SUPERVISED SYSTEM IDENTIFICATION USING MANIFOLD LEARNING

Presented by

Tomer Koren

EE049035
Speech Processing in Reverberant Environments
Spring 2011
The basic setup – echo cancellation

- **Standard conference room**
  - Reverberant environment
  - Single microphone, single loudspeaker

- **Feedback from loudspeaker to microphone**
  - Feedback signal is distorted by acoustic reflections

- **Problem:** echoes should be suppressed from the microphone signal
  - Room Impulse Response (RIR) between speaker and microphone is *unknown*
  - Local *noise*
Problem formulation

- For the sequel – consider an algebraic formulation

- Acoustic coupling is characterized by \( h \in \mathbb{R}^d \)
  - Usually \( d \) is determined by T60 of the room

- Speaker signal \( x \in \mathbb{R}^n \) and microphone signal \( y \in \mathbb{R}^m \) are known
  - with \( m = n + d - 1 \)

- Assuming there is no double-talk
  \[
  y = x \ast h + n
  \]
  - Local noise \( n \in \mathbb{R}^m \)

- In matrix notation
  \[
  y = X^\top h + n
  \]
  - The columns of \( X \in \mathbb{R}^{d \times m} \) are frames of \( x \)
Naïve identification (LS)

\[
\hat{h} = \arg \min_{h} \| X^\top h - y \|^2
\]

- Solution given by \( \hat{h} = (XX^\top)^{-1}Xy \)

Works well if noise-level is very low

Otherwise – this problem is usually \textit{ill-posed}

- The resulting linear system is under-determined
- Number of parameters in \( h \) is “too large”

We have to use some \textit{prior information} on \( h \)

- Usually a model (parametric, statistical, etc.)
Optimization approach: use \textit{regularization}

\[ \hat{h} = \arg \min_h \{\|X^\top h - y\|^2 + \lambda \cdot R(h)\} \]

- \( R \) measures the “complexity” of \( h \)
- \( \lambda \) tradeoffs between fidelity and complexity

Regularization promotes “favorable” solutions

- Actually, penalizes the non-favorable solutions

But \textbf{how} to choose \( R \)??

- Which \( h \) should be favored?
- Should be feasible for optimization (e.g., convex)
Supervised system identification

- Use examples (training set) to get prior information on $h$
  - From the same room, representative speaker locations

- Use pairs $\{x_i, y_i\}_{i=1}^M$ to acquire training set $\{h_i\}_{i=1}^M$
  - For example, by naïve identification (assuming low noise level)
  - In practice: calibration in advance

- Then – use $\{h_i\}_{i=1}^M$ to form a model
  - Manifested by the regularizer $\mathcal{R}$

- How?
Principal Component Analysis (PCA)

- We can use **PCA** to form a model [Fozunbal, Kalker, Schafer 08’]
- PCA assumes \( \{h_i\}_{i=1}^{M} \) are **normally distributed** in \( \mathbb{R}^d \)

- Calculate empirical mean and covariance

\[
\bar{h} = \frac{1}{M} \sum_{i=1}^{M} h_i, \quad \Sigma = \frac{1}{M} \sum_{i=1}^{M} (h_i - \bar{h})(h_i - \bar{h})^\top
\]

- Model is given by \( \bar{h}, \Sigma \)

- Eigenvalue decomposition of \( \Sigma \) gives the “axes” of data
  - Principal components correspond to the **largest eigenvalues**
  - Usually only few principal components are kept
  - Model is “compact”
PCA regularization

- In terms of regularization
  \[ R(h) = (h - \bar{h})^\top \Sigma^{-1} (h - \bar{h}) \]
  - Favors deviations along principal axes

- Solve
  \[ \hat{h} = \arg \min_h \left\{ \| X^\top h - y \|^2 + \lambda \cdot (h - \bar{h})^\top \Sigma^{-1} (h - \bar{h}) \right\} \]

- In practice – only few principal components are used
  - Optimization is over a “small” subspace of \( \mathbb{R}^d \)
  - Number of parameters is reduced
  - Efficient, avoids ill-conditioning
PCA aims to discover the **global** structure of the data

However, the data actually lies on a **non-linear manifold** in $\mathbb{R}^d$

- **Global** structure is very complex
- But **local structure** may be approximated

PCA fails to capture local behavior

- Important for system identification

Modern machinery: **Manifold Learning**

- Aims to “unfold” non-linear manifolds
- Attempts to preserve locality
Manifold Learning methodology

- Construct a **graph** over the data points \( \{\mathbf{h}_i\}_{i=1}^M \)
  - Nodes are data points
  - “Close” points are connected by an edge

- Assign **weights** to edges using a pairwise **kernel**, e.g.

  \[
  w_{ij} = k(\mathbf{h}_i, \mathbf{h}_j) = \exp \left\{ - \frac{\|\mathbf{h}_i - \mathbf{h}_j\|^2}{\gamma} \right\}
  \]

  - Large weight \( \leftrightarrow \) high proximity
  - Captures locality (controlled by \( \gamma \))

- Use spectral properties of weight matrix \( \mathbf{W} \)
  - Instead of the covariance matrix
Model is usually defined **only** on the original points \( \{h_i\}_{i=1}^M \)

- Unlike in the case of PCA
- Can be extended to (i.e., approximated on) a new point

Recall that we seek for a regularization

- Must be defined everywhere
- Should be simple enough for optimization

Hence, we employ **Locality Preserving Projections** [He and Niyogi, 03’]

- Rather simple Manifold Learning approach
- Provides a **linear** model
- Successfully applied for face recognition, image retrieval, ...
Locality Preserving Projections (LPP)

- **LPP** aims to find a linear transformation $P : \mathbb{R}^d \rightarrow \mathbb{R}^k$ ($k \ll d$) that best preserves local structure.

- Minimizes the following cost function w.r.t. $P$

$$
\frac{1}{2} \sum_{i,j} w_{ij} \| P h_i - P h_j \|^2
$$

  - Under a proper normalization constraint
  - Ensuring that if $h_i$ and $h_j$ are “close” then so are $P h_i$ and $P h_j$

- Reduces to solving a **Generalized Eigenvalue Problem**

- Resulting eigenvectors can be used as “principal axes”, exactly as with PCA
Algorithm outline

- **Calibration (Training) phase**
  - Use signal pairs to acquire training set
  - Train a model using PCA / LPP
  - “Principal axes” by spectral decomposition

- **Identification phase**
  - Get a new signal pair
  - Solve regularized optimization problem
  - (Use obtained system to estimate feedback signal)
Preliminary experiment

- Used RIR simulator of E. Habets
- Generated training set of RIRs around two source locations
  - At distances of 0.5 and 2.0 from receiver
  - Room dimensions 5 × 4 × 6
- Tested on sources at the vicinity of the training set
  - Source signal is white
  - Contaminated with white noise (various SNRs)
- Measured MSE of estimated feedback signal
Some results

- No. of taps = 1024, reverberation time = 0.2 s
What’s next

• Compare with PCA regularization, optimize parameters

• Other Manifold Learning techniques?

• Experiment with real recorded data (recorded at Sharon Gannot’s lab)