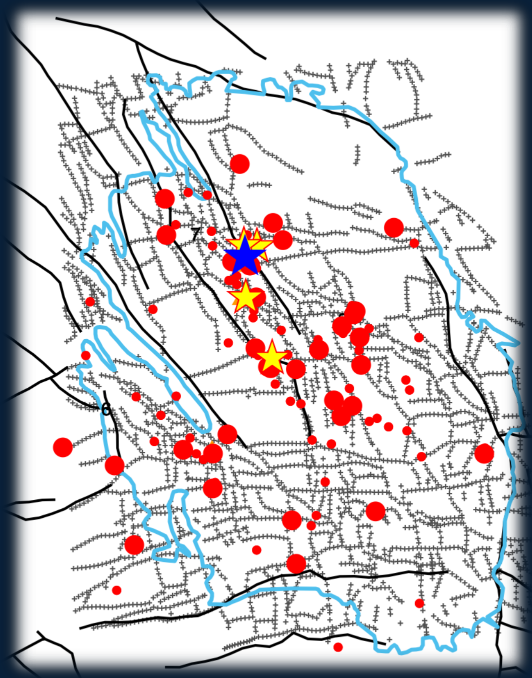
MIT EARTH RESOURCES LABORATORY
ANNUAL FOUNDING MEMBERS MEETING 2019

MIT | Earth Resources Laboratory

Deep learning applied to induced seismicity in the Groningen gas field in the Netherlands – Why safe AI and what do we need?

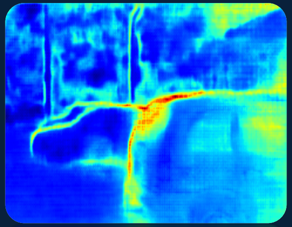


Chen Gu

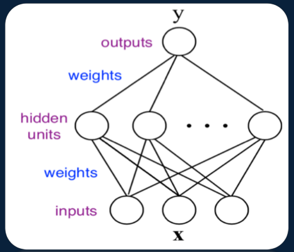
POSTDOCTORAL ASSOCIATE [EARTH, ATMOSPHERIC AND PLANETARY SCIENCES]

In collaboration with Youssef M. Marzouk and M. Nafi Toksöz

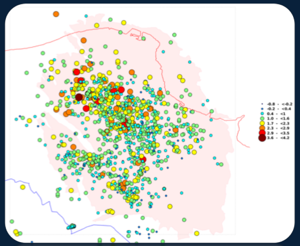
Outline



- Research motivation – Why safe AI and what do we need



- Method – Uncertainty in deep learning and Bayesian neural networks



- Examples – Induced seismicity in the Groningen gas field in the Netherlands

Why safe AI

- Many AI methods do not consider uncertainty quantification (UQ) from weights of neural net, choice of architecture, choice of hidden layers, etc, which may result in serious problems (e.g., autopilot car).

Raw photo

Estimated depth

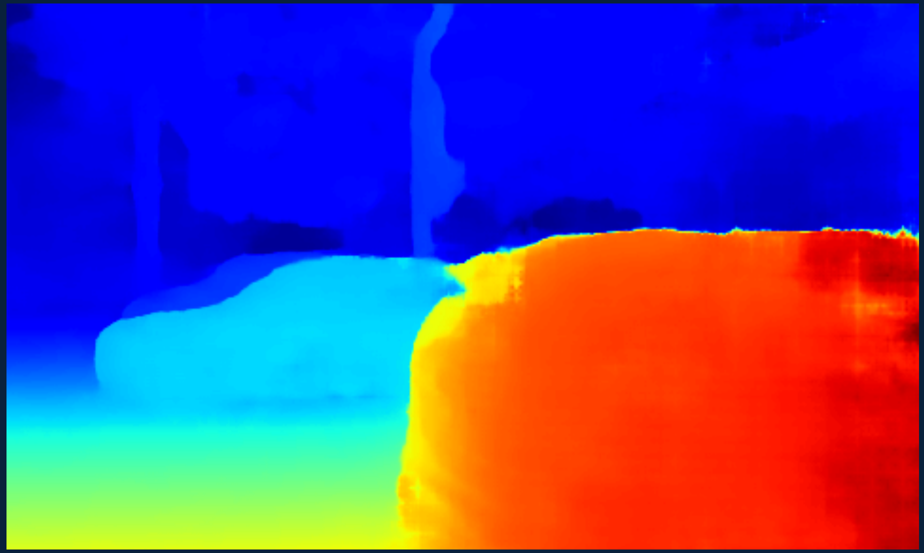


Figure source: https://alexgkendall.com/computer_vision/bayesian_deep_learning_for_safe_ai/

Why safe AI

- Bayesian neural network provides a solution to understand uncertainties of deep learning system to make AI safe.

Estimated depth uncertainty

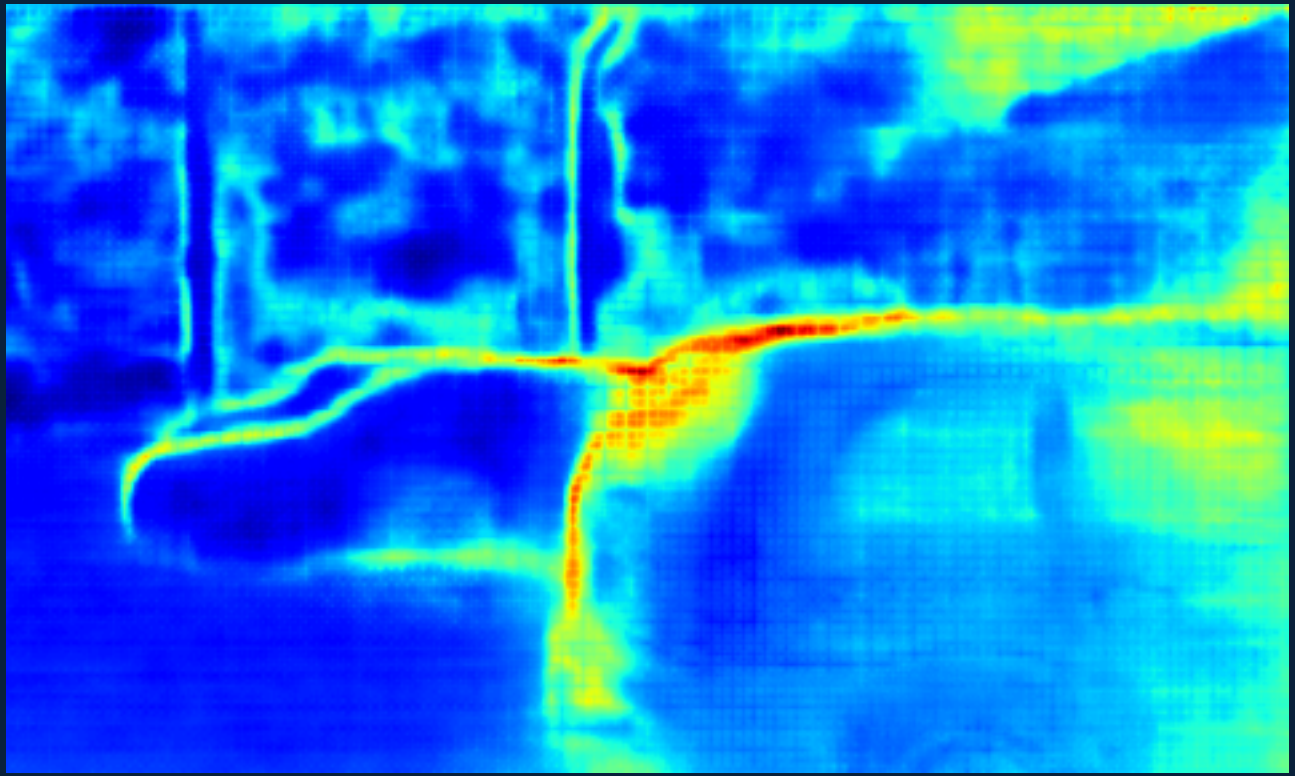
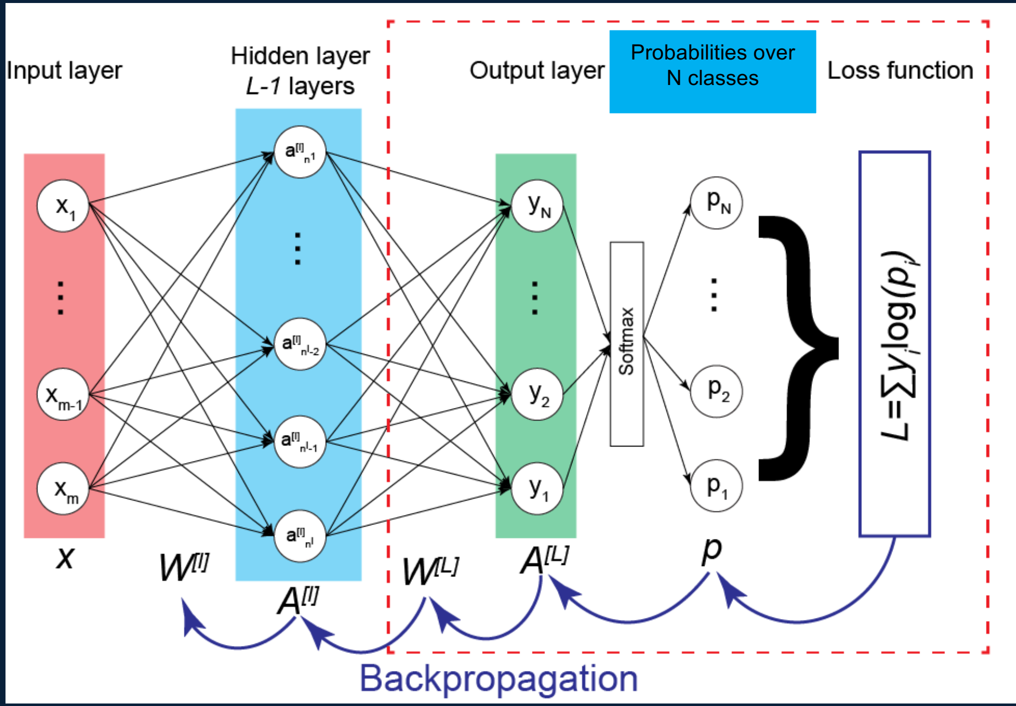


Figure source: https://alexgkendall.com/computer_vision/bayesian_deep_learning_for_safe_ai/

Erroneous uncertainty interpretation of Softmax



- Softmax function:

$$\sigma(\mathbf{z})_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}}, \qquad j = 1, \ldots, K$$

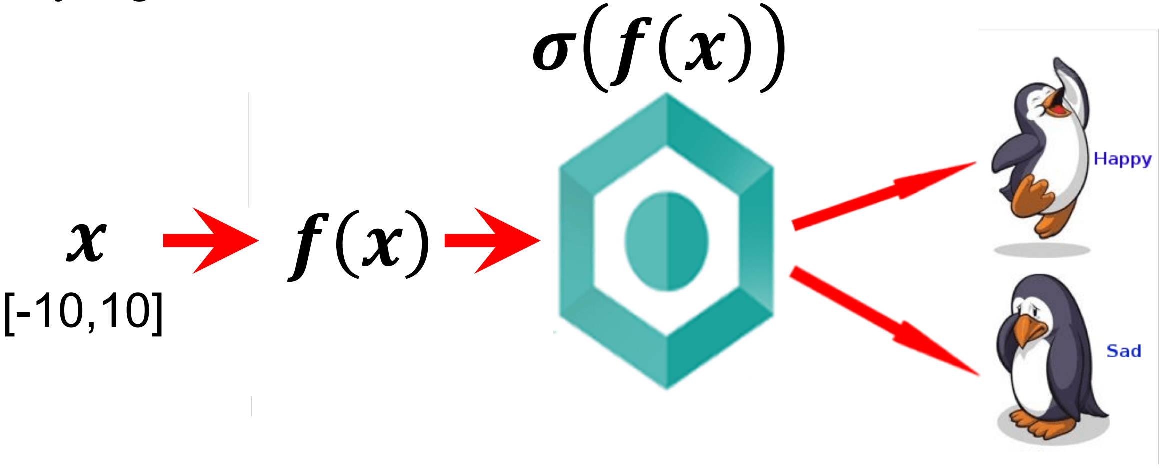
- Probability of **x** belong to **j**th class:

$$P(y = j|\boldsymbol{x}) = \frac{e^{\boldsymbol{x}^T w_j}}{\sum_{k=1}^{K} e^{\boldsymbol{x}^T w_k}}$$

But not model uncertainty!!!

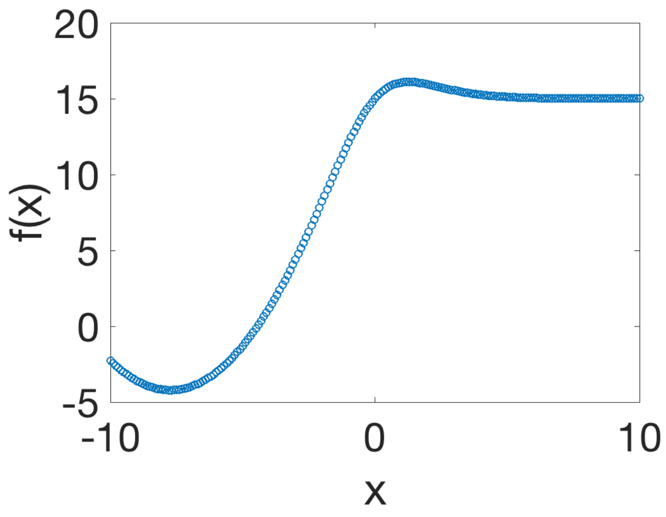
Erroneous uncertainty interpretation of Softmax

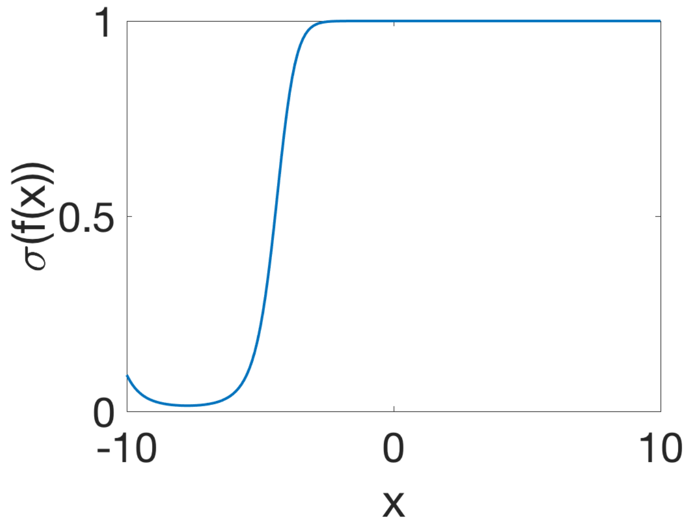
- Binary logistic classification



$$P(y = Happy|\boldsymbol{x}) = \frac{e^{f(\boldsymbol{x})}}{1 + e^{f(\boldsymbol{x})}}$$

$$P(y = Sad|\boldsymbol{x}) = \frac{1}{1 + e^{f(\boldsymbol{x})}}$$

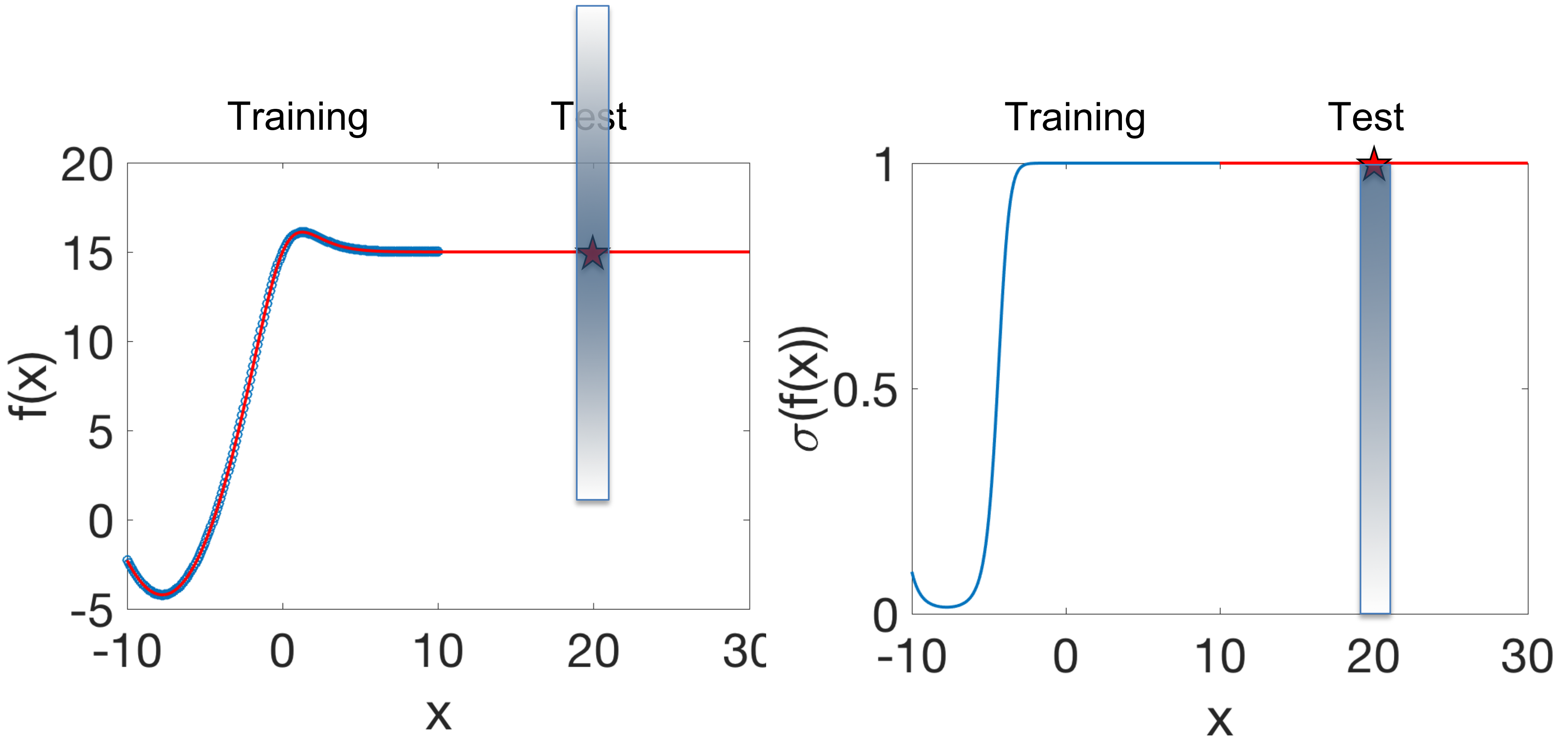




Modified from: https://towardsdatascience.com/logistic-regression-detailed-overview-46c4da4303bc

Erroneous uncertainty interpretation of Softmax

Q: Probability of **x* = 20** belong to ***Happy*** class



Modified from Gal and Ghahramani, 2016

Uncertainty in deep learning

MIT | Earth Resources Laboratory

Data distribution
- Training
- Test

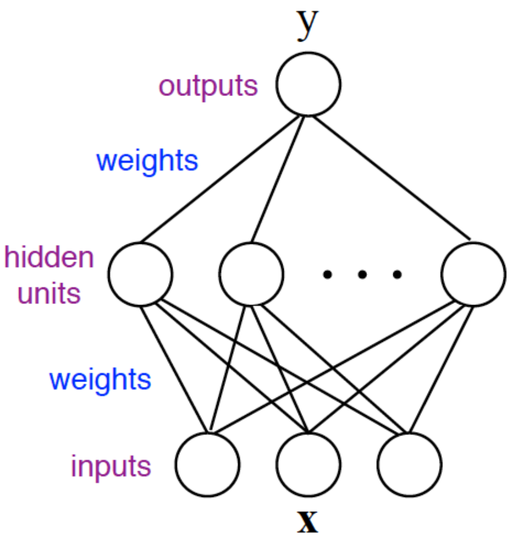
Data noise
- Measurement
- Label

Model parameters
- Weights
- Kernels (CNN)
- Gates (RNN)

Model structures
- Architectures
- Hidden layers
- Activation functions

Bayesian neural networks

- Dealing with all sources of parameter uncertainty
- Also potentially dealing with structure uncertainty



Bayesian neural network

Data: $\mathcal{D} = \{(\mathbf{x}^{(n)}, y^{(n)})\}_{n=1}^{N} = (X, \mathbf{y})$

Parameters $\boldsymbol{\theta}$ are weights of neural net

prior $\quad p(\boldsymbol{\theta}|\boldsymbol{\alpha})$

posterior $\quad p(\boldsymbol{\theta}|\boldsymbol{\alpha}, \mathcal{D}) \propto p(\mathbf{y}|X, \boldsymbol{\theta}) p(\boldsymbol{\theta}|\boldsymbol{\alpha})$

prediction $\quad p(y'|\mathcal{D}, \mathbf{x}', \boldsymbol{\alpha}) = \int p(y'|\mathbf{x}', \boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathcal{D}, \boldsymbol{\alpha})\, d\boldsymbol{\theta}$

Ghahramani, 2015; Ghahramani, 2016 NIPS

Bayesian inference and approximation

- **Formula of Bayesian inference for neural networks:**

$$p(\boldsymbol{y}^*|\boldsymbol{x}^*,\boldsymbol{X},\boldsymbol{Y}) = \int p(\boldsymbol{y}^*|\boldsymbol{x}^*,\boldsymbol{\omega})p(\boldsymbol{\omega}|\boldsymbol{X},\boldsymbol{Y})d\boldsymbol{\omega}$$

Training data: $\{\boldsymbol{X},\boldsymbol{Y}\}$
Testing data: $\{\boldsymbol{x}^*,\boldsymbol{y}^*\}$
Random variables: $\{\boldsymbol{\omega}\}$

- **$\{\boldsymbol{\omega}\}$ in different deep neural networks can be:**

Standard neural networks: $\{\boldsymbol{W}^l,\boldsymbol{b}^l\}$
Convolutional neural networks: $\{\boldsymbol{K}^l\}$
Recurrent neural networks: $\{\boldsymbol{W}_h,\boldsymbol{U}_h,\boldsymbol{b}_h,\boldsymbol{W}_y,\boldsymbol{b}_y\}$

- **Variational inference (VI):**

$$\mathrm{KL}(q_\theta(\boldsymbol{\omega})\|p(\boldsymbol{\omega}|\boldsymbol{X},\boldsymbol{Y})) = \int q_\theta(\boldsymbol{\omega})\log\frac{q_\theta(\boldsymbol{\omega})}{p(\boldsymbol{\omega}|\boldsymbol{X},\boldsymbol{Y})}d\boldsymbol{\omega}$$

$$p(\boldsymbol{y}^*|\boldsymbol{x}^*,\boldsymbol{X},\boldsymbol{Y}) \approx \int p(\boldsymbol{y}^*|\boldsymbol{x}^*,\boldsymbol{\omega})q_\theta^*(\boldsymbol{\omega})d\boldsymbol{\omega} =: q_\theta^*(\boldsymbol{y}^*|\boldsymbol{x}^*)$$

Hinton and Van Camp, 1993

What would be a practical tool?

Scalable

- Large data
- Complex models

Simple

- Well-studied models
- Easy implementation

Stochastic regularization techniques (SRT)

- Probability theory and Bayesian modeling
- SRT: Dropout, multiplicative Gaussian noise, dropConnect

Stochastically inject noise to model during training: $\{\omega^i\} \sim q_\theta(\omega)$

Repeat

1. Sample random variables $\boldsymbol{\omega}^{\boldsymbol{i}} \sim \boldsymbol{q}_{\boldsymbol{\theta}}(\boldsymbol{\omega})$
2. Randomly choose a minibatch ***S*** of size ***M***
3. Calculate derivatives relative to $\boldsymbol{\theta}$:

$$\boldsymbol{\Delta\theta} \leftarrow -\frac{1}{M\tau} \sum_{i \in S} \frac{\partial}{\partial \boldsymbol{\theta}} \log p\left(\boldsymbol{y}_i \middle| f^{\boldsymbol{\omega}^i}(\boldsymbol{x})\right) + \frac{\partial}{\partial \boldsymbol{\theta}} \sum_d \boldsymbol{\lambda}_d \|\boldsymbol{\theta}_d\|^2$$

4. Update $\boldsymbol{\theta}$:

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \eta \boldsymbol{\Delta\theta}$$

until $\boldsymbol{\theta}$ converged.

Stochastic forward pass

Hinton, et al., 2012; Srivastava et al., 2014; Wan et al., 2013; Gal and Ghahramani, 2016;Gal, thesis, 2016

Stochastic regularization techniques (SRT)

- *T* realizations of model parameters according to posterior model distribution (stochastic forward pass):

$$\{\boldsymbol{\omega}^i\}_{i=1\ldots T} \sim q_\theta(\boldsymbol{\omega})$$

- Then, we obtain mean and uncertainty:

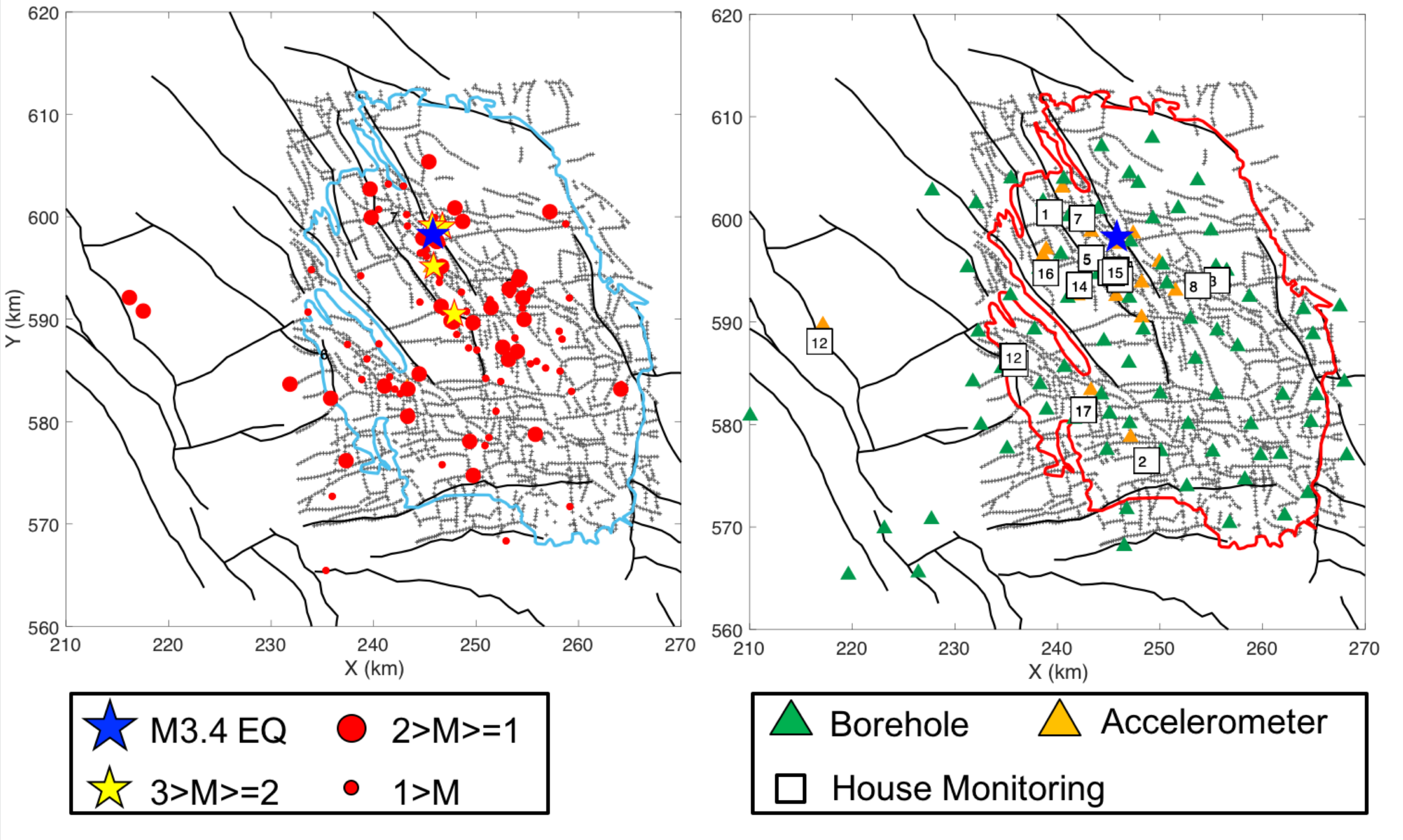
$$\mathbb{E}[\boldsymbol{y}^*] \approx \frac{1}{T}\sum_{t=1}^{T} \hat{\boldsymbol{y}}_t^*(\boldsymbol{x}^*)$$

$$\mathrm{Var}[\boldsymbol{y}^*] \approx \tau^{-1}\boldsymbol{I}_D + \frac{1}{T}\sum_{t=1}^{T} \hat{\boldsymbol{y}}_t^*(\boldsymbol{x}^*)^T \hat{\boldsymbol{y}}_t^*(\boldsymbol{x}^*) - \mathbb{E}[\boldsymbol{y}^*]^T \mathbb{E}[\boldsymbol{y}^*]$$

Practical with **large models** and **big data**
Applicable to **image based models**, **sequence based models**, **reinforcement learning** and **active learning**

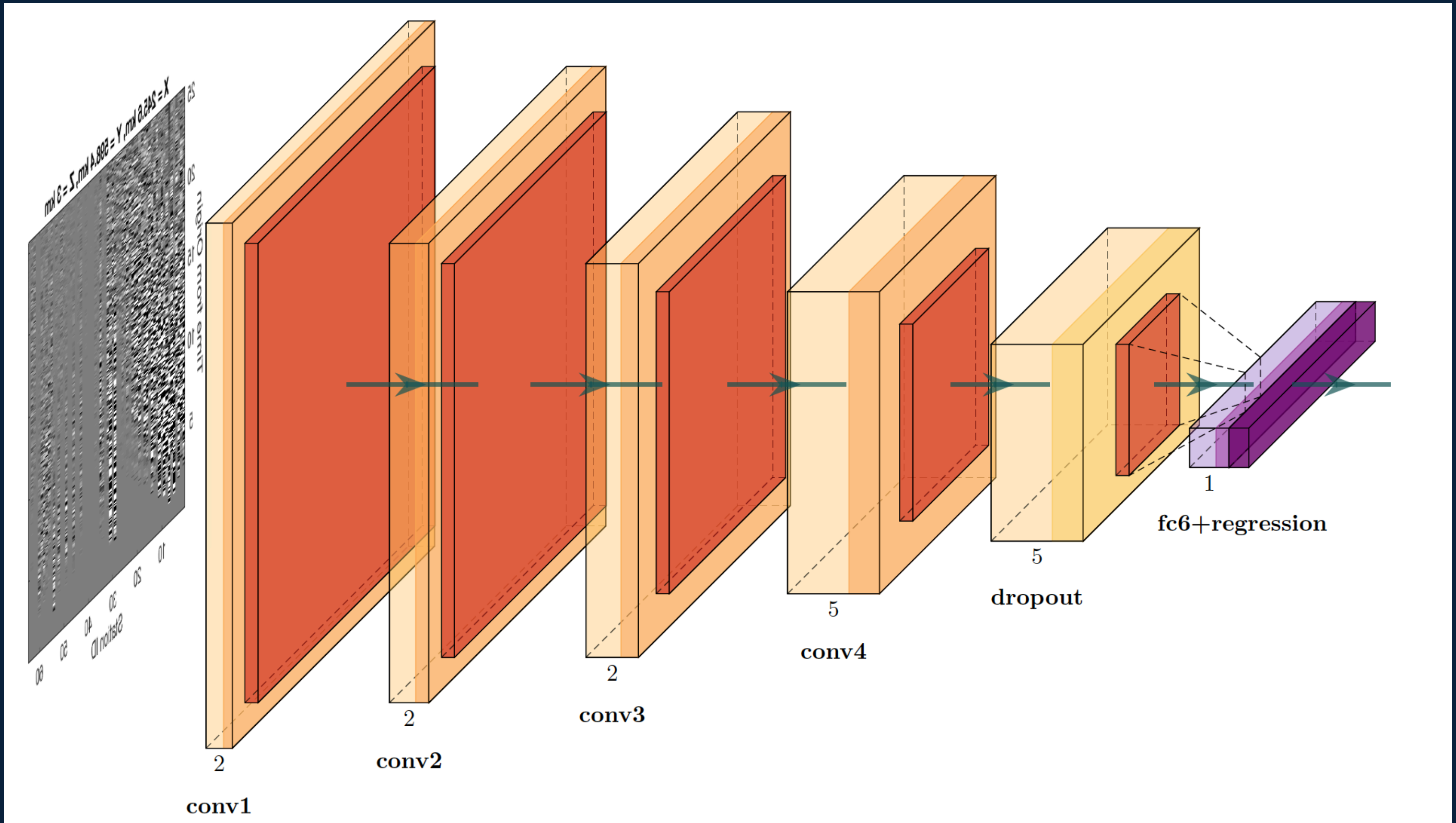
Hinton, et al., 2012; Srivastava et al., 2014; Wan et al., 2013; Gal, thesis, 2016

Induced Seismicity 09/13/2017-9/30/2018



Bayesian convolutional neural networks (BCNNs)

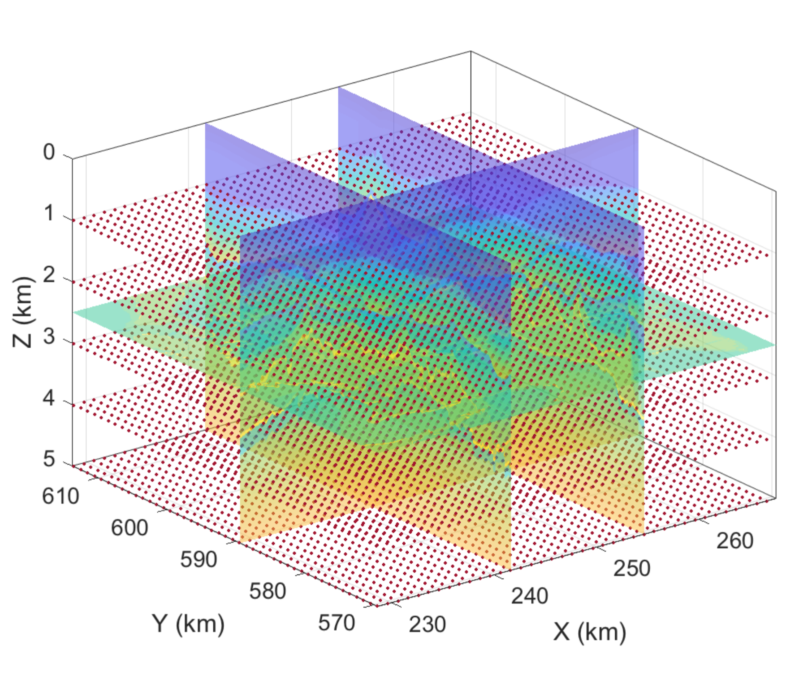
- 4 convolutional layers + 1 MC dropout layer + 1 fully connected layer + regession
- seismic source gathers from 64 observation stations as input

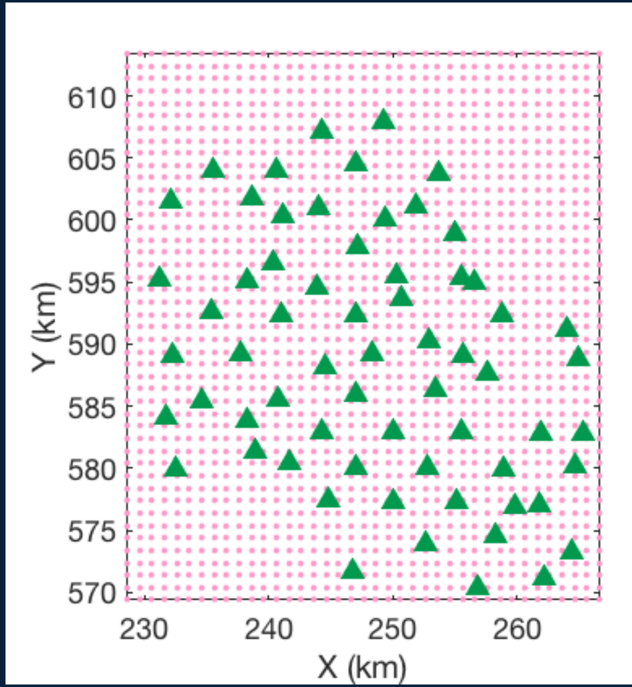


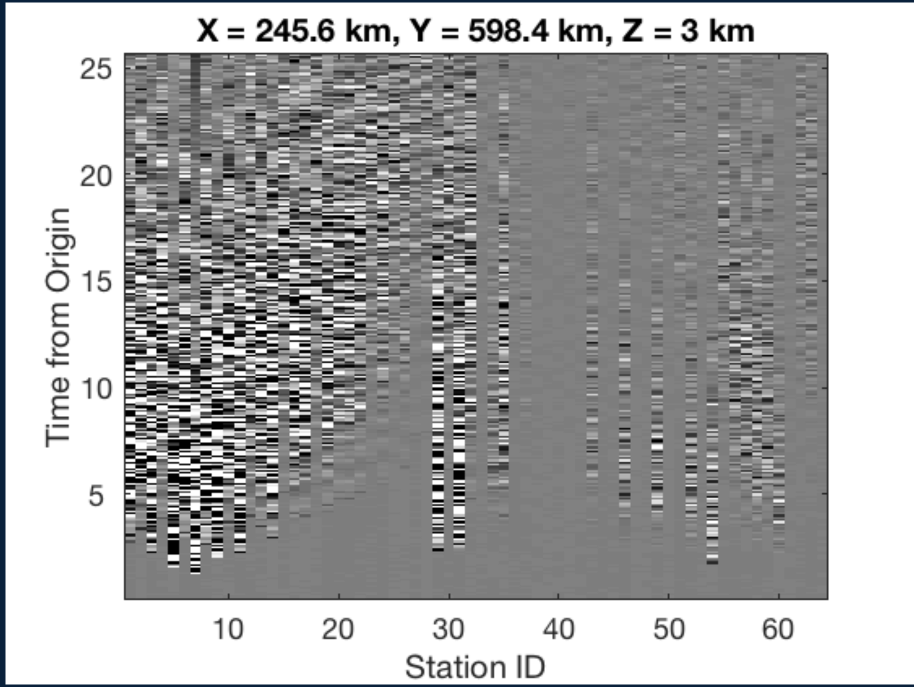
Bayesian convolutional neural networks (BCNNs)

MIT | Earth Resources Laboratory

- 39*45*5=8775 synthetic events at trial earthquake locations
- 64 observation stations

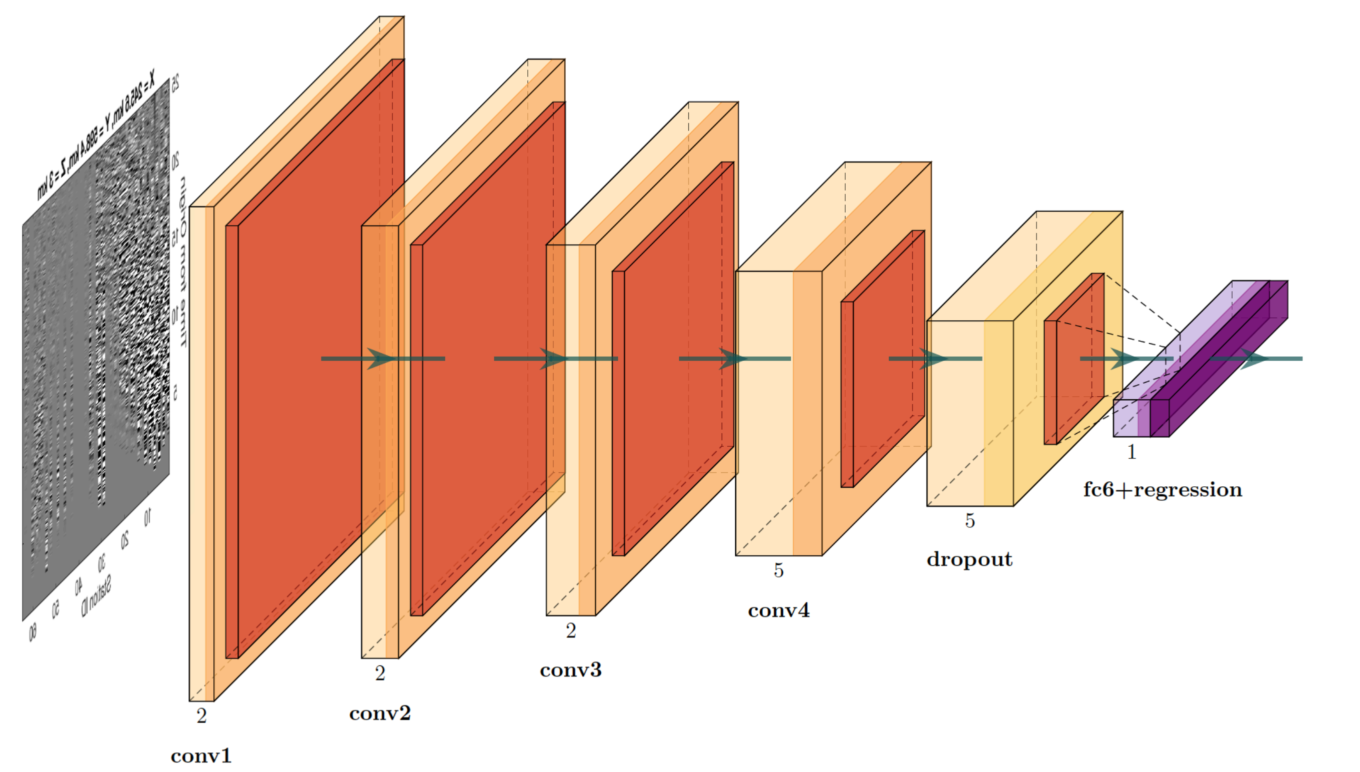


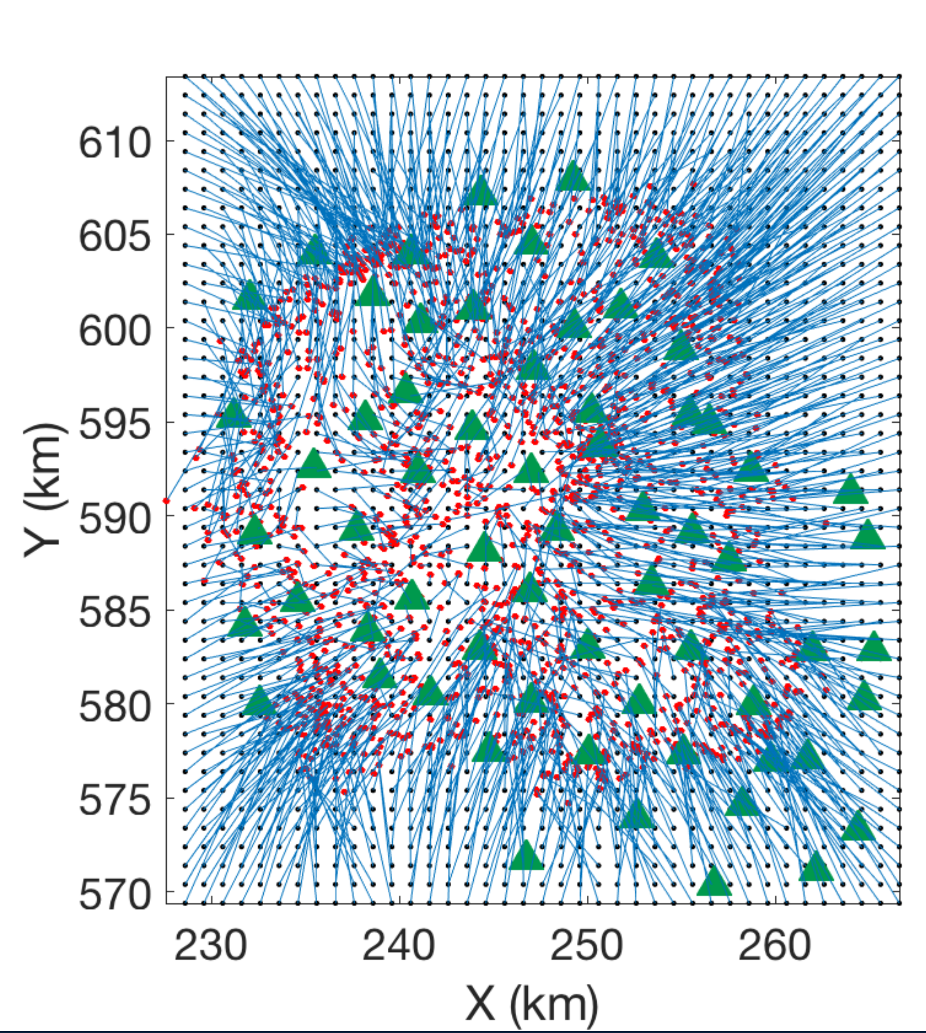




Noise perturbed waveform data (10% Gaussian)

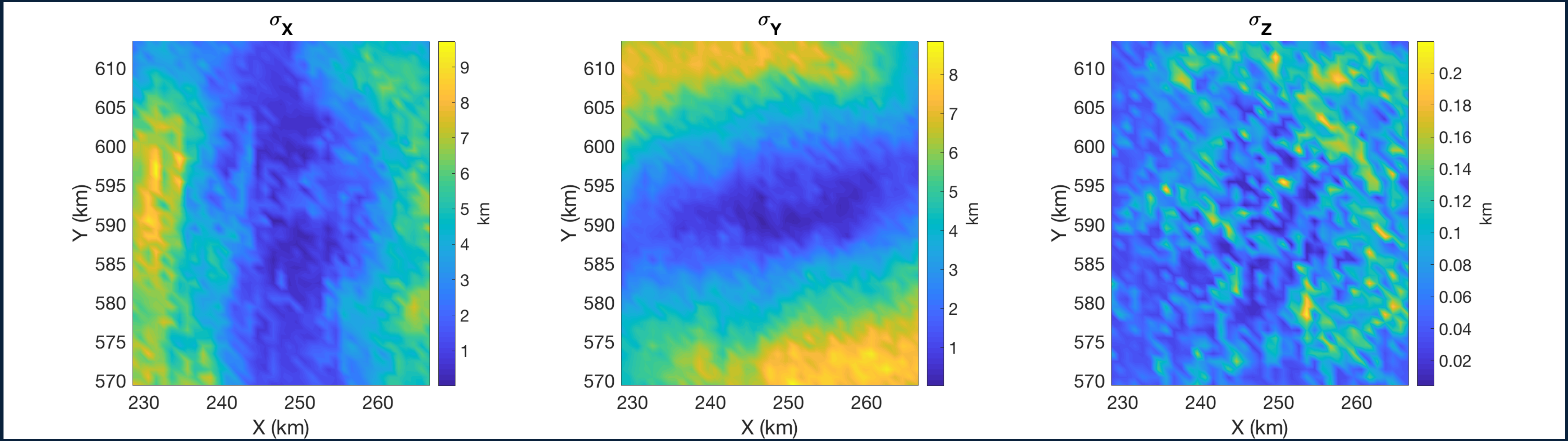
- Testing set: noise perturbed source gathers for 1775 events
- 10^4 stochastic forward pass





Location uncertainties

- Standard deviation from 10^4 stochastic forward passes



Conclusion

- The uncertainty quantification (UQ) of parameters and structures of deep neural networks is important to make AI safer. We replace deterministic neural networks with Bayesian neural networks to quantify the uncertainty of deep learning system.

- The stochastic regularization techniques are practical tools to implement Bayesian deep learning, and are scalable to complex neural nets and deep learning.

- This work uses deep learning, Bayesian neural networks, to locate earthquakes using a complex 3-D velocity model in the Groningen field. The deep neural network is trained using synthetic data and will apply to real seismic and building data.

Earth Resources Laboratory

Thank you!

Email: guchch@mit.edu

Conclusion

- The uncertainty quantification (UQ) of parameters and structures of deep neural networks is important to make AI safer. We replace deterministic neural networks with Bayesian neural networks to quantify the uncertainty of deep learning system.

- The stochastic regularization techniques are practical tools to implement Bayesian deep learning, and are scalable to complex neural nets and deep learning.

- This work uses deep learning, Bayesian neural networks, to locate earthquakes using a complex 3-D velocity model in the Groningen field. The deep neural network is trained using synthetic data and will apply to real seismic and building data.