Machine Learning for Signal Processing

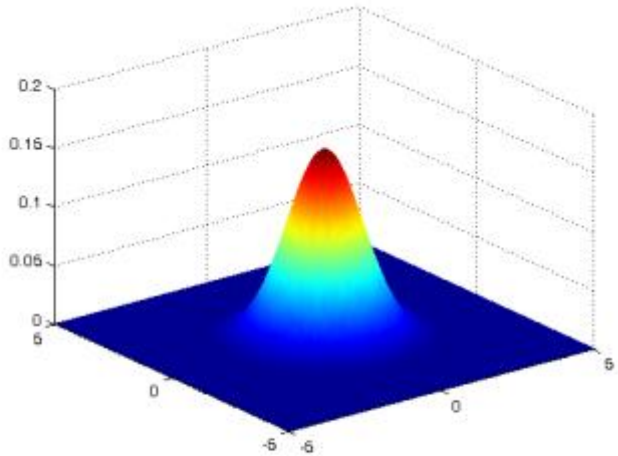
Supervised Representations:

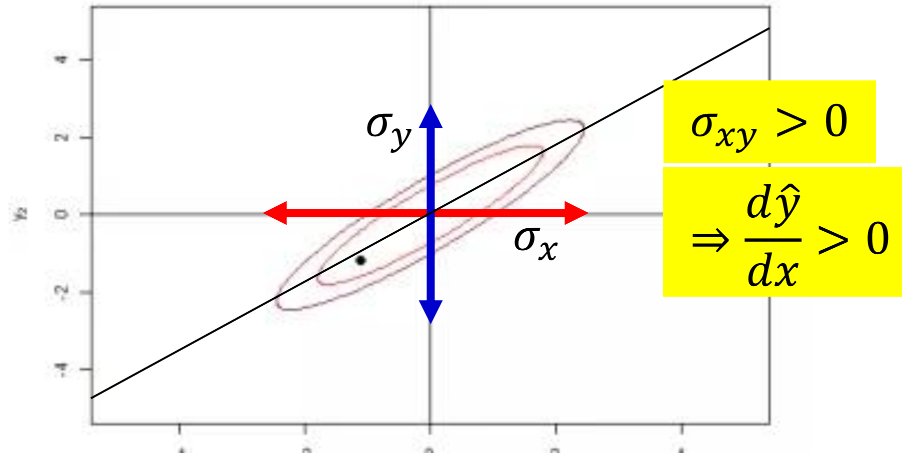
Class 19. 8 Nov 2016

Bhiksha Raj

Slides by Najim Dehak

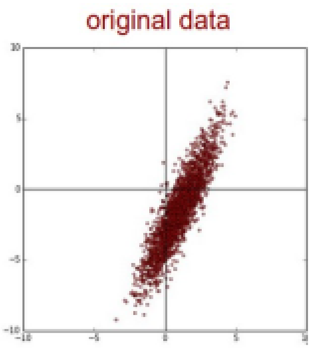
Definitions: Variance and Covariance





- Variance: $\Sigma_{XX} = E(XX^T)$, estimated as $\Sigma_{XX} = (1/N)\ \mathbf{XX}^T$
  - How "spread" is the data in the direction of X
  - Scalar version: $\sigma_x^2 = E(x^2)$
- Covariance: $\Sigma_{XY} = E(XY^T)$ estimated as $\Sigma_{XY} = (1/N)\ \mathbf{XY}^T$
  - How much does X predict Y
  - Scalar version: $\sigma_{xy} = E(x^2)$

Definition: Whitening Matrix



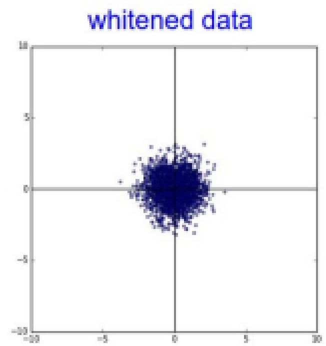
$P(X)$

$$\bar{X} = \sum_i X_i$$

$$\boxed{Z = \Sigma_{XX}^{-0.5}(X - \bar{X})}$$

If X is already centered

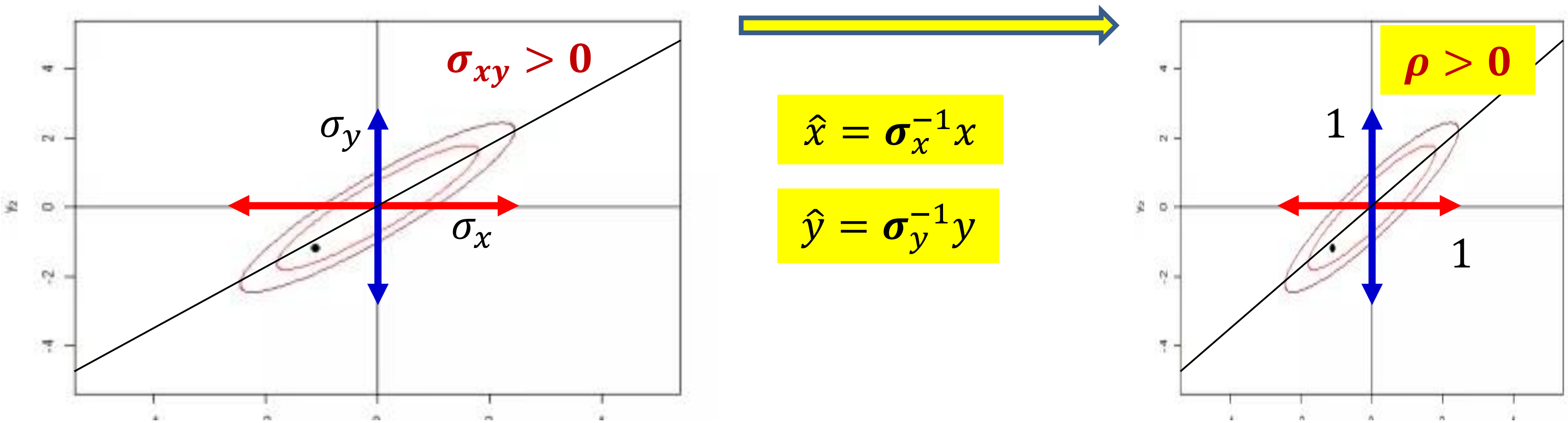
$$\boxed{Z = \Sigma_{XX}^{-0.5}X}$$



$P(Z)$

- Whitening matrix: $\Sigma_{XX}^{-0.5}$
- Transforms the variable to unit variance
- Scalar version: σ_x^{-1}

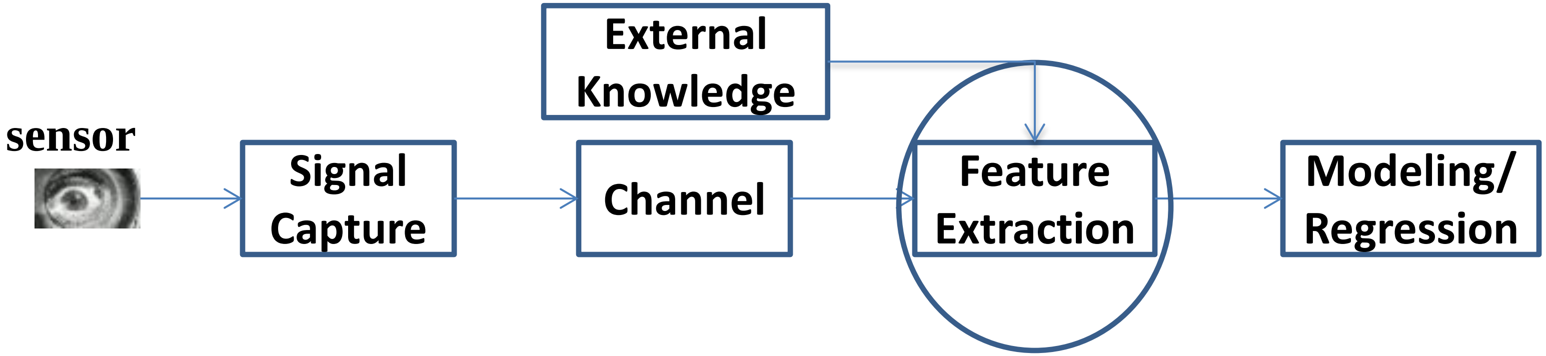
Definition: Correlation Coefficient



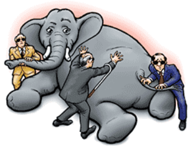
- Whitening matrix: $\Sigma_{XX}^{-0.5} \Sigma_{XY} \Sigma_{YY}^{-0.5}$
- Scalar version: $\rho_{xy} = \dfrac{\sigma_{xy}}{\sigma_y \sigma_y}$
  - Explains how Y varies with X, after *normalizing out* innate variation of X and Y

MLSP

- Application of Machine Learning techniques to the analysis of signals



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



Data specific bases?

- **Issue**: **The bases we have considered so far are *data agnostic***
  - Fourier / Wavelet type bases for all data may not be optimal
- **Improvement I**: **The bases we saw next were data specific**
  - PCA, NMF, ICA, ...
  - The bases changed depending on the data
- **Improvement II**: **What if bases are both data specific and task specific?**
  - Basis depends on both the data and a task

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

Recall: Unsupervised Basis Learning

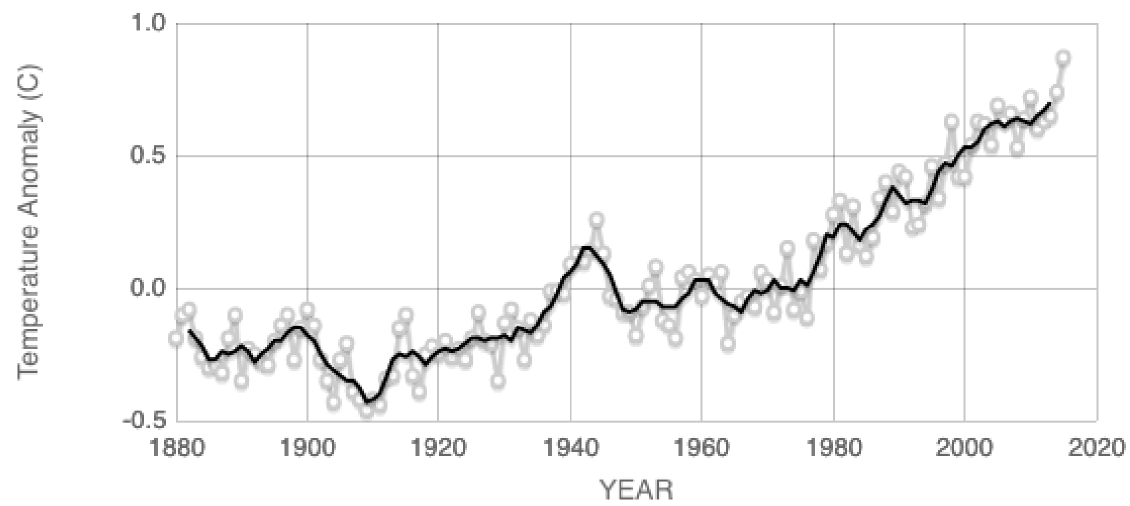
- What is a good basis?
  - Energy Compaction → Karkhonen-Loève
  - Uncorrelated → PCA
  - Sparsity → Sparse Representation, Compressed Sensing, …
  - Statistically Independent → ICA
- We create a narrative about how the data are created

Supervised Basis Learning?

- What is a good basis?
  - Basis that gives best classification performance
  - Basis that maximizes shared information with another ‘view’
- We have some external information guiding our notion of optimal basis
  - Can we learn a basis for a set of variables that will best predict some value(s)

Regression

- Simplest case
  - Given a bunch of scalar data points predict some value
  - Years are independent
  - Temperature is dependent



Source: climate.nasa.gov

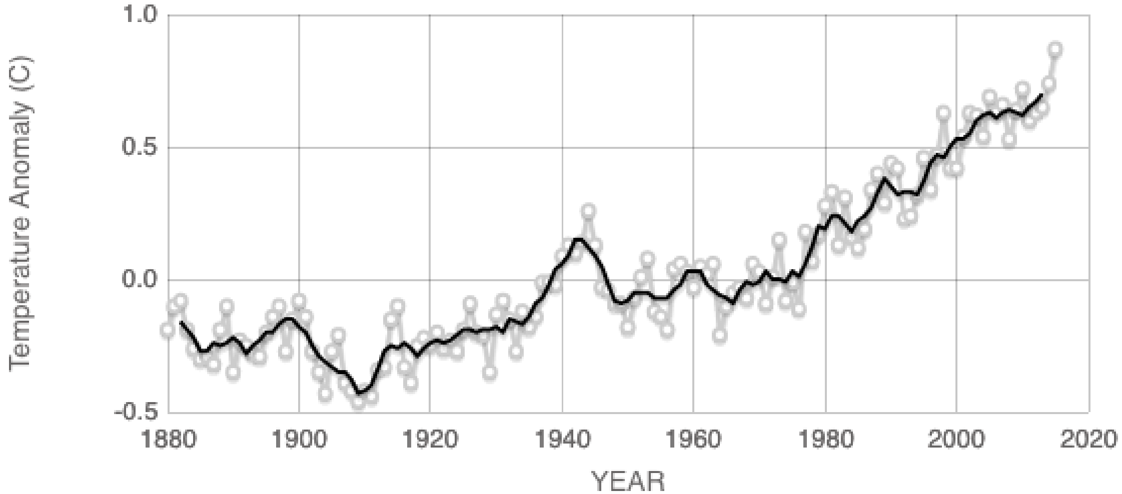
Regression

- Formulation of problem

$$\arg\min_{\beta_1,\beta_0} \sum_{i=1}^{N} (y_i - \beta_1 x_i - \beta_0)^2$$

$$= \arg\min_{\beta} \|Y - \beta^T X\|_F^2$$

- Let's solve!



Source: climate.nasa.gov

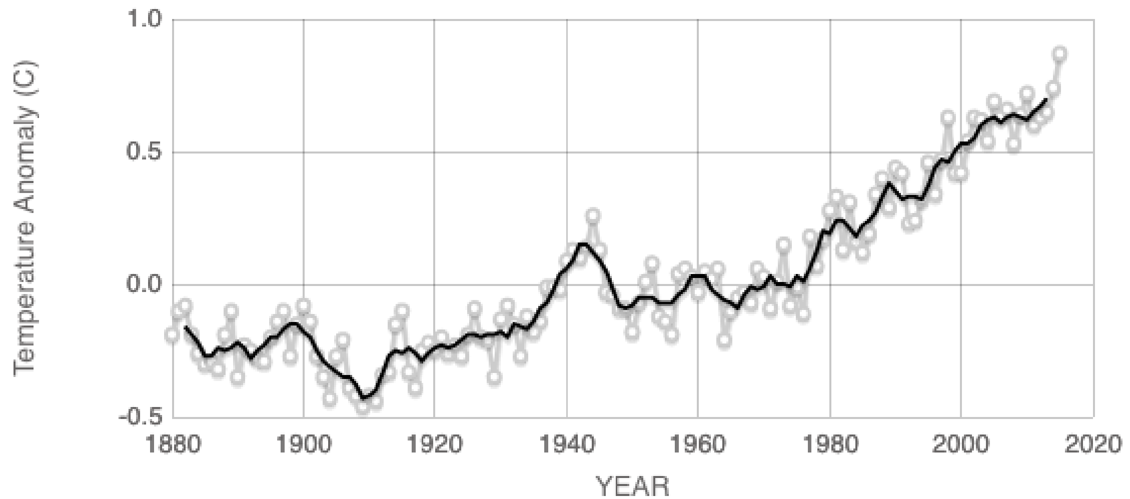
Regression

- Expand out the Frobenius norm

$$\arg\min_{\beta} \|Y - \beta^T X\|_F^2 = \arg\min_{\beta} Tr[(Y - \beta^T X)^T (Y - \beta^T X)]]$$

$$= \arg\min_{\beta} Tr(X^T \beta \beta^T X) - 2Tr(Y^T \beta^T X)$$

- Take derivative
- Solve for 0



Source: climate.nasa.gov

Regression

$$\nabla_\beta \; Tr(X^T\beta\beta^T X) - 2Tr(Y^T\beta^T X) = 2XX^T\beta - 2XY^T = 0$$

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

- This is just basically least squares again
- Note that this looks a lot like the following

$$\Sigma_{XX}^{-1}\Sigma_{XY}$$

  – In the 1-d case where x predicts y this is just …

$$\frac{Cov(X,Y)}{\sigma_X^2} = \rho\frac{\sigma_Y}{\sigma_X}$$

Multiple Regression

- Robot Archer Example
  - Our 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

$$Y_i = \beta X_i$$

Multiple Regression

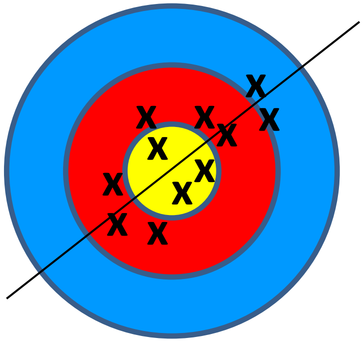
- Answer

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

  - Here Y contains measurements of the distance of the arrow from the center
  - We are fitting a plane
  - Correlation is basically just the gradient

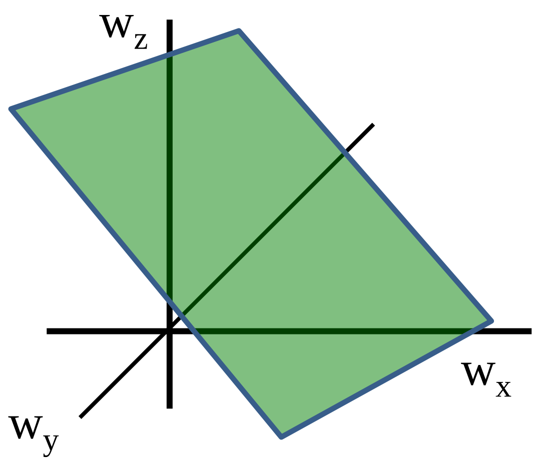
Canonical Correlation Analysis

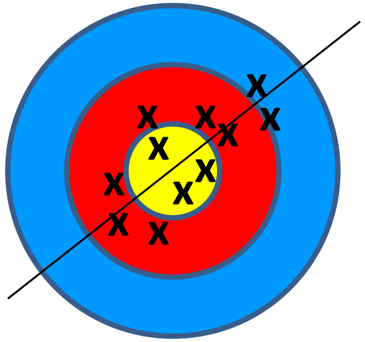
- Further Generaliztion (CCA)
  - Do *all* wind factors affect the position
    - Or just some low-dimensional combinations $\hat{X} = AX$
  - Do they affect both coordinates individually
    - Or just some of combination $\hat{y} = BY$



Canonical Correlation Analysis

- Let’s call the arrow location vector Y and the wind vectors X
  - Let’s find the projection of the vectors for Y and X respectively that are most correlated

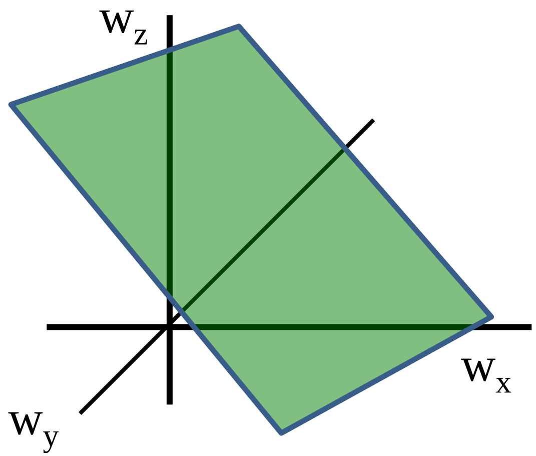


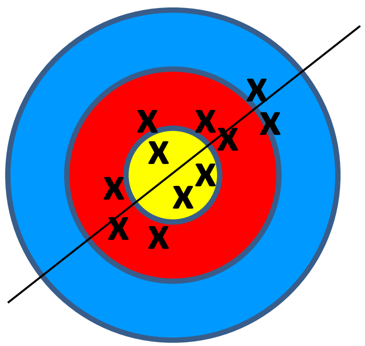


Best X projection plane ⇒ Predicts best Y projection

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!



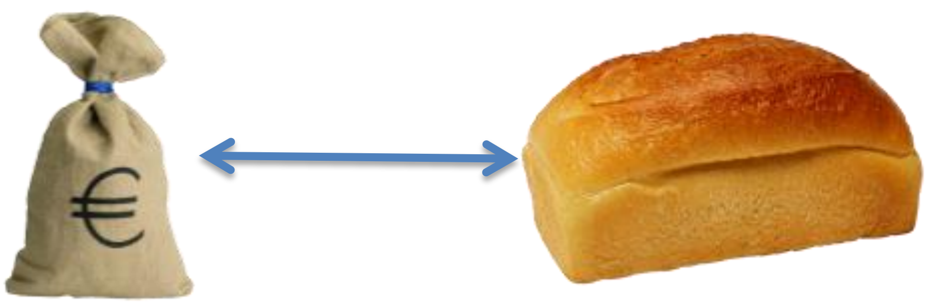


Best X projection plane → Predicts best Y projection

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

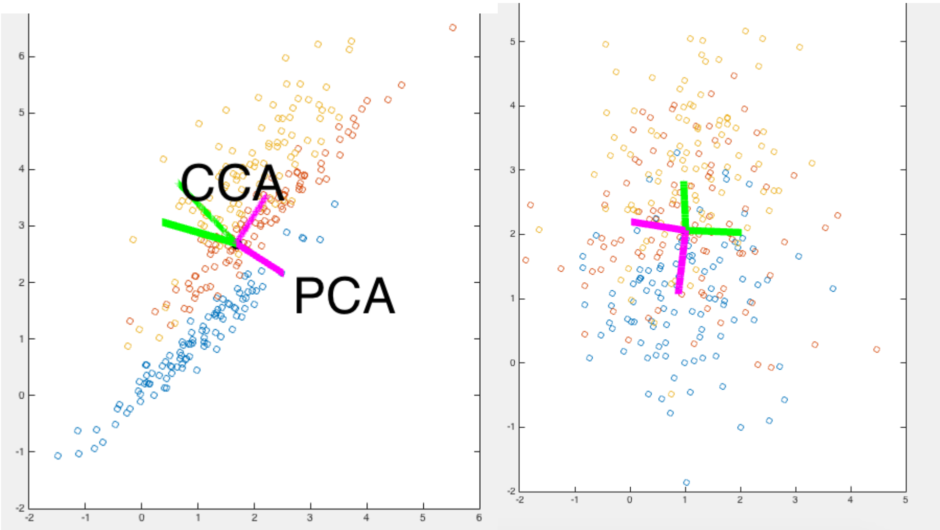
X = Prices



Y = Consumption

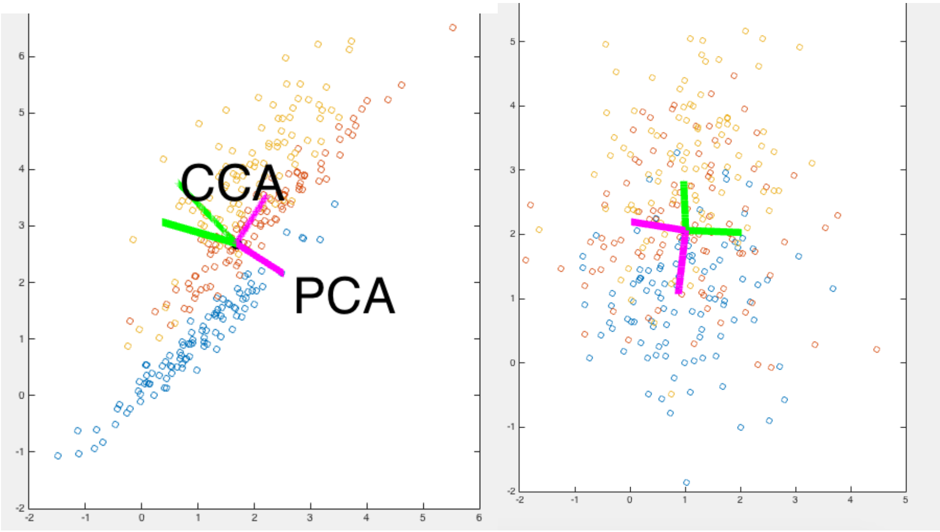
CCA Motivation and History

- 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

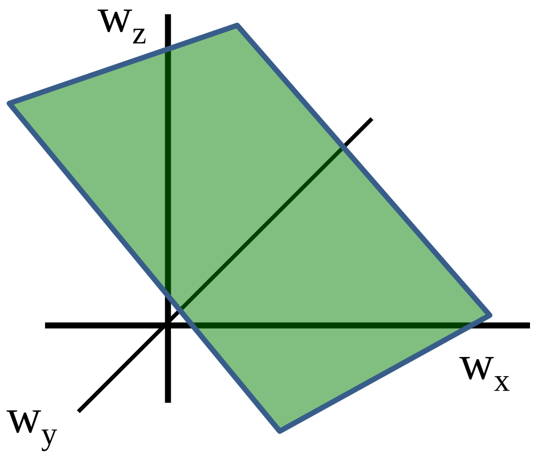
- Why did CCA work?
  - Soft supervision
    - External Knowledge
  - The 2 views track each other in a direction that does not correspond to noise
  - Noise suppression (sometimes)
- Preview
  - If one of the sets of signals are true labels, CCA is equivalent to Linear Discriminant Analysis
  - Hard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

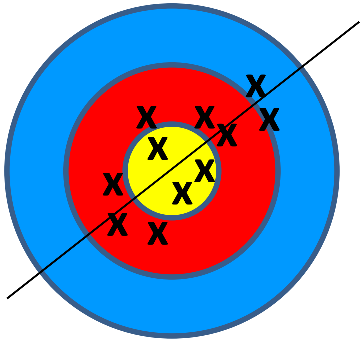
Multiview Assumption

- 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

Multiview Assumption

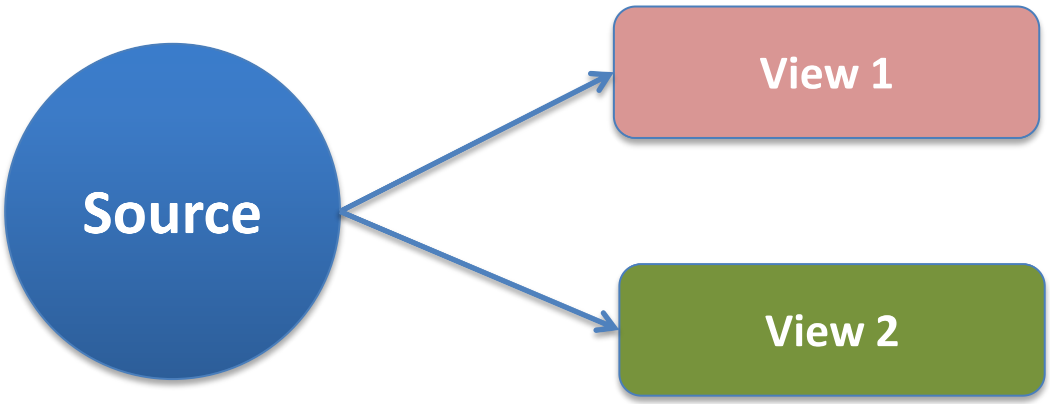
- Why not just concatenate both views?
  - It does not exploit the extra structure of the signal (more on this in 2 slides)
    - PCA on joint data will decorrelate *all variables*
      - Not good for prediction
    - We want to decorrelate X and Y, but maximize cross-correlation between X and Y
  - High dimensionality → over-fit





Multiview Assumption

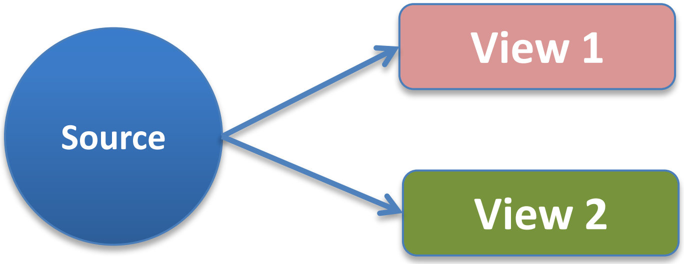
- We can sort of think of a model for how our data might be generated



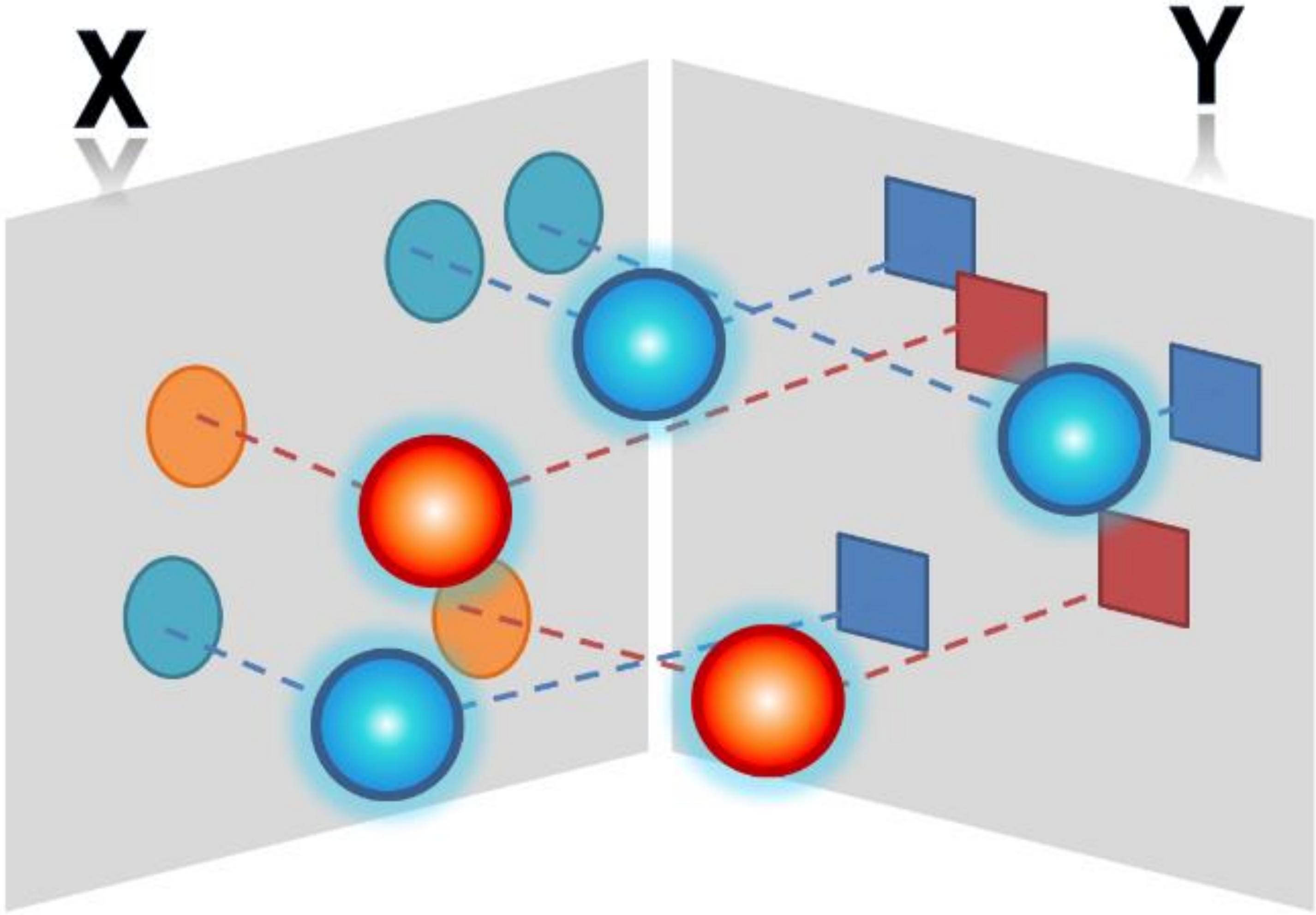
- We want View 1 independent of View 2 conditioned on knowledge of the source
  - All correlation is due to source

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



Multiview Examples



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

Matrix Representation

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

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

$$\|\mathbf{X}\|_F^2 = \sum_i X_i^T X_i = trace\ \mathbf{X}\mathbf{X}^T$$

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

- Expressing total error as a matrix operation

Recall: Objective Functions

- **Least Squares**

$$\arg\min_{Y \in \mathbb{R}^{kxN}} \|X - UY\|_F \quad s.t. \quad U \in \mathbb{R}^{dxk} \quad rank(U) = k$$

- **What is a good basis?**
  - **Energy Compaction → Karkhonen-Loève**

$$\arg\min_{Y \in \mathbb{R}^{kxN}, U \in \mathbb{R}^{dxk}} \|X - UY\|_F \quad s.t. \quad U^T U = I_k$$

  - **Positive Sparse → NMF**

$$\arg\min_{Y \in \mathbb{R}^{kxN}, U \in \mathbb{R}^{dxk}} \|X - UY\|_F \quad s.t. \quad U, Y \geq 0$$

  - **Regression**

$$\arg\min_{\beta} \|Y - \beta^T X\|_F^2$$

A Quick Review

- Cross Covariance

$$\mathbb{E}\left[\begin{bmatrix} X \\ Y \end{bmatrix}\begin{bmatrix} X \\ Y \end{bmatrix}^T\right] \approx \frac{1}{N}\begin{bmatrix} X \\ Y \end{bmatrix}\begin{bmatrix} X \\ Y \end{bmatrix}^T$$

$$= \begin{bmatrix} C_{xx} & C_{xy} \\ C_{yx} & C_{yy} \end{bmatrix}$$

A Quick Review

- The effect of a transform

$$Z = UX$$

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

$$C_{ZZ} = E[ZZ^T] = UC_{XX}U^T$$

Recall: Objective Functions

- So far our objective needs to external data
  - No knowledge of task

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

$$s.t.\ \ U \in \mathbb{R}^{d \times k}$$
$$rank(U) = k$$

- CCA requires an extra view
  - We force both views to look like each other

$$\min_{U \in \mathbb{R}^{d_x \times k},\ V \in \mathbb{R}^{d_y \times k}} \|U^T \mathbf{X} - Y^T \mathbf{Y}\|_F^2$$

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

Interpreting the CCA Objective

- Minimize the reconstruction error between the projections of both views of data
- Find the subspaces U,V onto which we project views X and Y such that their correlation is maximized
- Find combinations of both views that best predict each other

A Quick Review

- Cross Covariance

$$\mathbb{E}[\begin{bmatrix} X \\ Y \end{bmatrix} \begin{bmatrix} X \\ Y \end{bmatrix}^T] \approx \frac{1}{N} \begin{bmatrix} X \\ Y \end{bmatrix} \begin{bmatrix} X \\ Y \end{bmatrix}^T$$

$$= \begin{bmatrix} C_{xx} & C_{xy} \\ C_{yx} & C_{yy} \end{bmatrix}$$

A Quick Review

- Matrix representation

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

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

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

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

Interpreting the CCA Objective

- CCA maximizes correlation between two views
- While keeping individual views uncorrelated
  - Uncorrelated measurements are easy to model

$$\min_{U\in\mathbb{R}^{\boldsymbol{d_x}\times\boldsymbol{k}},\ V\in\mathbb{R}^{\boldsymbol{d_y}\times\boldsymbol{k}}} \|U^T\mathbf{X} - Y^T\mathbf{Y}\|_F^2$$

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

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

CCA Derivation

$$\min_{U \in \mathbb{R}^{d_x \times k},\ V \in \mathbb{R}^{d_y \times k}} \|U^T\mathbf{X} - Y^T\mathbf{Y}\|_F^2$$

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

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

- Assume $C_{XX},\ C_{XX}$ are invertible
- Create the Lagrangian and differentiate

CCA Derivation

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

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

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

- So we can solve the equivalent problem below

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

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

CCA Derivation

- Incorporating Lagrangian, maximize

$$\begin{aligned}\mathcal{L}(\Lambda_X, \Lambda_Y) \\ &= tr(U^T\mathbf{X}\mathbf{Y}^T V) \\ &- tr\big(\big((U^T\mathbf{X}\mathbf{X}^T U) - I_k\big)\Lambda_X\big) \\ &- tr\big(\big((V^T\mathbf{Y}\mathbf{Y}^T V) - I_k\big)\Lambda_Y\end{aligned}$$

- Remember that the constraints matrices are symmetric

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

$$C_{XX}^{-1} C_{XY} C_{YY}^{-1} C_{YX} \tilde{U} = \tilde{U} D^2$$

$$C_{YY}^{-1} C_{YX} C_{XX}^{-1} C_{XY} \tilde{V} = \tilde{V} D^2$$

CCA Derivation

- We just have to find eigenvectors for

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

- 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 as Generalized Eigenvalue Problem

- Rayleigh Quotient

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

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

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

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

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

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

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

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

$$\max_{U,V} trace(U^T\mathbf{XY}^TV)$$

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

- The previous solution still applies but with slightly different autocovariance matrices

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

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

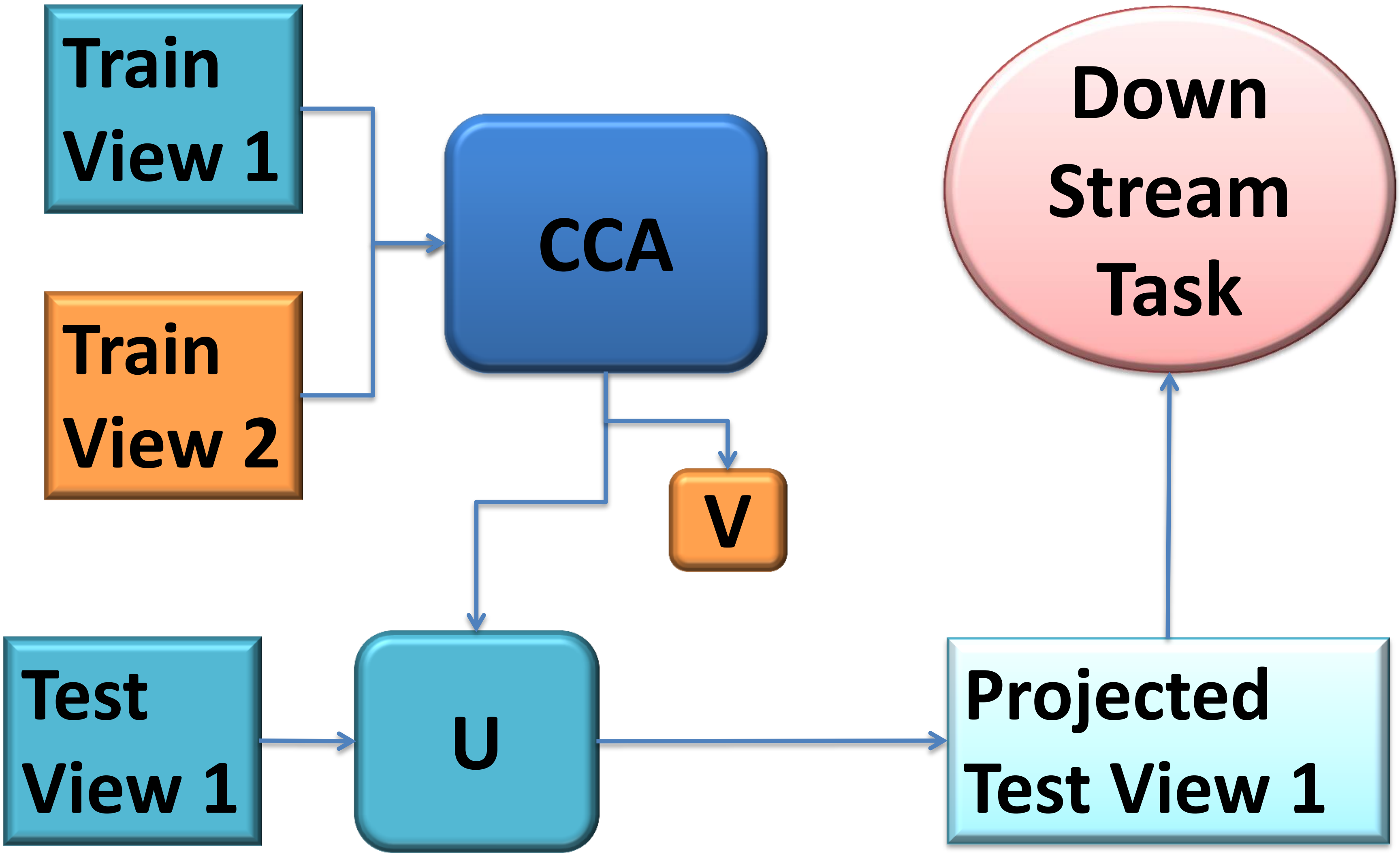
- We note that the matrix is symmetric and
- So the problem is solved by SVD on the matrix M

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

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



Johns Hopkins Whiting School of Engineering

Multiview Setup

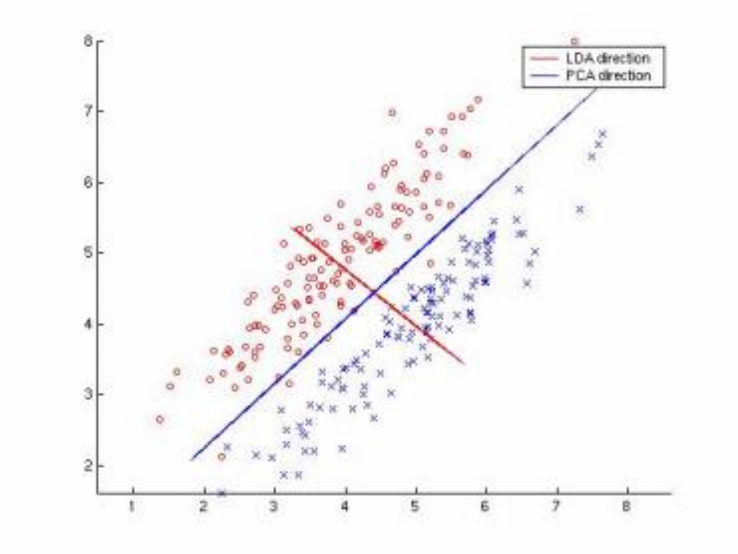
- Often one view consists of measurements that are very hard to collect
  - Speakers all saying the same sentence
  - Articulatory measurements along with speech
  - Odd camera angles
  - Etc.

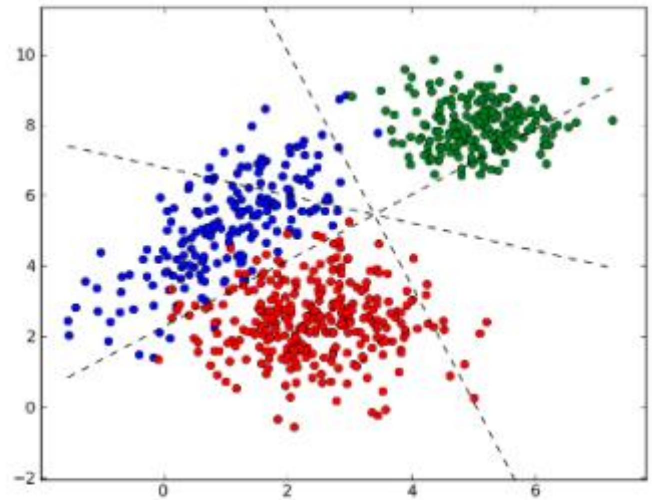
Multiview Setup

- 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/

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

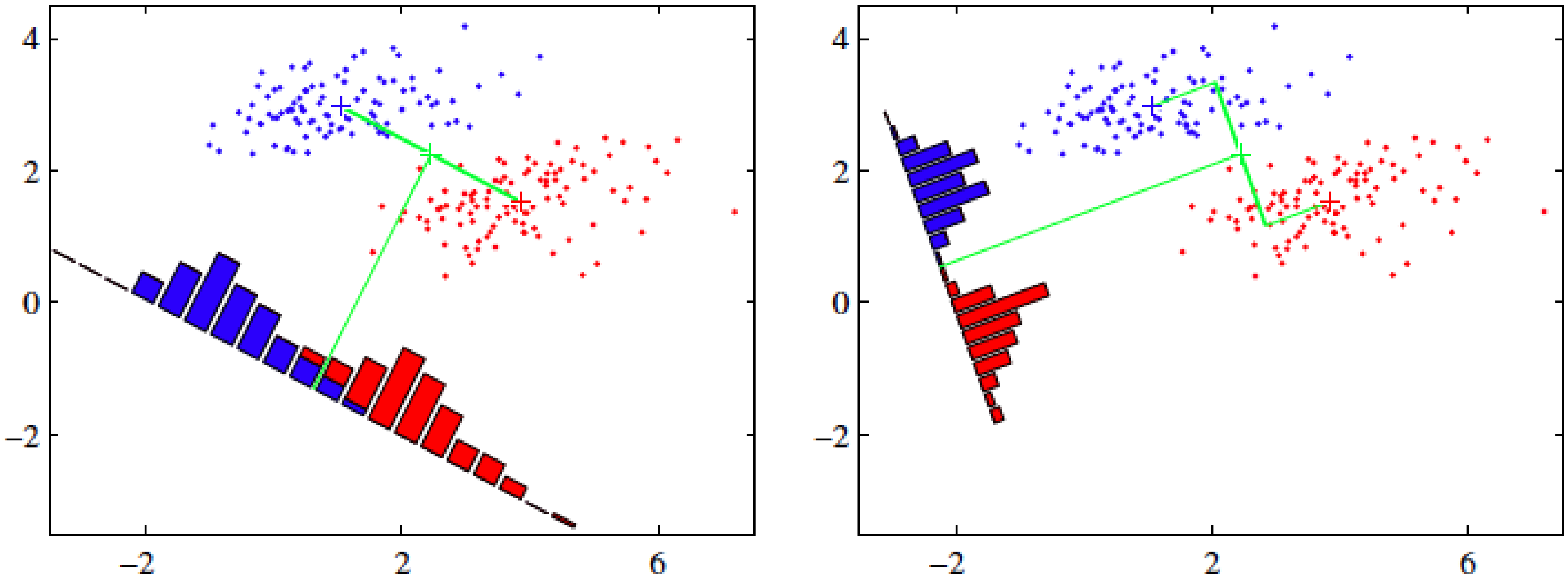
- We have 2 views as in CCA
- What if one view is the true labels for the task at hand?
  - Learn the direction that is maximally correlated with the right answers!
- It turns out that LDA and CCA are equivalent when the situation above is true

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

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

    - 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
  – In both cases we are finding covariance type functions of transformations of Random Vectors
    - What is the covariance of $Y = W^T X$?

LDA Formulation

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

$$\underset{W \in \mathbb{R}^{dxk}}{\arg\max}\, Tr\ \left(W^T S_B W\right)\ \ s.t.\ \ W^T S_W W = I$$

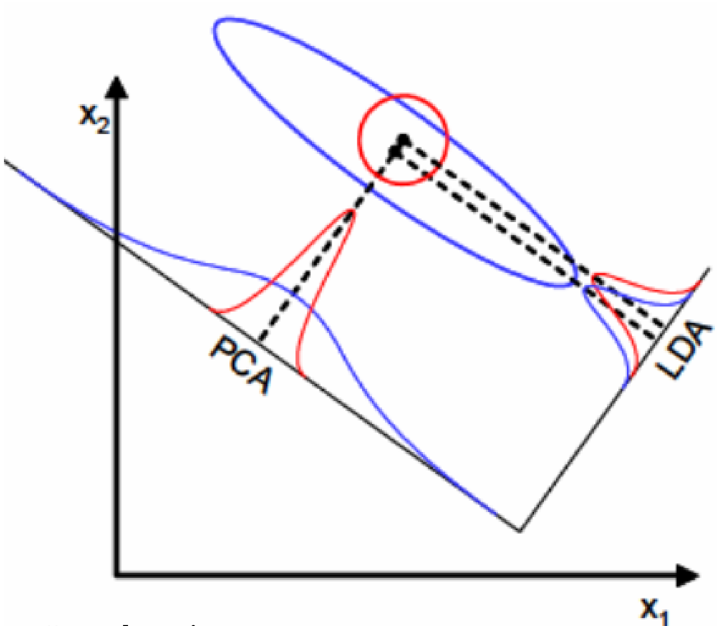
- The Lagrangian is …

$$\mathcal{L}(\Lambda) = \underset{W \in \mathbb{R}^{dxk}}{\arg\max}\, Tr\ \left(W^T S_B W\right) - Tr((W^T S_W W - I)\Lambda)$$

- So we see that we have a generalized eigenvalue solution

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

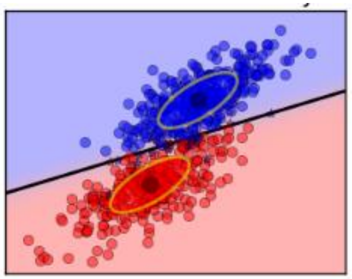
LDA as Classifier

- 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)

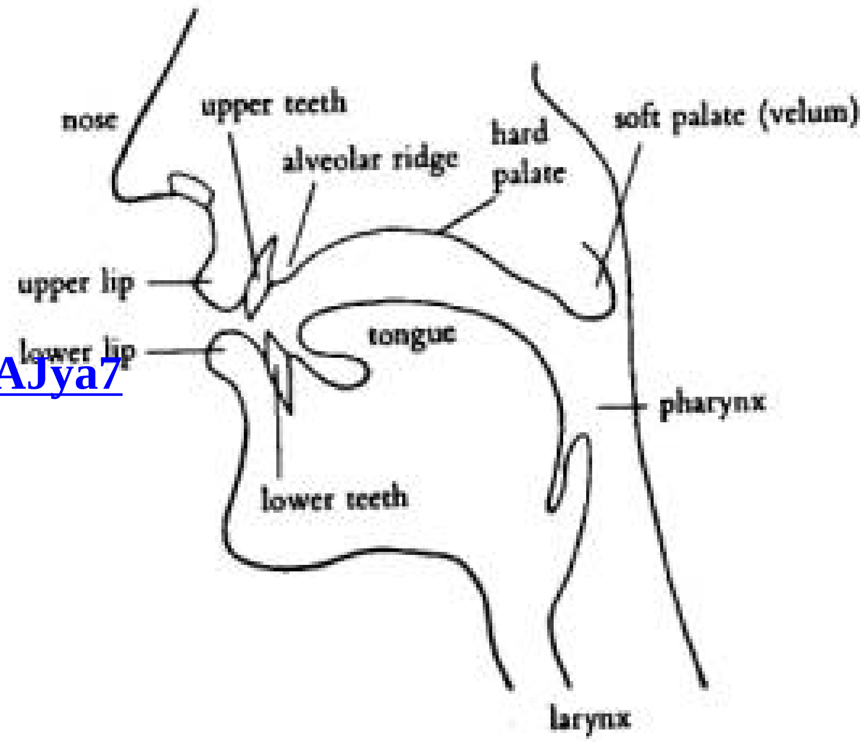
  - We end up with linear decision surfaces between classes

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



Bakeoff – PCA, CCA, LDA on Vowel Classification

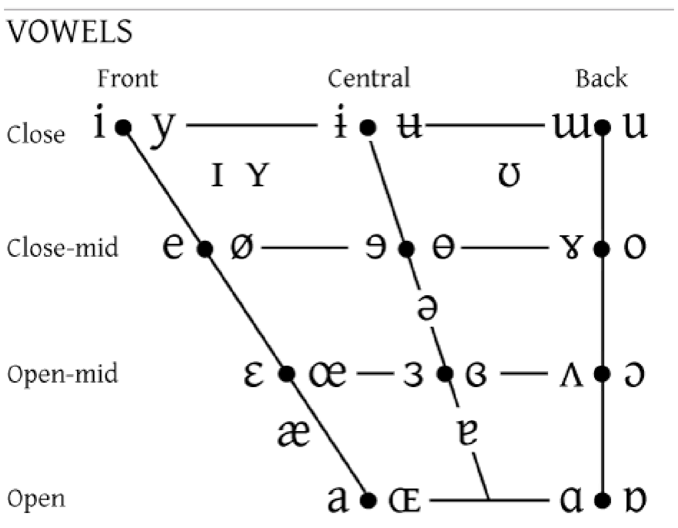
- Speech is produced by an excitation in the glottis (vocal folds)
- Sound is then shaped with the tongue, teeth, soft palate …
- This shaping is what generates the different vowels



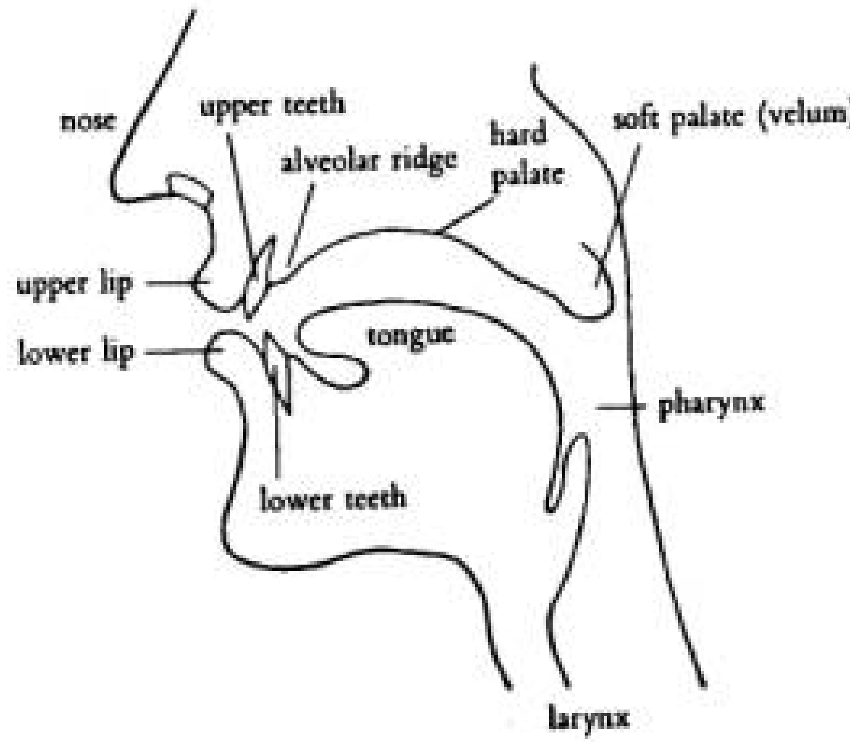
https://www.youtube.com/watch?v=58AJya7JzOU#t=00m36s

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



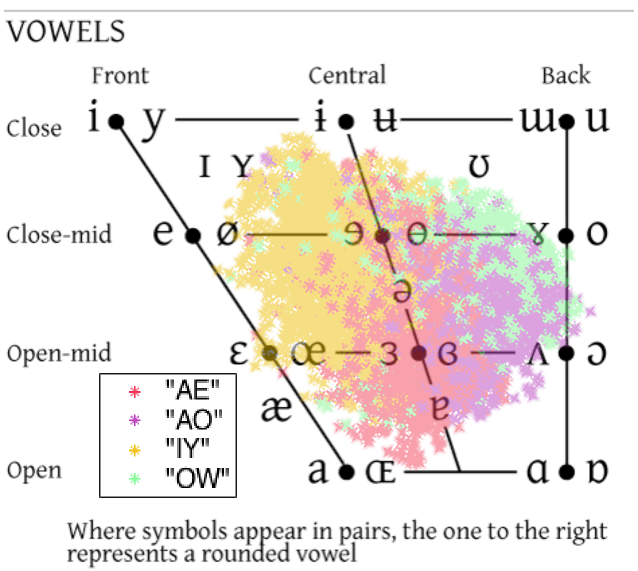
Where symbols appear in pairs, the one to the right represents a rounded vowel



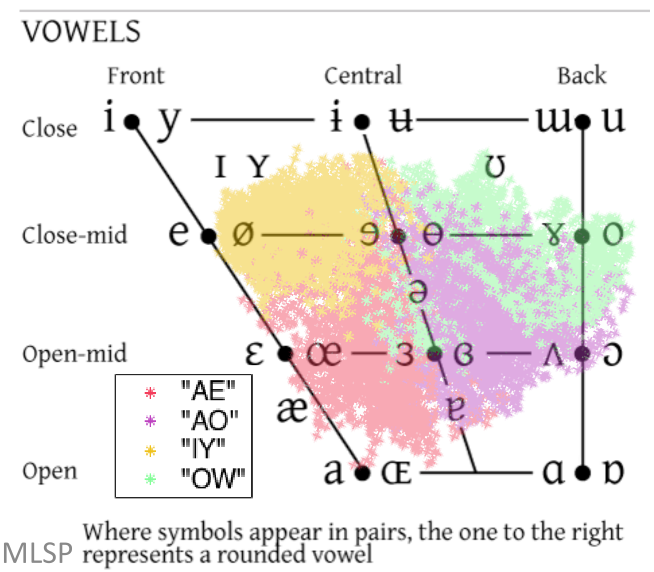
Bakeoff – PCA, CCA, LDA on Vowel Classification

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

PCA



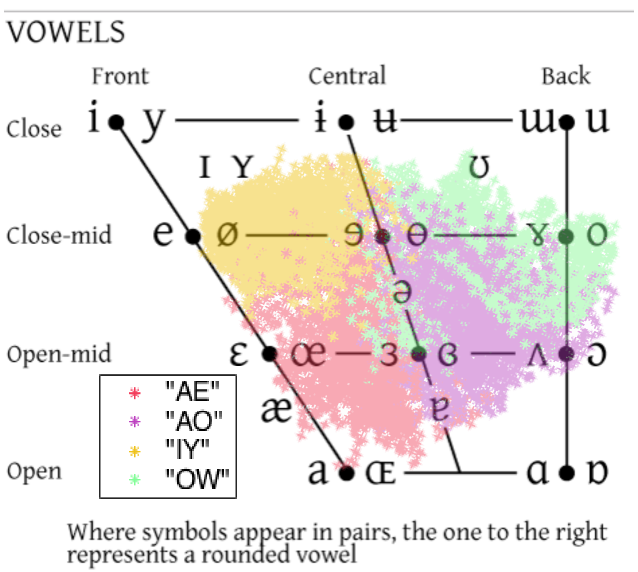
CCA



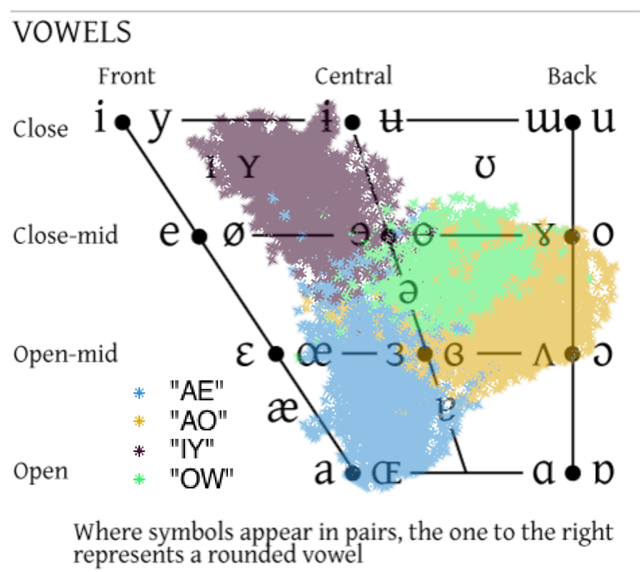
Bakeoff – PCA, CCA, LDA on Vowel Classification

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

CCA

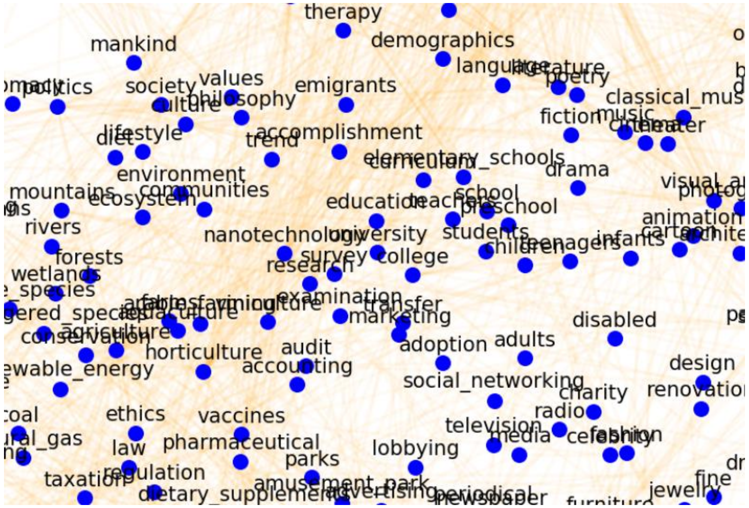


LDA



Multilingual CCA

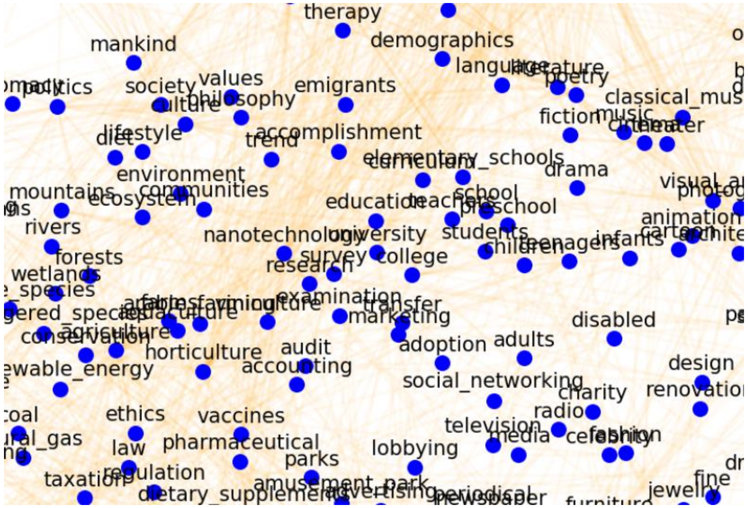
- Another Example of CCA
  - Word is mapped into some vector space
  - A notion of distance between words is defined and the mapping is such that words that are semantically similar are mapped to near to each other (hopefully)



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

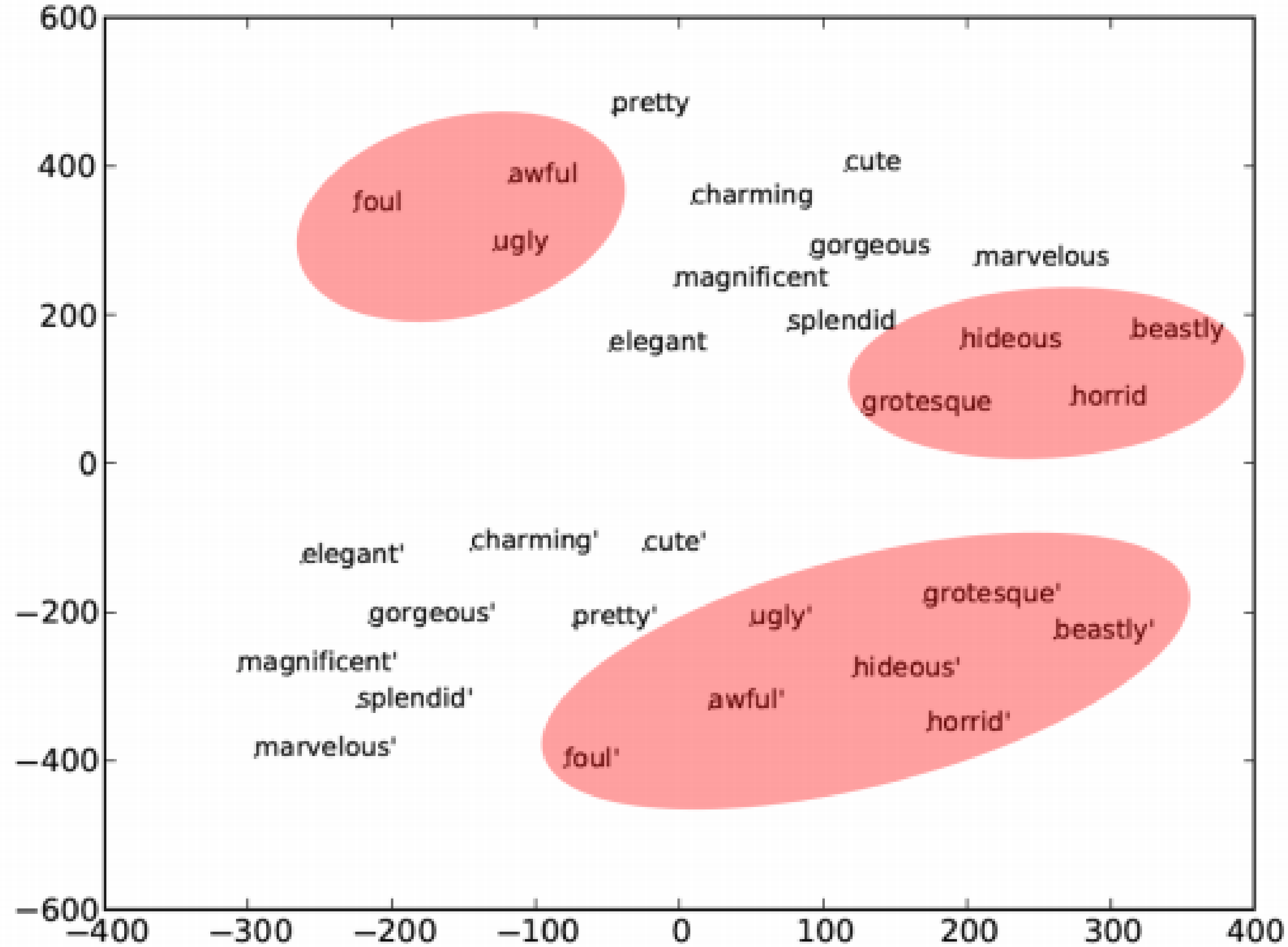
Multilingual CCA

- 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