

# Continuous Latent Variables

Oliver Schulte - CMPT 419/726

Bishop PRML Ch. 12

# Outline

## Principal Component Analysis

# Outline

## Principal Component Analysis

# PCA: Motivation and Intuition

- Basic Ideas over 100 years old (stats). Still useful!
- Think about linear regression. If basis functions are not given, can we **learn** them from data?
- Goal: find a small set of **hidden basis functions** that explains the data as well as possible.
- Intuition: Suppose that your data is generated by a few hidden causes or **factors**. Then you could compactly describe each data point by how much each cause contributes to generate it.
- Principal Component Analysis (PCA) assumes that the contribution of each factor to each data point is *linear*.

## Informal Example: Student Performance

- Each student's performance is summarized in 4 assignments, 1 midterm, 1 project = 6 numbers.
- Suppose that on each item, a student's performance can be explained in terms of two factors.
  - Her intelligence  $I_n$
  - Her diligence  $D_n$ .
  - Combine these into a vector  $z_n$ .
- The importance of each factor varies with the assignment. So we have 6 numbers for each. Put them in a 6x2 matrix  $W$ .
- Then the performance numbers of student  $n$  can be predicted by the model

$$\mathbf{x}_n = W\mathbf{z}_n + \varepsilon,$$

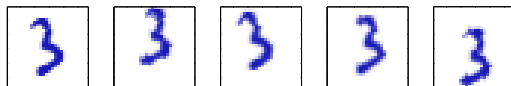
where  $\varepsilon$  is (Gaussian) noise.

## Informal Example: Blind Source Separation

- Two people are talking in a room, sometimes at the same time. <http://www.youtube.com/watch?v=Qr74sM7oqQc&feature=related>
- Two microphones are set up at different parts of the room. Each mike catches each person from a different position. Let  $x_i$  be the combined signal at microphone  $i$ .
- The contribution of person 1 to mike  $i$  depends on the position of mike  $i$ , can be summarized as a pair of numbers  $w_{i1}, w_{i2}$ .
- Similarly for person 2. Combine these into a  $2 \times 2$  matrix  $W$ .
- Let  $z_i$  be the (amplitude of) the voice signal of person  $i$ . Then the combined signal at mike 1 is given by  $x_1 = w_{11} \cdot z_1 + w_{12} \cdot z_2$ .
- Similarly for mike 2. Overall, we have that

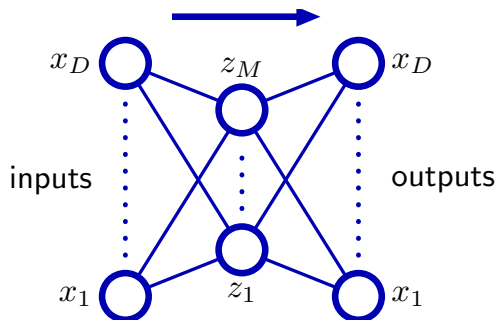
$$\mathbf{x} = W\mathbf{z}.$$

## Example: Digit Rotation



- Take a single digit (3), make 100x100 pixel image
- Create multiple copies by translating and rotating
- This dataset could be represented as vectors in  $\mathbb{R}^{100 \times 100} = \mathbb{R}^{10000}$
- But the dataset only has 3 **degrees of freedom**... why are 10,000 needed?
  - Shouldn't a **manifold** or **subspace** of **intrinsic dimension 3** suffice?
- Teapot demo  
<http://www.youtube.com/watch?v=BfTMmoDFXyE>

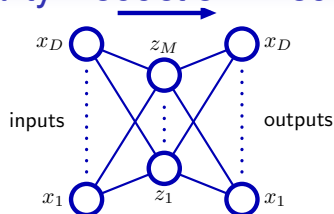
## Auto-Associative Neural Nets



- An **auto-associative** neural net has just as many input units as output units, say  $D$ .
- The error is the squared difference between input unit  $x_i$  and output unit  $o_i$ , i.e. the network is supposed to recreate the input.



# Dimensionality Reduction: Neural Net View



- Suppose we have 1 hidden layer with just one node.
  - The network then has to map each input to a single number that allows it to recreate the entire input as well as possible.
- More generally, we could have  $M \ll D$  hidden nodes.
  - The network then has to map each input to a lower-dimensional vector that allows it to recreate the entire input as well as possible.
- You can in fact use this set-up to train an ANN to perform dimensionality reduction.
- But because of the linearity assumption, we can get a fast closed-form solution.

# Component Analysis: Pros and Cons

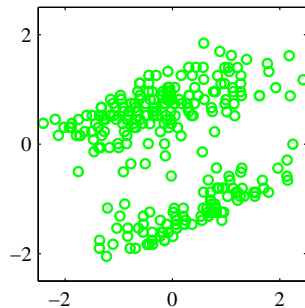
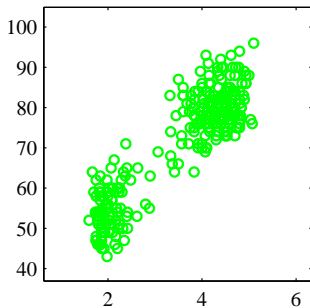
## Pros

- Reduces dimensionality of data: easier to learn.
- Removes noise, filters out important regularities.
- Can be used to standardize data (whitening).

## Cons

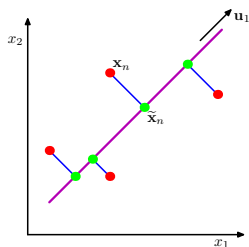
- PCA is restricted to linear hidden models. (Relax later).
- Black box: data vectors become hard to interpret.

# Pre-processing Example



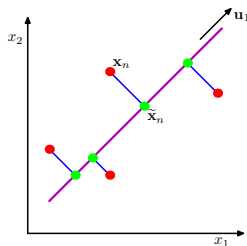
After preprocessing the original data (left), we obtain a data set with mean 0 and unit covariance.

# Dimensionality Reduction



- We will study one simple method for finding a lower dimensional manifold – principal component analysis (PCA)
- PCA finds a lower dimensional **linear** space to represent data
- How to define the **right** linear space?
  - Subspace that maximizes variance of projected data
  - Minimizes projection cost
- Turns out they are the same!

# Maximum Variance



- Consider dataset  $\{\mathbf{x}_n \in \mathbb{R}^D\}$
- Try to project into space with dimensionality  $M < D$
- For  $M = 1$ , space given by  $\mathbf{u}_1 \in \mathbb{R}^D$ ,  $\mathbf{u}_1^T \mathbf{u}_1 = 1$
- **Optimization problem: find  $\mathbf{u}_1$  that maximizes variance**

## Projected variance

- The projection of a datapoint  $\mathbf{x}_n \in \mathbb{R}^D$  by  $\mathbf{u}_1$  is  $\mathbf{u}_1^T \mathbf{x}_n$
- The mean of the projected data is

$$\frac{1}{N} \sum_{n=1}^N \mathbf{u}_1^T \mathbf{x}_n = \mathbf{u}_1^T \left( \frac{1}{N} \sum_{n=1}^N \mathbf{x}_n \right) = \mathbf{u}_1^T \bar{\mathbf{x}}$$

- The variance of the projected data is

$$\begin{aligned} \frac{1}{N} \sum_{n=1}^N \{ \mathbf{u}_1^T \mathbf{x}_n - \mathbf{u}_1^T \bar{\mathbf{x}} \}^2 &= \mathbf{u}_1^T \left( \frac{1}{N} \sum_{n=1}^N (\mathbf{x}_n - \bar{\mathbf{x}})(\mathbf{x}_n - \bar{\mathbf{x}})^T \right) \mathbf{u}_1 \\ &= \mathbf{u}_1^T \mathbf{S} \mathbf{u}_1 \end{aligned}$$

where  $\mathbf{S}$  is the **sample covariance**.

# Optimization

- How do we maximize the projected variance  $\mathbf{u}_1^T \mathbf{S} \mathbf{u}_1$  subject to the constraint that  $\mathbf{u}_1^T \mathbf{u}_1 = 1$ ?
  - Lagrange multipliers:

$$\mathbf{u}_1^T \mathbf{S} \mathbf{u}_1 + \lambda_1 (1 - \mathbf{u}_1^T \mathbf{u}_1)$$

- Taking derivatives, stationary point when

$$\mathbf{S} \mathbf{u}_1 = \lambda_1 \mathbf{u}_1$$

i.e.  $\mathbf{u}_1$  is an **eigenvector** of  $\mathbf{S}$

## Optimization – Which Eigenvector

- There are up to  $D$  eigenvectors, which is the right one?
- Maximize variance!
- Variance is:

$$\begin{aligned} & \mathbf{u}_1^T \mathbf{S} \mathbf{u}_1 \\ &= \mathbf{u}_1^T \lambda_1 \mathbf{u}_1 \text{ since } \mathbf{u}_1 \text{ is an eigenvector} \\ &= \lambda_1 \text{ since } \|\mathbf{u}_1\| = 1 \end{aligned}$$

- Choose the eigenvector  $\mathbf{u}_1$  corresponding to the largest eigenvalue  $\lambda_1$ 
  - This is the first direction ( $M = 1$ )
  - If  $M > 1$ , simple to show eigenvectors corresponding to largest  $M$  eigenvalues are the ones to choose to maximize projected variance



## Reconstruction Error

- Can also phrase problem as finding set of orthonormal basis vectors  $\{\mathbf{u}_i\}$  for projection
- Find set of  $M < D$  vectors to minimize reconstruction error

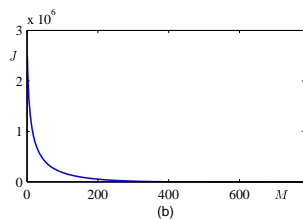
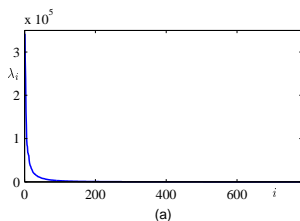
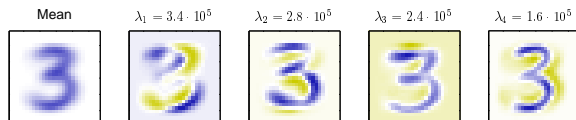
$$J = \frac{1}{N} \sum_{n=1}^N \|\mathbf{x}_n - \tilde{\mathbf{x}}_n\|^2$$

where  $\tilde{\mathbf{x}}_n$  is projected version of  $\mathbf{x}_n$

- $\tilde{\mathbf{x}}_n$  will end up being same as before – mean plus leading eigenvectors of covariance matrix  $S$

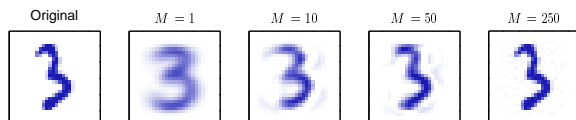
$$\tilde{\mathbf{x}}_n = \bar{\mathbf{x}} + \sum_{i=1}^M \beta_{ni} \mathbf{u}_i$$

## PCA Example – MNIST Digits



- PCA of digits “3” from MNIST
- First  $\approx 100$  dimensions capture most variance / low reconstruction error

# Reconstruction – MNIST Digits



- PCA approximation to a data vector  $\mathbf{x}_n$  is:

$$\tilde{\mathbf{x}}_n = \bar{\mathbf{x}} + \sum_{i=1}^M \beta_{ni} \mathbf{u}_i$$

- As  $M$  is increased, this reconstruction becomes more accurate
- $D = 784$ , but with  $M = 250$  quite good reconstruction
  - Dimensionality reduction

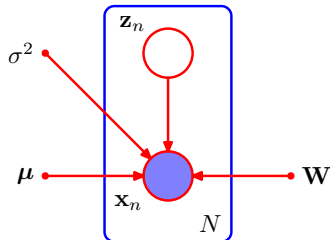
## PCA Example – Eigenfaces



Kirby and Sirovich, PAMI 1990

<http://en.wikipedia.org/wiki/Eigenface>.

# Probabilistic PCA

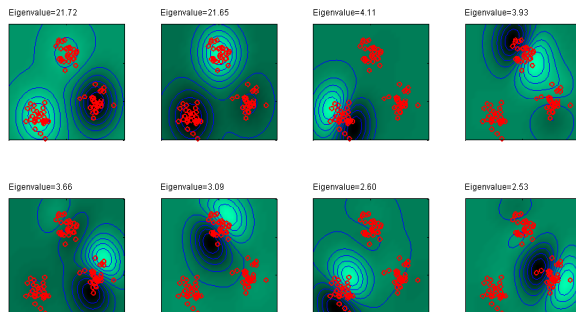


- Probabilistic model of PCA: For each data point  $x_n$ , there is a latent variable vector  $z_n$ .
- Linear Gaussian model:

$$x = Wz + \mu + \varepsilon.$$

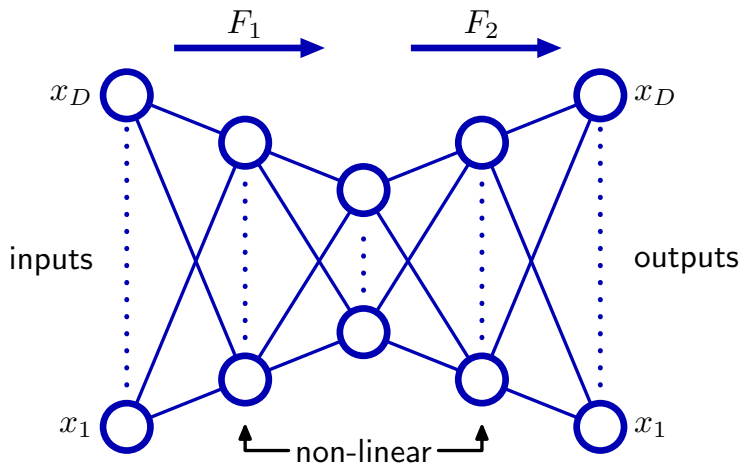
- Can train using EM.
- Handles missing data.
- Can take mixtures of PCA models.
- Closely related to **factor analysis**.  
<http://cscs.umich.edu/~crshalizi/weblog/>.
- .... (see text).

# Nonlinear PCA: Kernel Methods



- Can use the kernel trick: replace dot products with kernel evaluations.
- In the figure, the first 2 eigenvectors separate 3 clusters.
- The next 3 split the clusters in halves.
- The last 3 split the clusters in orthogonal halves.

## Nonlinear PCA: Neural Nets



With *more than one* hidden layers, neural nets perform non-linear dimensionality reduction.

# Conclusion

- Readings: Ch. 12.1
- We discussed one method for finding a lower dimensional manifold – principal component analysis (PCA)
- PCA is a basic technique
  - Finds linear manifold (hyperplane)
- In general, manifold will be non-linear
  - Simple example – translating digit 1
  - Also “dimensions” corresponding to style of digit
- Other important techniques for non-linear dimensionality reduction: **independent component analysis (ICA)**, **isomap**, **locally linear embeddings**