Optimization in Machine Learning: Lecture 5

Machine Learning Models

by Xiaolin Huang xiaolinhuang@sjtu.edu.cn SEIEE 2-429

Institute of Image Processing and Pattern Recognition

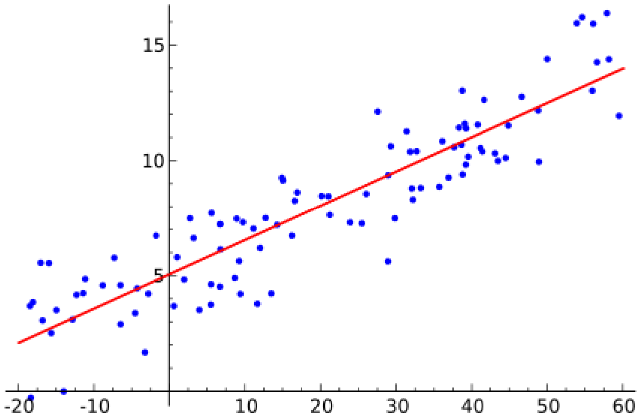
http://www.pami.sjtu.edu.cn/

上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

目录 Contents

Linear Regression

- **Linear regression**
  - from training data $\{a_i, b_i\}_{i=1}^m, a_i \in \mathbf{R}^n, b_i \in \mathbf{R}$,
  - to establish a linear model



$$f(a) = x_0 + x_1 a_1 + \cdots + x_n a_n = x^\top a$$

bias: now we can simply regard it as an additional dimension before investigating learning theory

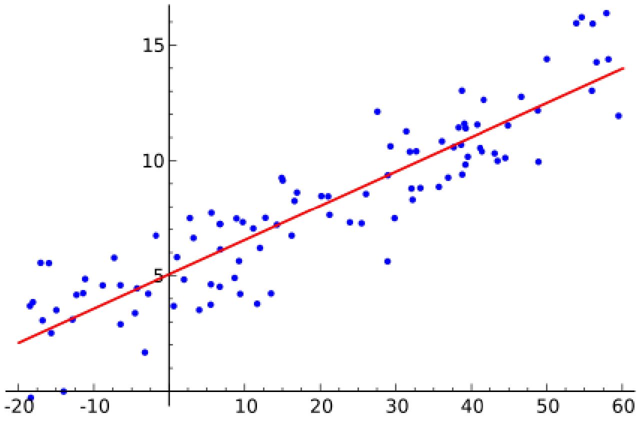
  - **matrix form**

$$A = \begin{bmatrix} a_1^\top \\ a_2^\top \\ \vdots \\ a_m^\top \end{bmatrix} = [a_1, a_2, \dots, a_m]^\top \in \mathbf{R}^{m \times n} \qquad Ax \in \mathbf{R}^m,\ b \in \mathbf{R}^m$$

Least Squares

- residual $r_i = x^\top a_i - b_i$
- squared residual $l(r_i) = r_i^2 = (x^\top a_i - b_i)^2$
- sum of the squared residual

$$\sum_{i=1}^{m} l(r_i) = \sum_{i=1}^{m} (x^\top a_i - b_i)^2$$



- convexity: nonnegative sum, composition of affine and quadratic functions
- (ordinary) least squares

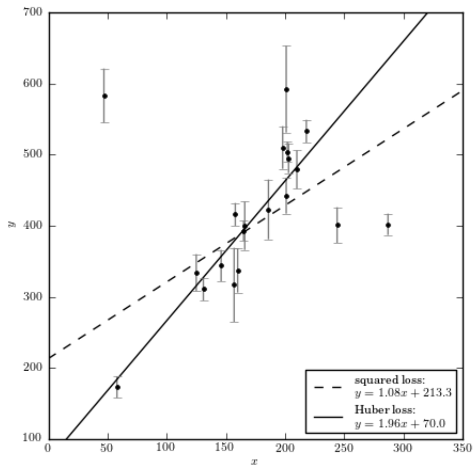
$$\min_x \|Ax - b\|_2^2 = (Ax - b)^\top (Ax - b) = x^\top A^\top x - 2b^\top Ax + b^\top b$$

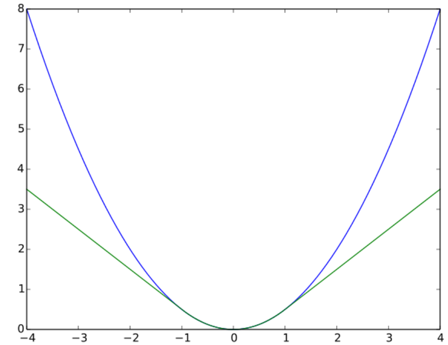
Huber Loss

- LS is very sensitive to outlier
    - large residual is more likely to be an outlier
    - squared loss gives too much attention on outliers
    - linear loss consider all the residuals equally
- an ideal loss function
    - give squared penalty when the residual is small
    - give linear penalty when the residual is large
- Huber loss

$$L_{\text{huber}}(r) = \begin{cases} r^2 & \text{if } |r| < c \\ |r| & \text{if } |r| \geq c \end{cases}$$

- what is the problem?



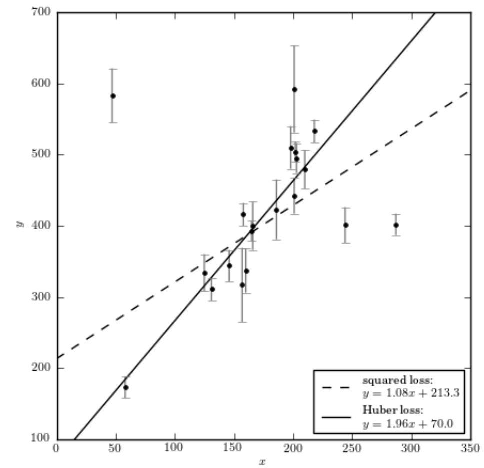


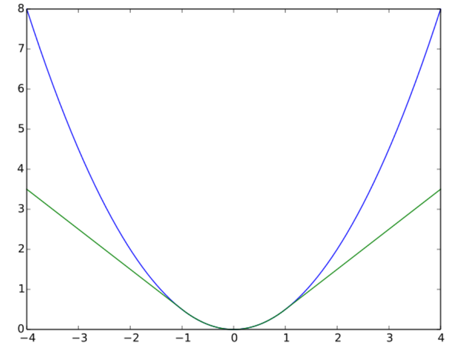
Huber Loss

- LS is very sensitive to outlier
  - large residual is more likely to be an outlier
  - squared loss gives too much attention on outliers
  - linear loss consider all the residuals equally
- an ideal loss function
  - give squared penalty when the residual is small
  - give linear penalty when the residual is large
- Huber loss

$$L_{\text{huber}}(r) = \begin{cases} r^2/2 & \text{if } |r| < c \\ c(|r| - c/2) & \text{if } |r| \geq c \end{cases}$$

- what is the problem?
- continuity
- smoothness





LAD and Quantile Regression

- absolute residual, l1-norm estimation

$$|r_i| = |a_i^\top x - b_i|$$

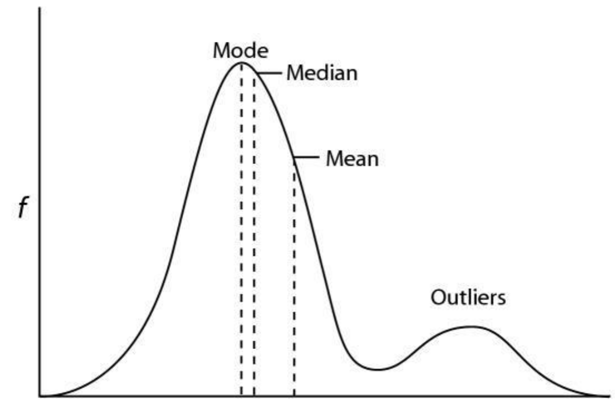
- least absolute derivations (LAD) regression

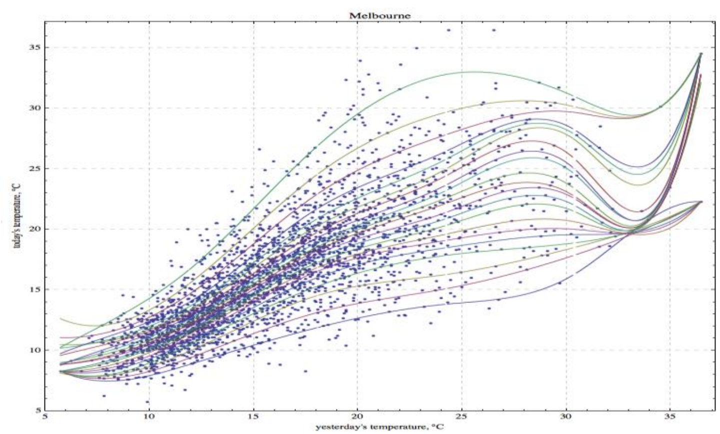
$$\min_x \ \|Ax - b\|_1$$

  - it is even older than OLS
  - the result is the median value

- quantile regression with asymmetric loss

$$L_\tau(r) = \begin{cases} -(1-\tau)r & \text{if } r < 0 \\ \tau r & \text{if } r \geq 0 \end{cases}$$



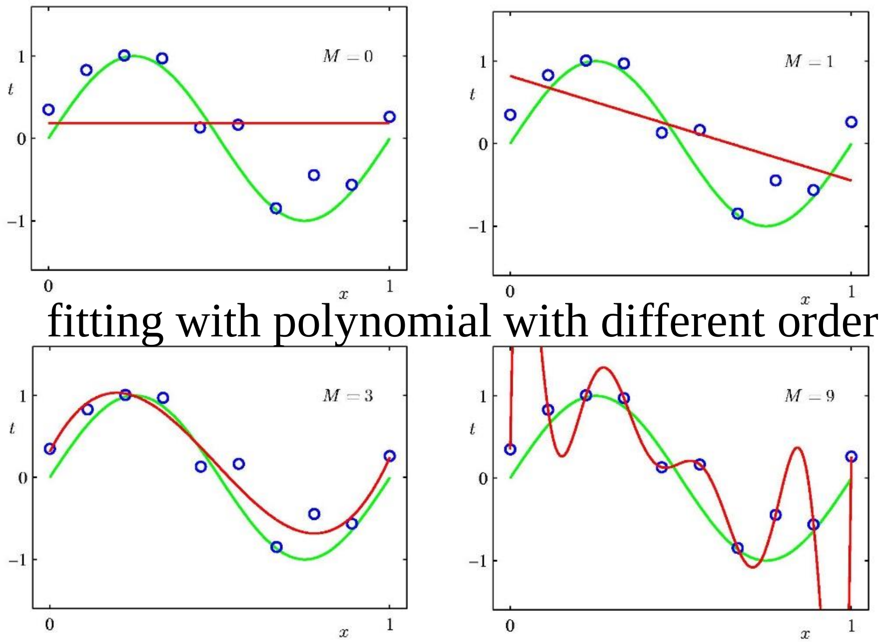


Overfitting

- the model complexity is too high
  - complex model: prefer simple model
  - large energy: as small as possible
  - effective region: prefer local basis rather than global one
- ridge regression

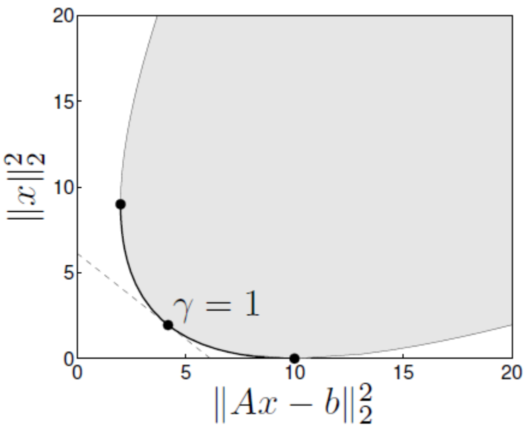
  Tikhonov regularization

$$\min_{x} \ \gamma\|x\|_2^2 + \|Ax-b\|_2^2$$



fitting with polynomial with different order

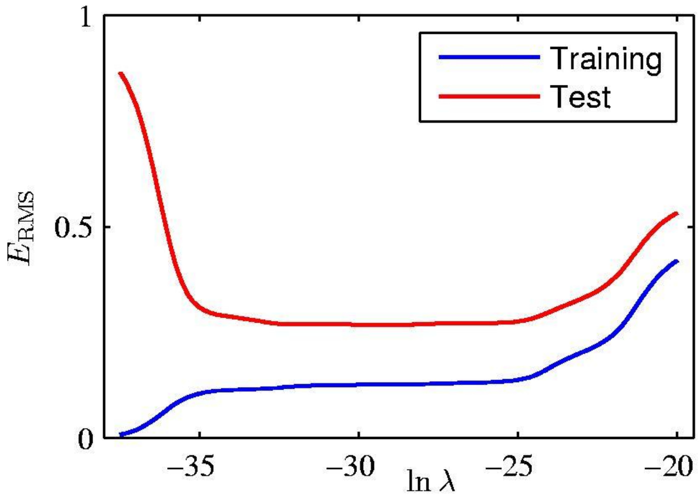
Overfitting



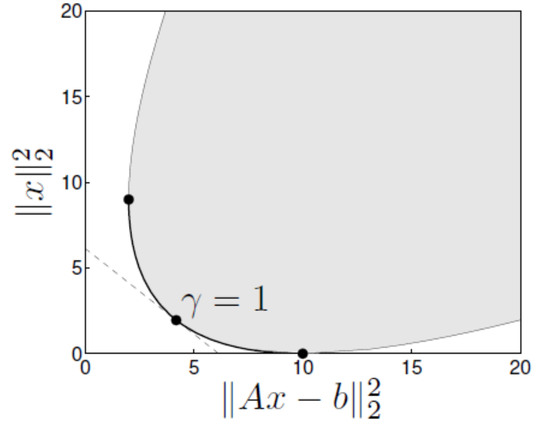
- the model complexity is too high
  - complex model: prefer simple model
  - large energy: as small as possible
  - effective region: prefer local basis rather than global one
- ridge regression

  Tikhonov regularization

$$\min_{x} \ \gamma\|x\|_2^2 + \|Ax - b\|_2^2$$



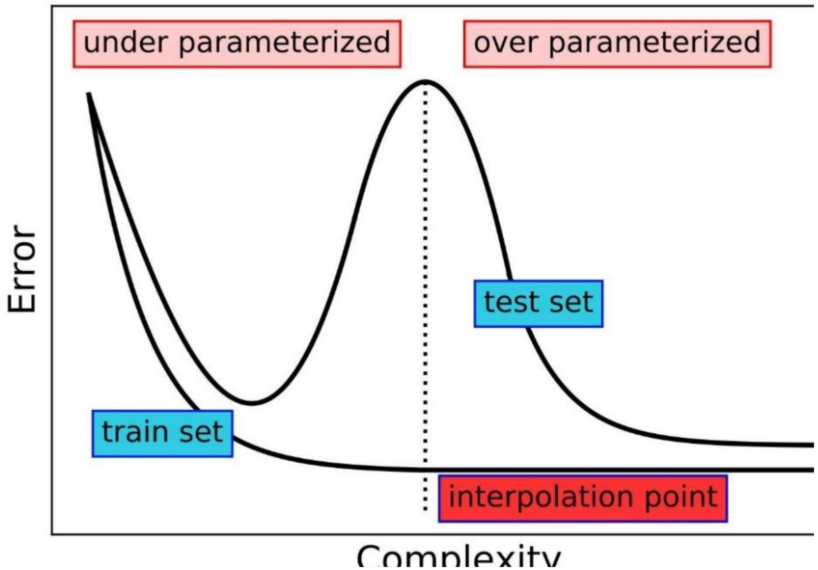
Overfitting



- the model complexity is too high
  - complex model: prefer simple model
  - large energy: as small as possible
  - effective region: prefer local basis rather than global one
- ridge regression

  Tikhonov regularization

$$\min_{x} \ \gamma\|x\|_2^2 + \|Ax - b\|_2^2$$



General Error Decomposition

- **Generalization problem:**
  - risk on the unknown probability ρ: $\boldsymbol{R}_\rho(f)$
  - risk on samples $z \sim \rho$: $\boldsymbol{R}_z(f)$
  - small $\boldsymbol{R}_z(f)$ does not necessarily lead to small $\boldsymbol{R}_\rho(f)$
  - denote $f^* = \mathrm{argmin}_f\ \boldsymbol{R}_\rho(f)$ $\quad f_F^* = \mathrm{argmin}_{f \in \boldsymbol{F}}\ \boldsymbol{R}_\rho(f)$

    $f_z^* = \mathrm{argmin}_{f \in \boldsymbol{F}}\ \boldsymbol{R}_z(f)$ $\quad \widetilde{f}_z$: **the learned function**

  - we have

$$E\left(\boldsymbol{R}_\rho(f_z^*) - \boldsymbol{R}_\rho(\widetilde{f}_z)\right)$$
$$= E\left(\boldsymbol{R}_\rho(f_F^*) - \boldsymbol{R}_\rho(f^*)\right) + E\left(\boldsymbol{R}_\rho(f_z^*) - \boldsymbol{R}_\rho(f_F^*)\right) + E\left(\boldsymbol{R}_\rho(f_z^*) - \boldsymbol{R}_\rho(\widetilde{f}_z)\right)$$

General Error Decomposition

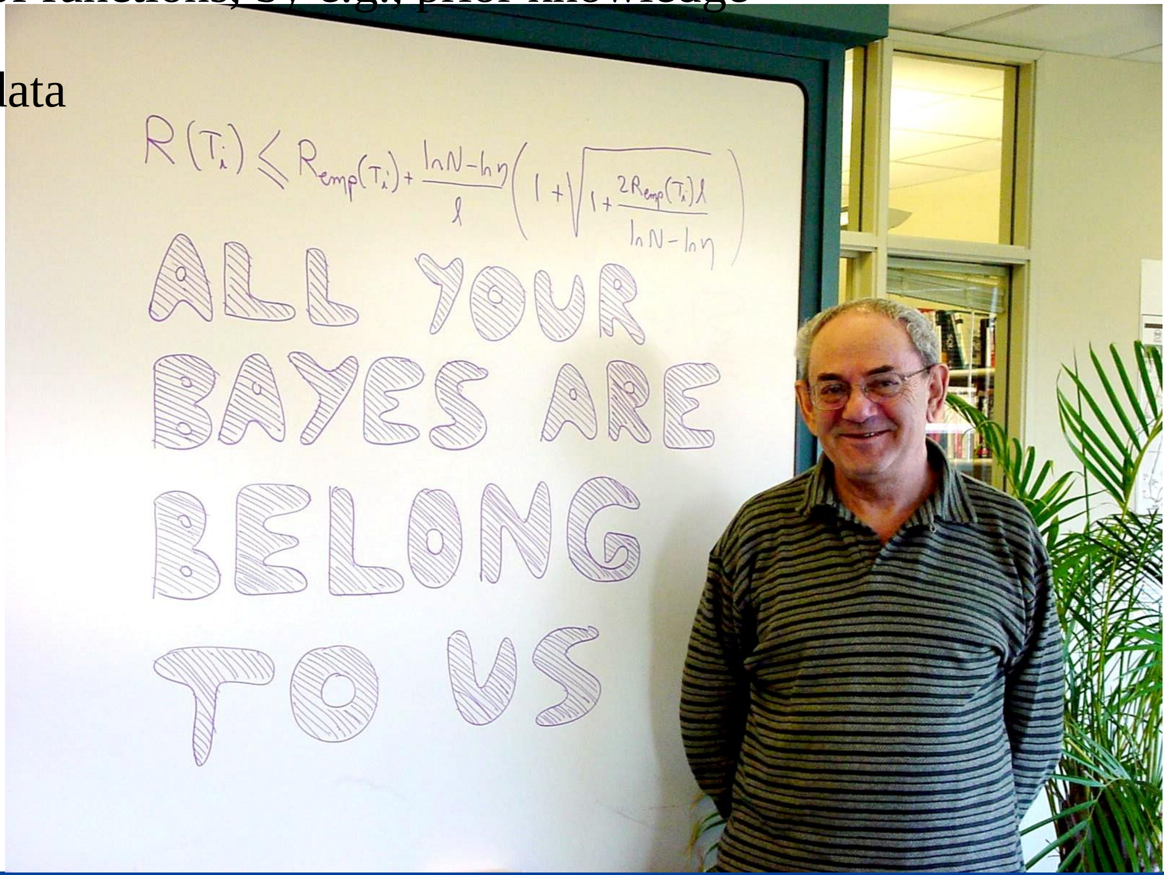
$$E\left(\boldsymbol{R}_\rho(f_Z^*) - \boldsymbol{R}_\rho(\widetilde{f}_Z)\right)$$
$$= E\left(\boldsymbol{R}_\rho(f_F^*) - \boldsymbol{R}_\rho(f^*)\right) + E\left(\boldsymbol{R}_\rho(f_Z^*) - \boldsymbol{R}_\rho(f_F^*)\right) + E\left(\boldsymbol{R}_\rho(f_Z^*) - \boldsymbol{R}_\rho(\widetilde{f}_Z)\right)$$

approximation error | estimation error | optimization error

- **approximation error:** learning capability of the functional space
  - enlarge the family of functions
- **estimation error:** gap between training data and test data
  - prefer a smaller family of functions, by e.g., prior knowledge
  - more data set
- **optimization error:**

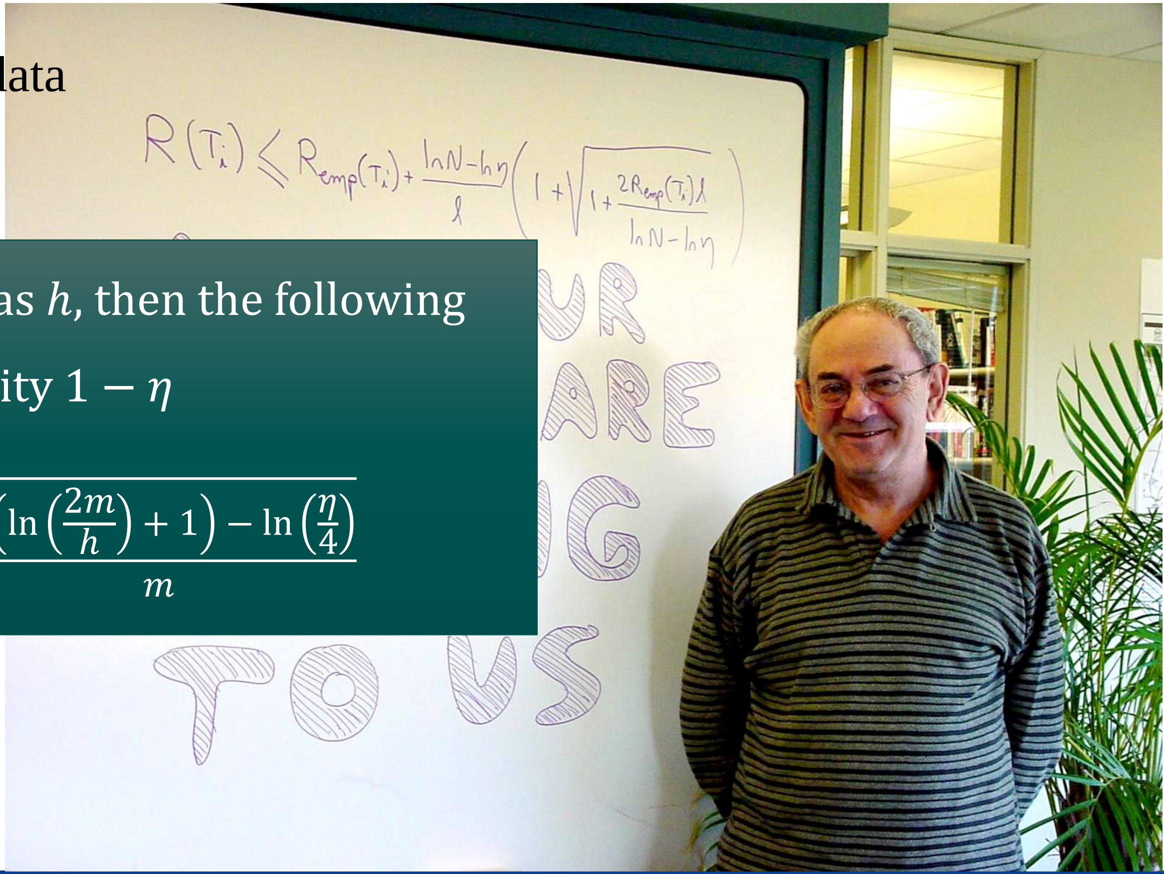
General Error Decomposition

- **estimation error:** gap between training data and test data
  - prefer a smaller family of functions, by e.g., prior knowledge
  - consider more training data



General Error Decomposition

- **estimation error:** gap between training data and test data
    - prefer a smaller family of functions, by e.g., prior knowledge
    - consider more training data



denote the VC dimension of f as h, then the following inequality holds with probability $1-\eta$

$$R_\rho(f) \le R_{\text{empirical}}(f) + \sqrt{\frac{h\left(\ln\left(\frac{2m}{h}\right)+1\right)-\ln\left(\frac{\eta}{4}\right)}{m}}$$

- few-shot learning

General Error Decomposition

$$E\left(\boldsymbol{R}_\rho(f_Z^*) - \boldsymbol{R}_\rho(\tilde{f}_Z)\right)$$
$$= E\left(\boldsymbol{R}_\rho(f_F^*) - \boldsymbol{R}_\rho(f^*)\right) + E\left(\boldsymbol{R}_\rho(f_Z^*) - \boldsymbol{R}_\rho(f_F^*)\right) + E\left(\boldsymbol{R}_\rho(f_Z^*) - \boldsymbol{R}_\rho(\tilde{f}_Z)\right)$$

approximation error

estimation error

optimization error

- **approximation error:** learning capability of the functional space
  - enlarge the family of functions
- **estimation error:** gap between training data and test data
  - prefer a smaller family of functions, by e.g., prior knowledge
  - more data
- **optimization error:**
  - running longer time, make problem simple, consider less data

LASSO

- regularization term is useful to control model complexity
- LASSO: least absolute shrinkage and **selection** operator

$$\min_x \quad \gamma \sum_{j=1}^{n} |x_j| + \frac{1}{2} \sum_{i=1}^{m} (x^T a_i - b_i)^2$$

- the use of l1 norm
  - sparsity

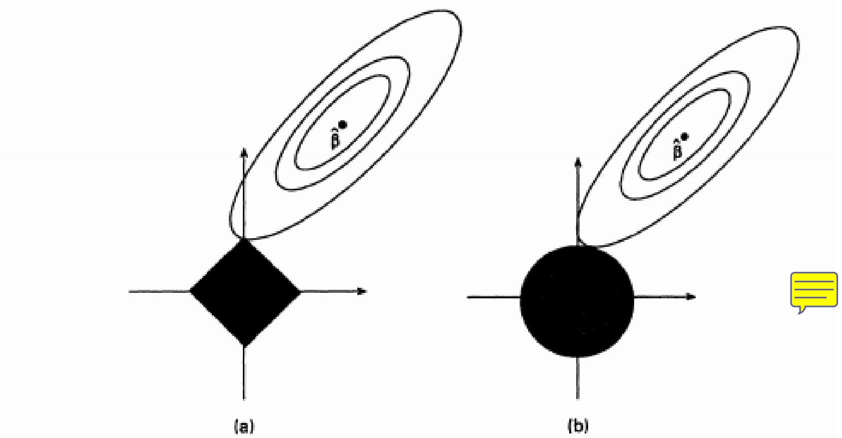


Fig. 2. Estimation picture for (a) the lasso and (b) ridge regression

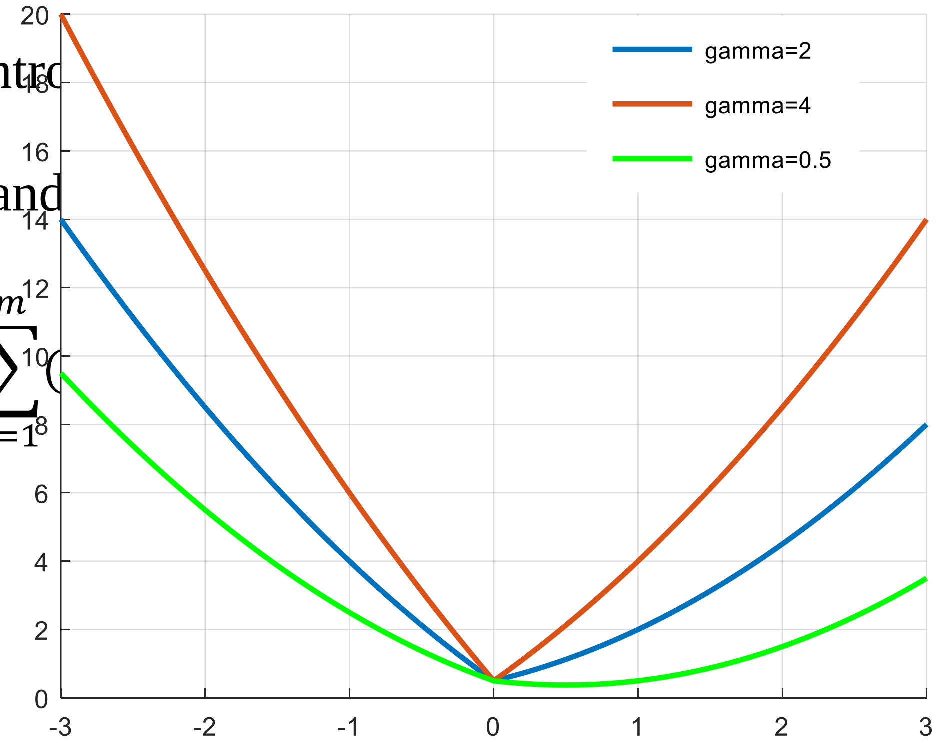
Robert Tibshirani

LASSO

- regularization term is useful to contro
- LASSO: least absolute shrinkage and

$$\min_{x} \quad \gamma \sum_{j=1}^{n} |x_j| + \frac{1}{2} \sum_{i=1}^{m} ($$

- the use of l1 norm
  - sparsity



$$\min_{x} \;\; \gamma|x| + 1/2(x-1)^2$$

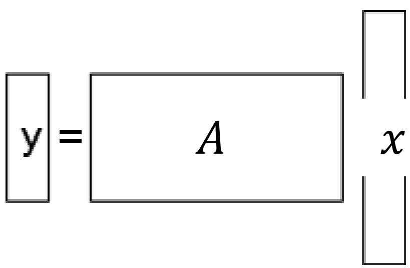
$$f'(x) = \gamma \mathrm{sgn}(x) + (x-1) \quad \Rightarrow \quad x^* = 0 \text{ if } \gamma > 1$$

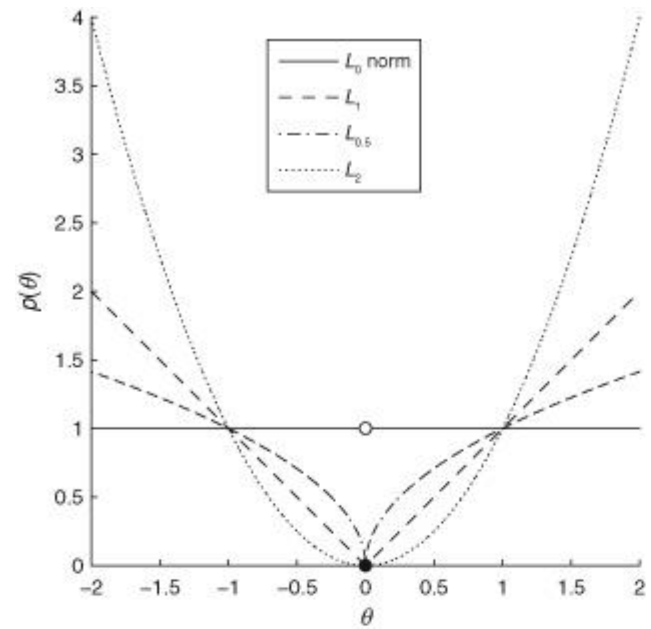
$$\min_{x} \;\; 1/2\gamma x^2 + 1/2(x-1)^2$$

$$f'(x) = \gamma x + (x-1) = 0 \quad \Rightarrow \quad x^* \neq 0$$

Compressive Sensing

- Sparsity measurement:
  - l0 norm: $\|x\|_0 = \sum I(x_i \neq 0)$
  - l1 norm: $\|x\|_1 = \sum |x_i|$
- Compressive sensing/compressed sensing

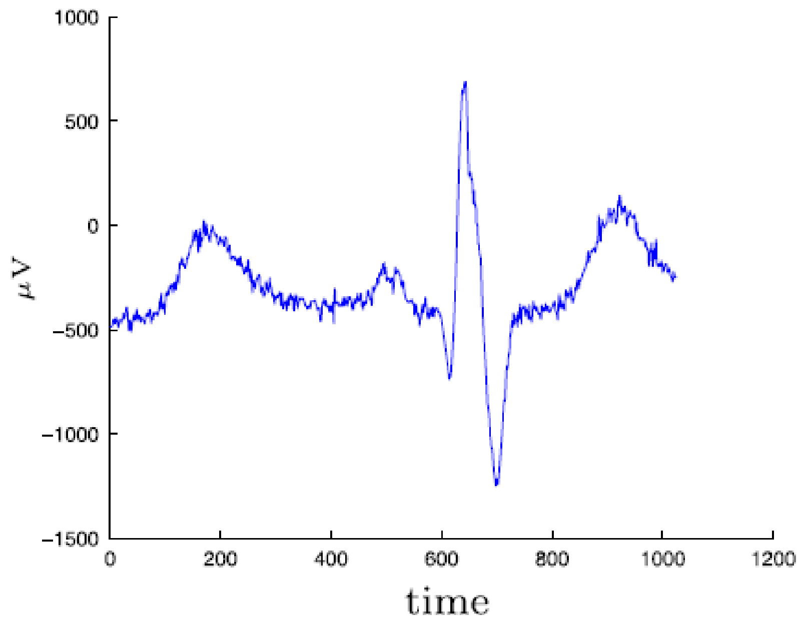


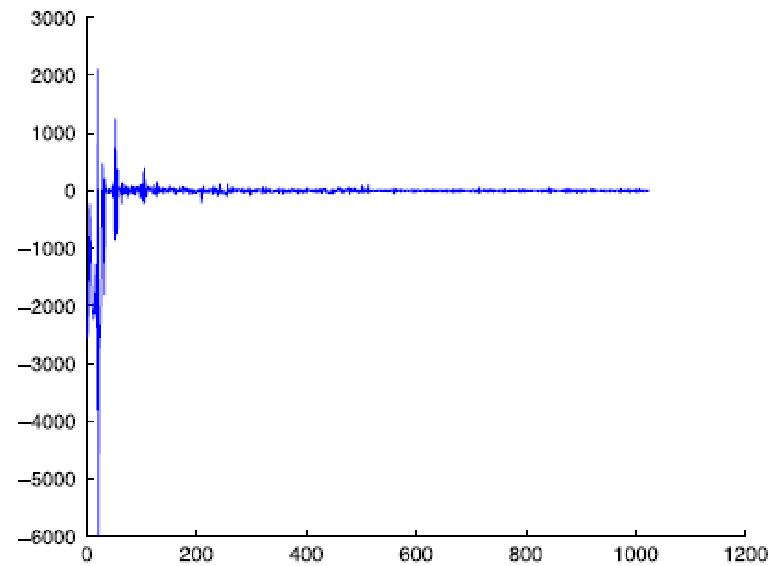


  - Around 2004, *Emmanuel Candès*, *Terence Tao*, and *David Donoho*: given knowledge about a signal's sparsity, the signal may be reconstructed with even fewer samples than the sampling theorem requires

LASSO and Compressive Sensing

- sparsity is one nature of big data
- the key is to find the a good representation that is sparse and accurate





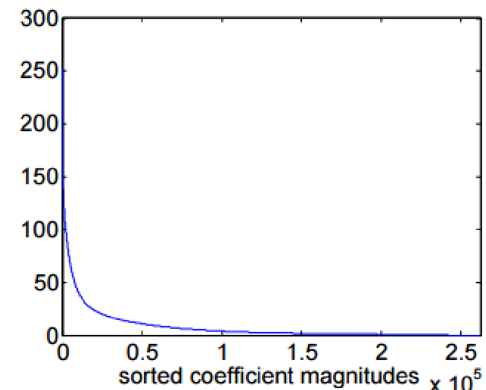
components of Daubechies wavelet

electro-cardiography (ECG)

LASSO and Compressive Sensing

- sparsity is one nature of big data
- the key is to find the a good representation that is sparse and accurate

(c) Local variation



(f) Local variation

$$\left|X_{i,j} - X_{i+1,j}\right|$$

$$\left|X_{i,j} - X_{i,j+1}\right|$$

Variant of Lasso: TV

- for an image $X \in \mathbf{R}^{n\times n}$, its total variation (TV) is defined as

$$\|X\|_{TV} = \sum_{i=1}^{n-1}\sum_{j=1}^{n-1}\sqrt{\left(X_{i,j} - X_{i+1,j}\right)^2 + \left(X_{i,j} - X_{i,j+1}\right)^2} \qquad \|X\|_{TV} = \sum_{i=1}^{n-1}\sum_{j=1}^{n-1}\left|X_{i,j} - X_{i+1,j}\right| + \left|X_{i,j} - X_{i,j+1}\right|$$

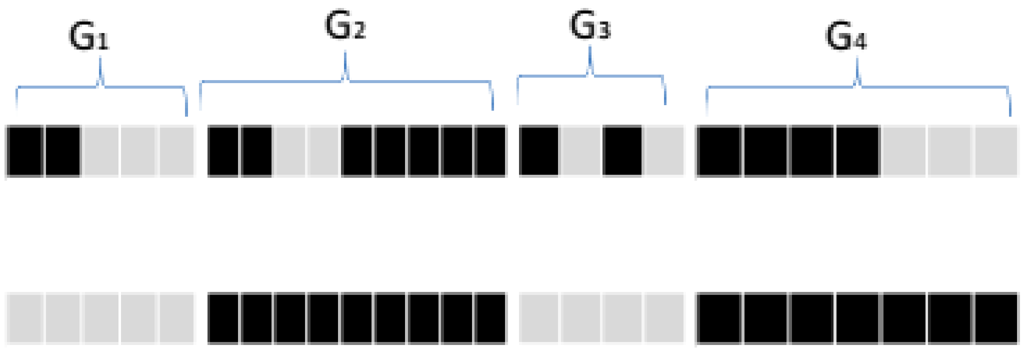
which is better?

$$\min_{X}\ \ \lambda\|X\|_{TV} + \|F - X\|_2^2$$

Original　　Noisy image F　　Denoised image X

QUIZ

- if the vector has several groups and only a part of them are non-zero



- regular l1 minimization does not help (why? and how to?)

$$\min_{x} \quad \gamma \sum_{i=1}^{n} |x_i| + \frac{1}{2} \sum_{i=1}^{m} (x^\top a_i - b_i)^2$$

$$\min_{x} \quad \gamma \sum_{k=1} \sqrt{\sum_{i \in G_k} x_i^2} + \frac{1}{2} \sum_{i=1}^{m} (x^\top a_i - b_i)^2$$

Dictionary Learning

- LASSO is to learn the coefficient

$$\min_{x} \quad \gamma \|x\|_1 + \frac{1}{2}\|AX - b\|_2^2$$

- to learn the coefficient and the basis together

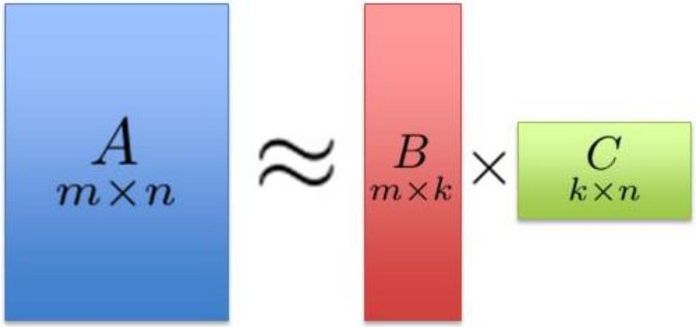
$$\min_{x,\Phi} \quad \gamma \|x\|_1 + \frac{1}{2}\|A\Phi X - b\|_2^2$$

  - non-convex
  - Dictionary Learning
    - gradient descent
    - optimal direction iteration
    - K-means Singular Value Decomposition

Extension to Matrix

- matrix norm
  - entrywise sparsity

$$\|A\|_1 = \Sigma_i \Sigma_j |a_{ij}|$$



  - nuclear norm

$$\|A\|_* = \Sigma_{i=1}^{n} |\sigma_i(A)|$$

the largest *i*-th singular value

  - Frobenius norm

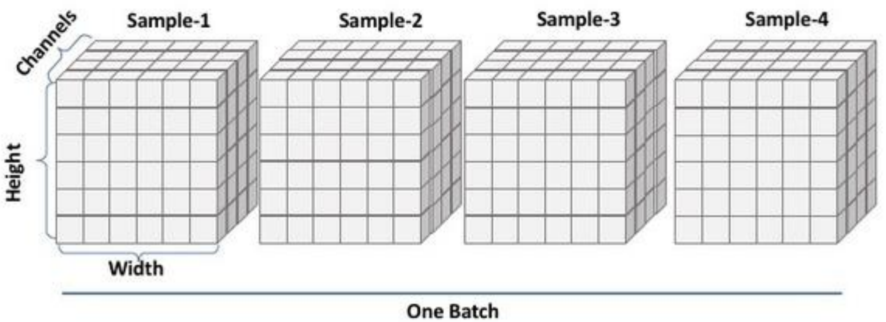
$$\|A\|_F = \sqrt{\Sigma_i \Sigma_j |a_{ij}|^2} = \sqrt{\mathrm{tr}(A^* A)} = \sqrt{\Sigma_i \sigma_i^2(A)}$$

  - norms induced by vector norms

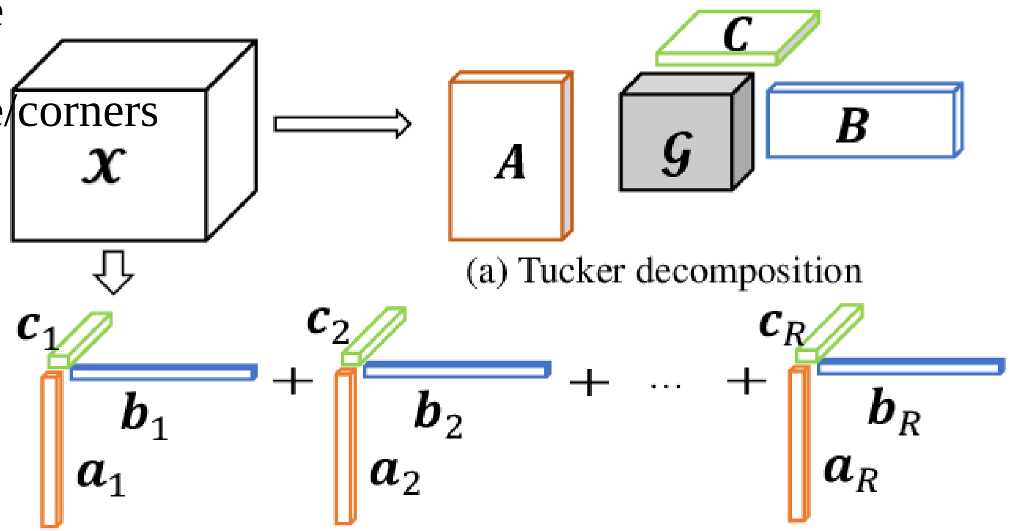
$$\|A\|_p = \sup\{\|Ax\|_p : \|x\|_p = 1\} = \begin{cases} \max_j \Sigma_i |a_{ij}|, & p = 1 \\ \sigma_i(A), & p = 2 \\ \max_i \Sigma_j |a_{ij}|, & p = \infty \end{cases}$$

Extension to Tensor

- if we go to even higher space
  - video
  - RGB depth
  - RGB-D in different angles
  - RGB-D in different angles/time
  - RGB-D in different angles/time/corners



https://www.quora.com/What-are-the-main-normalization-layers-in-artificial-neural-networks



(a) Tucker decomposition

(b) CP decomposition

https://www.researchgate.net/figure/The-illustration-of-a-Tucker-decomposition-and-b-CP-factorization-of-an-n-1-n-2-n_fig1_321902217

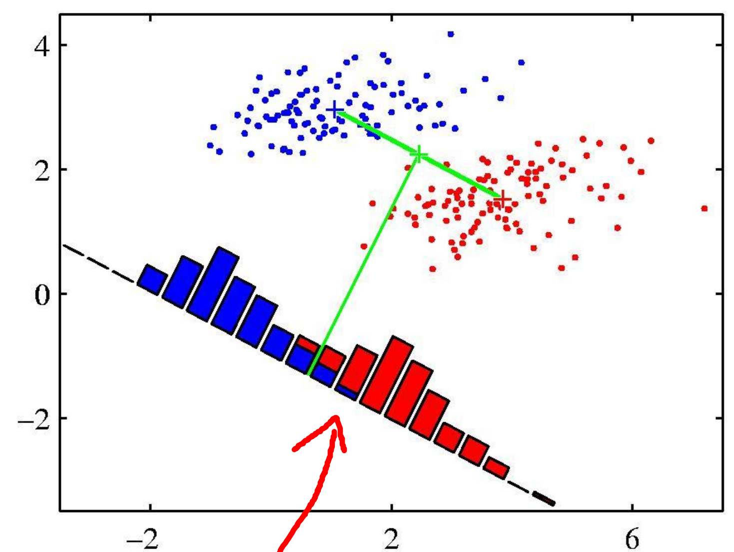
Linear Discriminant Analysis

- linear classification
  - from training data $\{a_i, b_i\}_{i=1}^m, a_i \in \mathrm{R}^n, b_i \in \{-1, +1\}$,
  - establish a linear model $x^\top a$ to classify the two data sets
- Fisher Discriminant
  - push away the projected centers

$$\frac{1}{m_+}\sum_{i:b_i=1} x^\top a_i - \frac{1}{m_-}\sum_{i:b_i=-1} x^T a_i$$

  - from linearity

$$\max_x \; x^\top(\overline{a_+} - \overline{a_-})$$



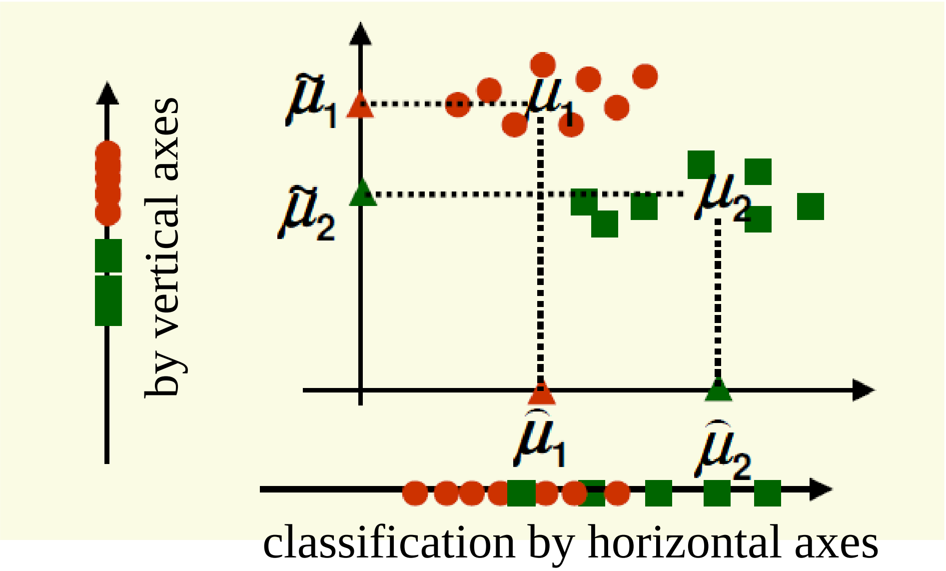
Linear Discriminant Analysis

- linear classification
  - from training data $\{a_i, b_i\}_{i=1}^m, a_i \in \mathrm{R}^n, b_i \in \{-1, +1\}$,
  - establish a linear model $x^\top a$ to classify the two data sets
- Fisher Discriminant
  - push away the projected centers

$$\frac{1}{m_+}\sum_{i:b_i=1} x^\top a_i - \frac{1}{m_-}\sum_{i:b_i=-1} x^T a_i$$

  - from linearity

$$\max_x \; x^\top(\overline{a_+} - \overline{a_-})$$



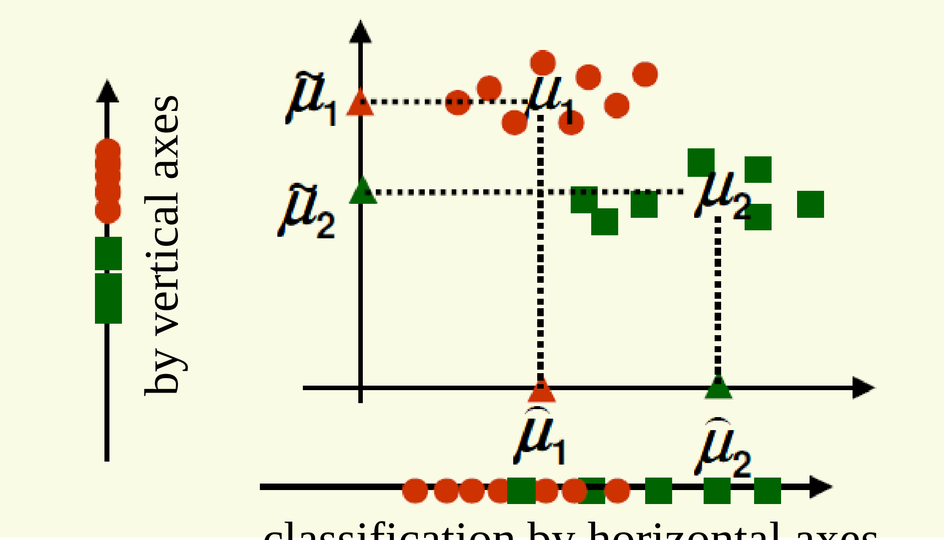
Linear Discriminant Analysis

- within-class scatter (variance)

$$S_+^2 = \sum_{i:b_i=1} (x^\top a_i - x^\top \overline{a_+})^2 \qquad S_-^2 = \sum_{i:b_i=-1} (x^\top a_i - x^\top \overline{a_-})^2$$

- Fisher Discriminant

$$\max_x \frac{(x^\top \overline{a_+} - x^\top \overline{a_-})^2}{\underline{S_+^2 + S_-^2}}$$



classification by horizontal axes

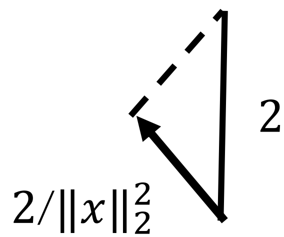
Canonical Correlation Analysis

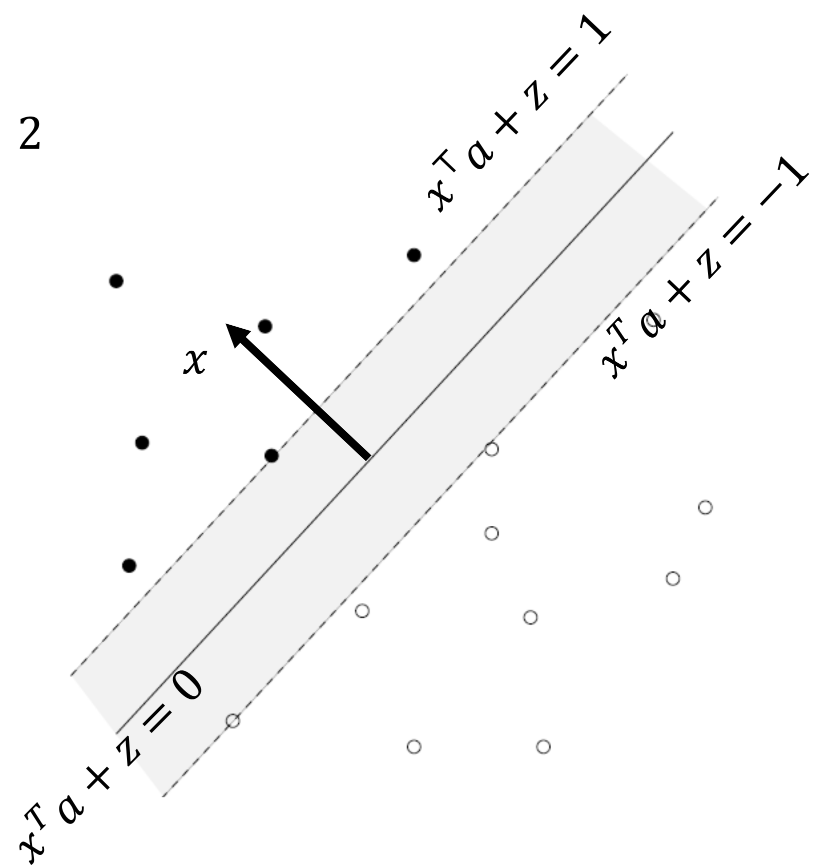
- if we have $a_i \in \mathbf{R}^n$ and $b_i \in \mathbf{R}^m$, two measurements for the same object
- find the common latent variable, i.e., to project them on the same space
- maximize the correlation between $x^\top a_i$ and $y^\top b_i$
- normalized by the variance
- CCA

$$\max_{x,y} \frac{x^\top \mathrm{cov}(A,B) y}{\sqrt{x^\top \mathrm{cov}(A,A) x}\sqrt{y^\top \mathrm{cov}(A,A) y}}$$

Margin-based Classification

- find a hyperplane to separate two classes
  - no misclassification
  - large margin





- optimization problem

$$
\begin{aligned}
\max_{x,z} \quad & 2/\|x\|_2 \\
\text{s.t.} \quad & b_i(x^\top a_i + z) \geq 1
\end{aligned}
$$

Support Vector Machine

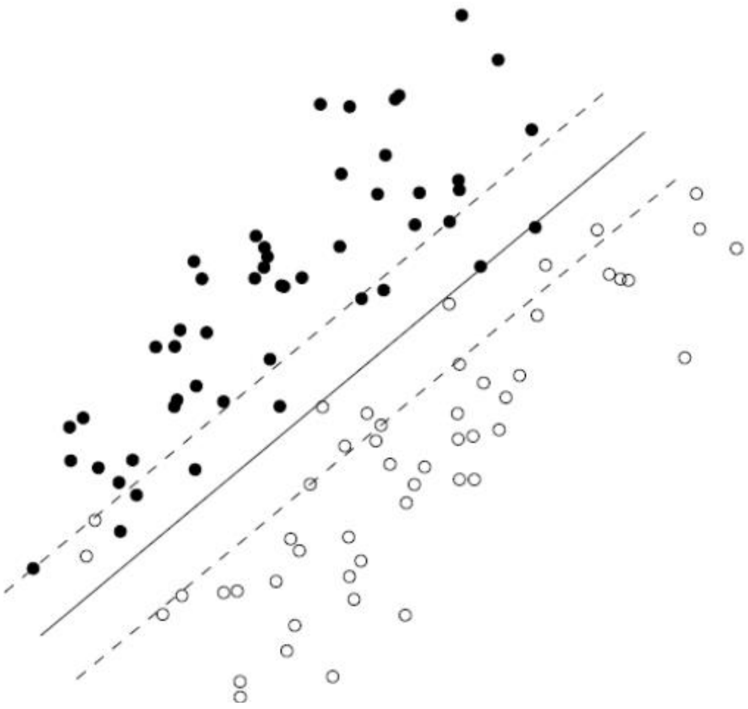
- linear SVM

$$\begin{aligned} \min_{x,z} \quad & \|x\|_2^2 \\ \text{s.t.} \quad & b_i(x^\top b_i + z) \geq 1 \end{aligned}$$

- if it is not linearly separable,

  the above problem is *infeasible*

$$\begin{aligned} \min_{x,z} \quad & \frac{1}{2}\|x\|_2^2 + C\sum s_i \\ \text{s.t.} \quad & b_i(x^\top b_i + z) \geq 1 - s_i \\ & s_i \geq 0 \end{aligned}$$



Hinge Loss

- loss function + regularization term

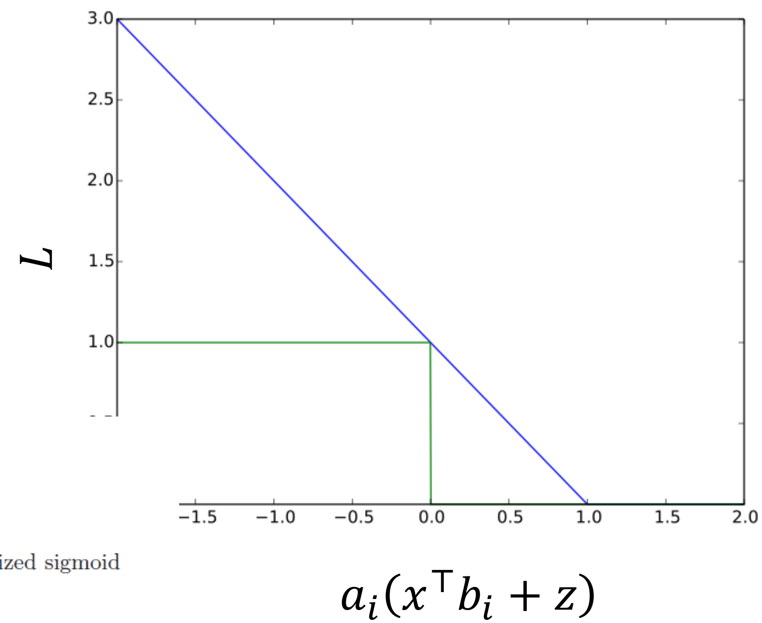
$$\min_{x,z} \quad \frac{1}{2}\|x\|_2^2 + C\sum_i L(b_i(x^\top a_i + z))$$

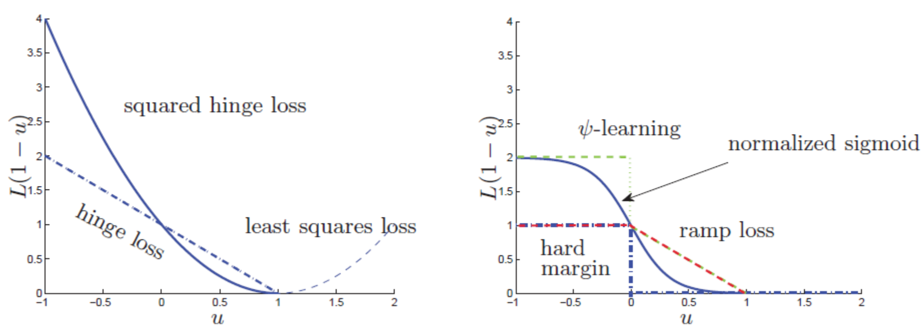
- hinge loss

$$L(u) = \max\{0, 1-u\}$$

- the convex approximation

  for misclassification loss



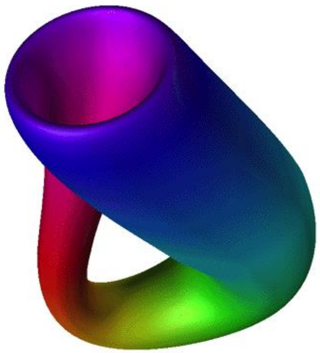


(a) (b)

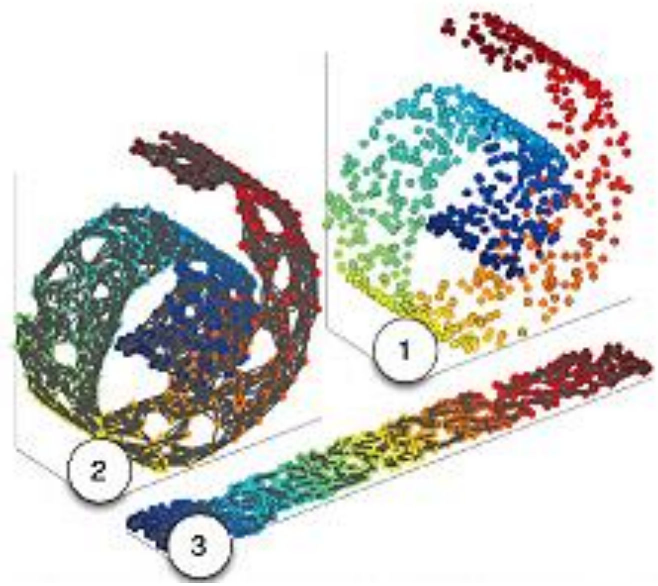
目录 Contents

Dimensionality reduction

- Find a low-dimensional surface to cover the training samples as well as possible
  - linear surface: subspace
  - nonlinear surface: manifold learning
  - one nature of big data

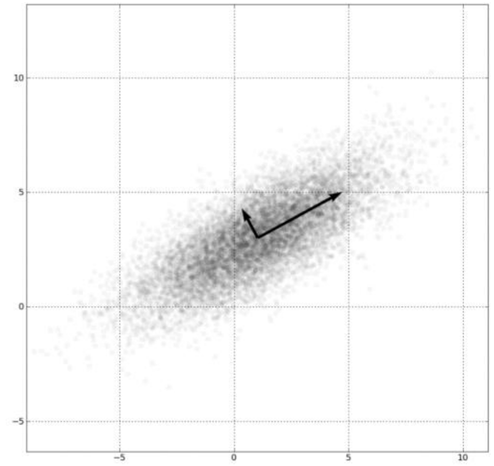


- An Example: images of human faces can be seen as vectors in a very high dimensional space. Actual faces reside in a small subspace of that large space.



Principle component analysis

- for a given dataset in $\mathbf{R}^n$, find a subspace to contain the information of the dataset as many as possible
- information → distinguish data → the variance
- find a direction that maximizes the data's variance



$$\max_{x^T x=1} \quad x^T C x$$

$$C = \frac{1}{n-1} \boxed{A^T A}$$

it should be zero-mean,
otherwise, the covariance
matrix will be ?

上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

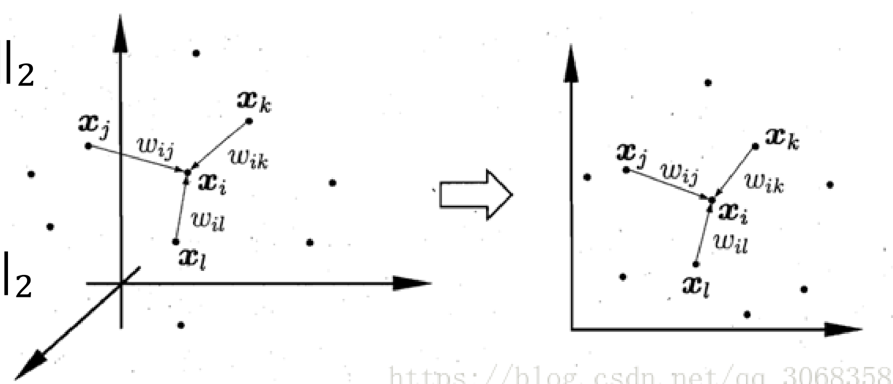
Locally linear embedding

- another criteria for dimension reduction, local preservation
  - if a_i and a_j are similar, so their projections in lower space
- Locally Linear Embedding (LLE): a point could be represented by its neighborhood and the relationship should be preserved in reduction
  - find the neighborhood
  - find the relationship $a_0 = \sum x_i a_i$

$$\min_{x} \|a_0 - \sum x_i a_i\|_2$$

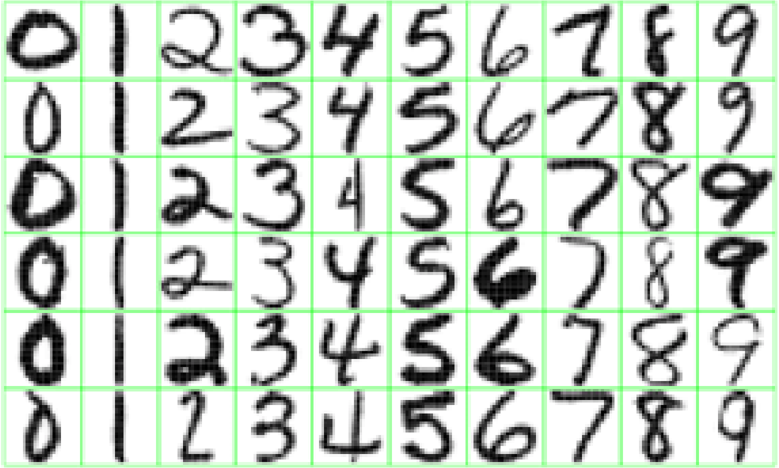
  - keep the relationship in reduction

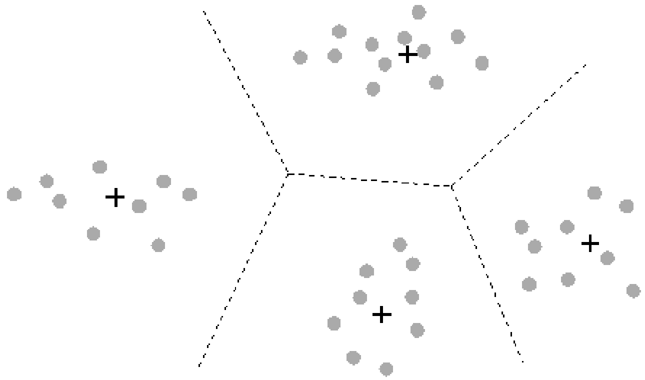
$$\min_{b} \|b_0 - \sum x_i b_i\|_2$$



Clustering

- assignment of a set of observations into subsets so that observations in the same subset are *similar*





$$\min_{x_i \in \{1,2\ldots,k\}} \sum_{k} \sum_{x_i, x_j = k} d(a_i, a_j)$$

Distance/dissimilarity

- distance: (dis)similarity measures for two samples
  - identity error

$$d(a_i, a_j) = I(a_i \neq a_j)$$

  - squared distance

$$d(a_i, a_j) = \|a_i - a_j\|_2^2$$

  - lq distance

$$d(a_i, a_j) = \|a_i - a_j\|_q$$

  - Canberra distance

$$d(a_i, a_j) = \sum_d \frac{|a_{id} - a_{jd}|}{|a_{id} + a_{jd}|}$$

Distance/dissimilarity

- distance: (dis)similarity measures for two samples
  - identity error

$$d(a_i, a_j) = I(a_i \neq a_j)$$

  - squared distance

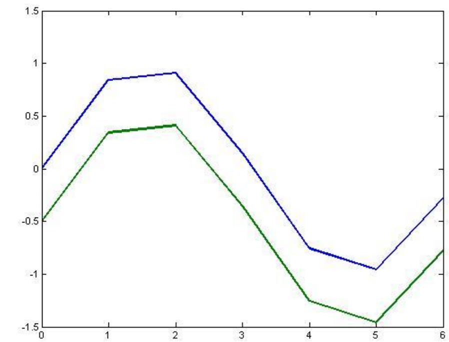
$$d(a_i, a_j) = \|a_i - a_j\|_2^2$$

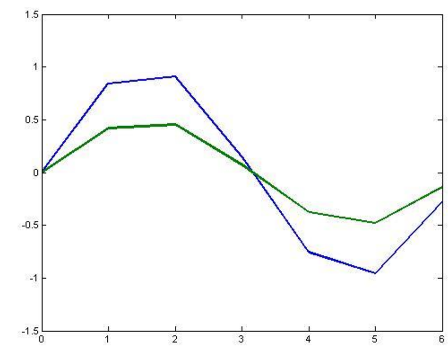
  - lq distance

$$d(a_i, a_j) = \|a_i - a_j\|_q$$

  - Canberra distance

$$d(a_i, a_j) = \sum_d \frac{|a_{id} - a_{jd}|}{|a_{id} + a_{jd}|}$$





- Person Linear Correlation

$$\rho(x, y) = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2} \sqrt{\sum (y_i - \bar{y})^2}}$$

Distance/dissimilarity

- distance: (dis)similarity measures for two samples
  - identity error

$$d(a_i, a_j) = I(a_i \neq a_j)$$

  - squared distance

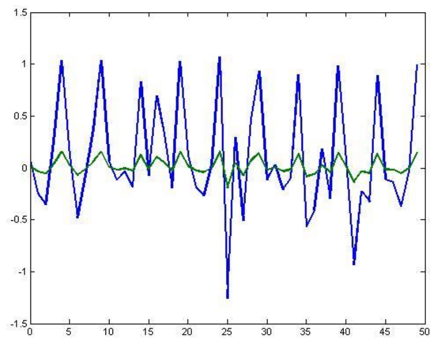
$$d(a_i, a_j) = \|a_i - a_j\|_2^2$$

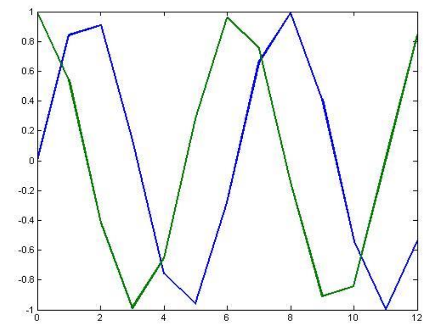
  - lq distance

$$d(a_i, a_j) = \|a_i - a_j\|_q$$

  - Canberra distance

$$d(a_i, a_j) = \sum_d \frac{|a_{id} - a_{jd}|}{|a_{id} + a_{jd}|}$$





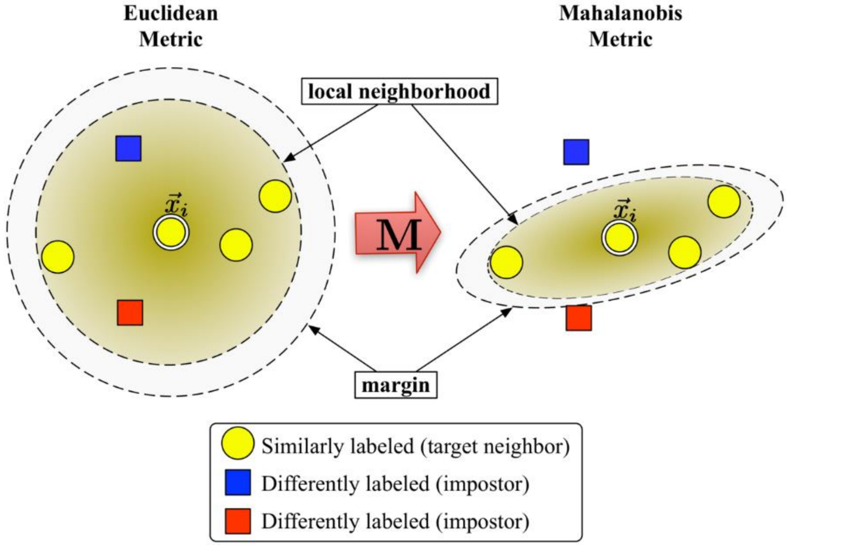
- Person Linear Correlation

$$\rho(x, y) = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2}\sqrt{\sum (y_i - \bar{y})^2}}$$

Metric Learning

- distance/similarity/dissimilarity/metric is very important
  - non-negative
  - identity
  - symmetry
  - subadditivity (triangle inequality)
- Mahalanobis distance

https://en.wikipedia.org/wiki/Large_margin_nearest_neighbor



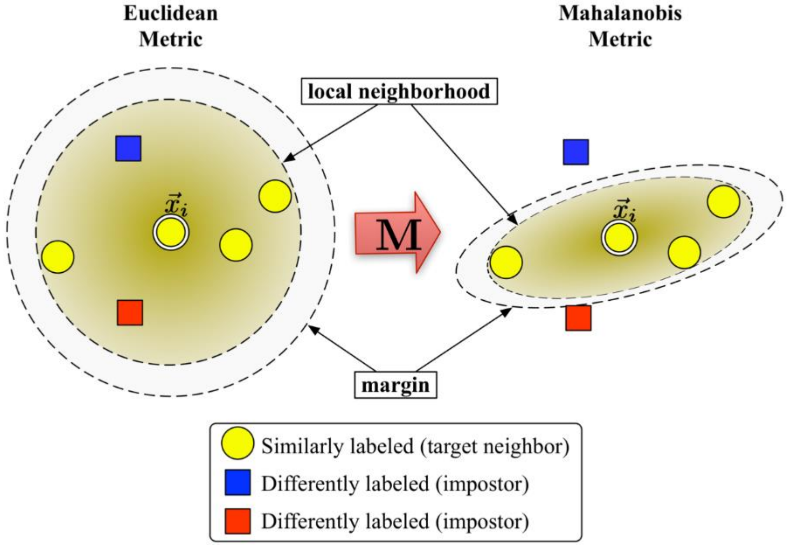
$$d_M(a,b) = \sqrt{(La-Lb)^\top(La-Lb)} = \sqrt{(a-b)^\top L^\top L(a-b)} = \sqrt{(a-b)^\top M(a-b)}$$

$$\min_M \sum_{i,j\in N_i} d(a_i,a_j) \ \text{s.t.} \begin{cases} d(a_i,a_j) \le d(a_i,a_l), \forall j \in N_i, l \notin N_i, \forall i \\ M \succcurlyeq 0 \end{cases}$$

Metric Learning

- distance/similarity/dissimilarity/metric is very important
  - non-negative
  - identity
  - symmetry
  - subadditivity (triangle inequality)
- Mahalanobis distance

https://en.wikipedia.org/wiki/Large_margin_nearest_neighbor



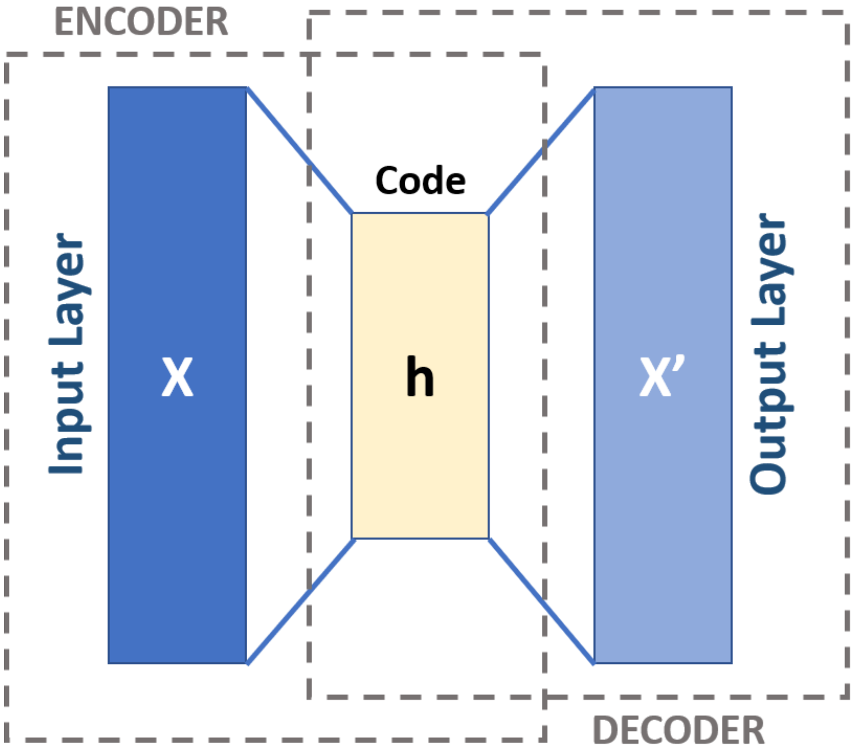
$$d_M(a,b)=\sqrt{(La-Lb)^\top(La-Lb)}=\sqrt{(a-b)^\top L^\top L(a-b)}=\sqrt{(a-b)^\top M(a-b)}$$

$$\min_M \sum_{i,j\in N_i} d(a_i,a_j)+\lambda\sum_{i,j,l}\xi_{ijl} \quad \text{s.t.}\begin{cases} d(a_i,a_j)+1\le d(a_i,a_l)+\xi_{ijl}, \forall j\in N_i, l\notin N_i, \forall i \\ \xi_{ijl}\ge 0 \\ M\succcurlyeq 0\end{cases}$$

Autoencoder

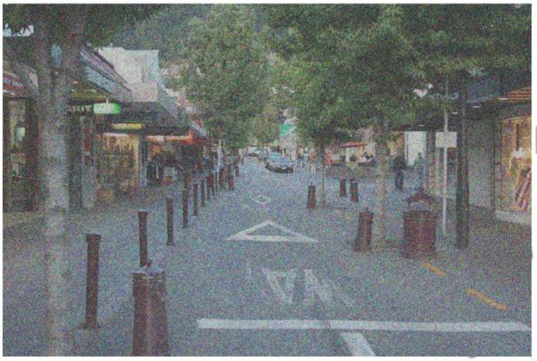
- autoencoder is to learn a representation (encoding) for a set of data
- autoencoder training
  - encoder ϕ
  - decoder ψ

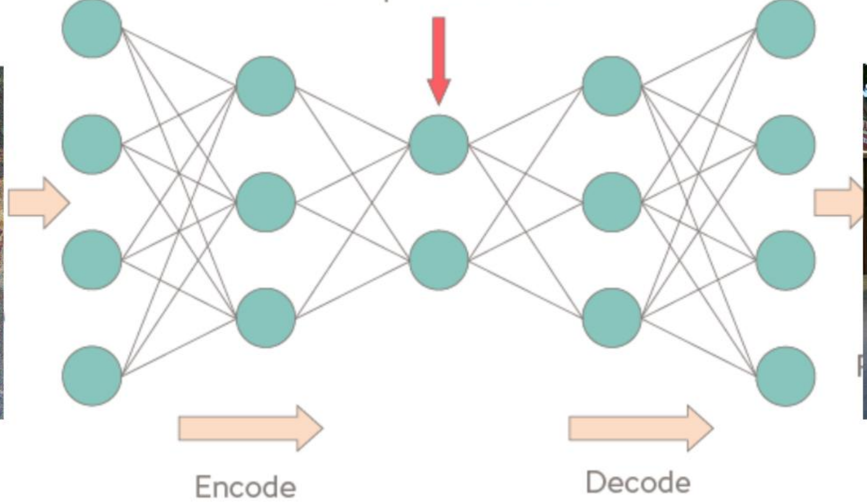
$$\min_{\phi,\psi} \quad \|X - \psi(\phi(X))\|$$

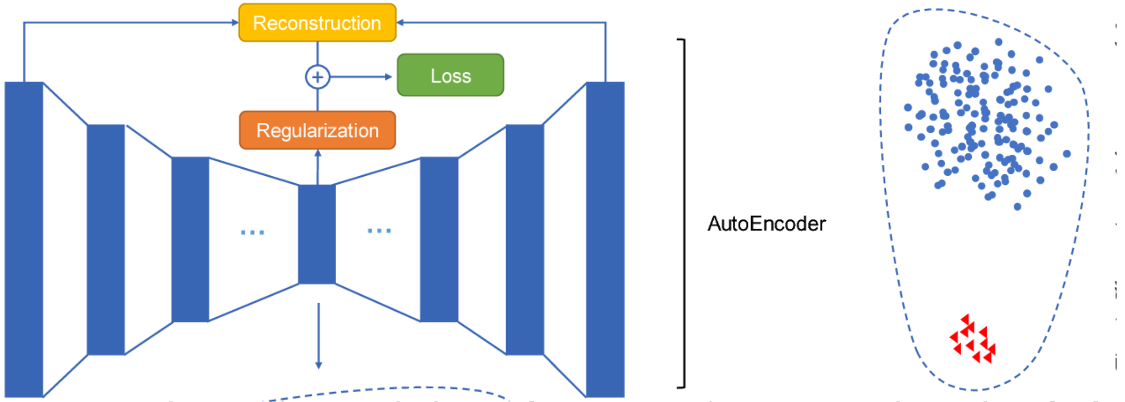


https://en.wikipedia.org/wiki/Autoencoder

Autoencoder







- sparse autoencoder

  denoising autoencoder

  anomaly detection

https://towardsdatascience.com/auto-encoder-what-is-it-and-what-is-it-used-for-part-1-3e5c6f017726

Semi-supervised Learning

- we have only labels for a part of data
  - supervised learning: to approach labels
  - unsupervised learning: to approach prior knowledge
- prior knowledge/assumptions
  - continuity assumption: close points are more likely to have the same label
  - cluster assumption: points in a cluster are more likely to have the same label
  - manifold assumption: data lie on a manifold

$$\min_f \lambda_1 \sum_{\text{labelled}} \max\{1 - b_i f(a_i), 0\} + \lambda_2 \sum_{\text{unlabelled}} (1 - |f(a_i)|)^2 + \|f\|$$

label assgiment

Semi-supervised Learning

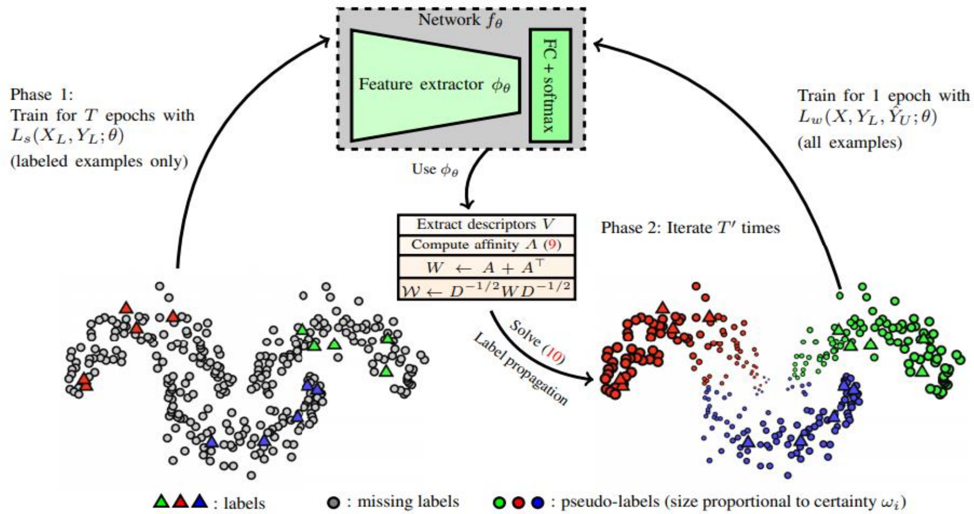
- we have only labels for a part of data
  - supervised learning: to approach labels
  - unsupervised learning: to approach prior knowledge
- prior knowledge/assumptions
  - continuity assumption: close points are more likely to have the same label
  - cluster assumption: points in a cluster are more likely to have the same label
  - manifold assumption: data lie on a manifold

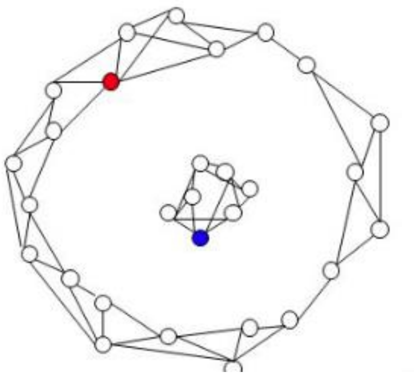
$$\min_f \lambda \sum_{\text{labelled}} \max\{1 - b_i f(a_i), 0\} + \sum_{\text{all data}} w_{ij} \left(f(a_i) - f(a_j)\right)^2$$

Laplacian SVM

Semi-supervised Learning

- we have only labels for a part of data
  - supervised learning: to approach labels
  - unsupervised learning: to approach prior knowledge
- prior knowledge/assumptions
  - Graph (given or by local similarity) transduction

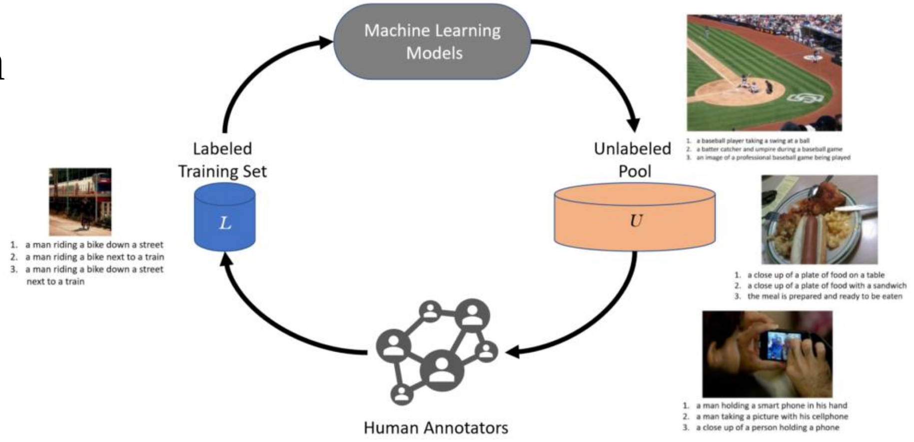




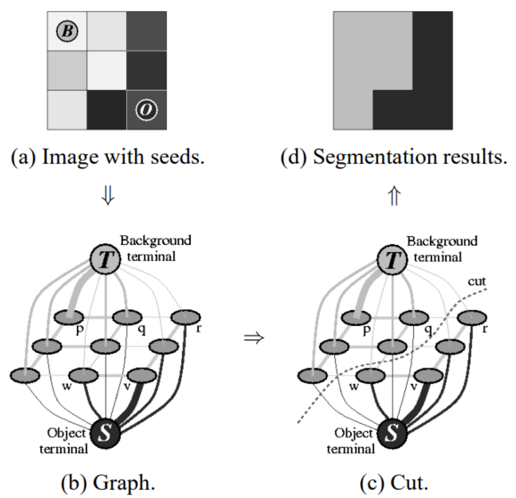
A. Iscen, et al. Label Propagation for Deep Semi-supervised Learning

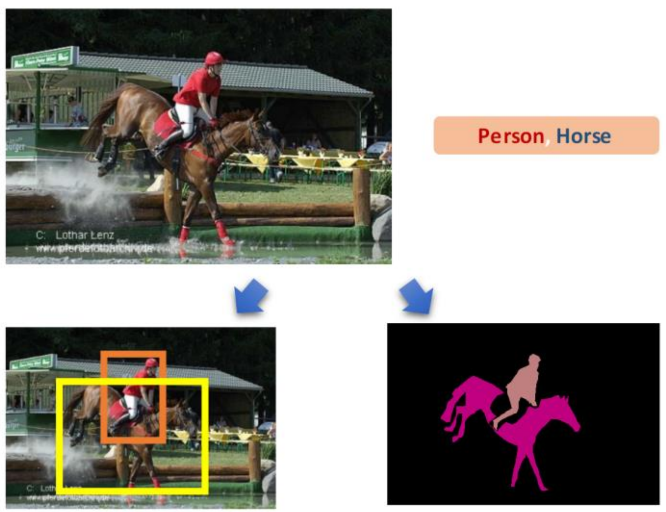
Weakly-Supervised Learning

- learning from incomplete supervision
  - semi-supervised learning
  - active learning



- learning from inexact supervision
  - Graph-based

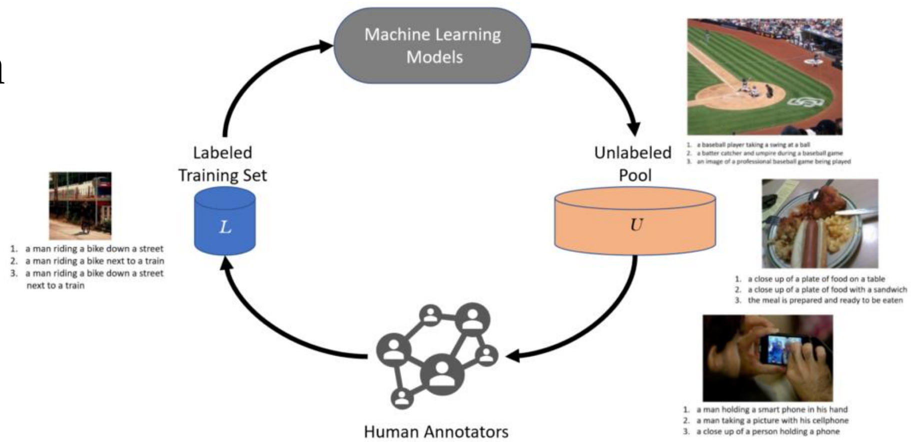


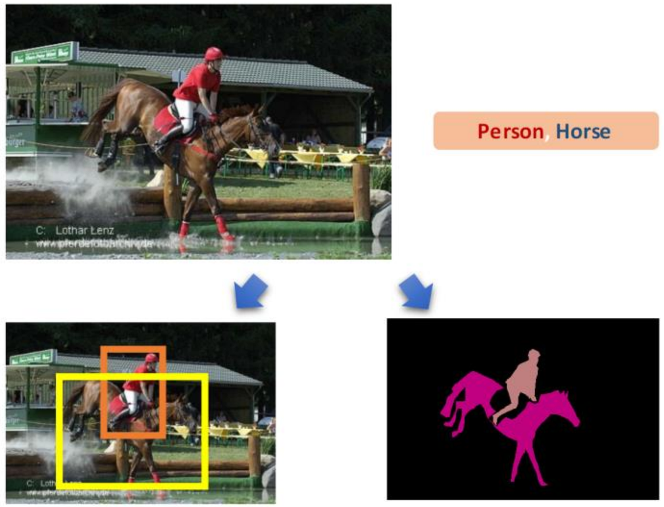


Z.-H. Zhou, A brief introduction to weakly supervised learning." National Science Review 5.1 (2017): 44-53.

Weakly-Supervised Learning

- learning from incomplete supervision
  - semi-supervised learning
  - active learning
- learning from inexact supervision
- learning from inaccurate supervision
  - noise in label





Z.-H. Zhou, A brief introduction to weakly supervised learning." National Science Review 5.1 (2017): 44-53.

目录 Contents

Maximum Likelihood Estimation

- data come from a specific distribution with unknown parameters
- for example $X \sim N(\mu, \sigma^2)$ $\quad p(X = x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$

i.i.d.

$$p(x_1, x_2, \dots, x_m | \mu, \sigma) = \prod_{i=1}^{m} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x_i-\mu)^2}{2\sigma^2}}$$

$$\log p(x_1, x_2, \dots, x_m | \mu, \sigma) = m\log\sqrt{2\pi} + m\log\sigma + \sum_{i=1}^{m} \frac{(x_i-\mu)^2}{2\sigma^2}$$

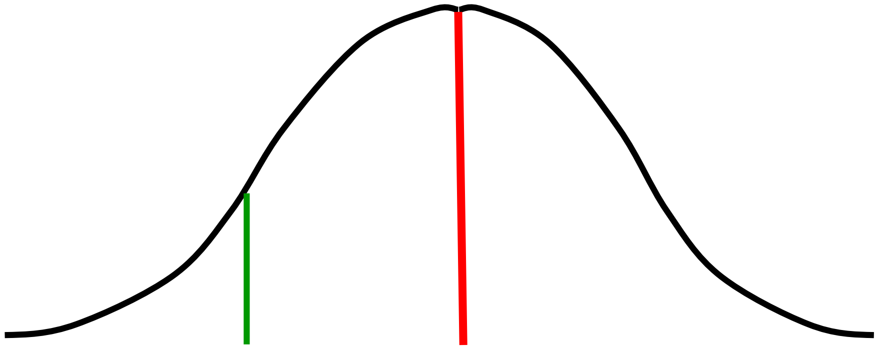
MLE

$$\frac{d\log p}{d\mu} = \sum_{i=1}^{m} (x_i - \mu) = 0 \qquad \frac{d\log p}{d\sigma} = \frac{m}{\sigma} - \frac{1}{\sigma^3}\sum_{i=1}^{m} (x_i - \mu)^2 = 0$$

$$\mu^* = \frac{1}{m}\sum_{i=1}^{m} x_i \qquad \sigma^* = \sqrt{\frac{1}{m}\sum_{i=1}^{m} (x_i - \mu^*)^2}$$

MLE for OLS

- Minimizing the squared residuals is equivalent to maximizing the log probability of the correct answer under a Gaussian centered at zero



b = the correct answer

f = model's estimate of most probable value

$$f_i = f(a_i, \boldsymbol{x})$$

$$p(b_i|f_i) = p(f_i + noise = b_i | x_i, \mathrm{w}) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(b_i - f_i)^2 / 2\sigma^2}$$

$$-\log p(b_i|f_i) = \log\sqrt{2\pi} + \log\sigma + \frac{(b_i - f_i)^2}{2\sigma^2}$$

Logistic regression

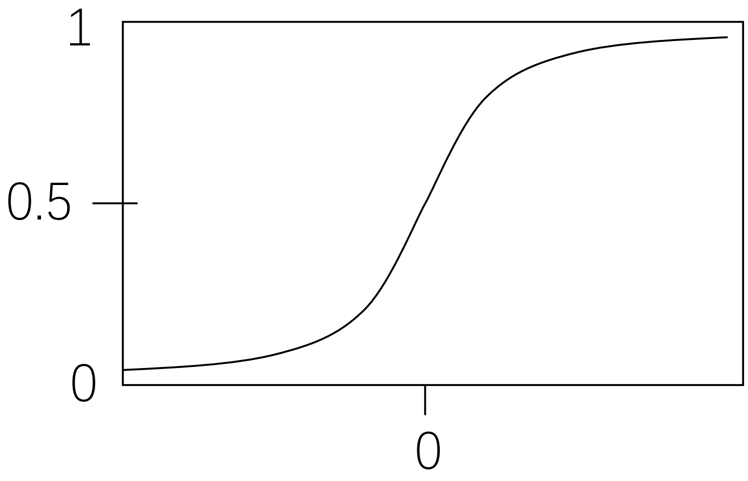
- For a binary classification problem, the conditional probabilities of two classes (now, assume the label is 0 and 1) are

$$p(b=1|x) = \rho(x^\top b), \qquad p(b=0|x) = 1-\rho(x^\top b)$$

- A popular used assumption is

$$\rho(x^\top b) = \frac{1}{1+\exp(-x^\top b)}$$

$$1-\rho(x^\top b) = \frac{1}{1+\exp(x^\top b)}$$



Logistic/sigmoid function

Logistic regression

- Therefore, log likelihood with given $\{a_i, b_i\}_{i=1}^{m}$ is

$$J(x) = \sum_i b_i \log(\rho(x^\top a_i)) + (1 - b_i)\log(1 - \rho(x^\top a_i))$$

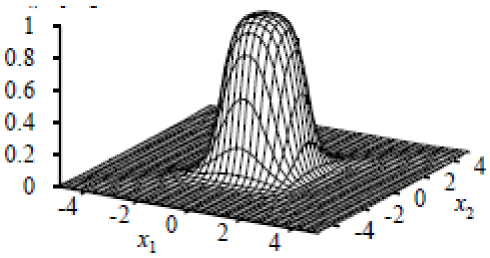
$$= \sum_i b_i(x^\top a_i) - \log(1 + \exp(x^\top a_i))$$

- The above is a concave function on x, and its maximization (maximum log likelihood) is convex and easy to solve.

$$\min_x \; J(x) = -\sum_i b_i(x^\top a_i) + \log(1 + \exp(x^\top a_i))$$

Mixed Gaussian and EM

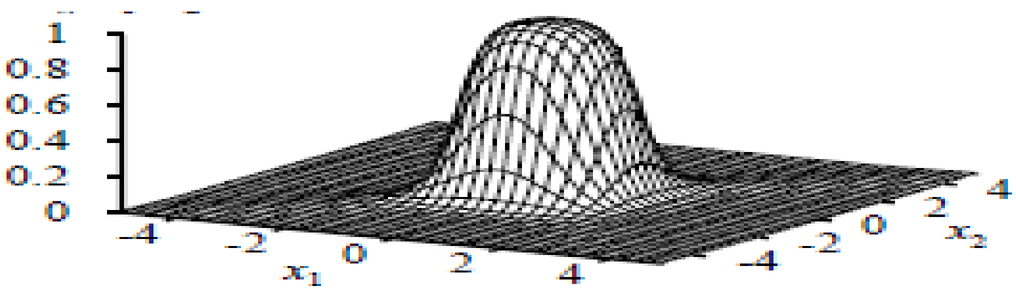
- data come from several classes, each of which follows a specific but an unknown distribution, e.g, Gaussian mixture model

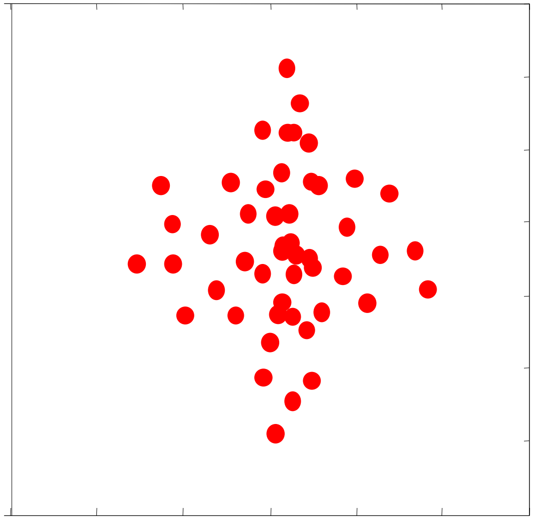


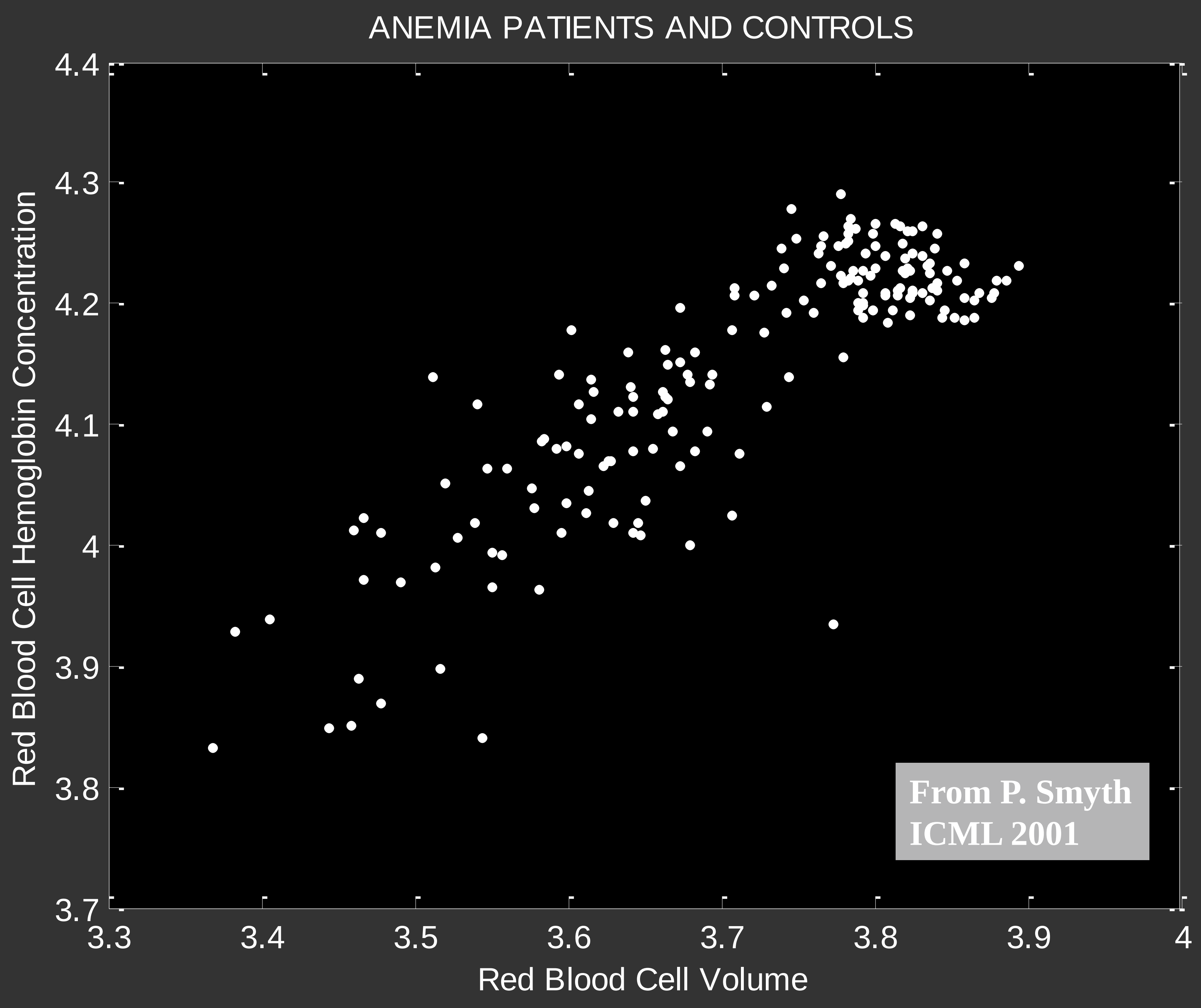
- the clustering can be molded

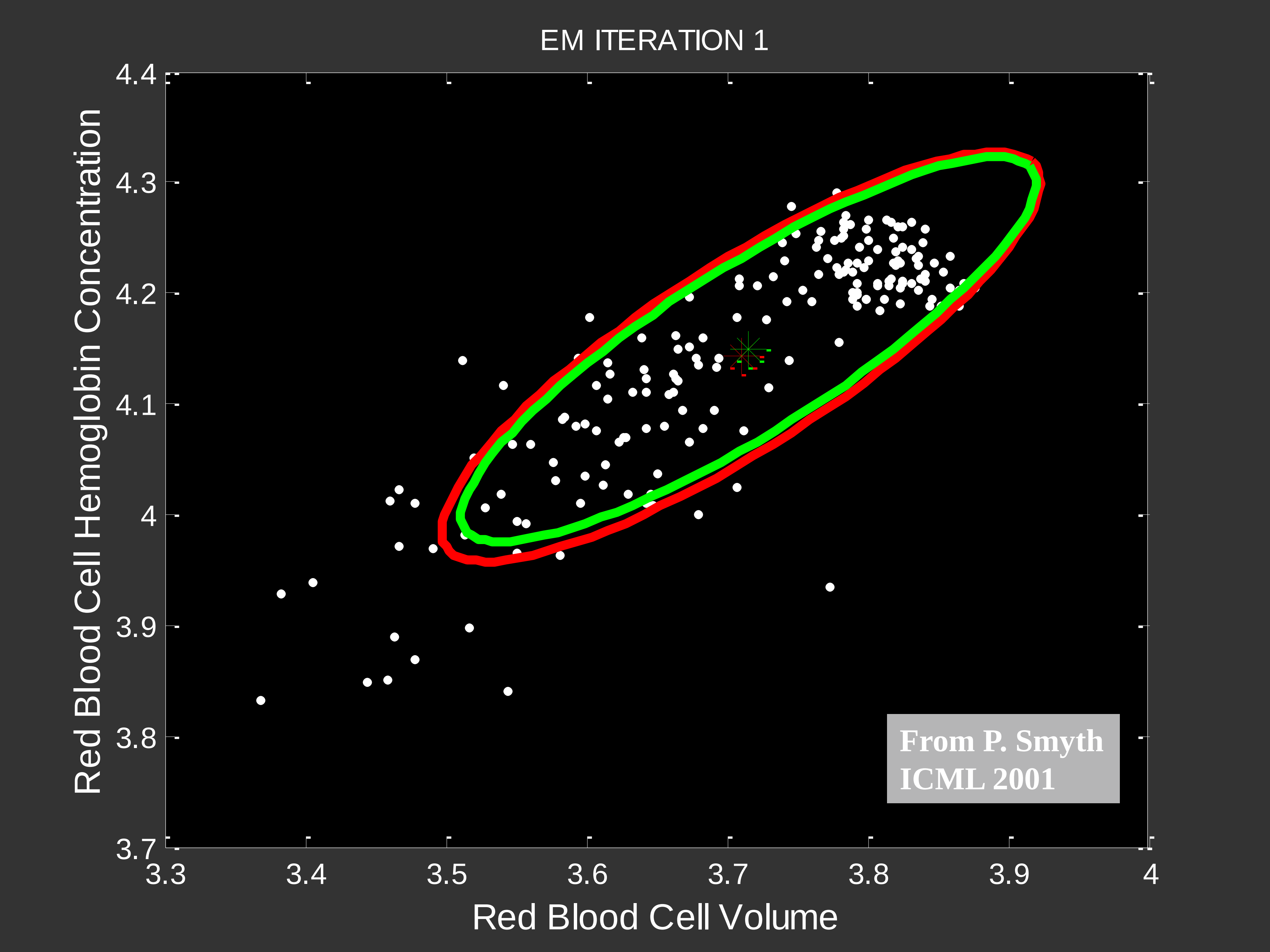
$$\max \quad \Sigma_k \Sigma_{i \in C_k} \log p(a_i | \mu_k, \sigma_k)$$

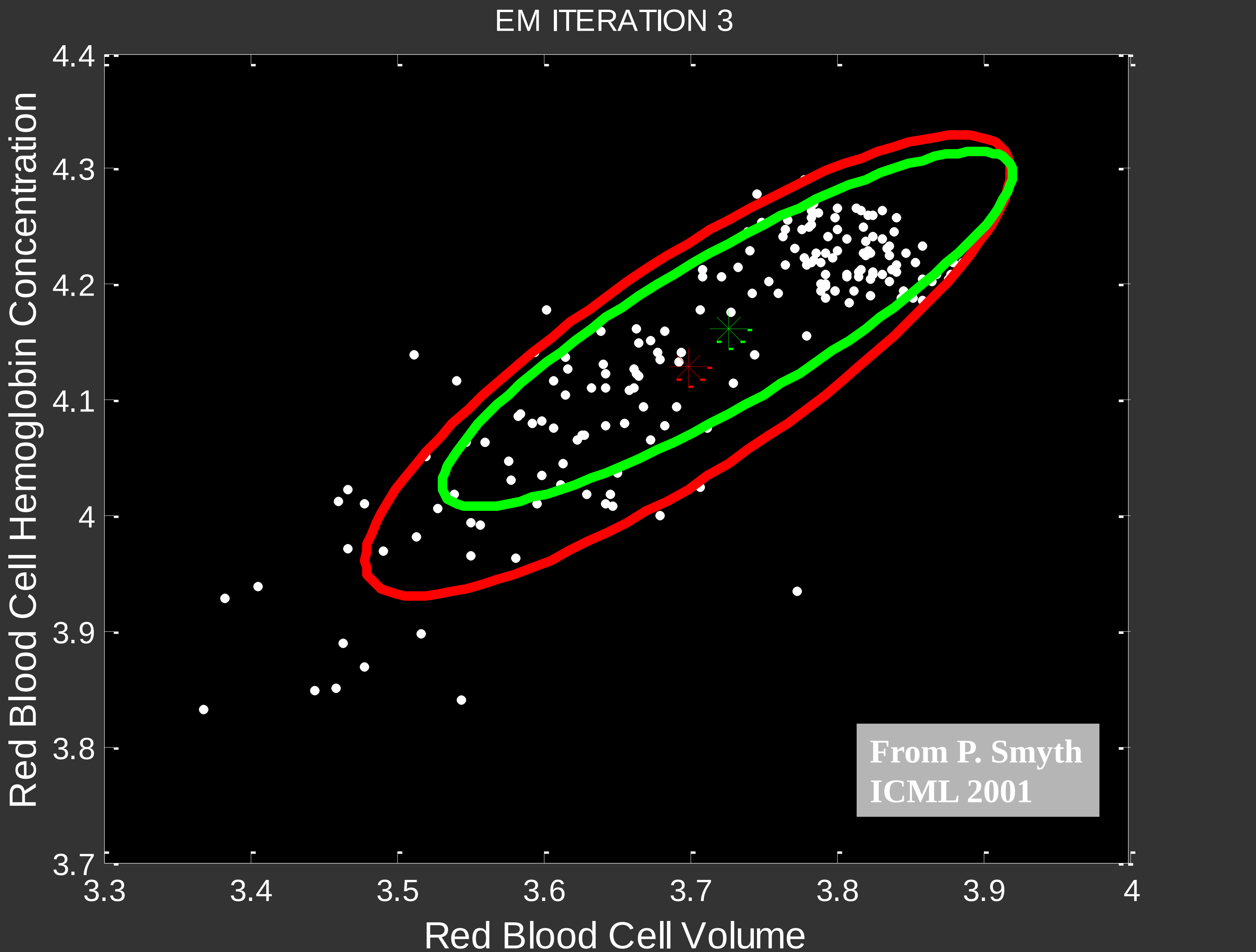
  - E: assign the observations to classes
  - M: find the distributions based on assigned data

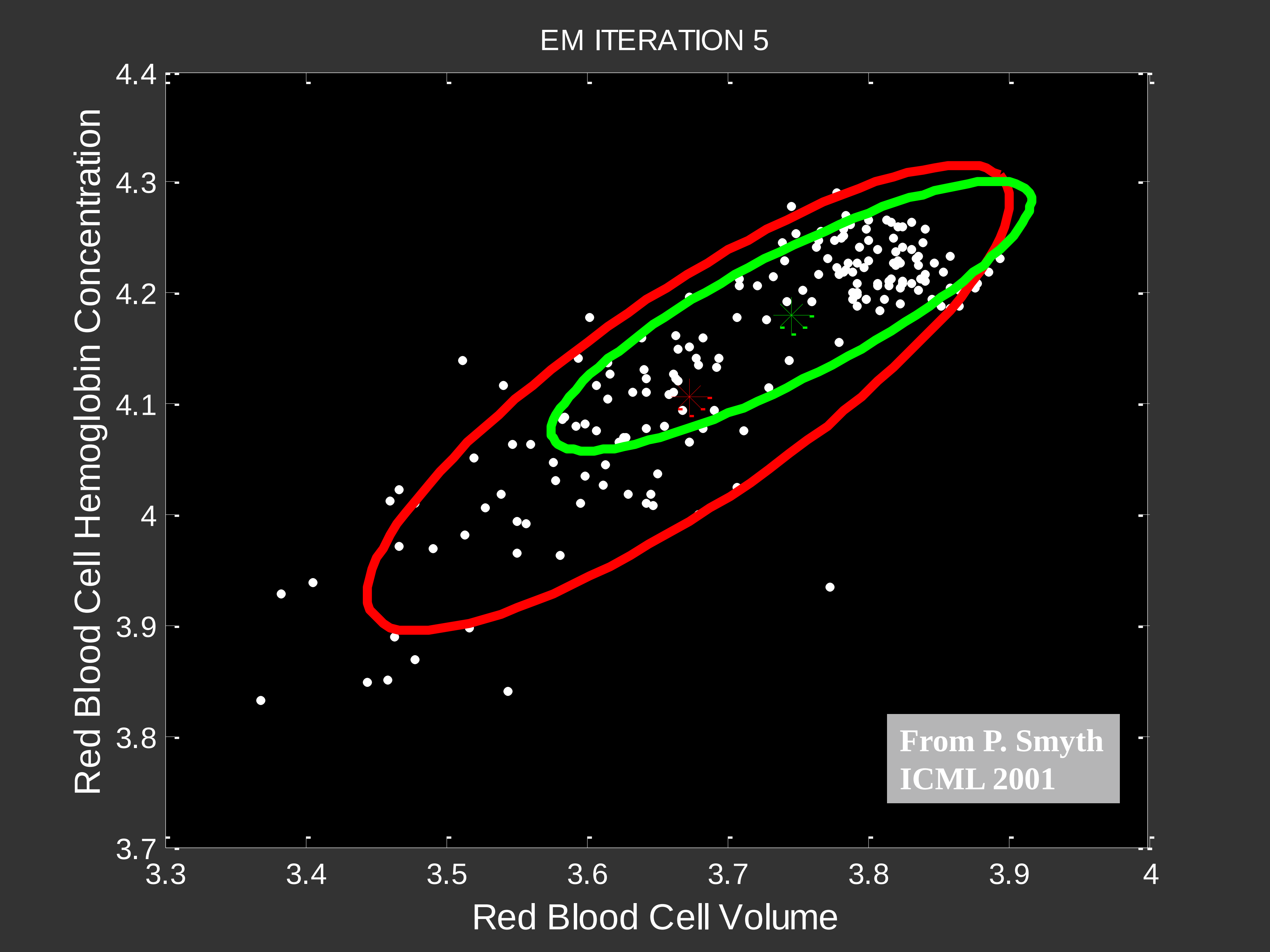


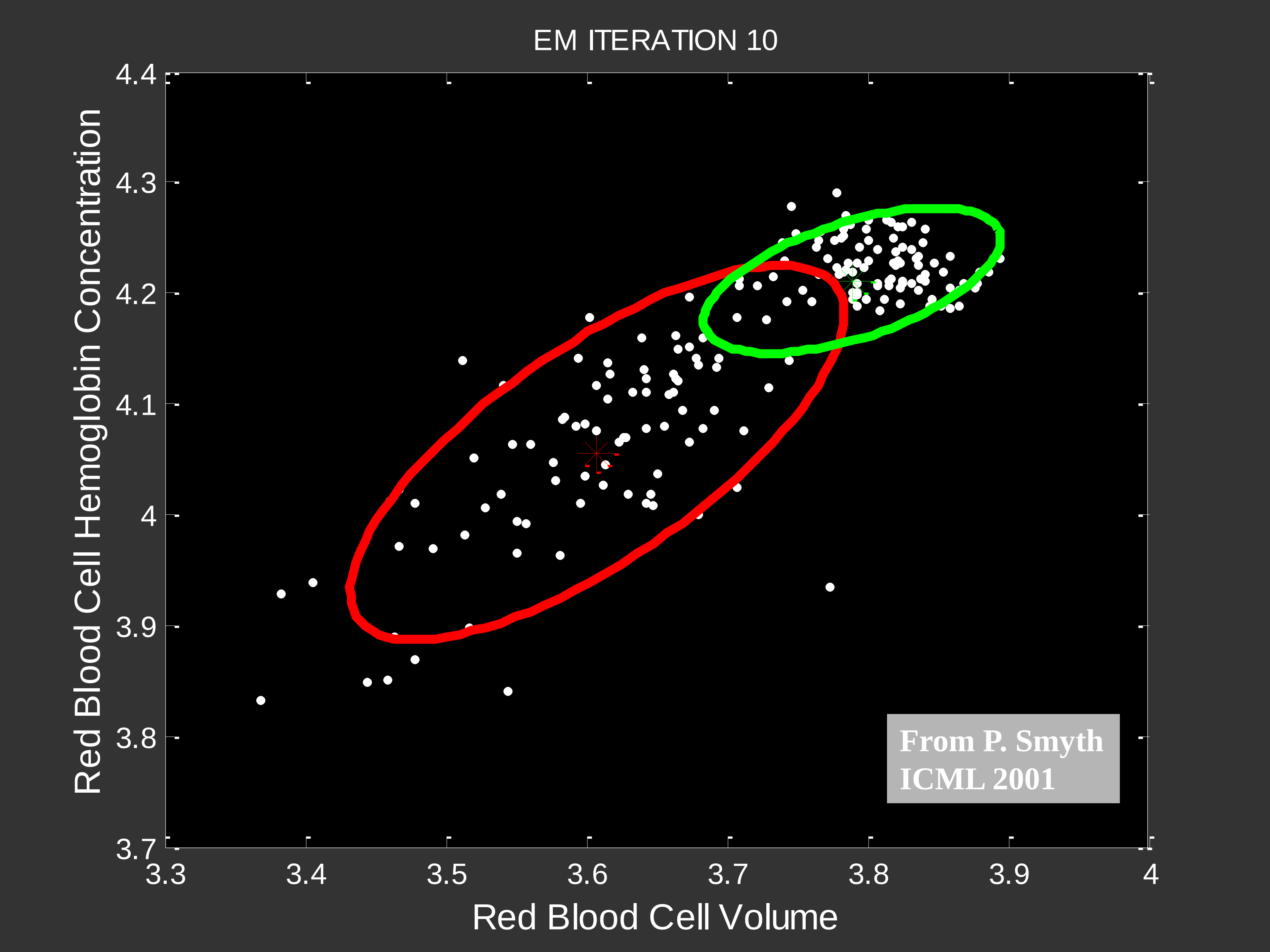


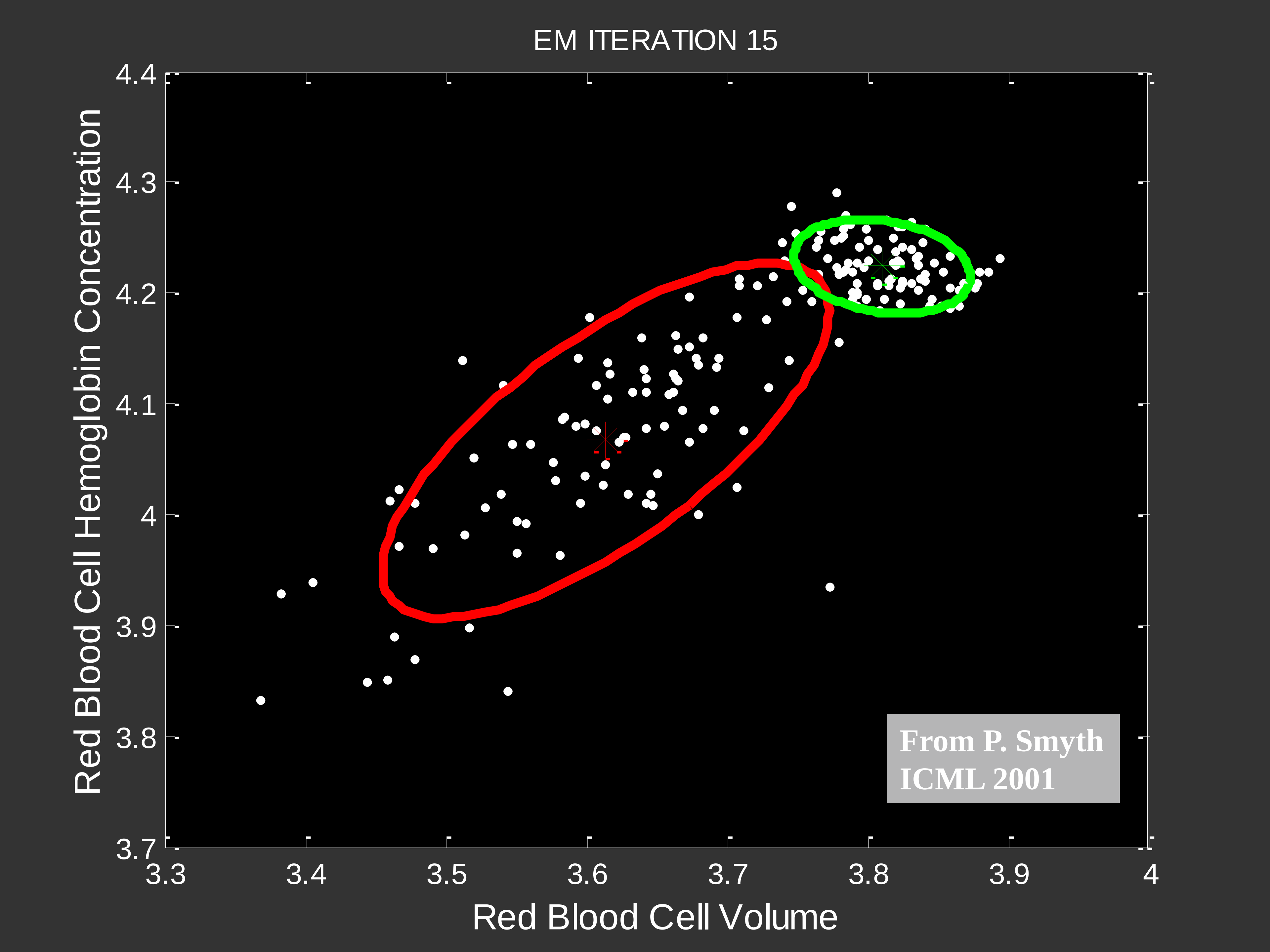

ANEMIA PATIENTS AND CONTROLS
4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
Red Blood Cell Hemoglobin Concentration
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4
Red Blood Cell Volume
From P. Smyth
ICML 2001

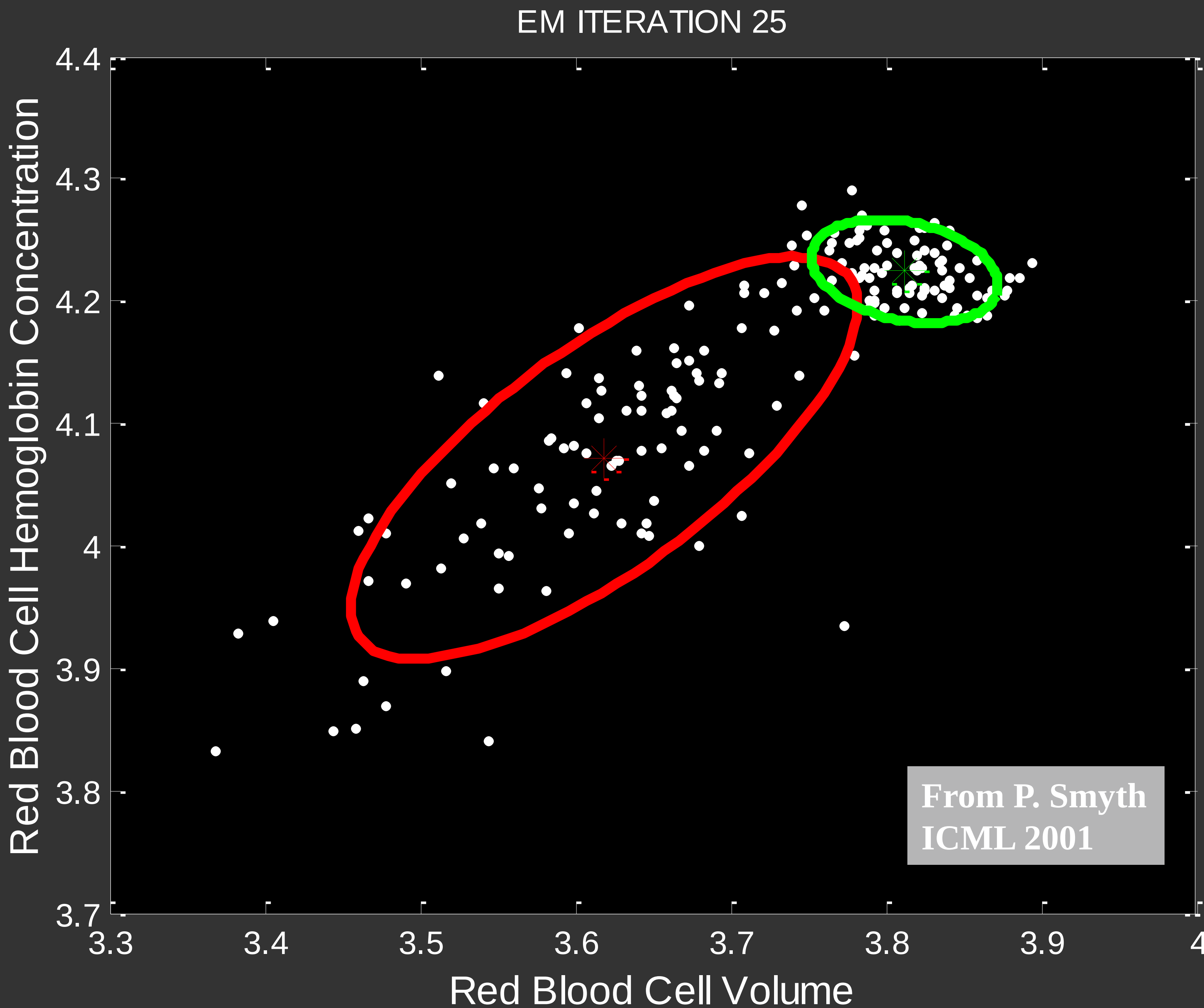

EM ITERATION 1
Red Blood Cell Hemoglobin Concentration
Red Blood Cell Volume
4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4
From P. Smyth
ICML 2001


EM ITERATION 3
Red Blood Cell Hemoglobin Concentration
4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4
Red Blood Cell Volume
From P. Smyth
ICML 2001


EM ITERATION 5
4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4
Red Blood Cell Hemoglobin Concentration
Red Blood Cell Volume
From P. Smyth
ICML 2001


EM ITERATION 10
Red Blood Cell Hemoglobin Concentration
4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4
Red Blood Cell Volume
From P. Smyth
ICML 2001


EM ITERATION 15
Red Blood Cell Hemoglobin Concentration
Red Blood Cell Volume
4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4
From P. Smyth
ICML 2001


EM ITERATION 25
Red Blood Cell Hemoglobin Concentration
Red Blood Cell Volume
4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4
From P. Smyth
ICML 2001

Maximum A Posteriori

- MLE is to estimate parameters x from observations A

$$x_{\mathrm{MLE}} = \operatorname*{argmax}_{x} p(A|x) = \operatorname*{argmax}_{x} \Pi_i p(a_i|x)$$

- Bayes' rule

$$p(x|A) = \frac{p(A|x)p(x)}{p(A)} \propto p(A|x)p(x)$$

- MAP

$$x_{\mathrm{MAP}} = \operatorname*{argmax}_{x} p(A|x)\, p(x)$$

  - it can be used when we have prior-knowledge on x
  - e.g., x is in a Laplacian distribution then it is sparse

data $A \sim N(\mu, \sigma^2)$
knowing $\mu \in N(\mu_0, \sigma_\mu^2)$

$$p(\mu)p(A|\mu) = \frac{1}{\sigma_\mu\sqrt{2\pi}} e^{-\frac{(\mu-\mu_0)^2}{2\sigma_\mu^2}} \prod \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(a_i-\mu)^2}{2\sigma^2}}$$

$$\mu_{\mathrm{MAP}} = \min_{\mu} \frac{(\mu-\mu_0)^2}{2\sigma_\mu^2} + \sum \frac{(a_i-\mu)^2}{2\sigma^2}$$

Neighbor embedding

- another criteria for dimension reduction, local preservation
  - if x_i and x_j are similar, so their projections in lower space
  - similarly, isotonic property: if $x_i > x_j$, so their projections in lower space
- Stochastic Neighbor Embedding (SNE) uses conditional probability to measure the distance between two points

Gaussian distribution centered at x_i

$$p_{x_j|x_i} = \frac{\exp(-\|x_i - x_j\|^2/2\sigma_i^2)}{\sum_{k\neq i}\exp(-\|x_i - x_k\|^2/2\sigma_i^2)}$$

$$q_{y_j|y_i} = \frac{\exp(-\|y_i - y_j\|^2/2\sigma_i^2)}{\sum_{k\neq i}\exp(-\|y_i - y_k\|^2/2\sigma_i^2)}$$

t-Stochastic Neighbor Embedding

- find the projected data y such that the Kullback-Leibler divergences between p and q, is minimized

$$C = \Sigma_i KL(P_i|Q_i) = \Sigma_i \Sigma_j p_{j|i} \log \frac{p_{j|i}}{q_{j|i}}$$

  - KL-divergence is not symmetric. It focus more in the near samples

- SNE is hard to optimized: one can use joint probability instead of conditional

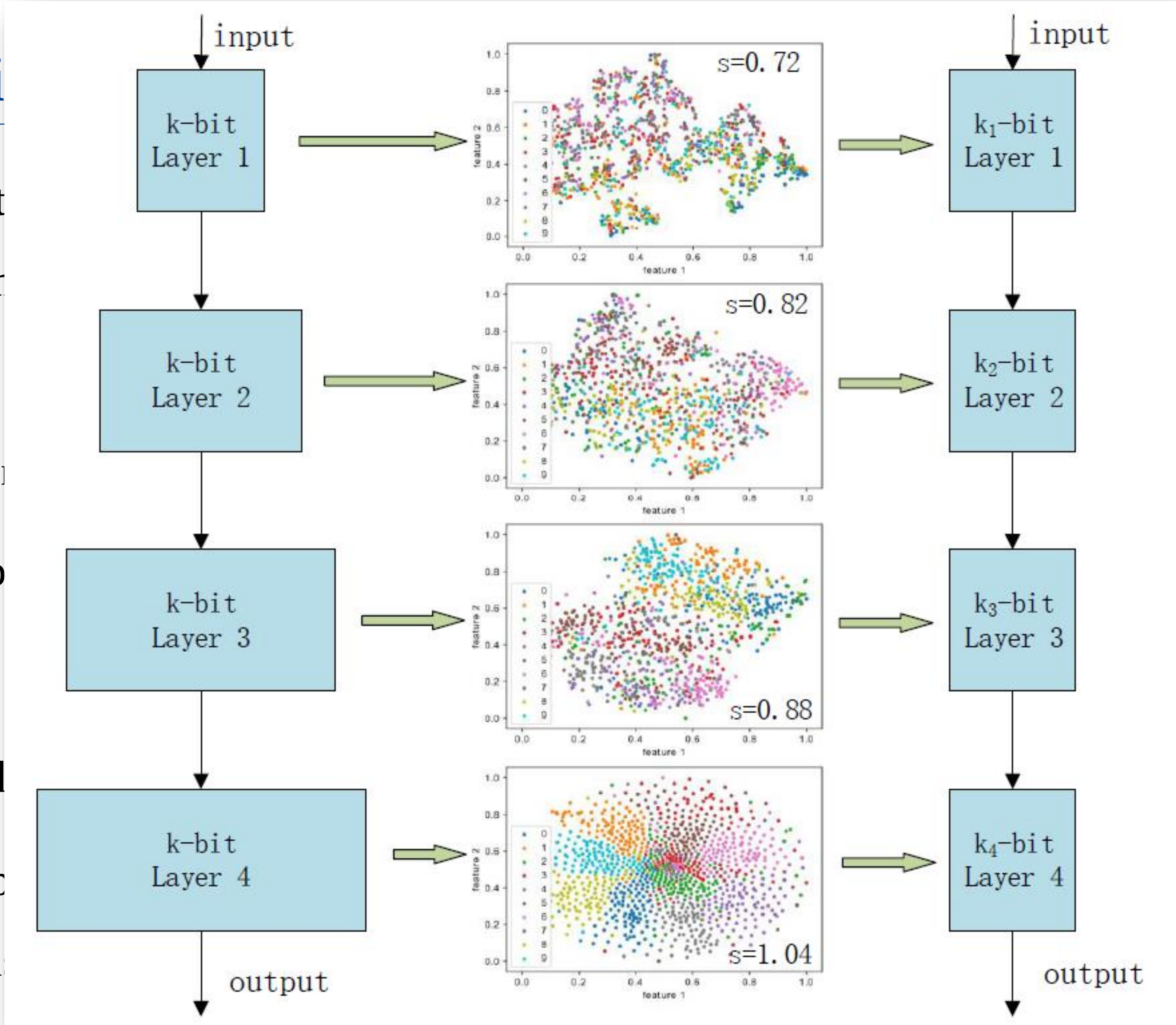
$$p_{x_j,x_i} = \frac{\exp(-\left\|x_i - x_j\right\|^2/2\sigma_i^2)}{\Sigma_{k\neq l} \exp(-\|x_k - x_l\|^2/2\sigma_i^2)}$$

- crowding problem

- t-SNE is to replace Gaussian distribution by student t-distribution

t-Stochasti

- find the project
  and q, is minim
    - KL-diverge
- SNE is hard to
- crowding probl
- t-SNE is to rep
  by student t-di



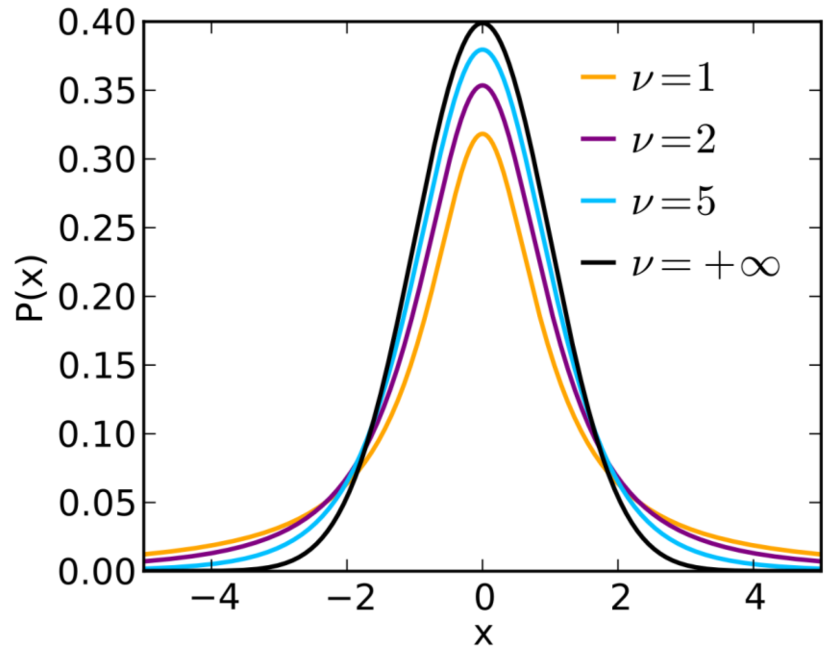
t-Stochastic Neighbor Embe

- find the projected data y such that the Kul

  and q, is minimized

$$C = \Sigma_i KL(P_i|Q_i) = \Sigma_i \Sigma$$

  - KL-divergence is not symmetric. It focus



- SNE is hard to optimized: one can use joint probability instead of conditional

$$p_{x_j,x_i} = \frac{\exp(-\|x_i - x_j\|^2 / 2\sigma_i^2)}{\Sigma_{k \neq l} \exp(-\|x_k - x_l\|^2 / 2\sigma_i^2)}$$

- crowding problem

- t-SNE is to replace Gaussian distribution by student t-distribution

- flexible nonparametric method
- no form, not applicable to new data
- not very good for high dimensional
- not good for intrinsic dimensionality.

Variational Autoencoders

- variational autoencoders learn the parameters of a probability distribution representing the data, instead of a function used in VAE

https://towardsdatascience.com/understanding-variational-autoencoders-vaes-f70510919f73

	input		latent representation		input reconstruction
simple autoencoders	**x**	encoding →	**z = e(x)**	decoding →	**d(z)**

	input		latent distribution		sampled latent representation		input reconstruction
variational autoencoders	**x**	encoding →	$p_\alpha(h\|x)$	sampling →	$q_\beta(z\|h)$	decoding →	z

$$\min_{\alpha,\beta} \; \underbrace{-E_{p_\alpha(h|x)}(\log q_\beta(z|h))}_{\text{reconstruction}} + \underbrace{KL(p_\alpha(h|x)||p(h))}_{\text{prior knowledge}}$$

目录 Contents

Conclusion and Home Work

- modeling optimization for machine learning
  - regression: OLS, ridge regression, lasso, CS, TV,
  - classification: LDA, CCA, SVM,
  - unsupervised: PCA, LLE, EM, autoencoder, VAE
  - probability-related: MLE, MAP, EM, t-SNE, KL

- **Find a topic you like from the above. Write down an introduction for its formulation, history, and frontiers. Preliminarily find an interesting problem you want to deal with and discuss with TAs.**

THANKS