Deep Learning (BEV033DLE)
Lecture 11
KL Divergence, t-SNE, Unsupervised RL

Czech Technical University in Prague

- Stochastic Neighbor Embedding (t-SNE)
- KL Divergence
- Unsupervised Representation Learning
  - Latent Variable Models
  - EM
  - ELBO, Variational Inference

Stochastic Neighbor Embedding

Motivation

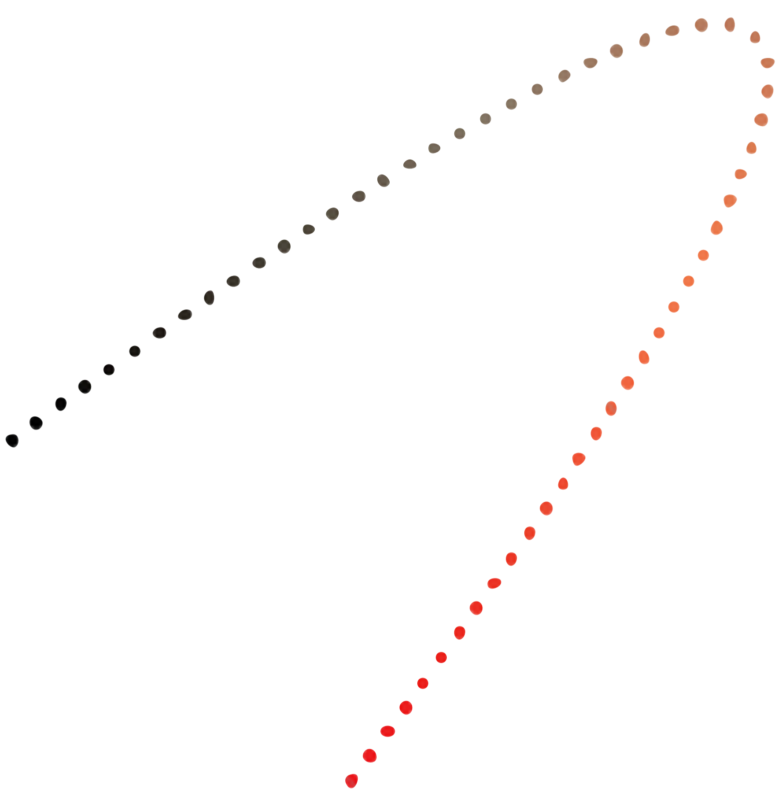
✦ Tool of representational geometry:

- dimensionality reduction for data visualization

✦ Goals:

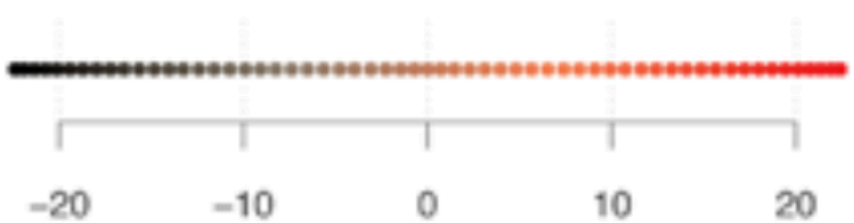
- Data often lies on a lower-dimensional manifold
- Preserve small distances accurately
- Large distances can be increased more

Data in $\mathbb{R}^n$

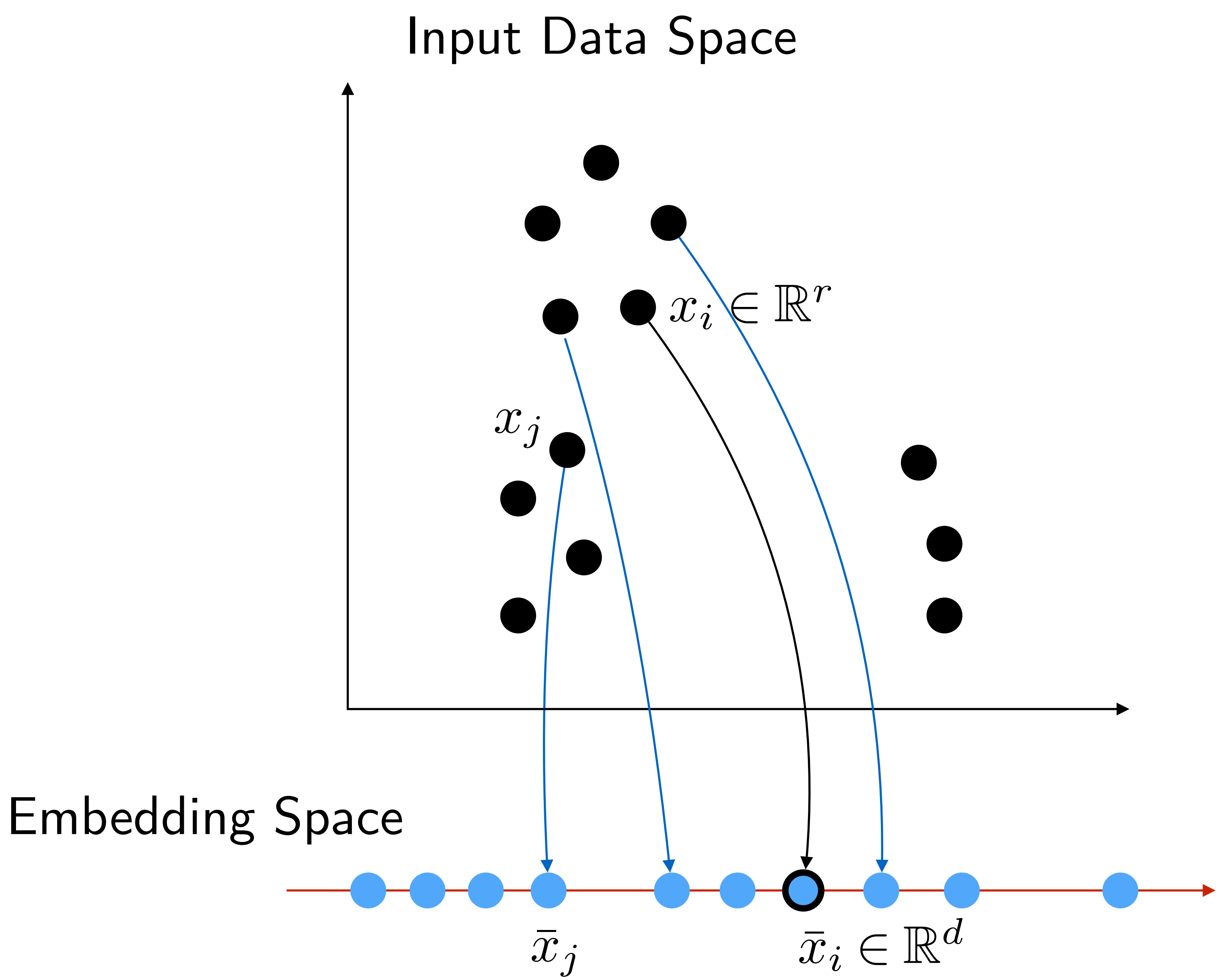


Non-linear embedding

Representation in $\mathbb{R}^d$

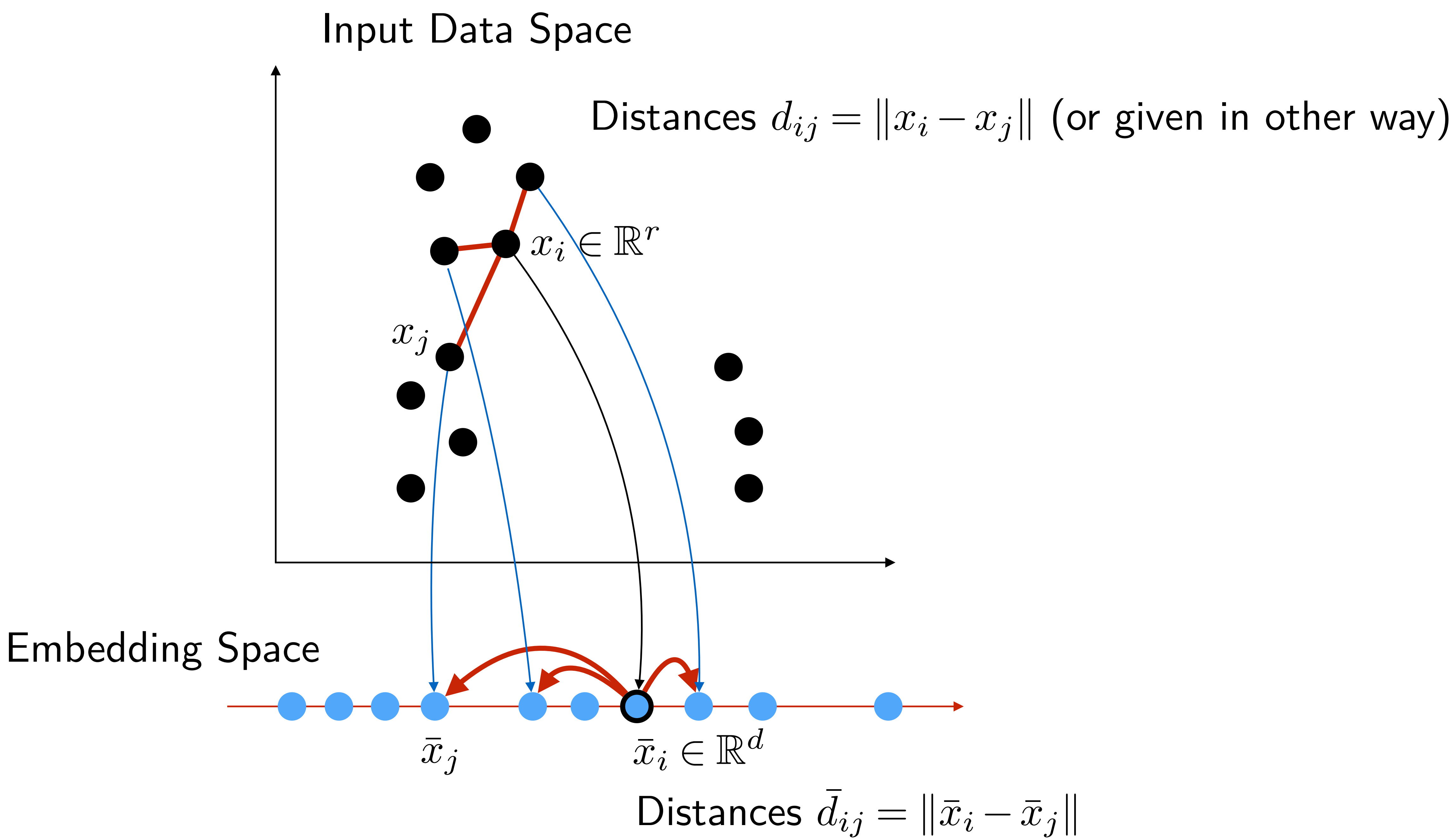


Multidimensional Scaling (MSD)

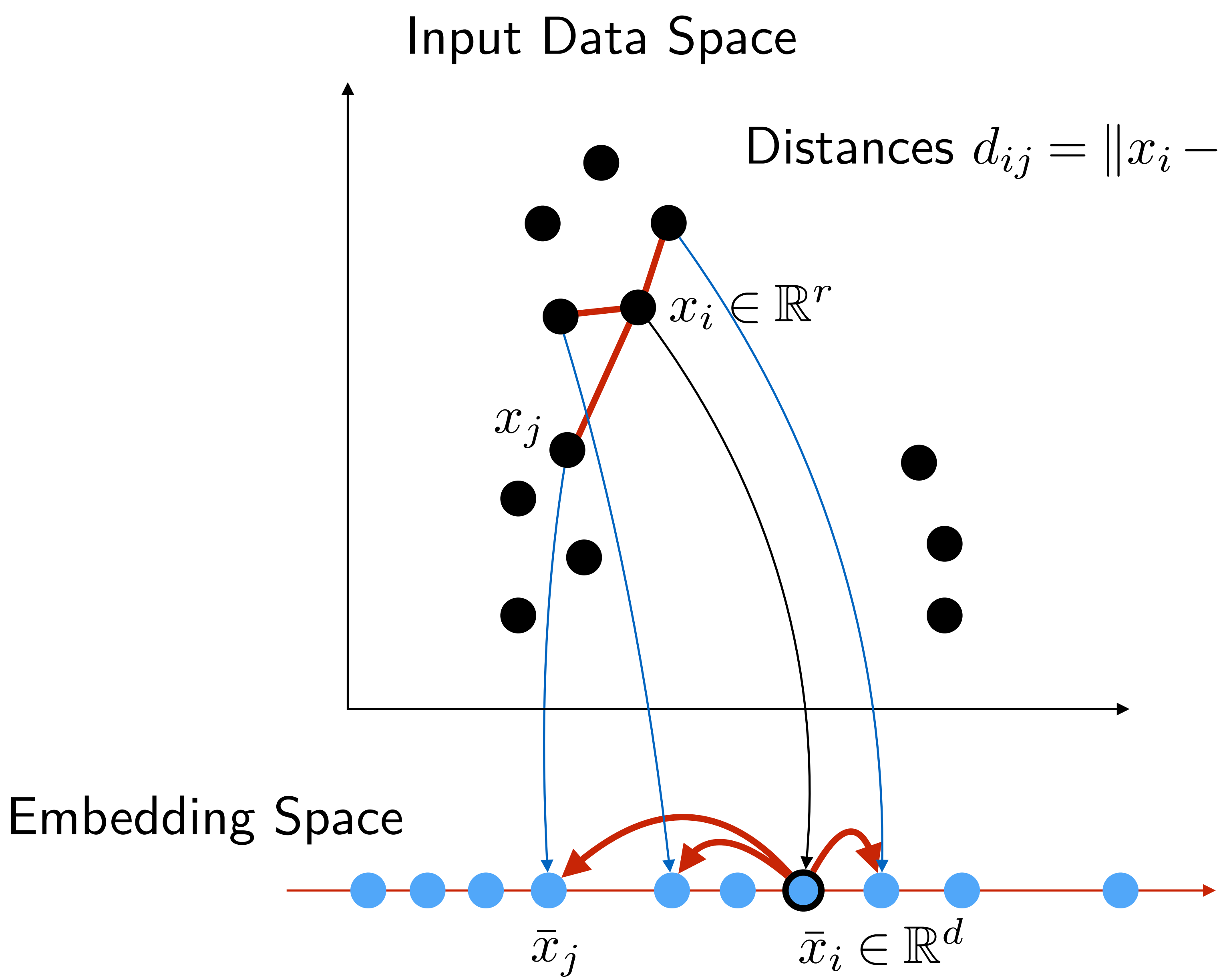


- Non-parametric model: for each data point x_t we find a corresponding embedding $\bar{x}_t$

Multidimensional Scaling (MSD)



Multidimensional Scaling (MSD)



Distances $d_{ij} = \|x_i - x_j\|$ (or given in other way)

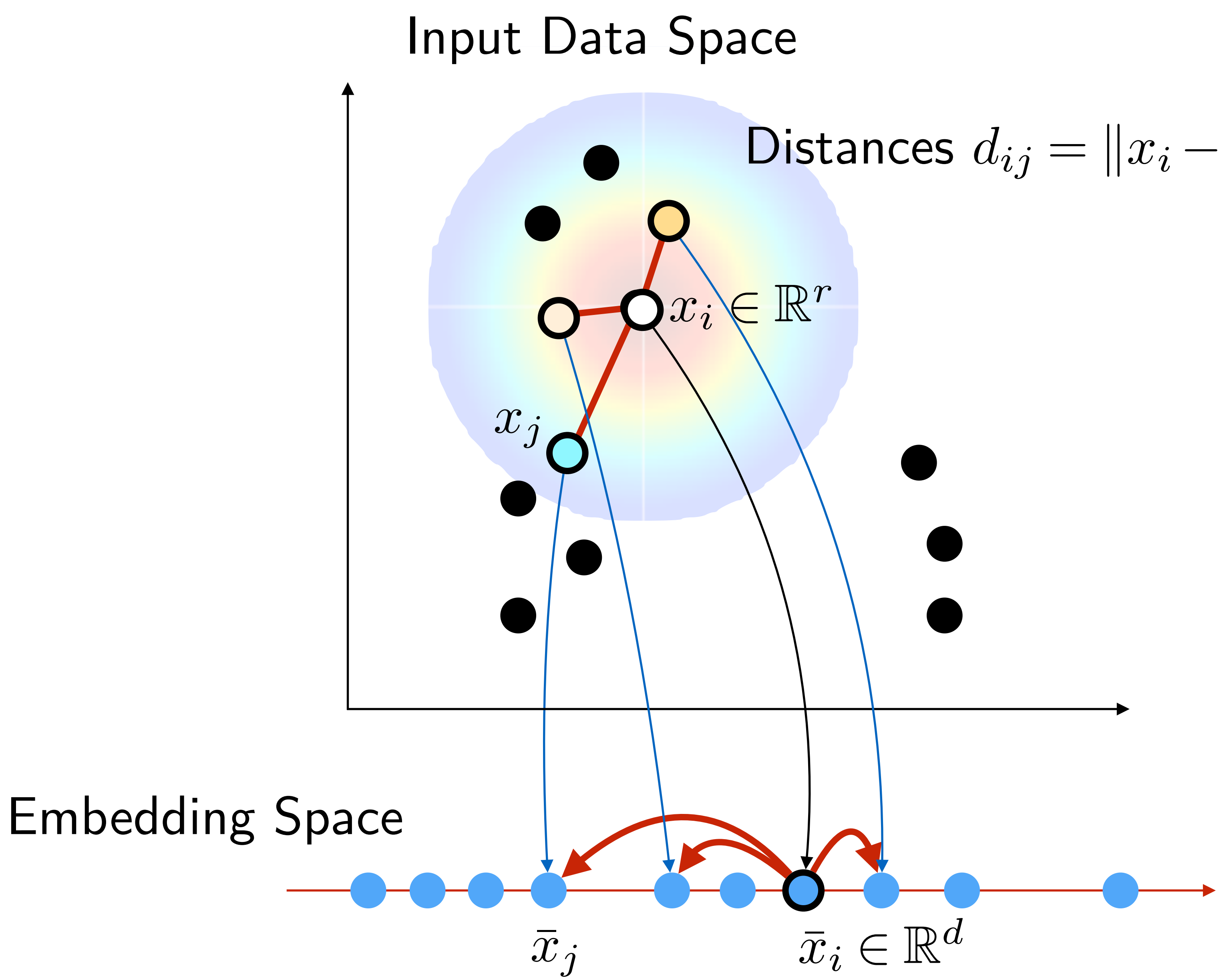
$$\min_{\bar{x}} \sum_{i \neq j} (d_{ij} - \bar{d}_{ij})^2$$

Want to preserve all distances

-- too stringent

Distances $\bar{d}_{ij} = \|\bar{x}_i - \bar{x}_j\|$

Stochastic Neighbor Embedding (SNE)



Distances $d_{ij} = \|x_i - x_j\|$ (or given in other way)

$$p_i(j) = \frac{e^{-d_{ij}^2/2\sigma_i^2}}{\sum_{j' \neq i} e^{-d_{ij'}^2/2\sigma_i^2}}, \quad \forall j \neq i$$

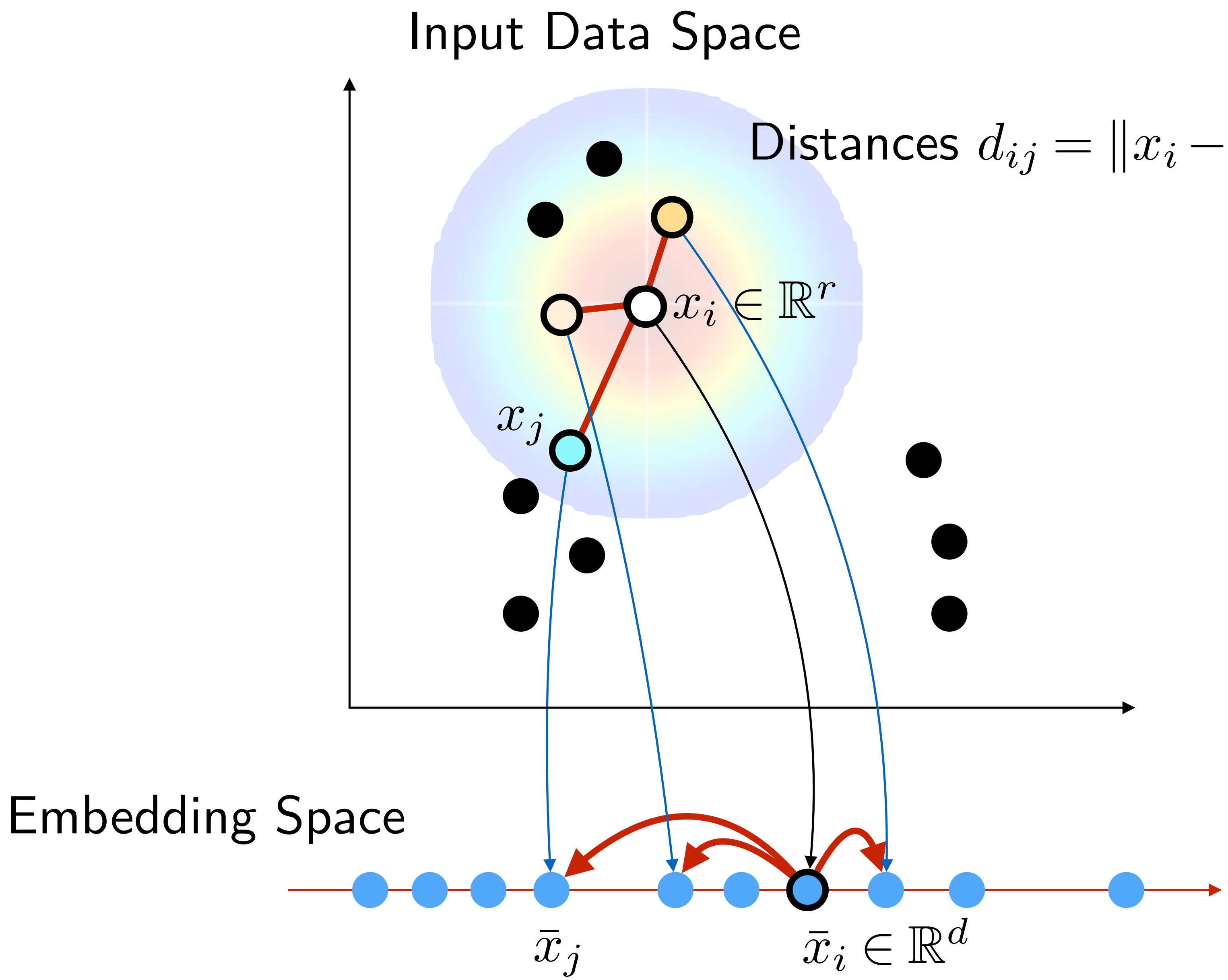
$$\min_{\bar{x}} \sum_i \mathrm{KL}(p_i \,\|\, q_i)$$

$$q_i(j) = \frac{e^{-\bar{d}_{ij}^2}}{\sum_{j \neq i} e^{-\bar{d}_{ij}^2}}, \quad \forall j \neq i$$

Distances $\bar{d}_{ij} = \|\bar{x}_i - \bar{x}_j\|$

- $\operatorname{argmin}_{\bar{x}} \sum_i \mathrm{KL}(p_i \,\|\, q_i) = \operatorname{argmax}_{\bar{x}} \sum_i \sum_j p_i(j) \log q_i(j)$
- Maximum likelihood learning to predict the “nearest neighbor” by q
- In comparison to MDS: normalization, distant neighbors are down-weighted
- In comparison to “Contrastive Learning”: distribution $p_i(j)$ instead of a known “positive”

t-Distributed SNE (t-SNE)



Distances $d_{ij} = \|x_i - x_j\|$ (or given in other way)

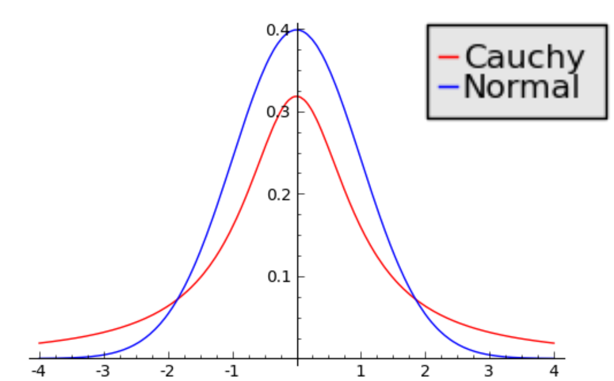
$$p(j|i) \propto e^{-d_{ij}^2/2\sigma^2}, \quad \forall j \neq i$$

$$\min_{\bar{x}} \sum_i \mathrm{KL}\big(p(\cdot|i) \,\|\, q(\cdot|i)\big)$$

$$q(j|i) \propto \left(1 + \bar{d}_{ij}^2\right)^{-1}, \quad \forall j \neq i$$

Distances $\bar{d}_{ij} = \|\bar{x}_i - \bar{x}_j\|$

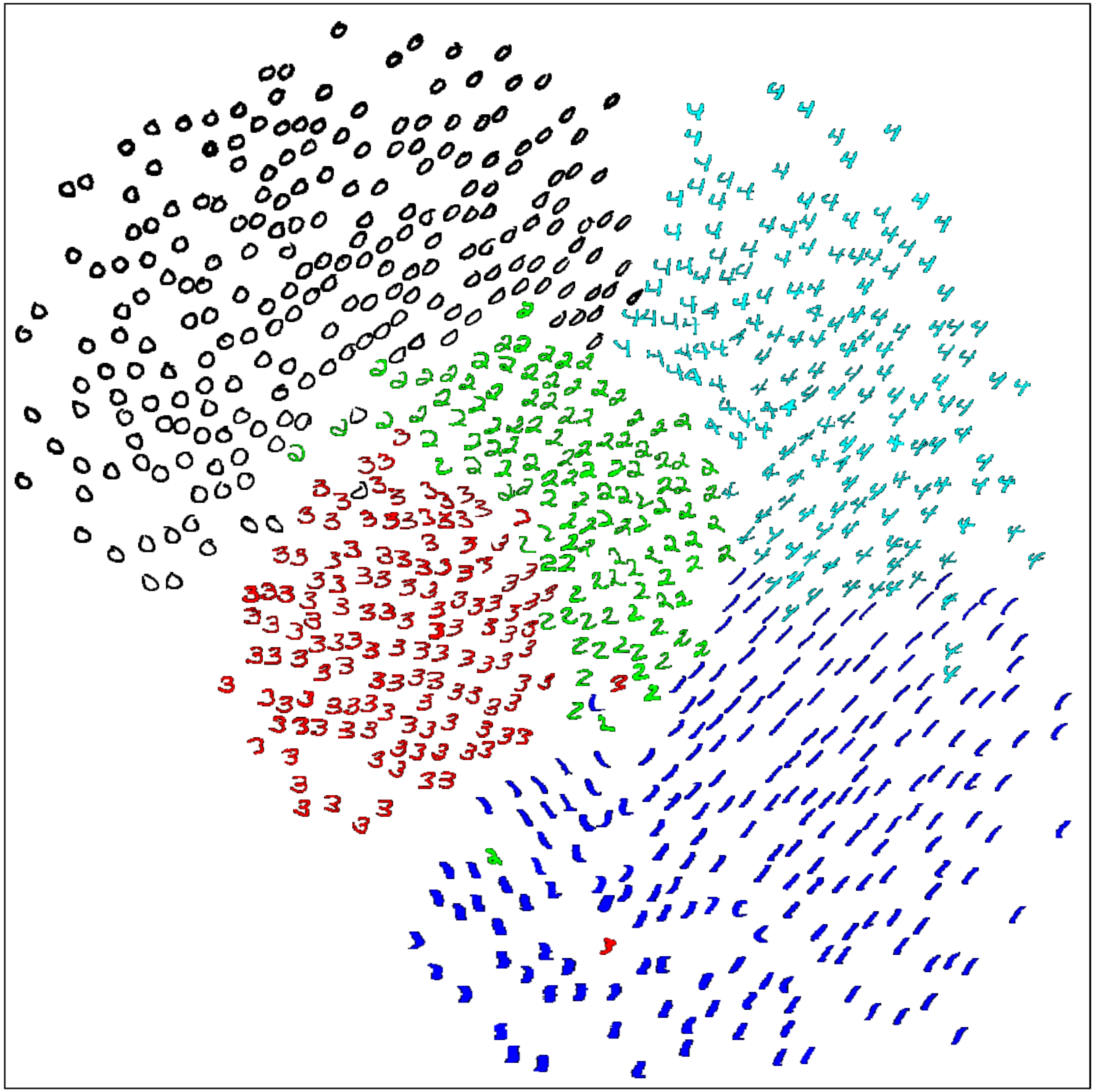
(Student t with 1 degree of freedom is Cauchy)



- Improves clustering of the data (sometimes too much)
- Omitted: symmetrization, initialization, adaptive sigma

[Maaten & Hinton (2008): Visualizing Data using t-SNE]

Examples

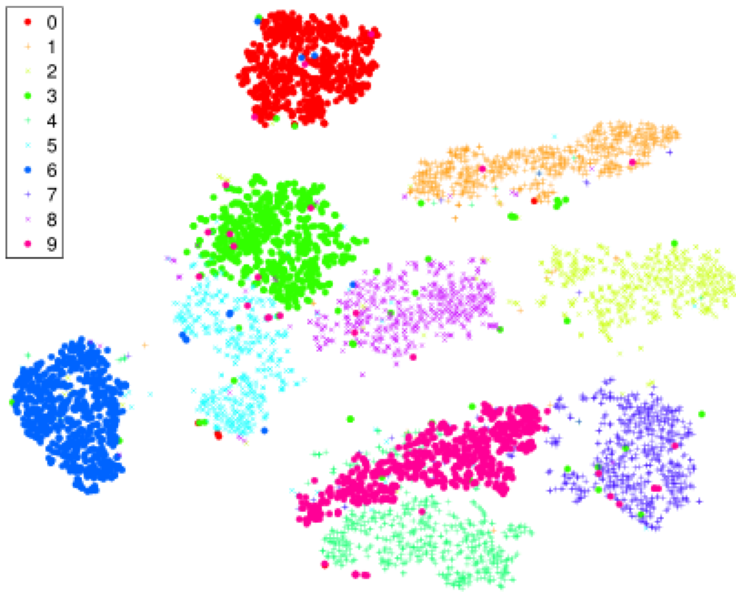


SNE algorithm on 256-dimensional grayscale images of handwritten digits

[Hinton & Roweis (2002): Stochastic Neighbor Embedding]

Examples

MNIST data

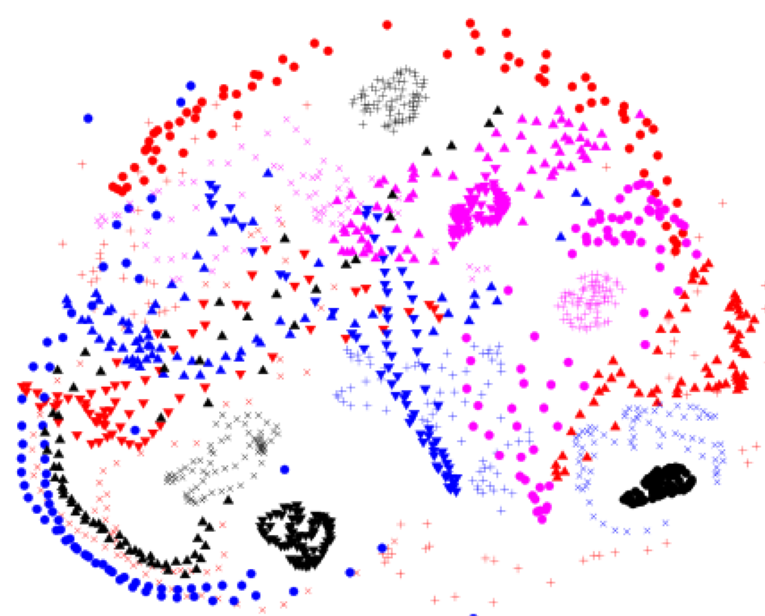


t-SNE

Sammon Mapping: $\mathcal{L} = \sum_{i \neq j} \frac{(d_{ij} - \bar{d}_{ij})^2}{d_{ij}}$

COIL data

t-SNE



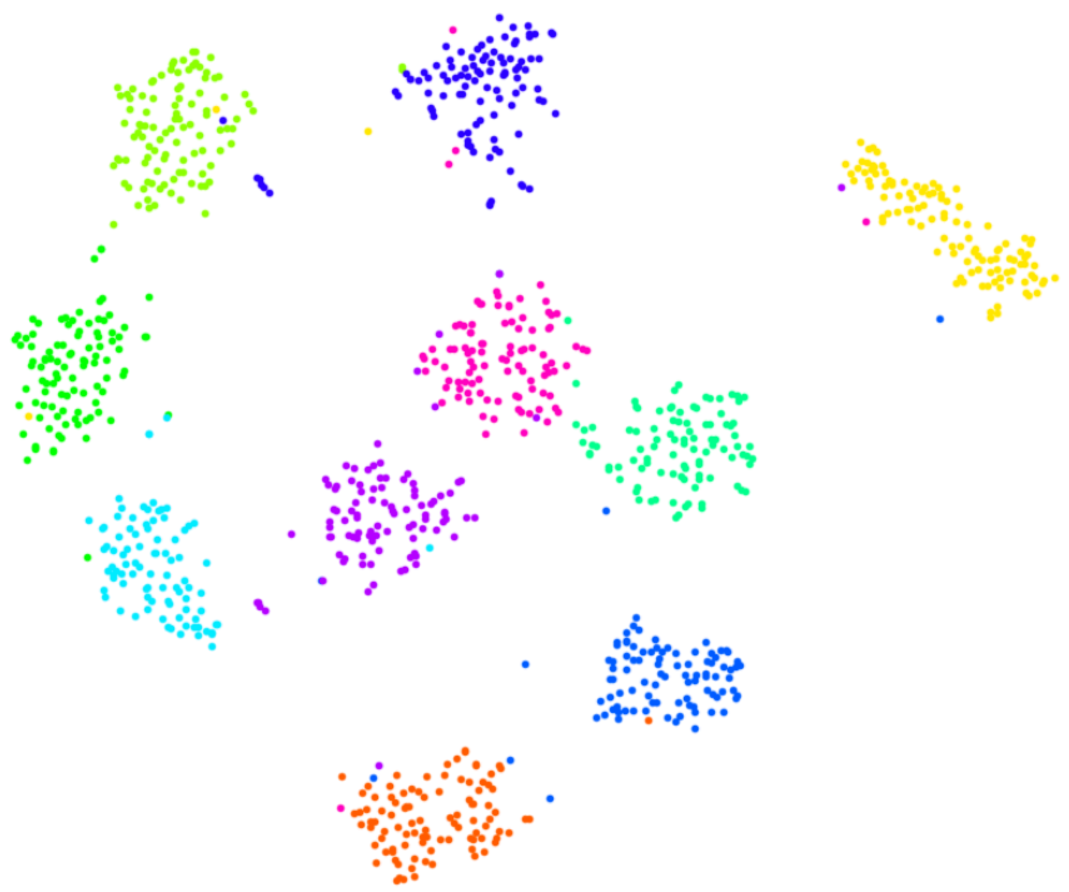
Sammon Mapping

[Maaten & Hinton (2008): Visualizing Data using t-SNE]

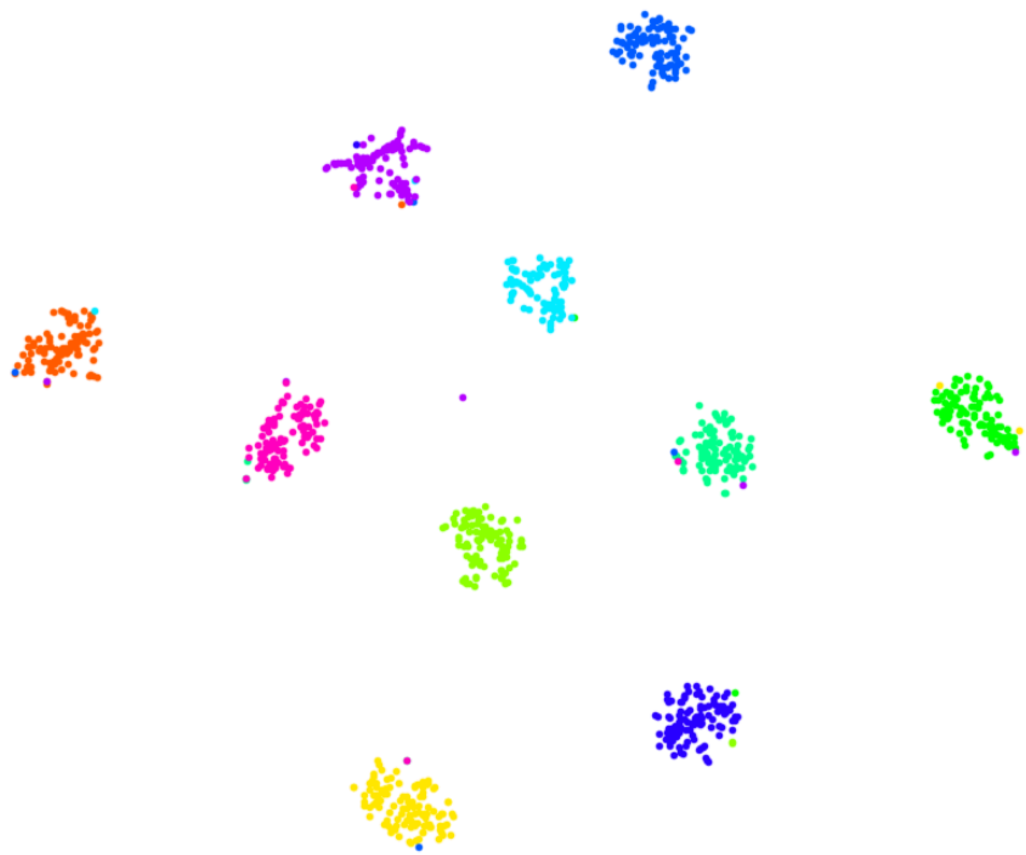
Example: Metric Learning

Embedding using similarity in the future space of the network, colored by class labels

Classifier feature



Triplet-trained feature



KL Divergence

KL Divergence

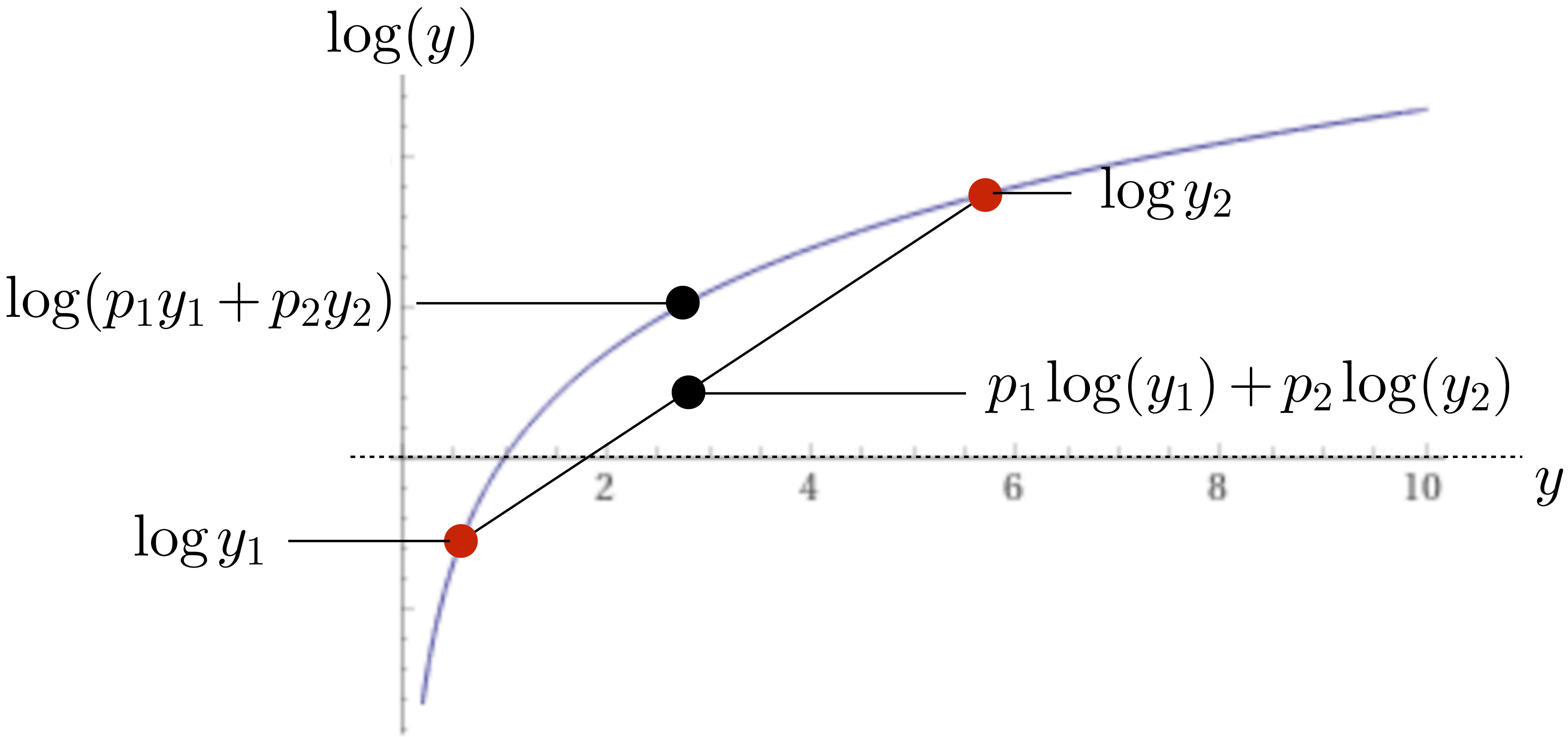
- Let $p(x)$ and $q(x)$ be two probability distributions.
- Kullback–Leibler divergence of p and q is

$$D_{\mathrm{KL}}(p \,\|\, q) = \sum_x p(x) \log \frac{p(x)}{q(x)}$$

  - Definition allows $p(x) = 0$ by the extension $\lim_{p \to 0} p \log p = 0$
  - Defined when $\mathrm{supp(p)} \subseteq \mathrm{supp(q)}$, i.e. $q(x) = 0 \Rightarrow p(x) = 0$
- Properties:
  - D_{KL} is a *divergence*: $D_{\mathrm{KL}} \geq 0$ with equality iff $q = p$
  - Non-symmetric
  - (Invariant under change of variables)
  - Information-theoretic properties (Amount of information lost when q is used to approximate p)

Non-negativity

- Non-negativity: $D_{\mathrm{KL}}(p\|q) \geq 0$
  - let $y(x) = \frac{q(x)}{p(x)}$
  - The inequality $\sum_x p(x) \log \frac{p(x)}{q(x)} \geq 0$ is equivalent to $\sum_x p(x) \log y(x) \leq 0$
  - Observe that $\log$ is concave, apply Jensen's inequality:
  - $\sum_x p(x) \log y(x) \leq \log \sum_x p(x) y(x) = \log \sum_x q(x) = \log 1 = 0.$
- From strict concavity follows that $D_{\mathrm{KL}}(p\|q) = 0$ iff $p = q$



Asymmetry

Minimizing **forward KL** divergence:

$$\min_q D_{\mathrm{KL}}(p\|q)$$

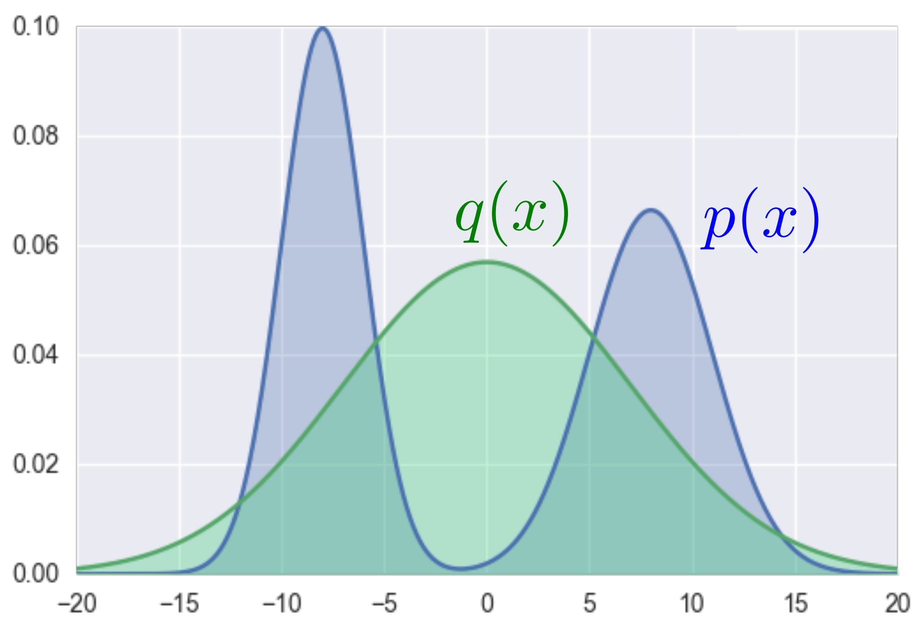
$$\min_q \int p(x)(\log p(x) - \log q(x))dx$$

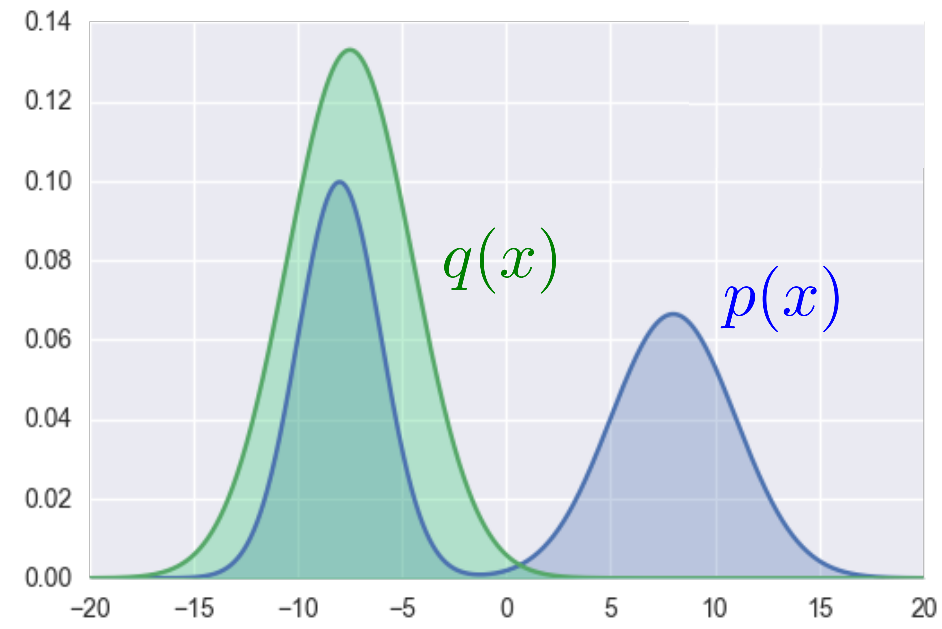
Minimizing **reverse KL** divergence:

$$\min_q D_{\mathrm{KL}}(q\|p)$$

$$\min_q \int q(x)(\log q(x) - \log p(x))dx$$

Example: q is Gaussian





- Approximates well on average in p
- Matches moments for q in EF (e.g. Gaussian)
- Suffices to sample from $p(x)$

- Approximates well on average in q
- Selects a mode
- Requires $\log(p)$

Maximum Likelihood, Cross-Entropy and KL

- Maximum Likelihood Learning for Classification:
  - $(x_i, y_i) \sim p^*$ – training data from true distribution p^*
  - Model: $p(y|x;\theta)$
  - Negative Log-Likelihood (NLL) minimization:

$$\min_\theta \mathbb{E}_{(x,y)\sim p^*}\Big[-\log p(y|x;\theta)\Big]$$

$$= \min_\theta \mathbb{E}_{x\sim p^*(x)}\Big[\underbrace{\sum_y p^*(y|x)(-\log p(y|x;\theta))}_{\text{Crossentropy of } p^*(y|x) \text{ and } p(y|x;\theta)}\Big]$$

Crossentropy of $p^*(y|x)$ and $p(y|x;\theta)$

- Soft labels $p^*(y|x)$
- learning from another model(distillation, generative), mixup, etc.

$$= \min_\theta \mathbb{E}_{x\sim p^*(x)}\Big[D_{\mathrm{KL}}(p^*(y|x)\,\|\,p(y|x;\theta))\Big] \underbrace{-\sum_y p^*(y|x)\log p^*(y|x)}_{\text{Entropy of } p^*(y|x) \text{ — constant in } \theta}$$

  - For minimization in θ, the NLL, Cross-entropy and forward KL are equivalent
- Can we use the reverse $D_{\mathrm{KL}}(p\|p^*)$ for learning from samples $(x, y) \sim p^*$?

Unsupervised Representation Learning

Latent Variable Models

- We explicitly model that multiple observations have some common causes (common factors) that are not directly observed or, *latent*
- Examples:
  - The true class labels for classification are not observed, only labels given by several experts, which may be error-prone. The true label is latent.
  - A text document has a particular topic that we do not know. The frequency of word occurrence and their meaning depend on this common latent topic.
  - In a handwritten note, the style and appearance of letters follow a particular style, unique for each writer and the writer is latent.
  - In our word vector example, words had multiple meanings

I eat grape **jam**.

I was in a traffic **jam**.

Be careful not to **jam** your finger in the door.

Unsupervised Learning

- Model:

  x – observed, z – latent, c – conditioning (side information)

  $p_\theta(x|z,c)$ – model of observations knowing the latent state

  $p_\theta(z|c)$ – model of latent states

  Generative model: $p_\theta(x,z|c) = p_\theta(x|z,c)p_\theta(z,c)$

- Maximum likelihood learning (omitting conditioning on c):

  Observations $\{x_i\}_{i=1}^n \sim p^*(x)$

  Likelihood of x_i: $p_\theta(x_i) = \sum_z p_\theta(x_i, z) = \sum_z p_\theta(x_i|z)p_\theta(z)$

  Log-likelihood:

$$L(\theta) = \mathbb{E}_{x\sim p^*}\left[\log \sum_z p_\theta(x|z)p_\theta(z)\right] \to \max_\theta$$

Would have been nice to swap?

Expectation Maximization (EM) Algorithm

$$
\begin{aligned}
L(\theta) &= \mathbb{E}_{x\sim p^*}\Big[\log p_\theta(x)\Big] \\
&= \mathbb{E}_{x\sim p^*}\Big[\sum_z p_\theta(z|x)\log p_\theta(x)\Big] \\
&= \mathbb{E}_{x\sim p^*}\Big[\sum_z p_\theta(z|x)\log \frac{p_\theta(x,z)}{p_\theta(z|x)}\Big] \\
&= \underbrace{\mathbb{E}_{x\sim p^*,\, z\sim p_\theta(z|x)}}_{\text{Complete x to a joint sample}}\Big[\underbrace{\log p_\theta(x,z)}_{\text{Supervised likelihood}}\Big] + \underbrace{\mathbb{E}_{x\sim p^*}\Big[H(p_\theta(z|x))\Big]}_{\text{Posterior entropy}} \to \max_\theta
\end{aligned}
$$

- EM Algorithm:
  - **E**-step: $p_{\theta^t}(z|x) = \frac{p_{\theta^t}(x,z)}{p_{\theta^t}(x)} = \frac{p_{\theta^t}(x,z)}{\sum_z p_{\theta^t}(x,z)}$ — estimate latents using current model
  - **M**-step: $\theta^{t+1} = \operatorname{argmax}_\theta \mathbb{E}_{x\sim p^*,\, z\sim p_{\theta^t}(z|x)}\Big[\log p_\theta(x,z)\Big]$ — supervised learning

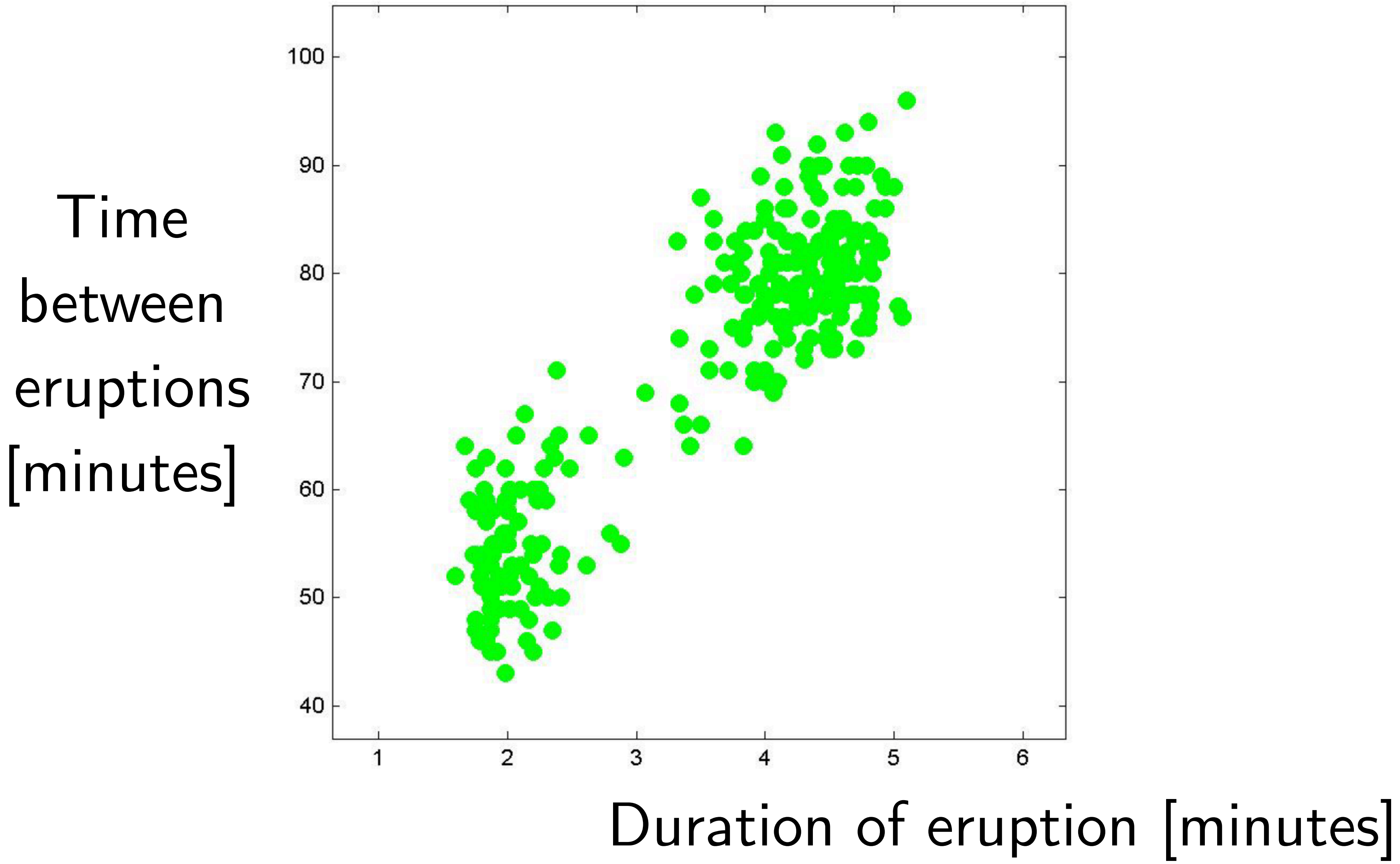
- Maximization is made simpler
- The estimation involving $\sum_z$ stays but is taken out of maximization

Basic EM Example from C. Bishop

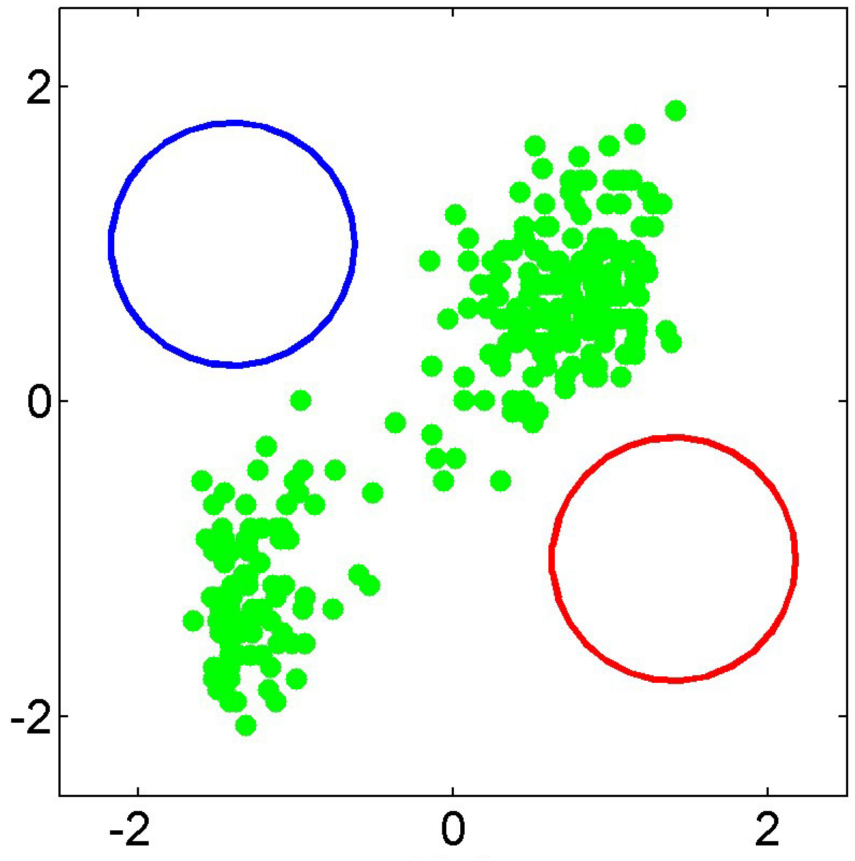
Old Faithful

Basic EM Example from C. Bishop

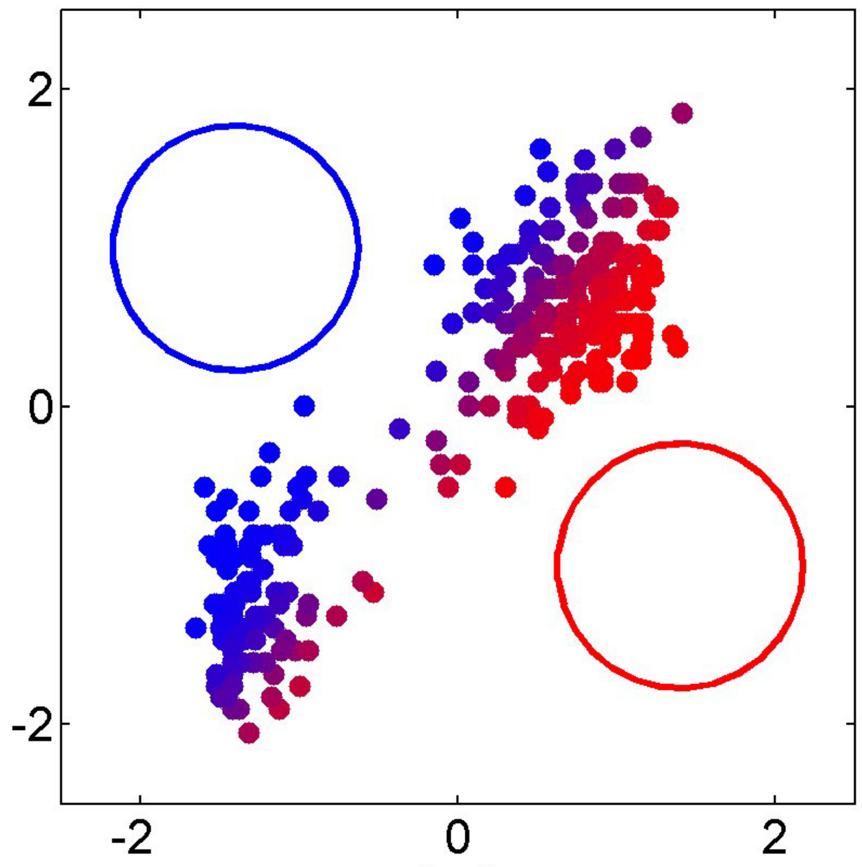
- Dataset



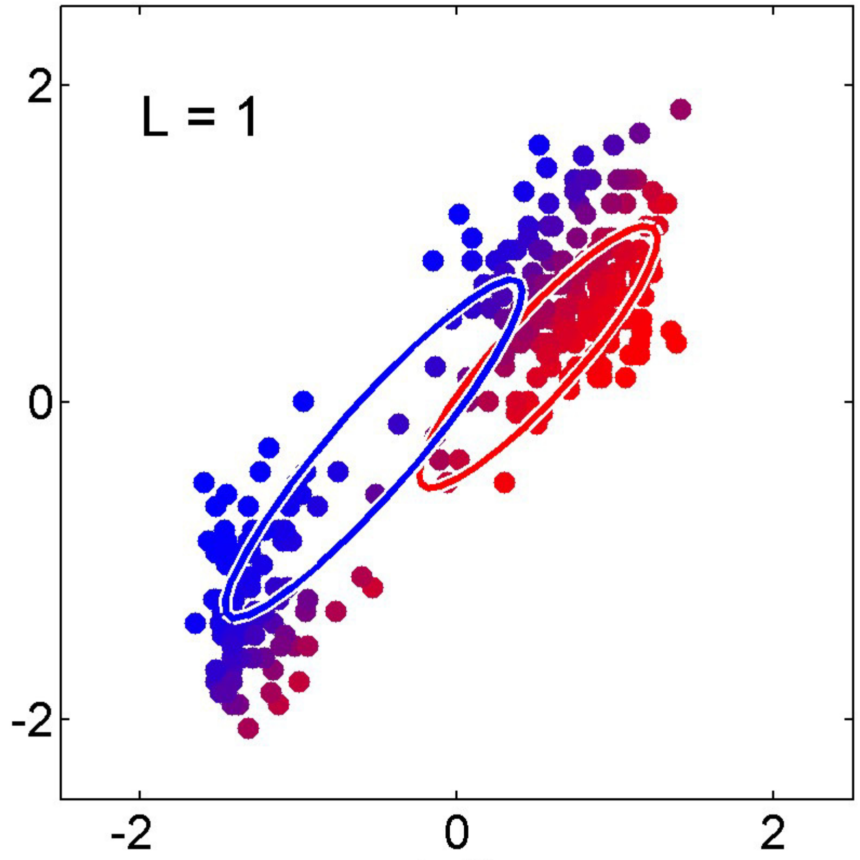
Basic EM Example from C. Bishop



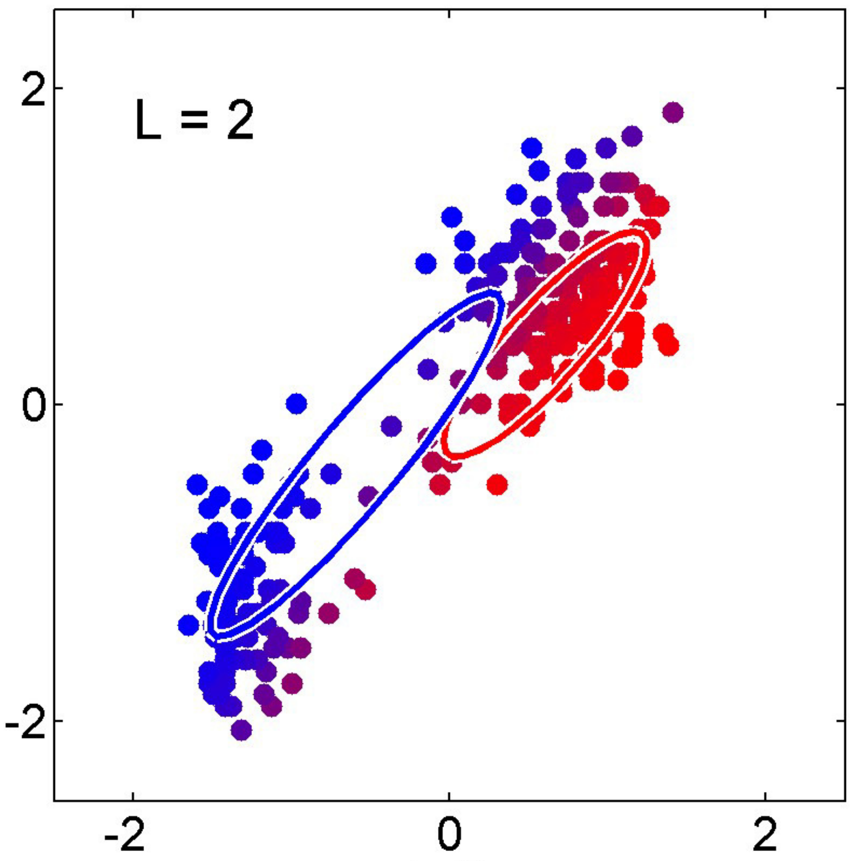
Basic EM Example from C. Bishop



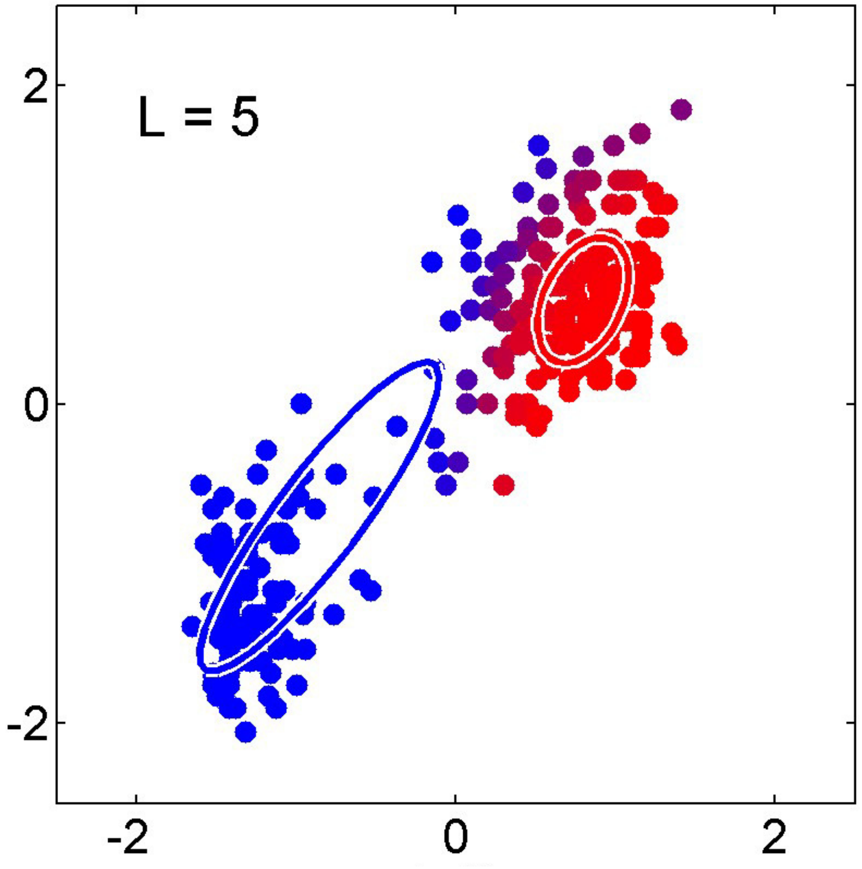
Basic EM Example from C. Bishop



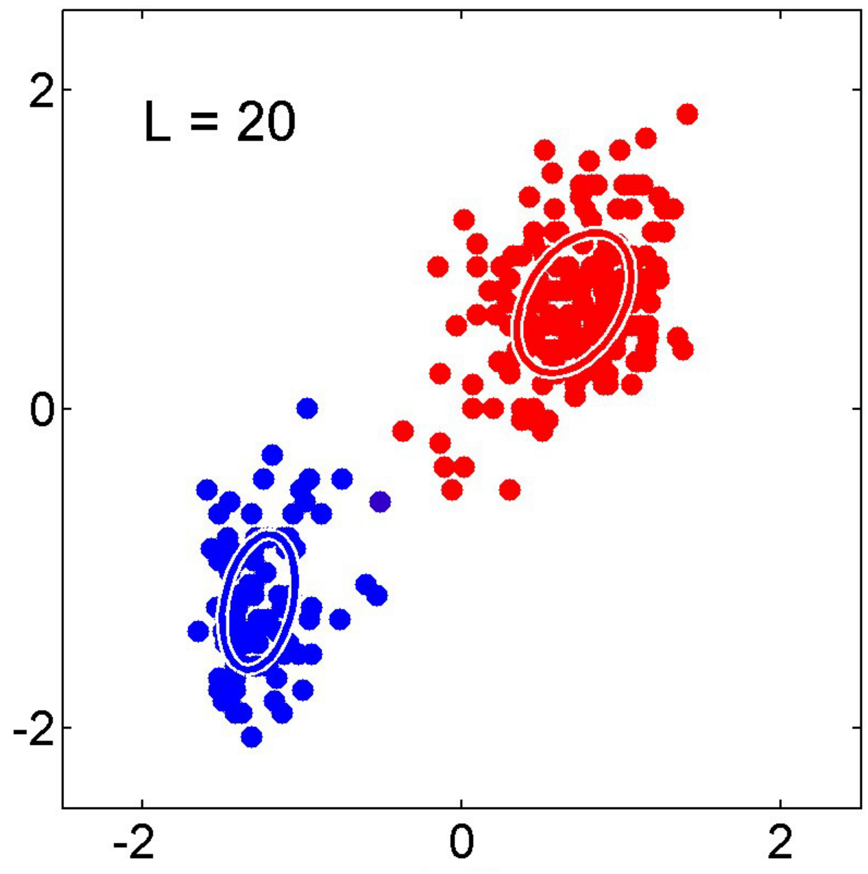
Basic EM Example from C. Bishop



Basic EM Example from C. Bishop



Basic EM Example from C. Bishop



Variational EM

- Want to maximize the log-likelihood of the **data evidence**:

$$\underbrace{\sum_i \log p(x_i)}_{\text{Evidence}} = \sum_i \log \underbrace{\sum_z p(x_i|z)p(z)}_{\text{difficult in general}}$$

$$= \sum_i \log \sum_z q(z|x_i) \frac{p(x_i|z)p(z)}{q(z|x_i)} \geq \underbrace{\sum_i \sum_z q(z|x_i) \log \frac{p(x_i|z)p(z)}{q(z|x_i)}}_{\text{Evidence Lower Bound (ELBO)}}$$

Holds for any distribution $q(z|x_i)$ by Jensen inequality

- Proof using KL (omitting the outer sum in i):

$$\underbrace{\log p(x)}_{\text{Evidence}} - \underbrace{\sum_z q(z|x) \log \frac{p(x,z)}{q(z|x)}}_{\text{ELBO}} = \sum_z q(z|x) \Big(\log p(x) - \log \frac{p(x,z)}{q(z|x)} \Big)$$

$$= \sum_z q(z|x) \Big(-\log \frac{p(x,z)}{p(x)q(z|x)} \Big)$$

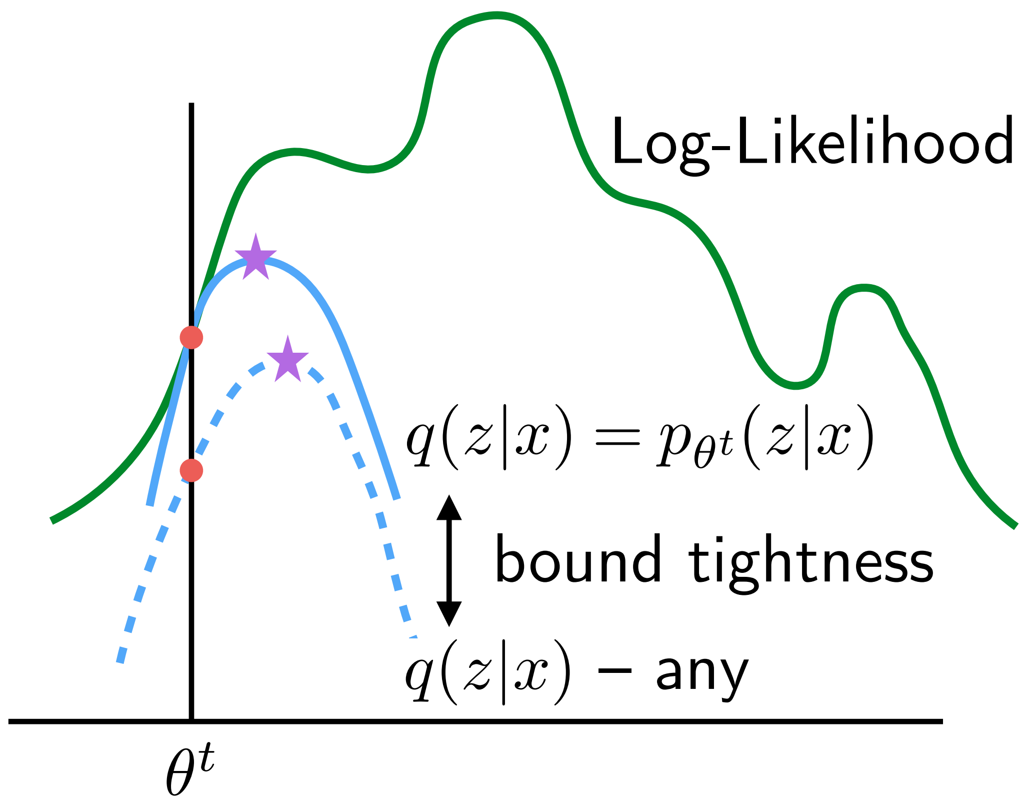
$$= \sum_z q(z|x) \log \frac{q(z|x)}{p(z|x)} = D_{\mathrm{KL}}(q(z|x) \,\|\, p(z|x)) \geq 0.$$

Variational EM Algorithm

$$\text{ELBO}(\theta, q) = \sum_i \sum_z q(z|x_i) \log \frac{p_\theta(x_i|z) p_\theta(z)}{q(z|x_i)}$$

$$= \mathbb{E}_{x \sim p^*,\, z \sim q(z|x)} \Big[\log p_\theta(x,z) \Big] + \mathbb{E}_{x \sim p^*,\, z \sim q(z|x)} \Big[H(q(z|x)) \Big]$$

Like supervised learning



- Variational EM Algorithm:
  - **M**-step: For current q maximize ELBO in θ – like supervised learning (forward KL)
  - **E**-step: For current θ maximize ELBO in q – variational inference (reverse KL)

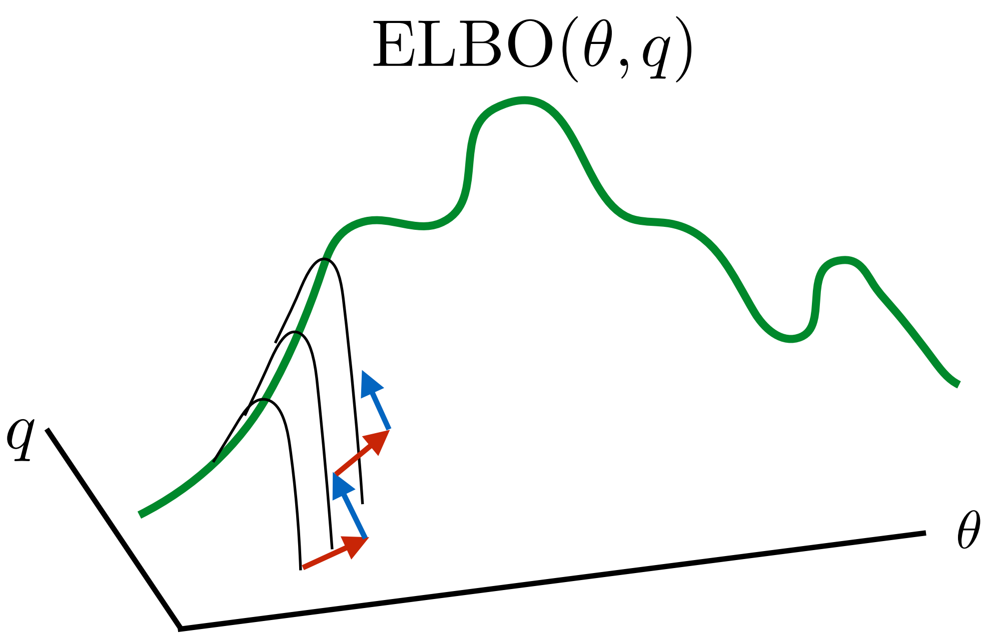
Variational EM Algorithm

$$\text{ELBO}(\theta, q) = \sum_i \sum_z q(z|x_i) \log \frac{p_\theta(x_i|z)p_\theta(z)}{q(z|x_i)}$$

$$= \mathbb{E}_{x\sim p^*,\, z\sim q(z|x)}\Big[\log p_\theta(x,z)\Big] + \mathbb{E}_{x\sim p^*,\, z\sim q(z|x)}\Big[H(q(z|x))\Big]$$

Like supervised learning

Inexact steps are Ok --
we have a global lower bound!



- Variational EM Algorithm:
  - **M**-step: For current q maximize ELBO in θ – like supervised learning (forward KL)
  - **E**-step: For current θ maximize ELBO in q – variational inference (reverse KL)

Variational Inference (E-step)

- Assume q from a parametric family $q_\varphi(z|x)$ (we go straight for amortized form)
- ELBO maximization in φ:

$$\operatorname*{argmax}_{\varphi} \sum_i \sum_z q_\varphi(z|x_i) \log \frac{p_\theta(x_i, z)}{q_\varphi(z|x_i)} \quad (1)$$

$$\operatorname*{argmax}_{\varphi} \sum_i \sum_z q_\varphi(z|x_i) \Big(\log \frac{p_\theta(z|x_i)}{q_\varphi(z|x_i)} + \log p_\theta(x_i) \Big)$$

$$= \operatorname*{argmax}_{\varphi} \sum_i D_{\mathrm{KL}}(q_\varphi(z|x_i) \,\|\, p_\theta(z|x_i)) \quad (2)$$

  - Efficiently approximates the posterior $p_\theta(z|x)$ using the reverse KL divergence
  - To optimize (1) need to evaluate only $p_\theta(x_i, z) = p_\theta(x_i|z) p_\theta(z)$
  - The summation in z is combined with expectation over training data — joint expectation
  - Can differentiate (1) in φ, but not obvious how to get stochastic gradient estimate
- After learning with ELBO, $q_\varphi(z|x)$ is a tractable approximation of the posterior $p_\theta(z|x)$