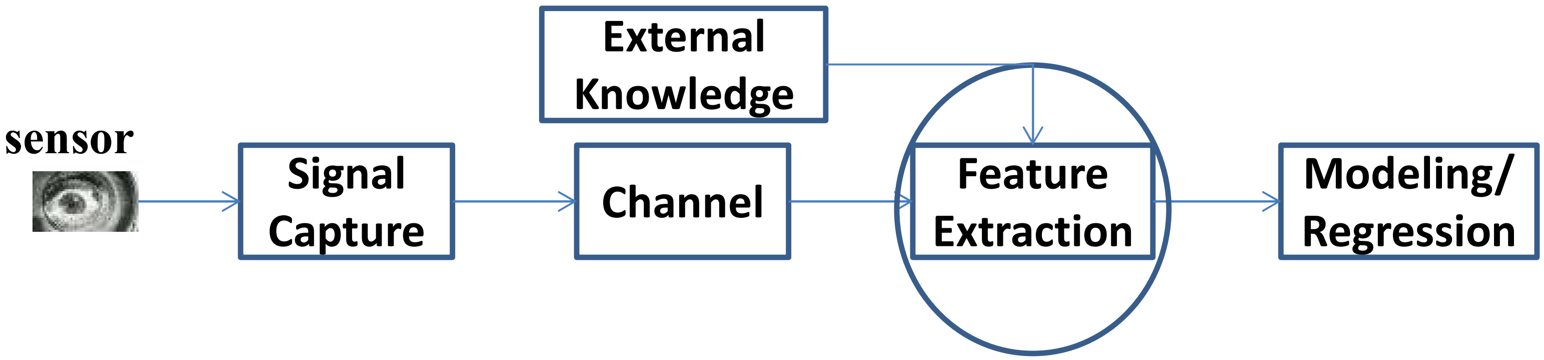
Machine Learning for Signal Processing

Supervised Representations

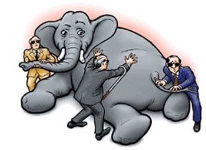
(Slides partially by Najim Dehak)

MLSP

- Application of Machine Learning techniques to the analysis of signals



- Feature Extraction:
  - *Supervised (Guided) representation*



Bases to represent data

- **Basic**: **The bases we considered first were *data agnostic***
  - Fourier / Wavelet type bases, which did not consider the characteristics of the data
- **Improvement I**: **The bases we saw next were data specific**
  - PCA, NMF, ICA, ...
    - Different techniques emphasize different aspects of the data
  - The bases changed depending on the data characteristics
  - But do not consider what the data are *used for*
    - I.e. they are data dependent, but independent of the task
- **Improvement II**: **What if bases are both data specific and task specific?**
  - Basis depends on both the data and the task being performed

Bases to represent data

- **Basic**: **The bases we considered first were *data agnostic***
  - Fourier / Wavelet type bases, which did not consider the characteristics of the data
- **Improvement I**: **The bases we saw next were data specific**
  - PCA, NMF, ICA, ...
    - Different techniques emphasize different aspects of the data
  - The bases changed depending on the data characteristics
  - But do not consider what the data are *used for*
    - I.e. they are data dependent, but independent of the task
- **Improvement II**: **What if bases are both data specific and task specific?**
  - Basis depends on both the data and the task being performed

Recall: Data-dependent bases

- What is a good basis?
  - Energy Compaction → Karkhonen-Loève
  - Retain maximum variance → PCA
    - Also uncorrelatedness of representation
  - Sparsity → Overcomplete bases
  - Constructive composability → NMF
  - Statistical Independence → ICA
- We create a narrative about how the data are created

Task-dependent bases?

- Task: Regression
  - We attempt to predict some variable Y using a variable X
    - Via linear regression
- Standard data-driven bases:
  - Find a representation of X that best captures the characteristics of X
    - Without considering Y
  - Find a representation of Y that best captures the characteristics of Y
    - Without considering X
  - The two representations are independently learned
  - Try to predict (learned representation of) Y from the (learned representation of) X
- Can we do better if the bases used to represent X and Y are *jointly* learned?
  - Such that the learned representation of X is now better able to predict the learned representation of Y

Task-dependent bases?

- Task: Classification
  – We attempt to assign a class Y to input data X

- Standard data-driven bases:
  – Find a representation of X that best captures the characteristics of X
    - Without considering Y
  – Try to predict Y from the (learned representation of) X

- Can we do better if the bases used to represent X *consider* the classes Y?
  – Such that the learned representation of X are more useful for classification of X into Y

Supervised learning of bases

- Problems are instances of *supervised* learning of bases
  - Supervision provided by variable Y
- What is a good basis?
  - Basis that gives best classification performance
  - Basis that results in best regression performance
    - Here bases can be jointly learned for both independent variable X and dependent variable Y
  - In general: Basis that maximizes shared information with another 'view'
    - The second "view" is the task

Multiple Regression

- Robot Archer Example
  - A robot fires defective arrows at a target
    - We don't know how wind might affect their movement, but we'd like to correct for it if possible.
  - Predict the distance from the center of a target of a fired arrow
- Measure wind speed in 3 directions

$$X_i = \begin{bmatrix} 1 \\ w_x \\ w_y \\ w_z \end{bmatrix}$$

Multiple Regression

- Wind speed $$X_i = \begin{bmatrix} 1 \\ w_x \\ w_y \\ w_z \end{bmatrix}$$

- Offset from center in 2 directions $$Y_i = \begin{bmatrix} o_x \\ o_y \end{bmatrix}$$
- Model

$$Y_i = \beta^T X_i$$

Multiple Regression

- Answer

$$\beta = (XX^T)^{-1}XY^T$$

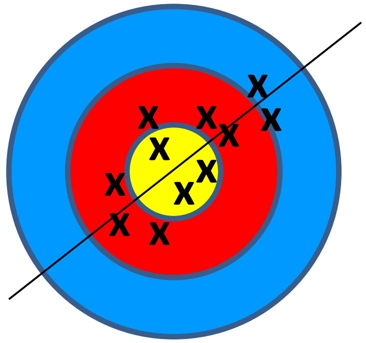
  – Here Y contains measurements of the distance of the arrow from the center

  – $Y_i = \beta^T X_i \rightarrow$ We are fitting a plane

  – Correlation is basically just the gradient of the plane

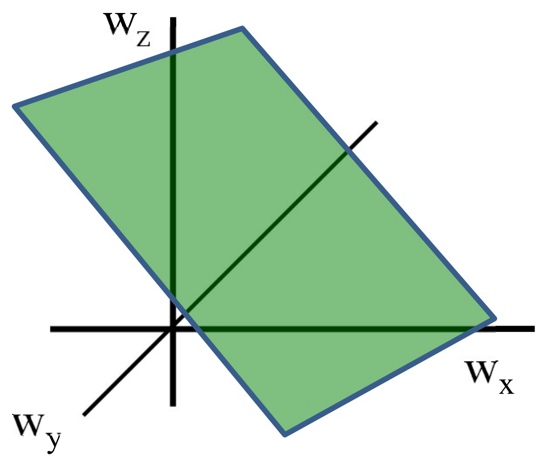
Focusing on what's important

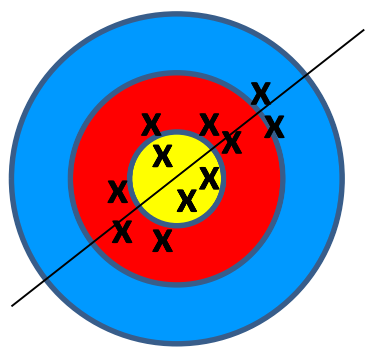
- Do *all* wind factors affect the position
  - Or just some low-dimensional combinations $\hat{X} = U^T X$
- Do they affect both coordinates individually
  - Or just some of combination $\hat{y} = V^T Y$



Canonical Correlation Analysis

- Find a projection of wind vector X, and a projection of arrow location vector Y such that the projection of X best predicts the projection of Y
  - The projection of the vectors for Y and X respectively that are most correlated

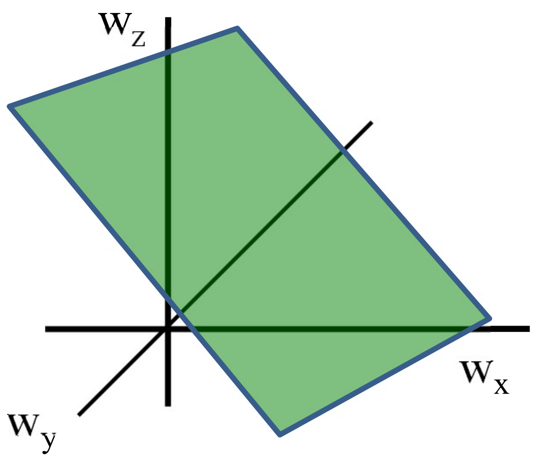


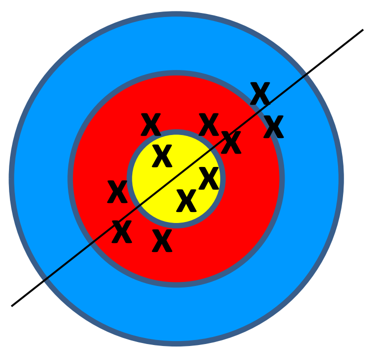


Best X projection plane ⇨ Predicts best Y projection

Canonical Correlation Analysis

- What do these vectors represent?
  - Direction of max correlation ignores parts of wind and location data that do not affect each other
    - Only information about the defective arrow remains!



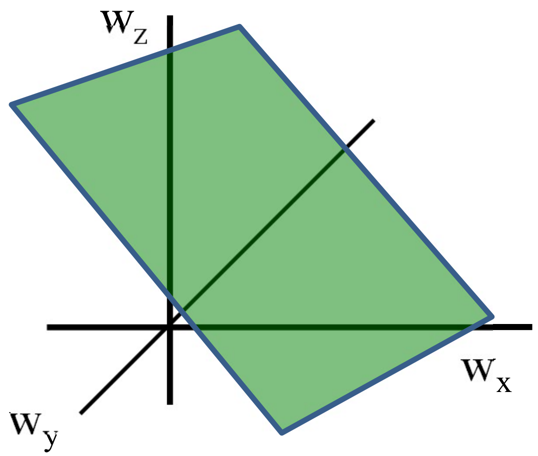


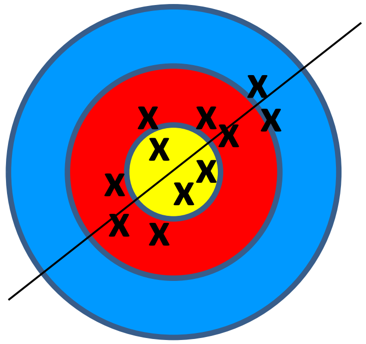
Best X projection plane ⇨ Predicts best Y projection

Why not just jointly analyze

- Why not just concatenate both variables?
  - E.g. create $Z = [X^T \; Y^T]^T$ and just perform PCA on Z

- It does not exploit the extra structure of the signal (more on this shortly)
  - PCA on joint data will decorrelate *all variables*
    - Also mixes X and Y, whereas we want to predict Y from X
  - We want to decorrelate X and Y, but *maximize* cross-correlation between X and Y





A Quick Review

- Matrix representation

$$\mathbf{X} = [X_1, X_2, \dots, X_N] \qquad \mathbf{Y} = [Y_1, Y_2, \dots, Y_N]$$

$$C_{XX} = \frac{1}{N}\sum_i X_i X_i^T = \frac{1}{N}\mathbf{X}\mathbf{X}^T$$

$$C_{YY} = \frac{1}{N}\sum_i Y_i Y_i^T = \frac{1}{N}\mathbf{Y}\mathbf{Y}^T$$

$$C_{XY} = \frac{1}{N}\sum_i X_i Y_i^T = \frac{1}{N}\mathbf{X}\mathbf{Y}^T$$

Cross Correlation

- Cross correlation between X and Y :

$$C_{XY} = E[XY^T]$$
$$= \frac{1}{N}\mathbf{XY}^T$$

- If we project X and Y down to k dimensions:

$$\hat{X} = U^T X$$
$$\hat{Y} = V^T Y$$

- U is $N \times k$, V is $M \times k$, where N and M are the dimension of X and Y

- Cross correlation between $\hat{X}$ and $\hat{Y}$ is

$$C_{\hat{X}\hat{Y}} = E\left[\hat{X}\hat{Y}^T\right] = U^T C_{XY} V$$
$$= \frac{1}{N} U^T \mathbf{XY}^T V$$

Maximizing Cross Correlation

- Maximize $C_{\hat{X}\hat{Y}}$

$$\hat{U}, \hat{V} = \underset{U,V}{\operatorname{argmax}} C_{\hat{X}\hat{Y}}$$

- Which becomes:

$$\hat{U}, \hat{V} = \underset{U,V}{\operatorname{argmax}} U^T C_{XY} V$$

- Where

$$C_{XY} = \frac{1}{N}\mathbf{X}\mathbf{Y}^T$$

- Is this enough?

Maximizing Cross Correlation

- Maximize $C_{\hat{X}\hat{Y}}$

$$\hat{U}, \hat{V} = \underset{U,V}{\operatorname{argmax}}\, U^T C_{XY} V$$

$$\hat{U}, \hat{V} = \underset{U,V}{\operatorname{argmax}} \frac{1}{N} U^T \mathbf{X}\mathbf{Y}^T V$$

- This can be arbitrarily maximized by simply scaling up U, or V …
  – Or X or Y
- So how do we constrain this?

Constraints

$$\hat{U}, \hat{V} = \underset{U,V}{\operatorname{argmax}} \frac{1}{N} U^T \mathbf{X}\mathbf{Y}^T V$$

- Options
  - Whiten X
  - Whiten Y
  - Constrain U to be an orthonormal matrix
    - All Eigen values are 1
  - Constrain V to be an orthonormal matrix

- Can we compact these further?

A Quick Review

- The effect of a transform on the covariance of an RV

$$\hat{X} = U^T X$$

$$C_{XX} = E[XX^T]$$

$$C_{\hat{X}\hat{X}} = E\left[\hat{X}\hat{X}^T\right] = U^T C_{XX} U$$

Canonical Correlation Analysis

- Constrain $\hat{X}$ and to $\hat{Y}$ be white

$$\hat{U}, \hat{V} = \underset{U,V}{\operatorname{argmax}}\, U^T C_{XY} V$$

- Such that

$$U^T C_{XX} U = I_k,\ V^T C_{YY} V = I_k$$

Canonical Correlation Analysis

- Constrain $\hat{X}$ and to $\hat{Y}$ be white

$$\hat{U}, \hat{V} = \underset{U,V}{\operatorname{argmax}} \; U^T C_{XY} V$$

How do you maximize a matrix??

- Such that

$$U^T C_{XX} U = I_k, \;\; V^T C_{YY} V = I_k$$

Canonical Correlation Analysis

- Constrain $\hat{X}$ and to $\hat{Y}$ be white

$$\hat{U},\hat{V} = \underset{U,V}{\operatorname{argmax}}\, \operatorname{trace}(U^T C_{XY} V)$$

- Such that

$$U^T C_{XX} U = I_k,\ V^T C_{YY} V = I_k$$

- Maximize the sum of the diagonals of the cross correlation matrix
  – I.e. maximize the sum of the Eigen values of the cross correlation matrix

Poll 1

Canonical Correlation Analysis

- Constrain $\hat{X}$ and to $\hat{Y}$ be white

$$\hat{U}, \hat{V} = \underset{U,V}{\operatorname{argmax}} \operatorname{trace}(U^T C_{XY} V)$$

- Such that

$$U^T C_{XX} U = I_k, \; V^T C_{YY} V = I_k$$

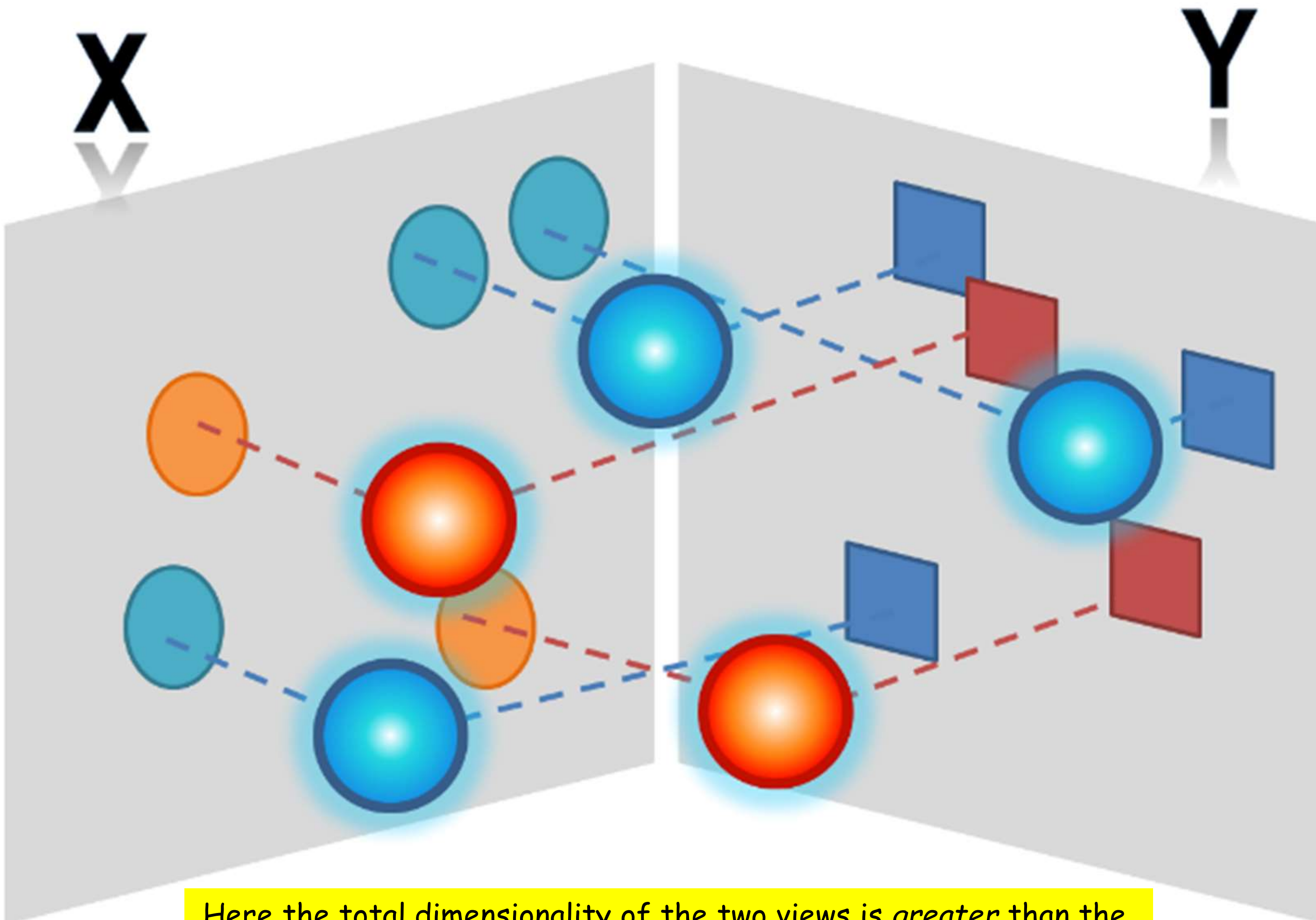
- Maximize the sum of the diagonals of the cross correlation matrix
  - I.e. maximize the sum of the Eigen values of the cross correlation matrix
    - **Why the *sum?***

Multiview Assumption

- CCA models both variables as different views of a common reality
  - X and Y are obtained from different views of the same common space
    - The two views are correlated
    - But each of the views also loses some information
      - E.g the total dimensions of the views of X and Y may be fewer than the total dimensions of the space
  - Each view locally perturbed by noise

- Challenge: Extract the correlated subspaces of X and Y from their noise

Multiview Examples

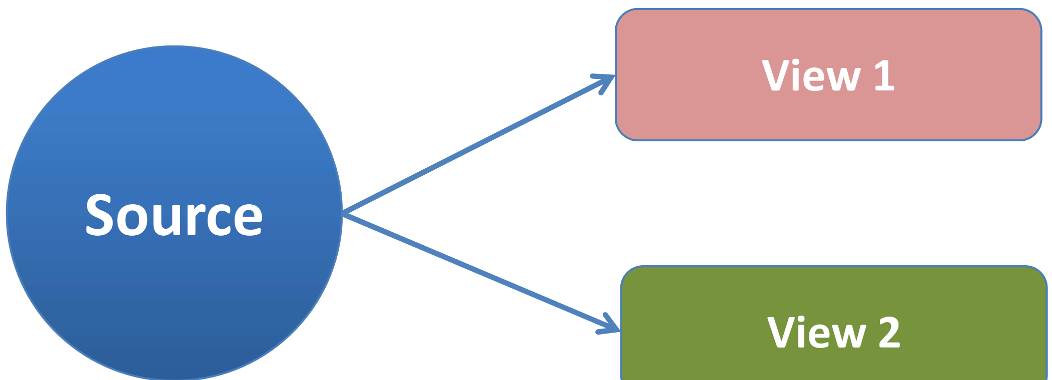


Here the total dimensionality of the two views is *greater* than the dimensions of the original data

http://mlg.postech.ac.kr/static/research/multiview_overview.png

Multiview Assumption

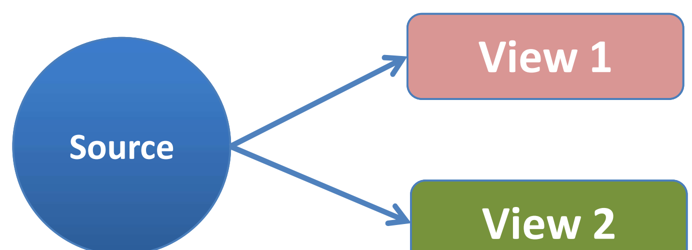
- We can sort of think of a model for how our data might be generated



- We want View 1 independent of View 2 conditioned on knowledge of the source
  - All correlation is due to source

Multiview Examples

- Look at many stocks from different sectors of the economy
  – Conditioned on the fact that they are part of the same economy they might be independent of one another
- Multiple Speakers saying the same sentence
  - The sentence generates signals from many speakers. Each speaker might be independent of each other conditioned on the sentence



Recall: Least squares formulae

$$E = \sum_i (X_i - Y_i)^2$$

$$\mathbf{X} = [X_1, X_2, \ldots, X_N] \qquad \mathbf{Y} = [Y_1, Y_2, \ldots, Y_N]$$

$$E = \|\mathbf{X} - \mathbf{Y}\|_F^2$$

- Expressing total error as a matrix operation

CCA objective

- CCA attempts to “reconstruct” a shared reality from both views
  - And tries to make them both look the same

$$\underset{U\in\mathbb{R}^{N\times k},\ V\in\mathbb{R}^{M\times k}}{\operatorname{argmin}} \|U^T\mathbf{X} - V^T\mathbf{Y}\|_F^2$$

minimize

$$s.t.\ \ U^T C_{XX} U = I_k,\ V^T C_{YY} V = I_k$$

CCA Derivation

$$\underset{U \in \mathbb{R}^{N \times k},\ V \in \mathbb{R}^{M \times k}}{\operatorname{argmin}} \|U^T \mathbf{X} - V^T \mathbf{Y}\|_F^2$$

$$s.t.\ \ U^T \mathbf{X}\mathbf{X}^{\mathrm{T}} U = N I_k,\ \ V^T \mathbf{Y}\mathbf{Y}^{\mathrm{T}} V = N I_k$$

$$s.t.\ \ U^T C_{XX} U = I_k,\ V^T C_{YY} V = I_k$$

CCA Derivation

$$\|U^T\mathbf{X} - V^T\mathbf{Y}\|_F^2 = trace(U^T\mathbf{X} - V^T\mathbf{Y})(U^T\mathbf{X} - V^T\mathbf{Y})^T$$

$$= trace(U^T\mathbf{X}\mathbf{X}^TU + V^T\mathbf{Y}\mathbf{Y}^TV - U^T\mathbf{X}\mathbf{Y}^TV - V^T\mathbf{Y}\mathbf{X}^TU)$$

$$= 2Nk - 2trace(U^T\mathbf{X}\mathbf{Y}^TV)$$

- So we can solve the equivalent problem below

$$\max_{U,V} \; trace(U^TC_{XY}V)$$

$$s.t. \;\; U^TC_{XX}U = I_k, \; V^TC_{YY}V = I_k$$

CCA Derivation

- Incorporating Lagrangian, maximize

$$\mathcal{L}(\Lambda_X, \Lambda_Y) = tr(U^T C_{XY} V)$$

$$-tr\left(\left((U^T C_{XX} U) - I_k\right)\Lambda_X\right) - tr\left(\left((V^T C_{YY} V) - I_k\right)\Lambda_Y\right)$$

- Remember that the constraints matrices are symmetric
- Also for any A, B ,

$$\nabla_A tr(AB) = B^T$$

$$\nabla_A tr(ABA^T) = A(B + B^T)$$

CCA Derivation

- Taking derivatives and after a few manipulations

$$\Lambda_X = \Lambda_Y = \Lambda$$

- We arrive at the following system of equation

$$C_{YX}\tilde{U} = C_{YY}\tilde{V}D$$
$$C_{XY}\tilde{V} = C_{XX}\tilde{U}D$$

CCA Derivation

- We isolate $\tilde{V}$

$$\tilde{V} = C_{YY}^{-1} C_{YX} \tilde{U} D^{-1}$$

- We arrive at the following system of equation

$$\boxed{\begin{aligned} C_{XX}^{-1} C_{XY} C_{YY}^{-1} C_{YX} \tilde{U} &= \tilde{U} D^2 \\ C_{YY}^{-1} C_{YX} C_{XX}^{-1} C_{XY} \tilde{V} &= \tilde{V} D^2 \end{aligned}}$$

CCA Derivation

- For $\tilde{U}$ we just have to find eigenvectors for

$$C_{XX}^{-1} C_{XY} C_{YY}^{-1} C_{YX}$$

  - Basically, the Eigen vectors for the correlation of the vector obtained by transforming X to Y and back to X
  - After normalizing out the local variance

- We then solve for the other view using the expression for $\tilde{V}$ on the previous slide.

- In PCA the eigenvalues were the variances in the PCA bases directions

- In CCA the eigenvalues are the squared correlations in the canonical correlation directions

CCA as Generalized Eigenvalue Problem

- Combine the system of eigenvalue eigenvector equations

$$\begin{bmatrix} 0 & C_{XY} \\ C_{YX} & 0 \end{bmatrix} \begin{bmatrix} \tilde{U} \\ \tilde{V} \end{bmatrix} = \begin{bmatrix} C_{XX} & 0 \\ 0 & C_{YY} \end{bmatrix} \begin{bmatrix} \tilde{U} \\ \tilde{V} \end{bmatrix} D$$

- Generalized eigenvalue problem

$$AU = BU\Lambda$$

- We assumed invertible C_{XX}, C_{YY} → $\exists B^{-1}$
- Solve a single eigenvalue/vector equation

$$B^{-1}A\tilde{U} = \tilde{U}D$$

CCA Fixes

- We assumed invertibility of covariance matrices.
  - Sometimes they are close to singular and we would like stable matrix inverses
  - If we added a small positive diagonal element to the covariances then we could guarantee invertibility.
- It turns out this is equivalent to regularization

CCA Fixes

- The following problems are equivalent
  - They have the same gradients

$$\min_{U,V} \|U^T\mathbf{X} - V^T\mathbf{Y}\|_F^2 + \lambda_x\|U\|_F^2 + \lambda_y\|V\|_F^2$$

$$\max_{U,V} \; trace(U^T\mathbf{X}\mathbf{Y}^T V)$$

$$s.t.\;\; U^T(C_{XX}+\lambda_x I)U = I_k,\;\; V^T(C_{YY}+\lambda_y I)V = I_k$$

- The previous solution still applies but with slightly different autocovariance matrices
  - "Diagonal load" the autocovariances

CCA Fixes

- Since we now have strictly positive autocovariance matrices, we know they have Cholesky decompositions.

$$(C_{XX} + \lambda_x I) = L_{XX} L_{XX}^T$$

- This results in the following problem

$$L_{XX}^{-\frac{1}{2}} C_{XY} (C_{YY} + \lambda_y I)^{-1} C_{YX} (L_{XX}^{-\frac{1}{2}})^T \tilde{U} = \tilde{U} D$$

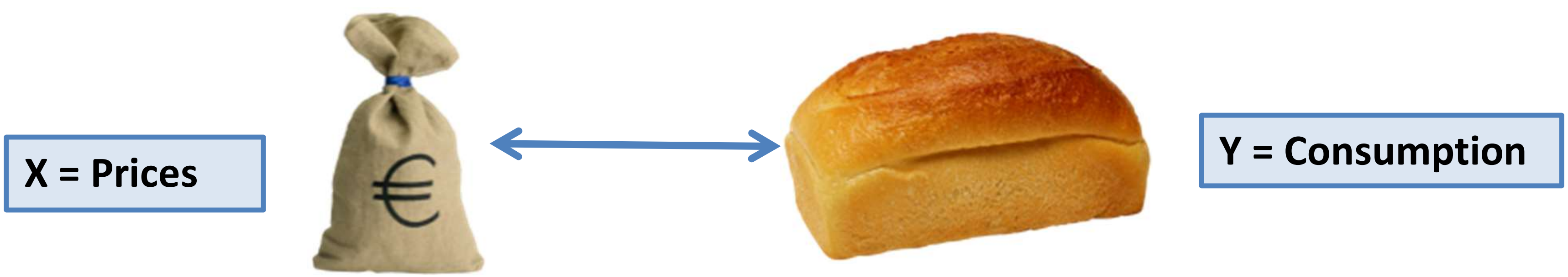
- We note that the matrix is symmetric and
- So the problem is solved by SVD on the matrix M

$$L_{XX}^{-\frac{1}{2}} C_{XY} (C_{YY} + \lambda_y I)^{-1} C_{YX} (L_{XX}^{-\frac{1}{2}})^T = M M^T \text{ with } M = L_{XX}^{-\frac{1}{2}} C_{XY} (C_{YY} + \lambda_y I)^{-\frac{1}{2}}$$

Poll 2

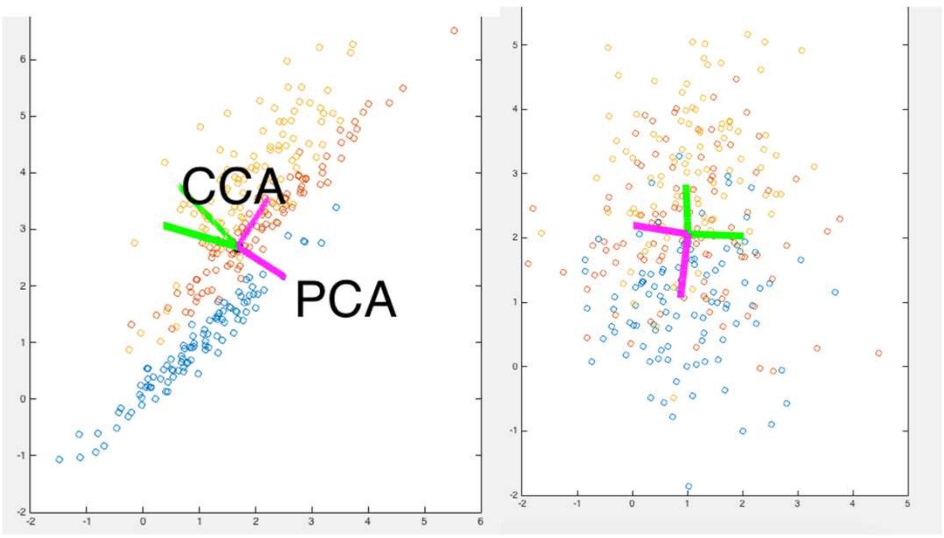
CCA Motivation and History

- Proposed by Hotelling (1936)
- Many real world problems involve 2 ‘views’ of data
- **Economics**
  - Consumption of wheat is related to the price of potatoes, rice and barley … and wheat
  - Random vector of prices X
  - Random vector of consumption Y



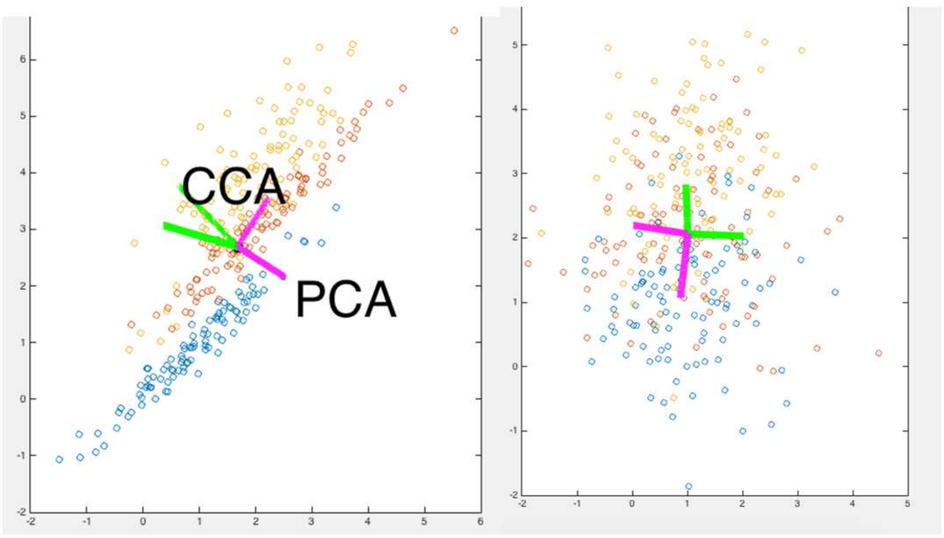
CCA Motivation and History

- Magnus Borga, David Hardoon popularized CCA as a technique in signal processing and machine learning
- Better for dimensionality reduction in many cases



CCA Dimensionality Reduction

- We keep only the correlated subspace
- Is this always good?
  - If we have measured things we care about then we have removed useless information



CCA Dimensionality Reduction

- In this case:
  - CCA found a basis component that preserved class distinctions while reducing dimensionality
  - Able to preserve class in both views

Comparison to PCA

- PCA fails to preserve class distinctions as well

Failure of PCA

- PCA is unsupervised
  - Captures the direction of greatest variance (Energy)
  - No notion of task or hence what is good or bad information
  - The direction of greatest variance can sometimes be noise
  - Ok for reconstruction of signal
  - Catastrophic for preserving class information in some cases

Benefits of CCA

- Why did CCA work?
  - Supervision
    - External Knowledge
  - The 2 views track each other in a direction that does not correspond to noise
  - Noise suppression (sometimes)
- Preview
  - If one of the sets of signals are true labels, CCA is equivalent to Linear Discriminant Analysis
  - Hard Supervision

Multiview Assumption

- When does CCA work?
  - The correlated subspace must actually have interesting signal
    - If two views have correlated noise then we will learn a bad representation

- Sometimes the correlated subspace can be noise
  - Correlated noise in both sets of views

CCA as Generalized Eigenvalue Problem

- Rayleigh Quotient

$$\lambda_{max}(B^{-1}A) = \max_x \frac{x^T A x}{x^T B x}$$

$$\frac{\delta}{\delta x}\frac{x^T A x}{x^T B x} = \frac{\delta}{\delta x} x^T A x (x^T B x)^{-1} = 0$$

$$= 2Ax(x^T B x)^{-1} - x^T A x (x^T B x)^{-2} 2Bx = 0$$

$$\implies \frac{1}{x^T B x}(Ax - \frac{x^T A x}{x^T B x} Bx) = 0$$

$$\implies Ax = \frac{x^T A x}{x^T B x} Bx$$

CCA as Generalized Eigenvalue Problem

- So the solutions to CCA are the same as those to the Rayleigh quotient
- PCA is actually also this problem with

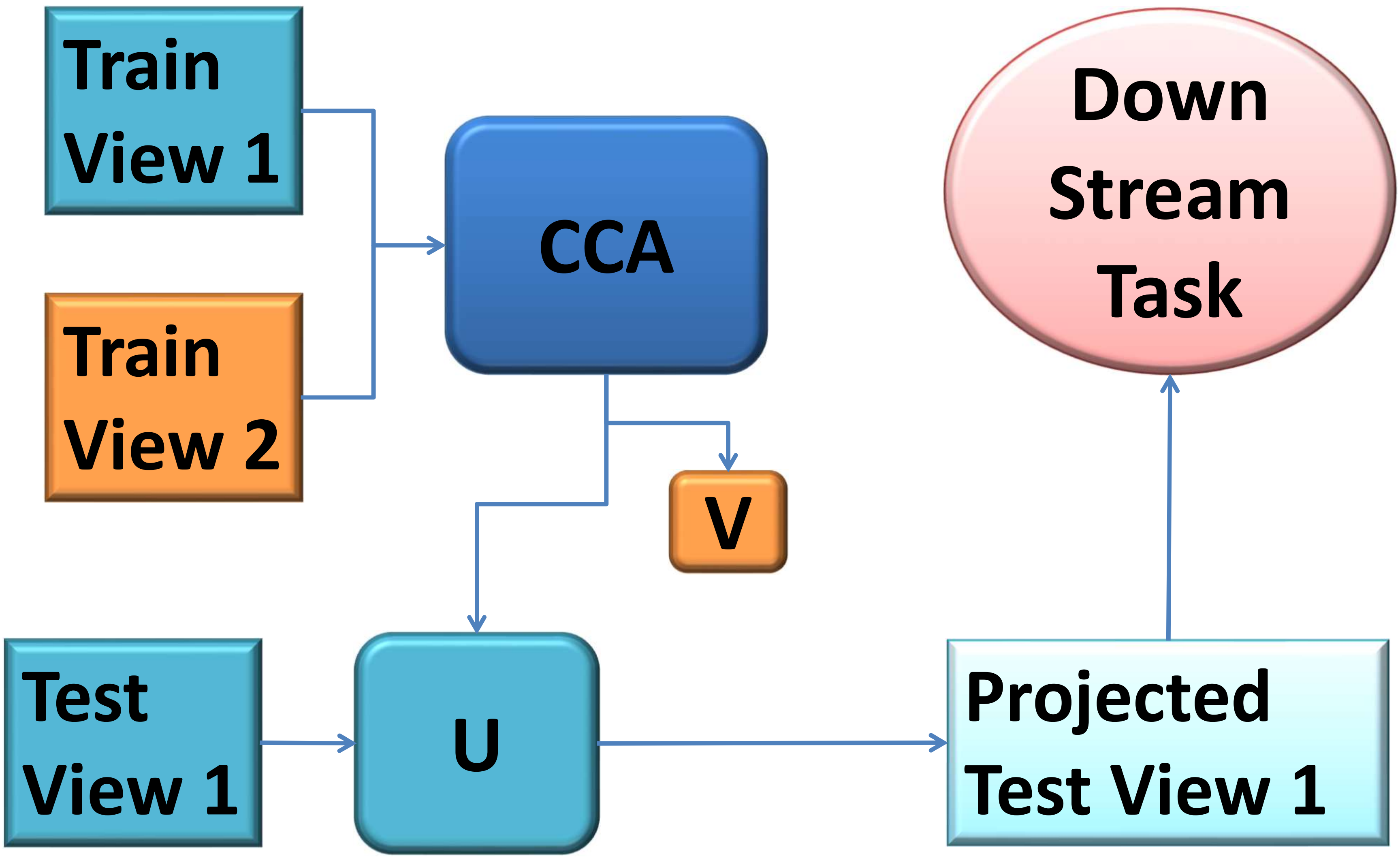
$$A = C_{XX}, \; B = I$$

- We will see that Linear Discriminant Analysis also takes this form, but first we need to fix a few CCA things

What to do with the CCA Bases?

- The CCA Bases are important in their own right.
  - Allow us a generalized measure of correlation
  - Compressing data into a compact correlative basis
- For machine learning we generally …
  - Learn a CCA basis for a class of data
  - Project new instances of data from that class onto the learned basis
  - This is called multi-view learning

Multiview Setup



Multiview Setup

- Often one view consists of measurements that are very hard to collect
  - Speakers all saying the same sentence
  - Articulatory measurements along with speech
  - Odd camera angles
  - Etc.

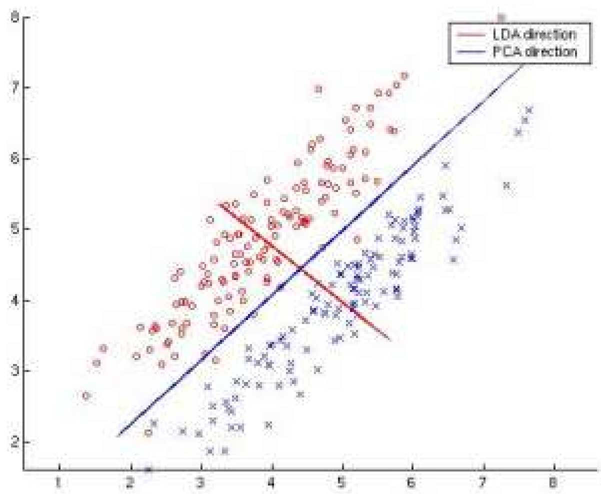
Multiview Setup

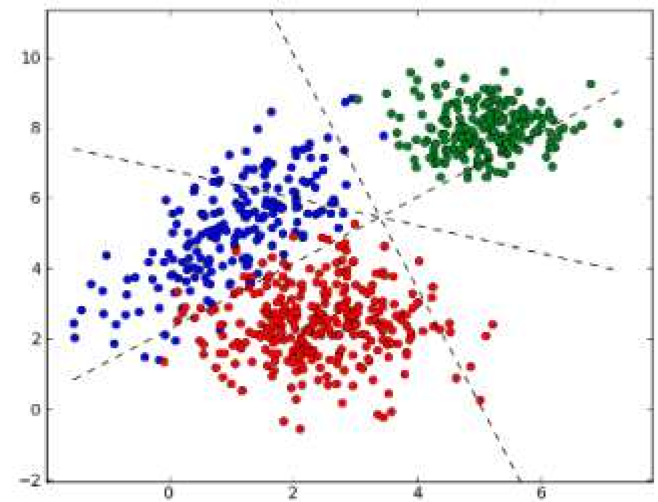
- We learn the correlated direction from data during training
- Constrain the common view to lie in the correlated subspace at test time
  - Removes useless information (Noise)

http://ema.umcs.pl/pl/laboratorium/

Poll 3

Linear Discriminant Analysis





- Given data from two classes
- Find the projection U
- Such that the separation between the classes is maximum along U
  - Y = $U^T X$ is the projection bases in U
  - No other basis separates the classes as much as U

Linear Discriminant Analysis

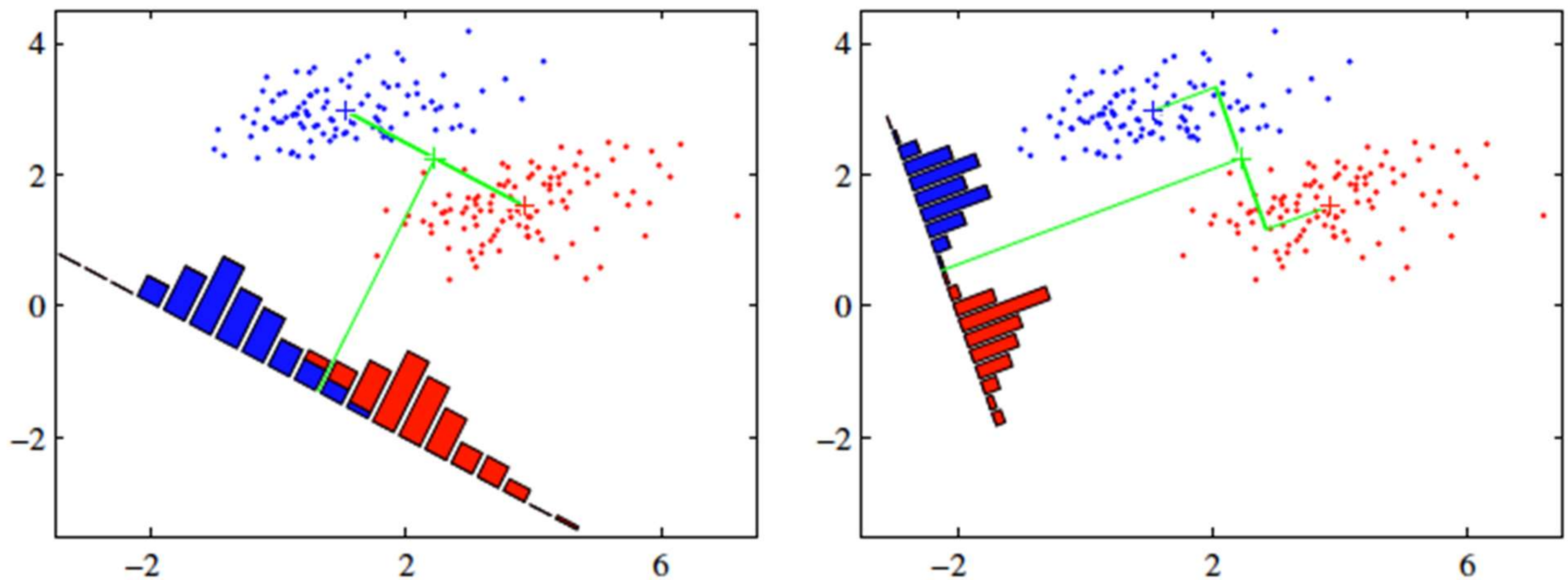
- We have 2 views as in CCA
- One of the views is the class labels of the data
  - Learn the direction that is maximally correlated with the class labels!
- It turns out that LDA and CCA are equivalent when the situation above is true

LDA Formulation

- LDA setup
  - Assume classes are roughly Gaussian
    - Still works if they are not, but not as well
  - We know the class membership of our training data
  - Classes are distinguishable by …
    - Big gaps between classes with no data points
    - Relatively compact clusters

LDA Formulation

- LDA setup



LDA Formulation

- We define a few Quantities
  - Within-class scatter

$$\mathbf{S}_{\mathrm{W}} = \sum_{k=1}^{K} \mathbf{S}_k \qquad \mathbf{S}_k = \sum_{n \in \mathcal{C}_k} (\mathbf{x}_n - \mathbf{m}_k)(\mathbf{x}_n - \mathbf{m}_k)^{\mathrm{T}}$$

    - Minimize how far points can stray from the mean
    - Compact classes
  - Between-class scatter
    - Maximize the variance of the class means (distance between means)

$$\mathbf{S}_{\mathrm{B}} = \sum_{k=1}^{K} N_k (\mathbf{m}_k - \mathbf{m})(\mathbf{m}_k - \mathbf{m})^{\mathrm{T}}$$

LDA Formulation

- We want a small within-class variance
- We want a high between-class variance
- Let's maximize the ratio of the two!!

- Remember we are looking for the basis W onto which projections maximize this ratio
  - Key concept: what is the covariance of $Y = W^T X$ given C_{XX}?

Recall: Effect of projection on scatter

- Let $Y = W^T X$
- Let S_B and S_W be the between and within class scatter of X

- Within class scatter of Y: $S_W^Y = W^T S_W W$
- Between class scatter of Y: $S_B^Y = W^T S_B W$
- Must maximize S_B^Y while minimizing S_W^Y.

LDA Formulation

- We actually have too much freedom
  - Without any constraints on W
    - Let's fix the within-class variance to be 1.

$$\underset{W \in R^{d\times k}}{\text{argmax}}\, tr(W^T S_B W) \;\; s.t. \;\; W^T S_W W = I$$

  - The Lagrangian is …

$$L(\Lambda) = \underset{W \in R^{d\times k}}{\text{argmax}} \quad tr(W^T S_B W) - tr\big((W^T S_W W - I)\Lambda\big)$$

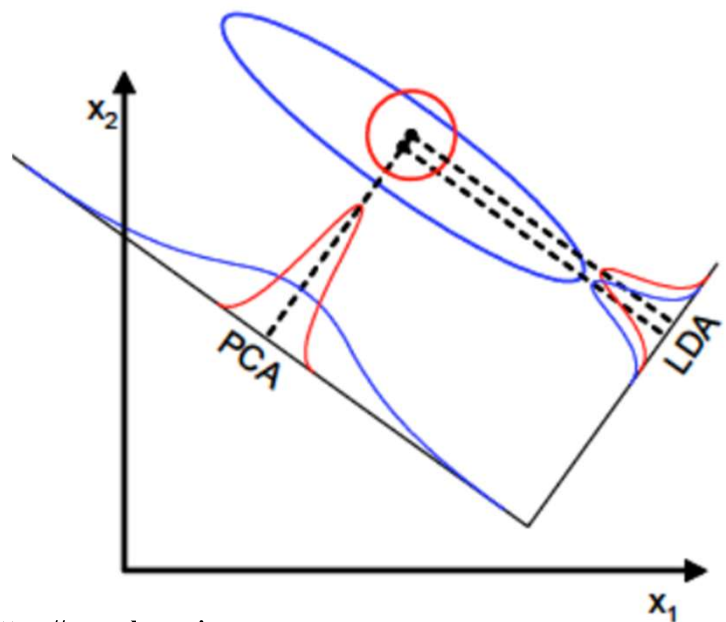
  - So we see that we have a generalized eigenvalue solution

$$S_B w = \lambda S_w w$$

    - w is any column of W and λ is a diagonal entry of Λ

LDA Formulation

- When does LDA fail?
  - When classes do not fit into our model of a blob
  - We assumed classes are separated by means
  - They might be separated by variance
  - We can fix this using heteroscedastic LDA
    - Fixes the assumption of shared covariance across class.



https://www.lsv.uni-saarland.de/fileadmin/teaching/dsp/ss15/DSP2016/matdid437773.pdf

Poll 4

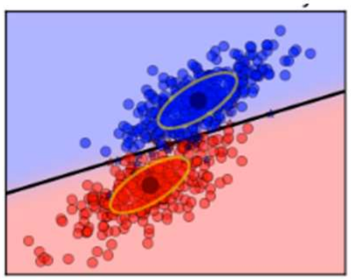
LDA for classification

- For each class assume a Gaussian Distribution
  - Estimate parameters of the Gaussian
  - We want argmax P(Y = K | X)
  - We use Bayes rule

  P(Y = K | X) = P(X | Y = K)P(Y = K)

  - We end up with linear decision surfaces between classes

$$\log\left(\frac{P(y=k|X)}{P(y=l|X)}\right) = 0 \Leftrightarrow (\mu_k - \mu_l)\Sigma^{-1}X = \frac{1}{2}(\mu_k^t\Sigma^{-1}\mu_k - \mu_l^t\Sigma^{-1}\mu_l)$$

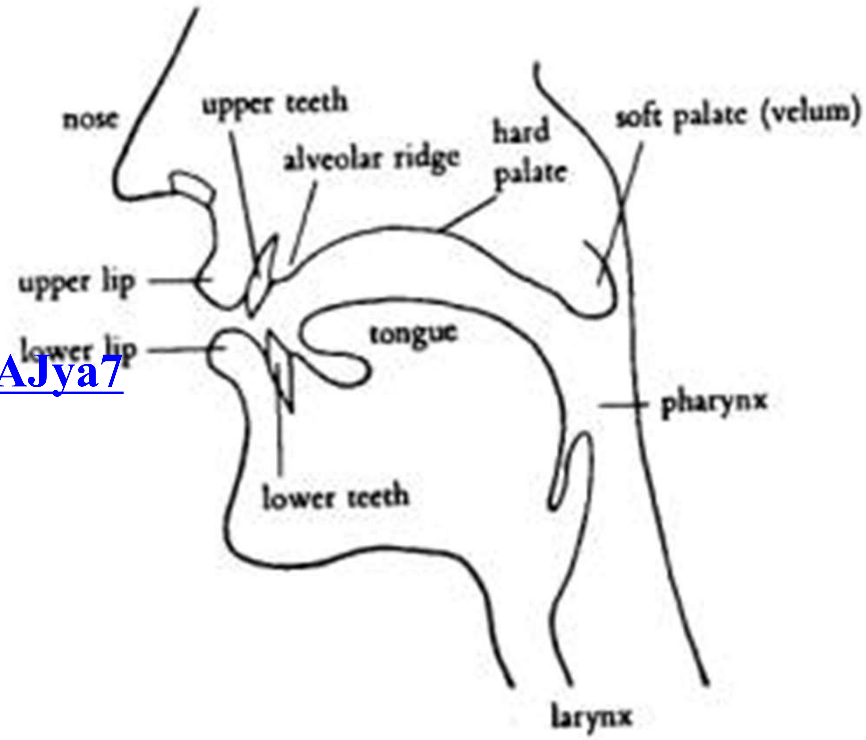


For the best classification, perform Bayes classification on the LDA projections

Bakeoff – PCA, CCA, LDA on Vowel Classification

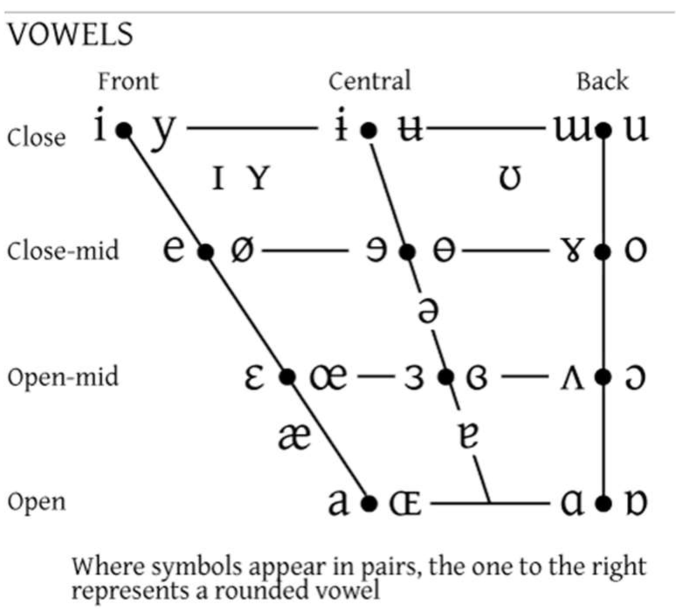
- Speech is produced by an excitation in the glottis (vocal folds)
- Sound is then shaped with the tongue, teeth, soft palate …
- This shaping is what
  generates the
  different vowels

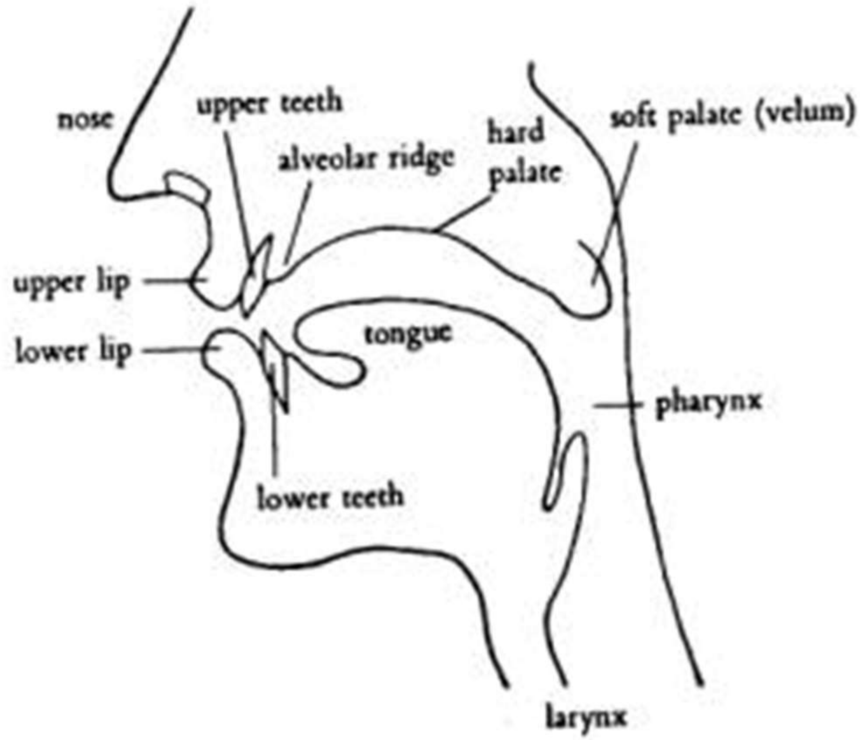
https://www.youtube.com/watch?v=58AJya7JzOU#t=00m36s



Bakeoff – PCA, CCA, LDA on Vowel Classification

- To represent where in the mouth the vowels are being shaped linguists have something called a vowel diagram
- It classifies vowels as front-back, open-closed depending on tongue position

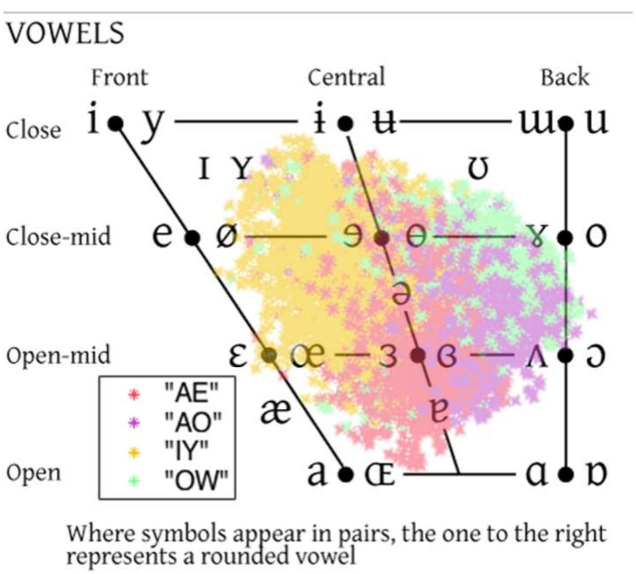




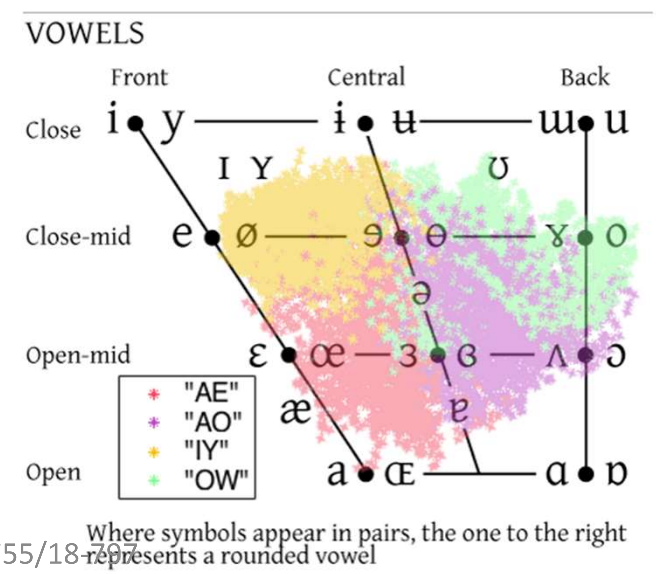
Bakeoff – PCA, CCA, LDA on Vowel Classification

- Task:
  - Discover the vowel chart from data
- CCA on Acoustic and Articulatory View
  - Project Acoustic data onto top 3 dimensions

PCA



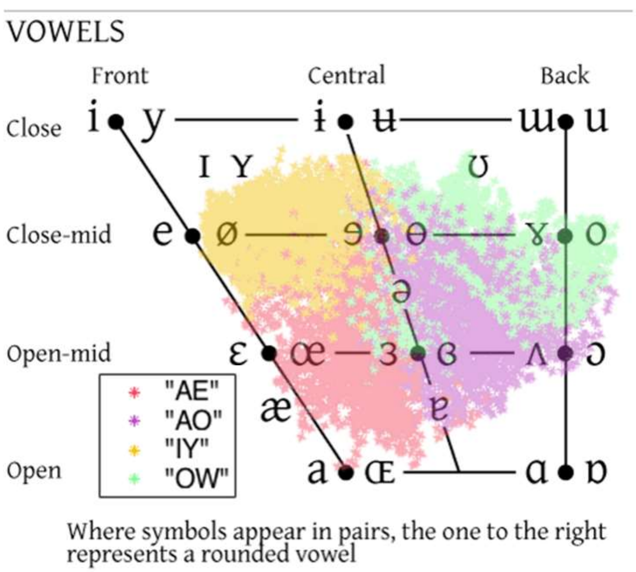
CCA



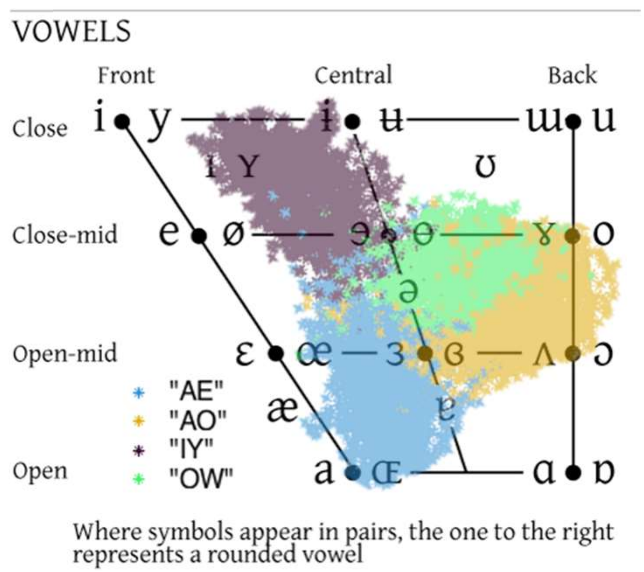
Bakeoff – PCA, CCA, LDA on Vowel Classification

- Using a one hot encoding of labels as a view gives LDA

CCA

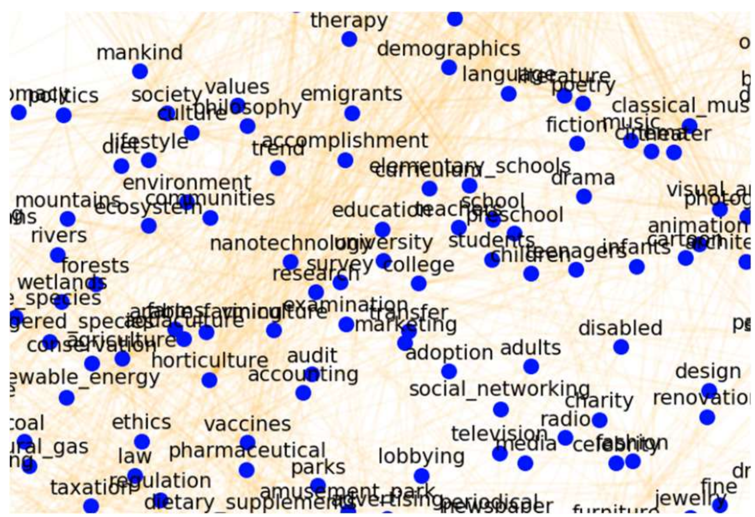


LDA



Multilingual CCA

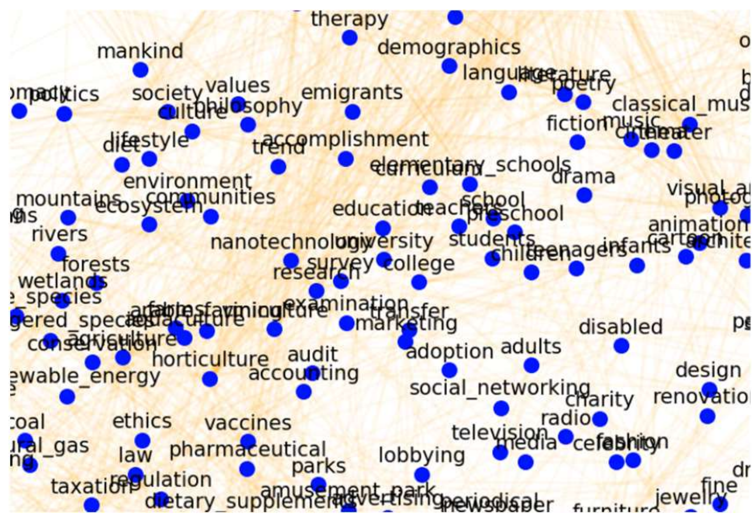
- Another Example of CCA
  - Word is mapped into some vector space
  - A notion of distance between words is defined and the mapping is such that words that are semantically similar are mapped to near to each other (hopefully)



http://www.trivial.io/word2vec-on-databricks/

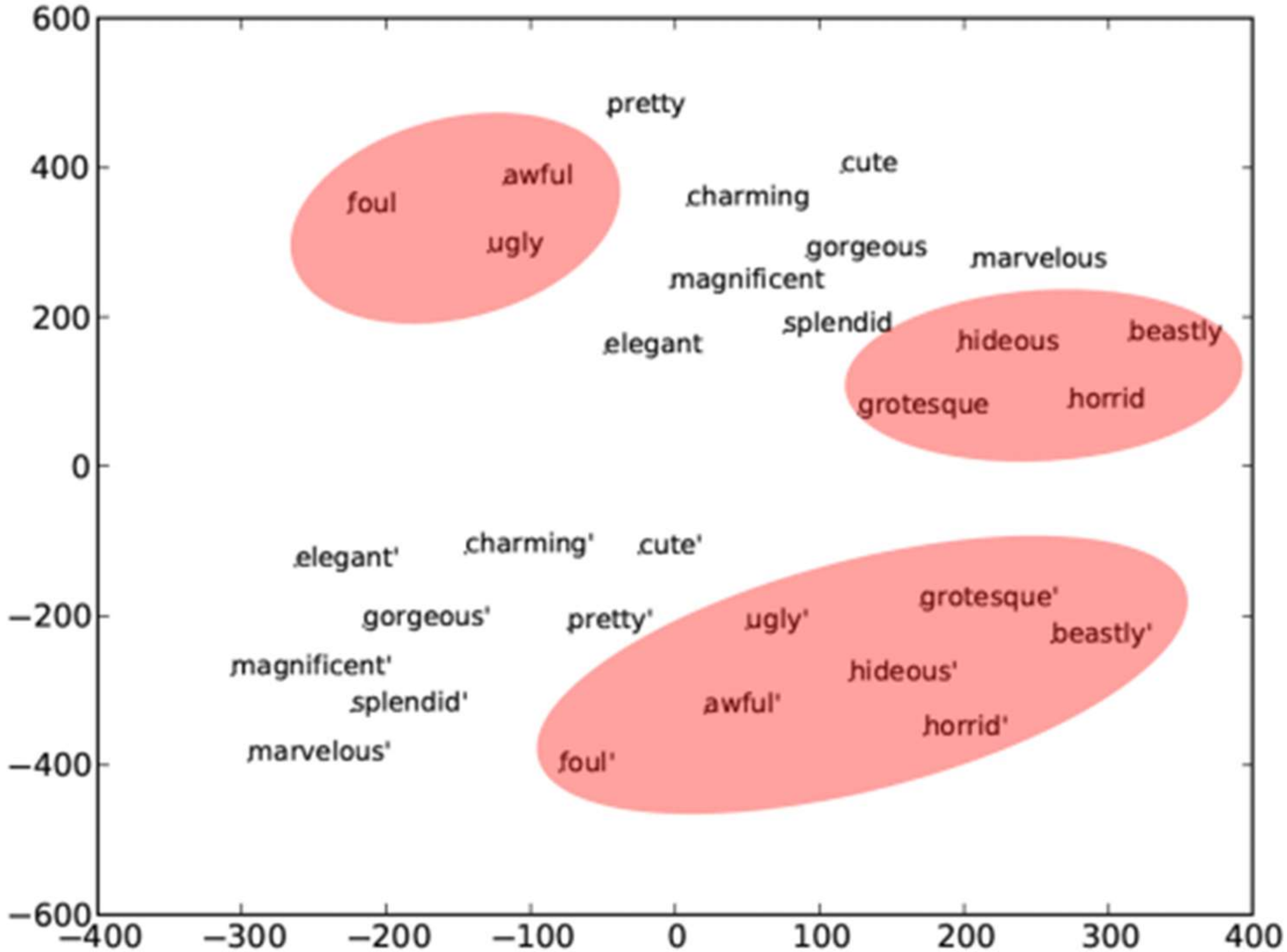
Multilingual CCA

- What if parallel text in another language exists?
- What if we could generate words in another language?
- Use different languages as different views



http://www.trivial.io/word2vec-on-databricks/

Multilingual CCA



Faruqui, Manaal, and Chris Dyer. "Improving vector space word representations using multilingual correlation." Association for Computational Linguistics, 2014.

Fisher Faces

- We can apply LDA to the same faces we all know and love.
  - The details, especially stranger ones such as eye depth emerge as discriminating features

Conclusions

- LDA learns discriminative representations by using supervision
  - Knowledge of Labels
- CCA is equivalent to LDA when one view is labels
  - CCA provides soft supervision by exploiting redundant view of data