# Chapter – 1

# INTRODUCTION

## 1.1. Multi-vew Learning

Multi-view learning which learns patterns or features from instances with multiple representations. It has been shown that learning from multiple representations of data often achieves better performance than traditional single view learning methods. Multi-view learning is also known as data fusion or data integration from multiple feature sets.

## 1.2. CCA (Canonical Correlation Analysis)

Canonical Correlation Analysis (CCA) is an unsupervised linear dimensionality reduction method. It is a learning method to find linear relationship between two groups of multi-dimensional variables. The goal of CCA is to seek two bases which would maximize the correlation of data by projecting two-view data obtained from various information sources, e.g. sound and image.

An important property of canonical correlations is that they are invariant with respect to affine transformations of the variables.

Major drawback of CCA is that it cannot preserve the local structure in canonical sub-spaces and either cannot reveal non-linear correlation relationship.

## 1.3. CSCCA

### ( Canonical Sparse Cross-view Correlation Analysis )

In order to cope with the non-linear problems and improve the performance of CCA in subsequent classification task, a feature extraction method called Canonical Sparse Cross-view Correlation Analysis (CSCCA) was introduced. In CSCCA, the local structure information is incorporated and the cross correlations between two views from within-class samples are used by sparse representation.

The proposed method not only preserves the local structure information in two views separately, but also the structure information in the cross view.

# Chapter 2

# Mathematical Formulation

## 2.1 Formulation

In CSCCA, the local structure information is incorporated and the cross correlations between two views from within-class samples are used by sparse representation. The optimization problem of CCA can be written in the equivalent form as:

$$\begin{aligned} \max \quad & \boldsymbol{w}_x^T \cdot \sum_{i=1}^{n}\sum_{j=1}^{n}(\boldsymbol{x}_i-\boldsymbol{x}_j)(\boldsymbol{y}_i-\boldsymbol{y}_j)^T \cdot \boldsymbol{w}_y \\ \text{s.t.} \quad & \boldsymbol{w}_x^T \cdot \sum_{i=1}^{n}\sum_{j=1}^{n}(\boldsymbol{x}_i-\boldsymbol{x}_j)(\boldsymbol{x}_i-\boldsymbol{x}_j)^T \cdot \boldsymbol{w}_y = \boldsymbol{1} \\ & \boldsymbol{w}_y^T \cdot \sum_{i=1}^{n}\sum_{j=1}^{n}(\boldsymbol{y}_i-\boldsymbol{y}_j)(\boldsymbol{y}_i-\boldsymbol{y}_j)^T \cdot \boldsymbol{w}_y = \boldsymbol{1} \end{aligned} \tag{4}$$

We incorporate the local structure information and the within- class cross correlations into Eq. (4). The objective function of CSCCA can be formulated as follows:

$$\begin{aligned} \max \quad & w_x^T \cdot \left(\sum_{i=1}^{n}\sum_{j=1}^{n} S_{ij}^x (x_i-x_j) S_{ij}^y (y_i-y_j)^T + \sum_{i=1}^{n}\sum_{j=1}^{n}(S_{ij}^x+S_{ij}^y) x_i y_j^T\right) \cdot w_y \\ \text{s.t.} \quad & w_x^T \cdot \left(\sum_{i=1}^{n}\sum_{j=1}^{n} S_{ij}^x (x_i-x_j) S_{ij}^y (y_i-y_j)^T \cdot w_x = 1\right) \\ & w_y^T \cdot \left(\sum_{i=1}^{n}\sum_{j=1}^{n} S_{ij}^y (y_i-y_j) S_{ij}^y (y_i-y_j)^T \cdot w_y = 1\right) \end{aligned} \tag{5}$$

where $S_{ij}^x$ and $S_{ij}^y$ are elements of similarity matrices Sx and Sy.

First the similarity matrices Sx and Sy are obtained by sparse reconstruction instead of the standard Euclidean distance.

Second we emphasize the correlations between within-class by adding the correlation term

$$\sum_{i=1}^{n}\sum_{j=1}^{n}(\mathrm{S}_{\mathrm{ij}}^{\mathrm{x}}+\mathrm{S}_{\mathrm{ij}}^{\mathrm{y}}) x_i\, y_i^T$$

The similarity matrix Sx is defined as follows. Given c classes samples:

$$\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n] = [\boldsymbol{x}_1^{(1)}, \ldots, \boldsymbol{x}_{n_1}^{(1)}, \ldots, \boldsymbol{x}_1^{(c)}, \ldots, \boldsymbol{x}_{n_c}^{(c)}]$$

where $x_i^{(j)} \in R^d$ denotes the i-th sample in the j-th class. For sample $x_i^{(j)}$ from the j-th class, we find a sparse reconstructive weight vector $S_i^{(w)}$ by the following minimization problem:

$$\min_{\boldsymbol{S}_i^w \geq 0} \frac{1}{2} \|\boldsymbol{x}_i - \boldsymbol{X}\boldsymbol{S}_i^w\| + \eta \|\boldsymbol{S}_i^w\|_1 \tag{6}$$

where η is a balancing factor which controls the sparsity of $S_i^w$ and $S_i^w$ is an vector where the element in the position of $x_i^j$ is zero to ignore degenerated solution.

$$\boldsymbol{S}_i^w = [\overbrace{0, \ldots, 0}^{\sum_{k=1}^{j-1} n_k}, S_{i1}, \ldots, S_{i,i-1}, 0, S_{i,i+1}, \ldots, S_{i,n_j}, \overbrace{0, \ldots, 0}^{n - \sum_{k=1}^{j} n_k}]^T$$

It is worth noting that we only use the samples with the same label to reconstruct $x_i$ for preserving the class information. When we get ~ $S_i^w$ which is the optimal solution of Eq. (6), the sparse reconstructive weight matrix $S_x$ can be denoted as follows:

$$\boldsymbol{S}^w = [\tilde{\boldsymbol{S}}_1^w, \ldots, \tilde{\boldsymbol{S}}_n^w]$$
$$\boldsymbol{S}_x = \boldsymbol{S}^w + (\boldsymbol{S}^w)^T$$

The optimization problem (5) can be rewritten after some algebraic manipulations as :

$$\begin{aligned} \max_{\boldsymbol{w}_x, \boldsymbol{w}_y} \quad & \boldsymbol{w}_x^T \boldsymbol{X}\boldsymbol{R}\boldsymbol{Y}^T \boldsymbol{w}_y \\ \text{s.t.} \quad & \boldsymbol{w}_x^T \boldsymbol{X}\boldsymbol{S}_{xx}\boldsymbol{X}^T \boldsymbol{w}_x = 1 \\ & \boldsymbol{w}_y^T \boldsymbol{Y}\boldsymbol{S}_{yy}\boldsymbol{Y}^T \boldsymbol{w}_y = 1 \end{aligned} \tag{7}$$

where **X** = $[x_1, \ldots, x_n]$, **Y** = $[y_1, \ldots, y_n]$, **R** = $2S_{xy} + S_x + S_y$ , $\boldsymbol{S_{xy}} = D_{xx} - S_{x}\circ S_x$ , $\boldsymbol{S_{yy}} = D_{yy} - S_{y}\circ S_y$ , $\boldsymbol{S_{xy}} = D_{xy} - S_{x}\circ S_y$ the symbol ○ denotes an operator such that $(A°B)_{ij} = A_{ij}B_{ij}$ for matrices A, B with the same size and $A_{ij}$ denotes the ij-th entry of A, $D_{xx}(D_{yy}, D_{xy})$ is a diagonal matrix of size n by n, and its i-th diagonal entry equal to the sum of the entries in the i-th row of the matrix $S_x°S_x(S_y°S_y, S_x°S_y)$.

According to Eq. (3), the optimization problem of CSCCA can be solved by following generalized eigenvalue decomposition:

$$\begin{bmatrix} & \boldsymbol{XRY}^T \\ \boldsymbol{YRX}^T & \end{bmatrix} \begin{bmatrix} \boldsymbol{w}_x \\ \boldsymbol{w}_y \end{bmatrix} = \lambda \begin{bmatrix} \boldsymbol{XS}_{xx}\boldsymbol{X}^T & \\ & \boldsymbol{YS}_{yy}\boldsymbol{Y}^T \end{bmatrix} \begin{bmatrix} \boldsymbol{w}_x \\ \boldsymbol{w}_y \end{bmatrix} \tag{8}$$

After obtaining d eigen vectors for each view corresponding to d generalized eigen values λi, i = 1,..,. d, we denote them as $\boldsymbol{W_x} = [\boldsymbol{w_{x,}^1}, \ldots, \boldsymbol{w_x^d}]$ and $\boldsymbol{W_y} = [\boldsymbol{w_{y,}^1}, \ldots, \boldsymbol{w_y^d}]$. For samples **x, y**, the features can be extracted as follows :

$$\boldsymbol{W}_x^T\boldsymbol{x} + \boldsymbol{W}_y^T\boldsymbol{y} \qquad \begin{bmatrix} \boldsymbol{W}_x^T\boldsymbol{x} \\ \boldsymbol{W}_y^T\boldsymbol{y} \end{bmatrix} \tag{9}$$

# Chapter 3

# Algorithm

**Algorithm 1.** CSCCA.

**Input:** Training sets $\boldsymbol{X} \in \mathbb{R}^{p \times n}, \boldsymbol{Y} \in \mathbb{R}^{q \times n}$
**Output:** Projection matrices $\boldsymbol{W}_x, \boldsymbol{W}_y$

**1** Construct $\boldsymbol{S}_x, \boldsymbol{S}_y$.
**2** Define $\boldsymbol{R} = \boldsymbol{S}_{xy} + \boldsymbol{S}_x + \boldsymbol{S}_y$;
**3** Compute matrices $\tilde{\boldsymbol{C}}_{xy} = \boldsymbol{X}\boldsymbol{R}\boldsymbol{Y}^T$, $\tilde{\boldsymbol{C}}_{xx} = \boldsymbol{X}\boldsymbol{S}_{xx}\boldsymbol{X}^T$, $\tilde{\boldsymbol{C}}_{yy} = \boldsymbol{Y}\boldsymbol{S}_{yy}\boldsymbol{Y}^T$.
**4** Compute matrix $\boldsymbol{H} = \tilde{\boldsymbol{C}}_{xx}^{-\frac{1}{2}} \tilde{\boldsymbol{C}}_{xy} \tilde{\boldsymbol{C}}_{yy}^{-\frac{1}{2}}$;
**5** Perform SVD decomposition on $\boldsymbol{H} : \boldsymbol{H} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T$;
**6** Choose $\boldsymbol{U} = [U_1 \dots U_d]$, $\boldsymbol{V} = [V_1 \dots V_d]$, $d < n$;
**7** Obtain $\boldsymbol{W}_x = \tilde{\boldsymbol{C}}_{xx}^{-\frac{1}{2}} \boldsymbol{U}$, $\boldsymbol{W}_y = \tilde{\boldsymbol{C}}_{yy}^{-\frac{1}{2}} \boldsymbol{V}$;

# Chapter 4 Documentation of API

## 4.1 Package organization

## 4.2 Methods

# Chapter 5

# Example

## 5.1 Example 1

# Chapter 6

## Learning Outcome

- Learned to analyse and interpret scientific and mathematical concepts and implement them from scratch.
- Learned the concept behind using CCA & CSCCA and its application.
- Understanding the concepts of latent space and how Canonical Correlation Analysis(CCA) can be used to get maximum insights from different views of the same data by considering them on a shared latent space and eliminating the drawbacks of the process with the use of CSCCA (Canonical Sparse Cross-view Correlation Analysis) as a improvement over the latter.

# Appendix A

# References


1. **Canonical sparse cross-view correlation analysis**
   https://sci-hub.se/10.1016/j.neucom.2016.01.053

2. https://www.sciencedirect.com/science/article/abs/pii/S0031320309001964

3. https://www.researchgate.net/publication/228631619_SLEP_Sparse_Learning_with_Efficient_Projections/link/0046351a36ba9d0adf000000/download

4. https://archive.ics.uci.edu/ml/machine-learning-databases/mfeat/