When is Clustering Hard?

Nati Srebro
University of Toronto

Gregory Shakhnarovich
Massachusetts Institute of Technology

Sam Roweis
University of Toronto

Outline

- Clustering is Hard
- Clustering is Easy
- What we would like to do
- What we propose to do
- What we did

"Clustering"

- Clustering with respect to a specific model / structure / objective
- Gaussian mixture model
  - Each point comes from one of *k* "centers"
  - Gaussian cloud around each center
  - For now: unit-variance Gaussians, uniform prior over choice of center
- As an optimization problem:
  - Likelihood of centers:

$$\Sigma_i \log(\Sigma_j \exp -(x_i-\mu_j)^2/2)$$

  - *k*-means objective—Likelihood of assignment:

$$\Sigma_i \min_j (x_i-\mu_j)^2$$

Clustering is Hard

- Minimizing *k*-means objective is NP-hard
  - For some point configurations, it is hard to find the optimal solution.
  - But do these point configurations actually correspond to clusters of points?
- Likelihood-of-centers objective probably also NP-hard (I am not aware of a proof)
- Side note: for general metric spaces, hard to approximate *k*-mean to within factor < 1.5

"Clustering is Easy", take 1: Approximation Algorithms

- $(1+\varepsilon)$-Approximation for k-means in time $O(2^{(k/\varepsilon)^{const}} nd)$ **[Kumar Sabharwal Sen 2004]**

$\mu_1 = (5,0,0,0,\ldots,0)$
$\mu_2 = (-5,0,0,0,\ldots,0)$

$0.5\,N(\mu_1,I) + 0.5\,N(\mu_2,I)$

$\text{cost}([\mu_1,\mu_2]) \approx \sum_i \min_j (x_i-\mu_j)^2 \approx d\cdot n$

$\text{cost}([0,0]) \approx \sum_i \min_j (x_i-0)^2 \approx (d+25)\cdot n$

$\Rightarrow$ [0,0] is a (1+25/d)-approximation

- Need $\varepsilon < sep^2/d$, time becomes $O(2^{(kds)^{const}} n)$

“Clustering is Easy”, take 2: Data drawn from a Gaussian Mixture

$$x_1, x_2, \ldots, x_n \sim 1/k\, N(\mu_1,\sigma^2 I) + 1/k\, N(\mu_2,\sigma^2 I) + \cdots + 1/k\, N(\mu_k,\sigma^2 I)$$

$$|\mu_i - \mu_j| > s \cdot \sigma$$

Dasgupta 1999	$s > 0.5 d^{1/2}$	$n = \Omega(k^{\log^2 1/\delta})$	Random projection, then mode finding
Dagupta Schulamn 2000	$s = \Omega(d^{1/4})$ (large d)	$n = \text{poly}(k)$	2 round EM with $\Theta(k \cdot \log k)$ centers
Arora Kannan 2001	$s = \Omega(d^{1/4} \log d)$		Distance based
Vempala Wang 2004	$s = \Omega(k^{1/4} \log dk)$	$n = \Omega(d^3 k^2 \log(dk/s\delta))$	Spectral projection, then distances

all between-class distance > all within-class distance

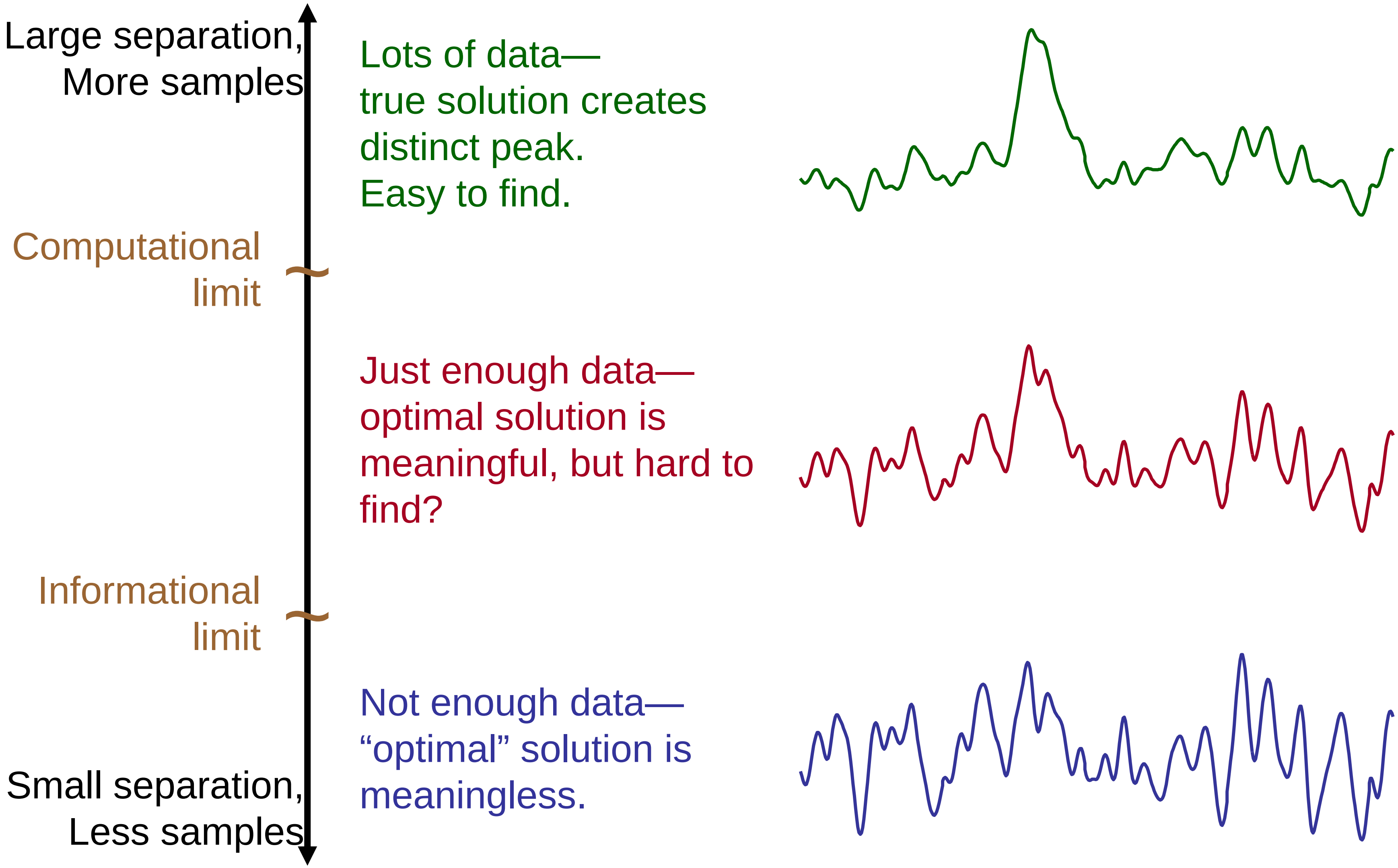
General mixture of Gaussians:

[Kannan Salmasian Vempala 2005] $s=\Omega(k^{5/2}\log(kd))$, $n=\Omega(k^2 d \cdot \log^5(d))$

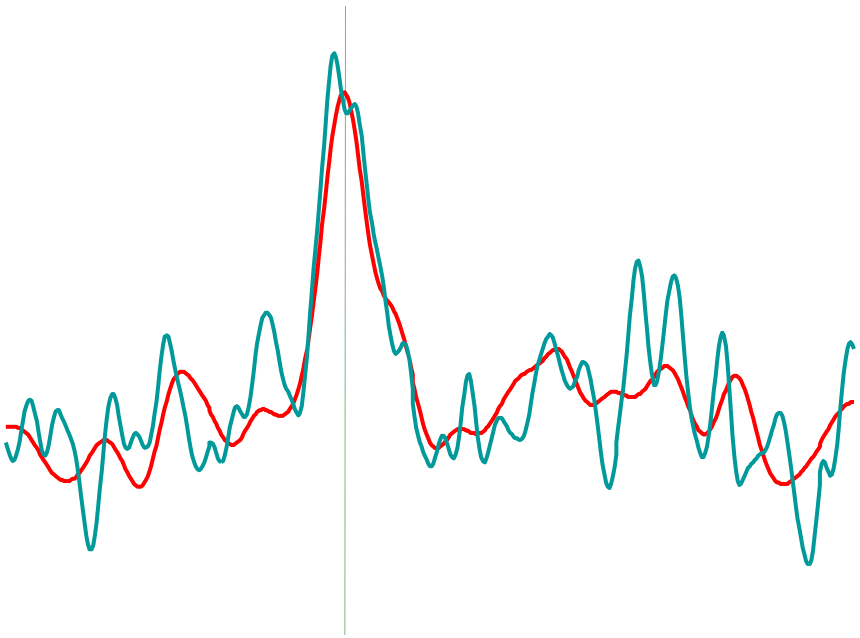
[Achliopts McSherry 2005] $s>4k+o(k)$, $n=\Omega(k^2 d)$

“Clustering isn’t hard—
it’s either easy,
or not interesting”

Effect of "Signal Strength"



Effect of “Signal Strength”



Infinite data limit:
E_X[cost(x;model)] = KL(true||model)

Mode always at true model

Determined by
- number of clusters (k)
- dimensionality (d)
- separation (s)

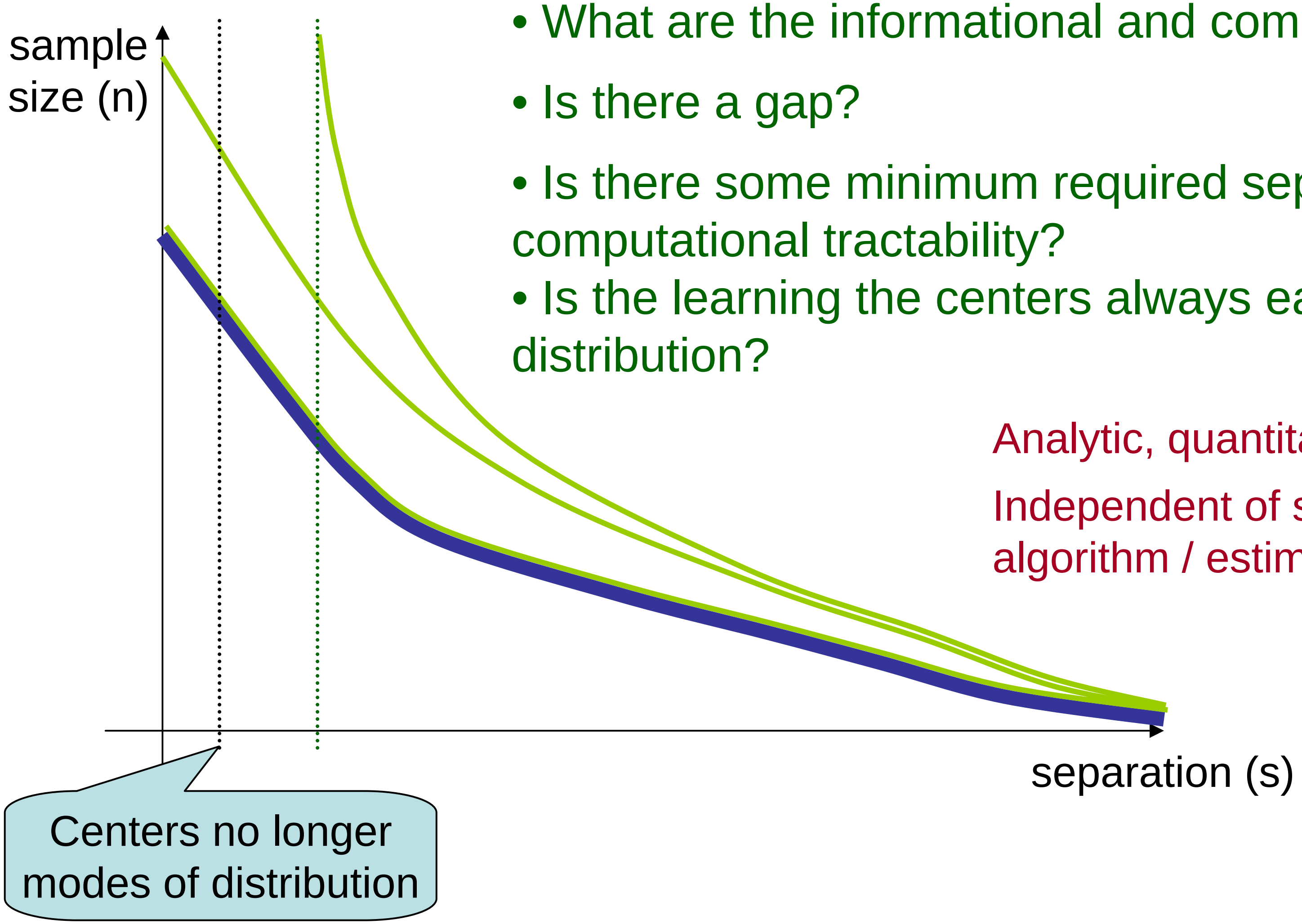
Actual log-likelihood

Also depends on:
- sample size (n)

“local ML model” ~ $N(\text{true}; \frac{1}{n} J_{Fisher}^{-1})$

[Redner Walker 84]

Informational and Computational Limits



- What are the informational and computational limits?
- Is there a gap?
- Is there some minimum required separation for computational tractability?
- Is the learning the centers always easy given the true distribution?

Analytic, quantitative answers.

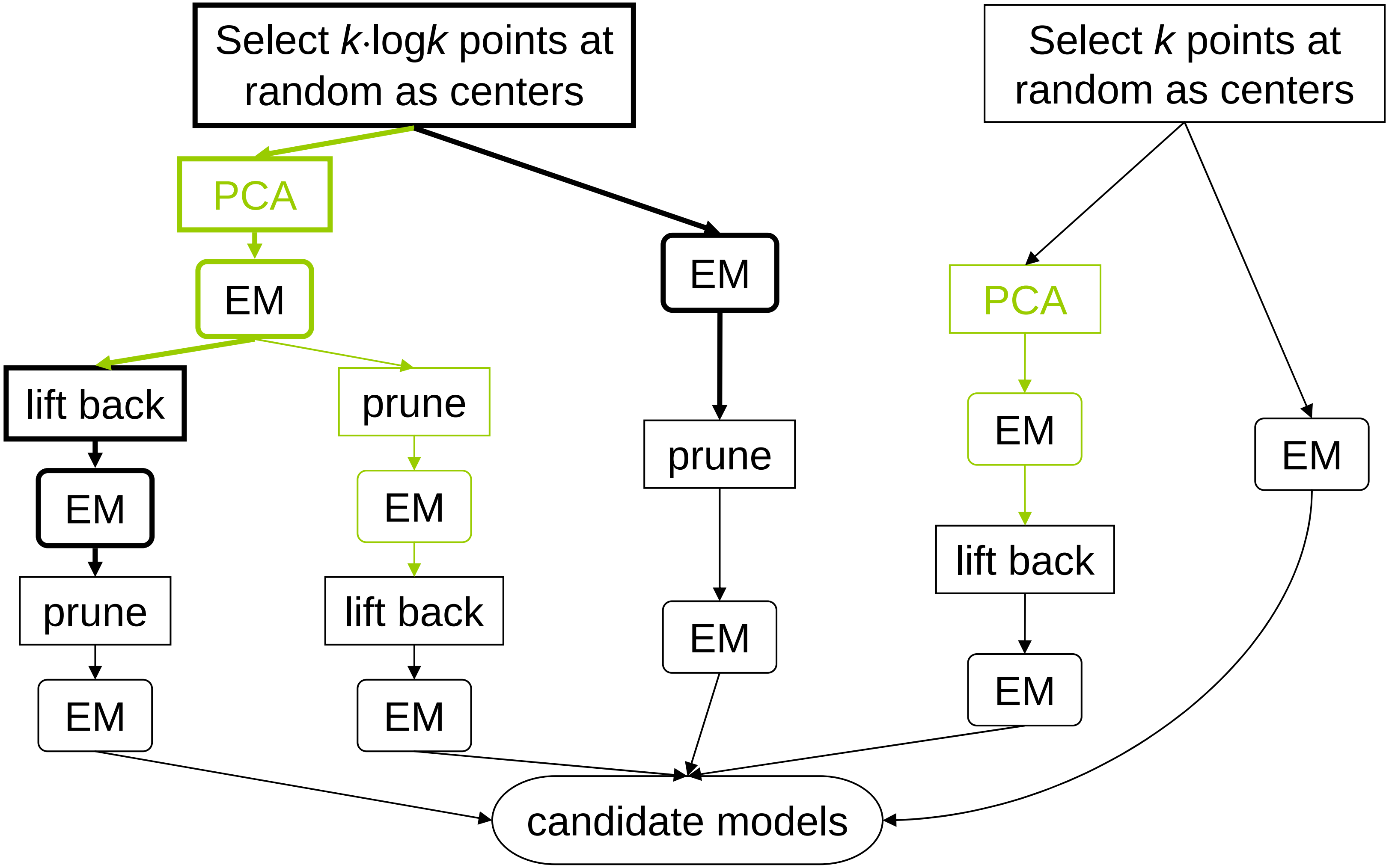
Independent of specific algorithm / estimator

Empirical Study

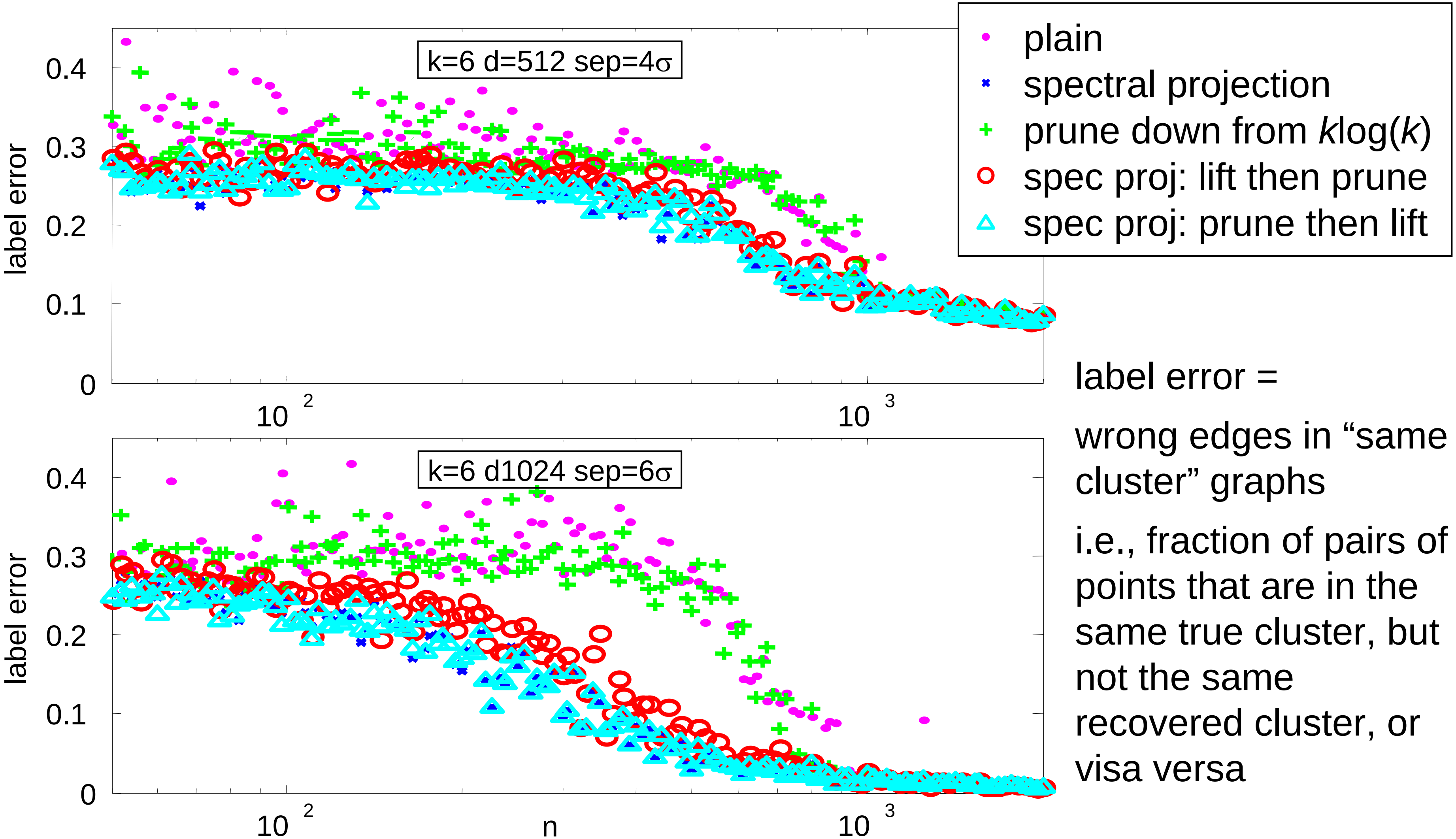
- Generate data from known mixture model
  - Uniform mixture of k unit variance spherical Gaussians in $\mathbb{R}^d$
  - Distance s between every pair of centers (centers at vertices of a simplex)
- Learn centers using EM
  - Spectral projection before EM
  - Start with $k \cdot \log k$ clusters and prune down to k
- Also run EM from true centers or true labeling

  (Cheating attempt to find ML solution)

EM with Different Bells and Whistles: Spectral Projection, Pruning Centers



EM with Different Bells and Whistles: Spectral Projection, Pruning Centers

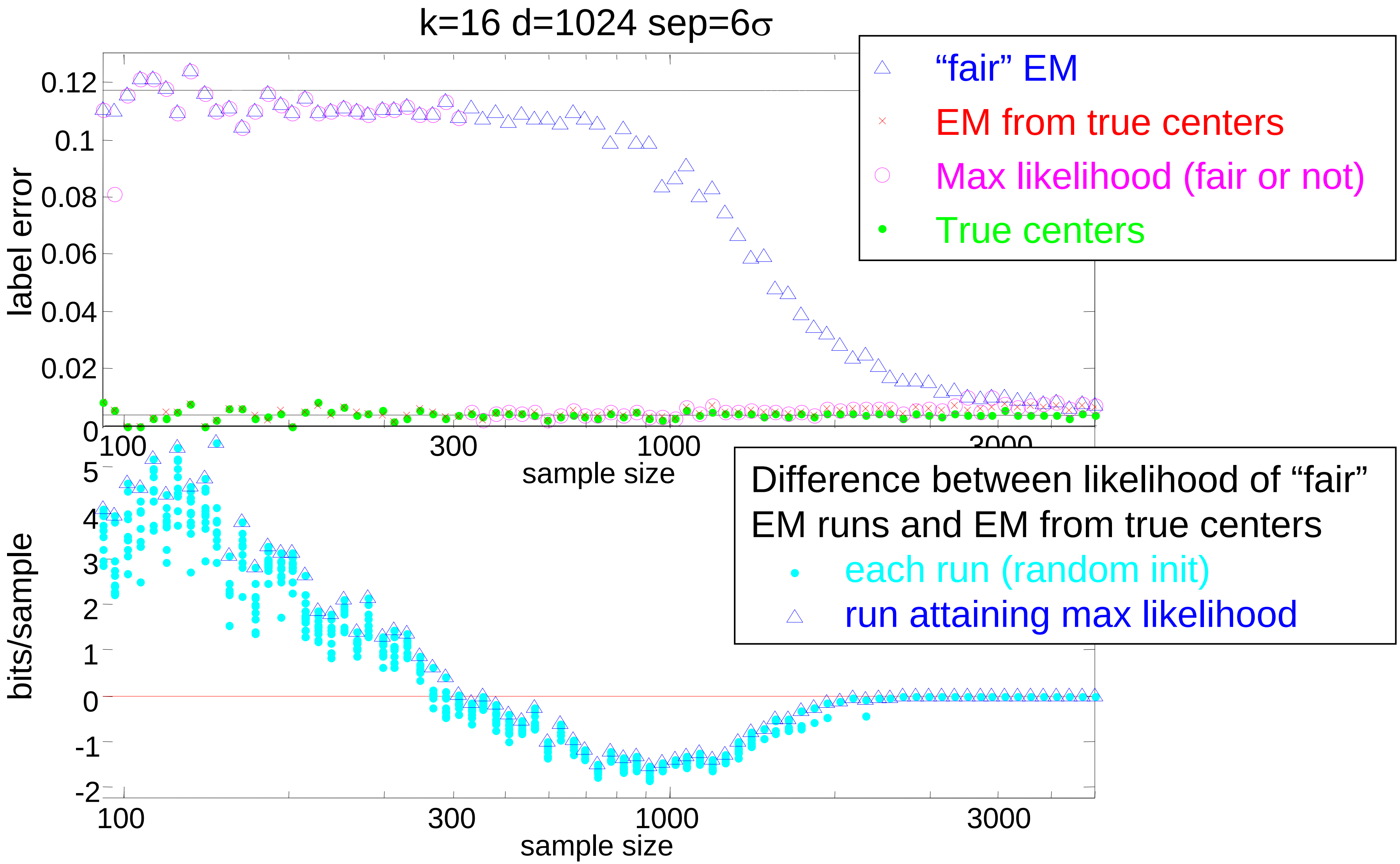


label error =

wrong edges in “same cluster” graphs

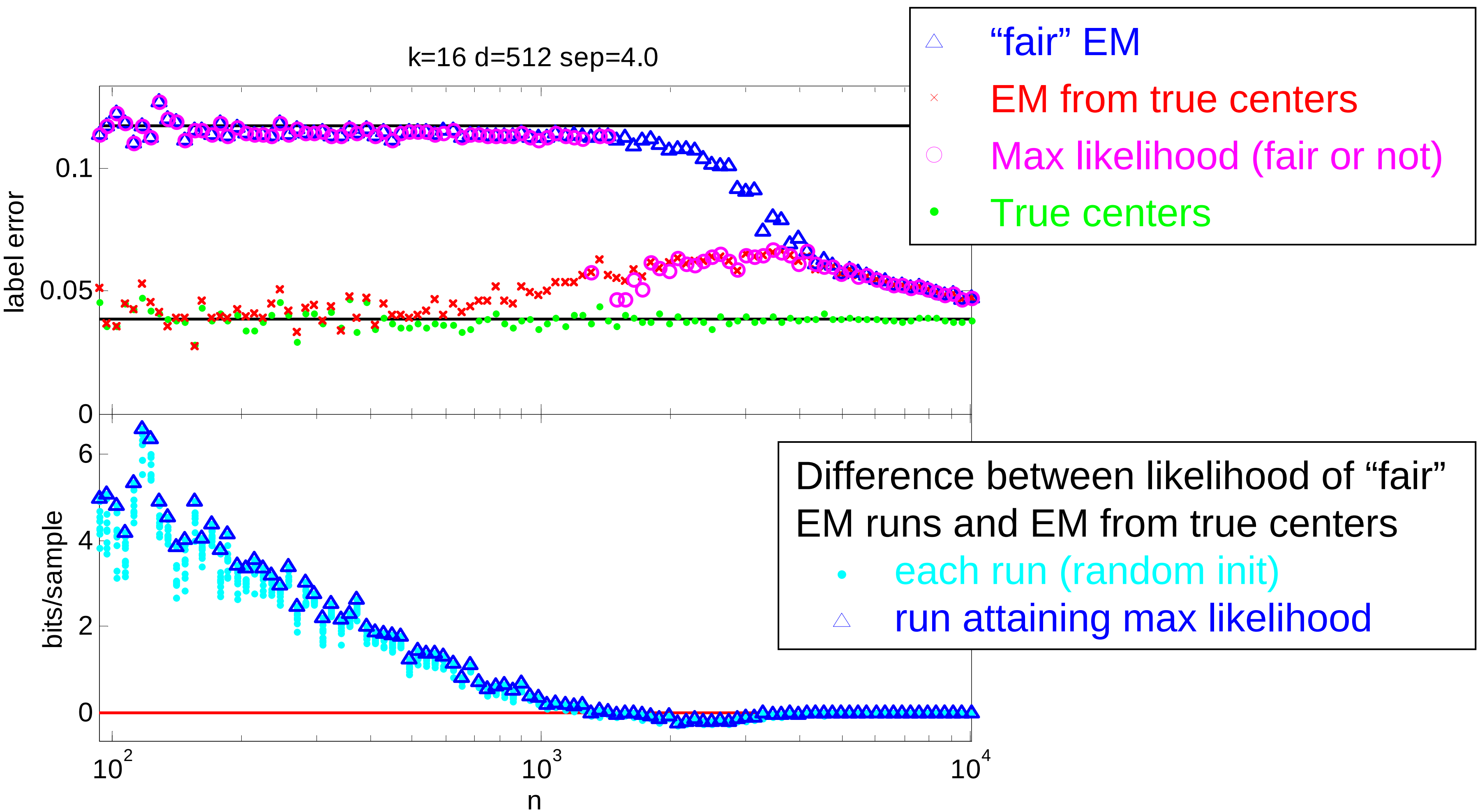
i.e., fraction of pairs of points that are in the same true cluster, but not the same recovered cluster, or visa versa

Behavior as a function of Sample Size



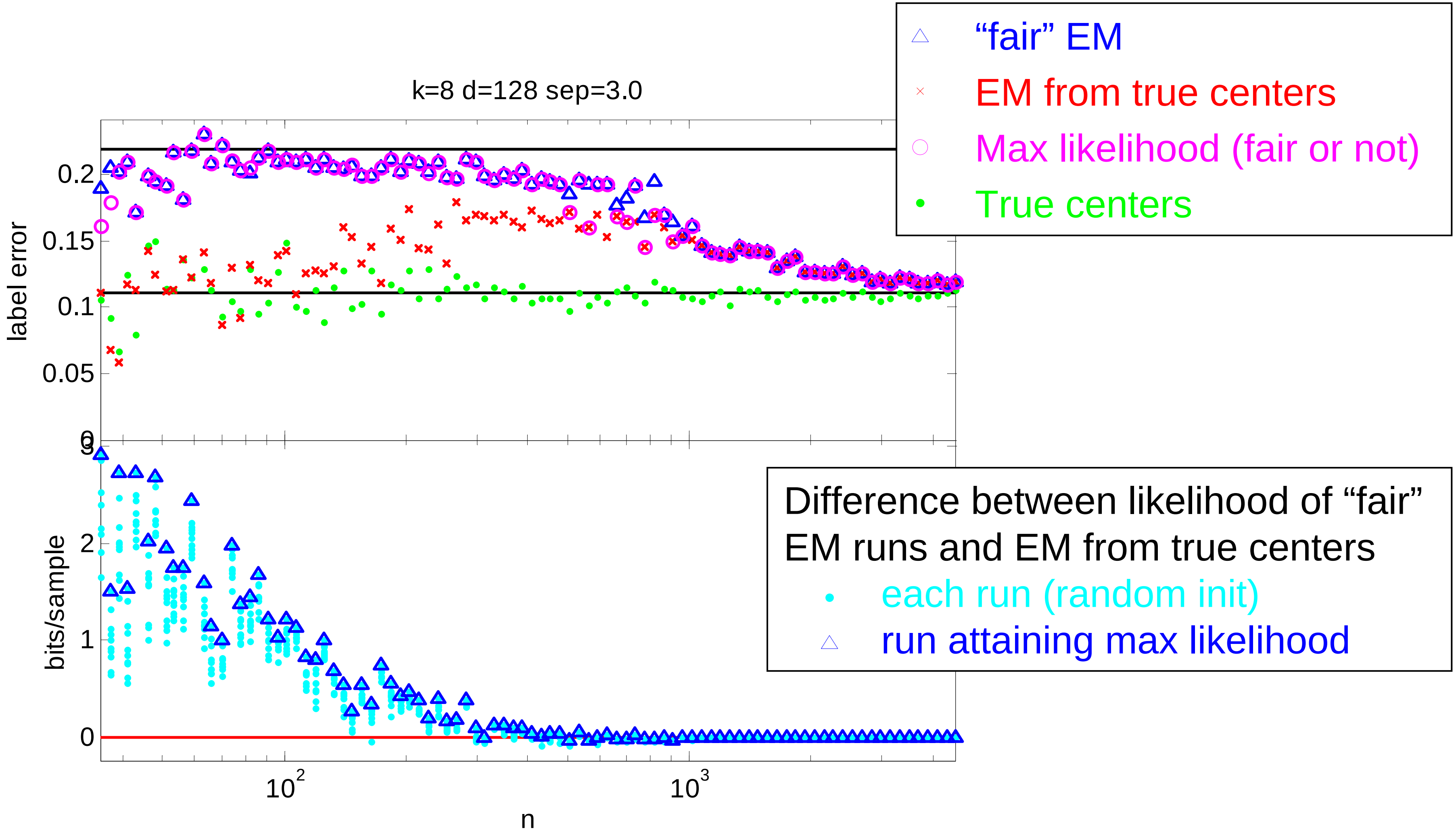
Behavior as a function of Sample Size:

Lower dimension, less separation

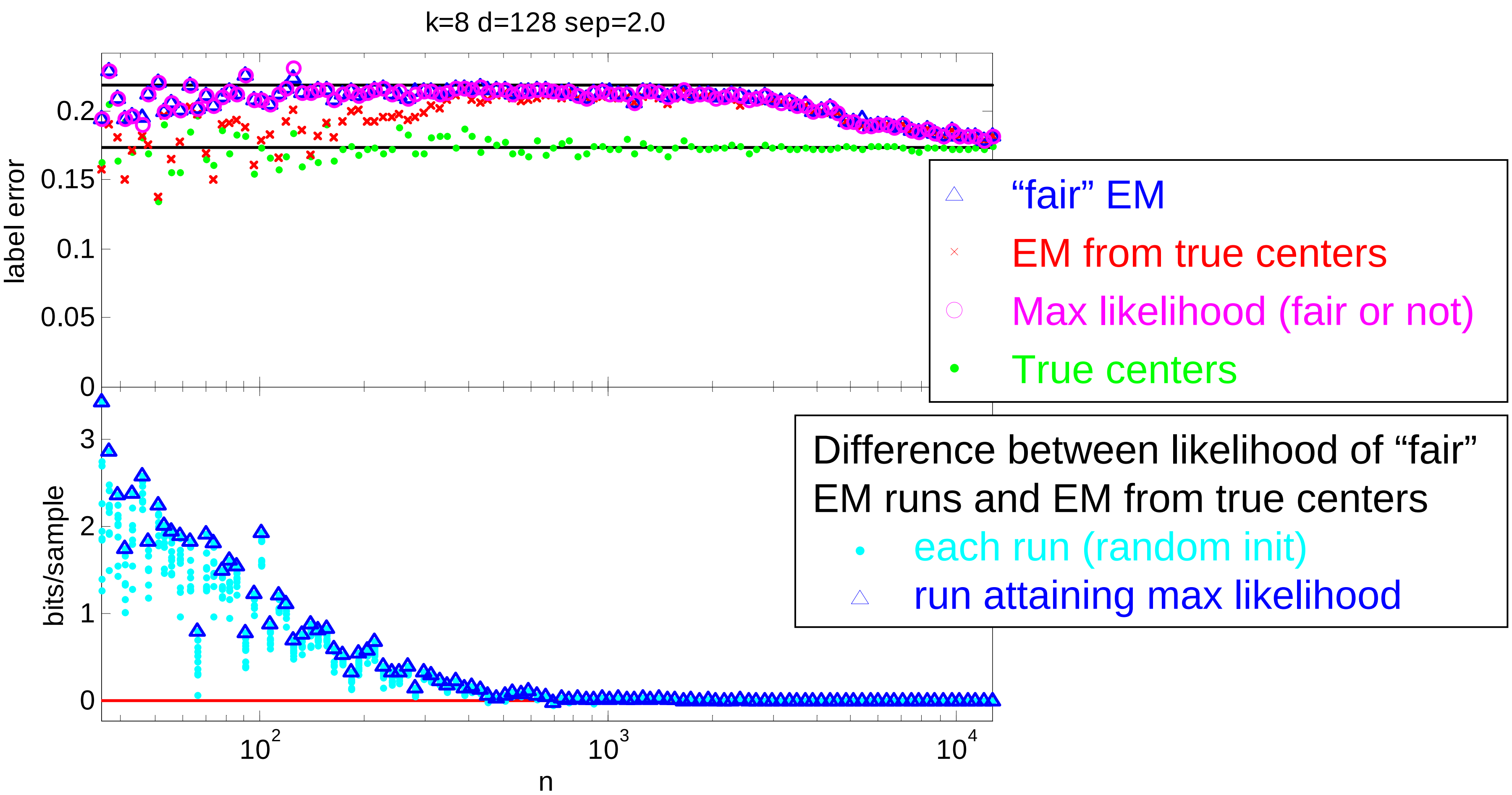


Behavior as a function of Sample Size:

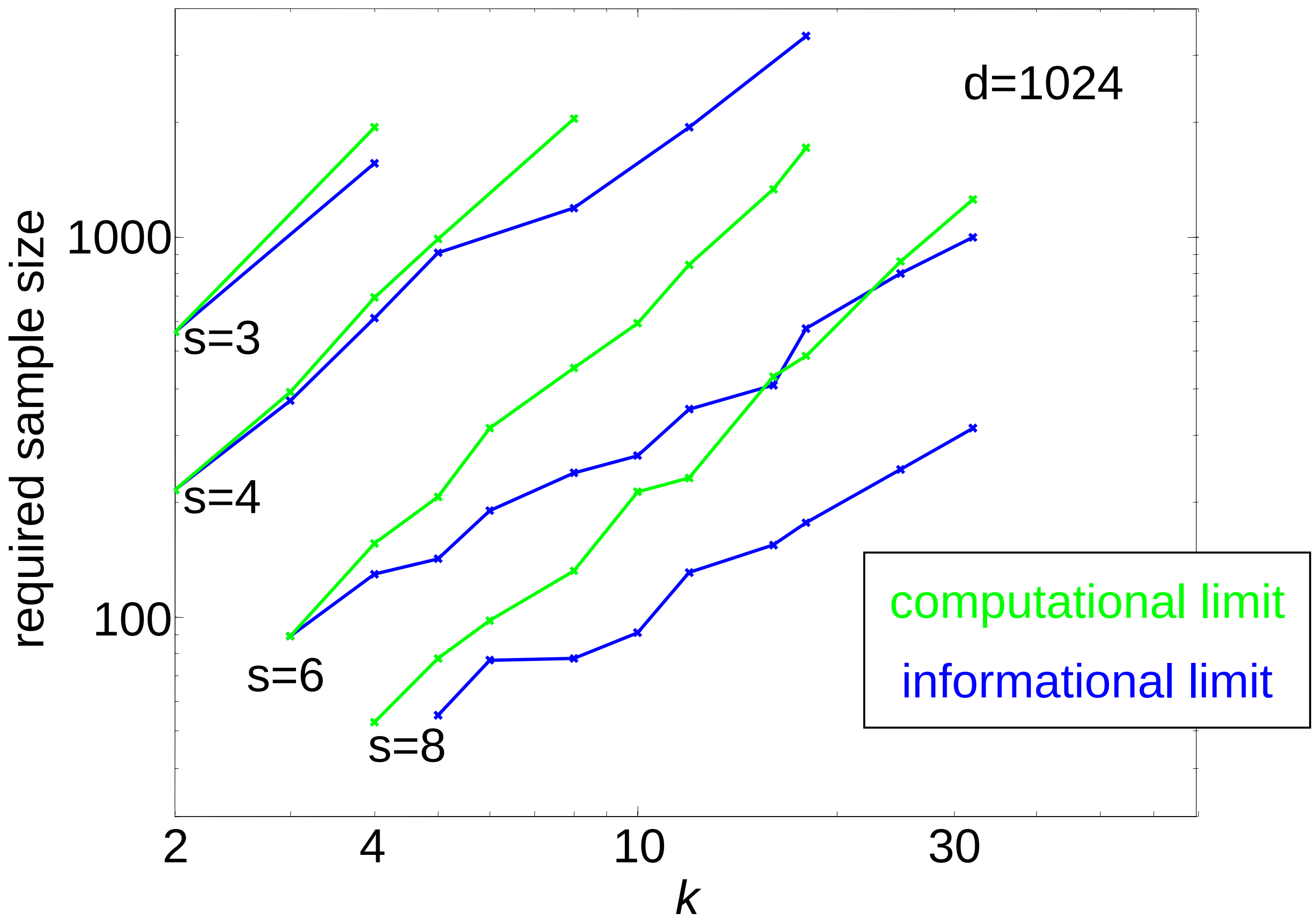
Lower dimension, less separation



Behavior as a function of Sample Size:
Lower dimension, less separation

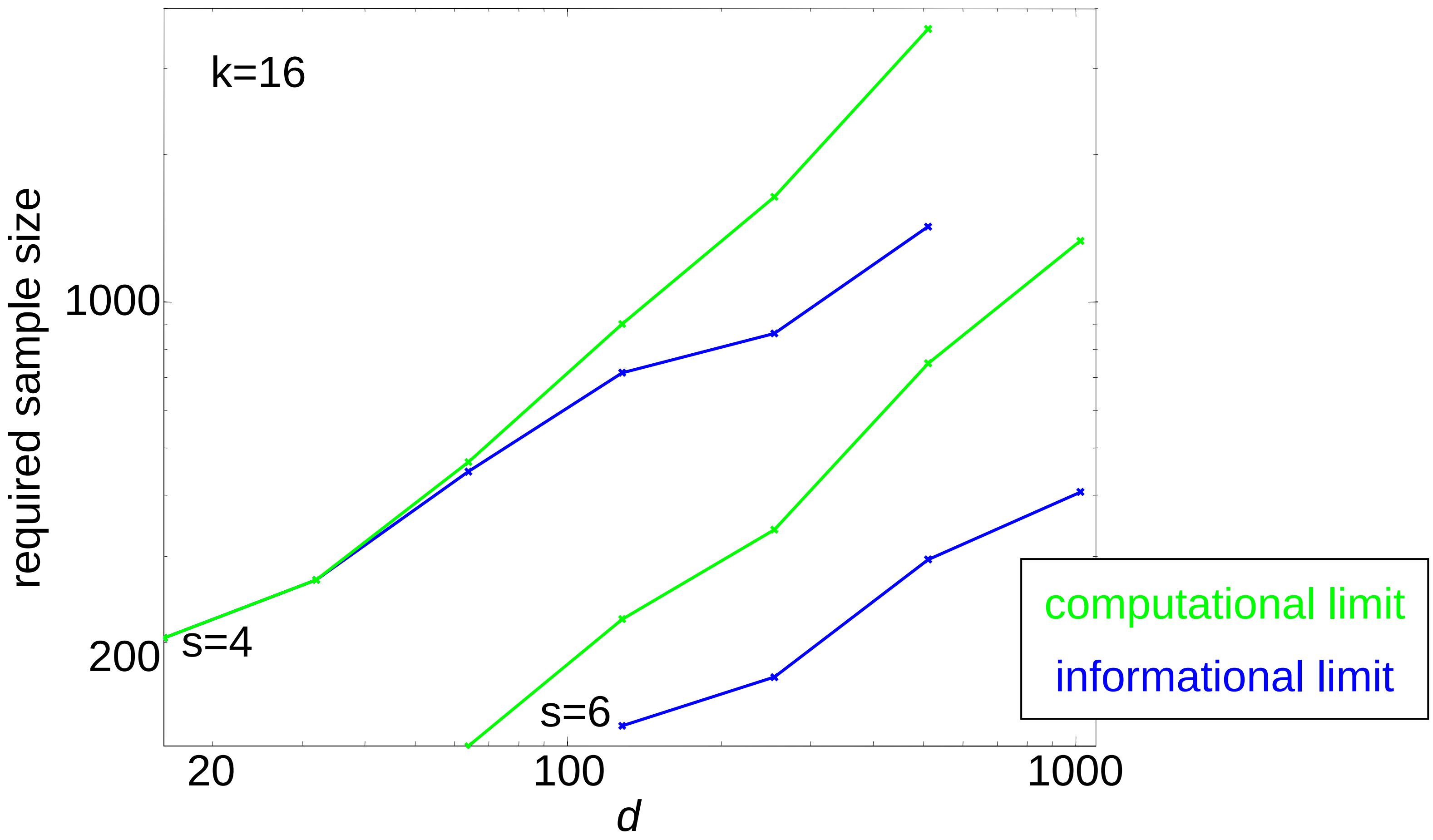


Informational and Computational Limits as a function of *k* and separation



$n \propto k^{1.5} - k^{1.6}$ for all *d*, separation

Informational and Computational Limits as a function of *d* and separation



Limitations of Empirical Study

- Specific optimization algorithm
  - Can only bound computational limit from above
- Do we actually find the optimum (max likelihood) solutions?
  - Can see regime in which EM fails even though there is a higher likelihood solution which *does* correspond to true model
  - But maybe there is an even higher likelihood solution the doesn’t?
- True centers always on a simplex
- Equal radius spherical Gaussians

Imperfect Learning

- So far, assumed data comes from specific model class (restricted Gaussian mixture)
- Even if data is not Gaussian, but clusters are sufficiently distinct and “blobby”, *k*-means / learning a Gaussian mixture model is easy.
- Can we give description of data for which this will be easy?

But for now, I’ll also be very happy with results on data coming from a Gaussian mixture…

Other Problems with Similar Behavior

- Graph partitioning (correlation clustering)
  - Hard in the worst case
  - Easy (using spectral methods) for large graphs with a "nice" statistically recoverable partition **[McSherry 03]**
- Learning structure of dependency networks
  - Hard to find optimal (max likelihood, or NML) structure in the worst case **[S 04]**
  - Polynomial-time algorithms for the large-sample limit **[Narasimhan Bilmes 04]**

Summary

- What are the informational and computational limits on Gaussian mixture clustering?
- Is there a gap?
- Is there some minimum required separation for computational tractability?
- Is the learning the centers always easy given the true distribution?
- Analytic, quantitative answers
- Hardness results independent of specific algorithm

- Limited empirical study:
  - There does seem to be a gap
  - Reconstruction via EM+spectral projection even from small separation (and a large number of samples)
  - Computational limit (very) roughly $\propto k^{1.5}d$