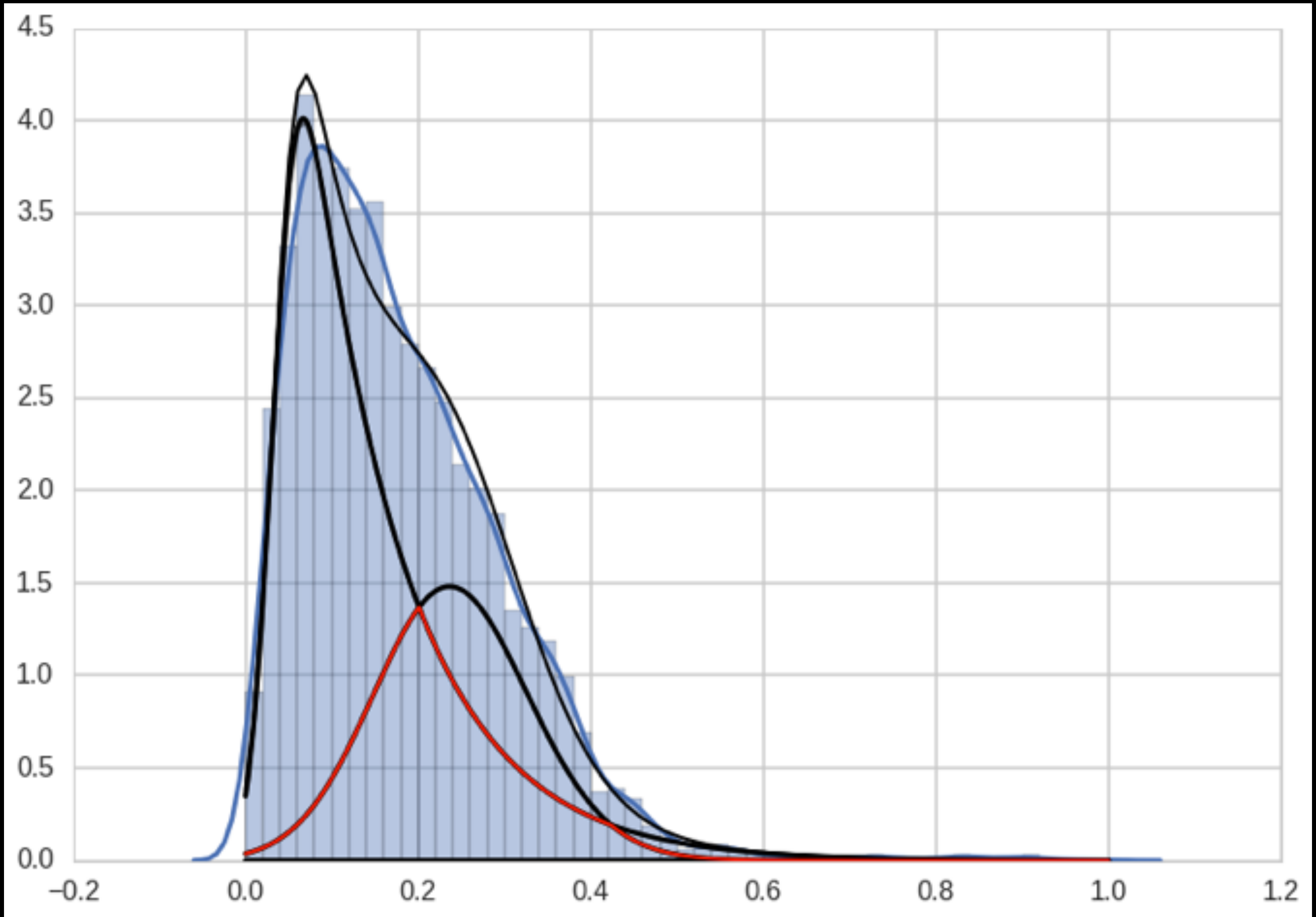
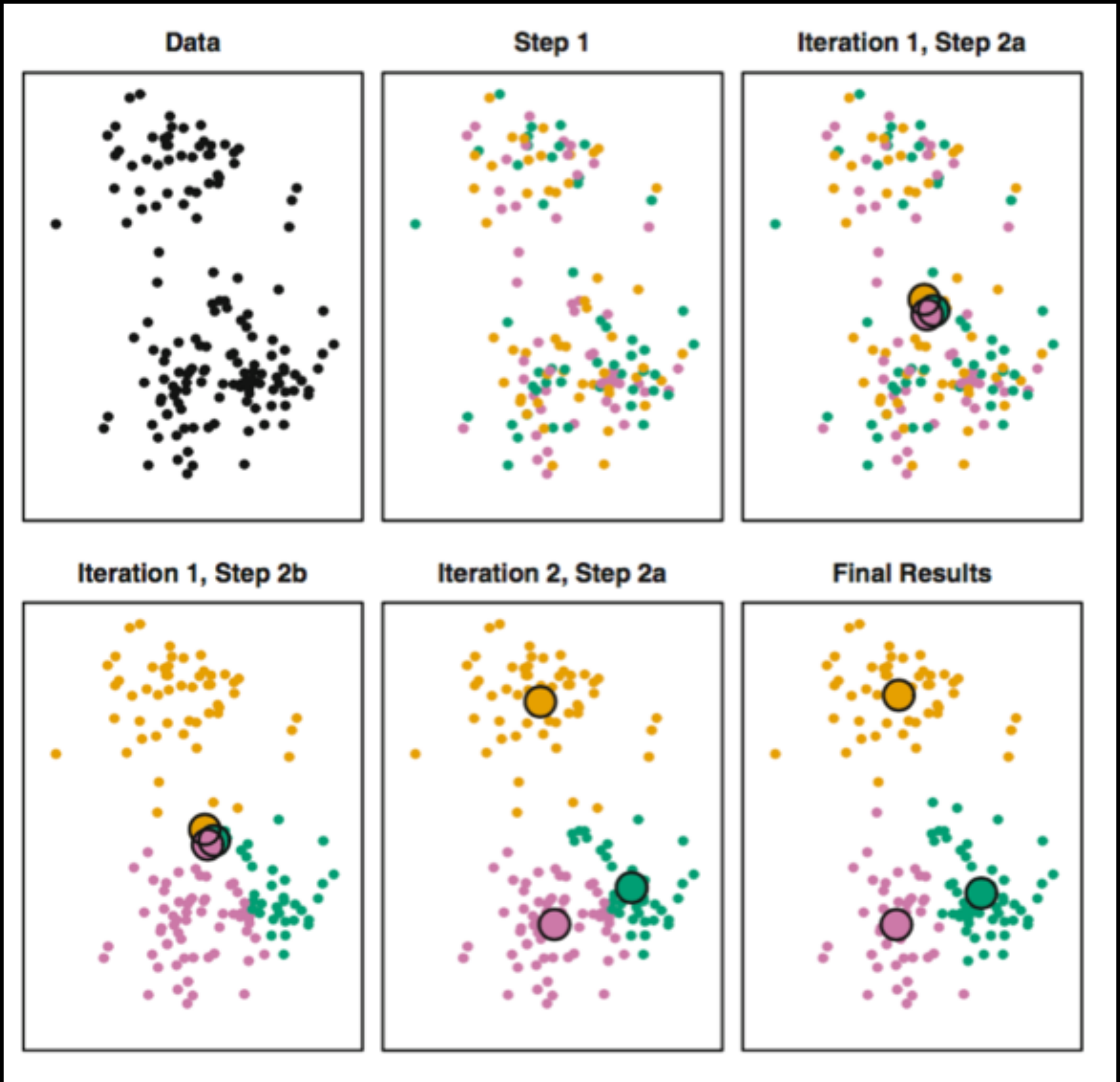
# Mixtures



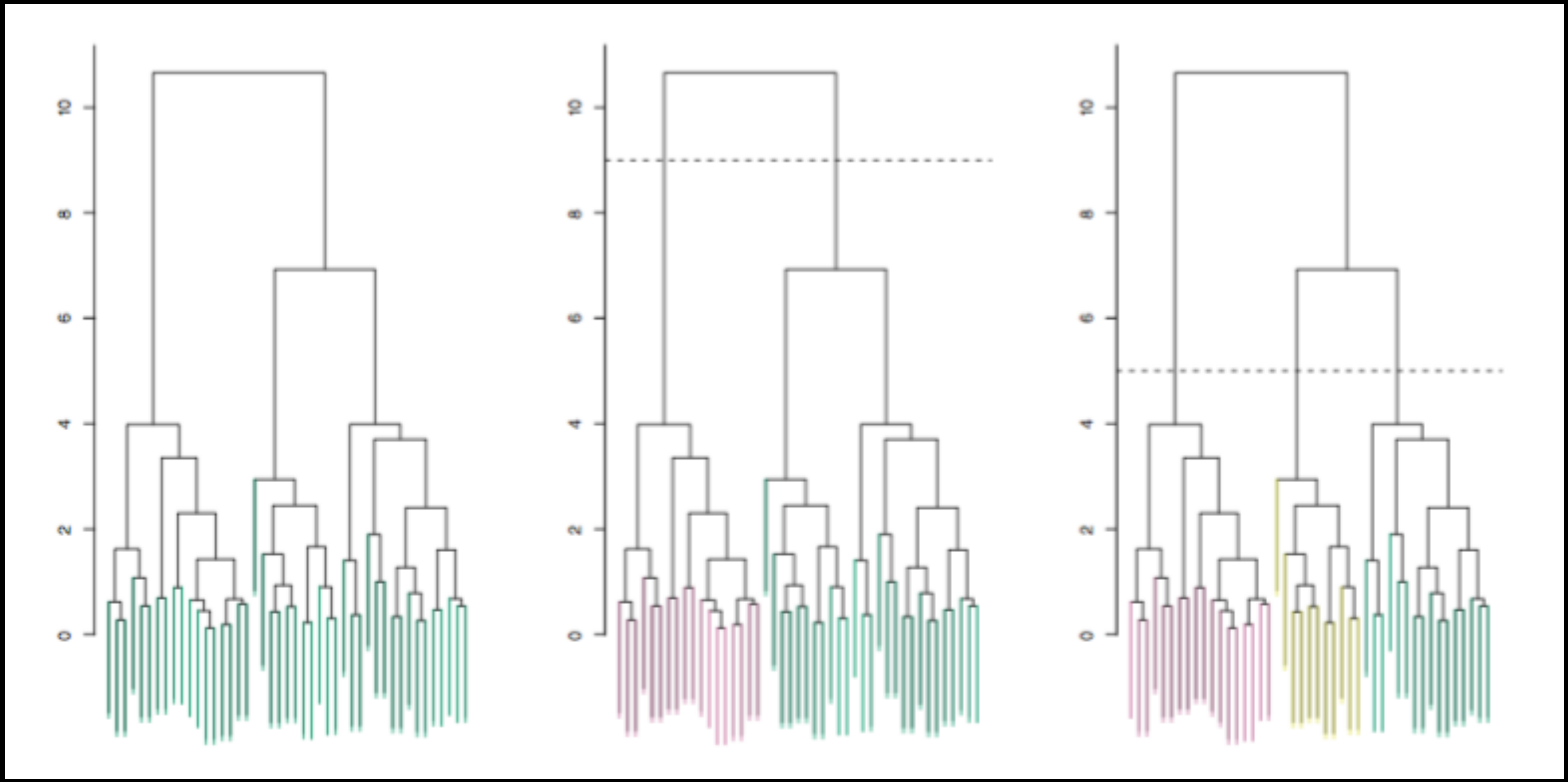

Gilles de Hollander
January 25, 2017
Machine Learning Reading Group

# Clustering



Data
Step 1
Iteration 1, Step 2a
Iteration 1, Step 2b
Iteration 2, Step 2a
Final Results

# Clustering



# Clustering

Given an initial set of $k$ means $m_1^{(1)},\ldots,m_k^{(1)}$ (see below), the algorithm proceeds by alternating between two steps:[5]

**Assignment step**: Assign each observation to the cluster whose mean yields the least within-cluster sum of squares (WCSS). Since the sum of squares is the squared Euclidean distance, this is intuitively the "nearest" mean.[6] (Mathematically, this means partitioning the observations according to the Voronoi diagram generated by the means).

$$S_i^{(t)} = \big\{x_p : \big\|x_p - m_i^{(t)}\big\|^2 \le \big\|x_p - m_j^{(t)}\big\|^2 \ \forall j, 1 \le j \le k\big\},$$

where each $x_p$ is assigned to exactly one $S^{(t)}$, even if it could be assigned to two or more of them.

**Update step**: Calculate the new means to be the centroids of the observations in the new clusters.

$$m_i^{(t+1)} = \frac{1}{|S_i^{(t)}|} \sum_{x_j \in S_i^{(t)}} x_j$$

1. *Begin with the disjoint clustering having level L(0) = 0 and sequence number m = 0.*
2. *Find the least dissimilar pair of clusters in the current clustering, say pair (r), (s), according to*

   *d[(r),(s)] = min d[(i),(j)]*

   *where the minimum is over all pairs of clusters in the current clustering.*
3. *Increment the sequence number : m = m +1. Merge clusters (r) and (s) into a single cluster to form the next clustering m. Set the level of this clustering to*

   *L(m) = d[(r),(s)]*
4. *Update the proximity matrix, D, by deleting the rows and columns corresponding to clusters (r) and (s) and adding a row and column corresponding to the newly formed cluster. The proximity between the new cluster, denoted (r,s) and old cluster (k) is defined in this way:*

   *d[(k), (r,s)] = min d[(k),(r)], d[(k),(s)]*
5. *If all objects are in one cluster, stop. Else, go to step 2.*

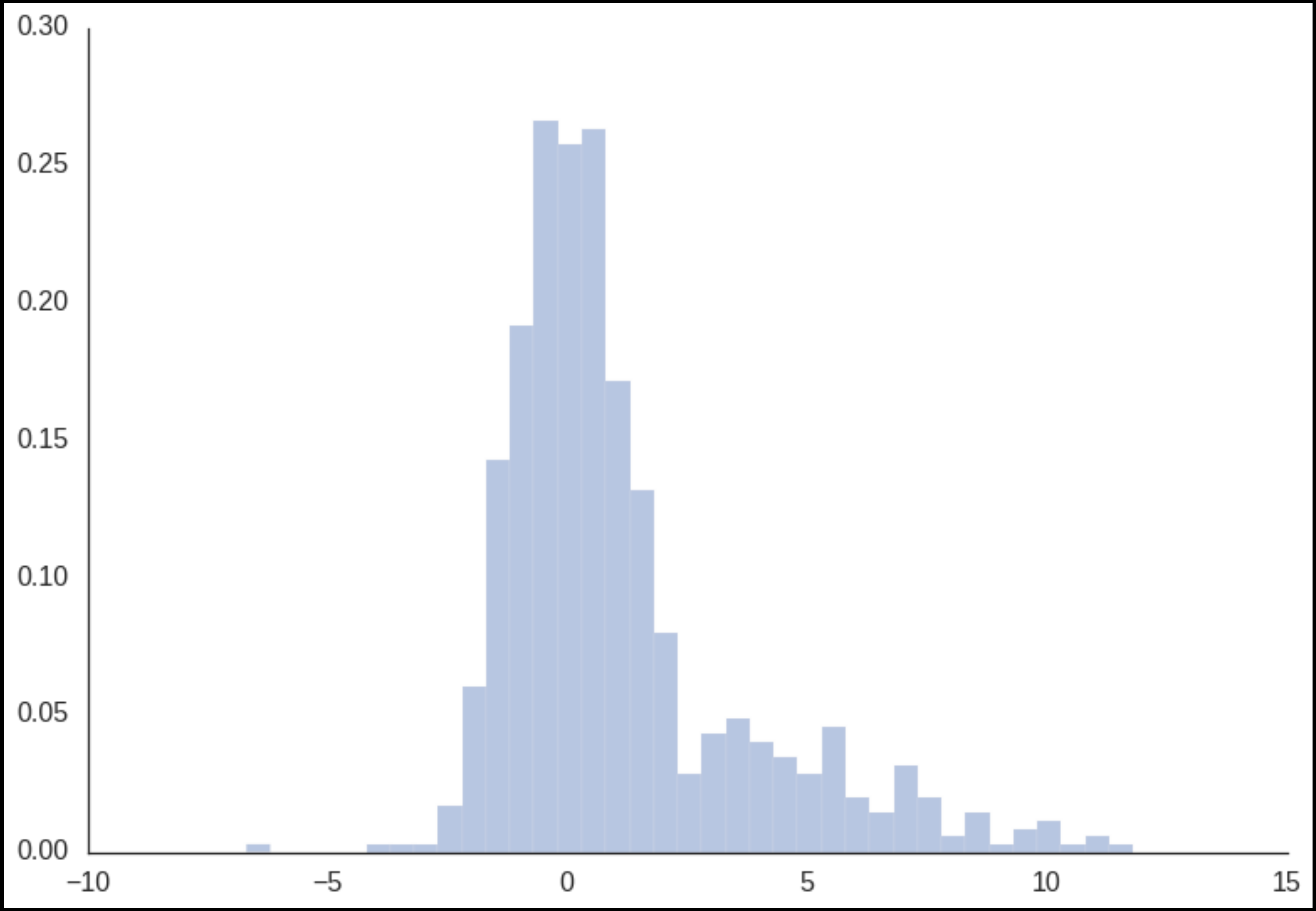
# Clustering

- Divide data over clusters minimising some quantity:
  - Distance of data points to cluster center
  - Dissimilarity of data points within cluster
- Many forms of distance (Euclidian, Manhattan, …)
- “Algorithmic” approach

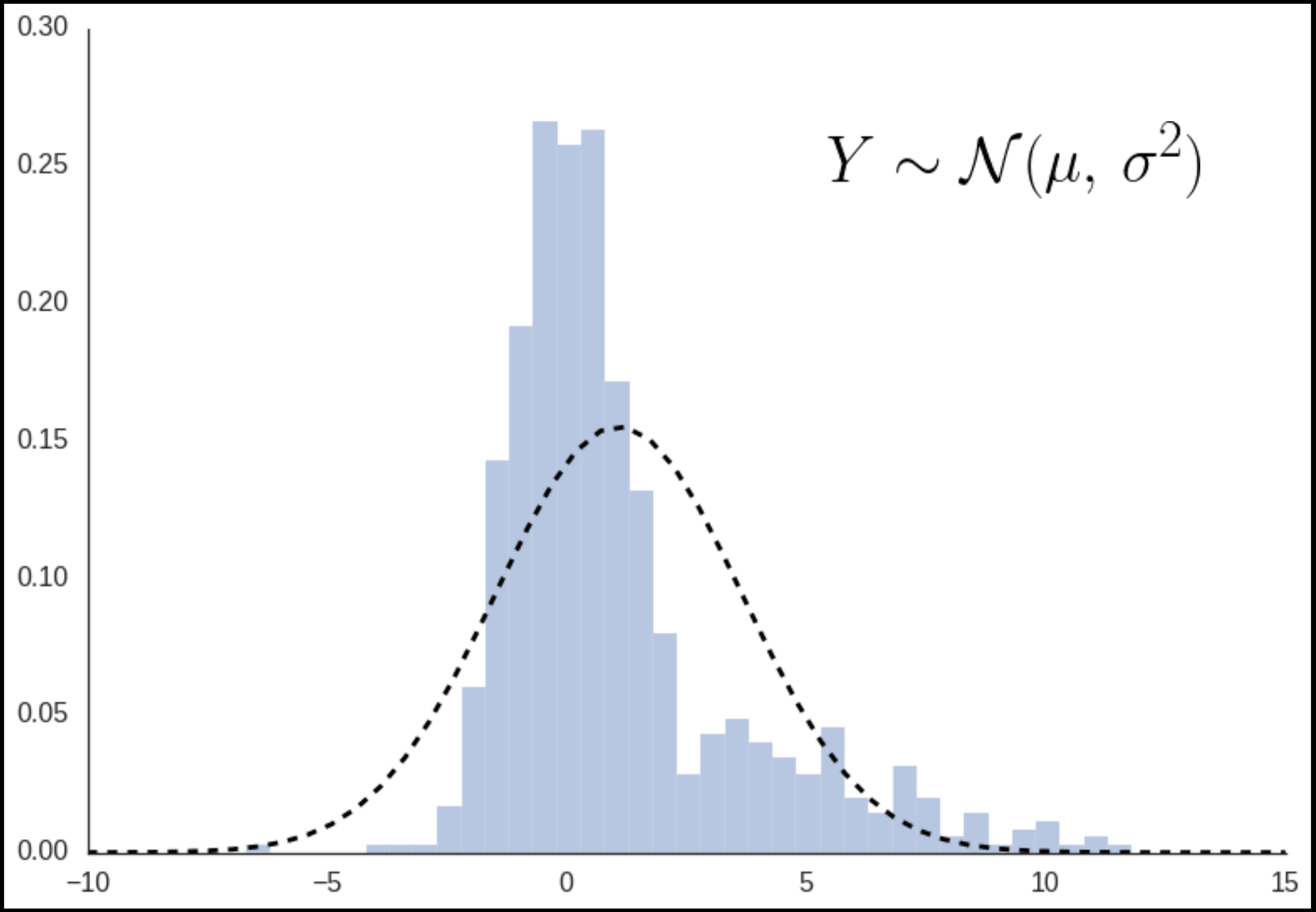
# Mixture model

- Generative Probabilistic Model
  - “Generative Story”
  - Interpretable parameters
  - Likelihood function
    - Maximum Likelihood apparatus
    - Bayesian apparatus

# Mixture model



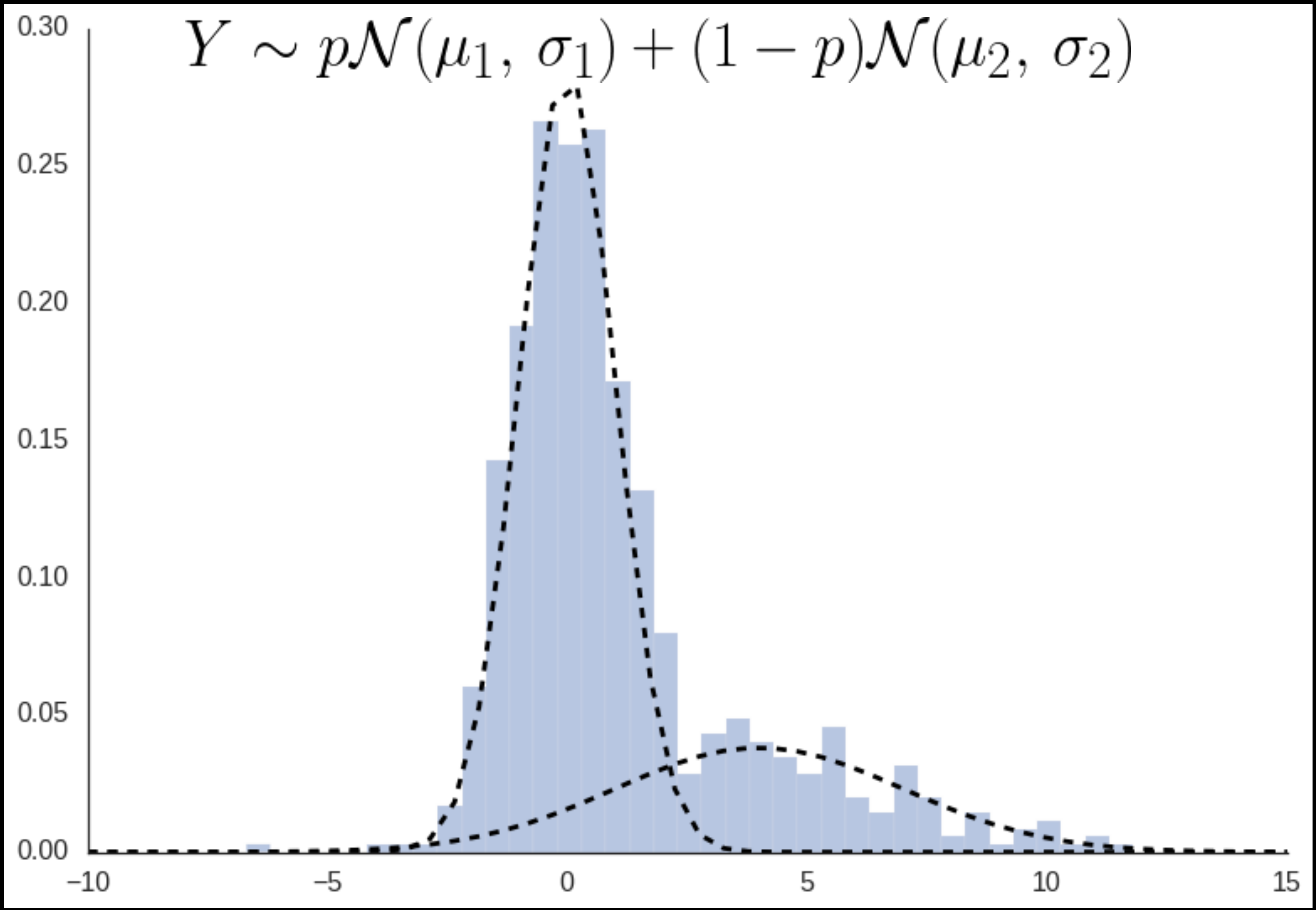
# Mixture model



# Mixture model

A *mixture distribution* is a convex combination

$$\sum_{j=1}^{k} p_j f_j(x), \qquad p_j \geq 0, \qquad \sum_{j=1}^{k} p_j = 1,$$


$Y \sim p\mathcal{N}(\mu_1,\, \sigma_1) + (1-p)\mathcal{N}(\mu_2,\, \sigma_2)$
0.30
0.25
0.20
0.15
0.10
0.05
0.00
−10
−5
0
5
10
15

# How to fit a mixture model?

- Maximum Likelihood
- Bayesian

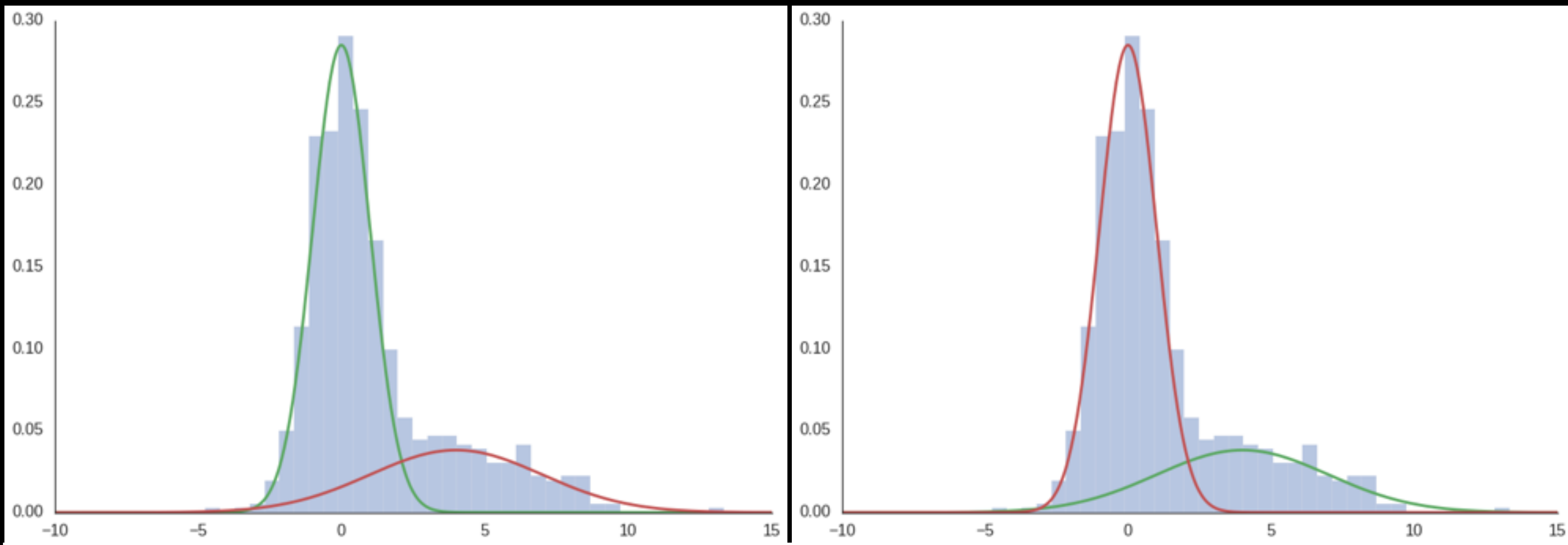
# Identifiability issues

- A parametric family of distributions is said to be identifiable if any two parameter sets define the same probability law on Y, if and only if they are identical.

# Identifiability issues

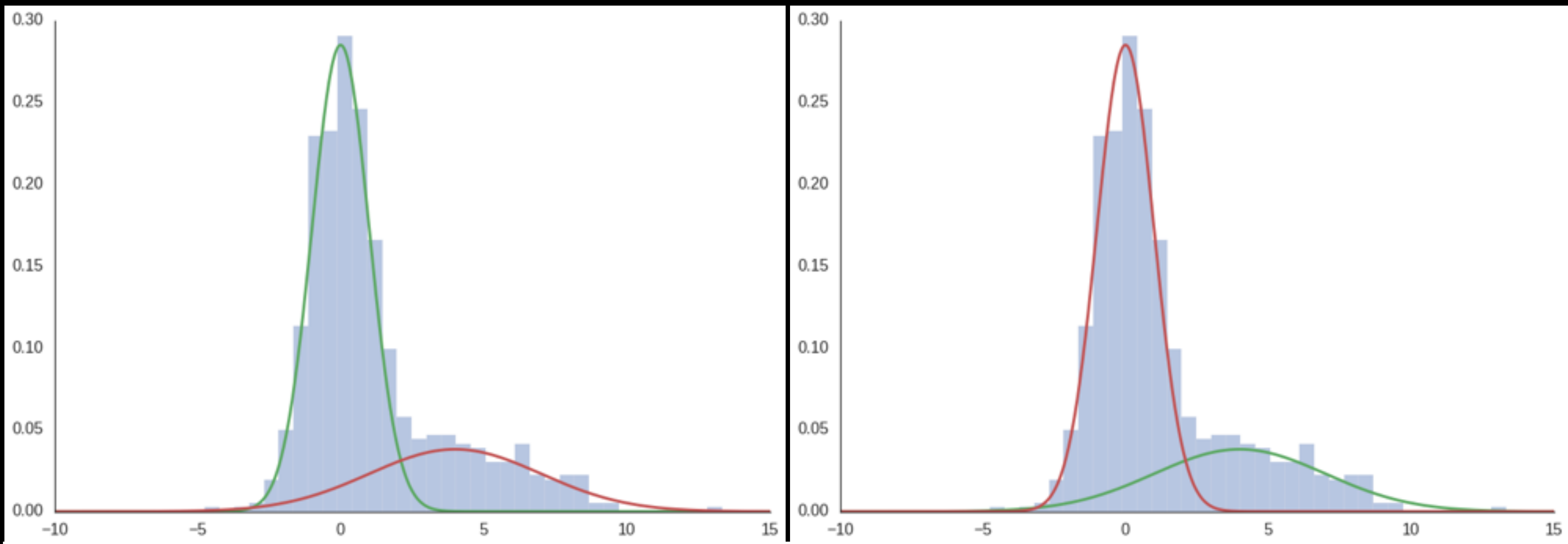
- Label switching
- Overfitting

# Label Switching




$$p\mathcal{N}(\mu_1,\,\sigma_1) + (1-p)\mathcal{N}(\mu_2,\,\sigma_2) \;=\; (1-p)\mathcal{N}(\mu_2,\,\sigma_2) + p\mathcal{N}(\mu_1,\,\sigma_1)$$

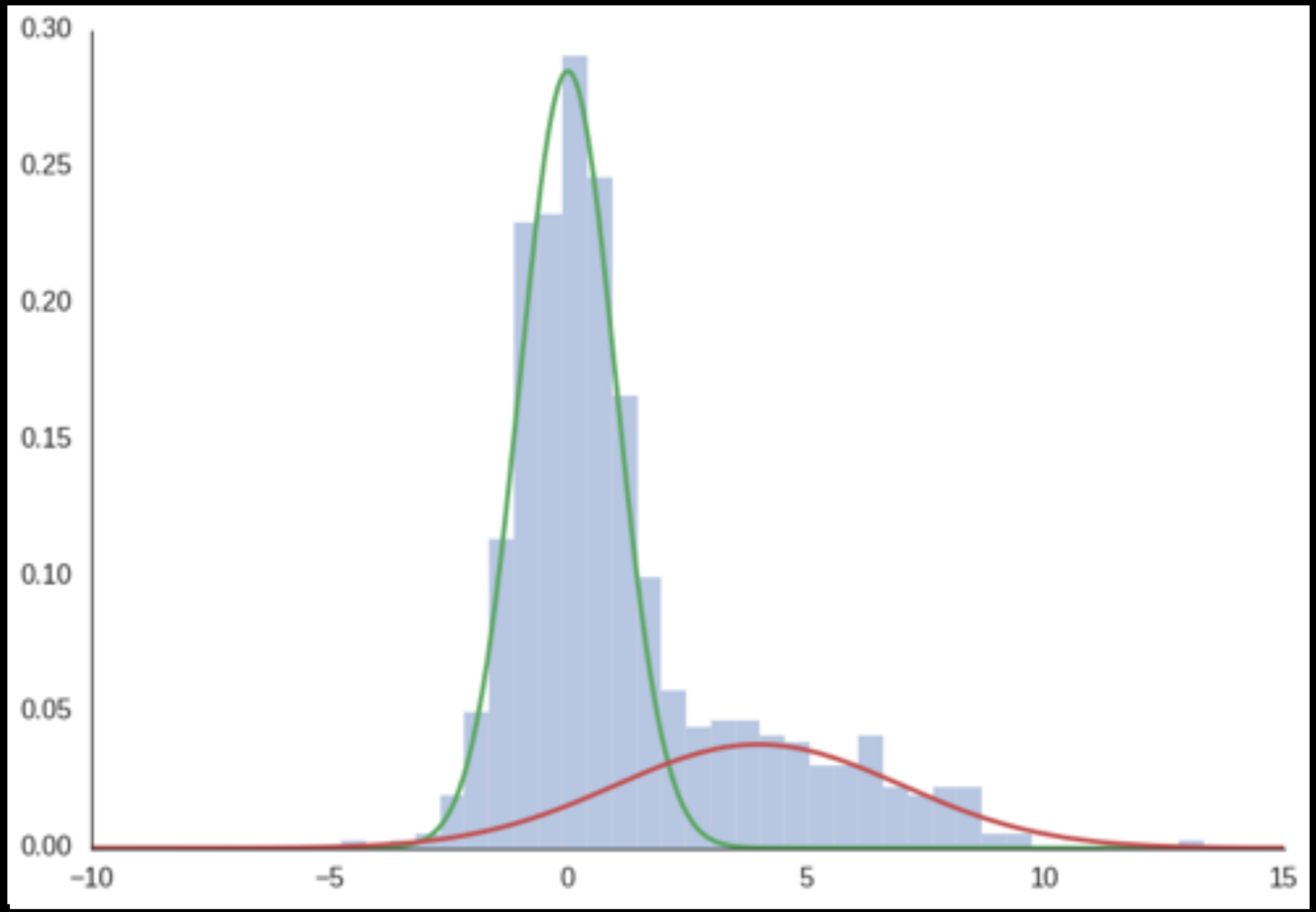
# Label Switching

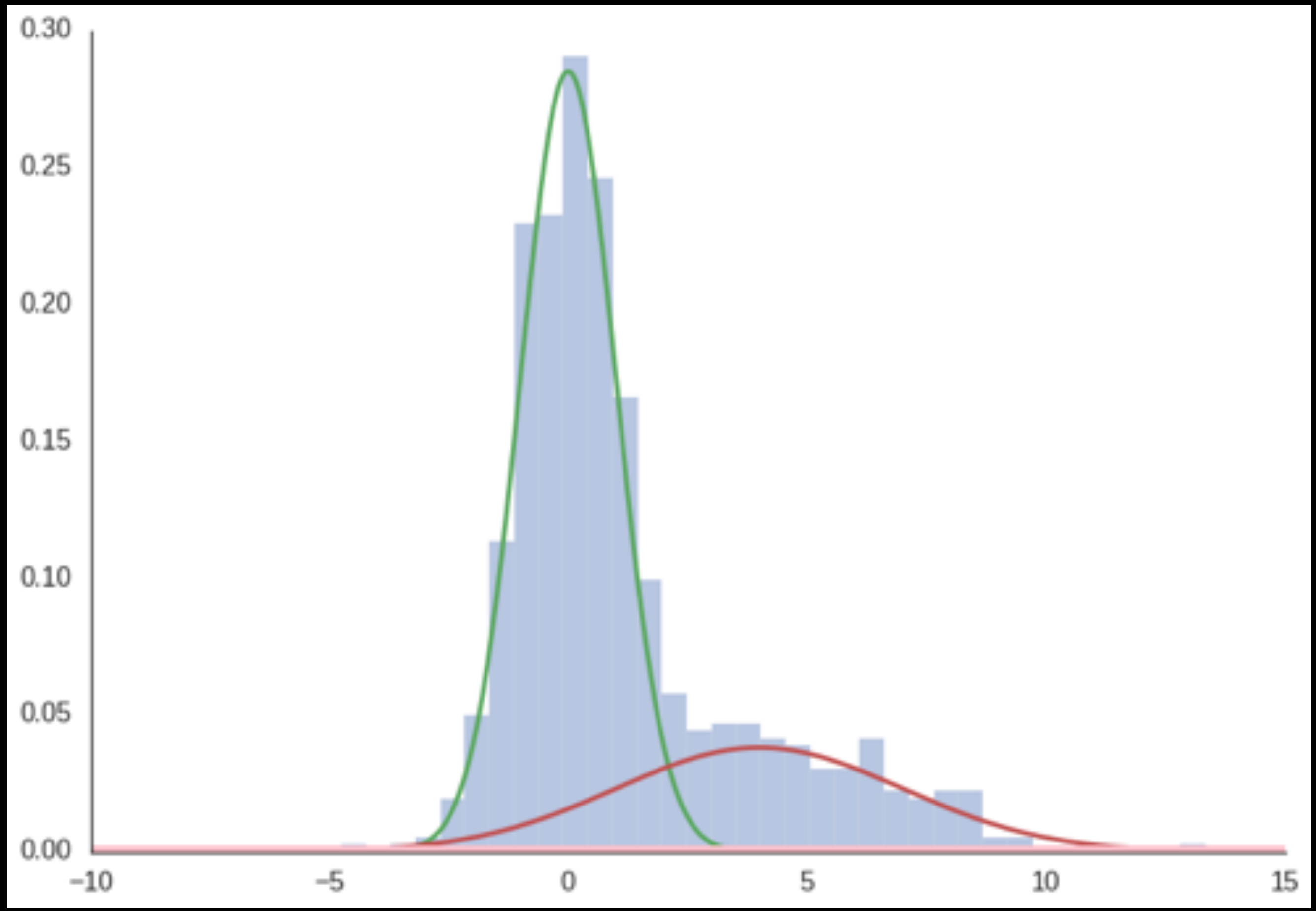



$$p\mathcal{N}(\mu_1, \sigma_1) + (1-p)\mathcal{N}(\mu_2, \sigma_2) = (1-p)\mathcal{N}(\mu_2, \sigma_2) + p\mathcal{N}(\mu_1, \sigma_1)$$

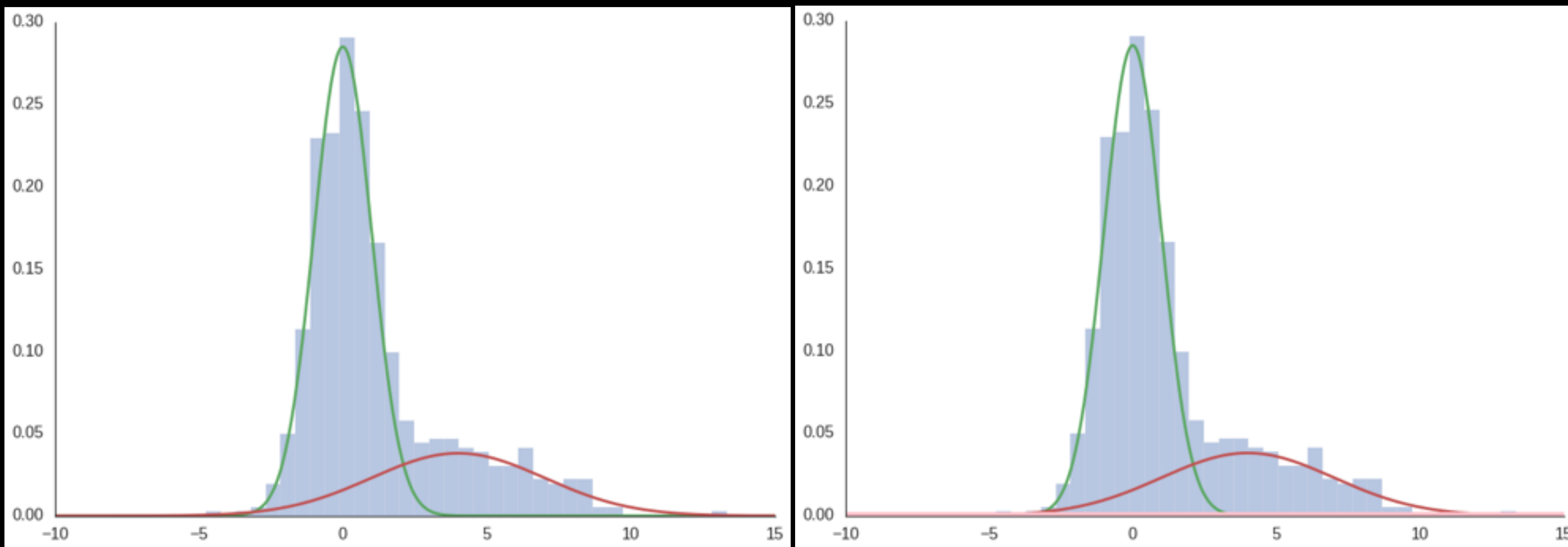
K! labeling orderings (1, 2, 6, 24, 120, 720…!)

# Overfitting



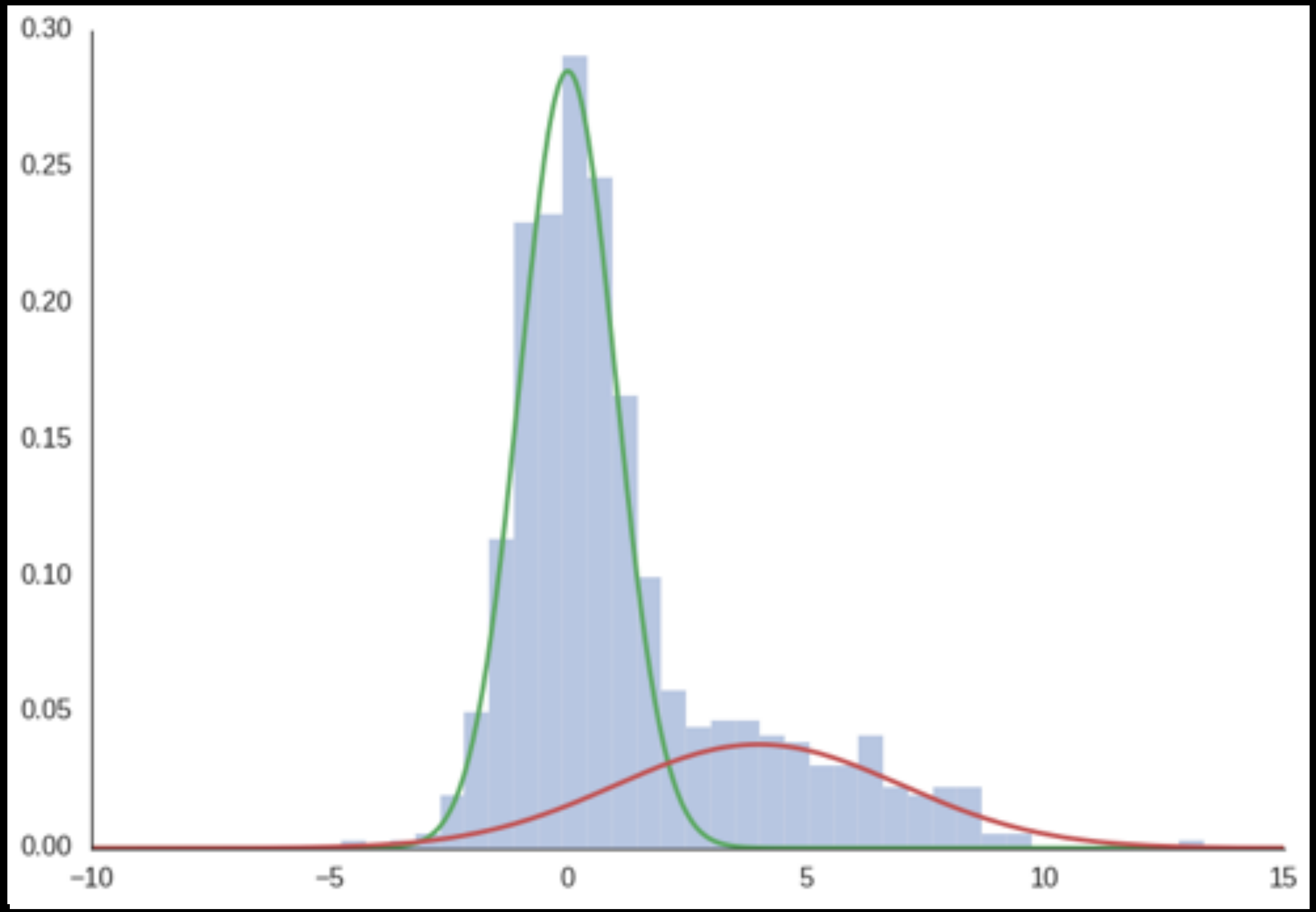



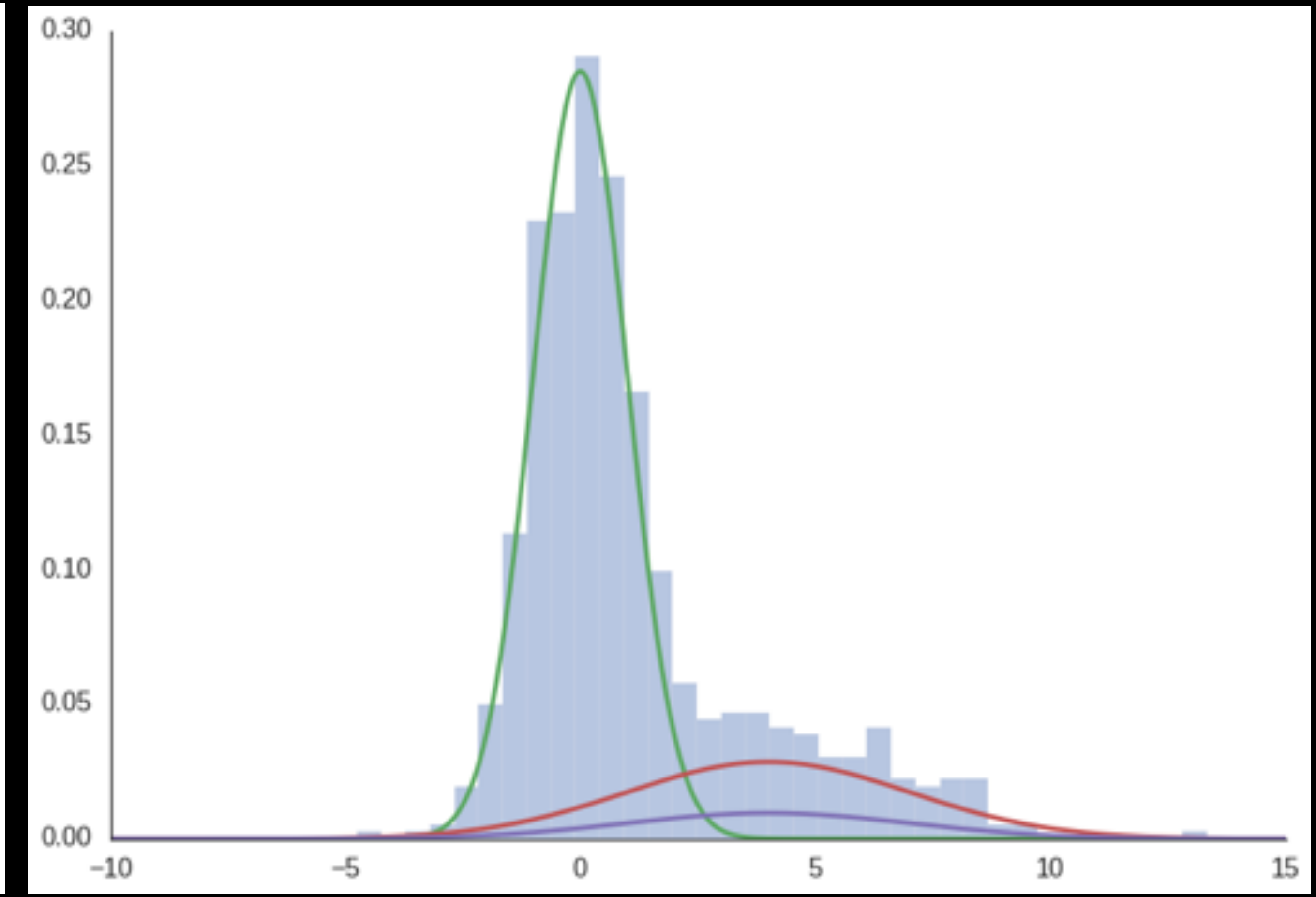
# Overfitting




(Weight of third component is 0.)

# Overfitting




Parameters of 2nd and 3rd component
are identical

# Larry Wasserman

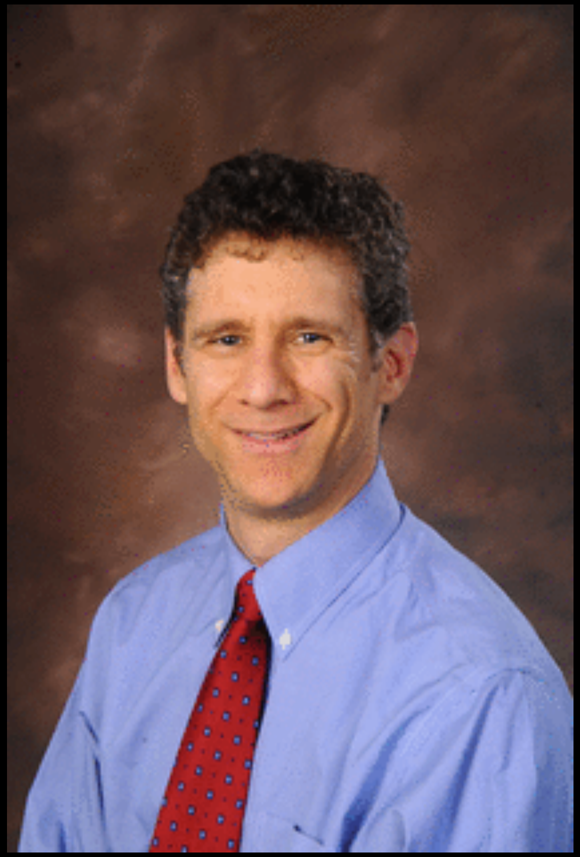
- *"I have decided that mixtures, like tequila, are inherently evil and should be avoided at all costs."*

# Formal constraints

- Parameters of every component have to be different
  - (How different)
  - (At least one? Or all?)
- Ordering of components is based on parameter values
  - e.g., The component with the smallest mean is component number 1, etc.
  - (What if two components have same mean, but different stand deviations)

# Andrew Gelman

*a mixture model can be a "beast" (as Larry puts it), but this beast can be tamed with a good prior distribution.*



# Latent variables

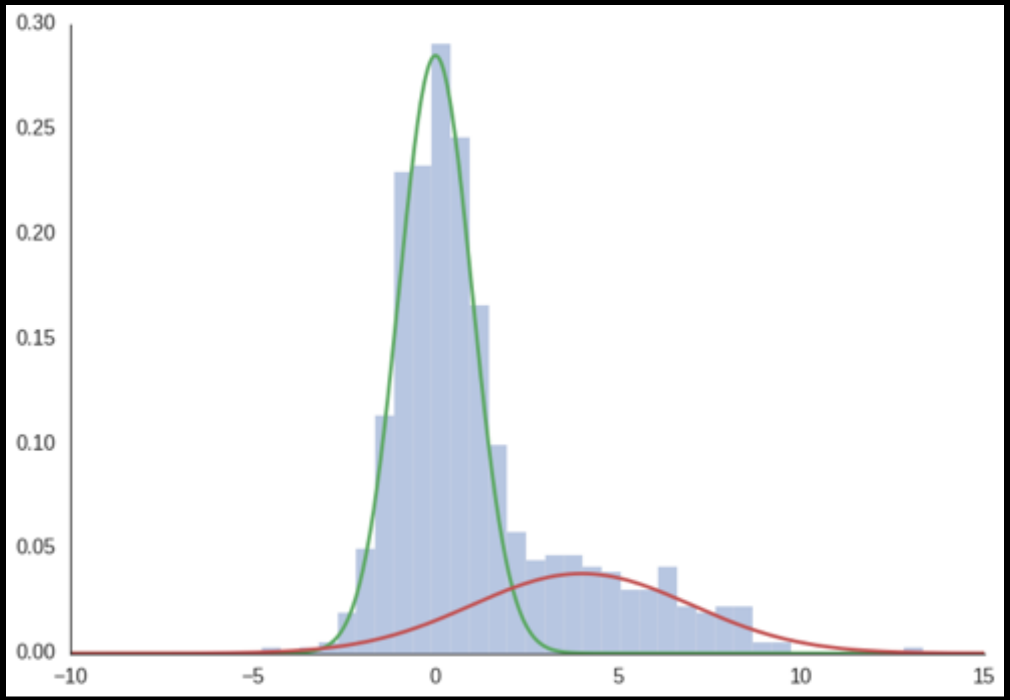
| | x |
|---|---|
| **0** | 4.247966 |
| **1** | 3.180180 |
| **2** | 7.912461 |
| **3** | 7.291698 |
| **4** | 5.474827 |
| **5** | 1.708631 |
| **6** | 5.608130 |
| **7** | 3.913172 |
| **8** | 5.513004 |
| **9** | -0.935138 |

# Latent variables

| | x | z1 | z2 |
|---|---|---|---|
| **0** | 4.247966 | False | True |
| **1** | 3.180180 | False | True |
| **2** | 7.912461 | True | False |
| **3** | 7.291698 | False | True |
| **4** | 5.474827 | True | False |
| **5** | 1.708631 | True | False |
| **6** | 5.608130 | False | True |
| **7** | 3.913172 | True | False |
| **8** | 5.513004 | True | False |
| **9** | -0.935138 | False | True |

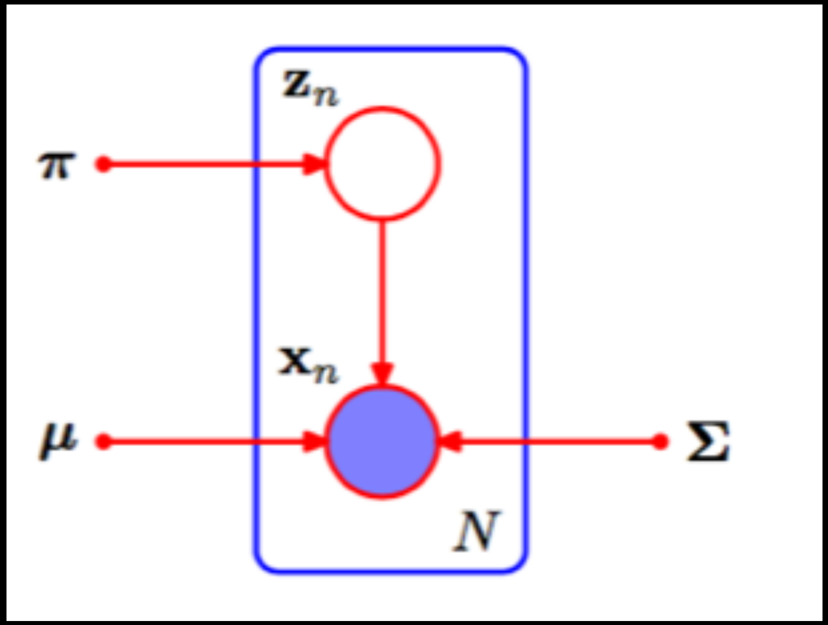
## Linear summation of Gaussians

$$p(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k).$$




## Marginalising over Possible values of z

$$p(\mathbf{x}) = \sum_{\mathbf{z}} p(\mathbf{z}) p(\mathbf{x}|\mathbf{z})$$



$$\ln p(\mathbf{X}|\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sum_{n=1}^{N} \ln \left\{ \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right\}. \quad (9.14)$$

- No closed-form solution for maximum
- *Expectation maximisation*
- (Generic optimisation)

# Expectation maximisation

- Expectation maximisation
- set parameters to some initial estimate
- Repeat until convergence:
  - E
    - Assume certain cluster parameters
    - Find out "responsibilities" of every data point  for every cluster
  - M
    - Set parameters of distributions to ML estimates, weighted by responsibilities of datapoints

# Expectation maximisation

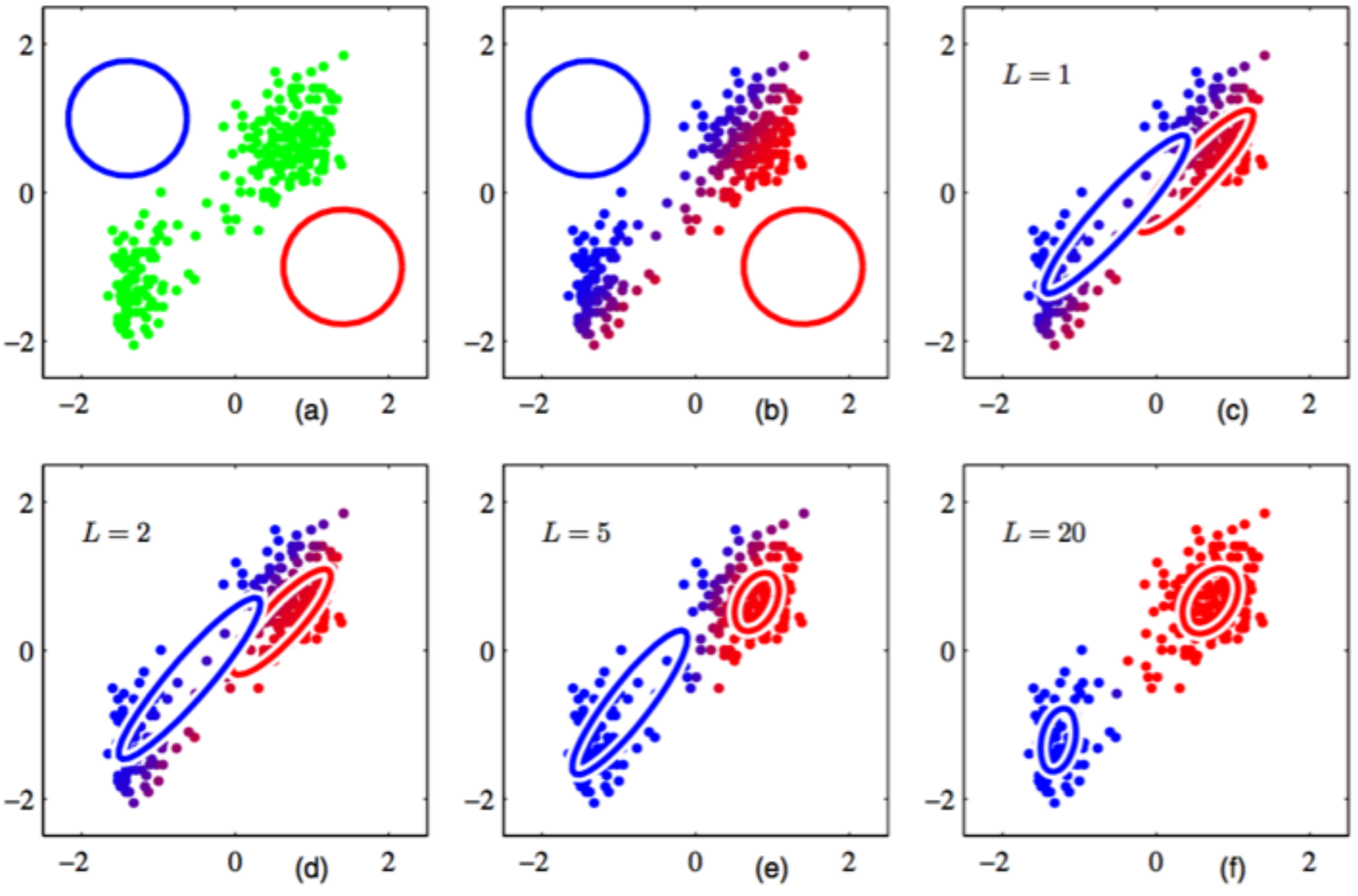



**Figure 9.8** Illustration of the EM algorithm using the Old Faithful set as used for the illustration of the $K$-means algorithm in Figure 9.1. See the text for details.

# Expectation maximisation

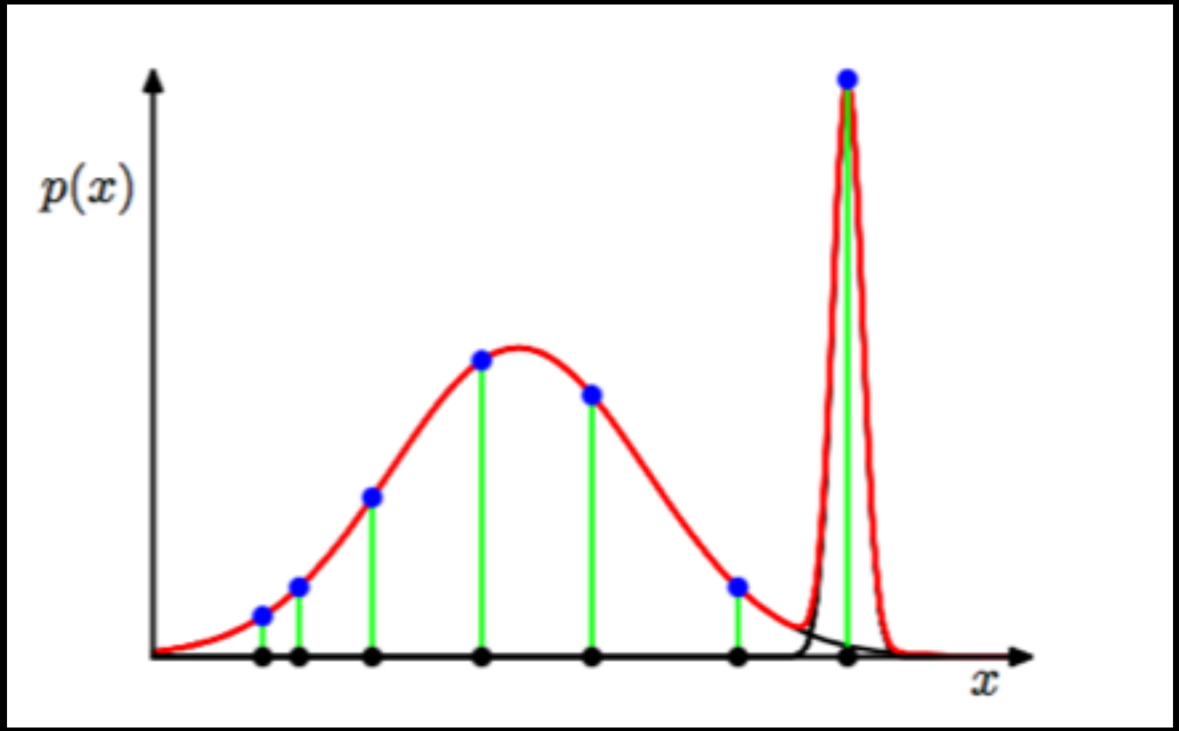
- Guaranteed to increase likelihood after every step by 0 or more
- (Find local minima)

# Other methods

- Optimisation
  - SIMPLEX
  - Differential evolution
  - Particle swarm
  - …

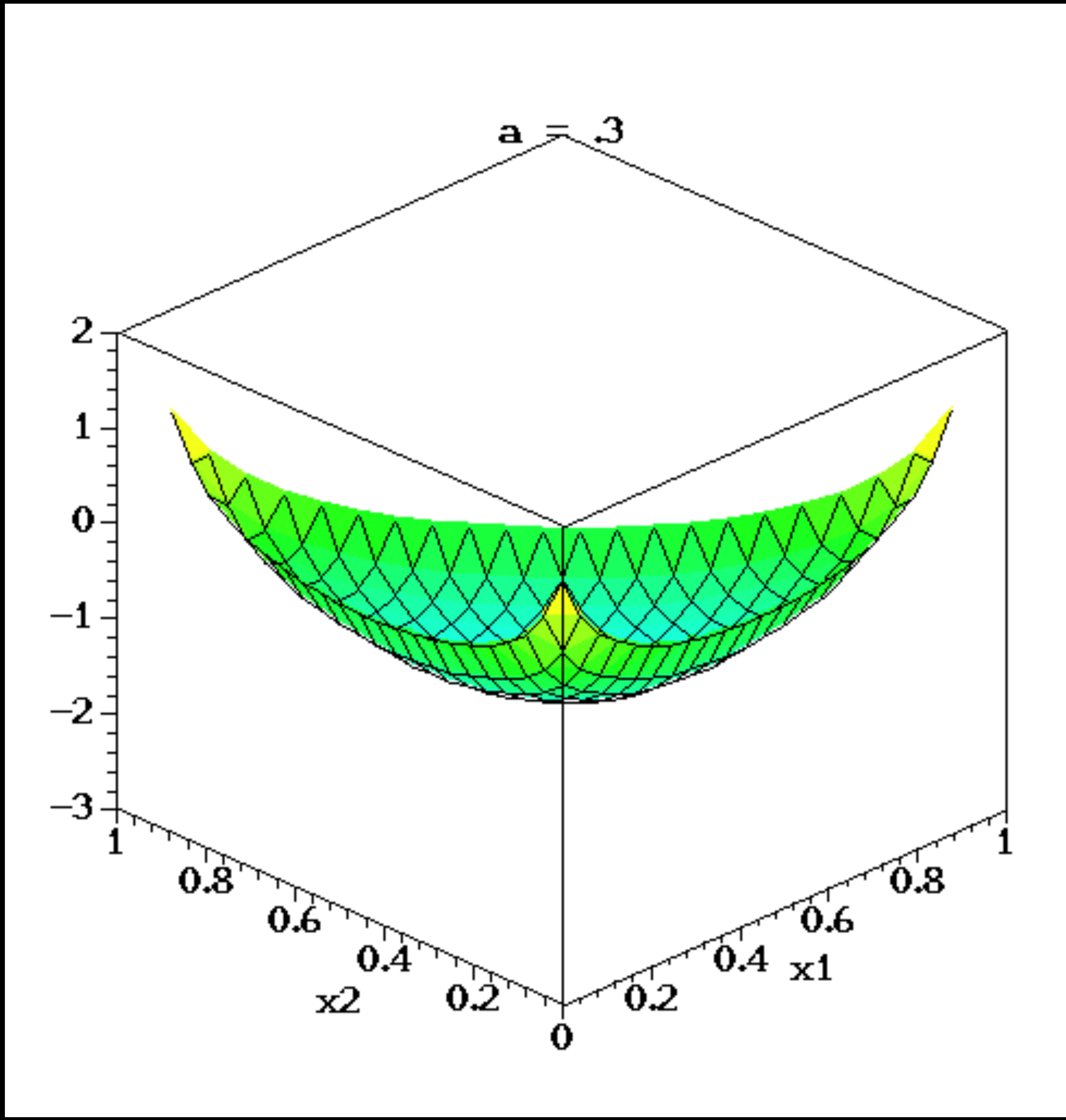
# Problems with ML

- “Wrong”
- No use of prior information
- More susceptible to singularities



# Bayesian perspective

- Use of priors
  - Loc and scale parameters
    - Normal/halfnormal/cauchy
  - Weights
    - **Dirichlet**



# Find posterior

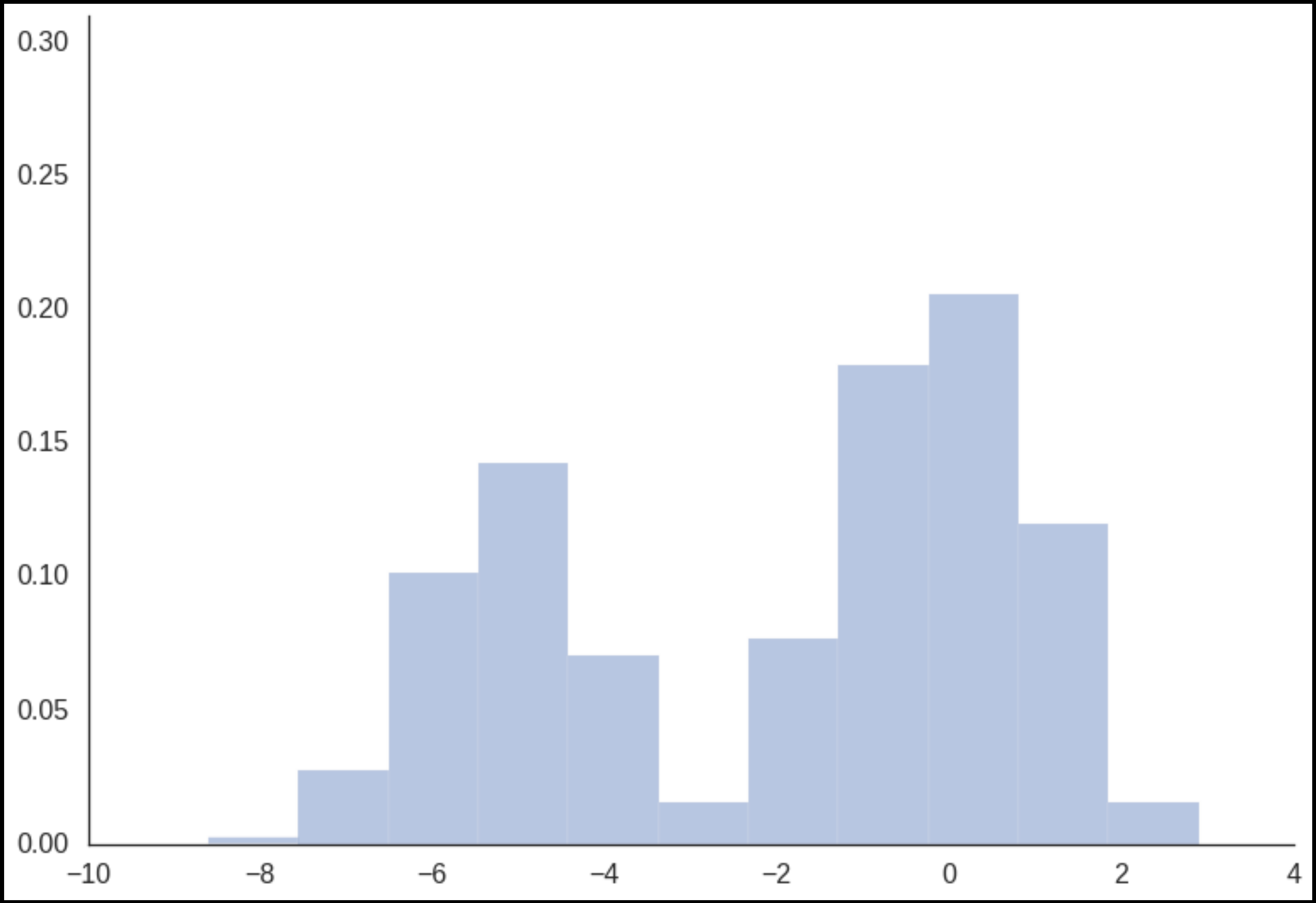
- MCMC sampling methods
  - Gibbs (Geman & Geman, 1984)
  - Metropolis-Hasting (Hastings, 1970)
  - NUTS (Hoffman et al., 2014)
  - DE-MCMC (Ter Braak, 2006)

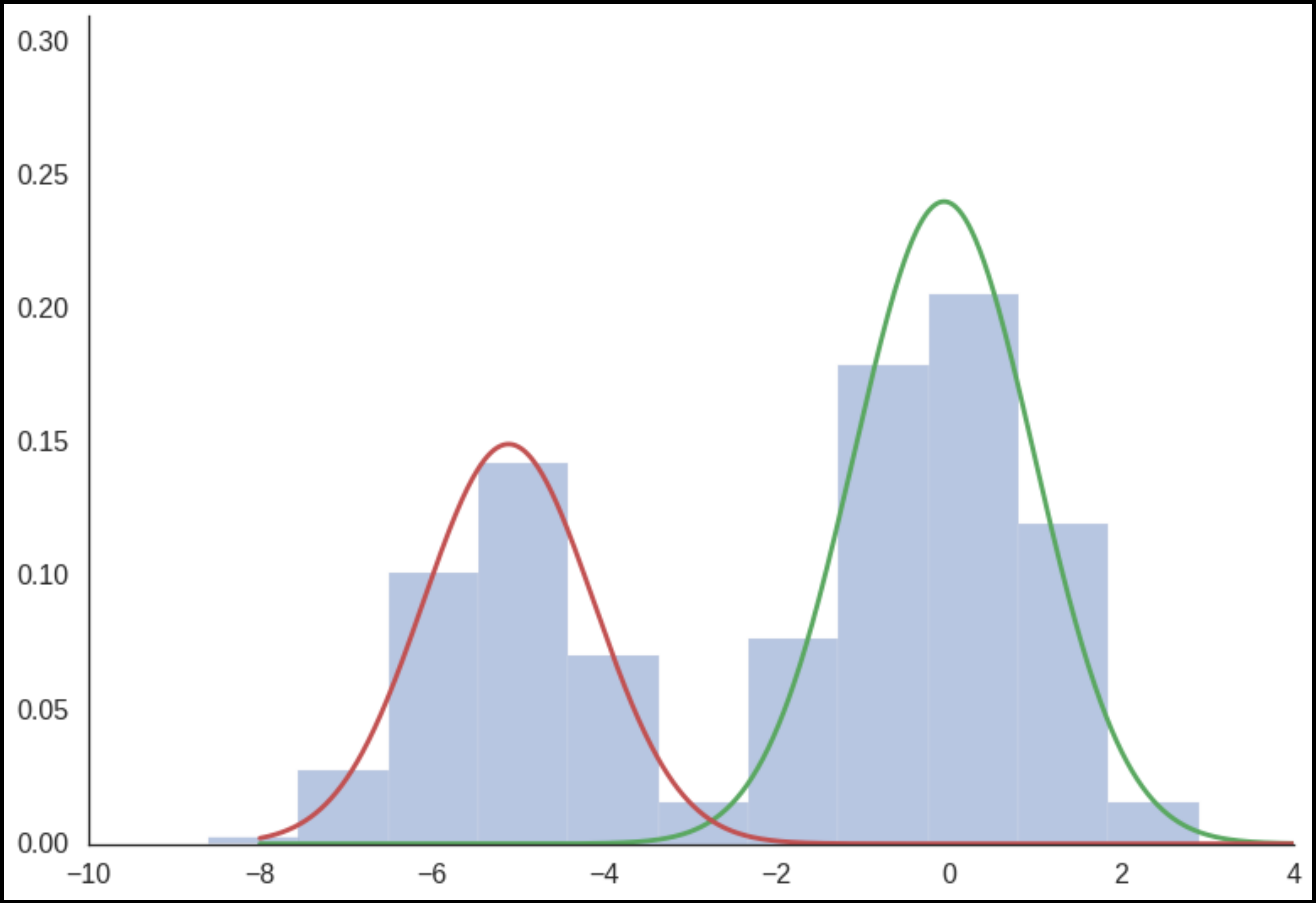
# Find posterior

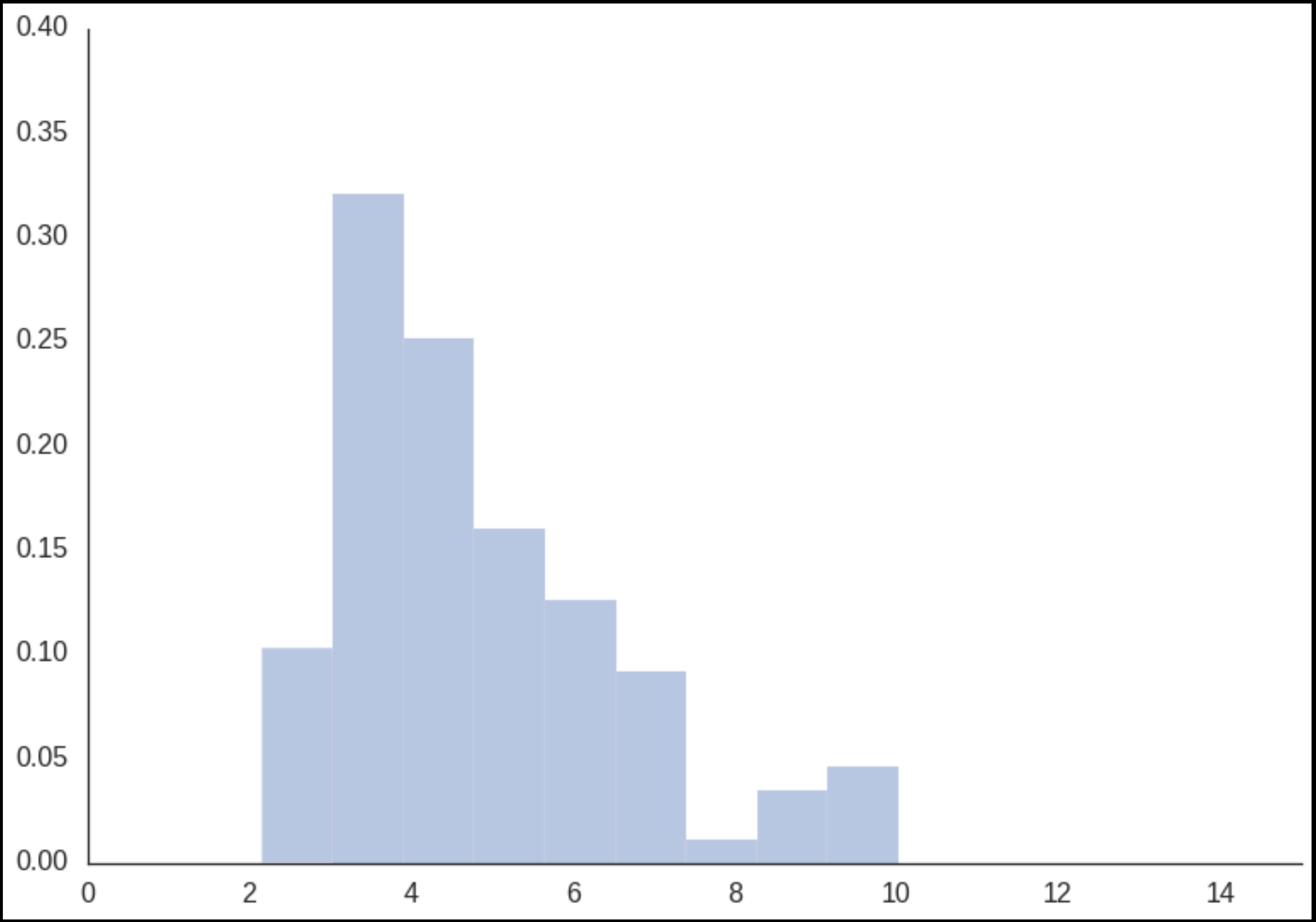
- Still hard:
  - Label switching
  - Local maxima
  - Irregular likelihood function

# How many clusters?

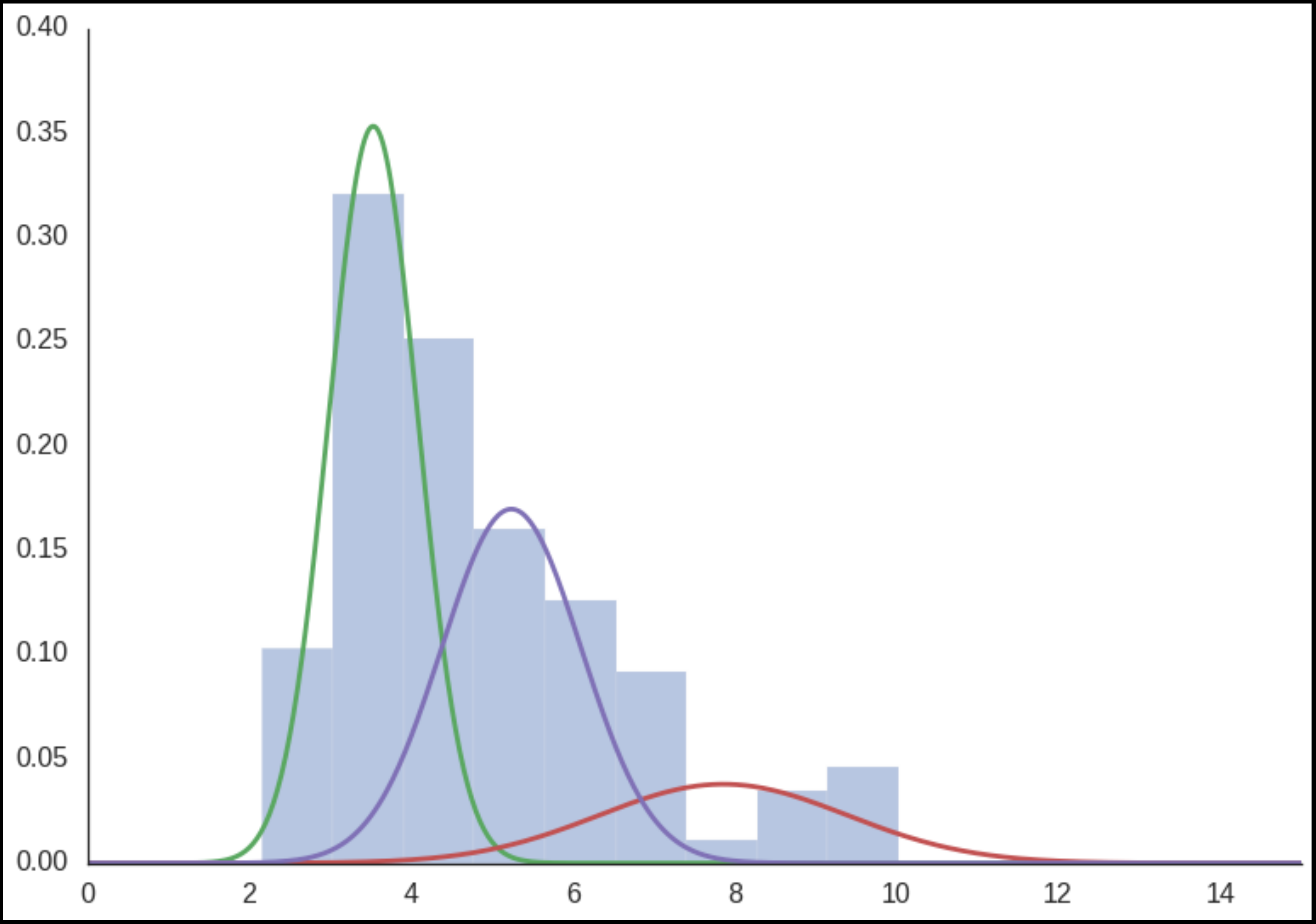
- Model comparison problem
  - BIC/AIC/DIC
  - Cross-validation
  - Bayes Factors (Marin & Robert, 2013)


0.30
0.25
0.20
0.15
0.10
0.05
0.00
−10
−8
−6
−4
−2
0
2
4

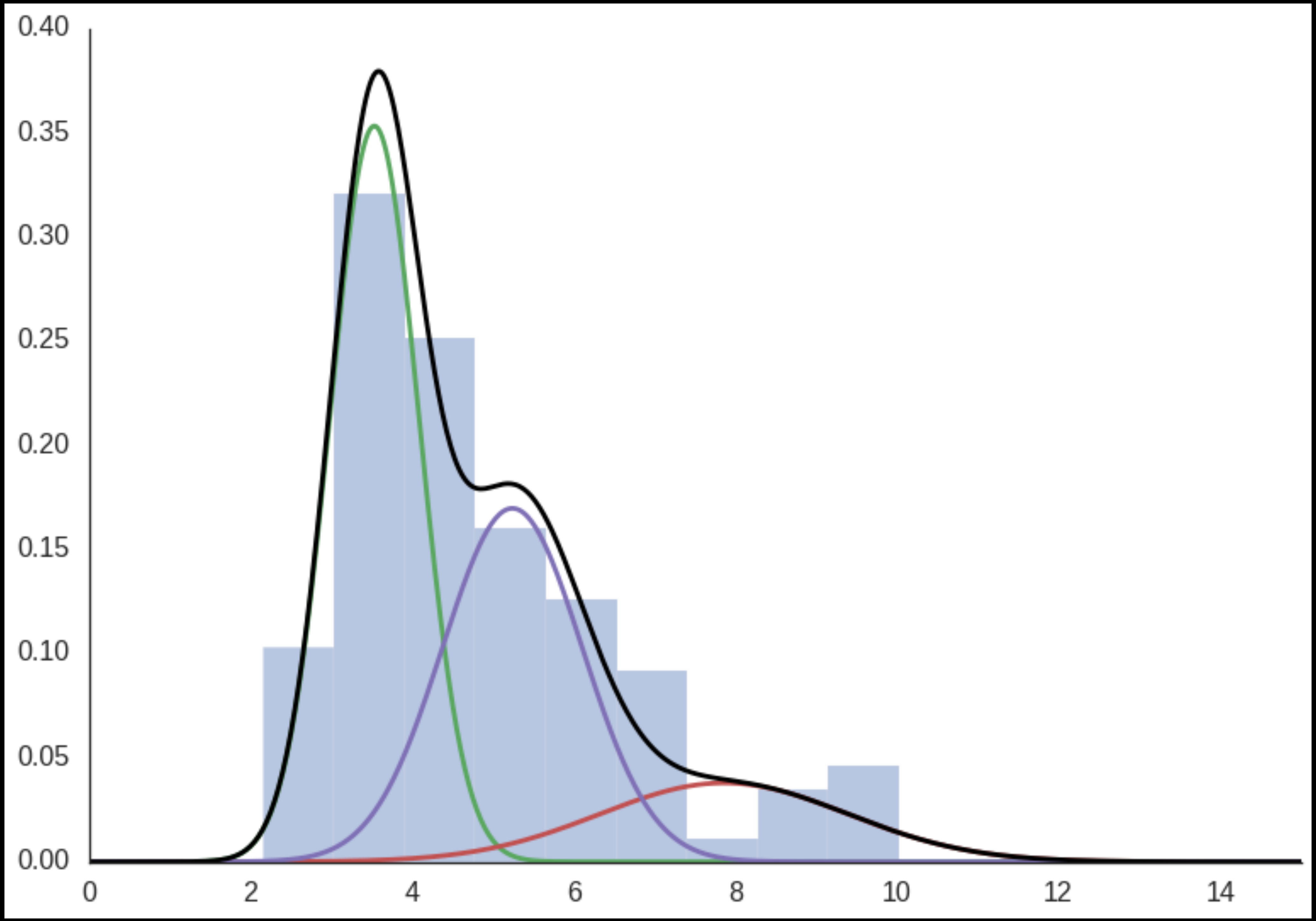

0.30
0.25
0.20
0.15
0.10
0.05
0.00
−10
−8
−6
−4
−2
0
2
4

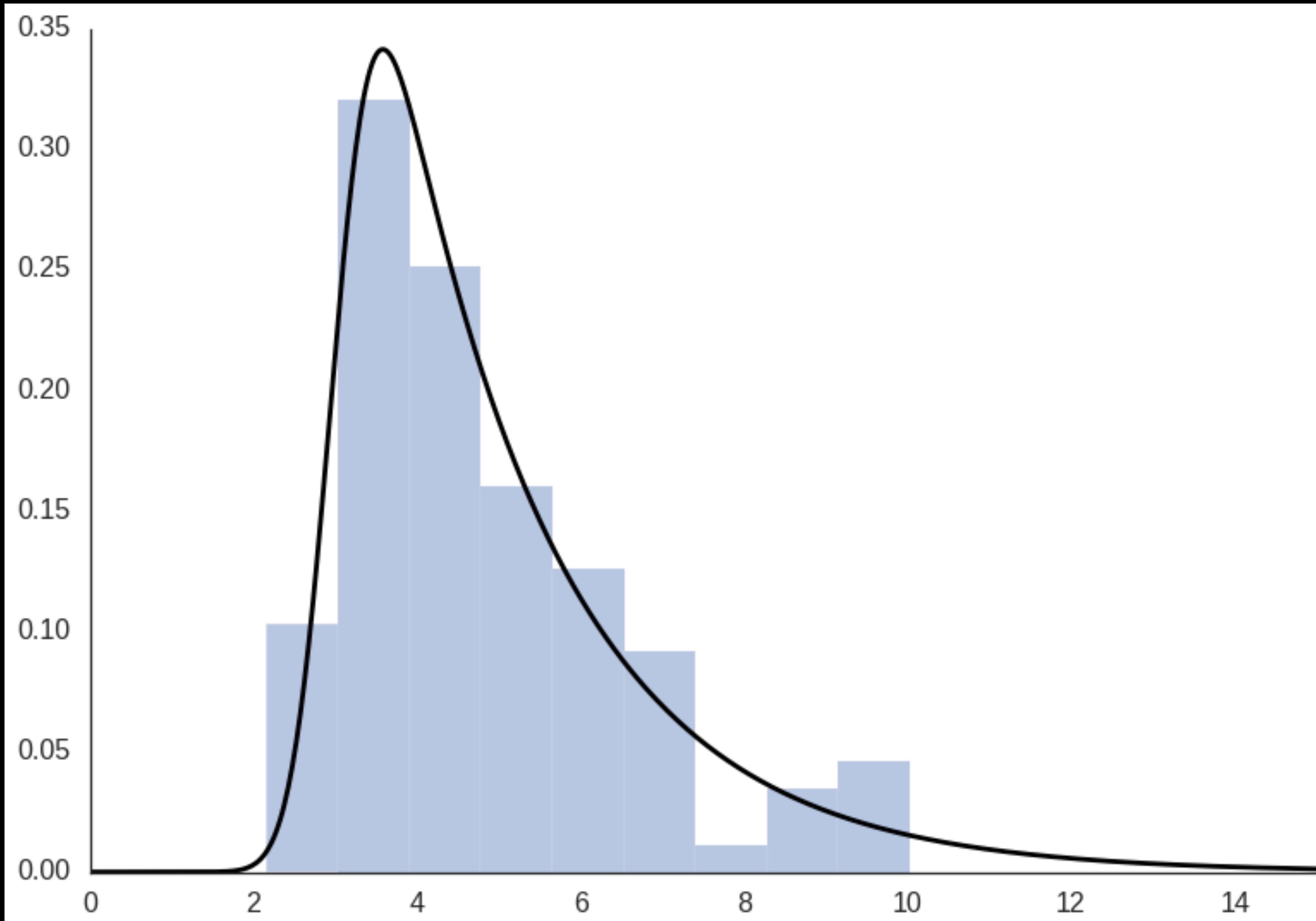


0.40
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
0
2
4
6
8
10
12
14



0.40
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
0
2
4
6
8
10
12
14


0.40
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
0
2
4
6
8
10
12
14


0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
0
2
4
6
8
10
12
14

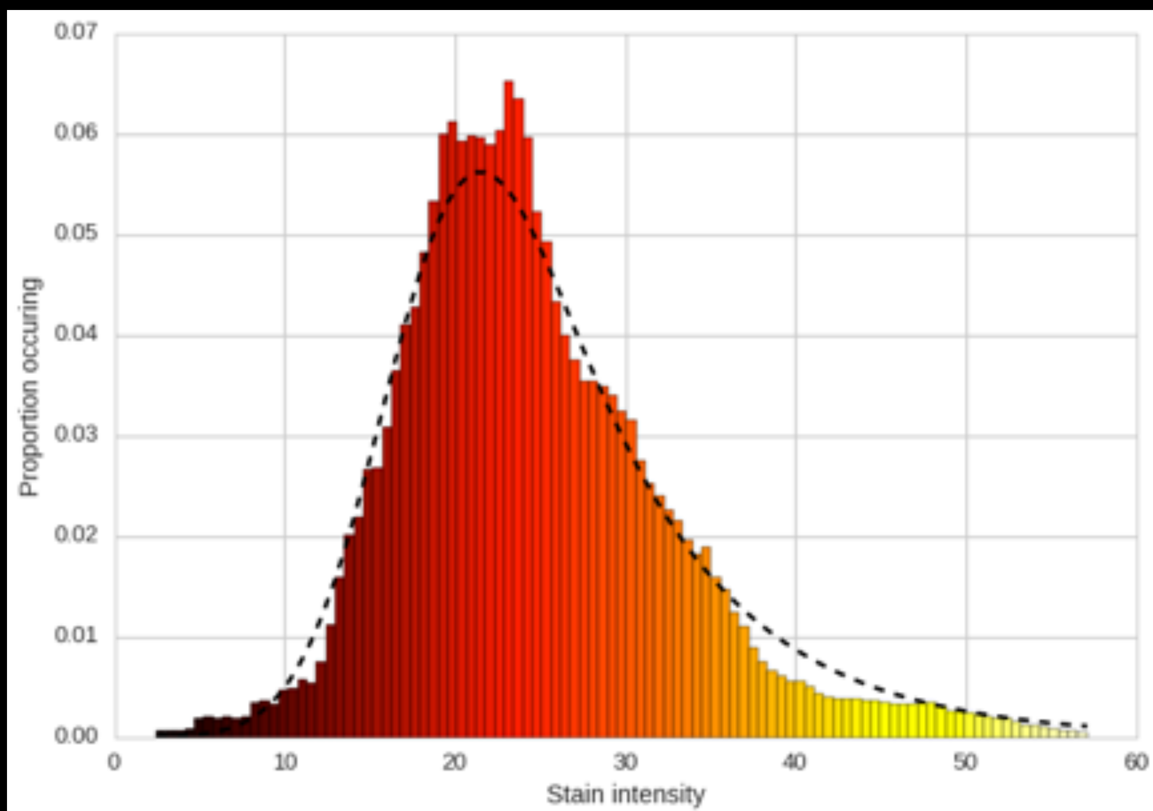
# Goal of a Mixture model

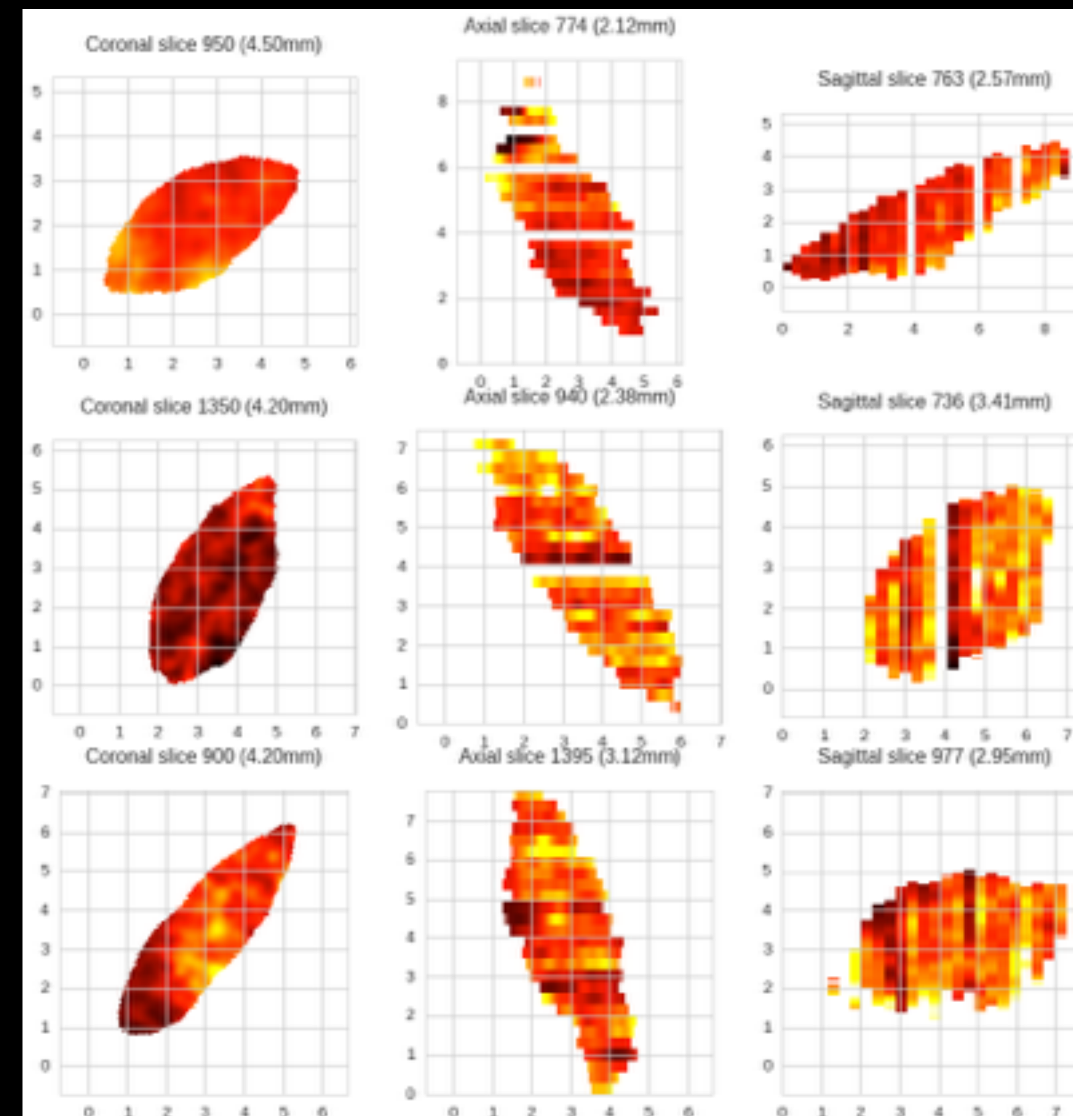
- Marin & Robert (2013): two goals of mixture models
  - Clustering perspective
  - Semiparametric perspective

# Conclusion

- Mixture models offer a class of models that can describe data coming from multiple distributions
- Their likelihood functions come with some additional challenges
- Parameters can be estimated using both ML and Bayesian techniques.
- There is a bag of tricks to find out “the” number of clusters, given a model specification.

# Infomercial




Friday 27th January, 16:00
G-1.<something>