Linear Dimension Reduction (in L_2)

Linear Dimension Reduction: $R^D \rightarrow R^d$

Goal: Find a low-dim. linear map that preserves the relevant information

ie find a d x D matrix **M**

- Application dependent
- *Different definitions yield different techniques*

Some canonical techniques…

- RP (Random Projections)
- PCA (Principal Component Analysis)
- LDA (Linear Discriminate Analysis)
- MDS (Multi-dimensional Scaling)
- ICA/BSS (Independent Component Analysis/Blind Source Separation)
- CCA (Canonical Correlation Analysis)
- DML (Distance Metric Learning)
- DL (Dictionary Learning)
- FA (Factor Analysis)
- NMF/MF ((Non-negative) Matrix Factorization)

Random Projections (RP)

Goal: Find a low-dim. linear map that preserves…

the worst case interpoint Euclidean distances by a factor of $(1 \pm \varepsilon)$

Solution: M with each entry N(0,1/d)

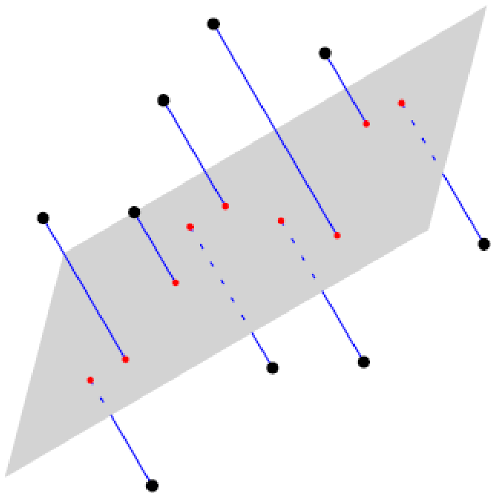
Given $\varepsilon > 0$, pick any $d = \Omega(\log n / \varepsilon^2)$

Given some d, we have $\varepsilon = O(\log n / d)^{1/2}$)

Reasoning: JL lemma.

Principal Component Analysis (PCA)

Goal: Find a low-dim. subspace that minimizes...
the average squared residuals of the given datapoints



Define $\Pi^d : \mathbf{R}^D \to \mathbf{R}^D$ *d-dimensional orthogonal linear projector*

minimize Π^d $$\frac{1}{n}\sum_{i=1}^{n}\left\|\vec{x}_i - \Pi^d(\vec{x}_i)\right\|^2$$

The problem is equivalent to

$$\arg\min_{\substack{Q\in\mathbf{R}^{D\times d}\\ Q^\mathsf{T}Q=I}} \frac{1}{n}\sum_{i=1}^{n}\left\|\vec{x}_i - QQ^\mathsf{T}\vec{x}_i\right\|^2 = \arg\max_{\substack{Q\in\mathbf{R}^{D\times d}\\ Q^\mathsf{T}Q=I}} \operatorname{tr}\left(Q^\mathsf{T}\left(\frac{1}{n}XX^\mathsf{T}\right)Q\right)$$

Solution: Basically is the top d eigenvectors of the matrix XX^T !

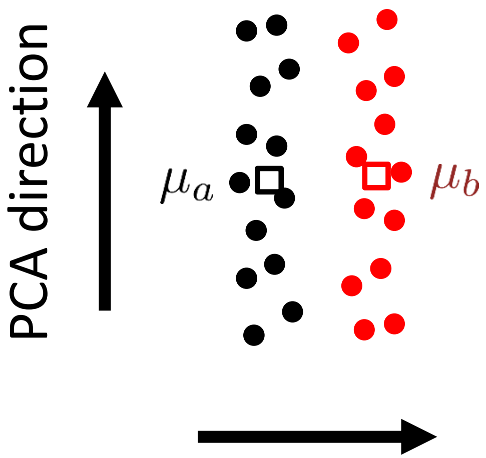
Fisher's Linear Discriminant Analysis (LDA)

Goal: Find a low-dim. map that improves…

classification accuracy!

Motivation:

PCA minimizes reconstruction error $\not\Rightarrow$ good classification accuracy



Classification direction

How can we get classification direction?

Simple idea: pick the direction w that separates the class conditional means as much as possible!

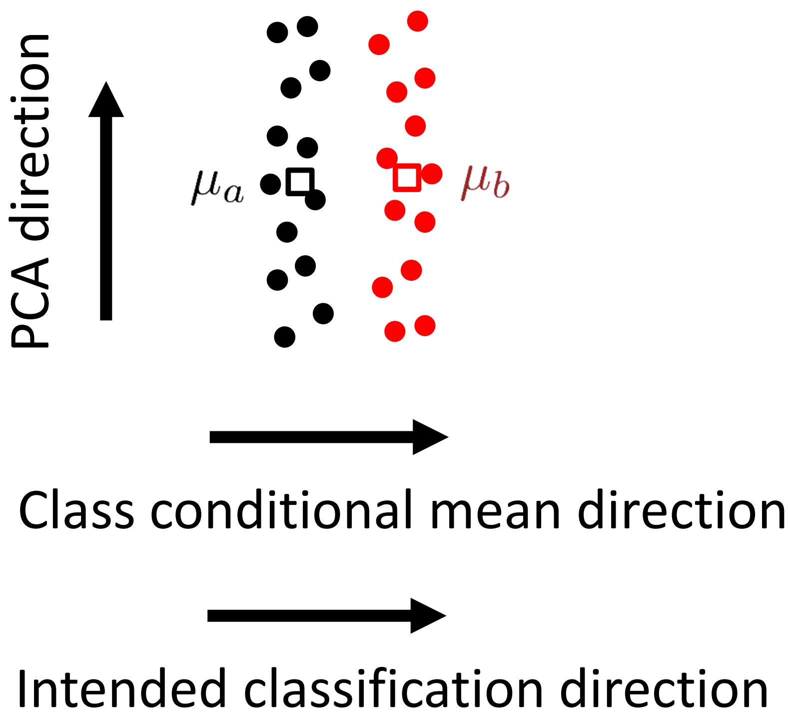
$$\mu_a := \frac{1}{|C_a|} \sum_{x \in C_a} x \qquad \mu_b := \frac{1}{|C_b|} \sum_{x \in C_b} x$$

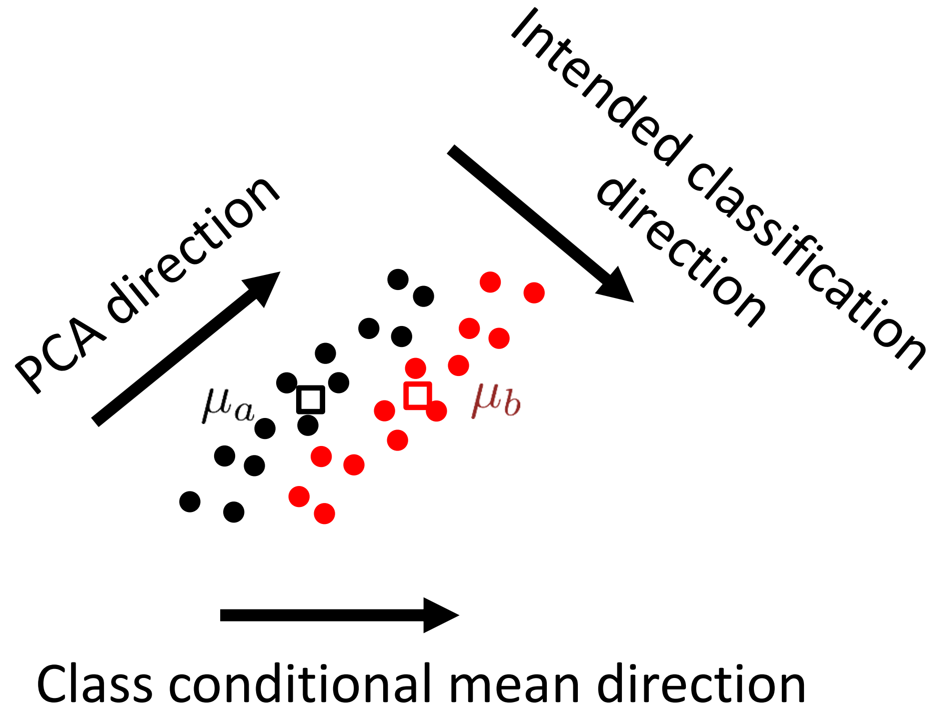
$$\bar{\mu}_a := w^\mathsf{T} \mu_a \qquad \bar{\mu}_b := w^\mathsf{T} \mu_b$$

$$\max_{w, \|w\|=1} L(w) = |\bar{\mu}_b - \bar{\mu}_a| \qquad w^* = \frac{\mu_a - \mu_b}{\|\mu_a - \mu_b\|}$$

Linear Discriminant Analysis (LDA)

So, the direction induced by class conditional means solves simple issues but may still not be the best direction

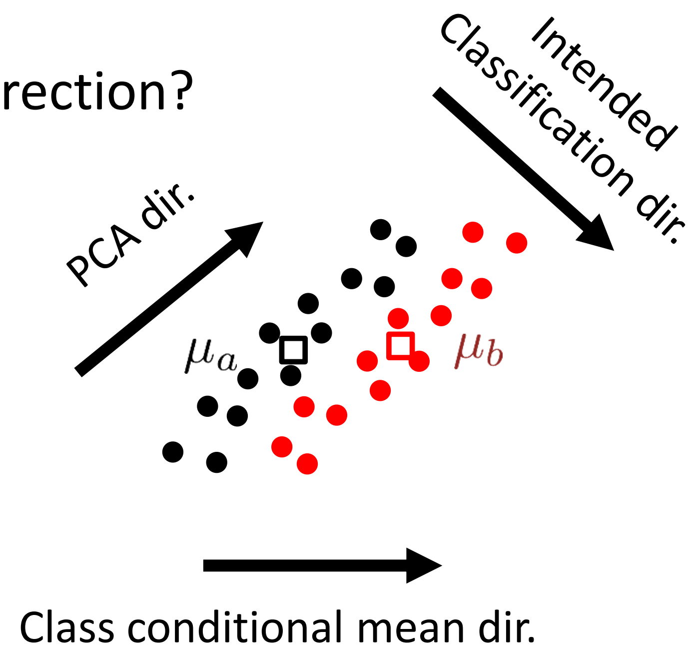




Fix: need to take the projected class conditional spread into account!

Linear Discriminant Analysis (LDA)

So how can we get this intended classification direction?



Want:

- Projected class means as far as possible
- Projected class variance as small possible

$$\mu_a := \frac{1}{|C_a|} \sum_{x \in C_a} x \qquad \mu_b := \frac{1}{|C_b|} \sum_{x \in C_b} x$$

$$\bar{\mu}_a := w^\mathsf{T} \mu_a \qquad \bar{\mu}_b := w^\mathsf{T} \mu_b$$

$$\bar{S}_a^2 := \sum_{\bar{x} \in C_a} (\bar{x} - \bar{\mu}_a)^2 \qquad \bar{S}_b^2 := \sum_{\bar{x} \in C_b} (\bar{x} - \bar{\mu}_b)^2$$

$$\max_w L(w) = \frac{(\bar{\mu}_b - \bar{\mu}_a)^2}{\bar{S}_a^2 + \bar{S}_b^2}$$

Let's study this optimization in more detail…

Linear Discriminant Analysis (LDA)

$$\max_w L(w) = \frac{(\bar{\mu}_b - \bar{\mu}_a)^2}{\bar{S}_a^2 + \bar{S}_b^2}$$

Consider the terms in the denominator…

$$\bar{S}_a^2 = \sum_{\bar{x} \in C_a} (\bar{x} - \bar{\mu}_a)^2 \quad = \sum_{x \in C_a} (w^\mathsf{T}(x - \mu_a))^2$$

$$= w^\mathsf{T}\Big(\sum_{x \in C_a} (x - \mu_a)(x - \mu_a)^\mathsf{T} \Big) w \quad = w^\mathsf{T} S_a w$$

ie, scatter in class "a"

So $\bar{S}_a^2 + \bar{S}_b^2 = w^\mathsf{T}(S_a + S_b)w$

=: S_W (within class scatter)

$$\mu_a := \frac{1}{|C_a|} \sum_{x \in C_a} x$$

$$\mu_b := \frac{1}{|C_b|} \sum_{x \in C_b} x$$

$$\bar{\mu}_a := w^\mathsf{T} \mu_a$$

$$\bar{\mu}_b := w^\mathsf{T} \mu_b$$

$$\bar{S}_a^2 := \sum_{\bar{x} \in C_a} (\bar{x} - \bar{\mu}_a)^2$$

$$\bar{S}_b^2 := \sum_{\bar{x} \in C_b} (\bar{x} - \bar{\mu}_b)^2$$

Linear Discriminant Analysis (LDA)

$$\max_w L(w) = \frac{(\bar{\mu}_b - \bar{\mu}_a)^2}{\bar{S}_a^2 + \bar{S}_b^2}$$

Consider the terms in the numerator…

$$(\bar{\mu}_a - \bar{\mu}_b)^2 = (w^\mathsf{T}(\mu_a - \mu_b))^2$$

$$= w^\mathsf{T}\Big((\mu_a - \mu_b)(\mu_a - \mu_b)^\mathsf{T}\Big)w$$

ie, scatter across classes

=: S_B (between class scatter)

$$\mu_a := \frac{1}{|C_a|}\sum_{x \in C_a} x$$

$$\mu_b := \frac{1}{|C_b|}\sum_{x \in C_b} x$$

$$\bar{\mu}_a := w^\mathsf{T}\mu_a$$

$$\bar{\mu}_b := w^\mathsf{T}\mu_b$$

$$\bar{S}_a^2 := \sum_{\bar{x} \in C_a} (\bar{x} - \bar{\mu}_a)^2$$

$$\bar{S}_b^2 := \sum_{\bar{x} \in C_b} (\bar{x} - \bar{\mu}_b)^2$$

Linear Discriminant Analysis (LDA)

$$\max_w L(w) = \frac{(\bar{\mu}_b - \bar{\mu}_a)^2}{\bar{S}_a^2 + \bar{S}_b^2} = \frac{w^\top S_B w}{w^\top S_W w}$$

So, how do we optimize?

$$0 = \frac{\partial}{\partial w} L(w) = (w^\top S_W w)(2 S_B w) - (w^\top S_B w)(2 S_W w)$$

Divide by $2 w^\top S_W w$

So, at optima

$$S_B w - \frac{w^\top S_B w}{w^\top S_W w}(S_W w) = 0$$

= L(w)

$$S_B w = L(w)(S_W w)$$

$$\Leftrightarrow \quad (S_B S_W^{-1}) w = L(w) w$$

Therefore, optimal w is the maximum eigenvalue of $S_B S_W^{-1}$

Multiclass case (for j classes): $S_W = \sum_j S_j^2; \quad S_B = \sum_j (\mu_j - \mu)(\mu_j - \mu)^\top$

Distance Metric Learning

Goal: Find a linear map that improves… classification accuracy!

Idea: Find a linear map *L* that brings data from same class closer together than different class (this would help improve classification via distance-based methods!)

also called Mahalanobis metric learning

If *L* is applied to the input data, what would be the resulting distance?

$$\rho(x_i, x_j; L) = \|Lx_i - Lx_j\| = \left[(x_i - x_j)^\top L^\top L(x_i - x_j)\right]^{1/2}$$

So, what L would be good for distance-based classification?

Distance Metric Learning

Want:

Distance metric: $\rho(x_i, x_j; L)$

such that: data samples from **same class** yield **small values**

data samples from different class yield large values

One way to solve it mathematically:

Create **two** sets: Similar set $S := \{(x_i, x_j) \mid y_i = y_j\}$

Dissimilar set $D := \{(x_i, x_j) \mid y_i \neq y_j\}$

i, *j* = 1,…, *n*

Define a cost function:

$$\Psi(L) := \lambda \sum_{(x_i,x_j)\in S} \rho^2(x_i, x_j; L) - (1-\lambda) \sum_{(x_i,x_j)\in D} \rho^2(x_i, x_j; L)$$

Minimize Ψ w.r.t. *L*

Several convex variants exist in the literature (e.g. MMC, LMNN, ITML)

Distance Metric Learning

Mahalanobis Metric for Clustering (MMC): [Xing et al. '02]

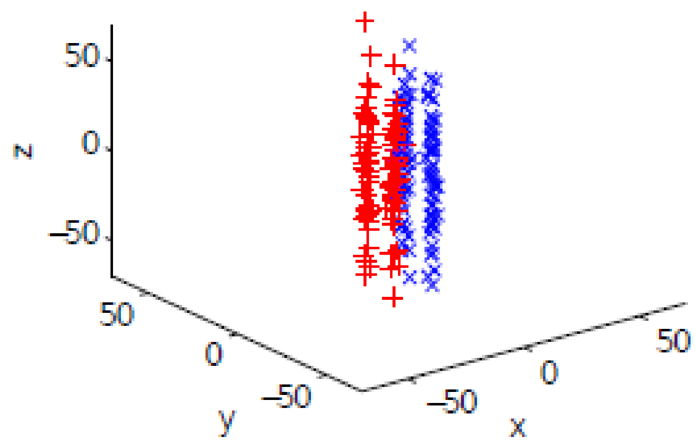
$$\text{maximize}_M \sum_{(x,x')\in D} \rho_M^2(x,x') \qquad (\textit{define } M = L^T L)$$

$$\text{s.t.} \sum_{(x,x')\in S} \rho_M^2(x,x') \le 1 \qquad M \in \text{PSD} \qquad \text{rank}(M) \le k$$

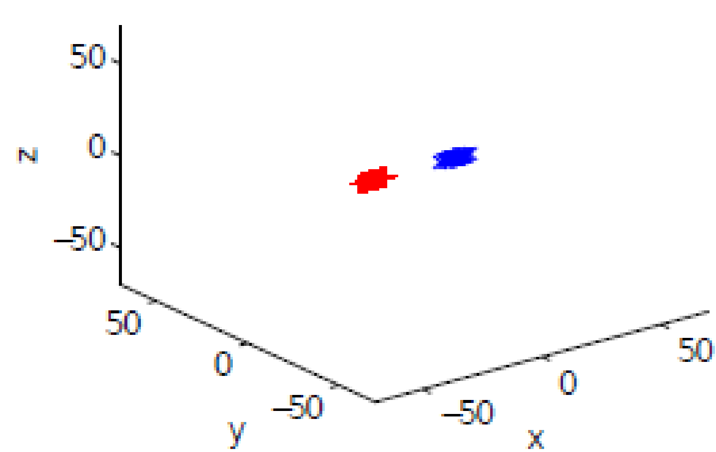
conic constraint

L_0-type non-convex constraint
can relax it to tr(M) ≤ k

Original data



Projected data



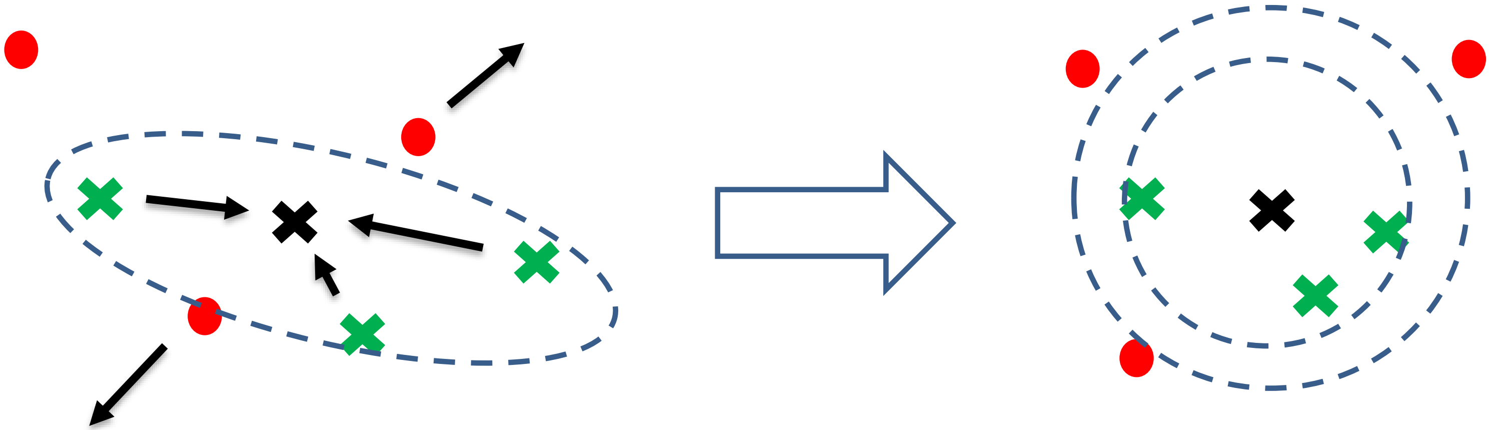
Distance Metric Learning

Large Margin Nearest Neighbor (LMNN): [Weinberger and Saul '09]

$$\Psi_{\text{pull}}(M) = \sum_{i,j(i)} \rho^2_M(x_i, x_j)$$

$$\Psi_{\text{push}}(M) = \sum_{i,j(i),l(i,j)} \left[1 + \rho^2_M(x_i, x_j) - \rho^2_M(x_i, x_l)\right]_+$$

point	i
true neighbor	$j(i)$
imposter	$l(i,j)$



$$\Psi(M) = \lambda \; \Psi_{\text{pull}}(M) \; + (1-\lambda) \; \Psi_{\text{push}}(M)$$

LMNN Performance

Query

After learning

Original metric

Multi-Dimensional Scaling (MDS)

Goal: Find a Euclidean representation of data given only interpoint distances

Given distances ρ_{ij} between (total n) objects, find a vectors $x_1,\ldots,x_n \in R^D$ s.t.

$$\|x_i - x_j\| = \rho_{ij}$$

Classical MDS

Deals with the case when an isometric embedding does exist.

Metric MDS

Deals with the case when an isometric embedding does not exist.

Non-metric MDS

Deals with the case when one only wants to preserve distance order.

Classical MDS

Let D be an n x n matrix s.t. $D_{ij} = \rho_{ij}$

If an isometric embedding exists, then

- One can show that

$$G = -\frac{1}{2} H^T D H \qquad H = I - \frac{1}{n} \mathbb{1}\mathbb{1}^{\mathbb{T}}$$

  is PSD

- Which can then be factorized to construct a Euclidean embedding!

How? See hwk ☺

Metric and non-metric MDS

Metric MDS – (when an isometric embedding does not exist)

There is no direct way; one can solve for the following optimization

$$\min_{x_1,\ldots,x_n} \sum_{i<j} \left(\|x_i - x_j\| - \rho_{ij}\right)^2$$

Stress function

$$\text{s.t.} \quad \sum_i x_i = 0$$

Just do standard constrained optimization

Non-Metric MDS – (only want to preserve distance order)

$$\min_{\substack{x_1,\ldots,x_n \\ g \text{ monotonic}}} \sum_{i<j} \left(g(\|x_i - x_j\|) - \rho_{ij}\right)^2$$

$$\text{s.t.} \quad \sum_i x_i = 0$$

Can do isotonic regression for monotonic g

Blind Source Separation (BSS)

Often the collected data is a mix from multiple sources and a practitioners are interested in extracting the clean signal of the individual sources.

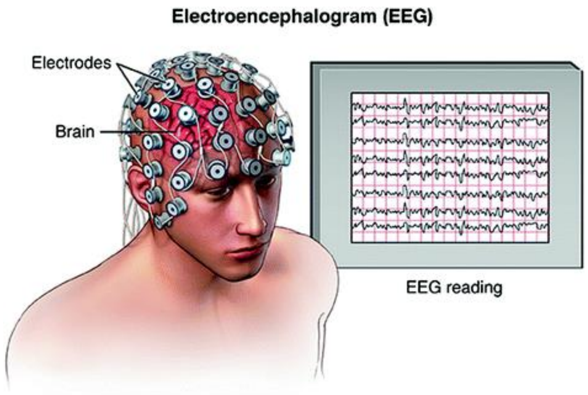
Motivating examples:

The cocktail party problem

- Multiple conversations are happening in a crowded room
- Microphones record a mix of conversations
- Goal is to separate out the conversations

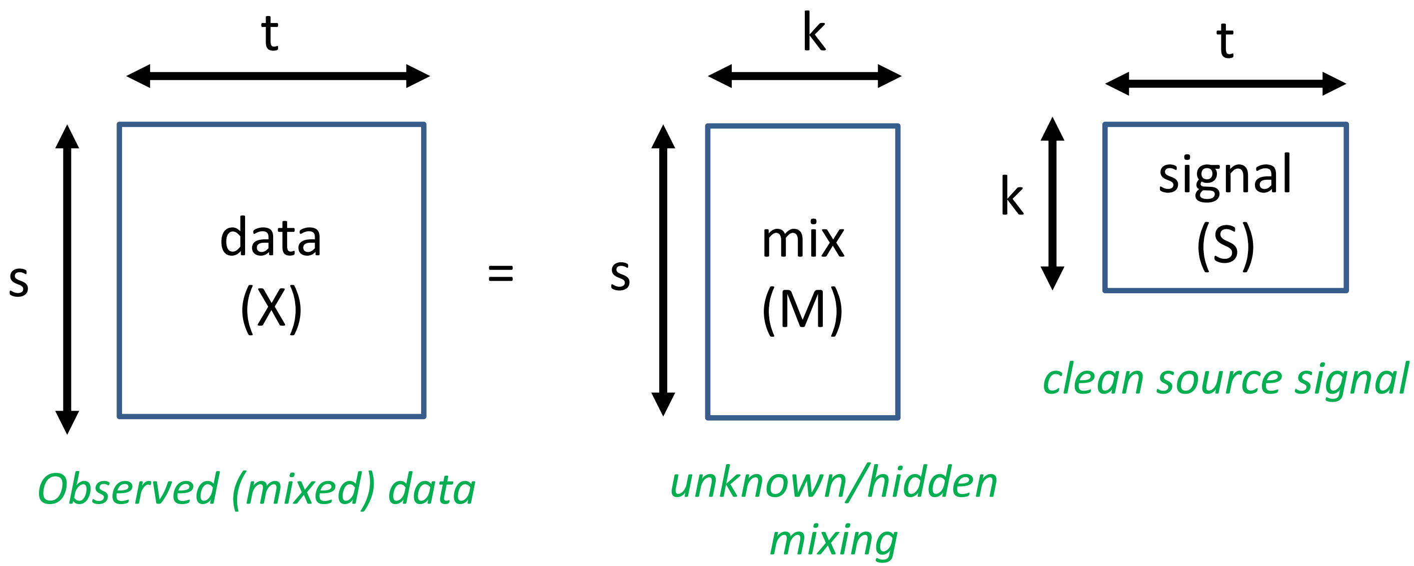
EEG recordings

- Non-invasive way of capturing brain activity
- Sensors pick up a mix of activity signals
- Isolate the activity signals



Blind Source Separation (BSS)

The Data Model:



X = MS

- Goal: given X, recover S (without knowing M)

issue: under-constrained problem, ie multiple plausible solutions. Which one is "correct"?

Blind Source Separation (BSS)

X = MS

Independent component analysis (ICA)

Assumption:

- The source signals S (rows) are generated independently from each other

The matrix M simply mixes these independent signals linearly to generate X

Then, what can we say about X (compared to S)?

Recall: Central Limit Theorem – a linear combination of independent random variables (under mild conditions) essentially looks like a Gaussian!

- X is more gaussian-like than S
- Modified goal: Find entries of S that are least gaussian-like

How to check how Gaussian-like is a distribution?

Blind Source Separation (BSS)

How to measure how “Gaussian-like” a distribution is?

- Kurtosis-based Methods

  kurtosis: fourth (standardized) moment of a distribution

$$\mathrm{Kurt}(X) = E\left[\, ((X-\mu)/\sigma)^4 \,\right]$$

For a gaussian distribution, kurtosis = 3	*Sub-gaussian (‘light’ tailed) , kurtosis < 3*	platykurtic
	Super-gaussian (‘heavy’ tailed), kurtosis > 3	leptokurtic

if we model the i^{th} signal $S_i = W_i^T X$

$$\max_{W_i} \quad \mathrm{Kurt}(W_i^T X)$$
$$\text{s.t.} \quad \mathrm{Var}[W_i^T X] = 1; \quad E[W_i^T X] = 0$$

Blind Source Separation (BSS)

How to measure how "Gaussian-like" a distribution is?

- Entropy-based Methods

  Entropy: measure of uncertainty in a distribution

$$H(X) = -E_x[\log(p(x))]$$

Fact: among all distributions with a fixed variance, Gaussian distribution has the highest entropy!

if we model the i^{th} signal $S_i = W_i^T X$

$$\max_{W_i} \quad -H(W_i^T X)$$

$$\text{s.t.} \quad Var[W_i^T X] = 1; \quad E[W_i^T X] = 0$$

Blind Source Separation (BSS)

Can we make source signals “independent” directly?

- Mutual Information-based Methods

  Mutual info: amount of info a variable contains about the other

$$I(X;Y) = E_{x,y}[\, \log(\, p(x,y) \,/\, p(x)p(y)\,)\,]$$

if we model the i^{th} signal $S_i = W_i^T X$

$$\min \quad \Sigma_{i<j}\; I(W_i^T X;\, W_j^T X)$$

Blind Source Separation (BSS)

Application (cocktail party problem)

- Audio clip

mic 1

mic 2

unmixed source 1

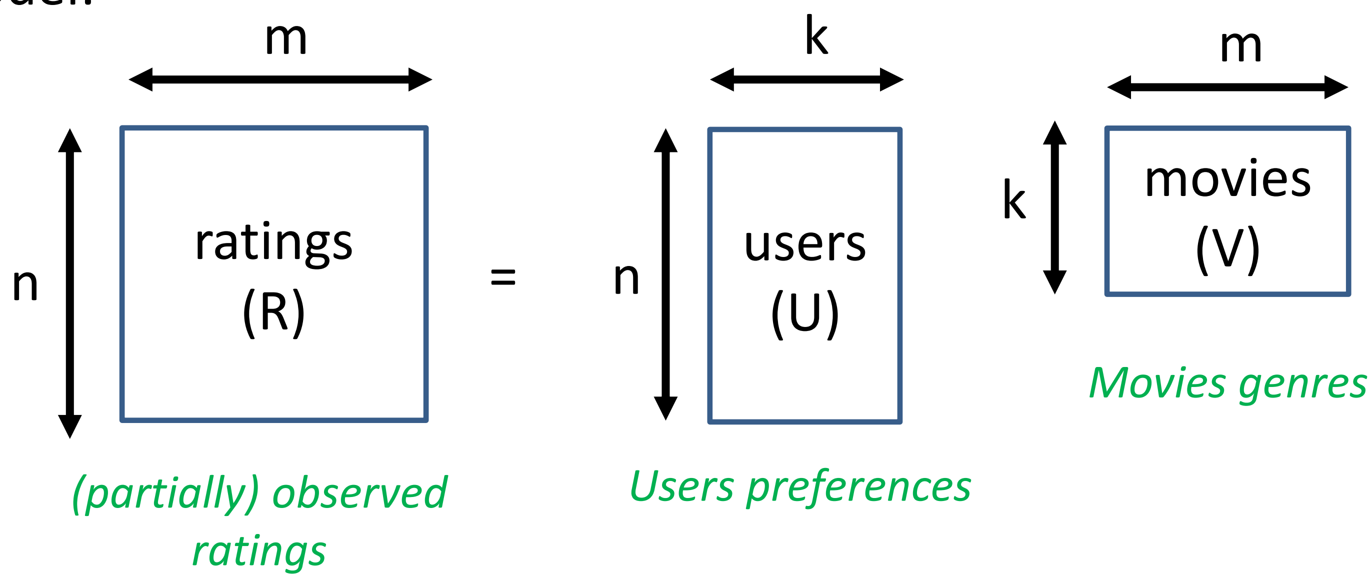
unmixed source 2

Matrix Factorization

Motivation: the Netflix problem

Given *n* users and *m* movies, with some users have rated some of the movies; the goal is to predict the ratings for all movies for all the users.

Data Model:



$R_{ij} = U_i \, . \, V_j$

Matrix Factorization

$R = UV$

$$\min_{U,V} \sum_{R_{ij}\ \text{observed}} (R_{ij} - U_i . V_j)^2$$

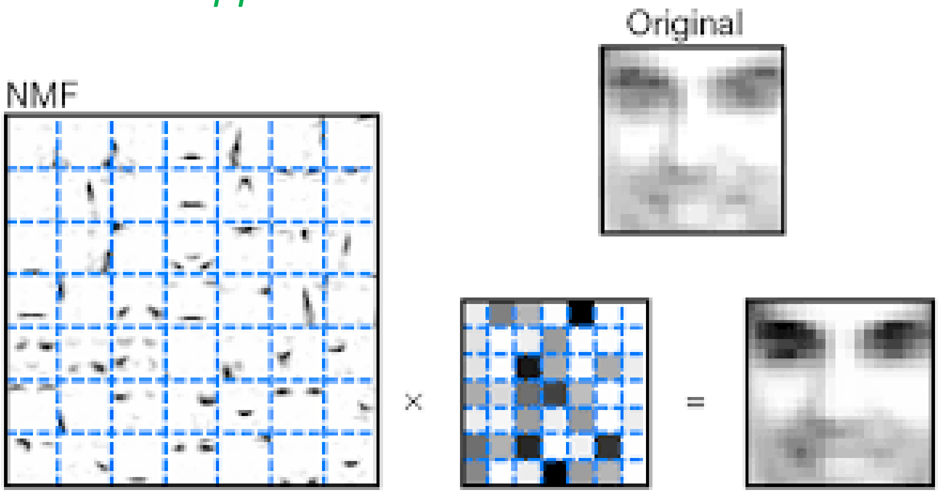
We can optimize using alternating minimization

Equivalent to the probabilistic model where the ratings are generated as

$$R_{ij} = U_i . V_j + \varepsilon_{ij} \qquad \varepsilon \sim N(0,\sigma^2)$$

It is possible to add priors to U and V, which would be helpful for certain applications

Important variations:
Non-negative matrix factorization



Canonical Correlations Analysis (CCA)

What can be done when the data comes in "multiple views"

Same observation – different set of measurements are made

Examples:

Social interaction between individuals

- Video recording of the interaction
- Audio recording of the interaction
- Brain activity recording of the interaction

Ecology – want to study how abundance of special relates to environmental variables

- Data on how species are distributed in various sites
- Data on what environmental variables are there for the same sites

How can we combine multiple views for effective learning?

Canonical Correlations Analysis (CCA)

Canonical correlation analysis (CCA):

- A way of measuring the linear relationship between two variables.
- Finds a projection (linear map) with maximizes the relationship between the variables, which can then be used for data analysis

Let X and Y be the data in two different "views", want to find W_x and W_y which maximally aligns (correlates) the data

Let a = $X^T W_x$; b = $Y^T W_y$ then maximize the correlation between a and b

$$\max_{W_x, W_y} \frac{E(ab)}{\sqrt{E[a^2]E[b^2]}} = \frac{E(W_x^\top X Y^\top W_y)}{\sqrt{E[W_x^\top X X^\top W_x] E[W_y^\top Y Y^\top W_y]}}$$

$$= \frac{E(W_x^\top C_{xy} W_y)}{\sqrt{E[W_x^\top C_{xx}^\top W_x] E[W_y^\top C_{yy} W_y]}}$$

Can be solved via eigendecomposition

Canonical Correlations Analysis (CCA)

Ecology application