Deep Canonical Correlation Analysis

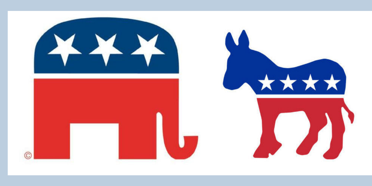
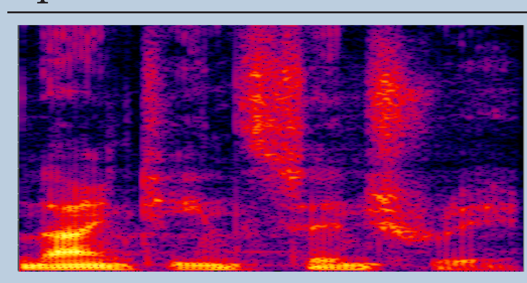
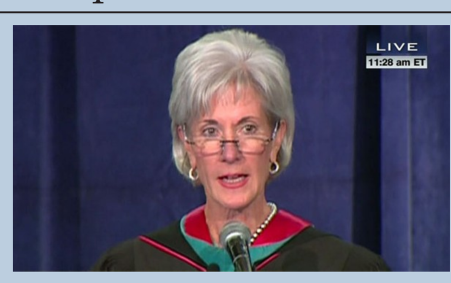
Galen Andrew[1], Raman Arora[2], Jeff Bilmes[1], Karen Livescu[2]
[1]University of Washington [2]Toyota Technological Institute at Chicago

0. Abstract

- We introduce DCCA, a method to learn complex nonlinear transformations of two views of data such that the resulting representations are highly linearly correlated.
- Unlike KCCA, DCCA is a parametric method and does not require an inner product.
- In experiments on real-world datasets, DCCA finds representations that are much more highly correlated than those of KCCA.
- We also introduce a novel non-saturating sigmoid function based on the cube root.

1. Correlated Representations

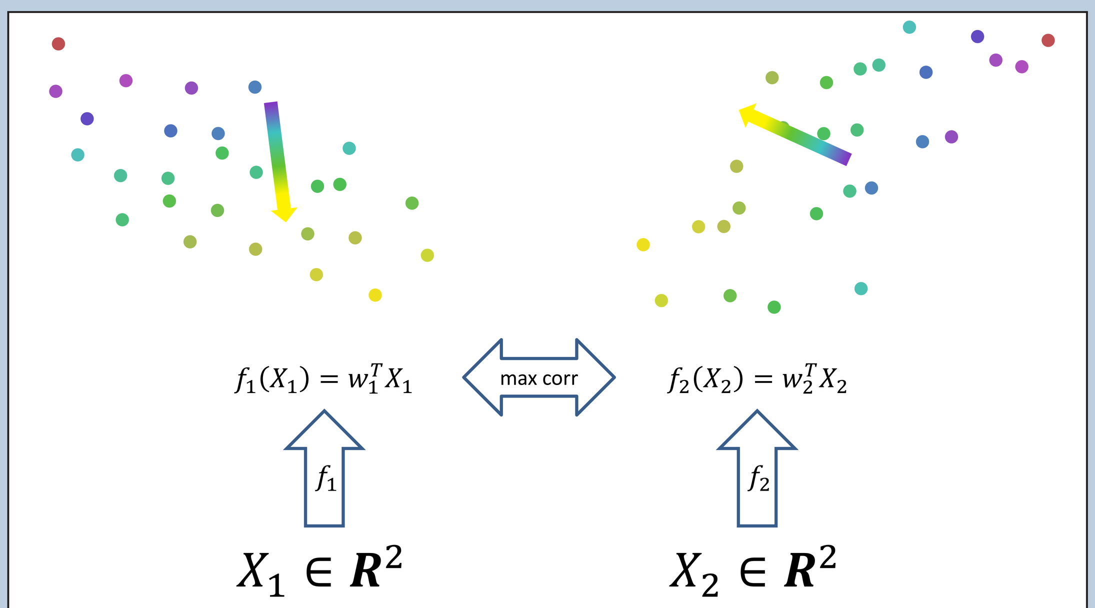
Consider a dataset in which each case has two multi-dimensional "views" $x_1^{(i)} \in \mathbb{R}^{n_1}$ and $x_2^{(i)} \in \mathbb{R}^{n_2}$.

$x_1^{(i)}$	$x_2^{(i)}$
 demographic properties of survey participant i	 responses of survey participant i to political questions
 acoustic features of a signal at time i	 pixel intensities of video signal at time i

- CCA, KCCA, and DCCA all learn functions $f_1(x_1) : \mathbb{R}^{n_1} \mapsto \mathbb{R}^k$ and $f_2(x_2) : \mathbb{R}^{n_2} \mapsto \mathbb{R}^k$ that maximize $\text{corr}(f_1(x_1), f_2(x_2))$.
- Finding correlated representations
  - May provide insight into the data
  - Can be used to induce features that capture some of the information of the other view, if it is unavailable at test time
  - Can be used to detect asychrony

2. CCA and KCCA

CCA detects linear relationships: $f_1(x_1) = w_1' x_1$.



Two views of each instance have the same color

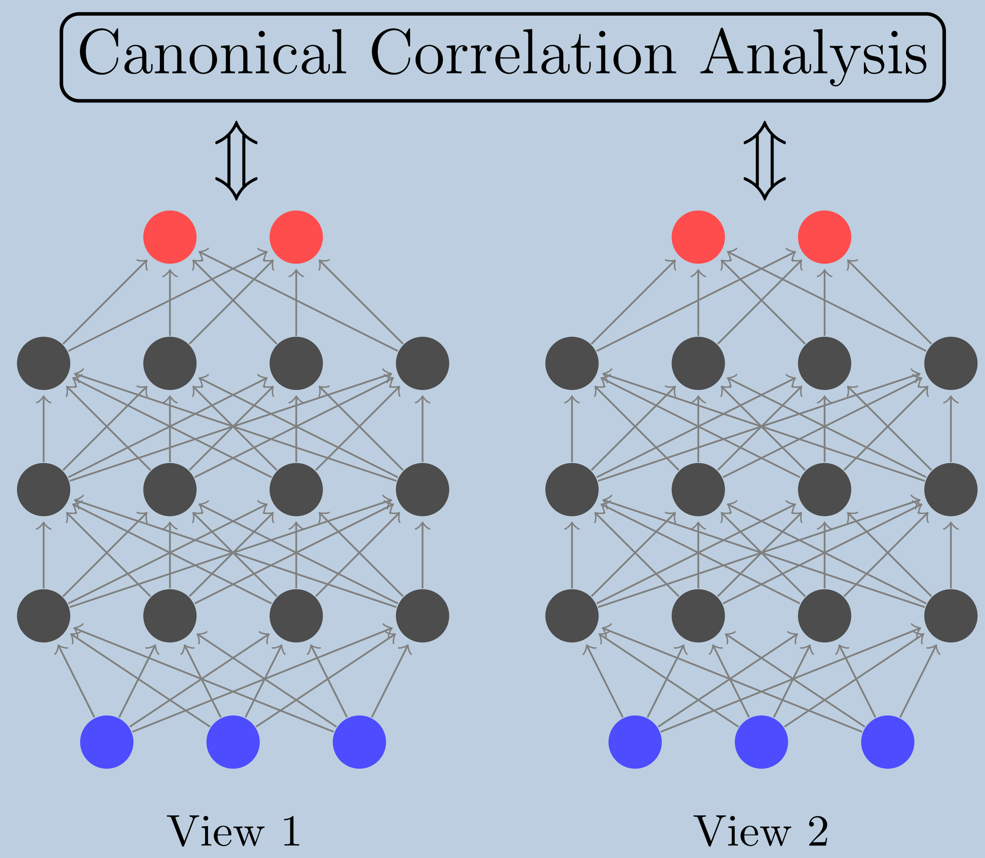
- Estimate within view covariance matrices Σ_{11} and Σ_{22}, and cross-covariance Σ_{12}.
- Let $T \triangleq \Sigma_{11}^{-1/2} \Sigma_{12} \Sigma_{22}^{-1/2}$, with SVD $T = UDV'$.
- The total correlation is $\sum_{i=1}^{k} D_{ii}$.
- The matrices of the first k pairs of projection vectors are $(W_1^*, W_2^*) = (\Sigma_{11}^{-1/2} U_k, \Sigma_{22}^{-1/2} V_k)$, where U_k is the first k columns of U.

Kernel CCA (KCCA) can use $f_1 \in \mathcal{H}$ for RKHS $\mathcal{H}$.

- May use different kernels for each view
- Can be used to produce features that improve performance of a classifier when second view is unavailable at test time (Arora & Livescu, 2012)
- Disadvantages
  - Slower to train
  - Training set must be stored and referenced when employing the model
  - Model is more difficult to interpret

3. Deep CCA

- DCCA learns mappings f_1 and f_2 represented by deep nonlinear networks under which the data is highly correlated.
- Using a deep MLP for the transformation may be well suited to natural, real-world data such as vision, audio, or other high-dimensional sensor measurements, compared to standard kernels.
- As a parametric model, the training set can be discarded once parameters have been learned.
- Unlike KCCA, does not require computing inner products.



4. Training

- To train a DCCA model
  1. Pretrain the layers of each side individually
     - We use denoising autoencoder pretraining (Vincent et al., 2008)
  2. Jointly fine-tune all parameters to maximize total correlation of the output layers H_1, H_2
- Correlation is a population objective, so it's not clear how to use typical stochastic training methods operating one instance at a time.
  - Instead, we use L-BFGS second-order method (full-batch)
- To fine-tune all parameters via backpropagation, we need to compute the gradient $\frac{\partial \,\text{corr}(H_1, H_2)}{\partial H_1}$.
- Let $\Sigma_{11}, \Sigma_{22}, \Sigma_{12}$, and $T = UDV'$ as in box 2. Then,

$$\frac{\partial \,\text{corr}(H_1, H_2)}{\partial H_1} = \frac{1}{m-1} \left(2\nabla_{11} \bar{H}_1 + \nabla_{12} \bar{H}_2 \right).$$

where

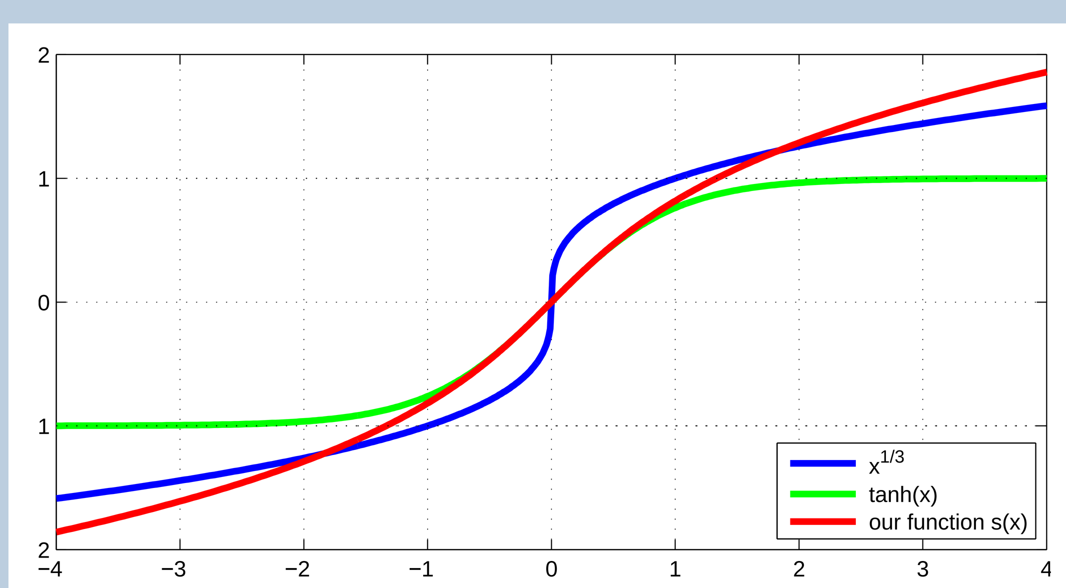
$$\nabla_{12} = \hat{\Sigma}_{11}^{-1/2} U V' \hat{\Sigma}_{22}^{-1/2}$$

and

$$\nabla_{11} = -\frac{1}{2} \hat{\Sigma}_{11}^{-1/2} U D U' \hat{\Sigma}_{11}^{-1/2}.$$

5. Nonsaturating nonlinearity

- Standard, saturating sigmoid nonlinearities (logistic, tanh) sometimes cause problems for optimization (plateaus, ill-conditioning), particularly for second-order methods.
- We obtained better results with a novel nonsaturating sigmoid.
- If $g : \mathbb{R} \mapsto \mathbb{R}$ is the function $g(y) = y^3/3 + y$, then our function is $s(x) = g^{-1}(x)$.
- Closely related to cube root, but differentiable at $x = 0$ with unit slope.
- Derivative: $s'(x) = (s^2(x) + 1)^{-1}$



This type of nonlinearity may be useful more generally in nonlinear networks (future work).

References/Acknowledgements

[1] R. Arora, and K. Livescu, "Kernel CCA for multi-view learning of acoustic features using articulatory measurements," in *Symp. on Machine Learning in Speech and Language Processing*, 2012.

[2] P. Vincent, H. Larochelle,J. Bengio, and P.A. Manzagol "Extracting and composing robust features with denoising autoencoders," in *ICML*, 2008.

This research was supported by NSF grant IIS-0905633 and by the Intel/UW ISTC.

6. Evaluation

- Evaluate model by estimating the total correlation of unseen test data after applying learned functions $f_1(x_1), f_2(x_2)$.
- CCA and KCCA order components by training correlation, but the output of a DCCA model has no inherent ordering.
- Fine to compare correlation of top k components when $k = o$, the DCCA output size.
- To evaluate at $k < o$
  - Perform linear CCA on output layers on training data to obtain transformations W_1, W_2
  - Map test data by W_1 and W_2, then compare correlation of top k components

7. Split MNIST

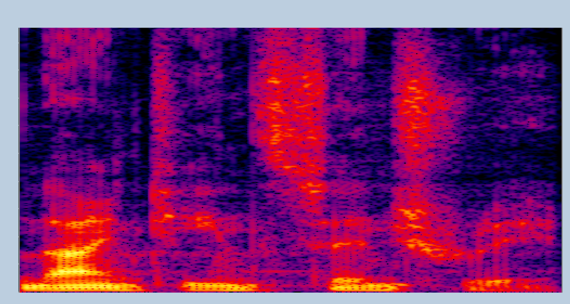
- MNIST handwritten digits, left/right halves
- 28x14 matrix of 256 grayscale values

- 60k train, 10k test
- 10% of train used for hyperparameter tuning
- $k = 50$ for all models (max score: 50)
- DCCA model has two layers, hidden layer widths chosen on development set as 2038 and 1608

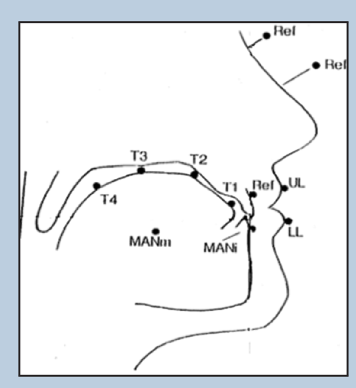
	CCA	KCCA (RBF)	DCCA (50-2)
Dev	28.1	33.5	**39.4**
Test	28.0	33.0	**39.7**

8. Speech

- Wisconsin XRMB database of simultaneous acoustic and articulatory recordings



Acoustic: 13 MFCCs + first and second derivatives, over seven frames (273 features)



Articulatory: horizontal & vertical displacements of 8 pellets on lips, tongue & jaws over 7 frames (112 features)