda f_1(z)}$$

Efron's two-groups model in CRISPR screens

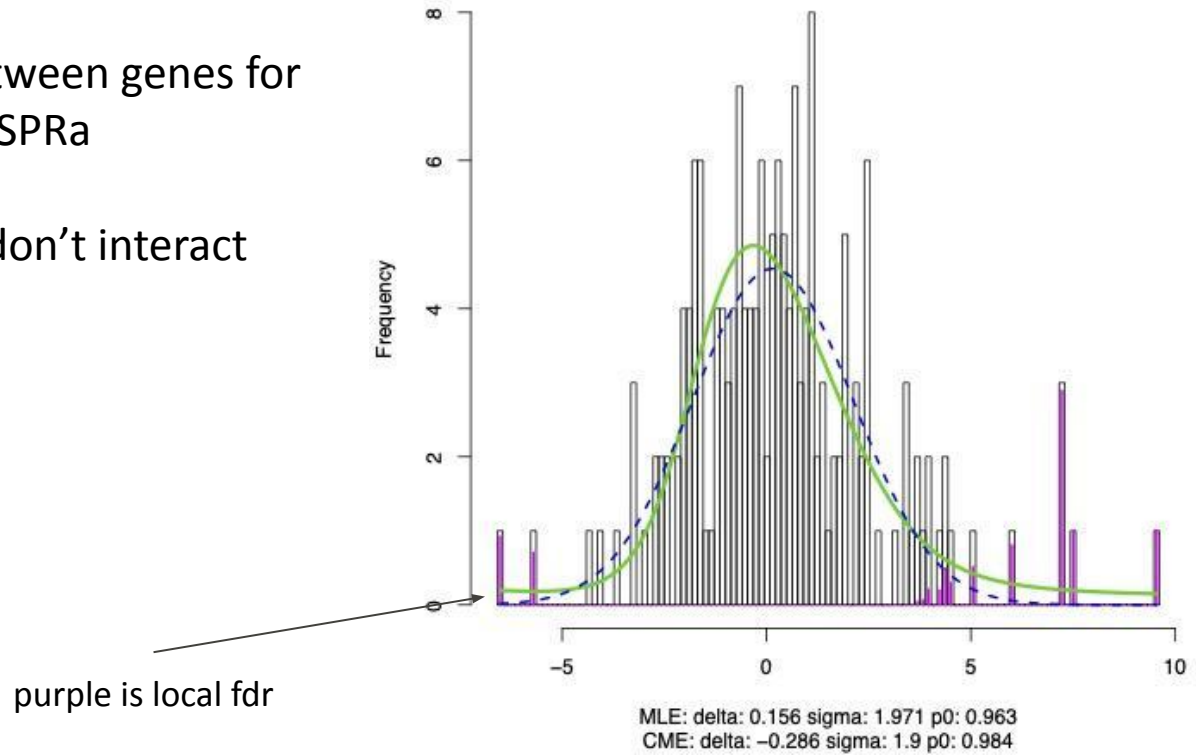
From Yanxia Liu *et al*

We tested interactions between genes for neural induction using CRISPRa

Assumption: most genes don't interact

- - - - - - - : f_0

——— : $(1 - \lambda) f_0 + \lambda f_1$



purple is local fdr