Deep Gaussian Processes: Theory and Applications

Petar M. Djurić

AI Institute
Stony Brook University, Stony Brook, NY 11794, USA

February 7, 2019
Outline

- Introduction
- Gaussian processes
- Deep Gaussian processes
- Applications
- Conclusions
Introduction

- Probabilistic modeling allows for representing and modifying uncertainty about models and predictions.
- This is done according to well defined rules.
- Probabilistic modeling has a central role in machine learning, cognitive science and artificial intelligence.
The Concept of Uncertainty

- Learning and intelligence depend on the mathematical representation of uncertainty.

- Probability theory is the main framework for handling uncertainty.

- Interestingly, in the recent progress of deep learning with deep neural networks, which are based on learning from huge amounts of data, the concept of uncertainty is somewhat bypassed.

- In the years to come, we will see further advances in artificial intelligence and machine learning within the probabilistic framework.
The Role of a Model

- To make inference from data, one needs models.
- Models can be simple (like linear models) or highly complex (like large and deep neural networks).
- In most settings, the models must be able to make predictions.
- Uncertainty plays a fundamental role in modeling observed data and in interpreting model parameters, the results of models, and the correctness of models.
The Learning

▶ Probability distributions are used to represent uncertainty.

▶ Learning from data occurs by transforming prior distributions (defined before seeing the data) to posterior distributions (after seeing the data).

▶ The optimal transformation from information-theoretic point of view is the Bayes rule.

▶ The beauty of the approach is the simplicity of the Bayes mechanism.
Gaussian Processes Regression

Essentially, a GP can be seen as the distribution of a real-valued function $f(x)$,

$$f(x) \sim \mathcal{GP}(m(x), k_f(x_i, x_j))$$

Some assumptions are often made when using GP regression:

1. the mean function $m(x) = 0$ for simplicity, and
2. the observation noise is additive white Gaussian noise for tractability.
Let $X = \{x_i\}_{i=1}^N$ and $y$ denote the collection of all input vectors and all observations, respectively, with the above assumptions, i.e.,

$$y = f(X) + \epsilon$$

where $\epsilon \sim \mathcal{N}(0, \sigma^2_\epsilon I)$. We also have

- **Likelihood**: $p(y|f) = \mathcal{N}(y|f, \sigma^2_\epsilon I)$, and

- **Prior**: $p(f|X, \theta) = \mathcal{N}(f|0, K_{ff})$, where $K_{ff} = k_f(X, X)$ and $\theta$ denote the hyper-parameters in the covariance function.
Gaussian Processes Regression (contd.)

The hyper-parameters $\theta$ can be learned from the training data $\{X, y\}$ by maximizing the log marginal likelihood

$\log p(y|X, \theta)$

$$
\log p(y|X, \theta) = \log \mathcal{N}(y|\mathbf{0}, \mathbf{K}_{ff} + \sigma^2 \mathbf{I}) \\
= \log \mathcal{N}(y|\mathbf{0}, \mathbf{K}) \\
= -\frac{1}{2} \mathbf{y}^T \mathbf{K}^{-1} \mathbf{y} - \frac{1}{2} \log |\mathbf{K}| - \frac{N}{2} \log 2\pi
$$

- The Occam’s razor is embedded in the model.
Gaussian Processes Regression (contd.)

Let \( X_* \) and \( f_* \) denote the collection of test inputs and the corresponding latent function values, respectively. Then we can express the predictive posterior as

\[
\begin{align*}
\text{Predictive posterior: } & \quad p(f_*|X_*, X, y, \theta) = \mathcal{N}(f_*|\mathbb{E}(f_*), \text{cov}(f_*)) \\
\mathbb{E}(f_*) &= [K_f(X_*, X)]K^{-1}y \\
\text{cov}(f_*) &= K_f(X_*, X*) - [K_f(X_*, X)]K^{-1}[K_f(X_*, X)]^T
\end{align*}
\]
Covariance Function

- For example: Radial basis function (RBF) or squared exponential (SE)

One dimensional form:

\[ k_{rbf}(x_i, x_j) = \sigma_f^2 \exp\left(-\frac{1}{\ell}(x_i - x_j)^2\right) \]

- \( \sigma_f^2 \) measures strength of signal, \( \frac{\sigma_f^2}{\sigma_e^2} \) is equivalent to signal-to-noise ratio (SNR).
- The characteristic length scale \( \ell \) encodes the model complexity in that dimension.
- \( r = \frac{1}{\ell} \) measures the relevance of that dimension.
- Automatic relevance determination (ARD)
Toy Example

- Goal: learn $f(x)$ from 5 noisy observations $\{x_i, y_i\}_{i=1}^5$.
- Ground truth: $y = \sin(x) + \epsilon$, $\epsilon \sim \mathcal{N}(0, \sigma_\epsilon^2)$.
- Test inputs: $x_* \in \mathbb{R}^{300 \times 1}$ equally spaced from $x = 0$ to $2\pi$.
- Test outputs: $f_* = f(x_*)$
Prior Distribution
Predictive (posterior) Distribution
Another Toy Example: The function $\sin \frac{x}{x}$
Example: Recovery of Missing Samples in FHR

- Goal: recover missing samples in FHR, using not only observed FHR but also UA samples

- Model:

\[ y_i = y(x_i) = f(x_i) + \epsilon \]

- \( y_i \): \( i \)-th sample in an FHR segment
- \( x_i = [i, u_i]' \) where \( u_i \) is the \( i \)-th UA sample
- \( \epsilon \): Gaussian white noise
- \( f(x_i) \): \( i \)-th latent noise-free FHR sample

---

CTG Segment for Experiments
CTG Segment for Experiments
Experiment I

- 120 missing samples were randomly selected, and we tried to recover their true values.
Experiment II

- The percentage of missing samples was increased from 1% to 85% with a step size of 1%.
Experiment III

- To demonstrate contribution of UA, we repeated the experiment I, but excluded $u_i$ from the input vector $x_i$. 

![Diagram showing contribution of UA signal with 95% confidence interval and mean predictions with and without UA.]
Experiment VI (An Extreme Case)

- 10 seconds of consecutive missing samples.
Limitations

- The general framework is computationally expensive, $O(N^3)$, due to the term $K_{N \times N}^{-1}$.

- Another limitation is the joint Gaussianity that is required by the definition of GPs.
Solutions

- Sparse GP, e.g., subset of data, deterministic training conditional (DTC) and fully independent training conditional (FITC) approximations

- Wrapped Gaussian processes

- Deep Gaussian processes with variational inference tackled both limitations
Deep Gaussian Processes

\[ Z \rightarrow X_{H-1} \rightarrow \cdots \rightarrow X_2 \rightarrow X_1 \rightarrow Y \]

- \( Y \in \mathbb{R}^{N \times d_y} \): observations, output of the network
  - \( N \) is the number of observation vectors.
  - \( d_y \) is the dimension of the vectors \( y_n \).

- \( \{X_h\}_{h=1}^{H-1} \): intermediate latent states
  - dimensions \( \{d_h\}_{h=1}^{H-1} \) are potentially different.

- \( Z \in \mathbb{R}^{N \times d_z} \): the input to the network
  - \( Z \) is observed for supervised learning.
  - \( Z \) is unobserved for unsupervised learning.
Deep Gaussian Processes (contd.)

- The joint Gaussianity limitation is overcome because nonlinear mappings generally will not preserve Gaussianity.
- DGPs immediately introduce intractabilities.
- One way of handling the difficulties is by introducing a set of inducing points and where within the variational framework, sparsity and a tractable lower bound on the marginal likelihood are obtained.
Example: Functions Sampled From DGP

- Gaussianity limitation is overcome by nonlinear function composition.
Example: Learning a Step Function

- Standard GP (top), 2 and 4 layer DGP (middle, bottom)
- DGPs achieved much better performance.

---

Deep GPs and Deep Neural Networks (a comparison)

▶ A single layer of fully connected neural network with an independent and identically distributed (iid) prior over its parameters and with an infinite width is equivalent to a GP.

▶ Therefore, deep GPs are equivalent to neural networks with multiple, infinitely wide hidden layers.

▶ Mappings of a DGP are governed by its GPs instead of activation functions.

▶ A DGP allows for propagations and quantifications of uncertainties through each layer as a fully Bayesian probabilistic model.

▶ There is ARD at each layer.
Generative Process

The generative process takes the form:

\[ x_{nl} = f^X_l(z_n) + \epsilon^X_{nl}, \quad l = 1, \ldots, d_x, \quad z_n \in \mathbb{R}^{dz} \]
\[ y_{ni} = f^Y_i(x_n) + \epsilon^Y_{ni}, \quad i = 1, \ldots, d_y, \quad x_n \in \mathbb{R}^{dx} \]

\( \epsilon^X_{nl} \) and \( \epsilon^Y_{ni} \) are additive white Gaussian processes.
Generative Process (contd.)

- We assume $Z$ is unobserved with a prior $p(Z) = \mathcal{N}(Z|0, I)$
- If we have specific prior knowledge about $Z$, we should quantify this knowledge into a prior accordingly.
Inference

The inference takes the reverse route, i.e., we observe high-dimensional data $\mathbf{Y}$, and we learn the low-dimensional manifold $\mathbf{Z}$ (of dimension $d_z$, where $d_z < d_x < d_y$) that is responsible for generating $\mathbf{Y}$. 
Inference Challenges

The learning requires maximization of the log marginal likelihood,

$$\log p(Y) = \log \int_{X,Z} p(Y|X)p(X|Z)p(Z)dXdZ$$

which is intractable.
Augmentation of Probability Space

- Original probability space:

$$p(Y, F^Y, F^X, X, Z) = p(Y|F^Y)p(F^Y|X)p(X|F^X) \times p(F^X|Z)p(Z)$$

- Augmentation using inducing points:
  - $$U^X = f^X(\tilde{Z}), \tilde{Z} \in \mathbb{R}^{N_p \times d_z}$$ and $$U^X \in \mathbb{R}^{N_p \times d_x}$$
  - $$U^Y = f^Y(\tilde{X}), \tilde{X} \in \mathbb{R}^{N_p \times d_x}$$ and $$\tilde{X} \in \mathbb{R}^{N_p \times d_x}$$
  - $$N_p \leq N$$
Augmentation of Probability Space

Augmented probability space:

\[
p(Y, F^Y, F^X, X, Z, U^Y, U^X, \tilde{X}, \tilde{Z})
= p(Y|F^Y)p(F^Y|U^Y, X)p(U^Y|\tilde{X})
\times p(X|F^X)p(F^X|U^X, Z)p(U^X|\tilde{Z})p(Z)
\]

Problematic terms:

\(A = p(F^Y|U^Y, X)\)

\(B = p(F^X|U^X, Z)\)
Variational Inference

- A variational distribution: $Q = q(U^Y)q(X)q(U^X)q(Z)$
- By Jensen’s inequality:
  \[
  \log p(Y) \geq \mathcal{F}_v = \int Q \cdot A \cdot B \log G \, dF^Y dX dF^X dZ dU^X dU^Y
  \]
- The function $G$ is defined as:
  \[
  G(Y, F^Y, X, F^X, Z, U^X, U^Y) = \frac{p(Y | F^Y)p(U^Y)p(X | F^X)p(U^X)p(Z)}{Q}.
  \]
- $\mathcal{F}_v$ is tractable for a collection of covariance functions, since $A$ and $B$ are canceled out in $G$. 
Studying Complex Systems

Used principles

- algorithmic compressibility,
- locality, and
- deep probabilistic modeling.
Applications
Applications-contd.
Applications-contd.\textsuperscript{3}

\textsuperscript{3}Figures obtained by Sima Mofakham and Chuck Mikell.
Example: Binary pH-based Classification

- Goal: to have the DGP classify CTG recordings into health and unhealthy classes.

- Features:
  - 14 FHR features
  - 6 (categorical) UA features

- Labeling:
  - Positive (unhealthy): pH < 7.1
  - Negative (healthy): pH > 7.2

---

Structure of DGP: our DGP network had two layers, and in each layer, we set the initial latent dimension to five.

Performance metrics:
1. Sensitivity (true positive rate)
2. Specificity (true negative rate)
3. Geometric mean of specificity and sensitivity
Features

**Table: Features for FHR**

<table>
<thead>
<tr>
<th>Category</th>
<th>Feature</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time domain</td>
<td>Mean, Standard deviation, STV, STI, LTV, LTI</td>
</tr>
<tr>
<td>Non-linear</td>
<td>Poincaré SD1, Poincaré SD2, CCM</td>
</tr>
<tr>
<td>Frequency</td>
<td>VLF, LF, MF, HF, ratio</td>
</tr>
</tbody>
</table>

**Table: Features for UA**

<table>
<thead>
<tr>
<th>Normal (0)</th>
<th>Abnormal (1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequency ≤ 8 contractions</td>
<td>&gt; 8 UC (tachysystole)</td>
</tr>
<tr>
<td>Duration &lt; 90s</td>
<td>&gt; 90s</td>
</tr>
<tr>
<td>Increased tonus With toco</td>
<td>Prolonged &gt; 120s</td>
</tr>
<tr>
<td>Interval A Interval – peak to peak</td>
<td>&lt; 2min</td>
</tr>
<tr>
<td>Interval B Interval – offset of UC to onset of next UC</td>
<td>&lt; 1min</td>
</tr>
<tr>
<td>Rest time &gt; 50%</td>
<td>&lt; 50%</td>
</tr>
</tbody>
</table>
Classification Results

- Support vector machine (SVM) was used as benchmarking model.

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Feature</th>
<th>Specificity</th>
<th>Sensitivity</th>
<th>Geometric Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM</td>
<td>FHR</td>
<td>0.82</td>
<td>0.73</td>
<td>0.77</td>
</tr>
<tr>
<td></td>
<td>FHR+UA</td>
<td>0.82</td>
<td>0.82</td>
<td>0.82</td>
</tr>
<tr>
<td>Deep GP</td>
<td>FHR</td>
<td>0.91</td>
<td>0.73</td>
<td>0.82</td>
</tr>
<tr>
<td></td>
<td>FHR+UA</td>
<td>0.82</td>
<td>0.91</td>
<td>0.86</td>
</tr>
</tbody>
</table>
Unsupervised Learning for FHR Recordings

- Goal: to have the DGP learn informative low-dimensional latent spaces that can generate the recordings.

- Labeling:
  - pH-based labeling combined with obstetrician’s evaluation.
  - Labels are only used for evaluation of learning results.

- Data:
  - The last 30 minutes of 10 FHR recordings, $\mathbf{Y} \in \mathbb{R}^{10 \times 7200}$.
  - 3 of them are abnormal and 7 are normal.
Performance Metric and Network Structure

- Performance metric: the number of errors in the latent space for one nearest neighbor.

- Structure of DGP: a five-layer DGP, and the initial dimensions of the latent spaces in the layers were $d_{X_{1:5}} = [6, 5, 5, 4, 3]^T$. 
Automatic Structure Learning
Visualization of the Latent Spaces with 2-D Projection.

- Red: the normal recordings
- Blue: the abnormal recordings
- Pixel intensity: proportional to precision
- The total errors in layers 1 to 5 are 2, 2, 1, 1, 0, respectively.
Example: Deep Gaussian Processes with Convolutional Kernels

- Goal: multi-class image classification
- Database: MNIST (handwritten digits)
- Methods:
  1. SGP: Sparse Gaussian processes
  2. DGP: Deep Gaussian processes
  3. CGP: Convolutional Gaussian processes
  4. CDGP: Convolutional deep Gaussian processes

---

## MNIST

<table>
<thead>
<tr>
<th>Model</th>
<th>Layer 1</th>
<th>Layer 2</th>
<th>Layer 3</th>
<th>Layer 4</th>
<th>Accuracy%</th>
<th>NLPP</th>
</tr>
</thead>
<tbody>
<tr>
<td>SGP</td>
<td>RBF</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>97.48</td>
<td>–</td>
</tr>
<tr>
<td>DGP1</td>
<td>RBF</td>
<td>RBF</td>
<td>–</td>
<td>–</td>
<td>97.94</td>
<td>0.073</td>
</tr>
<tr>
<td>DGP2</td>
<td>RBF</td>
<td>RBF</td>
<td>RBF</td>
<td>–</td>
<td>97.99</td>
<td>0.070</td>
</tr>
<tr>
<td>CGP1</td>
<td>Conv</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>95.59</td>
<td>0.170</td>
</tr>
<tr>
<td>CGP2</td>
<td>Wconv</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>97.54</td>
<td>0.103</td>
</tr>
<tr>
<td><strong>CDGP1</strong></td>
<td>Wconv</td>
<td>RBF</td>
<td>–</td>
<td>–</td>
<td><strong>98.66</strong></td>
<td><strong>0.046</strong></td>
</tr>
<tr>
<td>CDGP2</td>
<td>Conv</td>
<td>RBF</td>
<td>–</td>
<td>–</td>
<td>98.53</td>
<td>0.536</td>
</tr>
<tr>
<td>CDGP3</td>
<td>Conv</td>
<td>RBF</td>
<td>RBF</td>
<td>–</td>
<td>98.40</td>
<td>0.055</td>
</tr>
<tr>
<td>CDGP4</td>
<td>Conv</td>
<td>RBF</td>
<td>RBF</td>
<td>RBF</td>
<td>98.41</td>
<td>0.051</td>
</tr>
<tr>
<td>CDGP5</td>
<td>Wconv</td>
<td>Wconv</td>
<td>RBF</td>
<td>–</td>
<td>98.44</td>
<td>0.048</td>
</tr>
<tr>
<td>CDGP6</td>
<td>Wconv</td>
<td>Wconv</td>
<td>RBF</td>
<td>RBF</td>
<td>98.60</td>
<td>0.046</td>
</tr>
</tbody>
</table>
Example: Identification of Atmospheric Variable Using Deep Gaussian Processes

- Goal: modeling temperature using meteorological variables (features).

- Domain of interest: $25\text{Km} \times 25\text{Km}$ around the nuclear power plant in Krško, Slovenia.

- Features: relative humidity, atmosphere stability, air pressure, global solar radiation, wind speed.

---

The Geographical Features of the Surrounding Terrain

- The plant and its measurement station (marked as STOLP – Postaja) are situated in the basin surrounded by hills and valleys, which influence micro-climate conditions.
One-Step-Ahead Prediction

- Prediction results:
Example: Deep Gaussian Process for Crop Yield Prediction Based on Remote Sensing Data

- **Goal**: predicting crop yields before harvest
- **Model**: CNN and LSTM combined with GP

<table>
<thead>
<tr>
<th>Year</th>
<th>Ridge</th>
<th>Tree</th>
<th>DNN</th>
<th>Baselines</th>
<th>LSTM</th>
<th>LSTM + GP</th>
<th>CNN</th>
<th>CNN + GP</th>
</tr>
</thead>
<tbody>
<tr>
<td>2011</td>
<td>9.00</td>
<td>7.98</td>
<td>9.97</td>
<td>Baselines</td>
<td>5.83</td>
<td>5.77</td>
<td>5.76</td>
<td>5.7</td>
</tr>
<tr>
<td>2012</td>
<td>6.95</td>
<td>7.40</td>
<td>7.58</td>
<td>Baselines</td>
<td>6.22</td>
<td>6.23</td>
<td>5.91</td>
<td>5.68</td>
</tr>
<tr>
<td>2013</td>
<td>7.31</td>
<td>8.13</td>
<td>9.20</td>
<td>Baselines</td>
<td>6.39</td>
<td>5.96</td>
<td>5.50</td>
<td>5.83</td>
</tr>
<tr>
<td>2014</td>
<td>8.46</td>
<td>7.50</td>
<td>7.66</td>
<td>Baselines</td>
<td>6.42</td>
<td>5.70</td>
<td>5.27</td>
<td>4.89</td>
</tr>
<tr>
<td>2015</td>
<td>8.10</td>
<td>7.64</td>
<td>7.19</td>
<td>Baselines</td>
<td>6.47</td>
<td>5.49</td>
<td>6.40</td>
<td>5.67</td>
</tr>
<tr>
<td>Avg</td>
<td>7.96</td>
<td>7.73</td>
<td>8.32</td>
<td>Baselines</td>
<td>6.27</td>
<td>5.83</td>
<td>5.77</td>
<td>5.55</td>
</tr>
</tbody>
</table>

---

Comparing County-Level Error Maps

▶ The color represents the prediction error in bushel per acre.
Conclusions

- A case was made for using probability theory in treating uncertainties in inference from data.
- Deep probabilistic modeling based on deep Gaussian processes was addressed.
- The use of DGPs in studying complex interacting systems was addressed.
- Applications in various fields using DGPs were described.
- Although the development of DGPs is still in its relatively early stages, DGPs showed great potentials in many challenging machine learning tasks.