A Decoupled Approach to Exemplar-based Unsupervised Learning

Sebastian Nowozin, Gökhan Bakır

Department Empirical Inference for Machine Learning and Perception
Max Planck Institute for Biological Cybernetics
Tübingen, Germany

July 6, 2008

Unsupervised exemplar-based learning

- *Unsupervised learning*: recover structure from data.
- *Exemplar-based*: recovered structure is represented by a set of points in input space.

For example

- *Vector Quantization*: learn codebook vectors
- *Clustering*: learn set of representative clusters
- *Mixture-model Density Estimation*: Learn a set of component coordinates to represent a density

Exemplar-based Unsupervised Learning

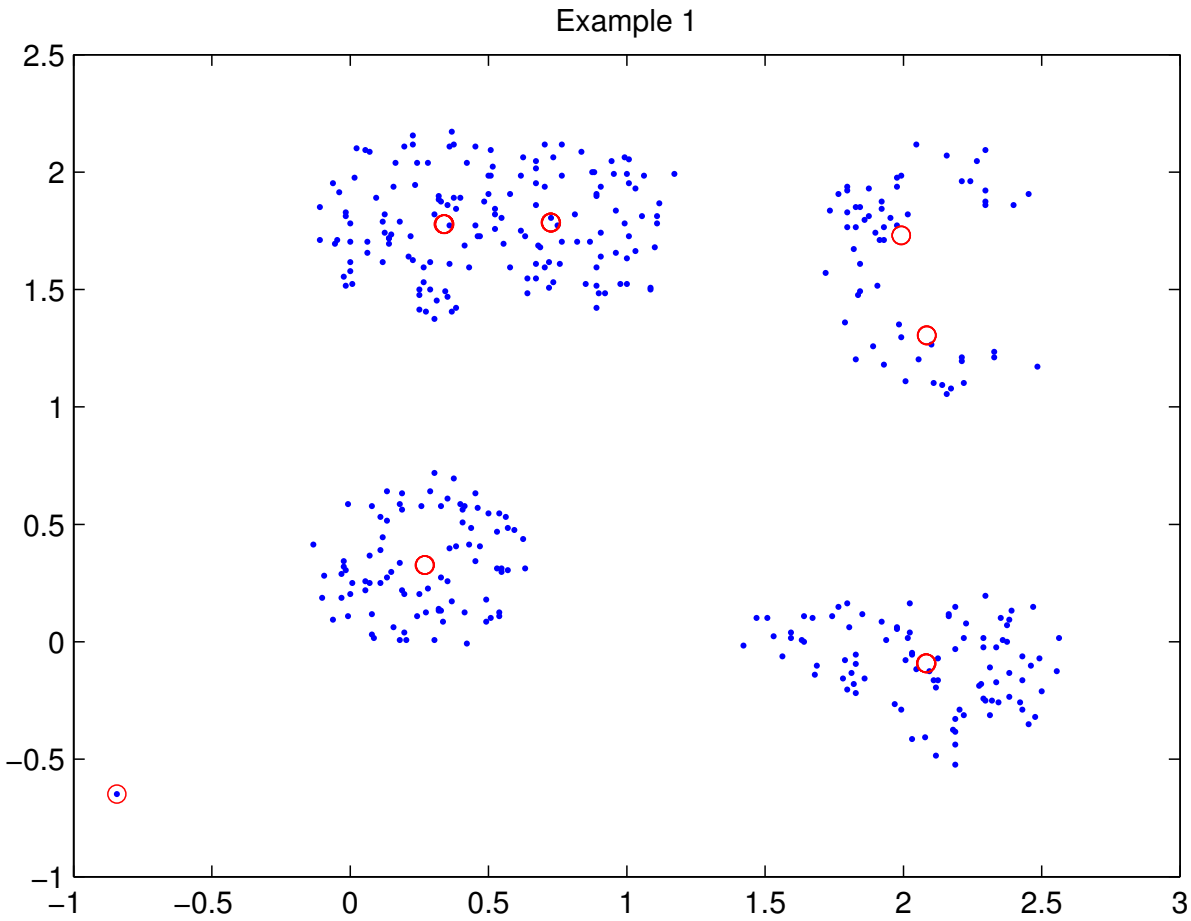

Figure: A set of points, "clusters"

Exemplar-based Unsupervised Learning

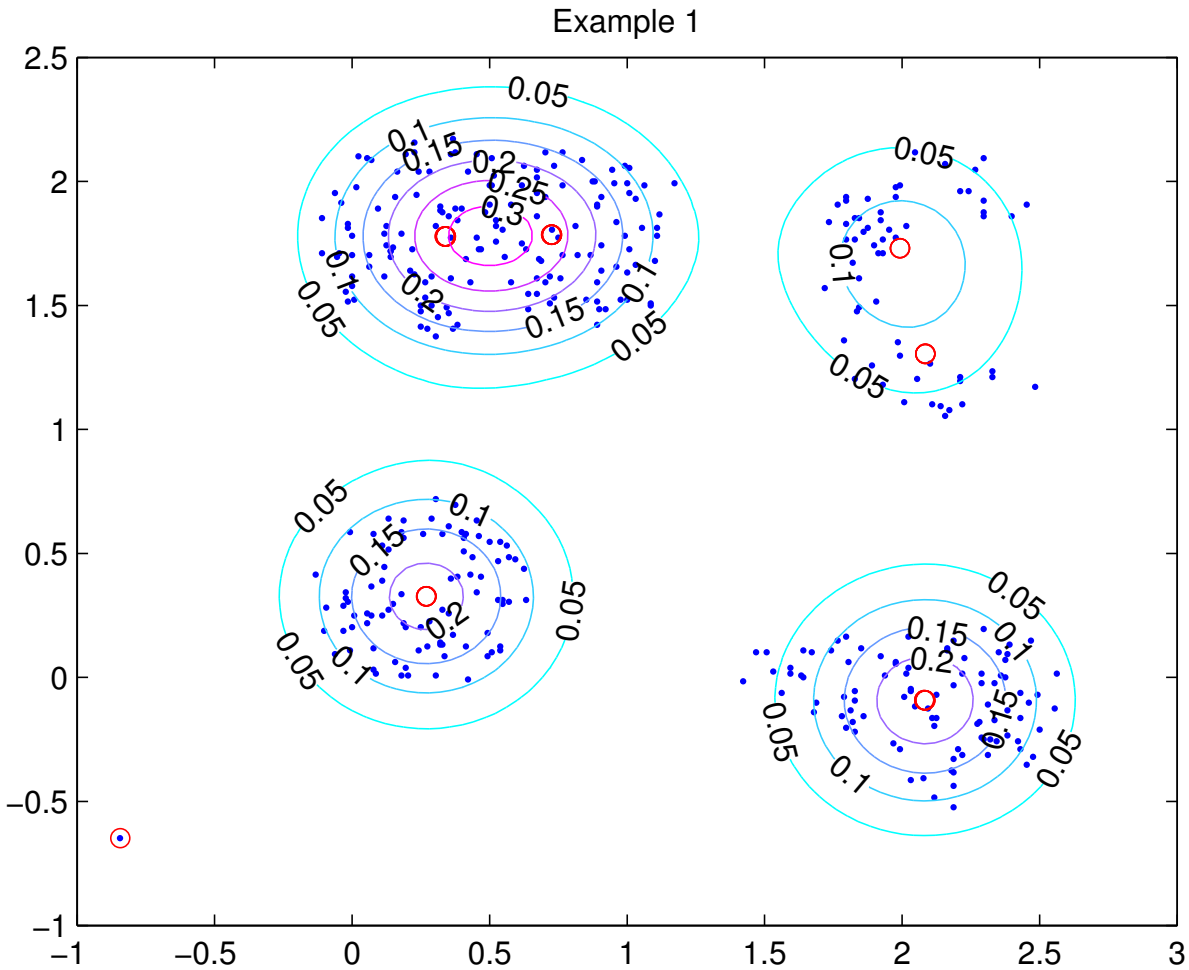

Figure: A set of points, "clusters", supported structure

Convex vs. Non-convex

Traditional standard algorithms

- *Vector Quantization*: LVQ
- *Clustering*: k-means
- *Mixture-model Density Estimation*: EM on Gaussian mixture

These are all efficient, well-behaved and require as main regularization a *number of components* K.

Recently, a number of *convex* approaches have been proposed.

- "Convex Clustering" [4], NIPS 2007
- Kernel Vector Quantization [7], AISTATS 2001
- (Boosting Density Estimation [5]), NIPS 2002

Convex Clustering

"Convex Clustering" is misnamed, as it is density estimation.

$$\begin{aligned} \max_{\mathbf{q}} \quad & \frac{1}{N}\sum_{i=1}^{N} \log\left[\sum_{j=1}^{N} q_j e^{-\beta d_\phi(\mathbf{x}_i,\mathbf{x}_j)}\right] \\ \text{sb.t.} \quad & \|\mathbf{q}\|_1 = \mathbf{1}, \\ & \mathbf{q} \geq \mathbf{0}. \end{aligned}$$

where d_ϕ is a Bregman divergence, $\mathbf{q}$ are component weights.

- Framework of Bregman Clustering [1],
- Maximize the log-likelihood of the model,
- Subject to the constraint that the resulting model is a proper mixture model.

In the optimum solution of the model, a sparse set of exemplars is selected, allowing the interpretation as clusters.

Kernel Vector Quantization

KVQ [7]

- Selects a set of codebook vectors from the training samples,
- Minimum distance from any training sample to its nearest codebook vector is bounded above by a given *maximum distortion h*,
- Balls of radius *h* around codebook vectors cover the entire training set

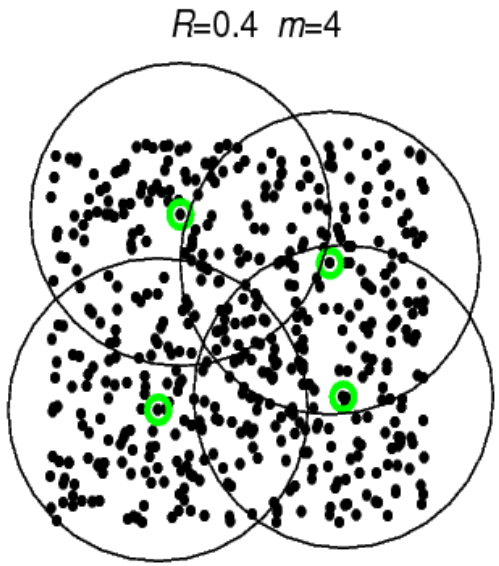

Figure: A covering produced by KVQ. From Tipping and Schölkopf [7].

Kernel Vector Quantization

KVQ [7]

- Selects a set of codebook vectors from the training samples,
- Minimum distance from any training sample to its nearest codebook vector is bounded above by a given *maximum distortion h*,
- Balls of radius *h* around codebook vectors cover the entire training set

Equivalent reformulation of the original linear program.

$$\begin{aligned} \max_{\mathbf{q},\rho} \quad & \rho \\ \text{sb.t.} \quad & K\mathbf{q} \geq \rho\mathbf{1}, \\ & \|\mathbf{q}\|_1 = 1, \\ & \mathbf{q} \geq \mathbf{0}. \end{aligned} \tag{1}$$

Here K is a (N, N) matrix with $K_{i,j} = I(\|\mathbf{x}_i - \mathbf{x}_j\| \leq h)$.

Observation

1. In CC and KVQ, the training set is used for two purposes
   - As set to be "explained", as measured by the objective,
   - As candidate set for selecting codebook/cluster vectors
2. However, these two sets can be chosen independently.
3. Therefore:
   - Explain the training set (eg. log-likelihood/covering wise),
   - Using an independent set of candidates.
4. Experiment: create candidates from a dense grid and see what happens on a toy data set. Here we use the Convex Clustering log-likelihood objective with Gaussian multivariate normal (unnormalized).

Motivational Experiment

Note: if training data is locally Gaussian, with high probality a sample exists nearby the true mean of the Gaussian. For non-Gaussian structures like the ring, no such example exists.

Hence, we can improve by using more samples.

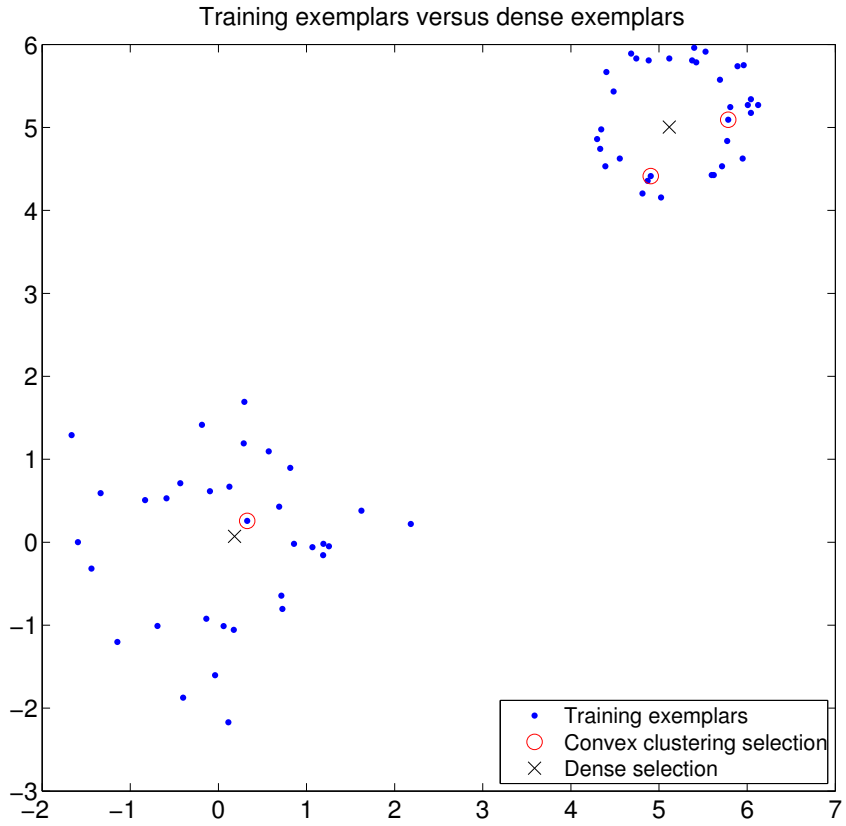

Figure: Exemplar selection within the training set versus the finest dense set of 900 exemplars on a regular grid.

Motivational Experiment (cont'd)

Using a finer and finer discretization improves log-likelihood.

- using the training set is not enough

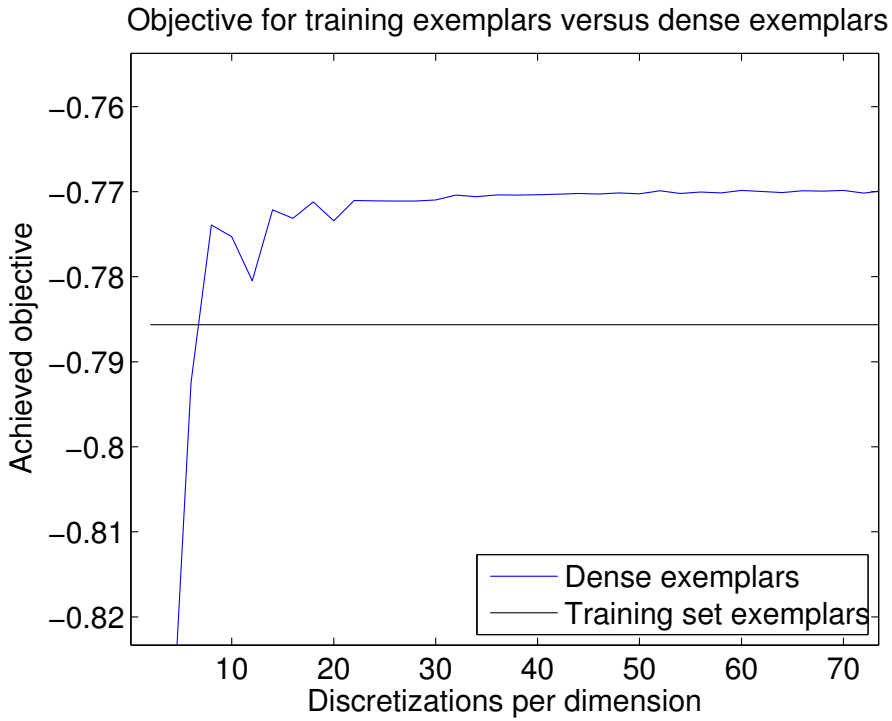

Figure: Training set vs. dense set log-likelihood.

Infinite Exemplars

Taking the idea of "more exemplars" to the limit and generalizing KVQ and CC, we propose the following general model:

$$\min_{\mathbf{q},\gamma,\rho} \quad \Omega(\gamma,\rho) \tag{2}$$

$$\text{sb.t.} \quad \int_{\mathcal{Z}} q_{\mathbf{z}} k_{\mathbf{z}}(\mathbf{x}_i)\, d\mathbf{z} = \gamma_i : \alpha_i,\ i = 1,\ldots,N \tag{3}$$

$$\rho \leq \gamma_i : \omega_i, \quad i = 1,\ldots,N, \tag{4}$$

$$q_{\mathbf{z}} \geq 0 : \mu_{\mathbf{z}}, \quad \forall \mathbf{z} \in \mathcal{Z}, \tag{5}$$

$$\int_{\mathcal{Z}} q_{\mathbf{z}}\, d\mathbf{z} = 1 : \sigma, \tag{6}$$

where $\boldsymbol{\alpha}$, $\boldsymbol{\omega}$, $\boldsymbol{\mu}$ and σ are the Lagrange multipliers.

- Constraint (3) evaluates a convex combination of responses for each sample. γ_i contains the combined response for sample $\mathbf{x}_i$.
- Constraint (4) identifies – if $\nabla_\rho \Omega(\gamma,\rho) < 0$ – the lowest response among all samples. The value of the lowest combined response is ρ.
- Constraints (5) and (6) define the combination simplex.

Smoothing Kernel

$k_{\mathbf{z}}(\mathbf{x}) = g\left(\left\|\frac{\mathbf{z}-\mathbf{x}}{h}\right\|\right)$, where $g : \mathbb{R} \to \mathbb{R}^+$ is one of

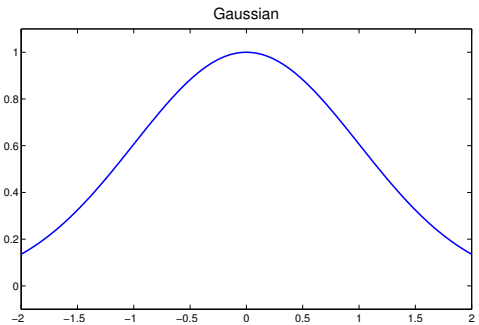

Figure: Gaussian smoothing kernel.

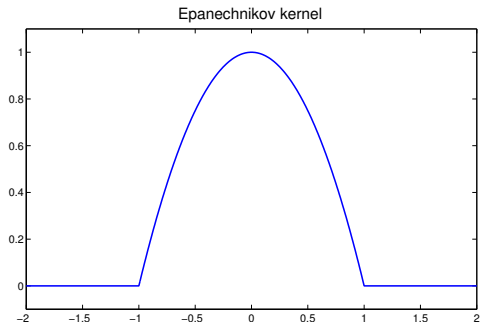

Figure: Epanechnikov smoothing kernel (AMISE optimal).

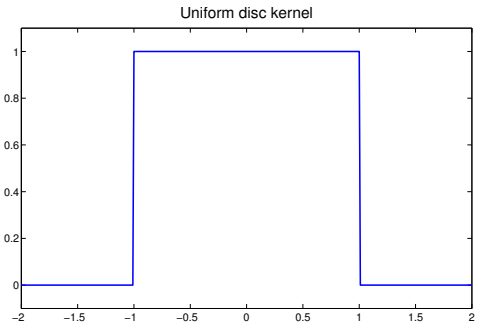

Figure: Uniform "disc" kernel.

Objectives

Some possible objective functions (we only use the first two)

1. $\Omega(\boldsymbol{\gamma}, \rho) = -\rho$
   The objective maximizes the lowest response among all samples. KVQ corresponds to this objective with $k_{\mathbf{z}}(\cdot)$ chosen as discussed earlier.
2. $\Omega(\boldsymbol{\gamma}, \rho) = -\frac{1}{N}\sum_{i=1}^{N} \log(\gamma_i)$
   This objective maximizes $\prod_{i=1}^{N} \gamma_i$. For the special case where the columns of K correspond to evaluations of probability density functions at the training samples this objective maximizes the log-likelihood of the samples under a mixture model.
3. $\Omega(\boldsymbol{\gamma}, \rho) = -\rho + \frac{C}{N}\sum_{i=1}^{N} (\gamma_i - \rho)^2$
   Similar to Margin-Minus-Variance (MMV) objective [6].
4. $\Omega(\boldsymbol{\gamma}, \rho) = -\frac{1}{N}\sum_{i=1}^{N} \gamma_i + \frac{C}{N}\sum_{i=1}^{N} (\gamma_i - \frac{1}{N}\sum_{i=1}^{N} \gamma_i)^2$
   Maximize *mean-minus-variance* popular in portfolio optimization [3].

Modeling Perspective

Change in model

- Original terms (finite): $q_j e^{-\beta d_\phi(\cdot, \mathbf{x}_j)}$
- New terms (infinite): $q_{\mathbf{z}} k_{\mathbf{z}}(\cdot)$

Results

- Known as "*complicated variables*" in operations research used to simplify problems by moving work into subproblems
- Breaks symmetry of the original solution set
- Related to decomposition methods (*Dantzig-Wolfe decomposition*)

Algorithm and Other Details

The optimization problem has infinitely many variables and is solved by *column generation*: start with an empty set of cluster/codebook candidates, then iteratively,

- we either establish global convergence, or
- we add one or more candidate exemplars.

Selecting the candidates to add in each iteration becomes a *subproblem*.

$$\mathcal{L}(\mathbf{q}, \gamma, \rho, \boldsymbol{\alpha}, \boldsymbol{\omega}, \boldsymbol{\mu}, \sigma) = \Omega(\gamma, \rho) + \sum_{i=1}^{N} \alpha_i \left(\int_{\mathcal{Z}} q_{\mathbf{z}} k_{\mathbf{z}}(\mathbf{x}_i)\, d\mathbf{z} - \gamma_i \right) + \boldsymbol{\omega}^\top (\rho \mathbf{1} - \boldsymbol{\gamma}) - \int_{\mathcal{Z}} \mu_{\mathbf{z}} q_{\mathbf{z}}\, d\mathbf{z} + \sigma \left(\int_{\mathcal{Z}} q_{\mathbf{z}}\, d\mathbf{z} - 1 \right)$$

$$\frac{\partial \mathcal{L}}{\partial q_{\mathbf{z}}} = \sum_{i=1}^{N} \alpha_i k_{\mathbf{z}}(\mathbf{x}_i) - \mu_{\mathbf{z}} + \sigma$$

Subproblem

Problem (Subproblem (SP))

Given a set of samples $\mathbf{x}_i \in \mathcal{X}$, $i = 1, \ldots, N$, *a corresponding non-positive sample weighting* $\alpha_i \leq 0$, $i = 1, \ldots, N$ *and a non-negative smoothing kernel* $k_{\mathbf{z}}(\mathbf{x}) : \mathcal{Z} \times \mathcal{X} \rightarrow \mathbb{R}_+$, *obtain* $\mathbf{z}^*$ *as the solution of*

$$\mathbf{z}^* = argmax_{\mathbf{z} \in \mathcal{Z}} - \sum_{i=1}^{N} \alpha_i k_{\mathbf{z}}(\mathbf{x}_i).$$

- Simple mode seeking on a weighted kernel density expanded around the sample points
- Does not depend on the objective function ("decoupled approach")

Better than any fixed candidate set

Theorem

Assume that the subproblem (SP) can be solved exactly in each iteration. Then the algorithm solves problem (2) globally to the desired accuracy ϵ.

Theorem

Given $\Omega(\gamma, \rho)$, a set $X = \{\mathbf{x}_i\}_{i=1,\dots,N}$, $\mathbf{x}_i \in \mathcal{X}$ and a finite set of exemplars $Z_F = \{\mathbf{z}_j\}_{j=1,\dots,M}$, the solution obtained by solving problem (2) with $\mathcal{Z} = Z_F$ can not achieve a better objective than the solution obtained by our algorithm with $\mathcal{Z} = \mathcal{X}$, $Z_0 = Z_F$.

(Proof in the paper.)

Nice result

- better or equal performance than any a-priori fixed candidate set.
- can be started from any given candidate set.

Example, GMM

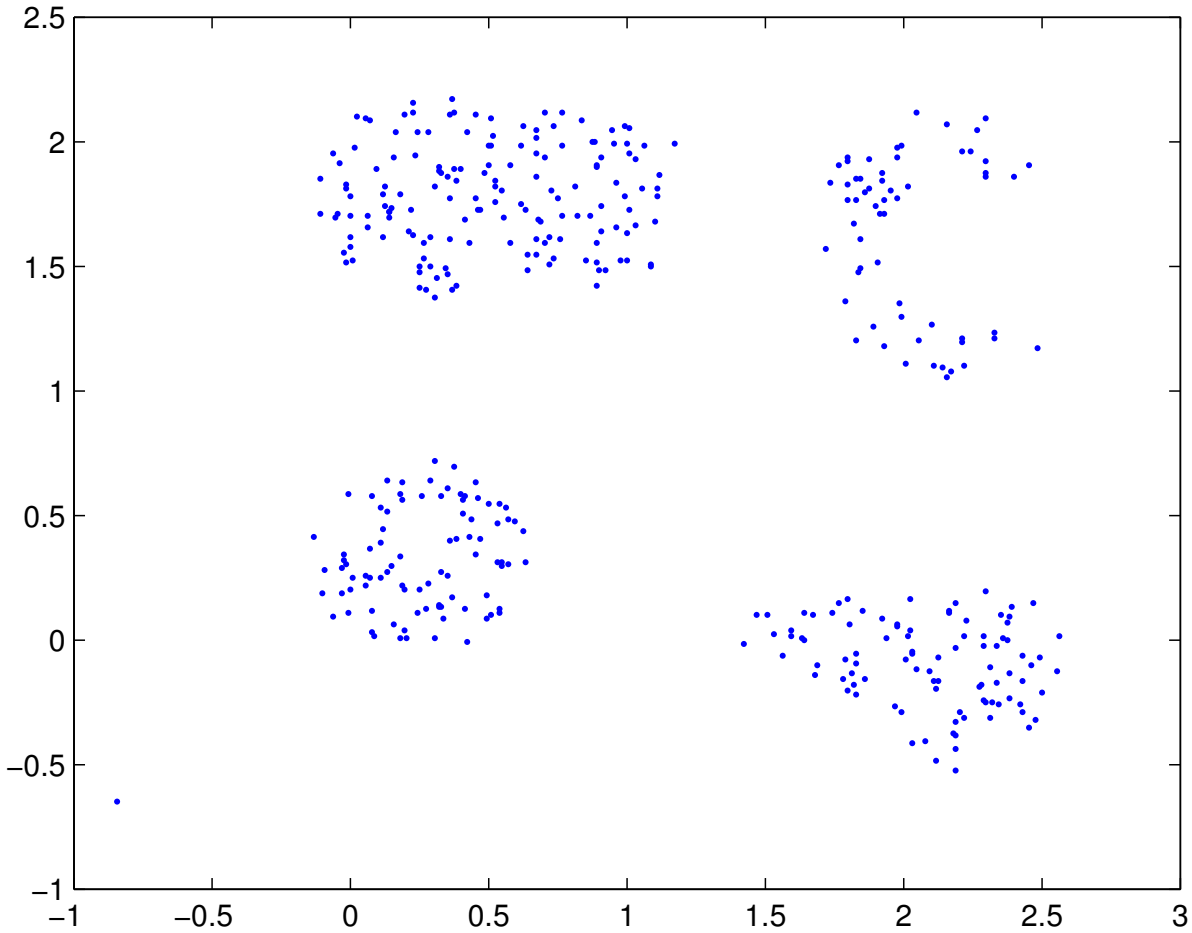

Figure: Iteration 0: Samples

Example, GMM

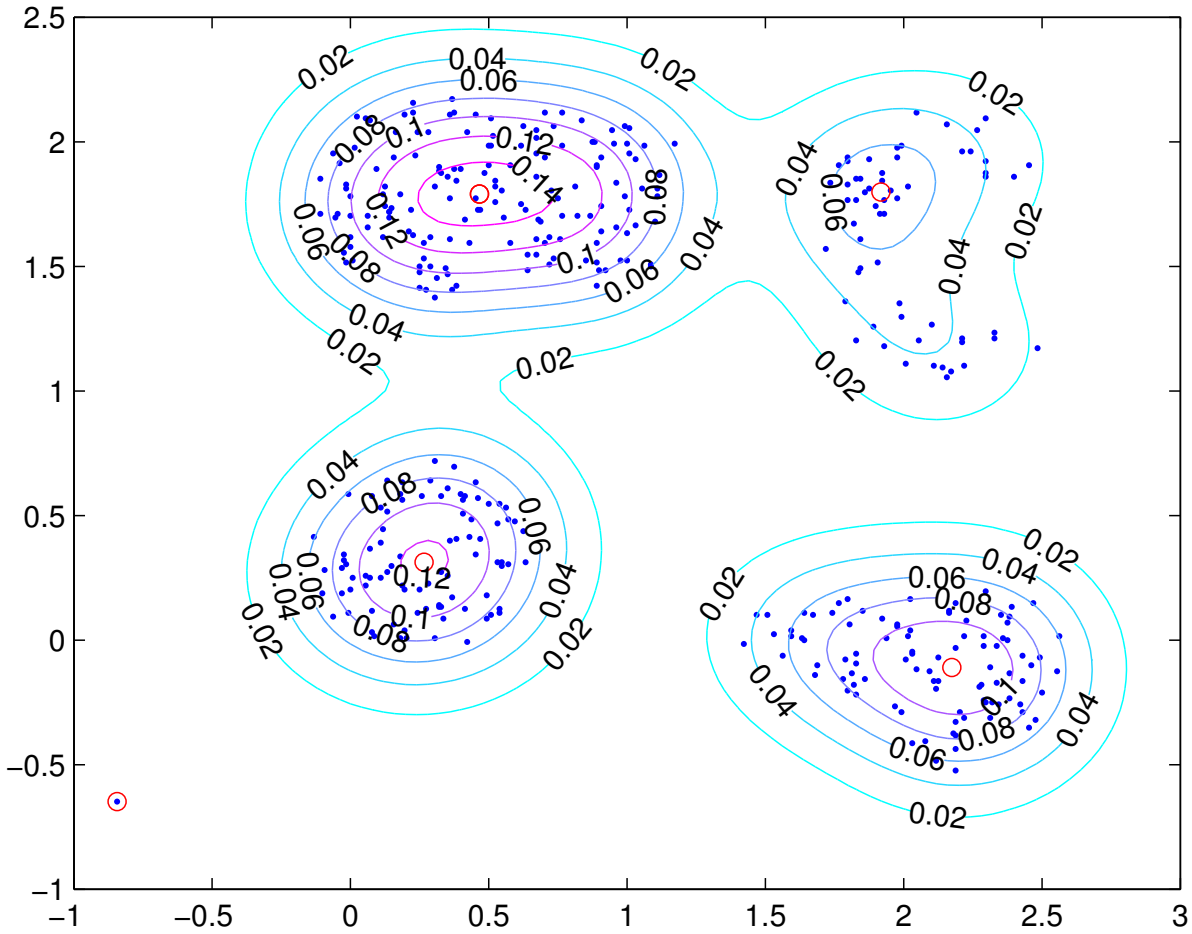

Figure: Iteration 1: Parzen modes (uniform weights α)

Example, GMM

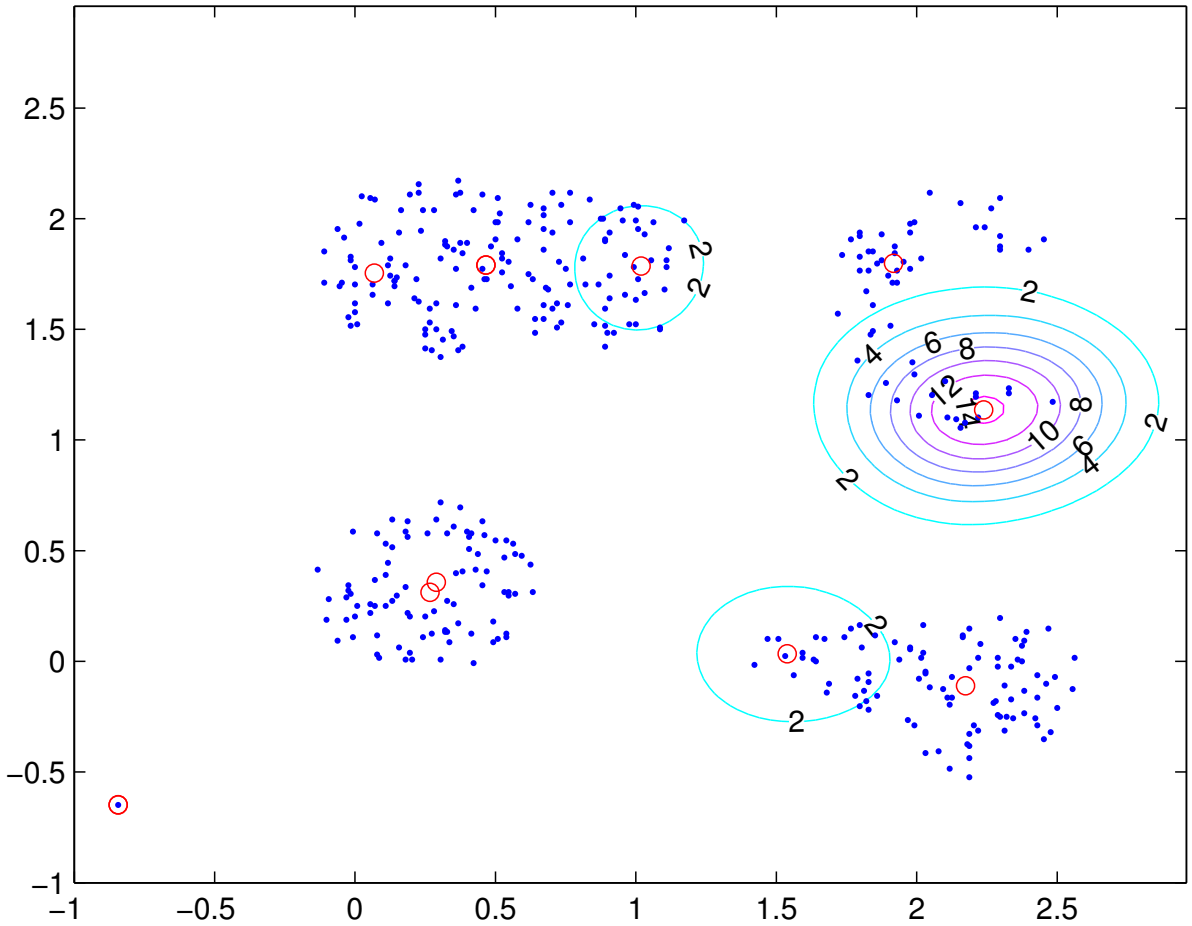

Figure: Iteration 2: Reweighted modes

Example, GMM

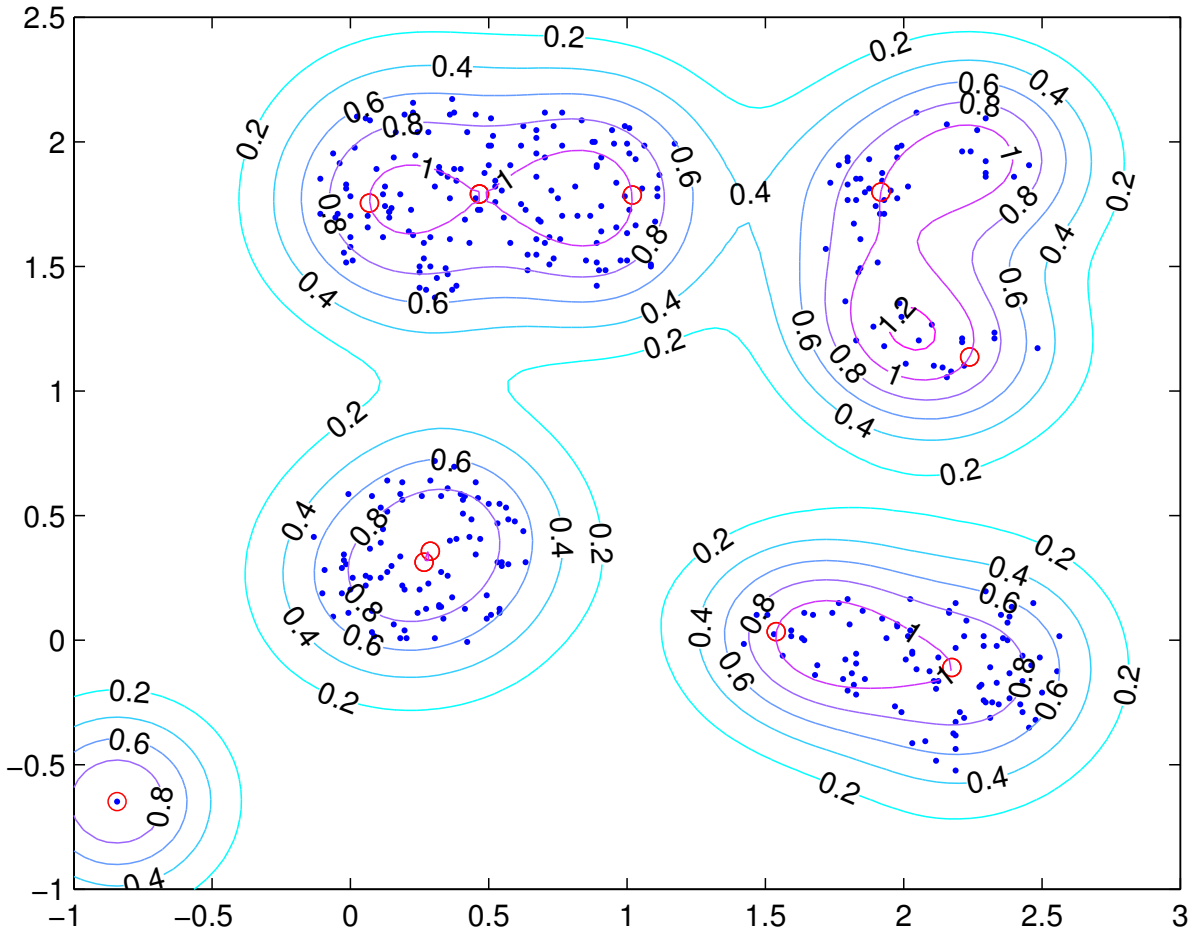

Figure: Iteration 2 result

Example, GMM

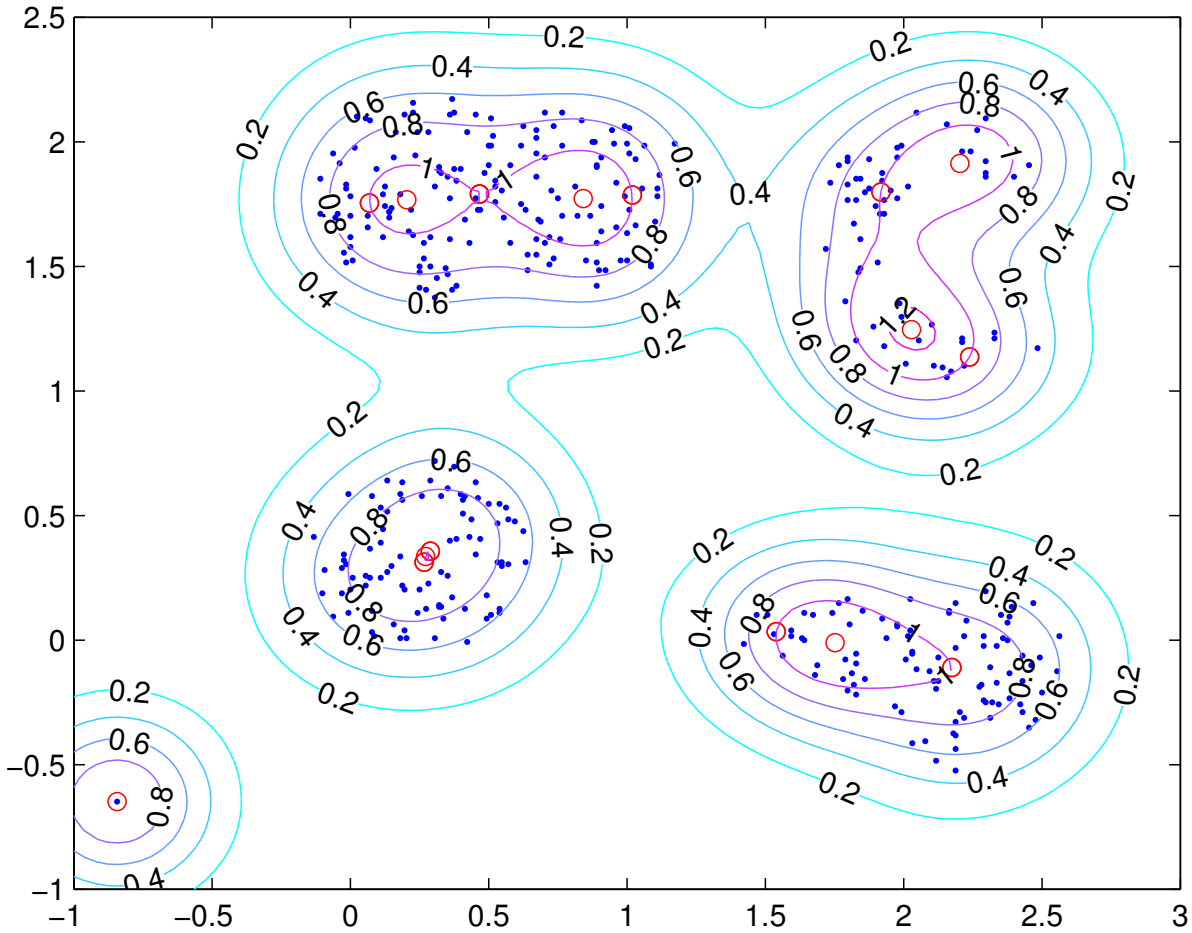

Figure: Iteration 3

Example, GMM

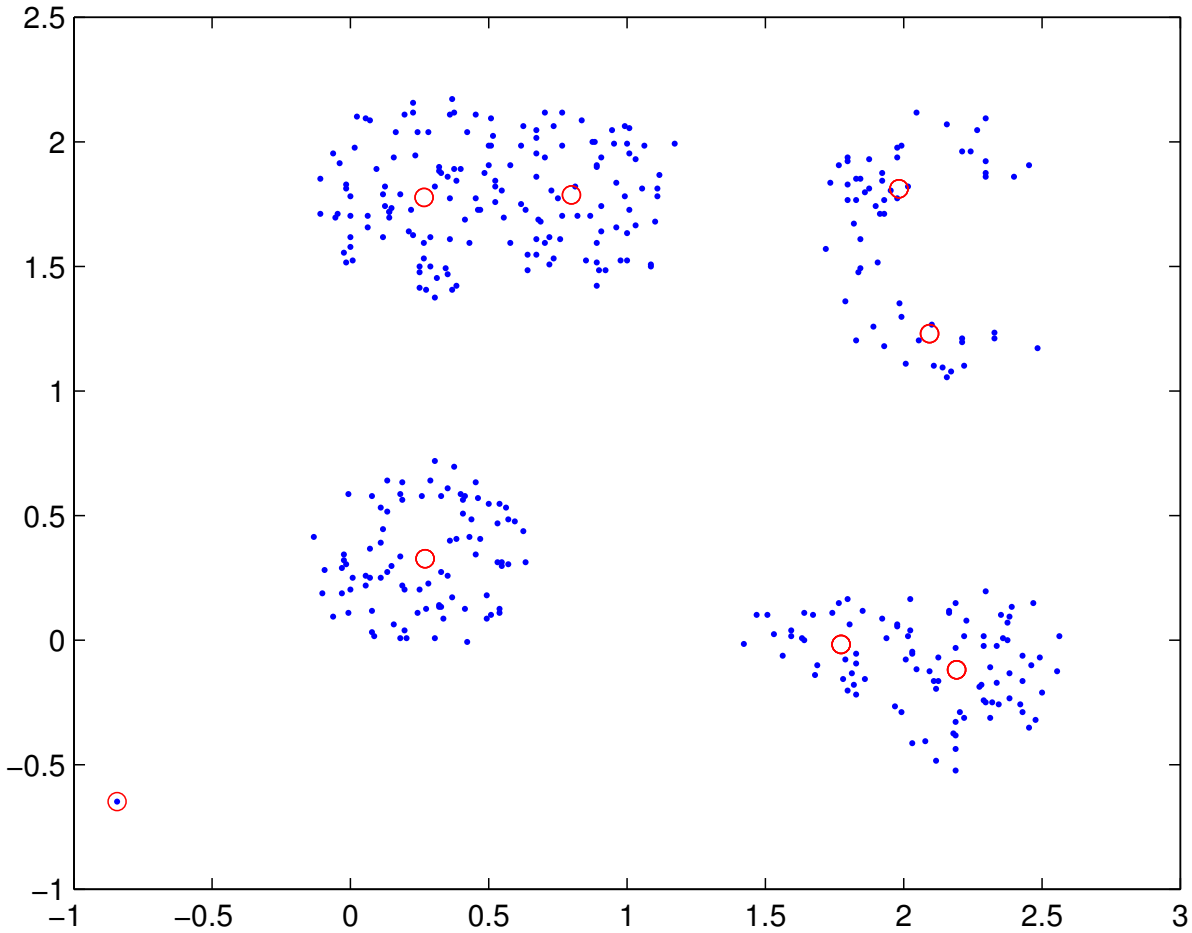

Figure: Final iteration

Experiment: GMM

We consider mixture model density estimation and compare our method with Convex Clustering and a homoscedastic Gaussian mixture ($\Sigma = \sigma^2 I$) learned with Expectation Maximization (EM). The dataset is a 1100 sample subset of USPS, and the log-likelihood objective is used.

σ	CC	INFEX	EM BEST	EM MEAN
440	−6.3356	**-5.1370**	−5.1442	−5.1485
460	−6.1269	**-4.7424**	−4.7486	−4.7503
480	−5.8705	**-4.3796**	−4.3823	−4.3834
500	−5.5813	**-4.0499**	−4.0507	−4.0520
520	−5.2780	**-3.7499**	−3.7502	−3.7512
540	−4.9779	**-3.4788**	−3.4789	−3.4795

Table: Achieved log-likelihoods. CC is Convex Clustering; for EM the best and mean of 20 runs are shown.

Summary: EM does well, CC is worst, we are best.

Experiment: vs KVQ

USPS subset (1100 samples, 110 per class), vary h

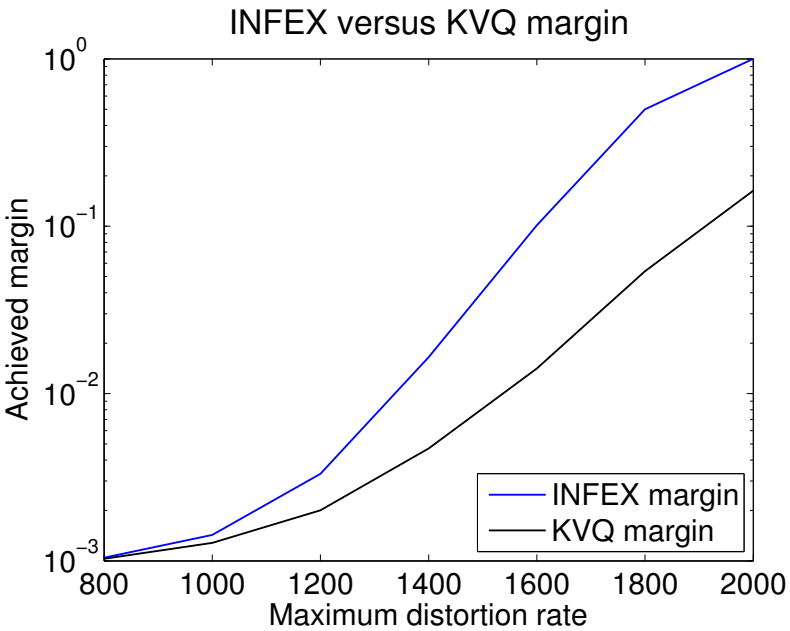

Figure: Optimal margin ρ^* as a function of the maximum allowed distortion. Note the log-scale.

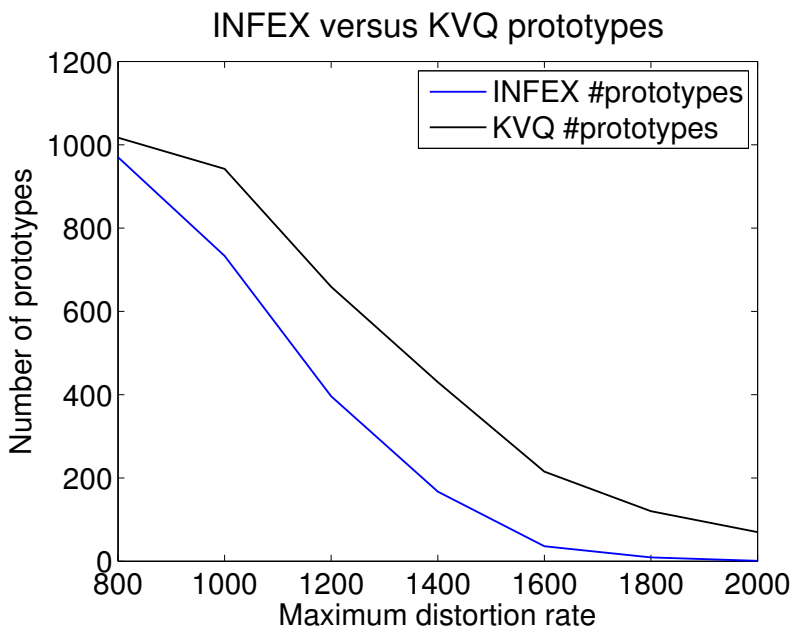

Figure: The number of selected prototypes as a function of the maximum allowed distortion.

Conclusions

Summary

- Unifying perspective on existing exemplar based methods for density estimation, clustering, and vector quantization
- Convex master problem, non-convex subproblem. Provably better than any proposed convex approach.

Open questions

- Does there exists a response function k that is useful for unsupervised learning and at the same time yields a globally solvable subproblem?
- What is the relation between objective Ω, kernel k and number of components $\|\mathbf{q}^*\|_0$?
- Can a decomposition similar to ours yield a training scheme for supervised learning of RBF networks in the line of [2]?

References


A. Banerjee, S. Merugu, I. S. Dhillon, and J. Ghosh.
Clustering with bregman divergences.
Journal of Machine Learning Research, 6:1705–1749, 2005.


Y. Bengio, N. L. Roux, P. Vincent, O. Delalleau, and P. Marcotte.
Convex neural networks.
In *NIPS*, 2005.


G. Cornuejols and R. Tütüncü.
Optimization methods in finance.
Mathematics, Finance and Risk. Cambridge University Press, 2007.


D. Lashkari and P. Golland.
Convex clustering with exemplar-based models.
In *NIPS*, 2007.


S. Rosset and E. Segal.
Boosting density estimation.
In *NIPS*, pages 641–648. MIT Press, 2002.


U. Rückert and S. Kramer.
A statistical approach to rule learning.
In W. W. Cohen and A. Moore, editors, *ICML*, volume 148 of *ACM International Conference Proceeding Series*, pages 785–792, 2006.


M. Tipping and B. Schölkopf.
A kernel approach for vector quantization with guaranteed distortion bounds.
In *AISTATS*, 2001.

Disc kernel lower bound

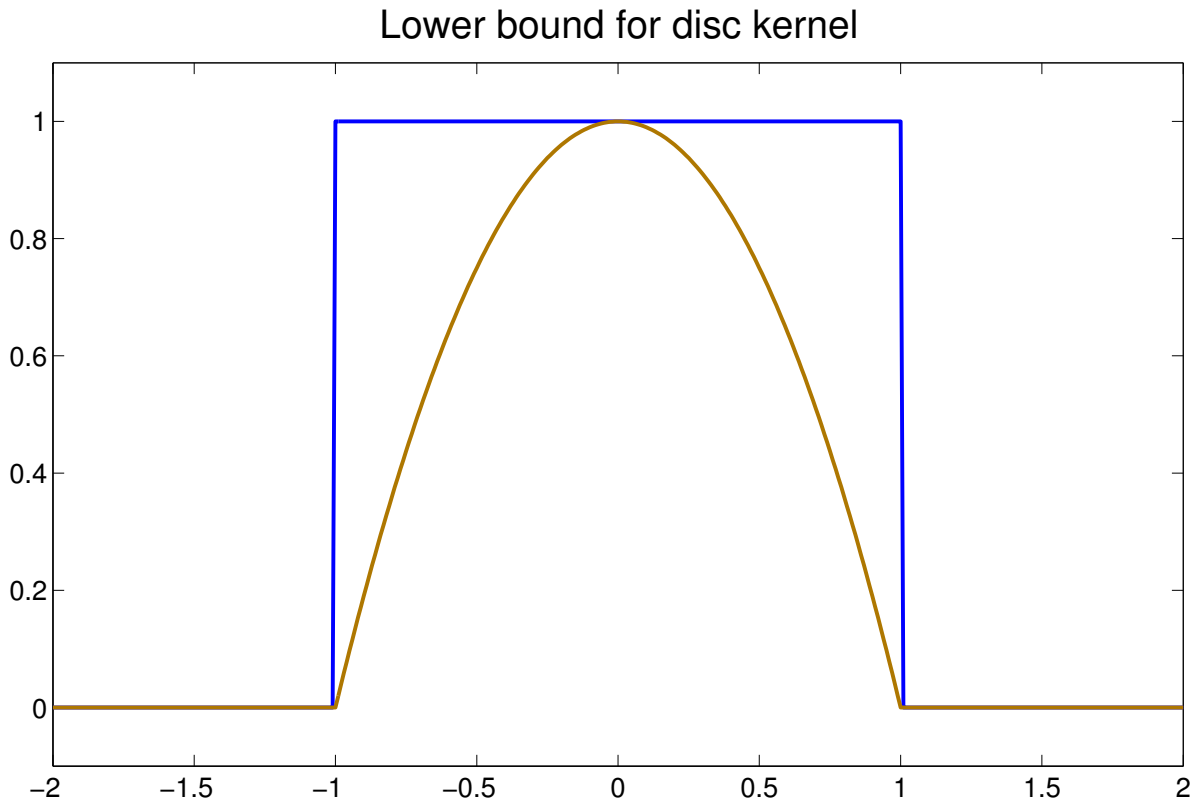

Figure: Epanechnikov kernel lower bounds the disc kernel.

Dual

$$
\begin{aligned}
\max_{\boldsymbol{\alpha},\sigma,\boldsymbol{\omega},\boldsymbol{\mu}} \quad & -\Omega^*(\boldsymbol{\alpha},\sigma,\boldsymbol{\omega},\boldsymbol{\mu}) - \sigma \\
\text{sb.t.} \quad & (\boldsymbol{\alpha},\sigma,\boldsymbol{\omega},\boldsymbol{\mu}) \in \mathbf{dom}(\Omega^*), \\
& \sum_{i=1}^{N} \alpha_i k_{\mathbf{z}}(\mathbf{x}_i) \geq \mu_{\mathbf{z}} - \sigma, \quad \forall \mathbf{z} \in \mathcal{Z} \\
& \boldsymbol{\omega} \geq \mathbf{0} \\
& \mu_{\mathbf{z}} \geq \mathbf{0}, \quad \forall \mathbf{z} \in \mathcal{Z}
\end{aligned}
$$

- $\Omega(\boldsymbol{\gamma}, \rho) = -\rho$
  Conjugate $\Omega^*(\boldsymbol{\alpha},\sigma,\boldsymbol{\omega},\boldsymbol{\mu}) = 0$ and domain $\mathbf{dom}(\Omega^*) = \{(\boldsymbol{\alpha},\sigma,\boldsymbol{\omega},\boldsymbol{\mu}) : \boldsymbol{\omega} + \boldsymbol{\alpha} \leq \mathbf{0}, \boldsymbol{\omega}^\top \mathbf{1} = 1\}$.
- $\Omega(\boldsymbol{\gamma}, \rho) = -\frac{1}{N}\sum_{i=1}^{N} \log(\gamma_i)$
  Conjugate $\Omega^*(\boldsymbol{\alpha},\sigma,\boldsymbol{\omega},\boldsymbol{\mu}) = -\frac{1}{N}\sum_{i=1}^{N} \log(-\alpha_i) + \log(N)$ with domain $\mathbf{dom}(\Omega^*) = \{(\boldsymbol{\alpha},\sigma,\boldsymbol{\omega},\boldsymbol{\mu}) : \boldsymbol{\alpha} < \mathbf{0}, \boldsymbol{\omega} = \mathbf{0}\}$.