Beyond Backpropagation: Uncertainty Propagation

NEIL LAWRENCE
UNIVERSITY OF SHEFFIELD
@lawrennd
\[ f(x) = \sum_{j=1}^{k} u_j \phi(a_j) \]

\[ a_j = \sum_{i=1}^{p} v_{i,j} x_i \]
\[ \nu_{i,j} \sim N(0, \alpha_u) \]

\[ u_i \sim N(0, \alpha_u) \]

\[ \mathbb{E}(u(Wx)u^T V) = \sum_{i=1}^{n} \frac{1}{\sqrt{\lambda_i}} \sum_{i=1}^{n} \left( f_i(x; \mathbf{u}; \mathbf{v}; y) \right)^2 - \frac{n}{2} \log 2\pi \sigma^2 \]
\[
\log p(y|x, u, V) = \frac{1}{2\sigma^2} \sum_{i=1}^{n} (y_i - f(x_i; u, V))^2 - \frac{1}{2\alpha_u} u_i^2 - \frac{1}{2\alpha_v} v_{i,j}^2 + \text{const.}
\]
GLOBAL INFORMATION STORAGE CAPACITY IN OPTIMALLY COMPRESSED BYTES

ConvNets Developed

SVMs dominate NIPS

1986 ANALOG 2.6 EXABYTES
1992 DIGITAL 0.02 EXABYTES
2002 “BEGINNING OF THE DIGITAL AGE”

Zero Mean Gaussian Sample

\[ \chi(t) \]

samples from Gaussian process

\[ y(t) \]

covariance function \( c(t, t') \)
Gaussian Processes

\[ p(f|x) \]

\[ p(y_1|f_1) \]
\[ p(f|y_k, x) \]
\[ p(y_2|f_2) \]
$u_i \sim N(0, \alpha_u)$

$v_{i,j} \sim \ast$
Bayesian Optimization

Open Data Science and Africa

Challenge

• “Whole pipeline challenge”
• Make software available
• Teach summer schools
• Support local meetings
  • Publicity in the Guardian
• Opportunities to deploy pipeline solutions
Disease Incidence for Malaria

The graph shows the incidence of malaria cases from 2003 to 2013, with data points plotted on a log scale. The trend line indicates a gradual decline in cases over the years.
Uganda

• Spatial models of disease
Deployed with UN Global Pulse Lab

http://pulselabkampala.ug/hmis/
Results

\[
\frac{dp_{TF}(t)}{dt} = s_f m_{TF}(t) - d_f p_{TF}(t)
\]

\[
\frac{dm_i(t)}{dt} = s_i p_{TF}(t) - d_i m_i(t)
\]
David: Have we thrown out the baby with the bathwater?
\[ g(x) = f_9\left(f_8\left(f_7\left(f_6(\cdots)\right)\right)\right) \]
\[ p(y, w|x) = p(y|w, x)p(w) \]

\[ p(y|x) = \int p(y|w, x)p(w)dw \]
\[ \log p(y|x) \geq \int q(w) \log \frac{p(y|w, x)p(w)}{q(w)} \, dw \]
\[
\mathcal{L}(y|x) = \sum_{i=1}^{n} \left( y_i \log q(y_i|w) - \frac{1}{2} \| w \|^2 \right) q(w) - \text{KL}[q(w) || p(w)] + \text{const}
\]
\[ f \mid x \sim N(0, K_{ff}) \]

\[ k_{ff}(x_i, x'_i) = \alpha \exp\left( -\frac{||x_i-x'_i||^2}{2\ell^2} \right) \]

\[ y_i \mid f_i \sim N(0, \sigma^2) \]
\[ p(y, f|x) = p(y|f)p(f|x) \]

\[ p(y|x) = \int p(y|f)p(f|x)df \]
\[
p(y, f, u) = p(y|f)p(f|x)p(x|u)
\]

\[
p(y|u, x)p(u) = \int p(y|f)p(f|u, x)dfp(u)
\]
\[ p(y, u|x) = p(y|u, x)p(u) \]

\[ p(y|x) = \int p(y|u, x)p(u)du \]

u looks like a parameter but we can change the dimensionality of u.
two Gaussian processes: apply bound recursively

\[ \int p(y|f_5)p(f_5|f_4)p(f_4|f_3)p(f_3|f_2)p(f_1|\mathbf{x}) df \]

\[ g(x) = f_5 \left( f_4 \left( f_3 \left( f_2 \left( f_1(x) \right) \right) \right) \right) \]
Render Gaussian Non Gaussian

\[ y = f(x) \]
Stochastic Process Composition

• A new approach to forming stochastic processes
• Mathematical composition:
  \[ g(x) = f_1 \left( f_2 (f_3 (x)) \right) \]
• Properties of resulting process highly non-Gaussian
• Allows for hierarchical structured form of model.
<table>
<thead>
<tr>
<th>model</th>
<th>MSE (train)</th>
<th>MSE (test)</th>
</tr>
</thead>
<tbody>
<tr>
<td>mlp (200 iters)</td>
<td>108.5</td>
<td>1185.1</td>
</tr>
<tr>
<td>mlp (converged)</td>
<td>24.0</td>
<td>1338.2</td>
</tr>
<tr>
<td>gp</td>
<td>59.2</td>
<td>1095.4</td>
</tr>
<tr>
<td>deep gp (2)</td>
<td>146.2</td>
<td>833.7</td>
</tr>
<tr>
<td>deep gp (3)</td>
<td>182.5</td>
<td>843.6</td>
</tr>
</tbody>
</table>

One hundred hidden nodes, one hundred inducing points
<table>
<thead>
<tr>
<th>data set</th>
<th>$n$</th>
<th>$p$</th>
<th>GP</th>
<th>Sparse GP</th>
<th>Deep GP</th>
</tr>
</thead>
<tbody>
<tr>
<td>housing</td>
<td>506</td>
<td>13</td>
<td>2.78±0.54</td>
<td>2.77±0.60</td>
<td>2.69±0.49</td>
</tr>
<tr>
<td>redwine</td>
<td>588</td>
<td>11</td>
<td>0.72±0.06</td>
<td>0.62±0.04</td>
<td>0.62±0.04</td>
</tr>
<tr>
<td>energy1</td>
<td>768</td>
<td>8</td>
<td>0.48±0.07</td>
<td>0.50±0.07</td>
<td>0.49±0.07</td>
</tr>
<tr>
<td>energy2</td>
<td>768</td>
<td>8</td>
<td>0.59±0.08</td>
<td>1.66±0.21</td>
<td>1.39±0.49</td>
</tr>
<tr>
<td>concrete</td>
<td>1030</td>
<td>8</td>
<td>5.26±0.67</td>
<td>5.81±0.62</td>
<td>5.66±0.62</td>
</tr>
</tbody>
</table>
Classical Latent Variables
Classical Treatment

• Assume \textit{a priori} that

\[ x \sim N(0, I) \]

• Relate linearly to \( y \)

\[ y = Wx + \epsilon \]

• Framework covers many classical models PCA, Factor Analysis, ICA
Classical Treatment

• Assume *a priori* that

\[ x \sim N(0, I) \]

• Relate to \( y \) using neural net

\[ y = f(x; u, V) + \epsilon \]

• Optimise over \( u, V \)

David applied importance sampling
MATLAB Demo

- demo_2016_05_03_iclr.m
New Treatment

- Assume *a priori* that

\[ f(x) \sim N(0, K) \]

- Relate to \( y \) using neural net

\[ y = f(x) + \epsilon \]

- Optimise over \( x \)

Originally inspired by density nets
MATLAB Demo

- demo_2016_05_03_iclr.m
\[ \mathcal{L}(y|u) = \langle \log \hat{p}(y|u, x) \rangle_{q(x)} - KL(q(x)||p(x)) \]

Expected log likelihood

Dissimilarity between \( q(x) \) and \( p(x) \)

Model remains linear in \( u \)
\[ \hat{p}(y|u, x) \geq N(y|m, \sigma^2 I) \exp \left( \frac{c_{ii}}{2\sigma^2} \right) \]

\[ c_{ii} = k_{ii}(x_i, x_i) - k_{iu}(x_i)K_{uu}^{-1}k_{ui}(x_i) \]

\[ m(x) = K_{fu}(x)K_{uu}^{-1}u \]

model is not linear in \( x \)
\[ \langle k_{ii}(x_i, x_i) \rangle_{q(x_i)} \]

\[ \langle K_{fu}(x) \rangle_{q(x)} \]

\[ \langle K_{uf}(x)K_{fu}(x) \rangle_{q(x)} \]
Use Abstraction for Complex Systems

High Level Ideas

Stratification of Concepts

Low Level Mechanisms
Example: Motion Capture Modelling
MATLAB Demo

- demo_2016_05_03_iclr.m
Modelling Digits

Optimised weights

Outputs obtained after sampling from (certain nodes) of layers 5,4,2,1

Generic feature encoding

Local feature encoding
MATLAB Demo

- demo_2016_05_03_iclr.m
Numerical Issues
Health

- Complex system
- Scarce data
- Different modalities
- Poor understanding of mechanism
- Large scale

PLoS Comp Bio, Nature Communications
To Find Out More

• Gaussian Process Summer School

• Posters at ICLR:
  • Recurrent Gaussian Processes
  • Variationally Auto-Encoded Deep Gaussian Processes

• Python software for GPs (GPy)
  • [https://github.com/SheffieldML/GPy/](https://github.com/SheffieldML/GPy/)

David’s “Gaussian Process Basics” talk
Thank you
Neil Lawrence
http://inverseprobability.com
@lawrennd