ECE598: Representation Learning Fall 2017

Lecture 2: Canonical Component Analysis

Lecturer: Pramod Viswanath Scribe: Bryan Clifford and Kaiqing Zhang, Sept. 8, 2017

2.1 Problem Setup

- We're given 2 data sets each consisting of N samples

$$\{\boldsymbol{x}_n\}_{n=1}^N \text{ and } \{\boldsymbol{y}_n\}_{n=1}^N.$$

- The data are two totally different measurement modalities of the same hidden (latent) variable. For example, $\boldsymbol{x}_n$ could be a video (vectorized) of a person saying a word and $\boldsymbol{y}_n$ could be the audio from the video, and the latent variable is the word the person is saying. Hence $\boldsymbol{x}_n$ and $\boldsymbol{y}_n$ live in different dimensional spaces:

$$\boldsymbol{x}_n \in \mathbb{R}^{d_x}, \qquad \boldsymbol{y}_n \in \mathbb{R}^{d_y} \qquad \forall n.$$

- We will organize the data into row matrices

$$\begin{aligned} \boldsymbol{X} &= (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N)^T \in \mathbb{R}^{N \times d_x} \\ \boldsymbol{Y} &= (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_N)^T \in \mathbb{R}^{N \times d_y} \end{aligned} \tag{2.1}$$

2.1.1 Goal:

Our goal is to find linear transforms for $\boldsymbol{x}_n$ and $\boldsymbol{y}_n$ that we can use to reduce the dimensionality of each data set to the dimension that we think the latent variable is. However, because both data sets are generated from the same latent variable, we want the transforms to be such that the transformed $\boldsymbol{x}$s and $\boldsymbol{y}$s are maximally covariant. Furthermore we want the transformed $\boldsymbol{x}$s to be uncorrelated from themselves (and the same for the transformed $\boldsymbol{y}$s).

This approach is called **Canonical Component/Correlation Analysis (CCA)**. We'll call the transformed variables the **canonical variates** and we'll call the correlations between pairs of transformed variables the **canonical correlations**.

More formally we want to find vectors $\{\boldsymbol{\alpha}_m\}_{m=1}^{d_z} \subset \mathbb{R}^{d_x}$ and $\{\boldsymbol{\beta}_m\}_{m=1}^{d_z} \subset \mathbb{R}^{d_y}$ that solve the problem

$$\begin{aligned} \max_{\substack{\{\boldsymbol{\alpha}_m\}_{m=1}^{d_z} \\ \{\boldsymbol{\beta}_m\}_{m=1}^{d_z}}} & \frac{\mathrm{E}\left[(\boldsymbol{x}^T\boldsymbol{\alpha}_m)(\boldsymbol{y}^T\boldsymbol{\beta}_m)\right]}{\sqrt{\mathrm{Var}\left[\boldsymbol{x}^T\boldsymbol{\alpha}_m\right]\mathrm{Var}\left[\boldsymbol{y}^T\boldsymbol{\beta}_m\right]}} \\ \text{s.t.} \quad & \mathrm{E}\left[(\boldsymbol{x}^T\boldsymbol{\alpha}_m)(\boldsymbol{x}^T\boldsymbol{\alpha}_k)\right] = \mathrm{E}\left[(\boldsymbol{y}^T\boldsymbol{\beta}_m)(\boldsymbol{y}^T\boldsymbol{\beta}_k)\right] = \delta_{m,k}. \end{aligned} \tag{2.2}$$

These vectors $\{\boldsymbol{\alpha}_m\}_{m=1}^{d_z} \subset \mathbb{R}^{d_x}$ and $\{\boldsymbol{\beta}_m\}_{m=1}^{d_z} \subset \mathbb{R}^{d_y}$ are called the **canonical directions**, and d_z is the dimension of the latent variable space.

2.2 Solution

This section shows that the solution to the CCA problem (2.2) is to first whiten the data, and then compute the SVD of the (now whitened) cross-correlation matrix.

The approach to deriving the solution will be to replace the population mean and variances with the sample mean and variances and then use some linear algebra tricks.

First we will assume that the data is centered (i.e. $\mathrm{E}[\boldsymbol{x}] = \mathbf{0}$, and $\mathrm{E}[\boldsymbol{y}] = \mathbf{0}$). This can be done by simply subtracting the sample means.

The correlation in the top of the optimization problem (2.2) can be replaced by

$$\mathrm{E}\left[(\boldsymbol{x}^T\boldsymbol{\alpha}_m)(\boldsymbol{y}^T\boldsymbol{\beta}_m)\right] \to \frac{1}{N}(\boldsymbol{X}\boldsymbol{\alpha}_m)^T(\boldsymbol{Y}\boldsymbol{\beta}_m) \tag{2.3}$$

and because the data are centered, the variances can be replaced by

$$\begin{aligned}\mathrm{Var}\left[\boldsymbol{x}^T\boldsymbol{\alpha}_m\right] \to &(\boldsymbol{X}\boldsymbol{\alpha}_m)^T(\boldsymbol{X}\boldsymbol{\alpha}_m) = \frac{1}{N}\|\boldsymbol{X}\boldsymbol{\alpha}_m\|_2^2 \\ \mathrm{Var}\left[\boldsymbol{y}^T\boldsymbol{\beta}_m\right] \to &(\boldsymbol{Y}\boldsymbol{\beta}_m)^T(\boldsymbol{X}\boldsymbol{\beta}_m) = \frac{1}{N}\|\boldsymbol{Y}\boldsymbol{\beta}_m\|_2^2.\end{aligned} \tag{2.4}$$

Substituting this into (2.2) we have

$$\begin{aligned}&\max_{\substack{\{\boldsymbol{\alpha}_m\}_{m=1}^{d_z} \\ \{\boldsymbol{\beta}_m\}_{m=1}^{d_z}}} \frac{(\boldsymbol{X}\boldsymbol{\alpha}_m)^T(\boldsymbol{Y}\boldsymbol{\beta}_m)}{\|\boldsymbol{X}\boldsymbol{\alpha}_m\|_2\|\boldsymbol{Y}\boldsymbol{\beta}_m\|_2} \\ \text{s.t.} \quad &(\boldsymbol{X}\boldsymbol{\alpha}_m)^T(\boldsymbol{X}\boldsymbol{\alpha}_k) = (\boldsymbol{Y}\boldsymbol{\beta}_m)^T(\boldsymbol{Y}\boldsymbol{\beta}_k) = \delta_{m,k}.\end{aligned} \tag{2.5}$$

Now, to make this easier we want to find a transformation on the $\boldsymbol{\alpha}$s and $\boldsymbol{\beta}$s such that we can simplify the denominator. This is where whitening comes in. The method is the same for both the $\boldsymbol{\alpha}$s and $\boldsymbol{\beta}$s as follows

$$\begin{aligned}\tilde{\boldsymbol{\alpha}}_m = \boldsymbol{A}^{1/2}\boldsymbol{\alpha}_m \to &\boldsymbol{\alpha} = \boldsymbol{A}^{-1/2}\tilde{\boldsymbol{\alpha}}_m \\ \tilde{\boldsymbol{\beta}}_m = \boldsymbol{B}^{1/2}\boldsymbol{\beta}_m \to &\boldsymbol{\beta} = \boldsymbol{B}^{-1/2}\tilde{\boldsymbol{\beta}}_m\end{aligned} \tag{2.6}$$

where

$$\begin{aligned}\boldsymbol{A}^{-1/2} =&(\boldsymbol{X}^T\boldsymbol{X})^{-1/2} = \boldsymbol{V}_x\,\mathrm{diag}\left\{1/\boldsymbol{\lambda}_x\right\}\boldsymbol{V}_y^T \\ \boldsymbol{B}^{-1/2} =&\boldsymbol{Y}^T\boldsymbol{Y} = \boldsymbol{V}_y\,\mathrm{diag}\left\{1/\boldsymbol{\lambda}_y\right\}\boldsymbol{V}_y^T\end{aligned} \tag{2.7}$$

and where $\boldsymbol{V}_x$, $\boldsymbol{\lambda}_x$ are the matrix of orthonormal eigen vectors and values of $\boldsymbol{X}^T\boldsymbol{X}$ respectively, and similarly for $\boldsymbol{V}_y$, $\boldsymbol{\lambda}_y$. Note also that $\boldsymbol{A}^{-1/2}$ and $\boldsymbol{B}^{-1/2}$ are symmetric.

With this definition we see that

$$\begin{aligned}\|\boldsymbol{X}\boldsymbol{\alpha}_m\|_2^2 =&\boldsymbol{\alpha}_m^T\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\alpha}_m \\ =&\tilde{\boldsymbol{\alpha}}_m^T\boldsymbol{A}^{-1/2}\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{A}^{-1/2}\tilde{\boldsymbol{\alpha}}_m \\ =&\tilde{\boldsymbol{\alpha}}_m^T\boldsymbol{A}^{-1/2}\boldsymbol{A}\boldsymbol{A}^{-1/2}\tilde{\boldsymbol{\alpha}}_m \\ =&\tilde{\boldsymbol{\alpha}}_m^T\boldsymbol{A}^{-1/2}\boldsymbol{A}^{1/2}\boldsymbol{A}^{1/2}\boldsymbol{A}^{-1/2}\tilde{\boldsymbol{\alpha}}_m \\ =&\|\tilde{\boldsymbol{\alpha}}_m\|_2^2\end{aligned} \tag{2.8}$$

and similarly for $\|\boldsymbol{Y}\boldsymbol{\beta}_m\|_2^2$.

Making the substitutions into (2.5) we have

$$\begin{aligned} \max_{\substack{\{\tilde{\boldsymbol{\alpha}}_m\}_{m=1}^{d_z} \\ \{\tilde{\boldsymbol{\beta}}_m\}_{m=1}^{d_z}}} & \frac{\tilde{\boldsymbol{\alpha}}_m^T \boldsymbol{A}^{-1/2} \boldsymbol{X}^T \boldsymbol{Y} \boldsymbol{B}^{-1/2} \tilde{\boldsymbol{\beta}}_m}{\|\tilde{\boldsymbol{\alpha}}_m\|_2 \|\tilde{\boldsymbol{\beta}}_m\|_2} \\ \text{s.t.} \quad & \tilde{\boldsymbol{\alpha}}_m^T \tilde{\boldsymbol{\alpha}}_k = \tilde{\boldsymbol{\beta}}_m^T \tilde{\boldsymbol{\beta}}_k = \delta_{m,k}. \end{aligned} \tag{2.9}$$

By replacing the denominator with a unit norm constraint the problem becomes

$$\begin{aligned} \max_{\substack{\{\tilde{\boldsymbol{\alpha}}_m\}_{m=1}^{d_z} \\ \{\tilde{\boldsymbol{\beta}}_m\}_{m=1}^{d_z}}} & \tilde{\boldsymbol{\alpha}}_m^T \boldsymbol{A}^{-1/2} \boldsymbol{X}^T \boldsymbol{Y} \boldsymbol{B}^{-1/2} \tilde{\boldsymbol{\beta}}_m \\ \text{s.t.} \quad & \tilde{\boldsymbol{\alpha}}_m^T \tilde{\boldsymbol{\alpha}}_k = \tilde{\boldsymbol{\beta}}_m^T \tilde{\boldsymbol{\beta}}_k = \delta_{m,k}, \quad \|\tilde{\boldsymbol{\alpha}}_m\|_2 = \|\tilde{\boldsymbol{\beta}}_m\|_2 = 1. \end{aligned} \tag{2.10}$$

The solution to (2.10) should be obvious. It is the SVD of the matrix $\boldsymbol{A}^{-1/2} \boldsymbol{X}^T \boldsymbol{Y} \boldsymbol{B}^{-1/2}$, and the $\tilde{\boldsymbol{\alpha}}$s and $\tilde{\boldsymbol{\beta}}$s are the left and right singular vectors! The canonical correlations are then given by the singular values $\sigma_1 \geq \sigma_2 \cdots \geq \sigma_{d_z}$.

Once the $\tilde{\boldsymbol{\alpha}}$s and $\tilde{\boldsymbol{\beta}}$s are known, we can compute the canonical directions from (2.6).

2.3 Probabilistic Interpretation

CCA has a very interesting probabilistic interpretation that was proven independently by [Bro79, BJ06]. The idea in both papers is that $\boldsymbol{x}$ and $\boldsymbol{y}$ are random variables generated by a latent variable $\boldsymbol{z} \sim \mathcal{N}(0, 1)$ with some additive noise $\boldsymbol{\epsilon}_x \sim \mathcal{N}(0, \Psi_x)$, $\boldsymbol{\epsilon}_y \sim \mathcal{N}(0, \Psi_y)$, as follows

$$\begin{aligned} \boldsymbol{x} &= \boldsymbol{W}_x \boldsymbol{z} + \boldsymbol{\mu}_x + \boldsymbol{\epsilon}_x \\ \boldsymbol{y} &= \boldsymbol{W}_y \boldsymbol{z} + \boldsymbol{\mu}_y + \boldsymbol{\epsilon}_y \end{aligned} \tag{2.11}$$

where $\boldsymbol{\mu}_x$ and $\boldsymbol{\mu}_y$ are the means of the measurements and $\boldsymbol{\Psi}_x$ and $\boldsymbol{\Psi}_y$ are both semi-positive definite and $\boldsymbol{z} \in \mathbb{R}^{d_z}$ has dimension $1 \leq d_z \leq \min\{d_x, d_y\}$.

Given this setup it is very easy to show that

$$\boldsymbol{\chi} = \begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{pmatrix} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}), \tag{2.12}$$

where

$$\boldsymbol{\mu} = \begin{pmatrix} \boldsymbol{\mu}_x \\ \boldsymbol{\mu}_y \end{pmatrix}, \qquad \boldsymbol{\Sigma} = \begin{pmatrix} \boldsymbol{W}_x \boldsymbol{W}_x^T + \boldsymbol{\Psi}_x & \boldsymbol{W}_x \boldsymbol{W}_y^T \\ \boldsymbol{W}_x \boldsymbol{W}_y^T & \boldsymbol{W}_y \boldsymbol{W}_y^T + \boldsymbol{\Psi}_y \end{pmatrix} \tag{2.13}$$

With this formulation, it will be shown that the maximum likelihood (ML) estimates of $\boldsymbol{W}_x$, $\boldsymbol{W}_y$, $\boldsymbol{\psi}_x$, $\boldsymbol{\psi}_y$, $\boldsymbol{\mu}_x$, and $\boldsymbol{\mu}_y$ are very closely related to the canonical directions and correlations.

The ML parameter estimates are symmetric with respect to the variables $\boldsymbol{x}$ and $\boldsymbol{y}$. The ML estimates for the parameters are:

$$\begin{aligned}\hat{\boldsymbol{W}}_x =& \bar{\boldsymbol{\Sigma}}_{xx}\boldsymbol{U}_x\boldsymbol{M}_x \\ \hat{\boldsymbol{\Psi}}_x =& \bar{\boldsymbol{\Sigma}}_{xx} - \hat{\boldsymbol{W}}_x\hat{\boldsymbol{W}}_x^T \\ \hat{\boldsymbol{\mu}}_x =& \bar{\boldsymbol{\mu}}_x \\ \\ \hat{\boldsymbol{W}}_y =& \bar{\boldsymbol{\Sigma}}_{yy}\boldsymbol{U}_y\boldsymbol{M}_y \\ \hat{\boldsymbol{\Psi}}_y =& \bar{\boldsymbol{\Sigma}}_{yy} - \hat{\boldsymbol{W}}_y\hat{\boldsymbol{W}}_y^T \\ \hat{\boldsymbol{\mu}}_y =& \bar{\boldsymbol{\mu}}_y\end{aligned} \tag{2.14}$$

where $\bar{\boldsymbol{\Sigma}}_{xx}$ and $\bar{\boldsymbol{\mu}}_x$ are the sample covariance matrix and mean for $\boldsymbol{x}$ and similarly for $\boldsymbol{y}$. **$\boldsymbol{U}_x$ and $\boldsymbol{U}_y$ are matrices with columns consisting of the canonical directions for $\boldsymbol{x}$ and $\boldsymbol{y}$.** $\boldsymbol{M}_x$ and $\boldsymbol{M}_y$ are arbitrary matrices such that $\boldsymbol{M}_x\boldsymbol{M}_y^T = \operatorname{diag}\{\boldsymbol{\sigma}\}$ where $\boldsymbol{\sigma}$ is the vector of the first (largest) d_z canonical correlations.

2.3.1 Proof:

The proof is very complicated and tedious although the basic idea is rather simple. The idea is to compute the log-likelihood function, then come up with a set of conditions for it to have at stationary points. Using these conditions and some challenging linear algebra, the ML estimates above can be proven.

It is instructive to see what the log-likelihood function is for this scenario. To start, note that the PDF of $\boldsymbol{\chi}$ is a multivariate Gaussian and is given by

$$\boldsymbol{\chi} \sim \mathcal{N}(\boldsymbol{\chi}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = (2\pi)^{-d/2}\det\{\boldsymbol{\Sigma}\}^{-1/2}\exp\{-\frac{1}{2}(\boldsymbol{\chi}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1/2}(\boldsymbol{\chi}-\boldsymbol{\mu})\} \tag{2.15}$$

where $d = d_x + d_y$.

The likelihood of the data set is given by

$$L = \prod_{n=1}^{N}\mathcal{N}(\boldsymbol{\chi}_n; \boldsymbol{\mu}, \boldsymbol{\Sigma}) \tag{2.16}$$

and the log-likelihood is thus given by

$$\begin{aligned}\log L =& -\frac{1}{2}\sum_{n=1}^{N} d\log 2\pi + \log\det\{\boldsymbol{\Sigma}\} + (\boldsymbol{\chi}_n-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{\chi}_n-\boldsymbol{\mu}) \\ =& -\frac{N}{2}\left[d\log 2\pi + \log\det\{\boldsymbol{\Sigma}\} + \frac{1}{N}\sum_{n=1}^{N}(\boldsymbol{\chi}_n-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{\chi}_n-\boldsymbol{\mu})\right].\end{aligned} \tag{2.17}$$

Noting the following (easily verified) matrix relation

$$\boldsymbol{x}^T\boldsymbol{A}\boldsymbol{x} = \operatorname{tr}\left\{\boldsymbol{A}(\boldsymbol{x}\boldsymbol{x}^T)\right\} \tag{2.18}$$

we have

$$\begin{aligned}\log L &= -\frac{N}{2}\left[d\log 2\pi + \log\det\{\boldsymbol{\Sigma}\} + \frac{1}{N}\sum_{n=1}^{N}\mathrm{tr}\left\{\boldsymbol{\Sigma}^{-1}(\boldsymbol{\chi}_n-\boldsymbol{\mu})(\boldsymbol{\chi}_n-\boldsymbol{\mu})^T\right\}\right] \\ &= -\frac{N}{2}\left[d\log 2\pi + \log\det\{\boldsymbol{\Sigma}\} + \mathrm{tr}\left\{\boldsymbol{\Sigma}^{-1}\left(\frac{1}{N}\sum_{n=1}^{N}(\boldsymbol{\chi}_n-\boldsymbol{\mu})(\boldsymbol{\chi}_n-\boldsymbol{\mu})^T\right)\right\}\right] \\ &= -\frac{N}{2}\left[d\log 2\pi + \log\det\{\boldsymbol{\Sigma}\} + \mathrm{tr}\left\{\boldsymbol{\Sigma}^{-1}\bar{\boldsymbol{\Sigma}}\right\}\right]\end{aligned} \tag{2.19}$$

where $\boldsymbol{\Sigma}$ is the sample covariance.

By differentiating this log-likelihood with respect to the parameters and setting the derivatives to be zero, a set of conditions can be determined for the parameters for which $\log L$ will have stationary points. With these and some difficult linear algebra, the ML estimates can be found.

2.4 Applications of CCA

2.4.1 A Simple Example

This section provides some simulated examples to illustrate the capabilities of CCA. First we consider simulated data generated with the model in (2.11). In the following figures, $\boldsymbol{W}_x$ and $\boldsymbol{W}_y$ are each generated randomly with uniform Gaussian iid elements. The dimensions of the variables are $d_x = 10$, $d_y = 5$, and $d_z = 2$. The noise is uniform iid noise with $\boldsymbol{\Psi}_x = \boldsymbol{\Psi}_y = 0.01\boldsymbol{I}$.

Figure 2.1 shows the canonical correlations and variates for the first 3 canonical directions. Notice that the canonical correlation falls off very quickly after the dimension of $\boldsymbol{z}$. **Hence, CCA can be used to estimate the dimension of the latent variable**.

As might be expected, CCA begins to perform more poorly as the number of samples becomes very small. Figures 2.2 – 2.4 show that as N decreases to near the dimension of $\boldsymbol{x}$ or $\boldsymbol{y}$, the canonical correlations all approach 1.

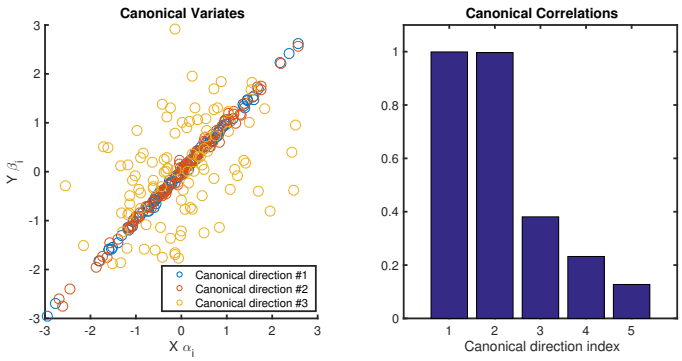

Figure 2.1: CCA analysis for the model in (2.11) with $d_x = 10$, $d_y = 5$, $d_z = 2$, and $N = 100$.

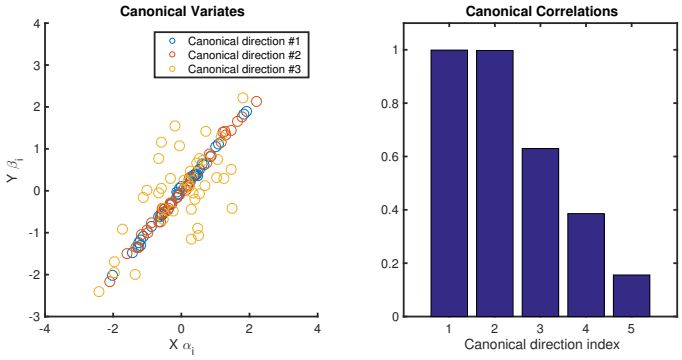

Figure 2.2: CCA analysis for the model in (2.11) with $d_x = 10$, $d_y = 5$, $d_z = 2$, and $N = 40$.

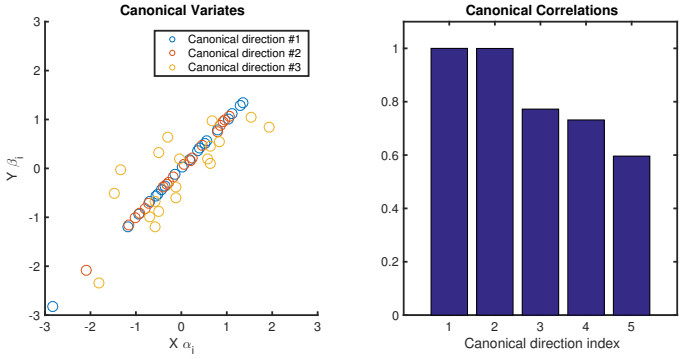

Figure 2.3: CCA analysis for the model in (2.11) with $d_x = 10$, $d_y = 5$, $d_z = 2$, and $N = 20$.

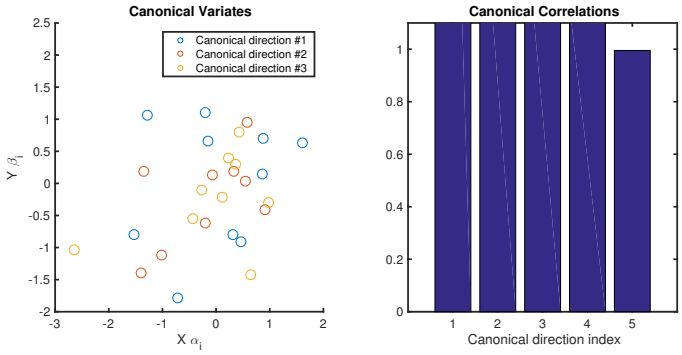

Figure 2.4: CCA analysis for the model in (2.11) with $d_x = 10$, $d_y = 5$, $d_z = 2$, and $N = 10$.

2.4.2 Application to Multimodal Signal Fusion

One significant application of CCA is to fuse signals of multiple modalities, where *modality* here means sources of data, e.g., text, image, or audio. The reference [NKK+11] is one of the earliest papers that shows the practical importance of CCA in this problem, the setting of which can be summarized as follows

- Goal: learn representation features for coupled speech and audio signals which capture the relationships across the modalities.

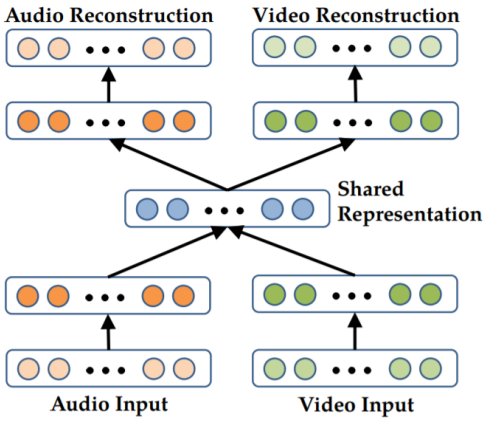

Figure 2.5: Bimodal Deep Autoencoder proposed in [NKK+11].

- Settings:
  - Multimodal fusion
  - Cross modality learning
  - Shared representation learning
- Difficulty:
  - The relationship between audio and video data are highly non-linear;
  - The modality of data used in supervised training and testing could be different from that present in feature learning

The authors propose a bimodal deep autoencoder model, with sparse restricted Boltzmann machines (RBMs) as initialization, for feature learning, as shown in Fig. 2.5. If only one modality presents as the input, data of the other modality are supplemented with zero values. The model is then applied to all the three settings and the features learned are compared with other types of features, including video-only RBM and video-only deep autoencoder, etc., by fed into a linear SVM classifier. Extensive experiments corroborate the effectiveness of the proposed feature learning model.

The CCA technique is incorporated in the *shared representation learning* setting, where the modalities of data used for supervised training and testing are different, while the representation feature is learned from the Bimodal deep autoencoder as in Fig. 2.5. Three methods for feature learning are compared

- Raw-CCA: perform CCA directly on concatenated raw video and audio data (see next section);
- RBM-CCA Feature: perform CCA on the features extracted after *first layer of RBM*;
- Bimodal Deep Autoencoder: the shared representation from the middle layer as shown in Fig. 2.5.

The experiments show that the RBM-CCA feature outperforms the other two types of features. This shows that a purely linear correlation captured by Raw-CCA cannot represent the relationship

between audio and video data efficiently. Interestingly, the features learned from Bimodal deep autoencoder fail to capture better relationship of two data sources than the simple *CCA+single-layer-RBM*s model. This implies that *it would be beneficial to perform CCA on the data after some non-linear transformation*, i.e., on $f(\boldsymbol{x})$ and $g(\boldsymbol{y})$ with f and g being single-layer-RBMs in this case. This conclusion motivates the non-linear representation learning techniques we will introduce in next lecture.

2.4.3 Raw-CCA

In this section we show how CCA can be used to learn shared representations between audio and visual signals. CCA was performed using a small data set consisting of a video of the same person saying the sounds */dah/* and */boo/* (6 videos of each) for $N = 12$. In this case we had $\boldsymbol{x}$ as the video and $\boldsymbol{y}$ as the audio samples. The duration of each video was 1 s. Each video had a frame rate of 30 frames/s and was of size $64 \times 64 \times 30$ so that $d_x = 122880$. The duration of each audio signal was thus also 1 s and with a sampling rate of 44.1 kHz, $d_y = 44100$.

As discussed in section 2.4.1, since $N \ll \max\{d_x, d_y\}$, CCA is not expected to work very well. For this reason, we first reduced the dimensionality of each dataset to 2 using SVD. Specifically if $\boldsymbol{V}_x$ was the matrix of the first 2 right singular vectors of $\boldsymbol{X}$ then we performed CCA using the matrix $\boldsymbol{X}' = \boldsymbol{X}\boldsymbol{V}_x$, and similarly for $\boldsymbol{Y}$. In this case the canonical directions will represent linear combinations of the basis vectors in $\boldsymbol{V}_x$ and $\boldsymbol{V}_y$ and so our "real" canonical directions will be $\{\boldsymbol{V}_x\boldsymbol{\alpha}_m\}_{m=1}^2$ and $\{\boldsymbol{V}_x\boldsymbol{\alpha}_m\}_{m=1}^2$.

The following figures show the canonical variates and correlations for the data set (Fig. 2.6) as well as the "real" canonical directions in the video and audio spaces (Figs. 2.7 and 2.8 respectively). Finally Fig. 2.9 shows how the variates can be used to classify (linearly separate) the two phonemes.

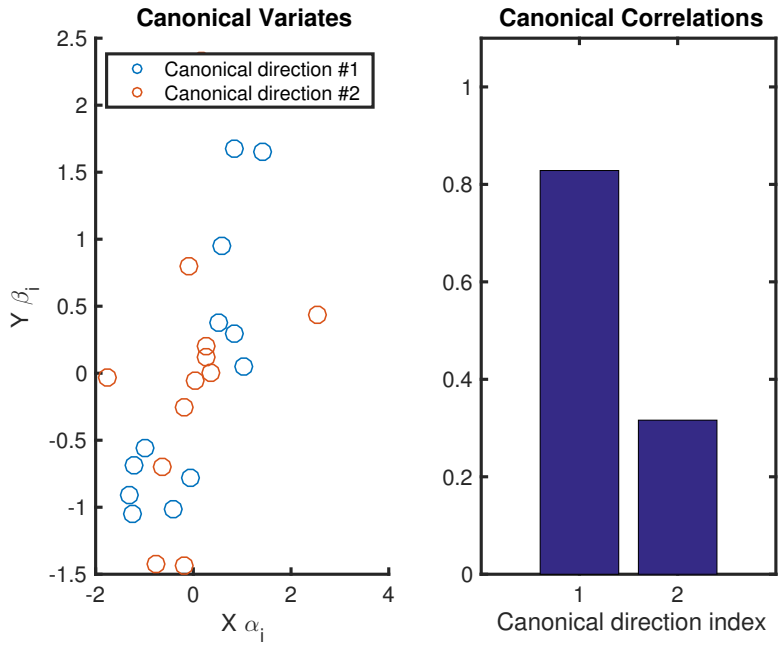

Figure 2.6: CCA analysis for the videos of phonemes with $d_x = 2$, $d_y = 2$, and $N = 12$.

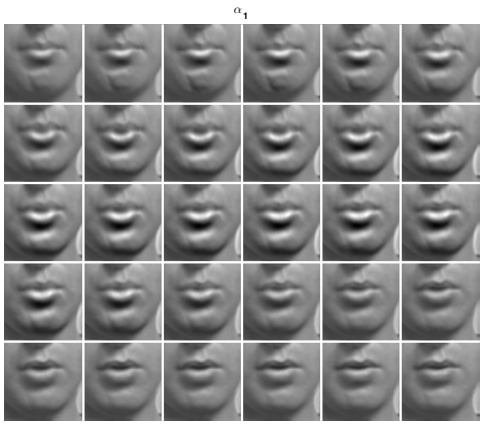

Figure 2.7: The frames of first "real" canonical direction of the video space, $\boldsymbol{\alpha}_1$. Frames are arranged left to right, top to bottom.

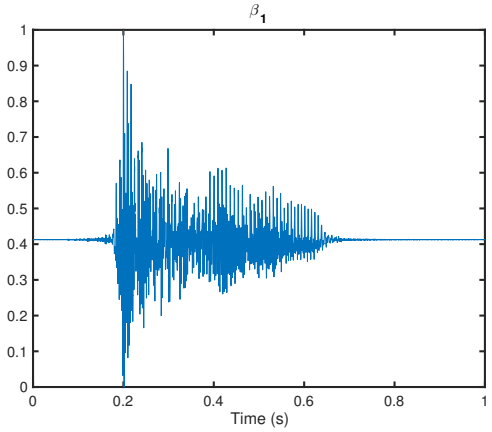

Figure 2.8: The audio waveform of first "real" canonical direction of the audio space, $\boldsymbol{\beta}_1$.

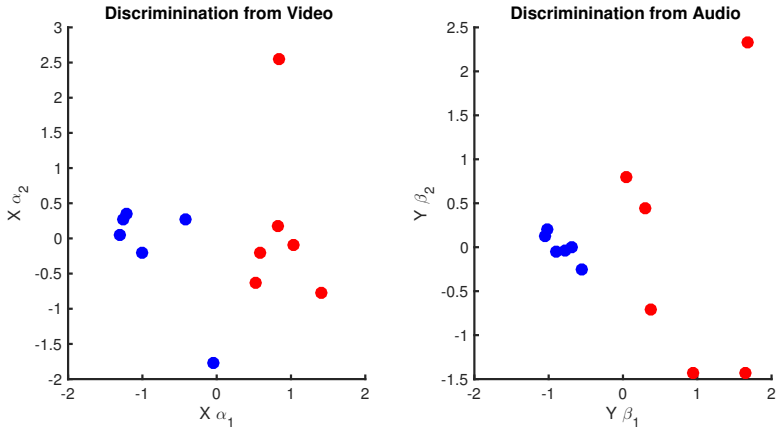

Figure 2.9: Discrimination of phonemes using either audio or video signals via the canonical variates. Blue dots are the variates for the */dah/* movies and red dots are the variates for the */boo/* movies.

2.5 Summary

In this lecture, we introduce another data analysis technique CCA, which aims to reduce the dimensions of data from two data sets while preserving the correlation between them as much as possible. In light of the formulation for PCA, we introduce an SVD-based approach for evaluating the canonical components efficiently. A probabilistic view of CCA is then presented, with two data models based on different latent variables. We also introduce a practical application of CCA in fusing audio and visual signals using deep learning, motivating the use of non-linear data transformation for CCA.

2.6 Code

The following is a MATLAB implementation of CCA. Note that there is already a MATLAB routine for CCA via the function `canoncorr`.

```
function [ Alpha, Beta, r, P, Q ] = cca( X,Y )
%CCA Computes CCA of X and Y which are row matricies of different data
%modalities with the same number of rows.
%
% Alpha, Beta = canoncial directions for X and Y respectively
% r           = canonical correllations
% P, Q        = canonical variates for X and Y respectively
%
% Created by Bryan Clifford @ UIUC, 2017
%-------------------------------------------------------------------------

    % Get sizes of data
    assert(ismatrix(X));
    assert(ismatrix(Y));
    [N,dx] = size(X);
    [~,dy] = size(Y);
    assert(N == size(Y,1));

    % center data
    ux = mean(X,1);
    uy = mean(Y,1);
    X = X - repmat(ux,N,1);
    Y = Y - repmat(uy,N,1);

    % compute sphering transforms
    [~,Sx,Vx] = svd(X,'econ');
    [~,Sy,Vy] = svd(Y,'econ');
    A = Vx*diag(1./diag(Sx))*Vx';
    B = Vy*diag(1./diag(Sy))*Vy';

    % compute canonical directions and correllations and variates
    [Alpha,r,Beta] = svd( A*X'*Y*B, 'econ' );
    Alpha = A*Alpha;
    Beta  = B*Beta;
    r = diag(r);
    P = X*Alpha;
    Q = Y*Beta;

    % normalize for unit variance
    stdP = diag(1./std(P,0,1));
    Alpha = Alpha*stdP;
```

```
    P       = P*stdP;

    stdQ = diag(1./std(Q,0,1));
    Beta = Beta*stdQ;
    Q    = Q*stdQ;

    return;

end
```

Bibliography

[BJ06] F. Bach and M. Jordan. A probabilistic interpretation of canonical correlation analysis. *University of California, Berkeley*, TR:688, 2006.

[Bro79] M. Brown. The maximum-likelihood solution in inter-battery factor analysis. *Brit. J. Math. Stat. Psyc.*, 32:75 – 86, 1979.

[NKK+11] Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y Ng. Multimodal deep learning. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 689–696, 2011.