Precision and Covariance Matrix Modelling

Khe Chai Sim & Mark Gales

30 September 2003

Cambridge University Engineering Department
Precision and Covariance Matrix Modelling

- Covariance matrix modelling
- Precision matrix modelling
- Extended MLLT
- EMLLT training (MLE and MPE)
- Initial CTS results
Gaussian Mixture Models

• Common distribution for state output distribution

\[ p(o) = \sum_{m=1}^{M} c^{(m)} \mathcal{N}(o; \mu^{(m)}, \Sigma^{(m)}) \]

• Number of model parameters for \( d \)-dimensional feature space:

<table>
<thead>
<tr>
<th>Parameters</th>
<th>( \text{Parameters} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priors ( c^{(m)} )</td>
<td>( M - 1 )</td>
</tr>
<tr>
<td>Means ( \mu^{(m)} )</td>
<td>( M \times d )</td>
</tr>
<tr>
<td>Covariance Matrix ( \Sigma^{(m)} )</td>
<td>( M \times \frac{d}{2}(d + 1) )</td>
</tr>
</tbody>
</table>

• Covariance matrix dominates the number of model parameters \( \mathcal{O}(d^2) \).

• This has motivated research into structured covariance matrix approximations.
Standard Covariance Approximations

- Standard approximations are:
  - **diagonal**: most commonly used in speech recognition
    assumes elements uncorrelated given the component
  - **block-diagonal**: usually static, delta and delta-delta blocks
    assumes each block is uncorrelated with other blocks.
  - **factor analysis**: covariance matrix has two aspects, a loading matrix $\Lambda$ and a diagonal noise covariance, $\tilde{\Sigma}$. Only applicable to single components
    \[
    \Sigma = AA^\top + \tilde{\Sigma}
    \]
    where $A$ is $d \times K$, $K < d$.
  - **factor analysed HMMs**: general form of FA
    \[
    \Sigma^{(m)} = A\Sigma^{(x)}A^\top + \Sigma^{(o)}
    \]
    where $A$ is $d \times K$, $K < d$. 
**Precision Matrix Approximations**

- The inverse covariance inverse, or precision matrix may be modelled
  - usually more efficient due to use of Gaussian distributions

- Forms are:
  - product of Gaussian pancakes: related to factor analysis (and probabilistic PCA). Only applied to a single Gaussian component ($K < d$).

  \[ \Sigma^{-1} = I + A^\top A = I + \sum_{i=1}^{K} a_i^\top a_i \]

  - semi-tied covariances (or MLLT)

  \[ \Sigma^{(m)}^{-1} = A^\top \Lambda^{(m)} A = \sum_{i=1}^{d} \lambda_{ii}^{(m)} a_i^\top a_i \]

  where $\Lambda^{(m)}$ is a diagonal matrix, $\lambda_{ii}^{(m)} = \frac{1}{\sigma_i^{(m)} \sigma_i^{(m)}}$. 
Heteroscedastic LDA

- Extension to LDA allowing within class covariances to vary
- May be viewed as a precision matrix model \( (K < d) \)

\[
\Sigma^{(m)-1} = A^\top \Lambda^{(m)} A = \sum_{i=1}^{K} \lambda_{ii}^{(m)} a_i^\top a_i + \sum_{i=K+1}^{d} \lambda_{ii} a_i^\top a_i
\]

Only first term above is a function of the component (the second term is global).

- Only the first \( K \) dimensions need to be computed (projection from \( d \) to \( K \) dimensions).
- Efficient estimation using semi-tied update formulae - \( K = d \) is a semi-tied system.
- Used in many state-of-the-art speech recognition systems.
Extended MLLT

- IBM’s EMLLT generalises semi-tied systems so that

\[ \Sigma^{(m)-1} = A^\top \Lambda^{(m)} A = \sum_{i=1}^{K} \lambda_{ii}^{(m)} a_i^\top a_i \]

Consists of two parts
- **basis**: the \( K \) basis, \( a_i \), or the \( K \times d \) matrix \( A \). This is shared between all components.
- **basis coefficients**: the \( K \) “inverse variances” \( \lambda_{ii}^{(m)} \), or as the \( K \times K \) diagonal matrix \( \Lambda^{(m)} \). These are component specific.

- Number of covariance parameters per component \( K \), note

\[ d \leq K \leq \frac{d}{2}(d + 1) \]

at one extreme an STC system and the other a full-covariance matrix system.
EMLLT Training

• Requires full covariance matrix statistics to be accumulated to update the basis (basis co-efficients only require diagonal $K$ dimensional ones).

• No simple closed form solution for ML training of the basis and basis coefficients.
  
  – second order gradient descent scheme used to estimate basis;
  
  – additive update rule from IBM used to update basis coefficients.

• Memory issues with large systems:
  
  – use “tied variances” limit the number of distinct component variances per state.

• MPE training implemented using weak-sense auxiliary functions
Experimental Set-Up

• Initial evaluation on English conversational telephone speech.

• h5etrain03 acoustic training data (290hrs 5446 speakers)

• dev01sub evaluation data (3hrs 59 speakers)

• Front-end
  – Reduced bandwidth 125–3800 Hz
  – 12 PLP cepstral parameters + C0 and 1st/2nd/3rd derivatives
  – Side-based cepstral mean and variance normalisation
  – Vocal tract length normalisation in training and test

• Acoustic Models
  – Gender independent models
  – Decision tree state clustered, context dependent triphones
  – 6192 distinct states
Maximum Likelihood Performance

<table>
<thead>
<tr>
<th>System</th>
<th>Matrix Dims</th>
<th>WER (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>HLDA</td>
<td>39×52</td>
<td>33.5</td>
</tr>
<tr>
<td>HLDA+EMLLT</td>
<td>39×52 + 78×39</td>
<td>33.1</td>
</tr>
<tr>
<td>EMLLT</td>
<td>78×52</td>
<td>32.6</td>
</tr>
</tbody>
</table>

MLE trained 16comp system WER on dev01sub using PLP_0_D_A_T

- HLDA+EMLLT shows slight performance gain over standard HLDA system.
- Using EMLLT directly in space of triples gives improved performance.
- EMLLT systems have about 50% more parameters - slightly unfair!
- 28comp HLDA system gave 32.3% WER.
- “Tied variances” EMLLT 28comp system gave 31.9%
### Speaker Adaptation

<table>
<thead>
<tr>
<th>System</th>
<th>Adaptation Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>—</td>
</tr>
<tr>
<td>HLDA</td>
<td>33.5</td>
</tr>
<tr>
<td>EMLLT</td>
<td>32.6</td>
</tr>
</tbody>
</table>

MLE trained 16comp system WER on dev01sub using PLP\_0\_D\_A\_T

- Direct using of MLLR update estimation highly inefficient
- Simple diagonal approximation works well (as good as exact estimation)
- Variance adaptation by updating the basis.
- Gains from EMLLT maintained after adaptation.
## Discriminative Training

<table>
<thead>
<tr>
<th>System</th>
<th>Number Comp.</th>
<th>WER(%) MLE</th>
<th>WER(%) MPE</th>
</tr>
</thead>
<tbody>
<tr>
<td>HLDA</td>
<td>16</td>
<td>33.5</td>
<td>30.8</td>
</tr>
<tr>
<td>EMLLT</td>
<td>16</td>
<td>32.6</td>
<td>30.1</td>
</tr>
<tr>
<td>HLDA</td>
<td>28</td>
<td>32.3</td>
<td>29.9</td>
</tr>
<tr>
<td>EMLLT</td>
<td>28µ, 16Σ</td>
<td>31.9</td>
<td>29.6</td>
</tr>
</tbody>
</table>

MPE trained system WER on dev01sub using PLP_0_D_A_T

- “Tied variances” EMLLT system used for 28comp.
- Currently only 4 iterations of MPE updates performed.
- MPE training works for EMLLT, though gains reduced for larger system.
- Further investigation required.
Summary

- Baseline MLE implementation of EMLLT in HTK.

- Initial investigation of EMLLT for LVCSR:
  - issues of basis initialisation;
  - memory (and speed) issues for larger number of components;
  - currently fixed value of $K$ (how to determine it?).

- Efficient MLLR mean and variance adaptation.

- Discriminative training formulation for MPE (and MMIE)

- Use of “tied variances” to reduce memory requirements.
Future Work

• Further investigation of model configuration (multiple sets of basis?).

• Current implementation of EMLLT can dramatically increase number of model parameters. Two forms of parameter tying possible:
  – multiple basis share the same basis coefficient - this is the IBM SPAM model;
  – the basis from multiple components share the same basis coefficient, the Hierarchical Basis Superposition (HBS) model.

• Super-space models: state-space models where the dimensionality of the state-space exceeds that of the observation space:
  – generalises some of the forms described;
  – allows improved?? temporal correlation modelling.