Mixtures of Truncated Basis Functions

Helge Langseth, Thomas D. Nielsen, Rafael Rumí, and Antonio Salmerón

This work is supported by an Abel grant from Iceland, Liechtenstein, and Norway through the EEA Financial Mechanism (Nils mobility project). Supported and Coordinated by Universidad Complutense de Madrid.
Mixture of Truncated Exponentials potential (Moral, Rumí, & Salmerón, 2001)

$$f(z) = \begin{cases} 
-0.0172 + 0.931e^{1.27z} & \text{if } -3 \leq z < -1 \\
0.442 - 0.0385e^{-1.64z} & \text{if } -1 \leq z < 0 \\
0.442 - 0.0385e^{1.64z} & \text{if } 0 \leq z < 1 \\
-0.0172 + 0.9314e^{-1.27z} & \text{if } 1 \leq z < 3 
\end{cases}$$
The ground work

Mixture of Truncated Exponentials potential (Moral, Rumí, & Salmerón, 2001)

\[ f(z) = \begin{cases} 
-0.0172 + 0.931e^{1.27z} & \text{if } -3 \leq z < -1 \\
0.442 - 0.0385e^{-1.64z} & \text{if } -1 \leq z < 0 \\
0.442 - 0.0385e^{1.64z} & \text{if } 0 \leq z < 1 \\
-0.0172 + 0.9314e^{-1.27z} & \text{if } 1 \leq z < 3 
\end{cases} \]

Mixture of Truncated Polynomials potential (Shenoy & West, 2009)

The MoP potential is similar to the MTE, except that a polynomial is used as a basis function instead of an exponential.

- Appears as a natural consequence of a Taylor expansion of the target \( f \).
The MoTBF model

MoTBF potential

The *Mixture of Truncated Basis Functions potential* is defined as for MTEs and MoPs, just abstracting the basis functions from exponentials/polynomials to functions $\psi_i(\cdot)$, where $\{\psi_i(\cdot)\}_{i=1}^{\infty}$ define a collection of real basis functions that are *closed under product and marginalisation*. 
The MoTBF model

MoTBF potential

The Mixture of Truncated Basis Functions potential is defined as for MTEs and MoPs, just abstracting the basis functions from exponentials/polynomials to functions $\psi_i(\cdot)$, where $\{\psi_i(\cdot)\}_{i=1}^{\infty}$ define a collection of real basis functions that are closed under product and marginalisation.

MoTBF rationale and properties

- More flexible than MTEs and MoPs (MoPs and MTEs are special cases of MoTBFS).
- Any distribution function can be approximated arbitrarily well by an MoTBF distribution (under certain assumptions).
- The class of MoTBF potentials is closed under combination and marginalisation $\Rightarrow$ (approximate) inferences in any hybrid domain using Shenoy-Shafer propagation.
The MoTBF model

MoTBF potential

The *Mixture of Truncated Basis Functions potential* is defined as for MTEs and MoPs, just abstracting the basis functions from exponentials/polynomials to functions $\psi_i(\cdot)$, where $\{\psi_i(\cdot)\}_{i=1}^{\infty}$ define a collection of real basis functions that are closed under product and marginalisation.

MoTBF rationale and properties

- More flexible than MTEs and MoPs (MoPs and MTEs are special cases of MoTBFS).
- Any distribution function can be approximated arbitrarily well by an MoTBF distribution (under certain assumptions).
- The class of MoTBF potentials is closed under combination and marginalisation $\Rightarrow$ (approximate) inferences in any hybrid domain using Shenoy-Shafer propagation.

The subject of this talk

An efficient (online tradeoff between complexity and quality) method for translating a density to an MoTBF.
A quick recall of how to do approximations in $\mathbb{R}^n$:

We want to approximate the vector $f = (3, 2, 5)$ with

- A vector along $e_1 = (1, 0, 0)$. 
A quick recall of how to do approximations in $\mathbb{R}^n$:

We want to approximate the vector $\mathbf{f} = (3, 2, 5)$ with

- A vector along $\mathbf{e}_1 = (1, 0, 0)$. Best choice is $\langle \mathbf{f}, \mathbf{e}_1 \rangle \cdot \mathbf{e}_1 = (3, 0, 0)$. 
Geometry of approximations

A quick recall of how to do approximations in $\mathbb{R}^n$:

We want to approximate the vector $\mathbf{f} = (3, 2, 5)$ with

- A vector along $\mathbf{e}_1 = (1, 0, 0)$. Best choice is $\langle \mathbf{f}, \mathbf{e}_1 \rangle \cdot \mathbf{e}_1 = (3, 0, 0)$.
- Now, add a vector along $\mathbf{e}_2$.
  - Best choice is $\langle \mathbf{f}, \mathbf{e}_2 \rangle \cdot \mathbf{e}_2$, independently of the choice made for $\mathbf{e}_1$.
  - Also, the choice we made for $\mathbf{e}_1$ is still optimal since $\mathbf{e}_1 \perp \mathbf{e}_2$. 
A quick recall of how to do approximations in $\mathbb{R}^n$:

We want to approximate the vector $f = (3, 2, 5)$ with

- A vector along $e_1 = (1, 0, 0)$. Best choice is $\langle f, e_1 \rangle \cdot e_1 = (3, 0, 0)$.
- Now, add a vector along $e_2$.
  - Best choice is $\langle f, e_2 \rangle \cdot e_2$, independently of the choice made for $e_1$.
  - Also, the choice we made for $e_1$ is still optimal since $e_1 \perp e_2$.

All of this maps over to approximations of functions!

We only need a definition of the inner product and the equivalent to orthonormal basis vectors.
Inner product

For two functions $f(x)$ and $g(x)$ defined on $\Omega \subseteq \mathbb{R}$, define

$$\langle f, g \rangle = \int_{\Omega} f(x) g(x) \, dx.$$
Function approximations

Inner product

For two functions $f(x)$ and $g(x)$ defined on $\Omega \subseteq \mathbb{R}$, define

$$\langle f, g \rangle = \int_{\Omega} f(x) g(x) \, dx.$$  

Finding an orthonormal basis

If we have a set of functions $\{\psi_k\}_{k=1}^{\infty}$ in $x$, e.g., $\{1, \exp(-x), \exp(x), \exp(-2x), \ldots\}$ we can “orthonormalise” the set using the Gram-Schmidt process.
**Inner product**

For two functions $f(x)$ and $g(x)$ defined on $\Omega \subseteq \mathbb{R}$, define

$$\langle f, g \rangle = \int_{\Omega} f(x) g(x) \, dx.$$ 

**Finding an orthonormal basis**

If we have a set of functions $\{\psi_k\}_{k=1}^{\infty}$ in $x$, e.g., $\{1, \exp(-x), \exp(x), \exp(-2x), \ldots\}$ we can “orthonormalise” the set using the Gram-Schmidt process.

We now have the machinery to *approximate any univariate function* (including densities) as a sum of basis-functions.
Doing the approximation

The function $f$ is approximated by $\hat{f} = \sum_i \langle f, \phi_i \rangle \cdot \phi_i$ (known as the Generalised Fourier Series approximation).

Properties

- $\hat{f}$ minimises $\int_{\Omega} (f(x) - \hat{f}(x))^2 \, dx$.
- The approximation can be made arbitrarily good even without splitting $\Omega$ into sub-intervals (given some technicalities that are fulfilled for both the polynomials and the exponentials).
- The approximation supports the Shenoy-Shafer scheme.
Example: Polynomials vs. the Std. Gaussian

\[ \hat{f} = 0.4362 \cdot \phi_1 \]

\[ \hat{f} = 0.4362 \cdot \phi_1 + 0 \cdot \phi_2 + (-0.1927) \cdot \phi_3 \]

\[ \hat{f} = 0.4362 \cdot \phi_1 + 0 \cdot \phi_2 + \cdots + 0.0052 \cdot \phi_9 \]

- Constant (\( \phi_1 \)) chosen so that the probability mass of \( \hat{f} \) is allocated correctly.
- \( \phi_i \), for \( i > 1 \), does not contribute to the probability mass:

\[ \int_{\Omega} \phi_i(x) \, dx \equiv 0 \text{ because } \phi_i \perp \phi_1 \text{ for } i > 1. \]
\( \hat{f} \) is chosen to minimise \( \int_{\Omega} (f(x) - \hat{f}(x))^2 \, dx \), but is \( \hat{f} \) a density?

1. \( \int_{\Omega} \hat{f}(x) \, dx \equiv \int_{\Omega} f(x) \, dx \) as long as \( \phi_1(x) \) is a constant.

2. However, we cannot guarantee \( \hat{f}(x) \geq 0, x \in \Omega \) in general.

Approximation of the \( \chi^2 \) distribution, and zooming in towards the origin.
Positivity condition

\( \hat{f} \) is chosen to minimise \( \int_{\Omega} (f(x) - \hat{f}(x))^2 \, dx \), but is \( \hat{f} \) a density?

1. \( \int_{\Omega} \hat{f}(x) \, dx \equiv \int_{\Omega} f(x) \, dx \) as long as \( \phi_1(x) \) is a constant.

2. However, we cannot guarantee \( \hat{f}(x) \geq 0, x \in \Omega \) in general.

Approximation of the \( \chi^2 \) distribution, and zooming in towards the origin.

\( \rightsquigarrow \) We must optimise the error subject to the constraint that \( \hat{f}(x) \geq 0, x \in \Omega. \)
Property: If \( h_k(x) = \sum_{i=1}^{k} \alpha_i \phi_i(x) \) then

\[
(error(f, h_k))^2 = \sum_{i=1}^{k} (\langle f, \phi_i \rangle - \alpha_i)^2 + \sum_{i=k+1}^{\infty} \langle f, \phi_i \rangle^2.
\]

Ensuring positivity

We now have the following optimisation problem:

Minimise \( \sum_{i=1}^{k} (\langle f, \phi_i \rangle - \alpha_i)^2 \)

Subject to \( \sum_{i=1}^{k} \alpha_i \phi_i(x) \geq 0, x \in \Omega, \) and \( \alpha_1 = \langle f, \phi_1 \rangle \)

This is a convex optimisation problem, and can be solved using “standard” methods (semi-definite programming).
A probabilistic interpretation

So far, \( \hat{f} \) lacks a probabilistic interpretation. However, it can be shown that

\[
D \left( f \| \hat{f} \right) \leq \frac{\left( \text{error}(f, \hat{f}) \right)^2}{\min_{x \in \Omega} \hat{f}(x)}.
\]
A probabilistic interpretation

So far, \( \hat{f} \) lacks a probabilistic interpretation. However, it can be shown that

\[
D \left( f \parallel \hat{f} \right) \leq \frac{\left( \text{error}(f, \hat{f}) \right)^2}{\min_{x \in \Omega} \hat{f}(x)}.
\]

The final estimation procedure

The updated optimisation problem wrt. \( (\alpha, \xi) \) is convex:

**Minimise**

\[
\frac{1}{\xi} \cdot \sum_{i=1}^{k} \left( \langle f, \varphi_i \rangle - \alpha_i \right)^2
\]

**Subject to**

\[
\sum_{i=1}^{k} \alpha_i \varphi_i(x) \geq \xi, \quad x \in \Omega, \quad \text{and} \quad \alpha_1 = \langle f, \varphi_1 \rangle
\]

This is an approximation to the KL-minimising MoTBF.
Definition: Conditional MoTBF density, version 0.1

\[ \hat{f}(y|x) = \sum_{j=1}^{m} \alpha_j \phi_j(y, x), \ x \in I_k \]

Requirement:
For each \( x \), we must have that \( \int_y \hat{f}(y|x) dy = 1 \).
Definition: Conditional MoTBF density, version 0.1

\[ \hat{f}(y|x) = \sum_{j=1}^{m} \alpha_j \phi_j(y,x), \; x \in I_k \]

Requirement:
For each \( x \), we must have that \( \int_y \hat{f}(y|x) \, dy = 1 \).

Consequence: We define conditional MoTBFs to only depend on their conditioning variable(s) through the relevant hypercube, and not the numerical value.

Definition: Conditional MoTBF density, version 0.2

\[ \hat{f}(y|x) = \sum_{j=1}^{m_k} \alpha^{(k)}_j \phi_j(y), \; x \in I_k \]

What is the target density?

**Question**

Assume we have $I_k$, and want to define $\hat{f}(y|x) = \sum_{j=1}^{m} \alpha_j \phi_j(y)$ for $x \in I_k$. What should it approximate?

Choose $\hat{f}(y|x) \sim f(y|x_0)$, for some fixed $x_0$.

- $x_0$ chosen as midpoint of hypercube $I_k$.
- $x_0$ chosen as mass-center of hypercube $I_k$.

Mixtures of Truncated Basis Functions

Conditional Distributions
What is the target density?

Question

Assume we have $\mathcal{I}_k$, and want to define $\hat{f}(y|x) = \sum_{j=1}^{m} \alpha_j \phi_j(y)$ for $x \in \mathcal{I}_k$. What should it approximate?

Choose

$$\hat{f}(y|x) \sim f(y|x \in \mathcal{I}_k) = \int_{x \in \mathcal{I}_k} f(y|x) f(x|x \in \mathcal{I}_k) \, dx \approx \sum_t f(y|x_t) f(x_t|x \in \mathcal{I}_k).$$
Cost-Benefit translation of arbitrary models

Algorithm

1. Initialisation: All distributions use 1 BF, all conditionals defined on only one hypercube
2. For each marginal: Calculate the KL gain of adding a BF
3. For each conditional:
   - Calculate the KL benefit of splitting each conditioning hypercube
   - Calculate the KL benefit of adding one BF for each hypercube
4. Perform the operation with best KL benefit vs. increase in complexity
5. If KL is not good enough, goto 2
Preliminary experiments

An MOP representation of the network given by $X \sim N(0, 1)$ and $Y|X = x \sim N(0.1 \cdot x, 1)$:

- Approximation found after 15 iterations.
- 3 basis vectors used for both the parent and the child (for each interval).
Future work

- Investigate heuristics for efficient calculation of approximate KL gain (ongoing).
- How to exploit the structure of MoTBFs when doing inference (ongoing).
- How to use the current translation procedure for learning MoTBFs from data (ongoing).
- How to efficiently re-translate the model to incorporate evidence (“dynamic discretisation”).
Some experiments

Comparison with the translation procedure by Cobb, Shenoy, and Rumí (2006):

<table>
<thead>
<tr>
<th>Distribution</th>
<th>Cobb et al.</th>
<th>MoTBFs (MTE)</th>
<th>MoTBFs (MoP)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( N(0, 1) )</td>
<td>3.46 ( \cdot 10^{-4} )</td>
<td>[8] 1.93 ( \cdot 10^{-04} )</td>
<td>[10] 5.83 ( \cdot 10^{-06} )</td>
</tr>
<tr>
<td>Gamma(6,1)</td>
<td>2.10 ( \cdot 10^{-3} )</td>
<td>[20] 6.41 ( \cdot 10^{-03} )</td>
<td>[13] 7.32 ( \cdot 10^{-05} )</td>
</tr>
<tr>
<td>Gamma(8,1)</td>
<td>8.56 ( \cdot 10^{-4} )</td>
<td>[7] 5.37 ( \cdot 10^{-03} )</td>
<td>[12] 6.63 ( \cdot 10^{-05} )</td>
</tr>
<tr>
<td>Gamma(11,1)</td>
<td>2.83 ( \cdot 10^{-4} )</td>
<td>[12] 6.58 ( \cdot 10^{-04} )</td>
<td>[13] 2.67 ( \cdot 10^{-05} )</td>
</tr>
<tr>
<td>Beta(2,2)</td>
<td>2.62 ( \cdot 10^{-6} )</td>
<td>[9] 3.98 ( \cdot 10^{-05} )</td>
<td>[15] 2.50 ( \cdot 10^{-05} )</td>
</tr>
<tr>
<td>Beta(2.7,1.3)</td>
<td>3.30 ( \cdot 10^{-4} )</td>
<td>[8] 5.33 ( \cdot 10^{-04} )</td>
<td>[11] 2.71 ( \cdot 10^{-04} )</td>
</tr>
<tr>
<td>Beta(1.3,2.7)</td>
<td>3.30 ( \cdot 10^{-4} )</td>
<td>[8] 5.33 ( \cdot 10^{-04} )</td>
<td>[16] 2.50 ( \cdot 10^{-04} )</td>
</tr>
<tr>
<td>LogNormal(0,0.25)</td>
<td>3.30 ( \cdot 10^{-4} )</td>
<td>[19] 9.05 ( \cdot 10^{-03} )</td>
<td>[11] 9.11 ( \cdot 10^{-05} )</td>
</tr>
<tr>
<td>LogNormal(0,0.5)</td>
<td>9.90 ( \cdot 10^{-5} )</td>
<td>[22] 5.69 ( \cdot 10^{-03} )</td>
<td>[10] 5.92 ( \cdot 10^{-03} )</td>
</tr>
<tr>
<td>LogNormal(0,1)</td>
<td>6.47 ( \cdot 10^{-3} )</td>
<td>[6] 4.05 ( \cdot 10^{-02} )</td>
<td>[6] 4.04 ( \cdot 10^{-02} )</td>
</tr>
</tbody>
</table>