ML in Space (MLSS Moscow, 2019)

Yarin Gal

yarin@cs.ox.ac.uk

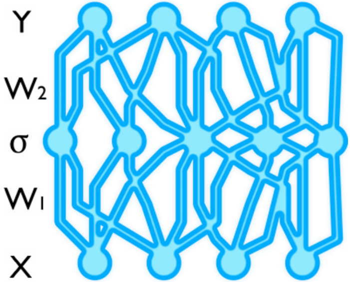
Pillar I: Deep learning

Conceptually simple models

Data: $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N\}$, $\mathbf{Y} = \{\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_N\}$
Model: given matrices $\mathbf{W}$ and non-linear func. $\sigma(\cdot)$, define "network"

$$\tilde{\mathbf{y}}_i(\mathbf{x}_i) = \mathbf{W}_2 \cdot \sigma\big(\mathbf{W}_1 \mathbf{x}_i\big)$$

Objective: find $\mathbf{W}$ for which $\tilde{\mathbf{y}}_i(\mathbf{x}_i)$ is close to $\mathbf{y}_i$ for all $i \leq N$.



Conceptually simple models

Data: $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N\}$, $\mathbf{Y} = \{\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_N\}$
Model: given matrices $\mathbf{W}$ and non-linear func. $\sigma(\cdot)$, define "network"

$$\tilde{\mathbf{y}}_i(\mathbf{x}_i) = \mathbf{W}_2 \cdot \sigma\big(\mathbf{W}_1 \mathbf{x}_i\big)$$

Objective: find $\mathbf{W}$ for which $\tilde{\mathbf{y}}_i(\mathbf{x}_i)$ is close to $\mathbf{y}_i$ for all $i \leq N$.

Deep learning is awesome ✓

- Simple and modular
- Huge attention from practitioners and engineers
- Great software tools
- Scales with data and compute
- Real-world impact

... but has many issues ✗

- What does a model not know?
- Uninterpretable black-boxes
- Easily fooled (AI safety)
- Lacks solid mathematical foundations (mostly ad hoc)
- Crucially relies on big data

Why should I care about uncertainty?

- We need a way to tell **what our model knows** and what not.
  - We train a model to recognise dog breeds

Why should I care about uncertainty?

- We need a way to tell **what our model knows** and what not.
  - We train a model to recognise dog breeds
  - And are given a cat to classify

Why should I care about uncertainty?

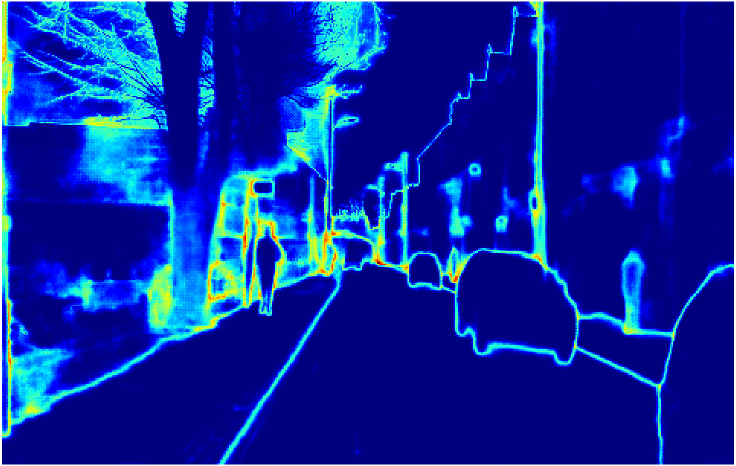
- We need a way to tell **what our model knows** and what not.
  - We train a model to recognise dog breeds
  - And are given a cat to classify
  - What would you want your model to do?

Why should I care about uncertainty?

- We need a way to tell **what our model knows** and what not.
  - We train a model to recognise dog breeds
  - And are given a cat to classify
  - What would you want your model to do?
  - Similar problems in *decision making*, *physics*, *life science*, etc.

Why should I care about uncertainty?

- We need a way to tell **what our model knows** and what not.
- Uncertainty gives insights into the black-box when it fails —where am I not certain?



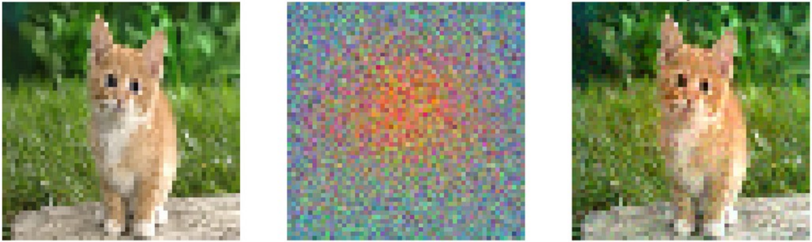
Why should I care about uncertainty?

UNIVERSITY OF OXFORD

- We need a way to tell **what our model knows** and what not.
- Uncertainty gives insights into the black-box when it fails —where am I not certain?
- Uncertainty might even be useful to identify when attacked with adversarial examples!

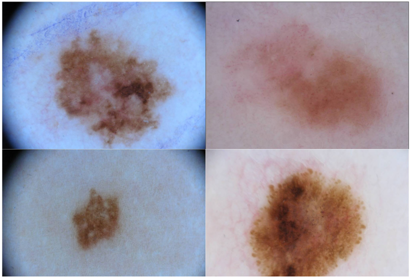
1.0% kit fox

8.0% goldfish



- Lastly, need less data if label only where **model is uncertain**: wear-and-tear in robotics, expert time in medical analysis

- We need a way to tell **what our model knows** and what not.
- Uncertainty gives insights into the black-box when it fails —where am I not certain?
- Uncertainty might even be useful to identify when attacked with adversarial examples!
- Lastly, need less data if label only where **model is uncertain**: wear-and-tear in robotics, expert time in medical analysis

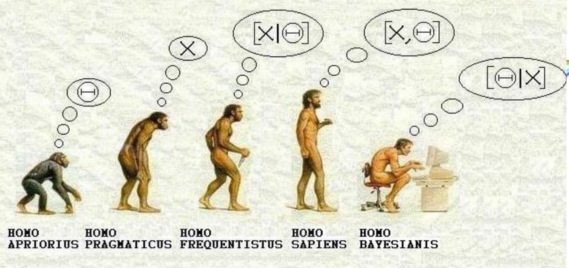
Pillar II: Bayes

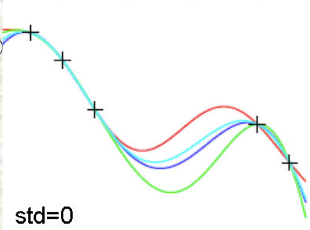
The language of uncertainty

- Probability theory
- Specifically *Bayesian probability theory* (1750!)

When applied to *Information Engineering*...

- Bayesian modelling

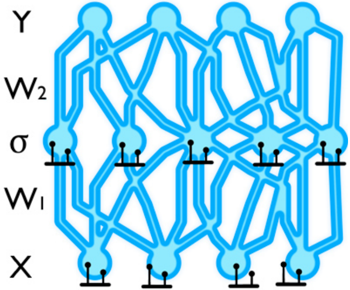




- Built on solid mathematical foundations
- Orthogonal to deep learning...

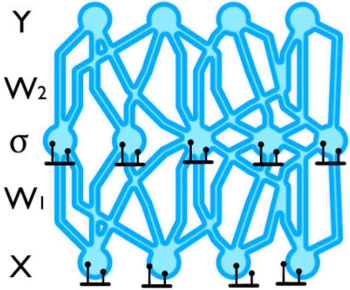
A simple way to tie the two pillars together

- "Dropout": a popular method in deep learning, cited hundreds and hundreds of times
- Works by randomly setting network units to zero
- This **somehow** improves performance and reduces over-fitting
- Used in almost **all** modern deep learning models



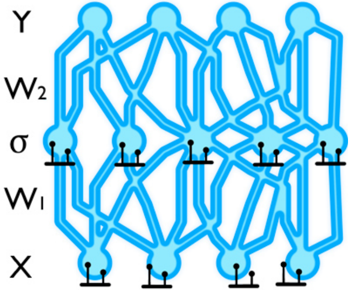
A simple way to tie the two pillars together

- "Dropout": a popular method in deep learning, cited hundreds and hundreds of times
- Works by randomly setting network units to zero
- This **somehow** improves performance and reduces over-fitting
- Used in almost **all** modern deep learning models



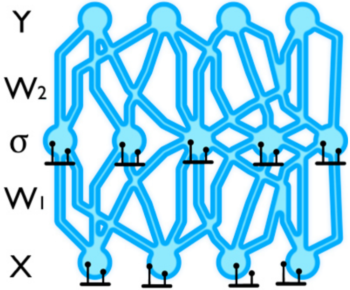
A simple way to tie the two pillars together

- "Dropout": a popular method in deep learning, cited hundreds and hundreds of times
- Works by randomly setting network units to zero
- This **somehow** improves performance and reduces over-fitting
- Used in almost **all** modern deep learning models



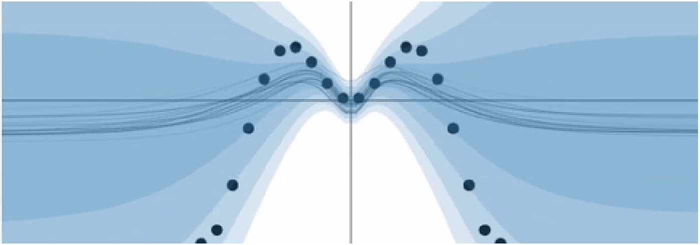
A simple way to tie the two pillars together

- "Dropout": a popular method in deep learning, cited hundreds and hundreds of times
- Works by randomly setting network units to zero
- This **somehow** improves performance and reduces over-fitting
- Used in almost **all** modern deep learning models



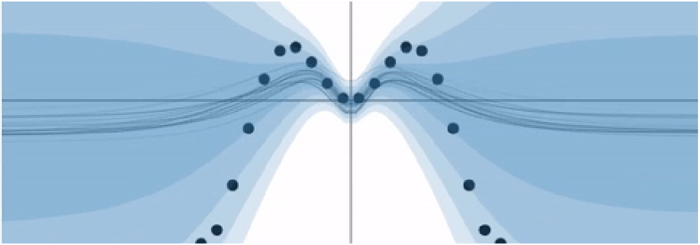
A simple way to tie the two pillars together

- Can be shown that dropout training is identical to *approximate inference in Bayesian modelling* [Gal, 2016],
- Connecting **Deep Learning to Bayesian probability theory.**
- The **mathematically grounded** connection gives a treasure trove of new research opportunities:
  - **uncertainty** in deep learning, e.g. interpretability and AI safety
  - **principled extensions** to deep learning
  - enable deep learning in **small data** domains



A simple way to tie the two pillars together

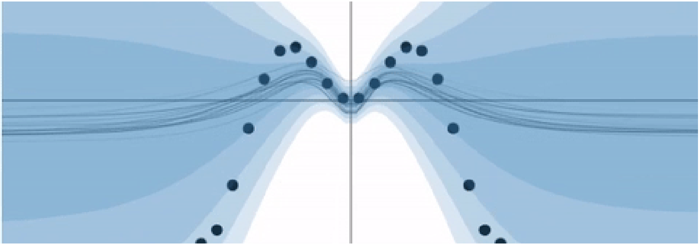
- Can be shown that dropout training is identical to *approximate inference in Bayesian modelling* [Gal, 2016],
- Connecting **Deep Learning to Bayesian probability theory.**
- The **mathematically grounded** connection gives a treasure trove of new research opportunities:
    - **uncertainty** in deep learning, e.g. interpretability and AI safety
    - **principled extensions** to deep learning
    - enable deep learning in **small data** domains



A simple way to tie the two pillars together

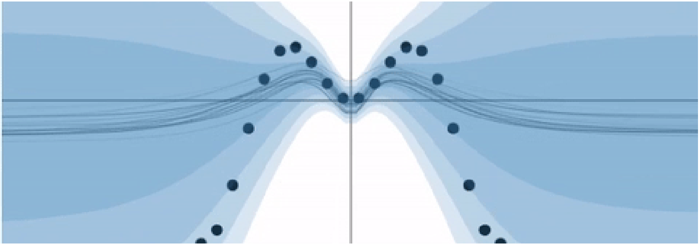
UNIVERSITY OF OXFORD

- Can be shown that dropout training is identical to *approximate inference in Bayesian modelling* [Gal, 2016],
- Connecting **Deep Learning to Bayesian probability theory.**
- The **mathematically grounded** connection gives a treasure trove of new research opportunities:
  - **uncertainty** in deep learning, e.g. interpretability and AI safety
  - **principled extensions** to deep learning
  - enable deep learning in **small data** domains



A simple way to tie the two pillars together

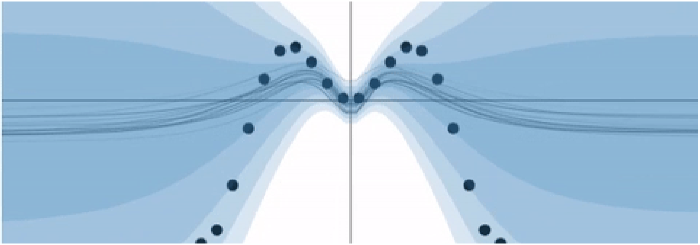
- Can be shown that dropout training is identical to *approximate inference in Bayesian modelling* [Gal, 2016],
- Connecting **Deep Learning to Bayesian probability theory.**
- The **mathematically grounded** connection gives a treasure trove of new research opportunities:
  - **uncertainty** in deep learning, e.g. interpretability and AI safety
  - **principled extensions** to deep learning
  - enable deep learning in **small data** domains



A simple way to tie the two pillars together

UNIVERSITY OF OXFORD

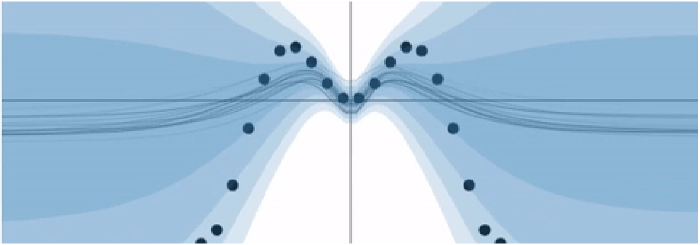
- Can be shown that dropout training is identical to *approximate inference in Bayesian modelling* [Gal, 2016],
- Connecting **Deep Learning to Bayesian probability theory.**
- The **mathematically grounded** connection gives a treasure trove of new research opportunities:
  - **uncertainty** in deep learning, e.g. interpretability and AI safety
  - **principled extensions** to deep learning
  - enable deep learning in **small data** domains



A simple way to tie the two pillars together

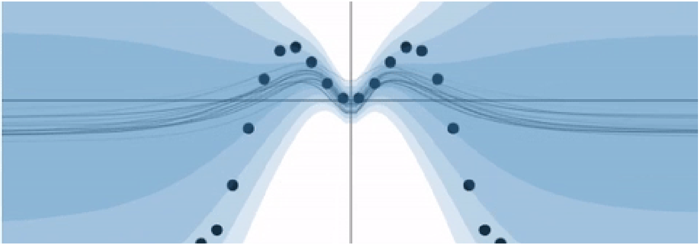
UNIVERSITY OF OXFORD

- Can be shown that dropout training is identical to *approximate inference in Bayesian modelling* [Gal, 2016],
- Connecting **Deep Learning to Bayesian probability theory.**
- The **mathematically grounded** connection gives a treasure trove of new research opportunities:
  - **uncertainty** in deep learning, e.g. interpretability and AI safety
  - **principled extensions** to deep learning
  - enable deep learning in **small data** domains



A simple way to tie the two pillars together

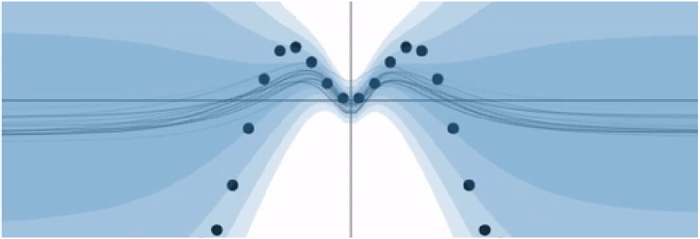
- Can be shown that dropout training is identical to *approximate inference in Bayesian modelling* [Gal, 2016],
- Connecting **Deep Learning to Bayesian probability theory.**
- The **mathematically grounded** connection gives a treasure trove of new research opportunities:
  - **uncertainty** in deep learning, e.g. interpretability and AI safety
  - **principled extensions** to deep learning
  - enable deep learning in **small data** domains



A simple way to tie the two pillars together

- Can be shown that dropout training is identical to *approximate inference in Bayesian modelling* [Gal, 2016],
- Connecting **Deep Learning to Bayesian probability theory**.
- The **mathematically grounded** connection gives a treasure trove of new research opportunities:
  - **uncertainty** in deep learning, e.g. interpretability and AI safety
  - **principled extensions** to deep learning
  - enable deep learning in **small data** domains

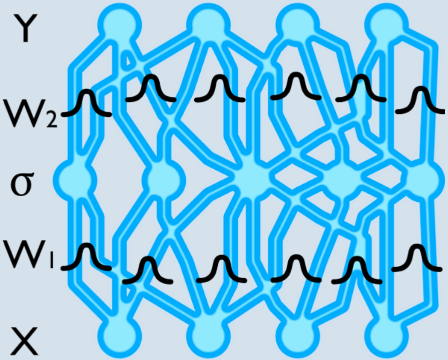
  More in a second. First, some **theory**.



Some theory

From Bayesian neural networks to Dropout

- Place **prior** $p(\mathbf{W})$ dist. on weights, making these r.v.s



- Given dataset $\mathbf{X}, \mathbf{Y}$, the r.v. $\mathbf{W}$ has a **posterior**: $p(\mathbf{W}|\mathbf{X}, \mathbf{Y})$

From Bayesian neural networks to Dropout

- Place **prior** $p(\mathbf{W})$ dist. on weights, making these r.v.s
- Given dataset $\mathbf{X}, \mathbf{Y}$, the r.v. $\mathbf{W}$ has a **posterior**: $p\big(\mathbf{W}|\mathbf{X}, \mathbf{Y}\big)$
- Which is difficult to evaluate—many great researchers tried
- Can define **simple distribution** $q_{\mathbf{M}}(\cdot)$ and approximate

$$q_{\mathbf{M}}(\mathbf{W}) \approx p\big(\mathbf{W}|\mathbf{X}, \mathbf{Y}\big)$$

- This is called **approximate variational inference.**

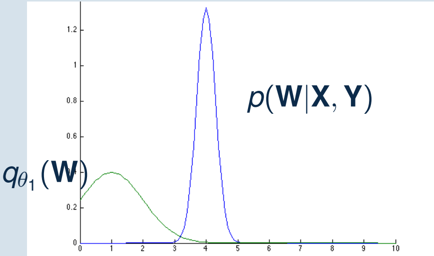
From Bayesian neural networks to Dropout

- Place **prior** $p(\mathbf{W})$ dist. on weights, making these r.v.s
- Given dataset $\mathbf{X}$, $\mathbf{Y}$, the r.v. $\mathbf{W}$ has a **posterior**: $p\big(\mathbf{W}|\mathbf{X}, \mathbf{Y}\big)$
- Which is difficult to evaluate—many great researchers tried
- Can define **simple distribution** $q_{\mathbf{M}}(\cdot)$ and approximate
$$q_{\mathbf{M}}(\mathbf{W}) \approx p\big(\mathbf{W}|\mathbf{X}, \mathbf{Y}\big)$$
- This is called **approximate variational inference.**

From Bayesian neural networks to Dropout

- Place **prior** $p(\mathbf{W})$ dist. on weights, making these r.v.s
- Given dataset $\mathbf{X}, \mathbf{Y}$, the r.v. $\mathbf{W}$ has a **posterior**: $p\big(\mathbf{W}|\mathbf{X}, \mathbf{Y}\big)$
- Which is difficult to evaluate—many great researchers tried
- Can define **simple distribution** $q_{\mathbf{M}}(\cdot)$ and approximate

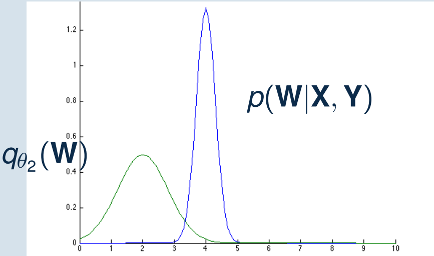
$$q_{\mathbf{M}}(\mathbf{W}) \approx p\big(\mathbf{W}|\mathbf{X}, \mathbf{Y}\big)$$



From Bayesian neural networks to Dropout

- Place **prior** $p(\mathbf{W})$ dist. on weights, making these r.v.s
- Given dataset $\mathbf{X}, \mathbf{Y}$, the r.v. $\mathbf{W}$ has a **posterior**: $p\big(\mathbf{W}|\mathbf{X}, \mathbf{Y}\big)$
- Which is difficult to evaluate—many great researchers tried
- Can define **simple distribution** $q_{\mathbf{M}}(\cdot)$ and approximate

$$q_{\mathbf{M}}(\mathbf{W}) \approx p\big(\mathbf{W}|\mathbf{X}, \mathbf{Y}\big)$$

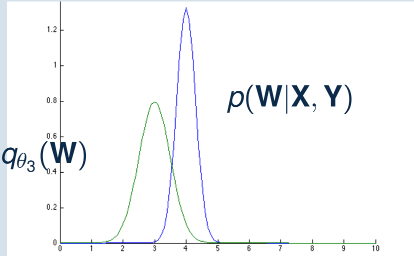


UNIVERSITY OF OXFORD

From Bayesian neural networks to Dropout

- Place **prior** $p(\mathbf{W})$ dist. on weights, making these r.v.s
- Given dataset $\mathbf{X}, \mathbf{Y}$, the r.v. $\mathbf{W}$ has a **posterior**: $p\big(\mathbf{W}|\mathbf{X}, \mathbf{Y}\big)$
- Which is difficult to evaluate—many great researchers tried
- Can define **simple distribution** $q_{\mathbf{M}}(\cdot)$ and approximate

$$q_{\mathbf{M}}(\mathbf{W}) \approx p\big(\mathbf{W}|\mathbf{X}, \mathbf{Y}\big)$$

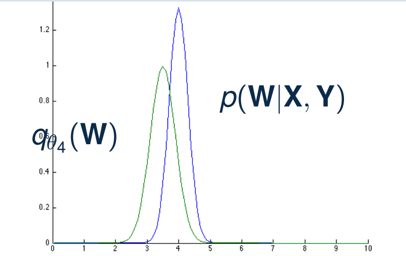


UNIVERSITY OF OXFORD

From Bayesian neural networks to Dropout

- Place **prior** $p(\mathbf{W})$ dist. on weights, making these r.v.s
- Given dataset $\mathbf{X}, \mathbf{Y}$, the r.v. $\mathbf{W}$ has a **posterior**: $p\big(\mathbf{W}|\mathbf{X}, \mathbf{Y}\big)$
- Which is difficult to evaluate—many great researchers tried
- Can define **simple distribution** $q_{\mathbf{M}}(\cdot)$ and approximate

$$q_{\mathbf{M}}(\mathbf{W}) \approx p\big(\mathbf{W}|\mathbf{X}, \mathbf{Y}\big)$$

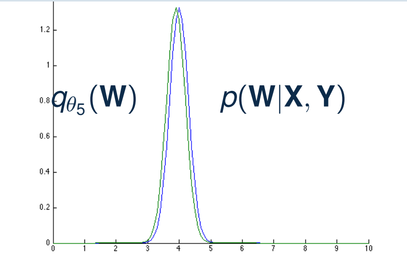


Some theory

From Bayesian neural networks to Dropout

- Place **prior** $p(\mathbf{W})$ dist. on weights, making these r.v.s
- Given dataset $\mathbf{X}, \mathbf{Y}$, the r.v. $\mathbf{W}$ has a **posterior**: $p\big(\mathbf{W}|\mathbf{X}, \mathbf{Y}\big)$
- Which is difficult to evaluate—many great researchers tried
- Can define **simple distribution** $q_{\mathbf{M}}(\cdot)$ and approximate

$$q_{\mathbf{M}}(\mathbf{W}) \approx p\big(\mathbf{W}|\mathbf{X}, \mathbf{Y}\big)$$



From Bayesian neural networks to Dropout

- Place **prior** $p(\mathbf{W})$ dist. on weights, making these r.v.s
- Given dataset $\mathbf{X}, \mathbf{Y}$, the r.v. $\mathbf{W}$ has a **posterior**: $p\big(\mathbf{W}|\mathbf{X}, \mathbf{Y}\big)$
- Which is difficult to evaluate—many great researchers tried
- Can define **simple distribution** $q_{\mathbf{M}}(\cdot)$ and approximate
$$q_{\mathbf{M}}(\mathbf{W}) \approx p\big(\mathbf{W}|\mathbf{X}, \mathbf{Y}\big)$$
- This is called **approximate variational inference.**

Some theory

Theorem (Dropout as approximate variational inference)

Define

$$q_{\mathbf{M}}(\mathbf{W}) := \mathbf{M} \cdot diag(Bernoulli)$$

with variational parameter $\mathbf{M}$*.*
The optimisation objective of (stochastic) variational inference with $q_{\mathbf{M}}(\mathbf{W})$ *is identical to the objective of a dropout neural network.*

Proof.

See Gal [2016]. □

UNIVERSITY OF OXFORD

Theorem (Dropout as approximate variational inference)

Define

$$q_{\mathbf{M}}(\mathbf{W}) := \mathbf{M} \cdot diag(Bernoulli)$$

with variational parameter $\mathbf{M}$*.*
The optimisation objective of (stochastic) variational inference with $q_{\mathbf{M}}(\mathbf{W})$ *is identical to the objective of a dropout neural network.*

Proof.

See Gal [2016]. □

Implementing **inference** with $q_{\mathbf{M}}(\mathbf{W})$
=
Implementing **dropout training**.
Line to line.

Theorem (Dropout as approximate variational inference)

Define

$$q_{\mathbf{M}}(\mathbf{W}) := \mathbf{M} \cdot diag(Bernoulli)$$

with variational parameter $\mathbf{M}$.
The optimisation objective of (stochastic) variational inference with $q_{\mathbf{M}}(\mathbf{W})$ *is identical to the objective of a dropout neural network.*

Corollary (Model uncertainty with dropout)

Given $p(\mathbf{y}^*|\mathbf{f}^{\mathbf{W}}(\mathbf{x}^*)) = \mathcal{N}(\mathbf{y}^*; \mathbf{f}^{\mathbf{W}}(\mathbf{x}^*), \tau^{-1}\mathbf{I})$ *for some* $\tau > 0$, *the model's predictive variance can be estimated with the unbiased estimator:*

$$\widetilde{Var}[\mathbf{y}^*] := \tau^{-1}\mathbf{I} + \frac{1}{T}\sum_{t=1}^{T} \mathbf{f}^{\widehat{\mathbf{W}}_t}(\mathbf{x}^*)^T \mathbf{f}^{\widehat{\mathbf{W}}_t}(\mathbf{x}^*) - \widetilde{\mathbb{E}}[\mathbf{y}^*]^T \widetilde{\mathbb{E}}[\mathbf{y}^*]$$

with $\widehat{\mathbf{W}}_t \sim q^*_{\mathbf{M}}(\mathbf{W})$.

Some code, just for fun

In practical terms[1], given point *x*:

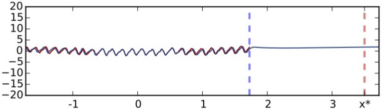
- drop units **at test time**
- **repeat 10 times**
- and look at **mean and sample variance.**
- Or in Python:

```
y = []
for _ in xrange(10):
    y.append(model.output(x, dropout=True))
y_mean = numpy.mean(y)
y_var = numpy.var(y)
```

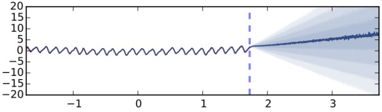
[1] Friendly introduction given in `yarin.co/blog`

What would be the CO_2 concentration level in Mauna Loa, Hawaii, *in 20 years' time*?

Normal deep learning:



Bayesian perspective:



What can we do with this?

Deep learning with small data • Interpretable AI • Safe AI

Human-in-the-loop AI for Galaxy Zoo morphology classification

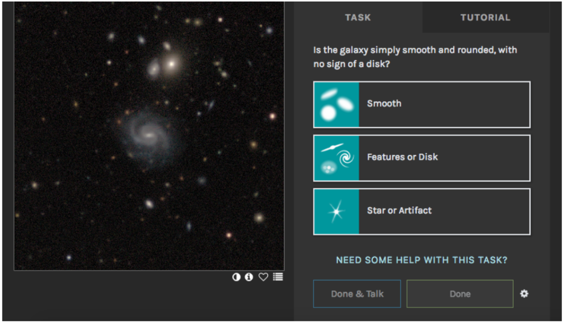
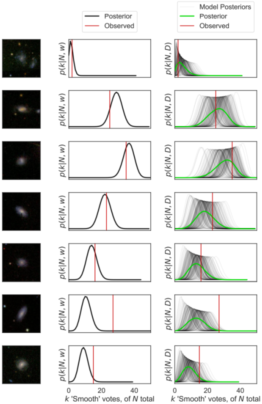
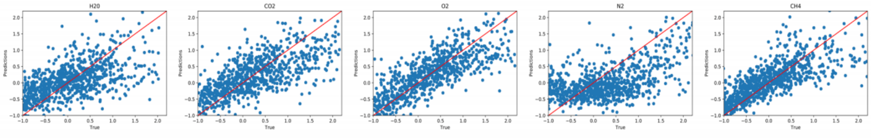


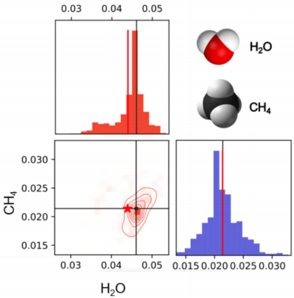
Figure 1. The Galaxy Zoo web interface as shown to volunteers. This screenshot shows the first question in the decision tree: is the galaxy smooth or featured?

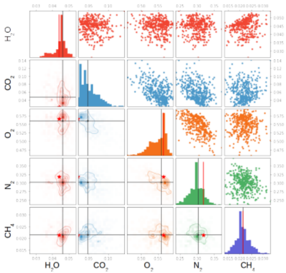


with Lewis Smith [work done w. Chris Lintott, Zooniverse Citizen Science Project]

Bayesian deep learning for exoplanet atmospheric retrieval



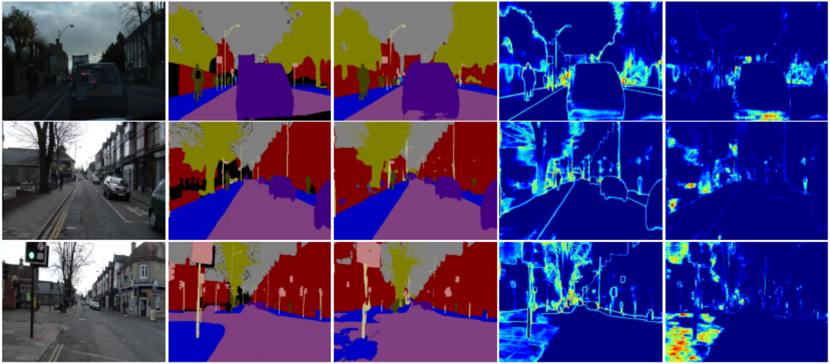




with Adam Cobb [work done with NASA Goddard while at NASA FDL]

Uncertainties in computer vision

- *Aleatoric uncertainty*, capturing inherent noise in the data
- *Epistemic uncertainty*, capturing model's lack of knowledge

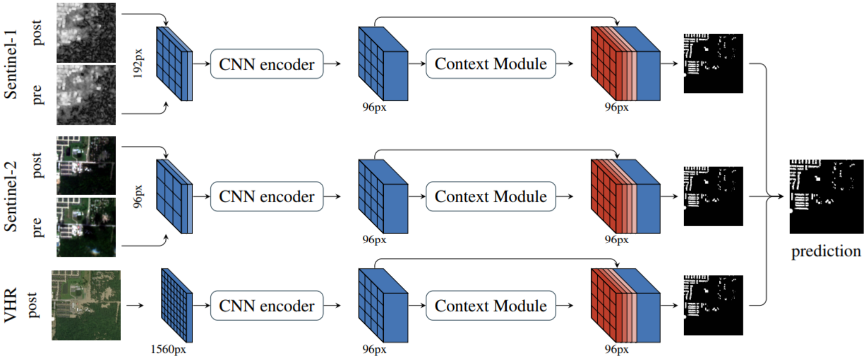


(a) Input Image (b) Ground Truth (c) Semantic Segmentation (d) Aleatoric Uncertainty (e) Epistemic Uncertainty

with Alex Kendall

UNIVERSITY OF OXFORD

Informal settlement detection

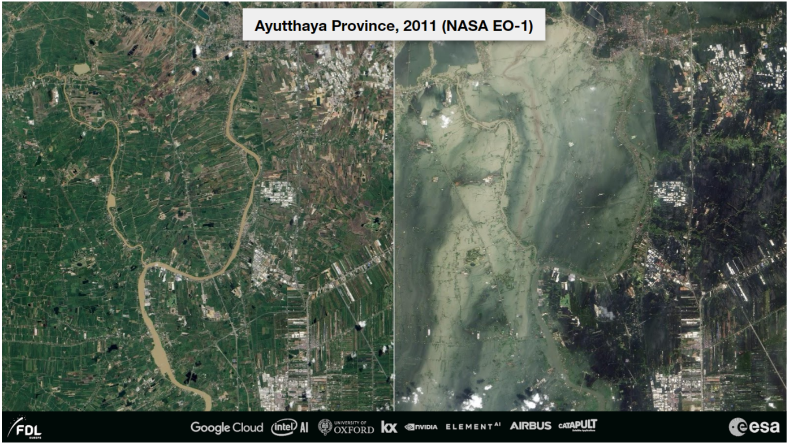


with Tim Rudner [work done with ESA while at FDL Europe]

Flood detection, from space

with Lewis Smith [work done with ESA while at FDL Europe]

Flood detection, from space



with Lewis Smith [work done with ESA while at FDL Europe]

Flood detection, from space

FRONTIER DEVELOPMENT LAB EUROPE 2019

How can we reduce the time from disaster to data?

American Red Cross

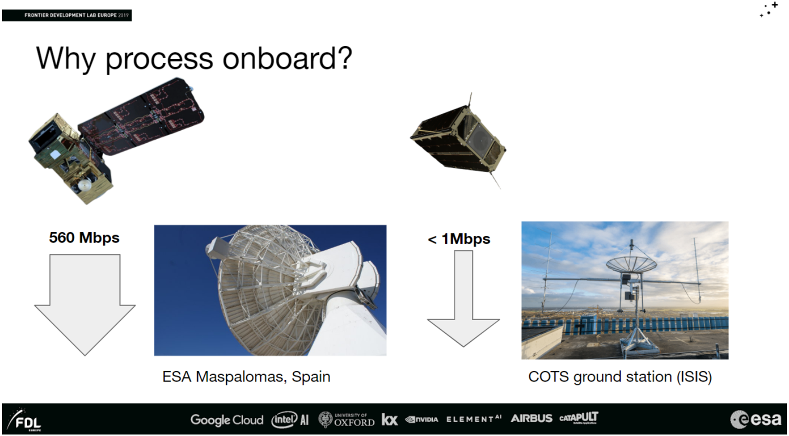
1. How do we get data to the ground more quickly?
2. How can we accelerate/automate the image analysis process?

FDL Google Cloud intel AI UNIVERSITY OF OXFORD kx NVIDIA ELEMENT AI AIRBUS CATAPULT esa

with Lewis Smith [work done with ESA while at FDL Europe]

ML in Space (literally)

Flood detection, from space



with Lewis Smith [work done with ESA while at FDL Europe]

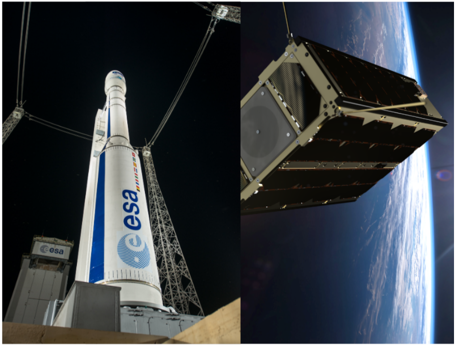
ML in Space (literally)

Flood detection, from space

FRONTIER DEVELOPMENT LAB EUROPE 2019

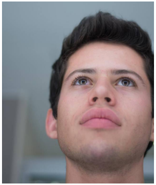
Project Proposal

- Demonstrate that low resolution images from CubeSats can be used to provide useful flood intelligence
- Perform flood segmentation onboard satellite to reduce downlinked data
- Deploy on ΦSat-1 in 2019/2020



with Lewis Smith [work done with ESA while at FDL Europe]

OATML



Oxford Applied and Theoretical Machine Learning Group

http://oatml.ox.ac.uk

Researchers coming from academia (Oxford, Cambridge, MILA, Yale, U of Toronto, U of Amsterdam, etc)

.. and industry (Google, DeepMind, Twitter, etc)