Clustering-based Gaussian Mixture Reduction
Dennis Schieferdecker - 09.06.09
Introduction
Gaussian Mixtures

Gaussian Mixture Density
- weighted sum of Gaussians

\[ f(x; \eta) = \sum_{i=1}^{N} \omega_i \cdot \mathcal{N}(x; \mu_i, \sigma_i^2) \]

- universal function approximator

- possible applications
  - target tracking,
  - density estimation,
  - ...

Problems in Application
- recursive multiplication of Gaussian mixtures
- number of components grows rapidly (exponential growth)
Gaussian Mixture Density

- weighted sum of Gaussians

\[ f(x; \eta) = \sum_{i=1}^{N} \omega_i \cdot \mathcal{N}(x; \mu_i, \sigma_i^2) \]

- universal function approximator

- possible applications
  - target tracking,
  - density estimation,
  - ...

Problems in Application

- recursive multiplication of Gaussian mixtures

- number of components grows rapidly (exponential growth)
Introduction
Gaussian Mixtures

Gaussian Mixture Density

- weighted sum of Gaussians
  \[ f(x; \eta) = \sum_{i=1}^{N} \omega_i \cdot \mathcal{N}(x; \mu_i, \sigma_i^2) \]
- universal function approximator
- possible applications
  - target tracking,
  - density estimation,
  - ...

Problems in Application

- recursive multiplication of Gaussian mixtures
- number of components grows rapidly (exponential growth)
Introduction
Gaussian Mixtures

Gaussian Mixture Density
- weighted sum of Gaussians

\[ f(x; \eta) = \sum_{i=1}^{N} \omega_i \cdot \mathcal{N}(x; \mu_i, \sigma_i^2) \]

- universal function approximator
- possible applications
  - target tracking,
  - density estimation,
  - ...

Problems in Application
- recursive multiplication of Gaussian mixtures
- number of components grows rapidly (exponential growth)
Problem Description
Gaussian Mixture Reduction

Goal
- given a mixture $\tilde{\eta}$ with $N$ components (true/original mixture),
- find a mixture $\eta$ with $K < N$ components (reduced mixture),
- so that a deviation measure $d(\tilde{\eta}, \eta)$ is minimized.

Deviation Measures
- Integrated Squared Distance (ISD):
  \[ d(f_1(x), f_2(x)) = \int_{\mathbb{R}} (f_1(x) - f_2(x))^2 \, dx \]
- Kullback-Leibler divergence (KLD):
  \[ d(f_1(x), f_2(x)) = \int_{\mathbb{R}} f_1(x) \log \frac{f_1(x)}{f_2(x)} \, dx \]
- normalized variants
Goal

- given a mixture $\tilde{\eta}$ with $N$ components (true/original mixture),
- find a mixture $\eta$ with $K < N$ components (reduced mixture),
- so that a deviation measure $d(\tilde{\eta}, \eta)$ is minimized.

### Deviation Measures

- **Integrated Squared Distance (ISD):**
  $$d(f_1(x), f_2(x)) = \int_\mathbb{R} (f_1(x) - f_2(x))^2 \, dx$$

- **Kullback-Leibler divergence (KLD):**
  $$d(f_1(x), f_2(x)) = \int_\mathbb{R} f_1(x) \log \frac{f_1(x)}{f_2(x)} \, dx$$

- normalized variants
Reduction Methods
Overview

**top-down approaches**
- greedy methods
- iteratively replace two Gaussians with one
- chosen according to a deviation measure (local, global, hybrid)

**bottom-up approach**
- constructive method
- starts with one Gaussian
- adds components as necessary
- progressive approximation
- PGMR
## Reduction Methods

### Overview

<table>
<thead>
<tr>
<th>top-down approaches</th>
<th>bottom-up approach</th>
</tr>
</thead>
<tbody>
<tr>
<td>greedy methods</td>
<td>constructive method</td>
</tr>
<tr>
<td>iteratively replace two Gaussians with one</td>
<td>starts with one Gaussian</td>
</tr>
<tr>
<td>chosen according to a deviation measure (local, global, hybrid)</td>
<td>adds components as neccessary</td>
</tr>
<tr>
<td></td>
<td>progressive approximation</td>
</tr>
<tr>
<td></td>
<td>PGMR</td>
</tr>
</tbody>
</table>

![Gaussian mixture reduction diagram](image)
Reduction Methods
Overview

**top-down approaches**
- greedy methods
- iteratively replace two Gaussians with one
- chosen according to a deviation measure (local, global, hybrid)

**bottom-up approach**
- constructive method
- starts with one Gaussian
- adds components as necessary
- progressive approximation
- PGMR
Reduction Methods
Overview

top-down approaches
- greedy methods
- iteratively replace two Gaussians with one
- chosen according to a deviation measure (local, global, hybrid)

bottom-up approach
- constructive method
- starts with one Gaussian
- adds components as necessary
- progressive approximation
- PGMR
Reduction Methods
Overview

top-down approaches
- greedy methods
- iteratively replace two Gaussians with one
- chosen according to a deviation measure (local, global, hybrid)

bottom-up approach
- constructive method
- starts with one Gaussian
- adds components as necessary
- progressive approximation
- PGMR
Reduction Methods
Overview

top-down approaches
- greedy methods
- iteratively replace two Gaussians with one
- chosen according to a deviation measure (local, global, hybrid)

bottom-up approach
- constructive method
- starts with one Gaussian
- adds components as necessary
- progressive approximation
- PGMR
Reduction Methods

Overview

top-down approaches
- greedy methods
- iteratively replace two Gaussians with one
- chosen according to a deviation measure (West, Williams, Runnalls)

bottom-up approach
- constructive method
- starts with one Gaussian
- adds components as necessary
- progressive approximation
- PGMR
## Reduction Methods

### Overview

<table>
<thead>
<tr>
<th><strong>top-down approaches</strong></th>
<th><strong>bottom-up approach</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>greedy methods</td>
<td>constructive method</td>
</tr>
<tr>
<td>iteratively replace two</td>
<td>starts with one Gaussian</td>
</tr>
<tr>
<td>Gaussians with one</td>
<td>adds components as neccessary</td>
</tr>
<tr>
<td>chosen according to a</td>
<td>progressive approximation</td>
</tr>
<tr>
<td>deviation measure</td>
<td>PGMR</td>
</tr>
<tr>
<td>(West, Williams, Runnalls)</td>
<td></td>
</tr>
</tbody>
</table>
Reduction Methods
Overview

**top-down approaches**
- greedy methods
- iteratively replace two Gaussians with one
- chosen according to a deviation measure (West, Williams, Runnalls)

**bottom-up approach**
- constructive method
- starts with one Gaussian
- adds components as necessary
- progressive approximation
- PGMR
Reduction Methods
Overview

top-down approaches
- greedy methods
- iteratively replace two Gaussians with one
- chosen according to a deviation measure (West, Williams, Runnalls)

bottom-up approach
- constructive method
- starts with one Gaussian
- adds components as necessary
- progressive approximation
- PGMR
Reduction Methods
Overview

top-down approaches

- greedy methods
- iteratively replace two Gaussians with one
- chosen according to a deviation measure (West, Williams, Runnalls)

bottom-up approach

- constructive method
- starts with one Gaussian
- adds components as necessary
- progressive approximation
- PGMR
## Reduction Methods

### Overview

<table>
<thead>
<tr>
<th>top-down approaches</th>
<th>bottom-up approach</th>
</tr>
</thead>
<tbody>
<tr>
<td>greedy methods</td>
<td>constructive method</td>
</tr>
<tr>
<td>iteratively replace two Gaussians with one</td>
<td>starts with one Gaussian</td>
</tr>
<tr>
<td>chosen according to a deviation measure (West, Williams, Runnalls)</td>
<td>adds components as necessary</td>
</tr>
<tr>
<td></td>
<td>progressive approximation</td>
</tr>
<tr>
<td></td>
<td>PGMR</td>
</tr>
</tbody>
</table>

![Diagram showing Gaussian mixture reduction](image-url)
Reduction Methods
Overview

top-down approaches

- greedy methods
- iteratively replace two Gaussians with one
- chosen according to a deviation measure
  (West, Williams, Runnalls)

bottom-up approach

- constructive method
- starts with one Gaussian
- adds components as necessary
- progressive approximation
- PGMR

![Graphs showing reduction methods](image)
Reduction Methods
Overview

top-down approaches
- greedy methods
- iteratively replace two Gaussians with one
- chosen according to a deviation measure (West, Williams, Runnalls)

bottom-up approach
- constructive method
- starts with one Gaussian
- adds components as necessary
- progressive approximation
- PGMR - state-of-the-art
Clustering Method
Overview

**Gaussian Mixture Reduction via Clustering (GMRC)**
- modular three-step algorithm
- input:
  - $\tilde{\eta}$ (parameter vector of the original mixture)
  - $K$ (number of reduced components)
- output:
  - $\eta$ (parameter vector of the reduced mixture)

---

**Conception**
- each component $\eta_i$ is interpreted as point (site) in a space with an underlying deviation measure
Clustering Method

Overview

Gaussian Mixture Reduction via Clustering (GMRC)

- modular three-step algorithm
- input:
  - $\tilde{\eta}$ (parameter vector of the original mixture)
  - $K$ (number of reduced components)
- output:
  - $\eta$ (parameter vector of the reduced mixture)

![Diagram]

$\eta \leftarrow \text{Initialization}(\tilde{\eta}, K)$

$\eta \leftarrow \text{Clustering}(\tilde{\eta}, \eta)$

$\eta \leftarrow \text{Refinement}(\tilde{\eta}, \eta)$

Conception

- each component $\eta_i$ is interpreted as point (site) in a space with an underlying deviation measure
Clustering Method
Initialization Step

- compute a preliminary solution \( \eta \) (i.e. using West, Runnalls, \ldots)
  \( \rightarrow \) initial cluster centers
- associate each original component (site) \( \tilde{\eta}_i \) with the nearest component of the reduced mixture
- replace each cluster center with a new one, retaining mean and variance of the associated sites
Clustering Method
Initialization Step

- compute a preliminary solution $\eta$ (i.e. using West, Runnalls, ...)
  $\rightarrow$ initial cluster centers
- associate each original component (site) $\tilde{\eta}_i$ with the nearest component of the reduced mixture
- replace each cluster center with a new one, retaining mean and variance of the associated sites
Clustering Method
Initialization Step

- compute a preliminary solution $\eta$ (i.e. using West, Runnalls, ...)
  → initial cluster centers
- associate each original component (site) $\tilde{\eta}_i$ with the nearest component of the reduced mixture
- replace each cluster center with a new one, retaining mean and variance of the associated sites
Clustering Method
Initialization Step

- compute a preliminary solution $\eta$ (i.e. using West, Runnalls, . . .) → initial cluster centers
- associate each original component (site) $\tilde{\eta}_i$ with the nearest component of the reduced mixture
- replace each cluster center with a new one, retaining mean and variance of the associated sites
Clustering Method
Initialization Step

- compute a preliminary solution $\eta$ (i.e. using West, Runnalls, . . . ) → initial cluster centers
- associate each original component (site) $\tilde{\eta}_i$ with the nearest component of the reduced mixture
- replace each cluster center with a new one, retaining mean and variance of the associated sites
Clustering Method

Clustering Step

- greedy approach
- based on Lloyd’s algorithm (k-means algorithm):
  - associate each site $\tilde{\eta}_i$ with the ‘nearest’ center $\eta_j$
  - recompute centers according to the current association
  - repeat until the deviation no longer changes or is good enough

**determine the ‘nearest’ center**

- associate site $\tilde{\eta}_i$ with each center $\eta_j$
- temporarily update the affected centers
- compute change in deviation between updated reduced and original mixture (ISD)
- retain association with smallest deviation
Clustering Method
Clustering Step

- greedy approach
- based on Lloyd’s algorithm (k-means algorithm):
  - associate each site $\tilde{\eta}_i$ with the ‘nearest’ center $\eta_j$
  - recompute centers according to the current association
  - repeat until the deviation no longer changes or is good enough

Determine the ‘nearest’ center

- associate site $\tilde{\eta}_j$ with each center $\eta_j$
- temporarily update the affected centers
- compute change in deviation between updated reduced and original mixture (ISD)
- retain association with smallest deviation
Clustering Method

Clustering Step

- greedy approach
- based on Lloyd’s algorithm (k-means algorithm):
  - associate each site $\tilde{\eta}_i$ with the ‘nearest’ center $\eta_j$
  - recompute centers according to the current association
  - repeat until the deviation no longer changes or is good enough

**determine the ’nearest’ center**

- associate site $\tilde{\eta}_j$ with each center $\eta_j$
- temporarily update the affected centers
- compute change in deviation between updated reduced and original mixture (ISD)
- retain association with smallest deviation
Clustering Method

Refinement Step

Parameter Optimization
- optimize parameter vector $\eta$ w.r.t. ISD
  $$\min_{\eta} \int_{R} (\tilde{f}(x; \tilde{\eta}) - f(x; \eta))^2 \, dx$$
- non-linear optimization problem → Newton approach
- finds local optimum

Weight Optimization
- system of linear equations
- finds global optimum
Clustering Method

Refinement Step

Parameter Optimization

- Optimize parameter vector $\eta$ w.r.t. ISD

$$\min_{\eta} \int_{\mathbb{R}} \left( \tilde{f}(x; \tilde{\eta}) - f(x; \eta) \right)^2 dx$$

- Non-linear optimization problem $\rightarrow$ Newton approach

- Finds local optimum

Weight Optimization

- System of linear equations

- Finds global optimum
Results

Overview

Simulation Setup

- Office PC (Intel Core2 Duo E8400)
- OpenSUSE 11.0
- Matlab 7.7.0 (R2008b)

- reduction of mixtures with \( N \in \{40, 120, 200, 500, 1000\} \) components down to \( K = 10 \)
- each evaluated with 1 000 simulation runs
Results

Approximation Quality

![Graph showing approximation quality vs. nSD % for different algorithms (PGMR, GMRC, GMRC+West, GMRC++, Runnalls, West, Williams).]
Results

Running Time

![Graph showing running time for different methods]

- PGMR
- GMRC
- GMRC+West
- GMRC++
- Runnalls
- West
- Williams

June 9, 2009

Dennis Schieferdecker – GRK 1194: Self-organizing Sensor-Actuator-Networks
Clustering-based Gaussian Mixture Reduction
## Results

### Impact of Individual Steps

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Running Time (s)</th>
<th>Norm. ISD</th>
</tr>
</thead>
<tbody>
<tr>
<td>GMRC complete</td>
<td>2.793 ± 0.052</td>
<td>0.658 ± 0.494</td>
</tr>
<tr>
<td>w. random init.</td>
<td>1.135 ± 0.045</td>
<td>1.272 ± 1.561</td>
</tr>
<tr>
<td>w/o clustering</td>
<td>1.742 ± 0.043</td>
<td>0.774 ± 0.872</td>
</tr>
<tr>
<td>w/o refinement</td>
<td>2.737 ± 0.036</td>
<td>1.697 ± 0.432</td>
</tr>
<tr>
<td>Runnalls</td>
<td>1.678 ± 0.024</td>
<td>3.606 ± 0.752</td>
</tr>
</tbody>
</table>

(initialization with Runnalls' algorithm; \( N = 200, K = 10 \))

- A good initial solution is mandatory.
- The clustering step primarily improves variance.
- Refinement has a single-most impact on approximation quality.
## Results

### Impact of Individual Steps

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Running Time</th>
<th>Norm. ISD</th>
</tr>
</thead>
<tbody>
<tr>
<td>GMRC complete</td>
<td>2.793 ± 0.052s</td>
<td>0.658 ± 0.494</td>
</tr>
<tr>
<td>w. random init.</td>
<td>1.135 ± 0.045s</td>
<td>1.272 ± 1.561</td>
</tr>
<tr>
<td>w/o clustering</td>
<td>1.742 ± 0.043s</td>
<td>0.774 ± 0.872</td>
</tr>
<tr>
<td>w/o refinement</td>
<td>2.737 ± 0.036s</td>
<td>1.697 ± 0.432</td>
</tr>
<tr>
<td>Runnalls</td>
<td>1.678 ± 0.024s</td>
<td>3.606 ± 0.752</td>
</tr>
</tbody>
</table>

(initialization with Runnalls’ algorithm; $N = 200, K = 10$)

- A good initial solution is mandatory
- Clustering step primarily improves variance
- Refinement has single-most impact on approximation quality
## Results
### Impact of Individual Steps

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Running Time</th>
<th>Norm. ISD</th>
</tr>
</thead>
<tbody>
<tr>
<td>GMRC complete</td>
<td>2.793 ± 0.052s</td>
<td>0.658 ± 0.494</td>
</tr>
<tr>
<td>w. random init.</td>
<td>1.135 ± 0.045s</td>
<td>1.272 ± 1.561</td>
</tr>
<tr>
<td>w/o clustering</td>
<td>1.742 ± 0.043s</td>
<td>0.774 ± 0.872</td>
</tr>
<tr>
<td>w/o refinement</td>
<td>2.737 ± 0.036s</td>
<td>1.697 ± 0.432</td>
</tr>
<tr>
<td>Runnalls</td>
<td>1.678 ± 0.024s</td>
<td>3.606 ± 0.752</td>
</tr>
</tbody>
</table>

(initialization with Runnalls’ algorithm; $N = 200, K = 10$)

- a good initial solution is mandatory
- clustering step primarily improves variance
- refinement has single-most impact on approximation quality
Results

Impact of Individual Steps

<table>
<thead>
<tr>
<th>algorithm</th>
<th>running time</th>
<th>norm. ISD</th>
</tr>
</thead>
<tbody>
<tr>
<td>GMRC complete</td>
<td>2.793 ± 0.052s</td>
<td>0.658 ± 0.494</td>
</tr>
<tr>
<td>w. random init.</td>
<td>1.135 ± 0.045s</td>
<td>1.272 ± 1.561</td>
</tr>
<tr>
<td>w/o clustering</td>
<td>1.742 ± 0.043s</td>
<td>0.774 ± 0.872</td>
</tr>
<tr>
<td>w/o refinement</td>
<td>2.737 ± 0.036s</td>
<td>1.697 ± 0.432</td>
</tr>
<tr>
<td>Runnalls</td>
<td>1.678 ± 0.024s</td>
<td>3.606 ± 0.752</td>
</tr>
</tbody>
</table>

(initialization with Runnalls’ algorithm; $N = 200, K = 10$)

- a good initial solution is mandatory
- clustering step primarily improves variance
- refinement has single-most impact on approximation quality
Results

Visualization

![Graph showing the comparison of original and GMRC data](image-url)
Results
Visualization

\[ f(x) \]

- Original
- PGMR
Results Visualization

Clustering-based Gaussian Mixture Reduction

KIT – The cooperation of Forschungszentrum Karlsruhe GmbH and Universität Karlsruhe (TH)
Clustering-based Gaussian Mixture Reduction

Original and Williams' approximations.
Conclusion

Summary
- novel top-down, global reduction algorithm
- competitive w.r.t. current state-of-the-art (PGMR)
- combines algorithmic and numerical ideas

Outlook
- extension to multivariate Gaussian mixtures
- impact of different clustering methods and deviation measures
- adaptive reduction of components
Conclusion

Summary

- novel top-down, global reduction algorithm
- competitive w.r.t. current state-of-the-art (PGMR)
- combines algorithmic and numerical ideas

Outlook

- extension to multivariate Gaussian mixtures
- impact of different clustering methods and deviation measures
- adaptive reduction of components
Thank you for your attention!

time for questions