CSE 291B 1/8

Review of clustering

Inputs $\{\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_N\}$ with $\vec{x}_n \in \mathbb{R}^D$

How to map $\vec{x} \in \mathbb{R}^D$ to $\{1, 2, \ldots, k\}$?

k-means

prototypes $\vec{\mu}_i$

assigment matrix $Y_{in} \in \{0, 1\}$

- minimize $\mathcal{E}(\mu, Y) = \sum_{n=1}^{N} \sum_{i=1}^{K} Y_{in} \| \vec{x}_n - \vec{\mu}_i \|^2$
- iterate: $Y_{in} = \begin{cases} 1 & \text{if } i = \arg\min_j \| \vec{x}_n - \vec{\mu}_j \|^2 \\ 0 & \text{otherwise} \end{cases}$

$$\vec{\mu}_i = \frac{\sum_n Y_{in} \vec{x}_n}{\sum_n Y_{in}}$$

Gaussian mixture models (GMMs)

$$P(\vec{x} \mid z=i) = \frac{1}{(2\pi)^{\frac{D}{2}} \sqrt{\det(\Sigma_i)}} \exp\left\{ -\frac{1}{2} (\vec{x} - \vec{\mu}_i) \Sigma_i^{-1} (\vec{x} - \vec{\mu}_i) \right\}$$

- Maximize: $\mathcal{L} = \sum_n \log \sum_i P(\vec{x} \mid z=i) P(z=i)$
- Relation of EM to k-means

  If $\Sigma_i = \sigma^2 I$ then $\lim_{\sigma^2 \to 0} P(z=i \mid \vec{x}_n) = \begin{cases} 1 & \text{if } i = \arg\min_j \| \vec{x}_n - \vec{\mu}_j \|^2 \\ 0 & \text{otherwise} \end{cases}$

Dimensionality reduction

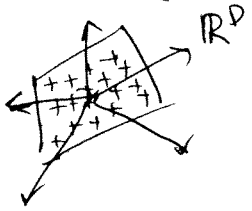
From unlabeled inputs $\{\vec{x}_1, \ldots, \vec{x}_N\}$ with $\vec{x}_n \in \mathbb{R}^D$,
how to map $\vec{x}_n \longrightarrow \vec{y}_n \in \mathbb{R}^d$ with $d \ll D$?

More generally:
How to map $\vec{x} \in \mathbb{R}^D$ to $\vec{y} \in \mathbb{R}^d$ with $d \ll D$?
(ignore subscripts here)

For now, assume d is specified.

Principal component analysis

- Suppose D-dim inputs lie in (or near) a d-dim (linear) subspace.



- Assume inputs are centered:
  $\sum_n \vec{x}_n = 0$ (no loss of generality)
- Compute direction $\hat{u}$ with $\|\hat{u}\|^2 = \hat{u} \cdot \hat{u} = 1$
  which maximizes the projected variance of inputs:

  Maximize:

$$V(\hat{u}) = \frac{1}{N} \sum_{n=1}^{N} (\vec{x}_n \cdot \hat{u})^2$$

$$= \frac{1}{N} \sum_{n=1}^{N} \left(\sum_{\alpha=1}^{D} x_{n\alpha} \hat{u}_\alpha \right) \left(\sum_{\beta=1}^{D} x_{n\beta} \hat{u}_\beta \right)$$

$$= \sum_{\alpha\beta} \hat{u}_\alpha \hat{u}_\beta \left(\frac{1}{N} \sum_{n=1}^{N} x_{n\alpha} x_{n\beta} \right)$$

By def: $C_{\alpha\beta}$ elements of $D \times D$ covariance matrix

$$V(\hat{u}) = \hat{u}^T C \hat{u}$$

- Constrained maximization:

$$\tilde{V}(\hat{u}, \lambda) = \hat{u}^T C \hat{u} + \lambda (1 - \hat{u} \cdot \hat{u})$$ ← Lagrange multiplier

$$\frac{\partial \tilde{V}}{\partial \hat{u}} = 2C\hat{u} - 2\lambda\hat{u} = 0 \longrightarrow C\hat{u} = \lambda\hat{u}$$

eigenvalue equation

Which eigenvector solution?

Let $\hat{u}^{(i)}$ denote eigenvector of C with i^{th} larger eigenvalue $\lambda^{(i)}$

$$V(\hat{u}^{(i)}) = \hat{u}^{(i)T} C \hat{u}^{(i)} = \hat{u}^{(i)T} \lambda^{(i)} \hat{u}^{(i)} = \lambda^{(i)}$$

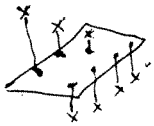
$\Longrightarrow$ Direction with max variance is $\hat{u}^{(1)}$ with largest eigenvalue.

- To compute d-dim subspace with max variance, take top d <u>eigenvectors as orthogonal basis</u> for subspace.

Map $\vec{x} \in \mathbb{R}^D$ to $\vec{y} \in \mathbb{R}^d$ by: $y_i = \vec{x} \cdot \hat{u}^{(i)}$ → since C is positive semi-definite

- Also look at reconstruction error:

$$\mathcal{E} = \sum_n \left\| \vec{x}_n - \sum_{i=1}^{d} \hat{u}^{(i)} (\vec{x}_n \cdot \hat{u}^{(i)}) \right\|^2$$

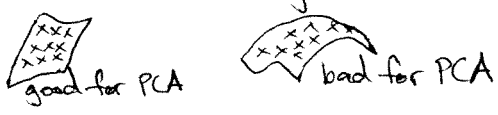


$$= \sum_n \left[\vec{x}_n^2 - 2 \sum_{i=1}^{d} (\vec{x}_n \cdot \hat{u}^{(i)})^2 + \sum_{i=1}^{d} (\vec{x}_n \cdot \hat{u}^{(i)})^2 \right]$$

$$= \sum_n \vec{x}_n^2 - \underbrace{\sum_{i=1}^{d} (\vec{x}_n \cdot \hat{u}^{(i)})^2}_{\text{variance of projected inputs}}$$

Minimizing reconstruction error is same as maximizing projected variance (for linear projections).

- Pros
  1) no local minima; eigenvector computation
  2) no tuning parameters other than choosing $d \ll D$
- Cons
  1) Scaling with dimensionality
     - $O(ND^2)$ to compute covariance matrix
     - $O(dD^2)$ to compute top d eigenvectors
  2) No probabilistic interpretation
     - no explicit model of out-of-space noise: implicit parametric assumptions?
     - hard to compose models in principled way
  3) restricted to discovering linear structure



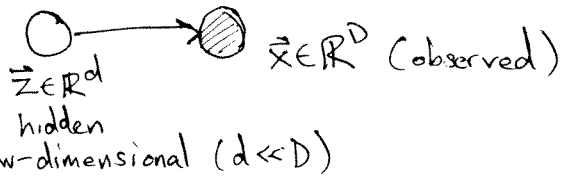
Factor analysis

Assume inputs $\{\vec{x}_1, \ldots, \vec{x}_N\}$ are sampled (i.i.d.) from probability density function $P(\vec{x})$.

Model $P(\vec{x})$ by gaussian BN with hidden variable.

Estimate model to maximize $\mathcal{L} = \sum_n \log P(\vec{x}_n)$.

- Belief network



- Gaussian prior for latent variable:

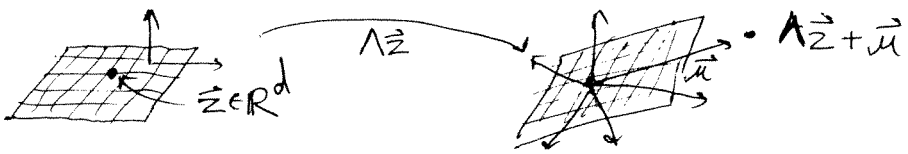
$$P(\vec{z}) = \frac{1}{(2\pi)^{d/2}} e^{-\frac{1}{2}\|\vec{z}\|^2}$$

zero mean
unit (identity) covariance matrix

- Generative model for $\vec{x}$
  - sample $\vec{z} \in \mathbb{R}^d$ from $P(\vec{z})$
  - linearly project into D dimensions:

$$\vec{z} \in \mathbb{R}^d \longrightarrow \Lambda\vec{z} + \vec{\mu} \in \mathbb{R}^D$$

Λ: D×d factor loading matrix; $\vec{\mu}$: offset from origin



  - add independent zero-mean gaussian noise with variance ψ_{ii} to i^{th} component $(\Lambda\vec{z} + \vec{\mu})_i$
    (this will "fuzz up" the transformation)

Let $\psi = \begin{bmatrix} \psi_{11} & & \varnothing \\ & \psi_{22} & \\ & & \ddots \\ \varnothing & & \psi_{DD} \end{bmatrix}$ denote <u>diagonal</u> covariance matrix of this (high-dimensional) noise
D-dim

- Gaussian conditional distribution $\vec{z} \in \mathbb{R}^d \rightarrow \vec{x} \in \mathbb{R}^D$

$$P(\vec{x}\,|\,\vec{z}) = \frac{1}{(2\pi)^{D/2}\sqrt{\det(\Psi)}} \exp\left\{-\frac{1}{2}\left[\vec{x}-\Lambda\vec{z}-\vec{\mu}\right]^T \Psi^{-1}\left[\vec{x}-\Lambda\vec{z}-\vec{\mu}\right]\right\}$$

Key intuition: for small ψ_{ii}, most variation occurs inside subspace $S(\Lambda)$ spanned by columns of Λ

- noise model: for non-uniform ψ_{ii}, vectors with same projection on $S(\Lambda)$ can have different likelihoods

- Marginal distribution

$$P(\vec{x}) = \int_{\vec{z}\in\mathbb{R}^d} P(\vec{x}\,|\,\vec{z})\,P(\vec{z})\,d\vec{z} \quad \text{(also gaussian)}$$

$$= \frac{1}{(2\pi)^{D/2}\sqrt{\det(\Psi+\Lambda\Lambda^T)}} \exp\left\{-\frac{1}{2}(\vec{x}-\vec{\mu})^T(\Psi+\Lambda\Lambda^T)^{-1}(\vec{x}-\vec{\mu})\right\}$$

mean $E[\vec{x}] = \vec{\mu}$

covariance matrix $(\Psi+\Lambda\Lambda^T) = E\left[(\vec{x}-\vec{\mu})(\vec{x}-\vec{\mu})^T\right]$

- Scaling with dimensionality

$(\Psi+\Lambda\Lambda^T)$ is $D\times D$ (non-sparse) matrix

Naively, $O(D^3)$ to invert or compute determinant

Saved by matrix identities...

- Matrix inversion lemma:

$$(\Psi+\Lambda\Lambda^T)^{-1} = \underbrace{\Psi^{-1}}_{D\times D \text{ diagonal}} - \Psi^{-1}\Lambda\underbrace{\left(\underbrace{I_d}_{d\times d \text{ identity matrix}} + \underbrace{\Lambda^T\Psi^{-1}\Lambda}_{d\times d \text{ matrix}}\right)^{-1}}_{O(d^3) \text{ to invert}}\Lambda^T\Psi^{-1}$$

- Matrix determinant lemma:

$$\det(\Psi+\Lambda\Lambda^T) = \underbrace{(\det\Psi)}_{\text{diagonal}} \cdot \det\underbrace{(I+\Lambda^T\Psi^{-1}\Lambda)}_{d\times d \text{ matrix}}$$

- Next lecture:
  - How to estimate parameters Λ, Ψ, $\vec{\mu}$?
  - How to combine FA and GMMs?