On Statistical Thinking in Deep Learning

Yee Whye Teh
Statistics @ Oxford
DeepMind

http://csml.stats.ox.ac.uk/people/teh/
Probabilistic and Deep Learning @ NeurIPS ’87–’15

Analysis from a Bayesian nonparametric dynamic topic model
Poisson random fields for dynamic feature models [Perrone et al 2016]
Statistics vs Machine Learning vs Artificial Intelligence

“No, Machine Learning is not just glorified Statistics”
Joe Davison (2018) Towards Data Science

“When you’re fundraising, it’s AI. When you’re hiring, it’s ML. When you’re implementing, it’s logistic regression.”

— everyone on Twitter ever
Actual Contents of Talk

- Meta-learning stochastic processes with neural processes:

Few Shot Image Classification

Training
- terrier
- beagle
- labrador
- cat
- poodle

Test
- ?
- ?
- ?
- ?
- ?

Credit: Andrei Rusu, miniImageNet
## Few Shot Image Classification

<table>
<thead>
<tr>
<th>Paper (only latest shown)</th>
<th>miniImageNet test accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>5-way 1-shot</td>
</tr>
<tr>
<td>Gidaris &amp; Komodakis (2018)</td>
<td>56.20 ± 0.86%</td>
</tr>
<tr>
<td>Bauer &amp; Rojas-Carulla (2017)</td>
<td>56.30 ± 0.40%</td>
</tr>
<tr>
<td>Oreshkin et al., (2018)</td>
<td>59.50 ± 0.30%</td>
</tr>
<tr>
<td>Siyuan Qiao et al. (2017)</td>
<td>59.60 ± 0.41%</td>
</tr>
<tr>
<td>LEO (Rusu et al 2018)</td>
<td>60.06 ± 0.05%</td>
</tr>
</tbody>
</table>

Credit: [Rusu, Sygnowski et al, ArXiv:1807.05960]
Optimisation Perspective on Meta-Learning

Meta-parameter $\eta$

Training Data $\{(x_i,y_i)\}$

Learning Algorithm $A_\eta : \text{train} \Rightarrow \theta$

Predictor $f_{\eta,\theta}(x)$

Test Data

Test Loss 20%
Optimisation Perspective on Meta-Learning

Training Data \( \{(x_i, y_i)\} \)

Learning Algorithm \( A_\eta : \text{train} \Rightarrow \theta \)

Predictor \( f_{\eta,\theta}(x) \)

Test Data

Test Loss

- 20%
- 0%
- 20%
Optimisation Perspective on Meta-Learning

Meta-parameter $\eta$

Training Data $\{(x_i, y_i)\}$

Learning Algorithm train $\Rightarrow \theta$

Predictor $f_{\eta, \theta}(x)$

Test Data

Test Loss

20%

0%

20%
Many (Supervised) Meta-Learning Methods

- LSTM meta-learner, learning to optimize
- Matching nets
- Prototypical nets
- MANN
- MAML
- LEO
- MANN
- MAML
- LEO
Probabilistic Perspective on Meta-Learning

\[ \hat{f}(y | x) \]

- terrier
- beagle
- labrador
- cat
- poodle

Gaussian Process

Yee Whye Teh
Probabilistic Perspective on Meta-Learning

$\Hat{f}(y|x)$

Neural Process

$\begin{align*}
    x_i & \rightarrow y_i \\
    f & \\
\end{align*}$

- terrier
- beagle
- labrador
- cat
- poodle
Specifying Stochastic Processes

- Gaussian processes are typically described via marginal distributions:

\[
\begin{pmatrix}
    f(x_1) \\
    f(x_2) \\
    \vdots \\
    f(x_t)
\end{pmatrix}
\sim \mathcal{N}
\begin{pmatrix}
    \begin{pmatrix}
        \mu(x_1) \\
        \mu(x_2) \\
        \vdots \\
        \mu(x_t)
    \end{pmatrix}, \\
    \begin{pmatrix}
        K(x_1, x_1) & K(x_1, x_2) & \cdots & K(x_1, x_t) \\
        K(x_2, x_1) & K(x_2, x_2) & \cdots & K(x_2, x_t) \\
        \vdots & \vdots & \ddots & \vdots \\
        K(x_t, x_1) & K(x_t, x_2) & \cdots & K(x_t, x_t)
    \end{pmatrix}
\end{pmatrix}
\]
Specifying Stochastic Processes

- Gaussian processes can equivalently be described via its conditional distributions:

\[ f(x_{t+1})|f(x_1) = y_1, \ldots, f(x_t) = y_t \]

\[ \sim \mathcal{N}(\mu(x_{t+1}) + K_{t+1,1:t}K_{1:t,1:t}^{-1}y_{1:t}, K_{t+1,t+1} - K_{t+1,1:t}K_{1:t,1:t}K_{1:t,t+1}) \]

- In general, stochastic processes can also be described using a consistent family of conditional distributions:

\[ \mathbb{P}(f(x_{t+1}) = y_{t+1}|f(x_1) = y_1, \ldots, f(x_t) = y_t) \]

for training sets \{x_{1:t}, y_{1:t}\} and test sets \{x_{t+1}, y_{t+1}\}.
Neural Processes: Learning Neural Stochastic Processes

- Use neural networks to parameterise and learn the conditional distributions.

![Diagram showing neural processes learning](image)
Neural Processes: Learning Neural Stochastic Processes

$max \sum_{\eta} \sum_{w} p(w) \sum_{j=1}^{s} \log p_{\eta}(y_{t+j}^w | x_{t+j}^w, \{x_i^w, y_i^w\}_{i=1}^t)$
Neural Processes

Task = Function on 1D space.

Given training points, use neural processes to predict mean and std of function values at other locations.
Image Super-resolution

Task = Image = Function on 2D space.

Bottom half prediction → Super-resolution
Image Super-resolution

context

4 x 4

8 x 8

16 x 16

32 x 32

target prediction at different resolutions x 3

context

8 x 8

16 x 16

32 x 32

target prediction

Interpolation baselines

256 x 256
Efficient Model-based Reinforcement Learning

\[ h_1, a_1 \rightarrow s_2, r_1 \]
\[ h_2, a_2 \rightarrow s_3, r_2 \]
\[ h_3, a_3 \rightarrow s_4, r_3 \]
\[ \vdots \]
\[ h_t, a_t \rightarrow s_{t+1}, r_t \]

\[ \hat{p}(s' | s, a) \]
\[ \hat{r}(s, a) \]

Plan

state, reward

action
Cart Pole
Adversarial Testing of RL Agents

\[ M \]

\[ p_g \]

Goal position

Start position

\[ p_s \]

\[ r(M, p_s, p_g | A) \]

Powerful RL Agent

Jonathan Schwarz
Alex Galashov
Adversarial Testing of RL Agents

Bayesian Optimization

\[ \min_{M, p_s, p_g} r(M, p_s, p_g | A) \]

\[ (M, p_s, p_g, A) \sim p(\mathcal{T}) \quad - \text{training & holdout samples (agents, mazes, positions)} \]
Adversarial Testing of RL Agents

Bayesian optimisation iterations

1st iteration
Mean return: 340
Permutation-Invariance in Neural Processes

Training data

Test data

inputs

outputs

encoding

r₁, r₂, r₃

Σ

Z

η

meta-parameter

parameter

sloth, hotdog, barrette

inputs

outputs

Yee Whye Teh
Actual Contents of Talk

● Meta-learning stochastic processes with neural processes:

Characterising Permutation-Invariant Functions

- Function \( h : \mathcal{X}^n \rightarrow \mathcal{Y} \) is permutation-invariant,
  \[ h(\pi \cdot (x_1, \ldots, x_n)) = h(x_{\pi(1)}, \ldots, x_{\pi(n)}) = h(x_1, \ldots, x_n) \]

- Can we characterise the class of permutation-invariant functions?

- If we use neural networks to parameterise permutation-invariant functions, how should we choose the architecture?

- Given an architecture choice, can the neural network approximate well any arbitrary permutation-invariant function?
Characterising Permutation-Invariant Functions

\[ \sum \text{product? mean? max? min? median?} \]
Characterising Permutation-Invariant Functions

\[ r_1 = \sum \text{product? mean? max? min? median?} \]

\[ r_2 \]

\[ r_3 \]

\[ y \]
Functional Symmetry Properties

- Function $h : \mathcal{X}^n \rightarrow \mathcal{Y}^n$ is permutation-equivariant,

$$h(x_1, \ldots, x_n) = (y_1, \ldots, y_n)$$

$$h(\pi \cdot (x_1, \ldots, x_n)) = \pi \cdot (y_1, \ldots, y_n) = \pi \cdot h(x_1, \ldots, x_n)$$

- Group $G$ acting on input space $\mathcal{X}$ and output space $\mathcal{Y}$.
  - $G$-invariant:
    $$h(g \cdot x) = h(x)$$
  - $G$-equivariant:
    $$h(g \cdot x) = g \cdot h(x)$$
Probabilistic Symmetries

- A distribution $P$ for a random sequence $X_n = (X_1, \ldots, X_n)$ is exchangeable if
  \[ P(X_1, \ldots, X_n) = P(\pi \cdot (X_1, \ldots, X_n)) \]
  \[ P(X_1 \in B_1, \ldots, X_n \in B_n) = P(X_{\pi(1)} \in B_1, \ldots, X_{\pi(n)} \in B_n) \]

- Exchangeability is permutation-invariance of $P$.

- $X_\mathbb{N}$ is infinitely exchangeable if all length $n$ prefixes are exchangeable.

- de Finetti’s Theorem:
  \[ X_\mathbb{N} \text{ is infinitely exchangeable} \iff X_i \mid Q \sim_{iid} Q \text{ for some random } Q. \]
Probabilistic Symmetries for Conditional Distributions

- A conditional distribution $P(Y \mid X)$ is a stochastic relaxation for a function $Y = h(X)$.

- $P(Y \mid X)$ is $G$-invariant if:
  
  $$P(Y \mid X) = P(Y \mid g \cdot X)$$
  
  $$P(Y \in B \mid X \in A) = P(Y \in B \mid g \cdot X \in A)$$

- $P(Y \mid X)$ is $G$-equivariant if:
  
  $$P(Y \mid X) = P(g \cdot Y \mid g \cdot X)$$

- Can we characterise the class of permutation-invariant conditional distributions?
Empirical Measure

- de Finetti’s Theorem may fail for finitely exchangeable sequences.

- The empirical measure of $X_n$ is

$$\mathbb{M}_{X_n}(\cdot) = \sum_{i=1}^{n} \delta_{X_i}(\cdot)$$

- The empirical measure is a sufficient statistic: $P$ is exchangeable iff

$$P(X_n \in \cdot | \mathbb{M}_{X_n} = m) = \mathbb{U}_m(\cdot)$$

where $\mathbb{U}_m$ is the uniform distribution over all sequences $(x_1, \ldots, x_n)$ with empirical measure $m$. 
Noise Outsourcing

- If $X$ and $Y$ are random variables in “nice” (e.g. Borel) spaces $\mathcal{X}$ and $\mathcal{Y}$, then there are a random variable $\eta \sim U[0,1]$ with $\eta \perp X$ and a function $h : [0,1] \times \mathcal{X} \mapsto \mathcal{Y}$ such that

$$(X, Y) =_{a.s.} (X, h(\eta, X))$$

- Furthermore, if there is statistic $S(X)$ with $X \perp Y \mid S(X)$, then

$$(X, Y) =_{a.s.} (X, h(\eta, S(X)))$$
Probabilistic Permutation-Invariance

- Now suppose we have random variables $X_n$ and $Y$.
  - $Y$ is conditionally permutation-invariant given $X_n$.
  - $X_n$ is marginally permutation-invariant (exchangeable).
- The empirical measure is a sufficient statistic for $X_n$.
- It is also an adequate statistic for $Y$ given $X_n$:
  $$P(Y | X_n = x_n) = P(Y | M_{X_n} = M_{X_n})$$
  We have the conditional independence $X_n \perp Y | M_{X_n}$.
- Noise outsourcing...
  $$\begin{align*}
  (X_n, Y) &= a.s. (X_n, h(\eta, M_{X_n}))
  \end{align*}$$
Probabilistic Permutation-Invariance

\[ X_1 \rightarrow \delta_{X_1} \rightarrow Y \]
\[ X_2 \rightarrow \delta_{X_2} \rightarrow Y \]
\[ X_3 \rightarrow \delta_{X_3} \rightarrow Y \]
Probabilistic Permutation-Invariance

\[ X_1 \rightarrow r_1 \quad X_2 \rightarrow r_2 \quad X_3 \rightarrow r_3 \]

\[ \sum \rightarrow Y \quad \eta \]
Functional Permutation-Invariance

\[ X_1 \rightarrow r_1 \rightarrow Y \]
\[ X_2 \rightarrow r_2 \rightarrow Y \]
\[ X_3 \rightarrow r_3 \rightarrow Y \]
Probabilistic Permutation-Equivariance

- Now suppose we have random sequences $X_n$ and $Y_n$.
  - $Y_n$ is conditionally permutation-equivariant given $X_n$.
  - $X_n$ is marginally permutation-invariant (exchangeable).

- Also suppose that $Y_i \perp Y_n \setminus Y_i \mid X_n$ for each $i$.

- Then:
  $$(X_n, (Y_1, \ldots, Y_n)) =_{a.s.} (X_n, (h(\eta_1, X_1, M_{X_n}), \ldots, h(\eta_n, X_n, M_{X_n})))$$
  for outsourced noise $(\eta_i)$ that is mutually independent and independent of $X_n$. 
Probabilistic Permutation-Equivariance

\[ X_1, X_2, X_3 \rightarrow r_1, r_2, r_3 \rightarrow \sum \rightarrow Y_1, Y_2, Y_3 \]
Composing Invariant and Equivariant Modules

\[ X_1 \rightarrow r_1 \rightarrow \sum \rightarrow Y \]
\[ X_2 \rightarrow r_2 \rightarrow \sum \rightarrow Y \]
\[ X_3 \rightarrow r_3 \rightarrow \sum \rightarrow Y \]

INV
Composing Invariant and Equivariant Modules

\[ X_1 \rightarrow Y_1 \]
\[ X_2 \rightarrow Y_2 \]
\[ X_3 \rightarrow Y_3 \]

INV

EQV
Composing Invariant and Equivariant Modules

\[ X_1 \xrightarrow{\text{EQV}} \xrightarrow{\text{EQV}} \xrightarrow{\text{EQV}} Y_1 \]
\[ X_2 \xrightarrow{\text{EQV}} \xrightarrow{\text{EQV}} \xrightarrow{\text{EQV}} Y_2 \]
\[ X_3 \xrightarrow{\text{EQV}} \xrightarrow{\text{EQV}} \xrightarrow{\text{EQV}} Y_3 \]
Composing Invariant and Equivariant Modules

$X_1$  $X_2$  $X_3$  $Y$
Attentive Neural Processes
Maximal Invariant and Maximal Equivariant

- Let $G$ be a compact group.

- A maximal invariant is a statistic $M : \mathcal{X} \mapsto \mathcal{S}$ such that

$$M(g \cdot x) = M(x) \quad \forall g \in G, x \in \mathcal{X}$$

$$M(x_1) = M(x_2) \Rightarrow \exists g \in G : x_1 = g \cdot x_2$$

- A maximal equivariant $\tau : \mathcal{X} \mapsto G$ satisfies

$$\tau(g \cdot x) = g \cdot \tau(x)$$
Probabilistic and Functional Symmetries

- Let $G$ be a compact group and $X$ be marginally $G$-invariant.

- Let $M$ be a maximal invariant, then
  
  $Y$ is conditionally $G$-invariant given $X \Leftrightarrow (X, Y) =_{a.s.} (X, h(\eta, M(X)))$
  
  for outsourced noise $\eta$ independent of $X$ and a function $h$.

- If a maximal equivariant $\tau$ exists and $G_X \subset G_Y$ a.s., then
  
  $Y$ is conditionally $G$-equivariant given $X \Leftrightarrow (X, Y) =_{a.s.} (X, h(\eta, X))$
  
  for outsourced noise $\eta$ independent of $X$ and a function $h$ that is $G$-equivariant in its second argument.
Concluding Remarks

● Neural processes allow us to learn domain-specific priors from data.
  ○ Focussing on predictive (conditional) distributions with latent processes marginalised out works very well [Garnelo et al 2018b, Le et al 2018].
  ○ Learnt conditional distributions are not guaranteed to be consistent.

● Tools from probabilities symmetries, sufficiency and adequacy allowed us to answer questions about neural architectures under symmetry.
  ○ Framework extends to graph and array structured data with node exchangeability.
  ○ How to relax assumptions of conditional independence of outputs?
Thank You!

- Collaborators, colleagues, mentors
- Openings @ Oxford: http://www.stats.ox.ac.uk/vacancies/
  - Director of Statistical Consultancy
  - Florence Nightingale Bicentennial Fellowship (5 year “super-postdocs”)