Fermilab U.S. DEPARTMENT OF ENERGY Office of Science

(Quantum) Machine Learning for Astrophysics

João Caldeira

SCD Postdoc Meeting

7 April 2020

Overview

- From before: DeepCMB (1810.01483 + 20xx.xxxxx)

- Using quantum computing for machine learning (1911.06259 + 20xx.xxxxx)

- Uncertainty quantification for deep learning (2004.xxxxx)

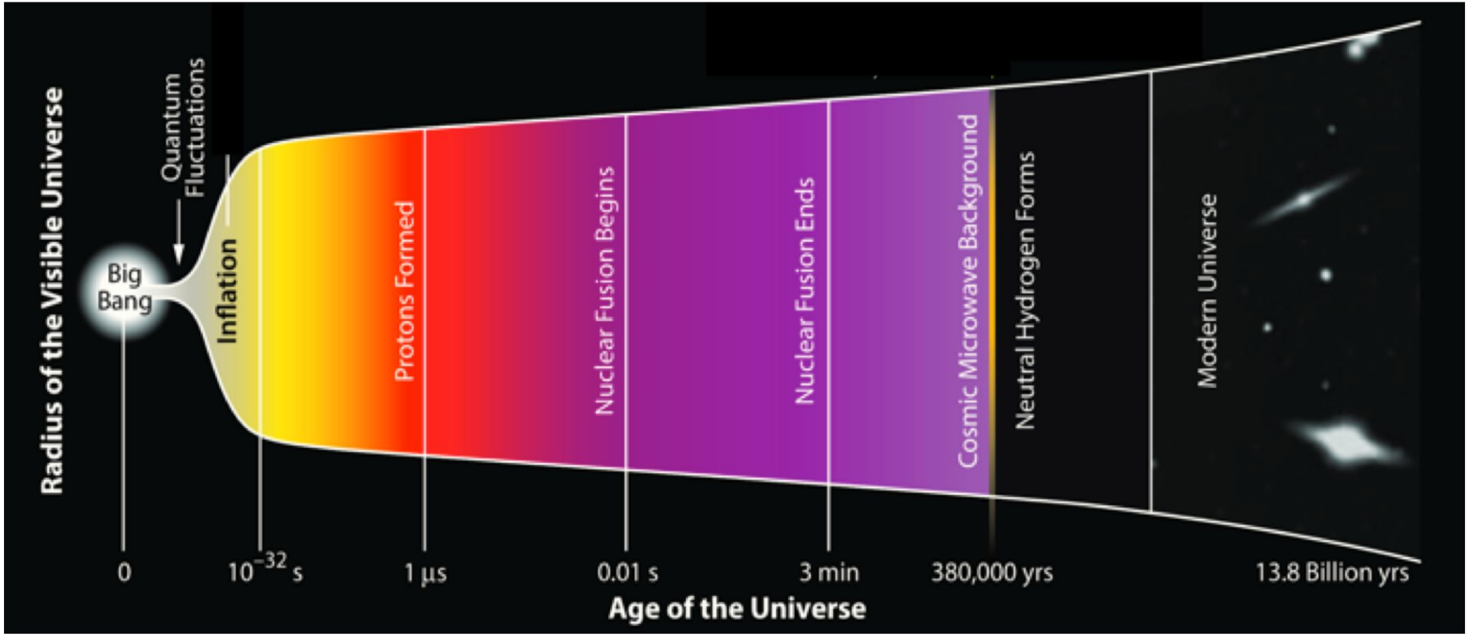
Overview

- **From before: DeepCMB (1810.01483 + 20xx.xxxxx)**

- Using quantum computing for machine learning (1911.06259 + 20xx.xxxxx)

- Uncertainty quantification for deep learning (2004.xxxxx)

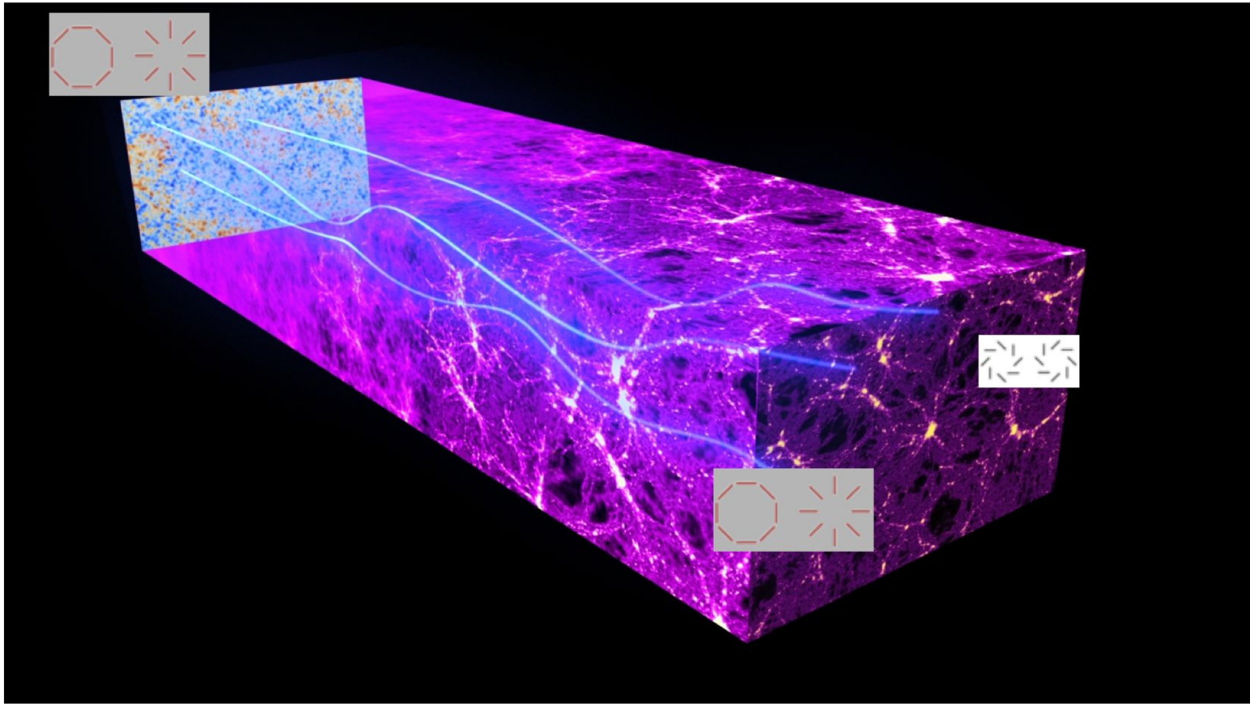
The CMB, and lensing of the CMB



Credit: BICEP/Keck collaboration

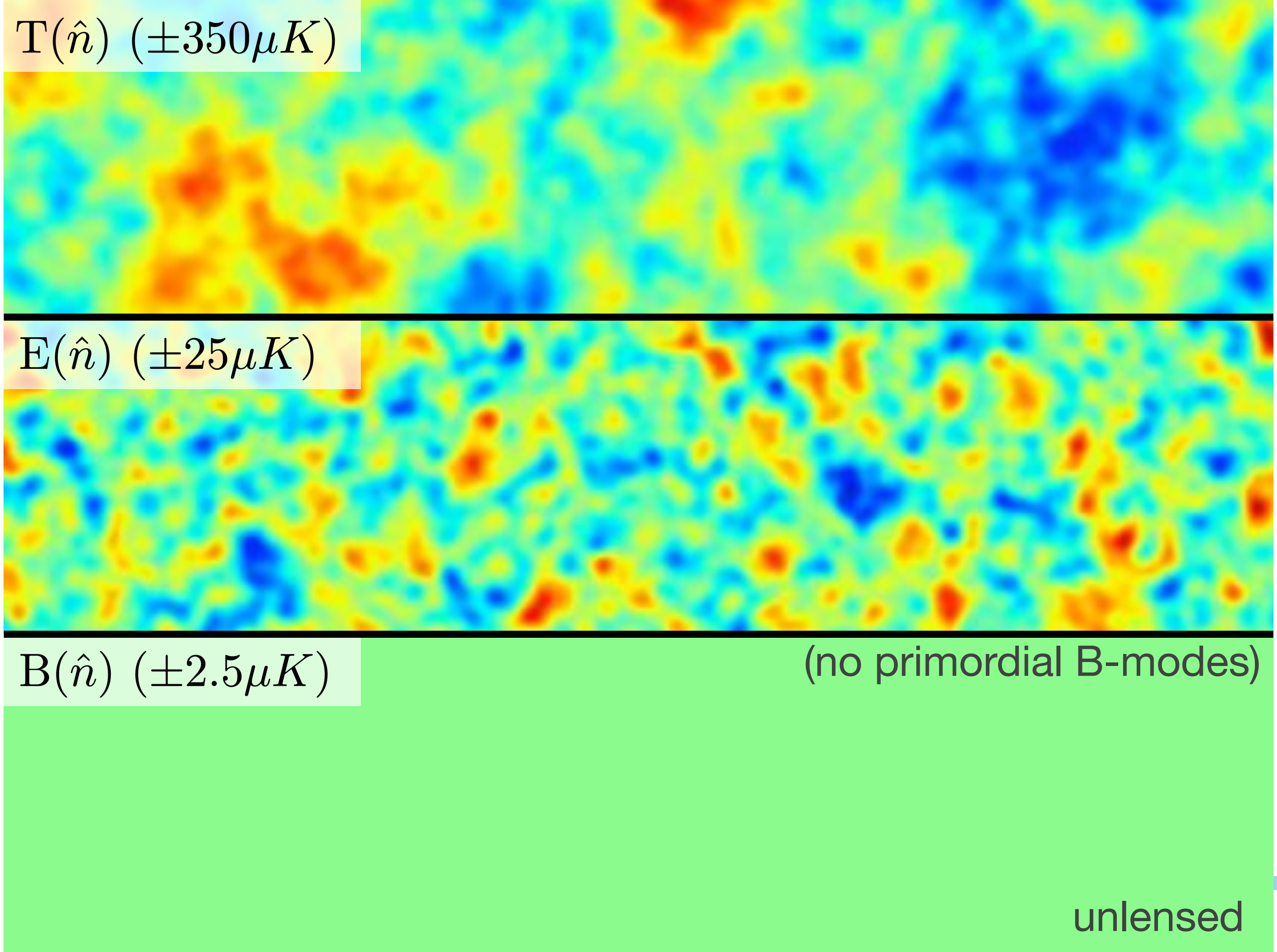
- primordial gravitational waves
- new particles that behave like radiation in the early universe (dark radiation), N_{eff}
- sum of neutrino masses

Why CMB lensing?



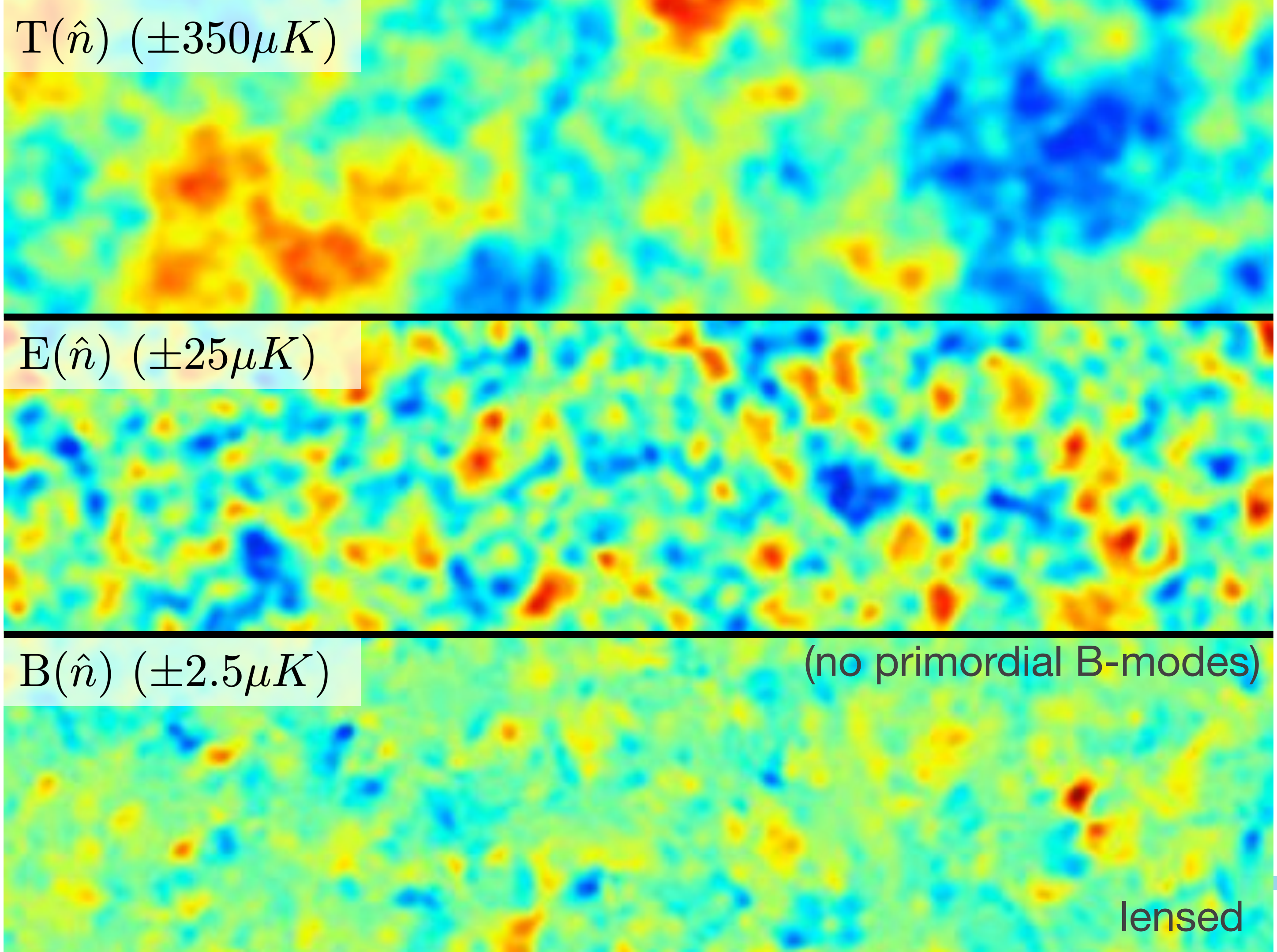
Credit: ESA

- info: provides handle on large-scale structure at low redshift
- nuisance: distorts primordial CMB —> use lensing potential to ‘undo’ the lensing (delensing)



Credit: Duncan Hanson

Fermilab



Credit: Duncan Hanson

Measuring lensing

Next-generation experiments (eg CMB-S4) will provide higher S/N CMB map measurements.

To optimally extract lensing information, we need

maximum-likelihood lensing estimators

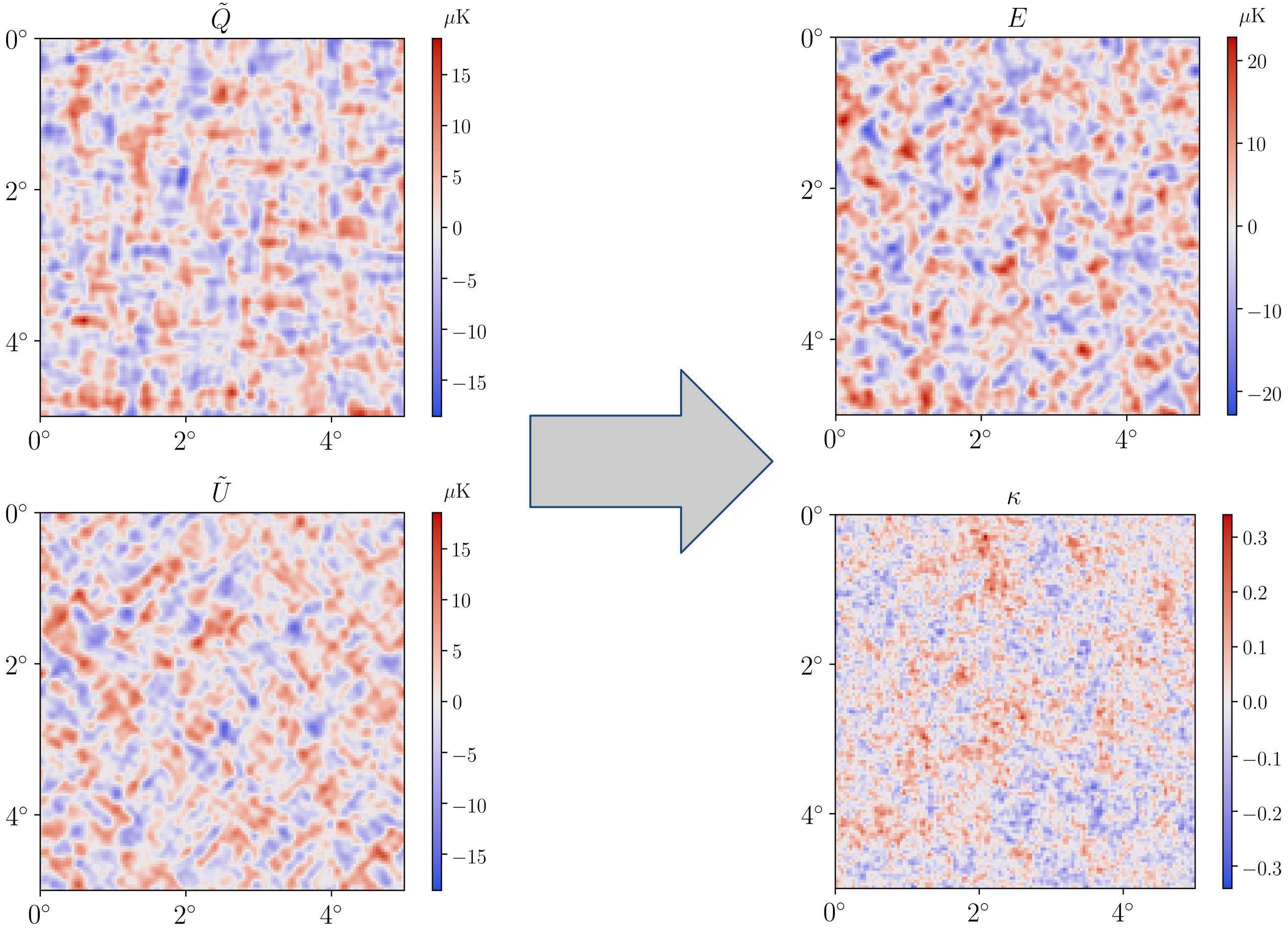
that can deal with foregrounds, systematics, ...

CMB lensing reconstruction: problem statement

Left: lensed polarization maps from the CMB.

Right: unlensed map (E) and lensing (κ).

Image-to-image regression!



Fermilab

Why neural networks?

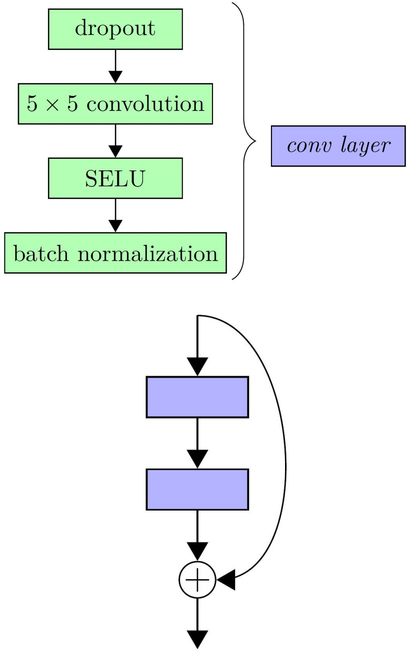
- lots of image-based technology developed

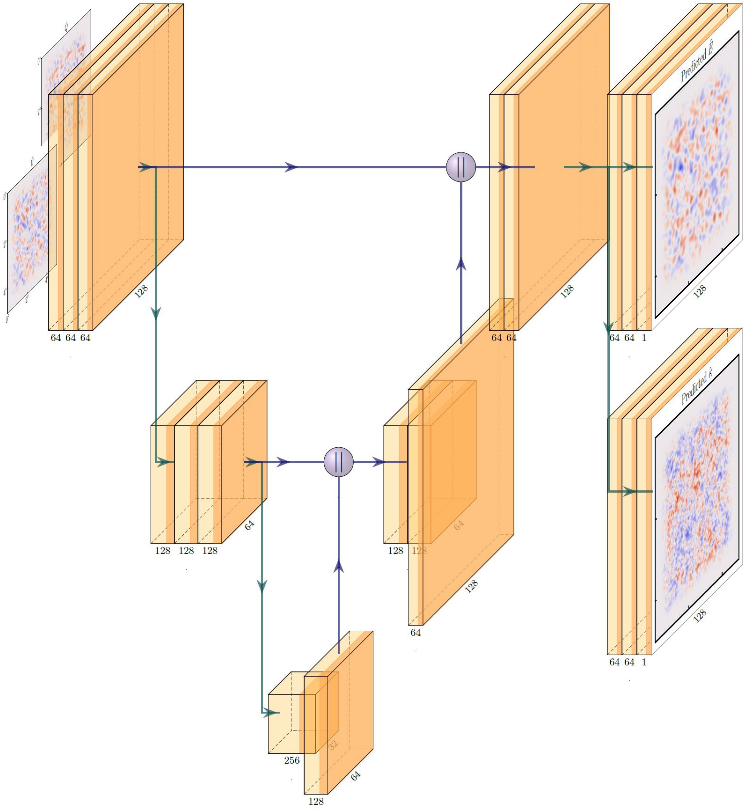
- powerful non-linearities could be the right tool to deal with beyond-Gaussian information

ResUNet architecture

11200 simulated (Q, U, E, κ) maps, divided 80:10:10 into training, validation and test sets.

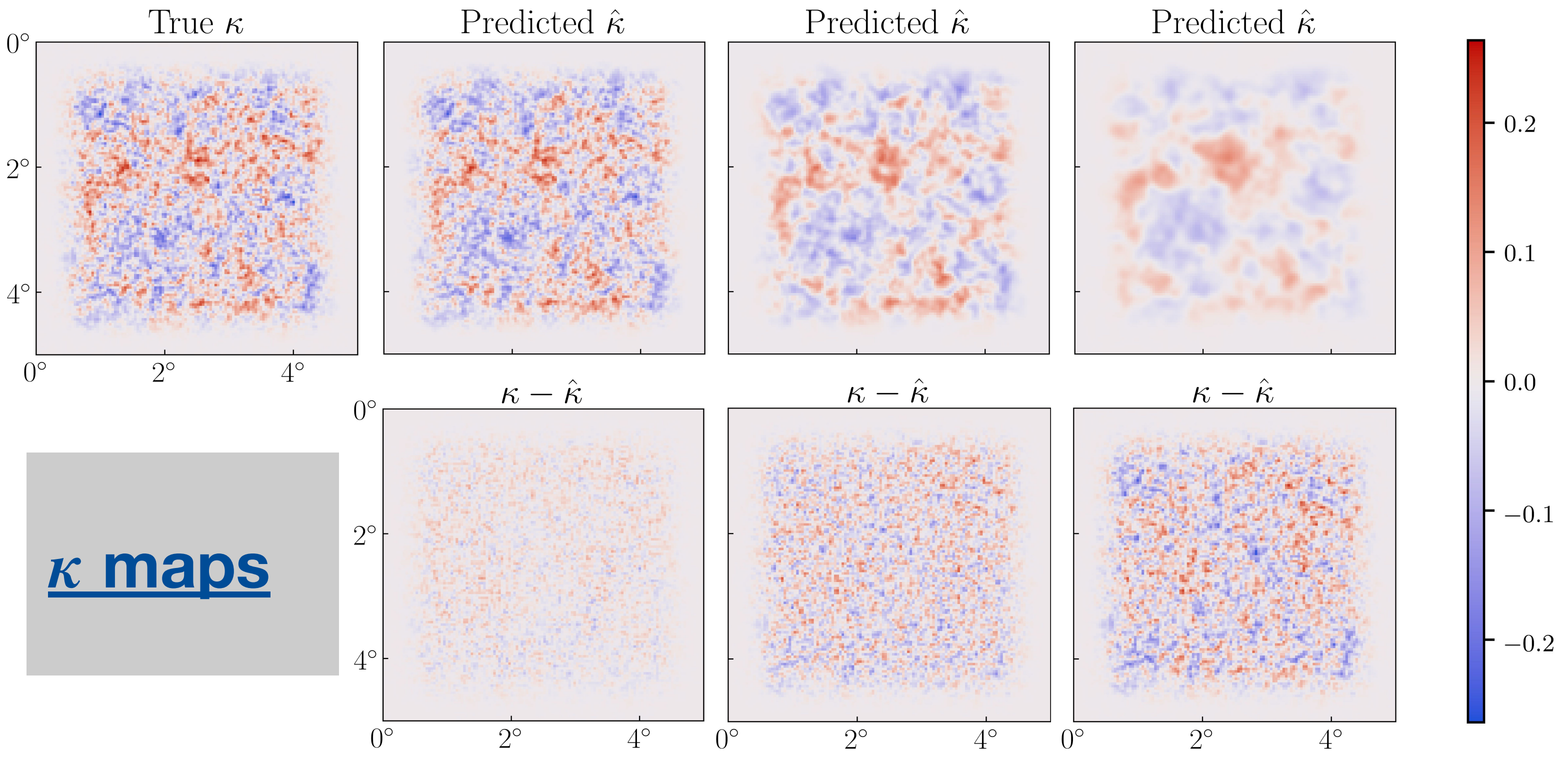
Loss function: MSE





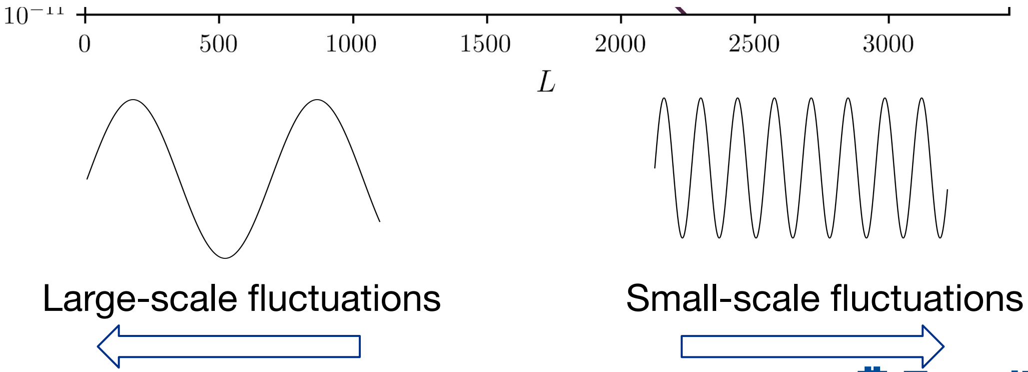
Results: lensing

0 μK-arcmin 1 μK-arcmin 5 μK-arcmin



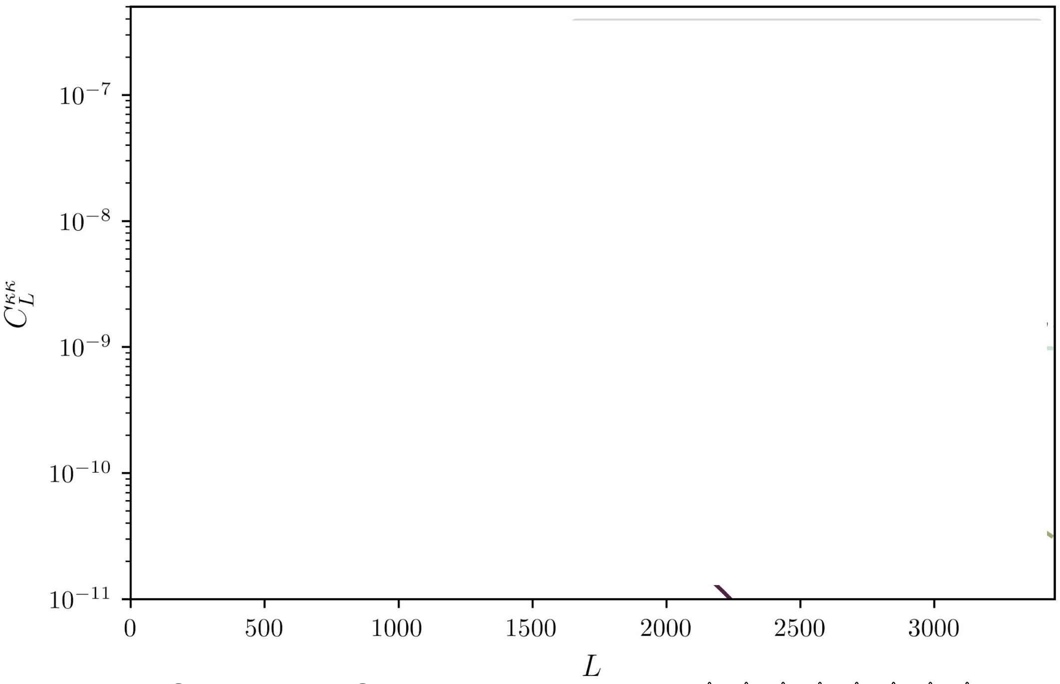
κ maps

Power spectra of κ



Power spectra of κ

Intensity/power

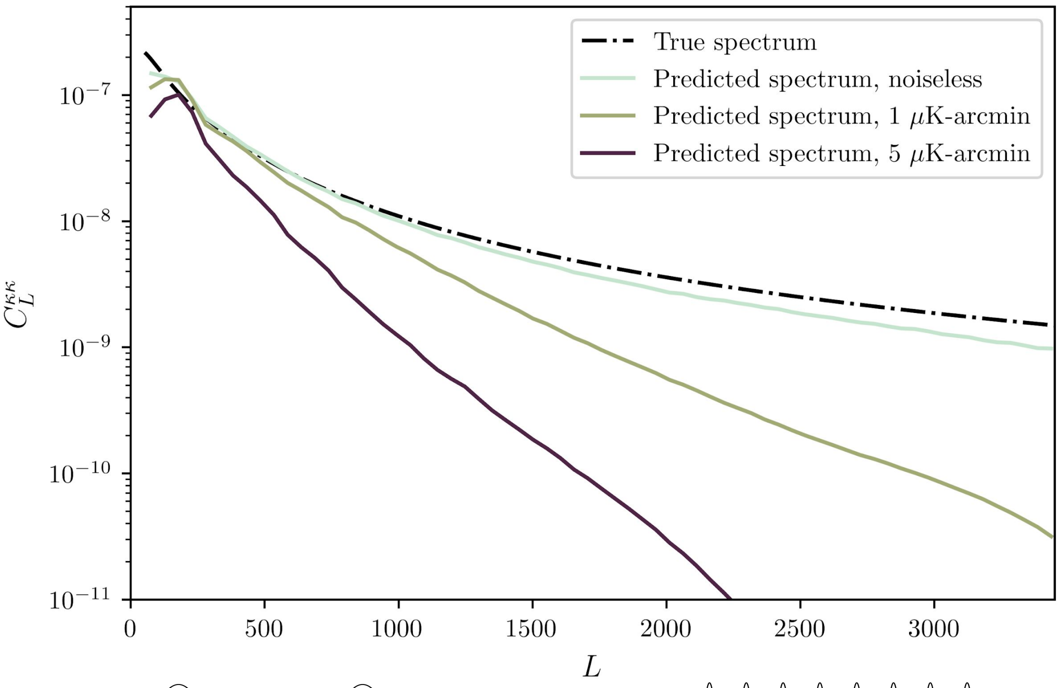


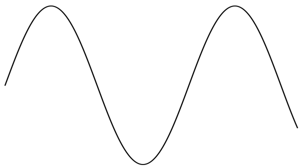
Large-scale fluctuations

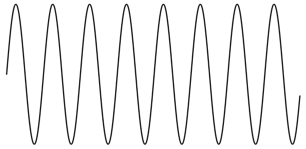
Small-scale fluctuations

Power spectra of κ

Intensity/power







Large-scale fluctuations

Small-scale fluctuations

Comparison with physics-based methods

Previous methods fixed cross-spectrum to be equal, so to compare we rescale

$$\hat{\hat{\kappa}} = \frac{1}{R}\hat{\kappa} \qquad R = \frac{\langle \kappa \hat{\kappa}^* \rangle}{\langle \kappa \kappa^* \rangle}$$

and then calculate the noise spectrum:

$$N_L^{\kappa\kappa} = \langle C_L^{\hat{\hat{\kappa}}\hat{\hat{\kappa}}} \rangle - \langle C_L^{\kappa\kappa} \rangle$$

κ : true map
$\hat{\kappa}$: predicted
$\hat{\hat{\kappa}}$: rescaled

Tests: different cosmologies

Q: The training, validation and test sets all have the same cosmology. Has the network just learned to reproduce that cosmology?

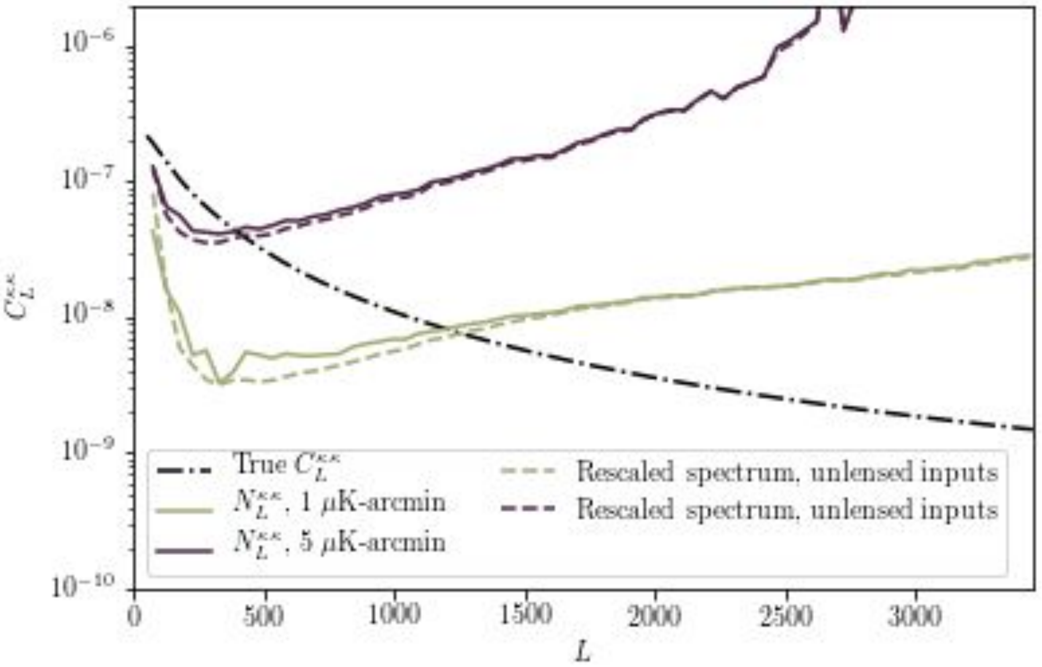
A: we have generated simulations with $\Omega_{CDM}h^2$=0.1085 and 0.1285 (0.1185 on fiducial cosmology), and ran those through the network.

We show that we can use the predicted lensing maps to estimate $\Omega_{CDM}h^2$ after correcting for bias in the noise correction.

Tests: no lensing

Q: What if we give the network unlensed Q and U maps? Does the network still try to reconstruct lensing?

A: No, it predicts a κ consistent with the noise spectrum we found.



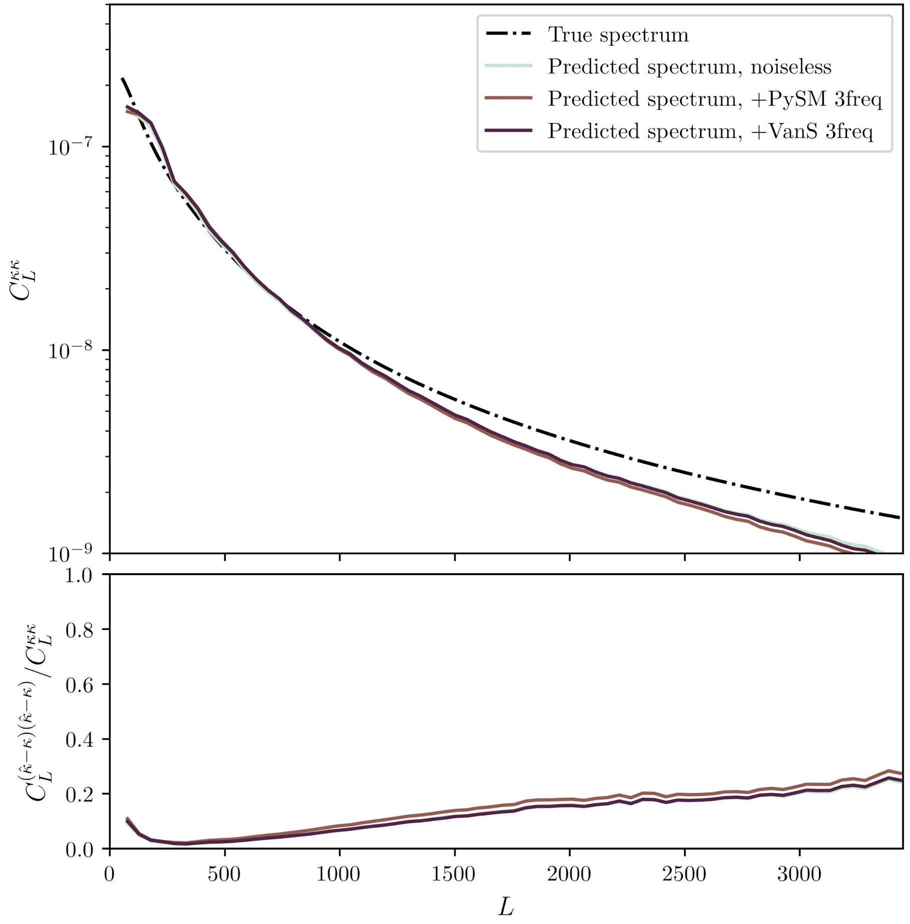
Fermilab

Follow-ups: foreground removal

Inputs are now lensed (Q, U) + foregrounds at different frequencies (95, 155, 220 GHz), so six maps in total. Same outputs.

Preliminary results are promising.

Of course, only so good as the foreground model is realistic… But that's true of any method



Fermilab

Overview

- From before: DeepCMB (1810.01483 + 20xx.xxxxx)

- **Using quantum computing for machine learning (1911.06259 + 20xx.xxxxx)**

- Uncertainty quantification for deep learning (2004.xxxxx)

We are entering the NISQ era

Quantum Computing in the NISQ era and beyond

John Preskill

Institute for Quantum Information and Matter and Walter Burke Institute for Theoretical Physics, California Institute of Technology, Pasadena CA 91125, USA

30 July 2018

Noisy Intermediate-Scale Quantum (NISQ) technology will be available in the near future. Quantum computers with 50-100 qubits may be able to perform tasks which surpass the capabilities of today's classical digital computers, but noise in quantum gates will limit the size of quantum circuits that can be executed reliably. NISQ devices will be useful tools for exploring many-body quantum physics, and may have other useful applications, but the 100-qubit quantum computer will not change the world right away — we should regard it as a significant step toward the more powerful quantum technologies of the future. Quantum technologists should continue to strive for more accurate quantum gates and, eventually, fully fault-tolerant quantum computing.

Current quantum computers

Quantum annealers

- non-universal
- essentially draw samples from an Ising Hamiltonian

$$H = -b_i s_i - J_{ij} s_i s_j$$

- can dial couplings b, J
- D-Wave 2000Q: 2048 qubits

Gate-based

- different technologies: ion trap, superconducting systems, photonic systems…
- we're up to ~50 qubits
- universal

Current quantum computers

Quantum annealers

- non-universal
- essentially draw samples from an Ising Hamiltonian

$$H = -b_i s_i - J_{ij} s_i s_j$$

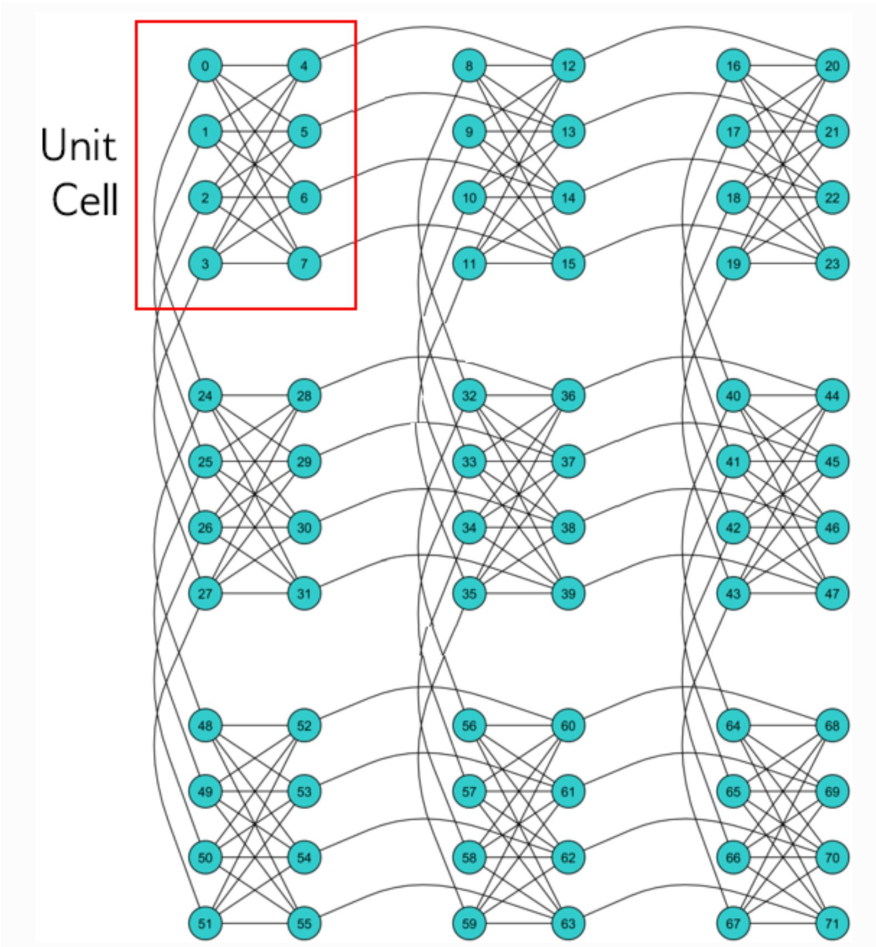
- can dial couplings b, J
- D-Wave 2000Q: 2048 qubits

Gate-based

- different technologies: ion trap, superconducting systems, photonic systems…
- we're up to ~50 qubits
- universal

D-Wave (2000Q)

Chimera graph:

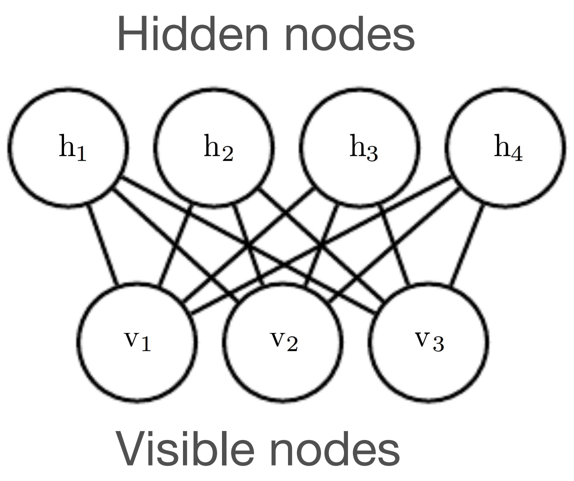


D-Wave 2000Q has 16x16 cells: total 2048 qubits

Need to embed problem graph into chimera graph (minor embedding)

Fermilab

Restricted Boltzmann Machines (RBM)



Learn the probability distribution of the data (which goes into the visible nodes),

$$P(v,h) = \frac{1}{Z} e^{-E(v,h)}$$

$$E(v,h) = -b_i v_i - c_j h_j - J_{ij} v_i h_j$$

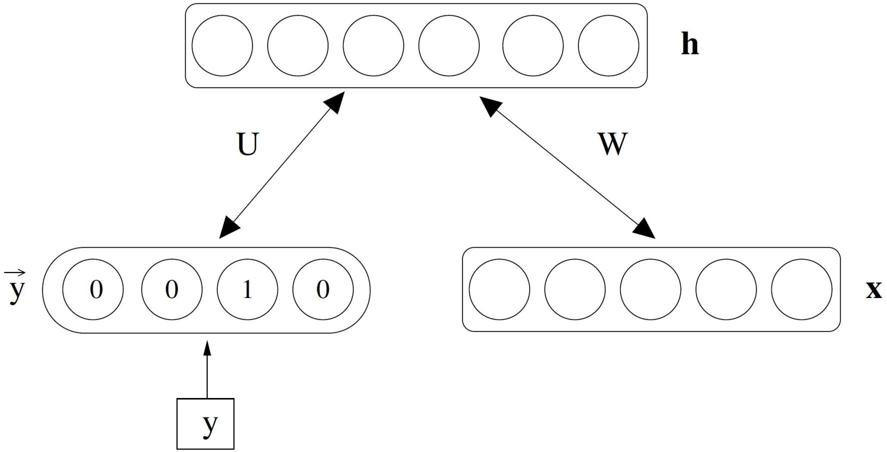
For inference, evaluate likelihood using free energy

$$\mathcal{F} = \log \sum_h e^{-E(v,h)}$$

Image credit: Goodfellow, Bengio and Courville, *Deep Learning*

RBM for classification

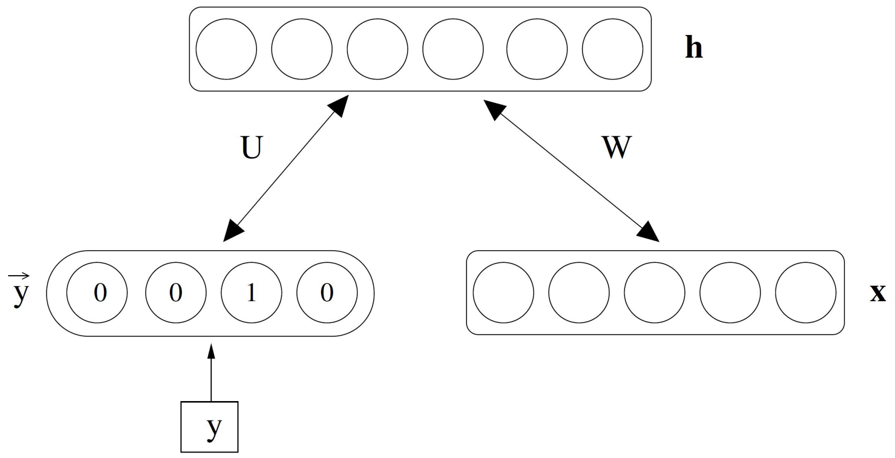
Encode label on visible nodes



For the test set, try data with all labels, predict the smallest free energy.

RBM for classification

Encode label on visible nodes



For the test set, try data with all labels, predict the smallest free energy.

Training:

$$\delta w_{ij} = \langle v_i h_j \rangle_{\mathrm{data}} - \langle v_i h_j \rangle_{\mathrm{model}}$$
$$\delta b_i = \langle v_i \rangle_{\mathrm{data}} - \langle v_i \rangle_{\mathrm{model}}$$
$$\delta c_j = \langle h_j \rangle_{\mathrm{data}} - \langle h_j \rangle_{\mathrm{model}}$$

Model expectations are hard to compute classically. Quantum advantage?

Image credit: Larochelle and Bengio (2008)

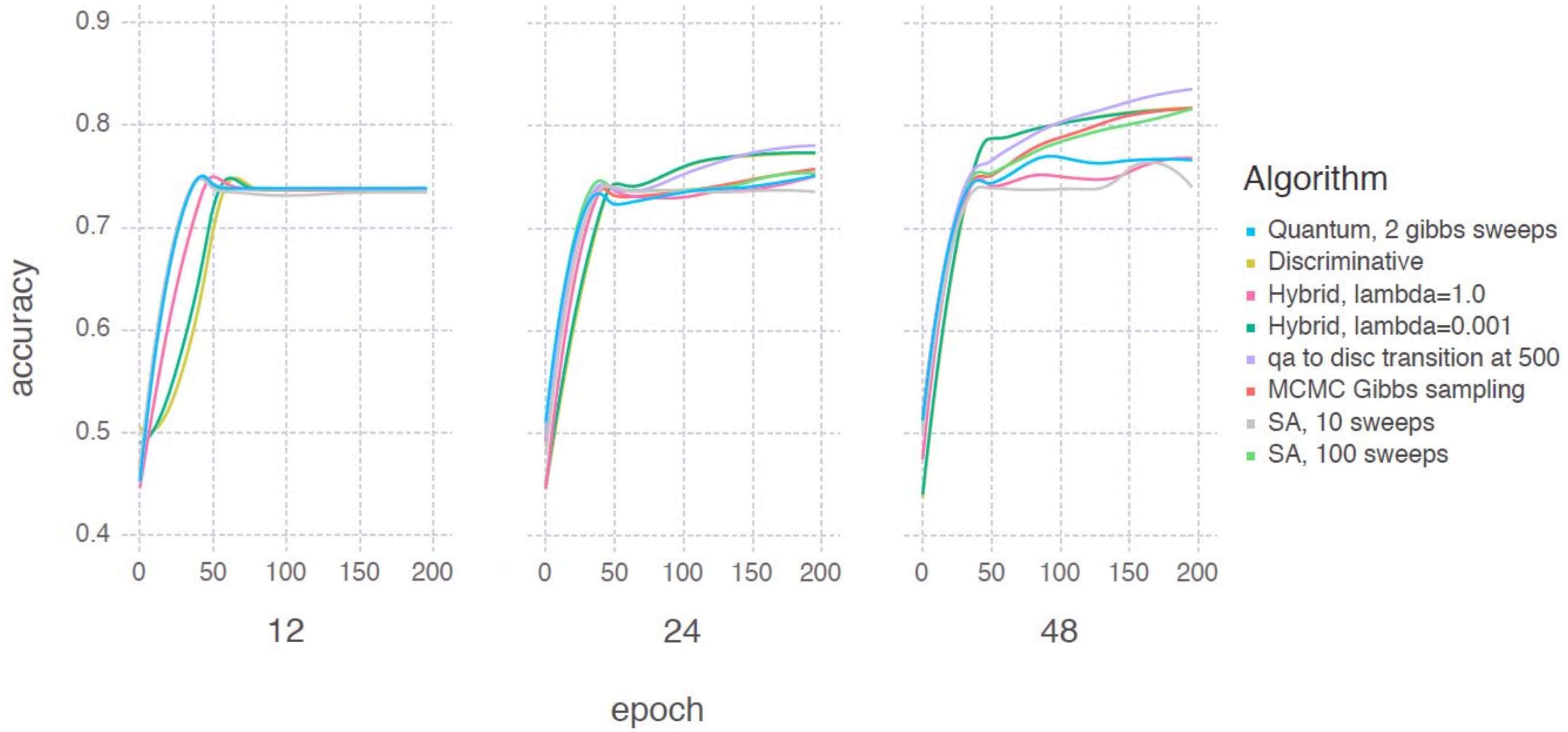
Use case

Galaxy morphology: spirals vs rounded

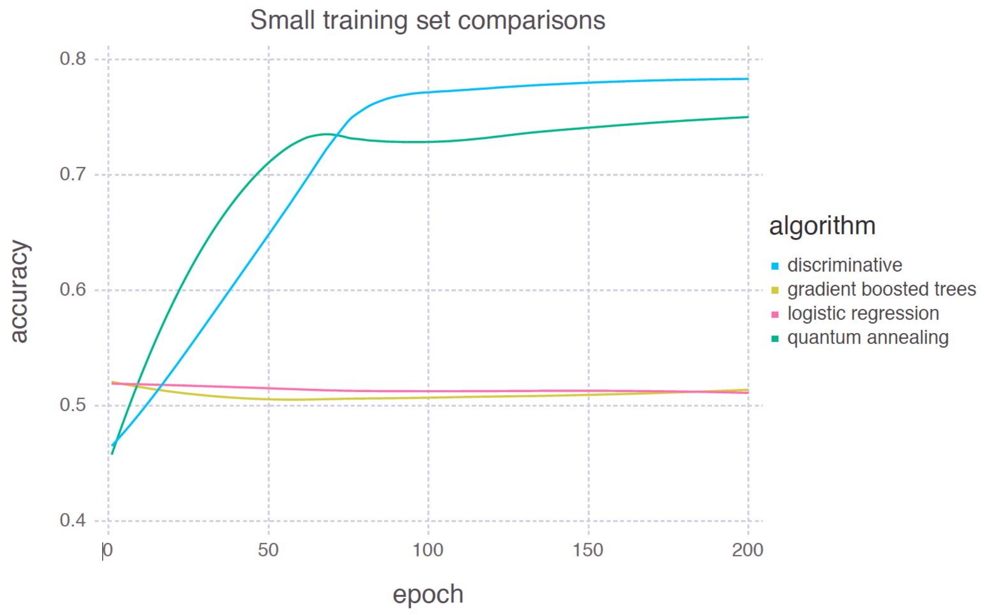
We tested quantum RBM vs classical RBM and other classical classifiers on this task. Note data has to be compressed a lot! We used PCA.

Accuracy



accuracy
0.9
0.8
0.7
0.6
0.5
0.4
0
50
100
150
200
12
24
48
epoch
Algorithm
Quantum, 2 gibbs sweeps
Discriminative
Hybrid, lambda=1.0
Hybrid, lambda=0.001
qa to disc transition at 500
MCMC Gibbs sampling
SA, 10 sweeps
SA, 100 sweeps

Silver linings: small training set, early training



Other observations

- Unless quantum annealer is also used for inference, need the annealer to provide Boltzmann distribution at known temperature

- We saw distributions far from Boltzmann, so hard to use in this way. Would not currently recommend quantum annealer as a Boltzmann sampler

- Quantum gate models, on universal quantum computers, potentially more useful… (paper using Google's sycamore chip forthcoming)

Overview

- From before: DeepCMB (1810.01483 + 20xx.xxxxx)
- Using quantum computing for machine learning (1911.06259 + 20xx.xxxxx)
- **Uncertainty quantification for deep learning (2004.xxxxx)**

How can you quantify uncertainty?

Many deep learning applications in physics and other fields. But we need accurate uncertainty quantification to trust the results.

In recent years, many methods have been put forward:

- deep ensembles
- concrete dropout
- bayesian neural networks
- …

How should one choose?

Epistemic and aleatoric, statistical and systematic

Uncertainty in deep learning is often divided into

aleatoric or *irreducible*: uncertainty related to corruption of input data, such as detector noise	*epistemic* or *reducible*: uncertainty stemming from an imperfect model, goes down with more data

Epistemic and aleatoric, statistical and systematic

Uncertainty in deep learning is often divided into

aleatoric or *irreducible*: uncertainty related to corruption of input data, such as detector noise	*epistemic* or *reducible*: uncertainty stemming from an imperfect model, goes down with more data

Epistemic is always systematic

Statistical is always aleatoric

No statistical epistemic uncertainties

Problem summary:
1. Which uncertainty quantification method to choose
2. How to interpret the results

Fermilab

Problem summary:

1. Which uncertainty quantification method to choose
2. How to interpret the results

Our contribution:

Build simple sandbox with pendulum problem

From (L, T, m, θ), predict $g = 4\pi^2 L/T^2$: a problem any physics undergrad is familiar with

We include many measurements of T, to allow calculation of a statistical uncertainty

Setup

3 hidden layers with 100 nodes each

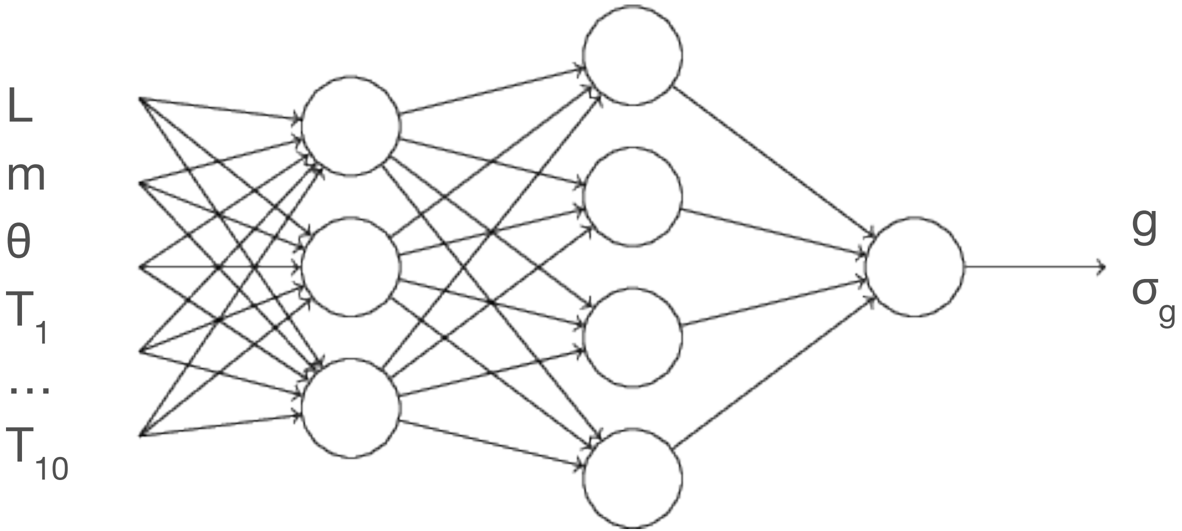


Image credit:
Michael Nielsen

Brief introduction to the UQ methods

For all these methods, optimize (g, σ_g) to maximize Gaussian log likelihood of right answer, so loss is

$$L = \log \sigma_g + \frac{1}{2}\left(\frac{g - g_{true}}{\sigma_g}\right)^2$$

σ_g provides an estimate of the aleatoric uncertainty, while the variance between different models' predictions gives epistemic uncertainty

Deep ensembles: different models

Concrete dropout: dropping different neurons

Bayesian NN: each weight is sampled from distribution

How to introduce noise

Statistical (aleatoric): add noise to the T measurements (sample them from a normal distribution)

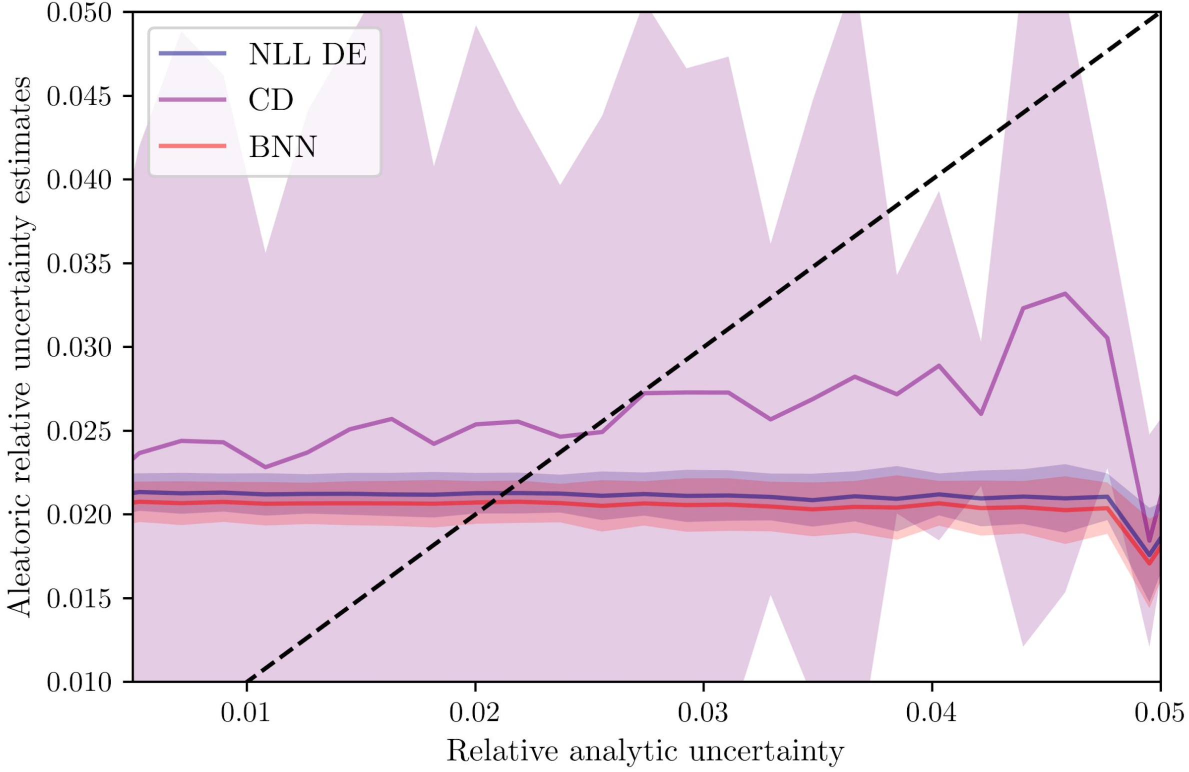
Systematic (aleatoric): add noise to the single L measurement

Systematic (epistemic): smaller training set, or test in different region from training set

Fermilab

Results: how well is aleatoric uncertainty captured?

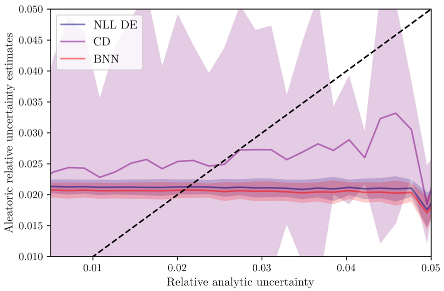
Trained on data with T variation in range 1-5%:



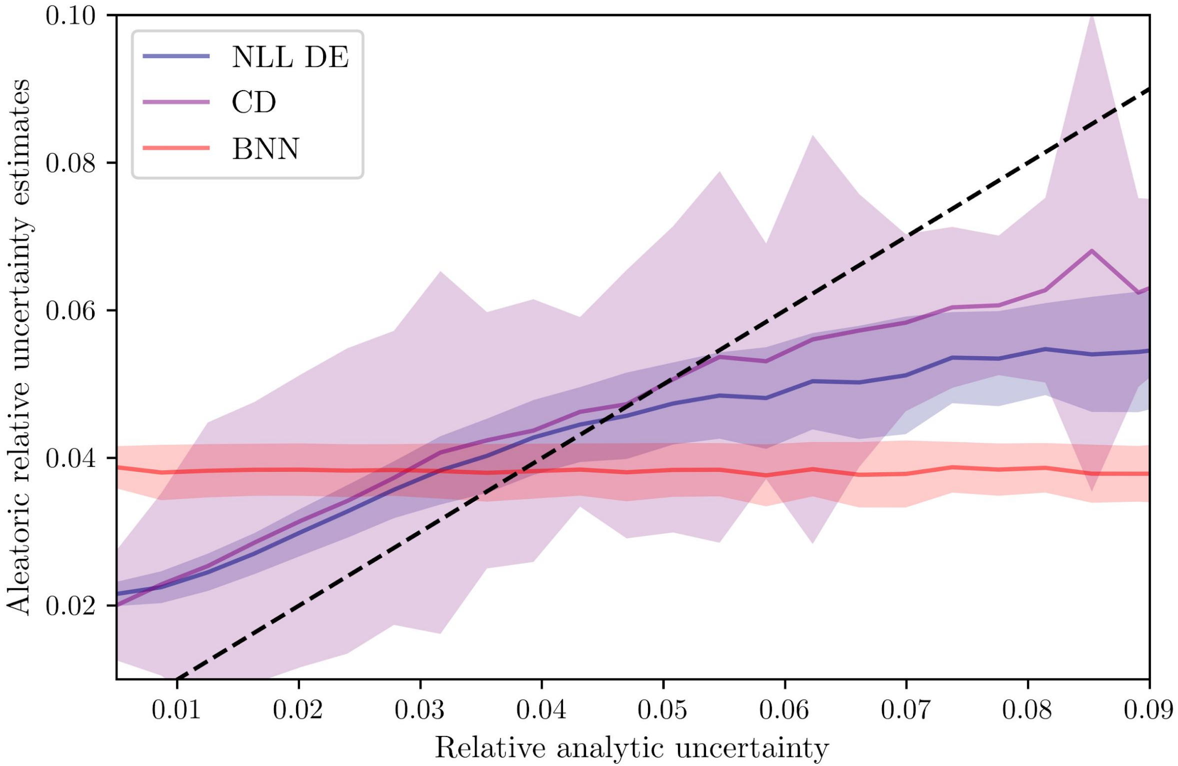
Fermilab

Results: how well is aleatoric uncertainty captured?

Range 1-5%:



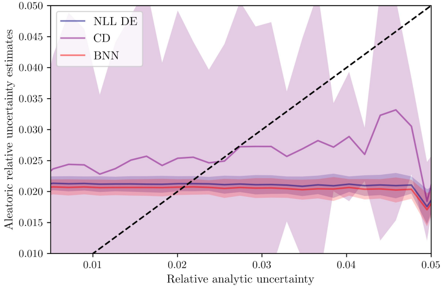
Trained on data with T variation in range 1-10%:



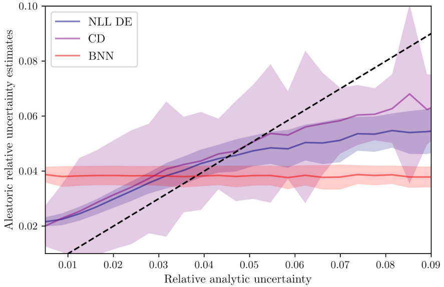
Fermilab

Results: how well is aleatoric uncertainty captured?

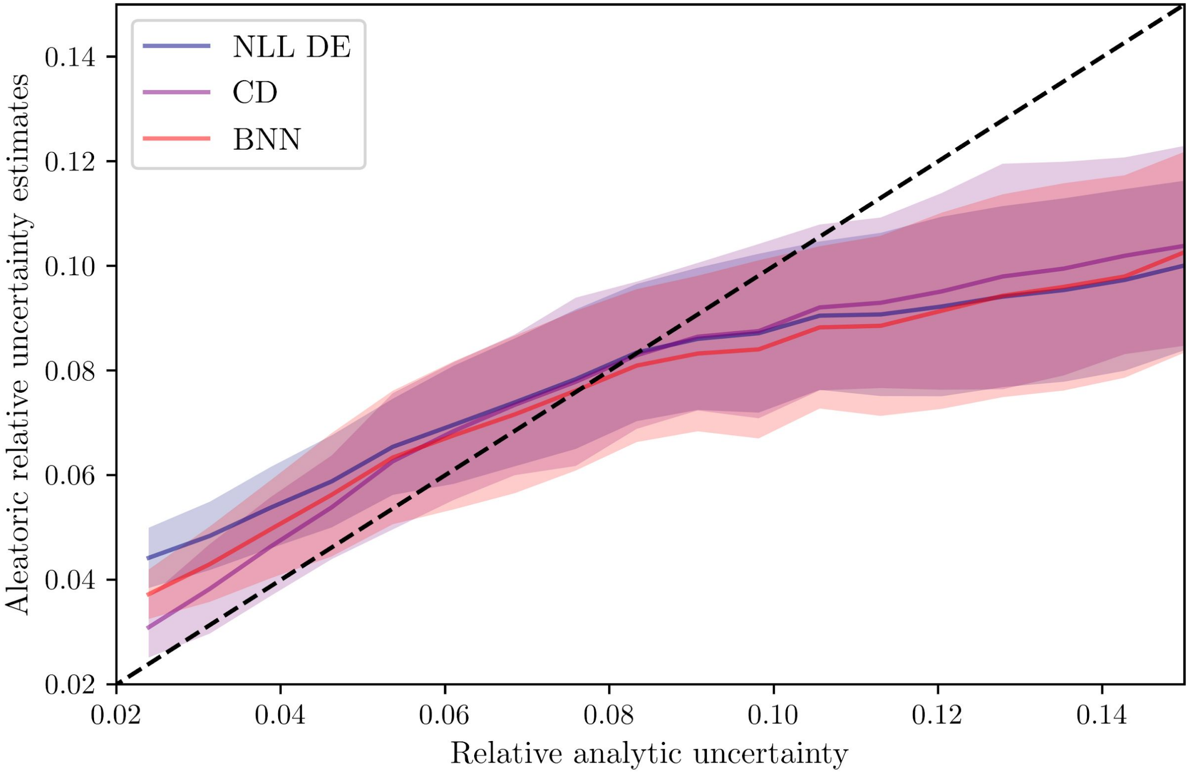
Range 1-5%



Range 1-10%



Trained on data with T variation in range 1-20%:

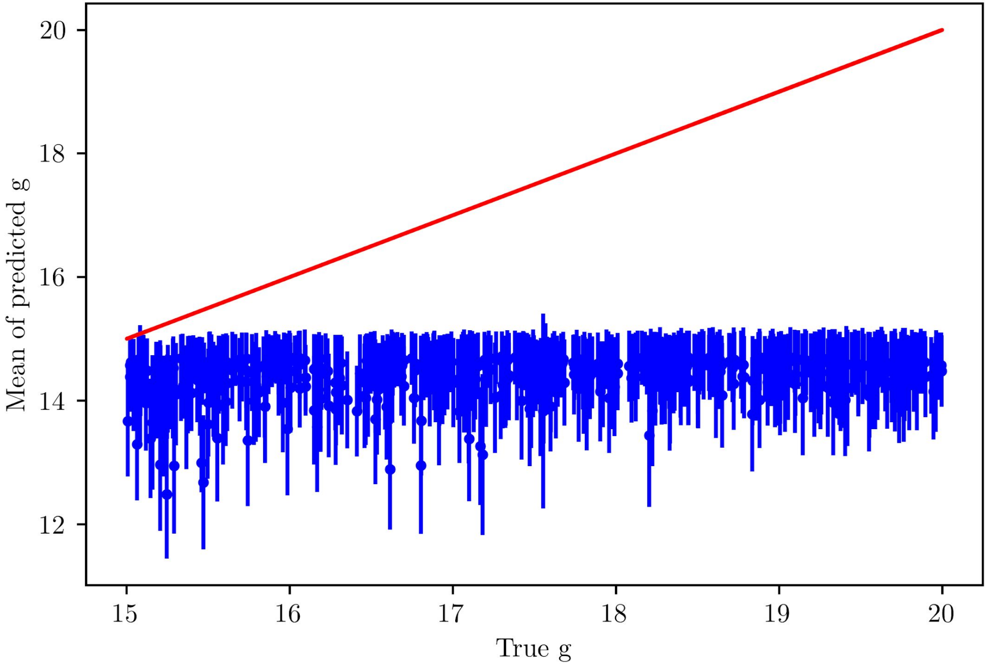


Fermilab

Results: out of distribution uncertainties

Several ways to go out of distribution. We train on g in [5, 15] m/s^2.

Can test on g in [15, 20] m/s^2:



Terrible results!

We can keep g same, vary L and T out of distribution.

But that problem is too easy!

Conclusions

- For aleatoric uncertainties, they are reasonably well-modeled: but we need to make sure to include a large range of uncertainties in the training set, or it won't see enough variation. Just like for predictions, but tricky!

- None of these methods know they go out of distribution in this simple test! Probably we need better methods in such a situation.

Overview

- From before: DeepCMB (1810.01483 + 20xx.xxxxx)
- Using quantum computing for machine learning (1911.06259 + 20xx.xxxxx)
- Uncertainty quantification for deep learning (2004.xxxxx)