

Visione Artificiale

Raffaella Lanza

Deep Learning

Limitations and New Frontiers

The Rise of Deep Learning

'Deep Voice' Software Can Clone Anyone's Voice With Just 3.7 Seconds of Audio

Using snippets of voices, Baidu's 'Deep Voice' can generate new speech, accents, and tones.



DEEPMIND I STARCRAFT TRIUMPH

Let There Be Sight: How Deep Learning Is Helping the Blind 'See'



Technology outpacing security measures



AI beats docs in cancer spotting

A new study provides a fresh example of machine learning as an important diagnostic tool, Paul Biegler reports.



AI Can Help in Predicting Cryptocurrency Value



'Creative' AlphaZero leads way for chess computers and, maybe, science

Former chess world champion Garry Kasparov likes what he sees of computer that could be used to find cures for diseases



How an A.I. 'Cat-and-Mouse Game' Generates Believable Fake Photos

By CARL WETZ and KEITH COLLINS JAN. 3, 2018



Stock Predictions Based On AI: Is the Market Truly Predictable?



Complex of bacteria-infecting viral proteins modeled in CASP13. The complex that were modeled individually. PROTEIN DATA BANK

Google's DeepMind aces protein folding

By Robert F. Service | Dec. 6, 2019, 12:05 PM

After Millions of Trials, These Simulated Humans Learned to Do Perfect Backflips and Cartwheels

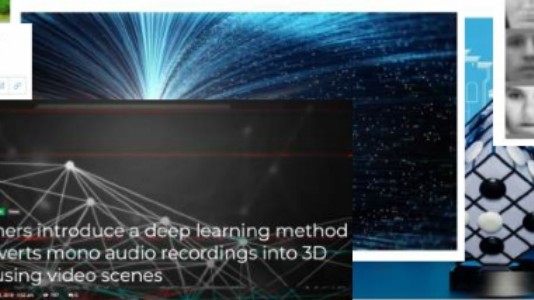
George Dooling 4/10/18 11:58am - Pinned to AI+



Neural networks everywhere

New chip reduces neural networks' power consumption by up to 95 percent, making them practical for battery-powered devices.

Wed, 01/16/2019 - Boston | Comment by Kenny Walter - Digital Reporter - @RandIMagazine



AI faces show how far AI image generation has come in just four years

Are on the right aren't real; they're the product of machine learning

By Pauline Wang, October 2017



Automation And Algorithms: De-Risking Manufacturing With Artificial Intelligence

Sarah Goshrie Contributor Manufacturing Focus on the industrialization of additive manufacturing

TWEET THIS

The two key applications of AI in manufacturing are pricing and manufacturability feedback

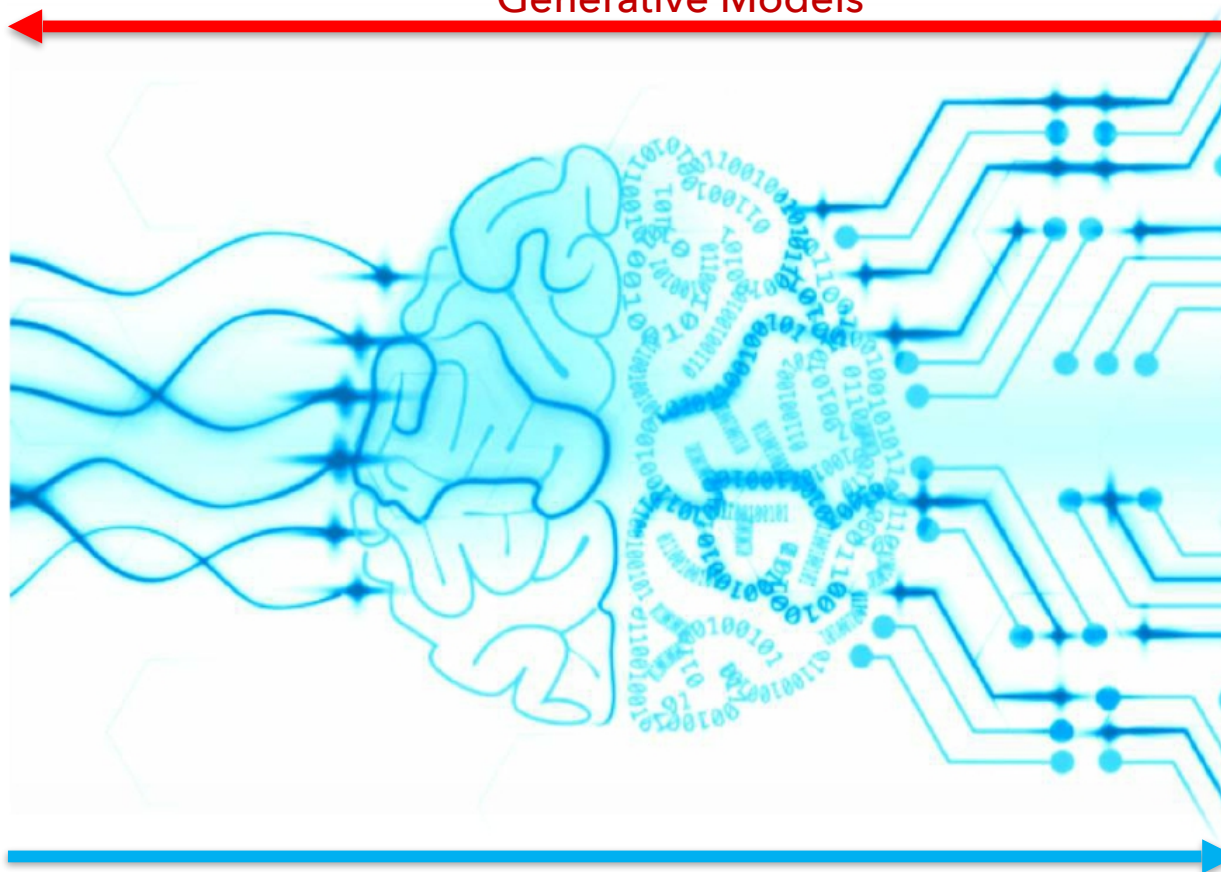
So far..

Generative Models

Data

- Signals
- Images
- Sensors

...



Decision

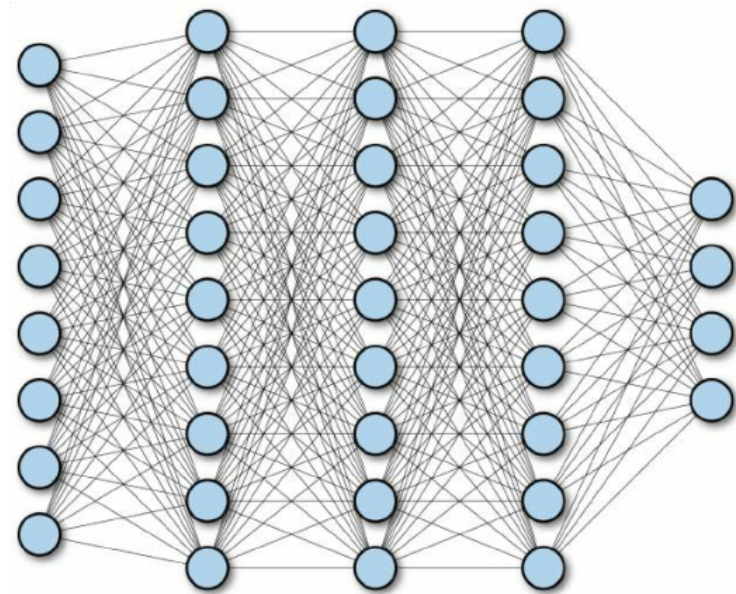
- Prediction
- Detection
- Action

...

Power of Neural Nets

Universal Approximation Theorem

A feedforward network with a single layer is sufficient to approximate, to an arbitrary precision, any continuous function.



Hornik et al. *Neural Networks*. (1989)

Power of Neural Nets

Universal Approximation Theorem

A feedforward network with a single layer is sufficient to approximate, to an arbitrary precision, any continuous function.

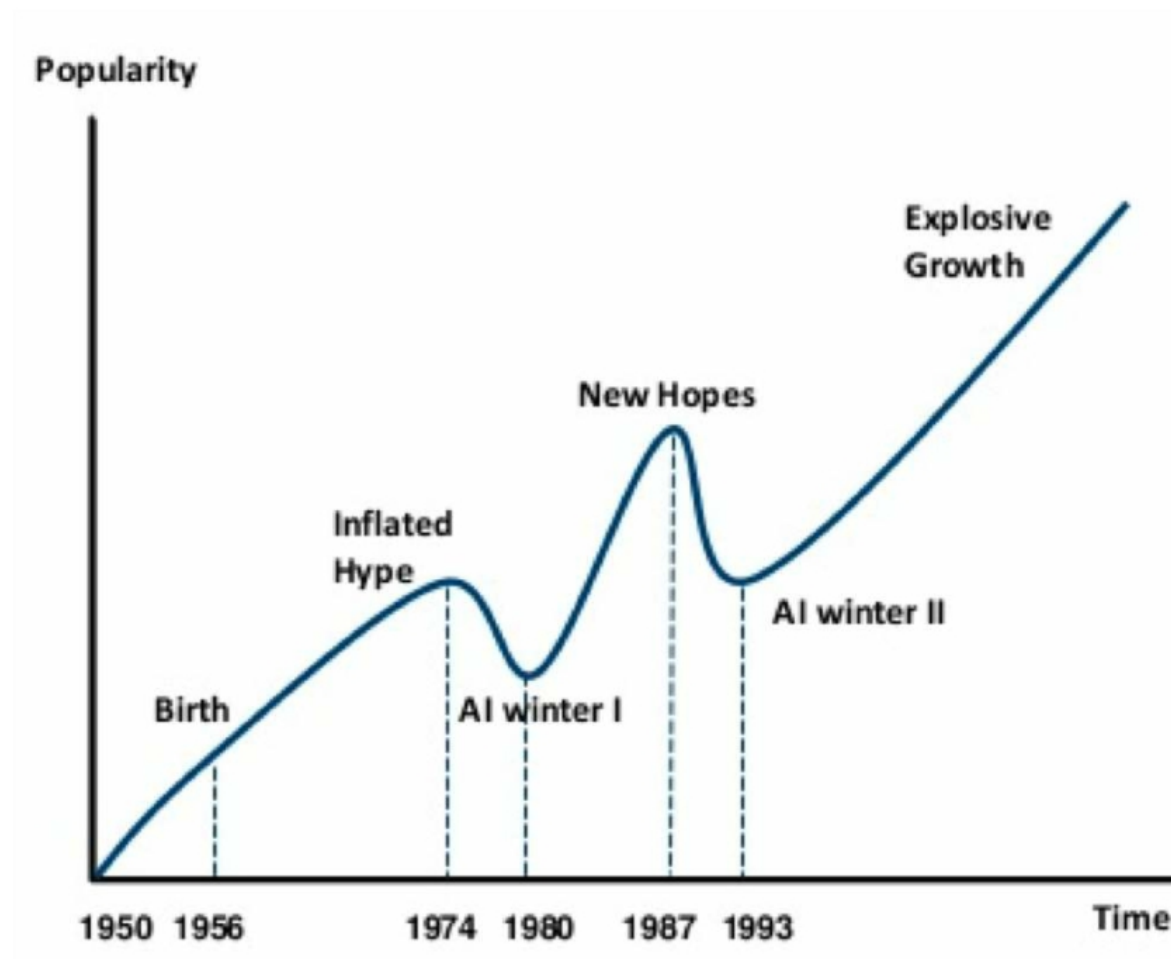
Caveats:

The number of hidden units may be infeasibly large

The resulting model may not generalize

Hornik et al. *Neural Networks*. (1989)

Artificial Intelligence "Hype": Historical Perspective



Limitations

Rethinking Generalization

“Understanding Deep Neural Networks Requires Rethinking Generalization



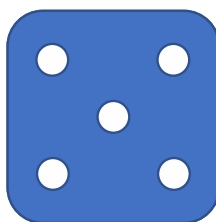
dog



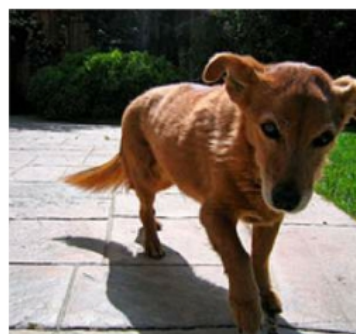
banana



banana



dog



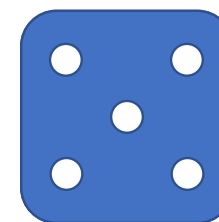
dog



tree



tree



dog

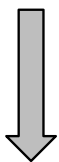
Zhang et al. *ICLR*. (2017)

Rethinking Generalization

“Understanding Deep Neural Networks Requires Rethinking Generalization



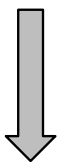
~~dog~~



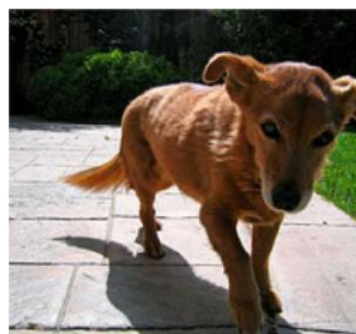
banana



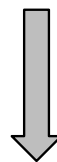
~~banana~~



dog



~~dog~~



tree



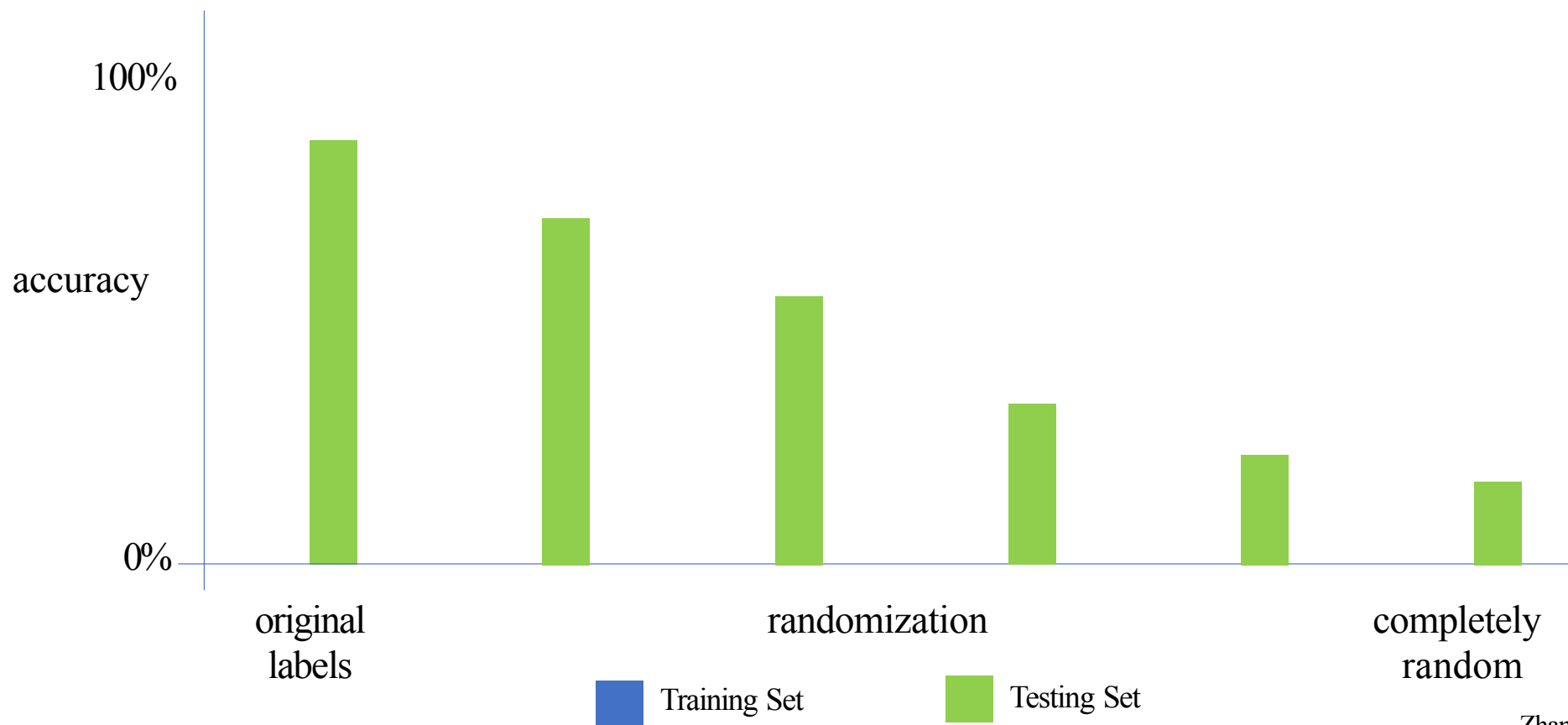
~~tree~~



dog

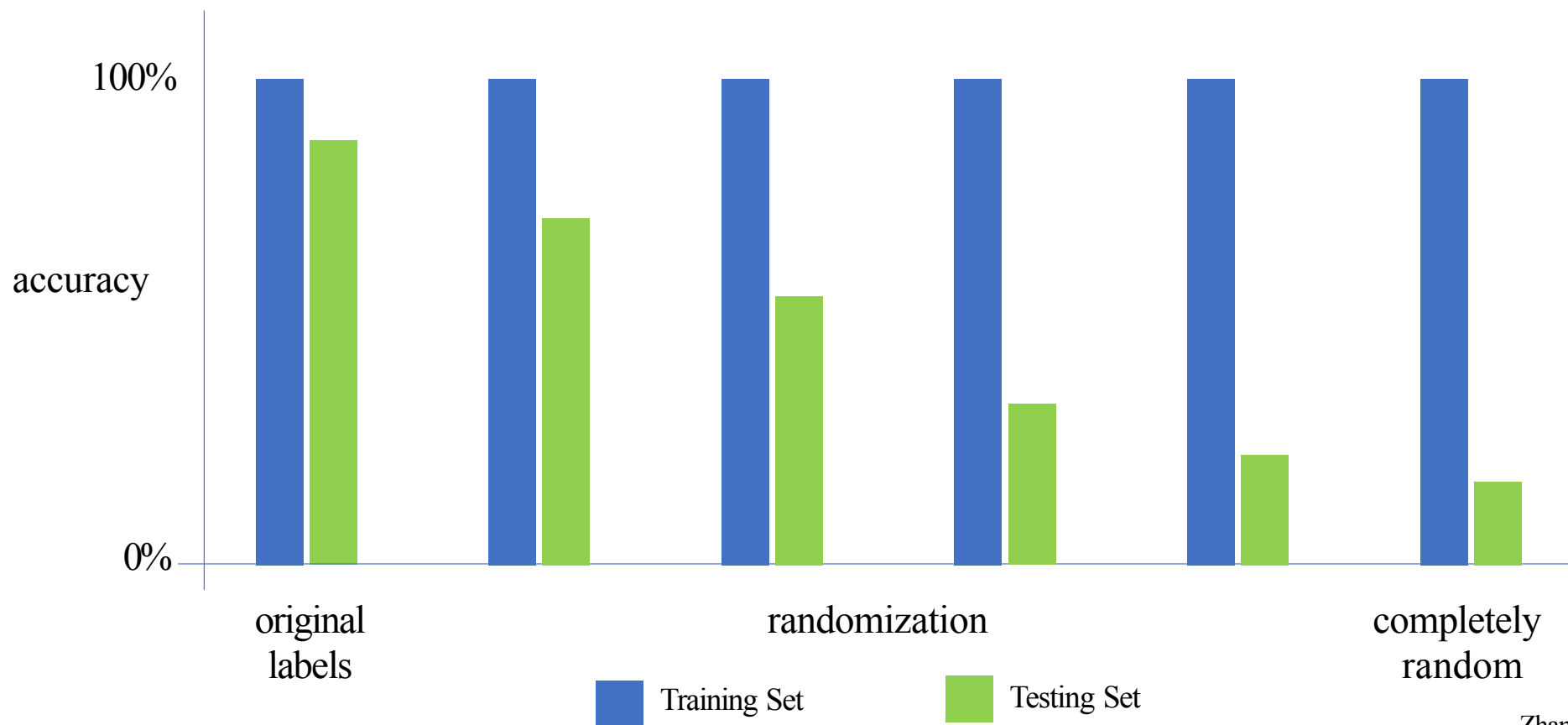
Zhang et al. *ICLR*. (2017)

Capacity of Deep Neural Networks



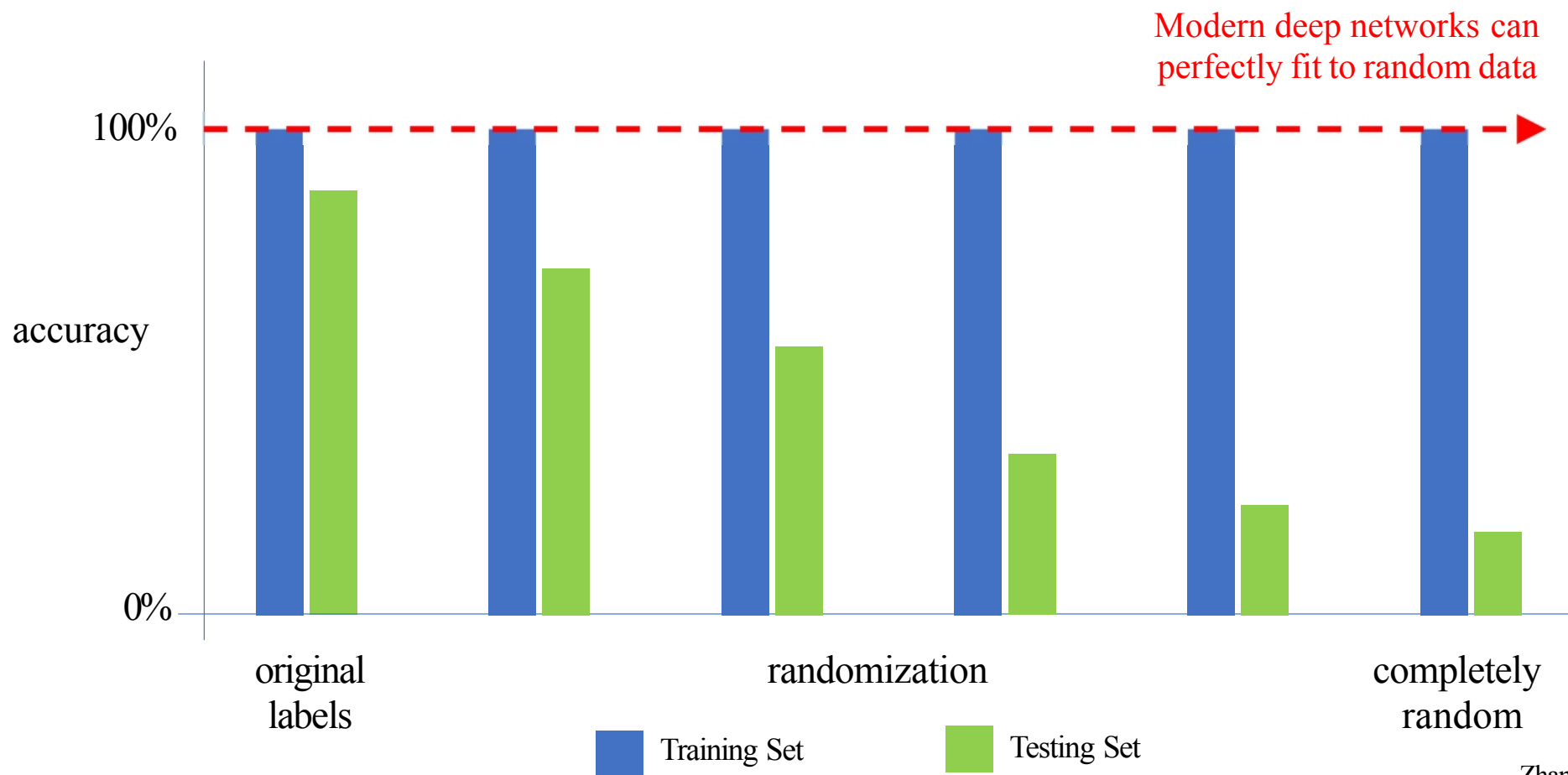
Zhang et al. *ICLR*. (2017)

Capacity of Deep Neural Networks



Zhang et al. *ICLR*. (2017)

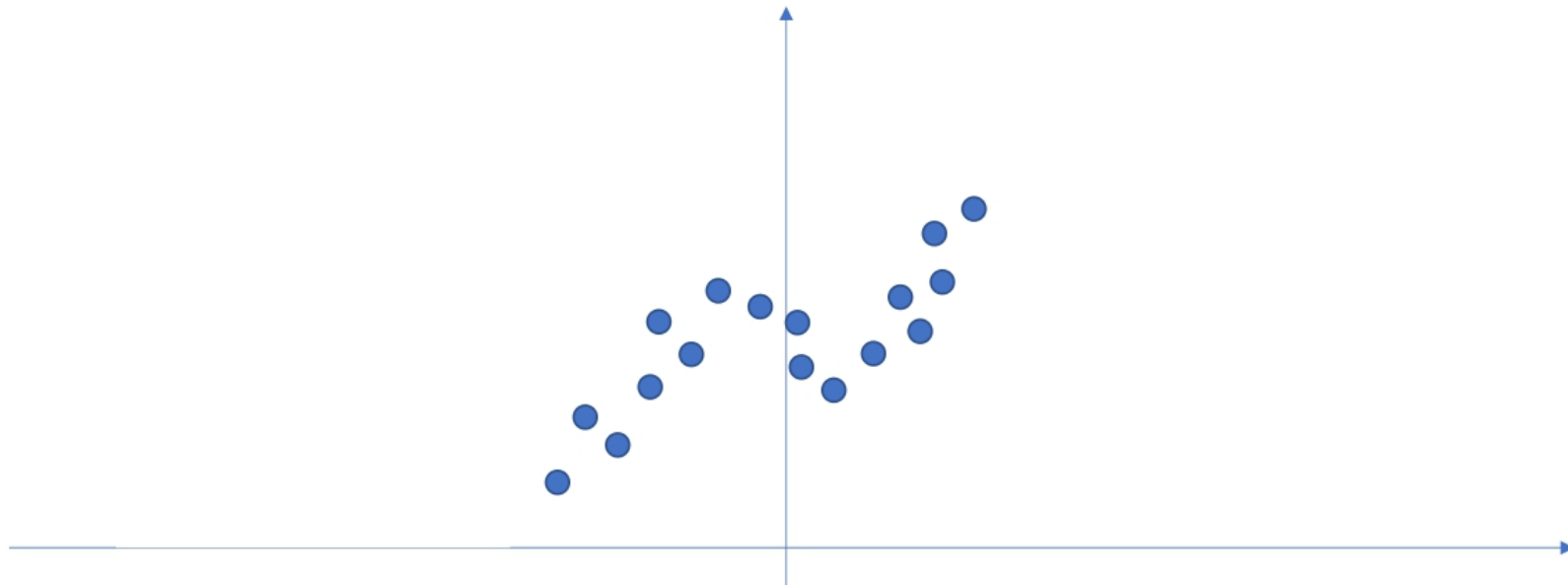
Capacity of Deep Neural Networks



Zhang et al. *ICLR*. (2017)

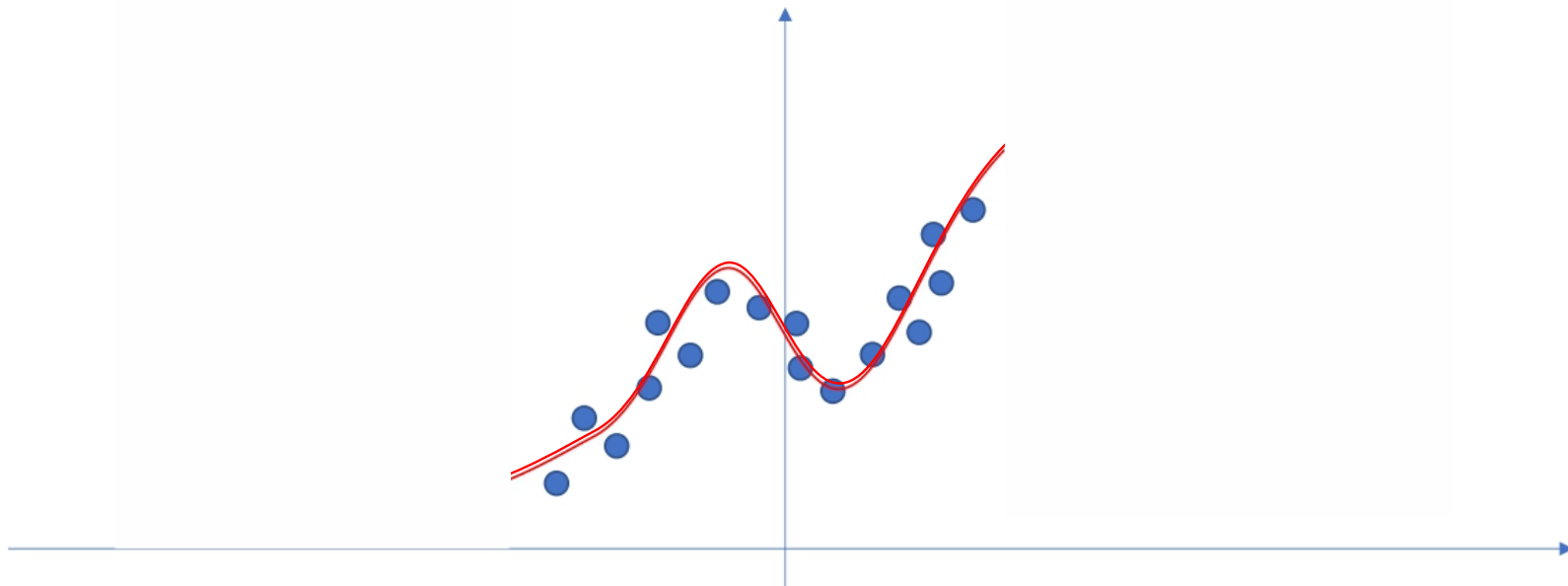
Neural Networks as Function Approximators

Neural networks are **excellent** function approximators



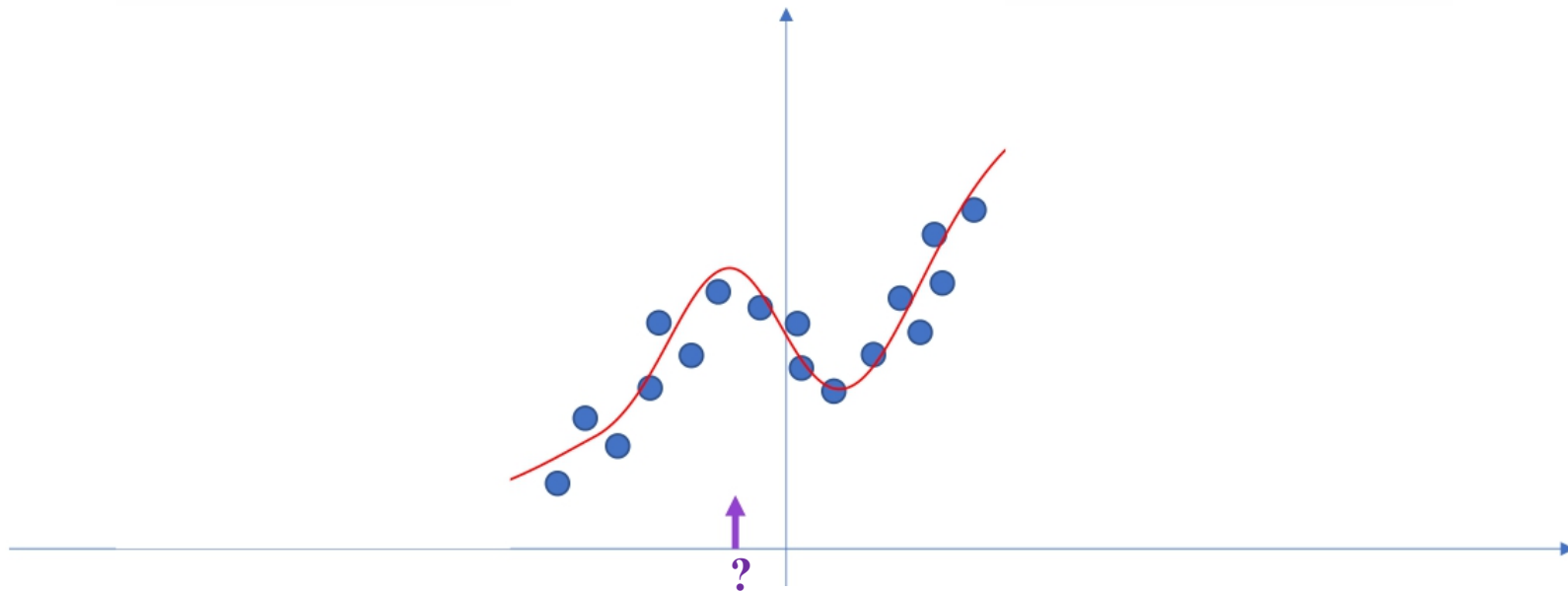
Neural Networks as Function Approximators

Neural networks are **excellent** function approximators



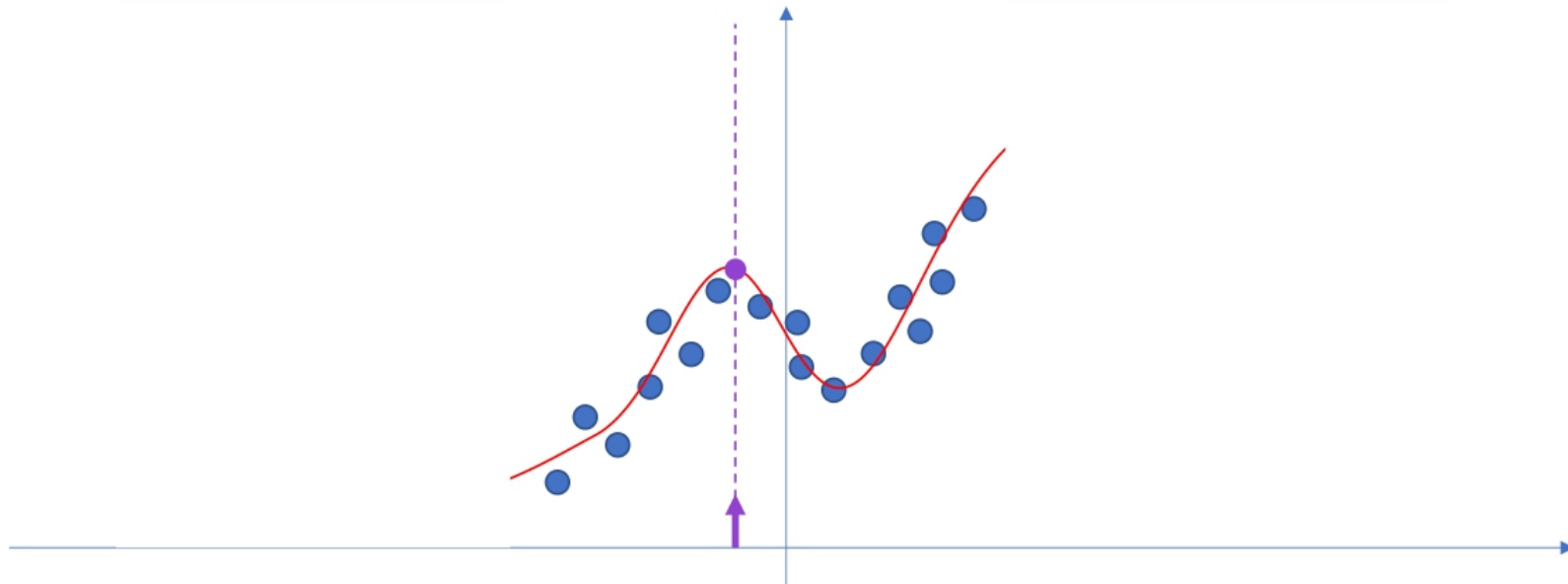
Neural Networks as Function Approximators

Neural networks are **excellent** function approximators



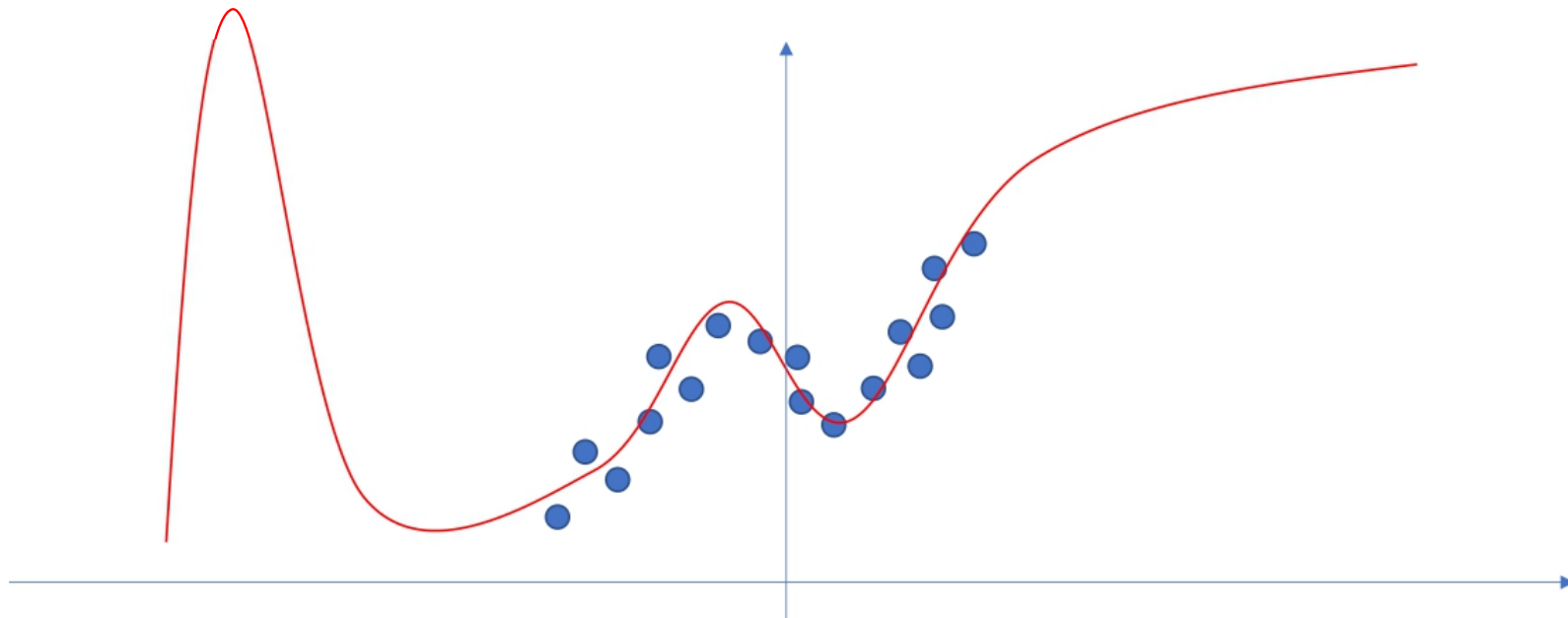
Neural Networks as Function Approximators

Neural networks are **excellent** function approximators



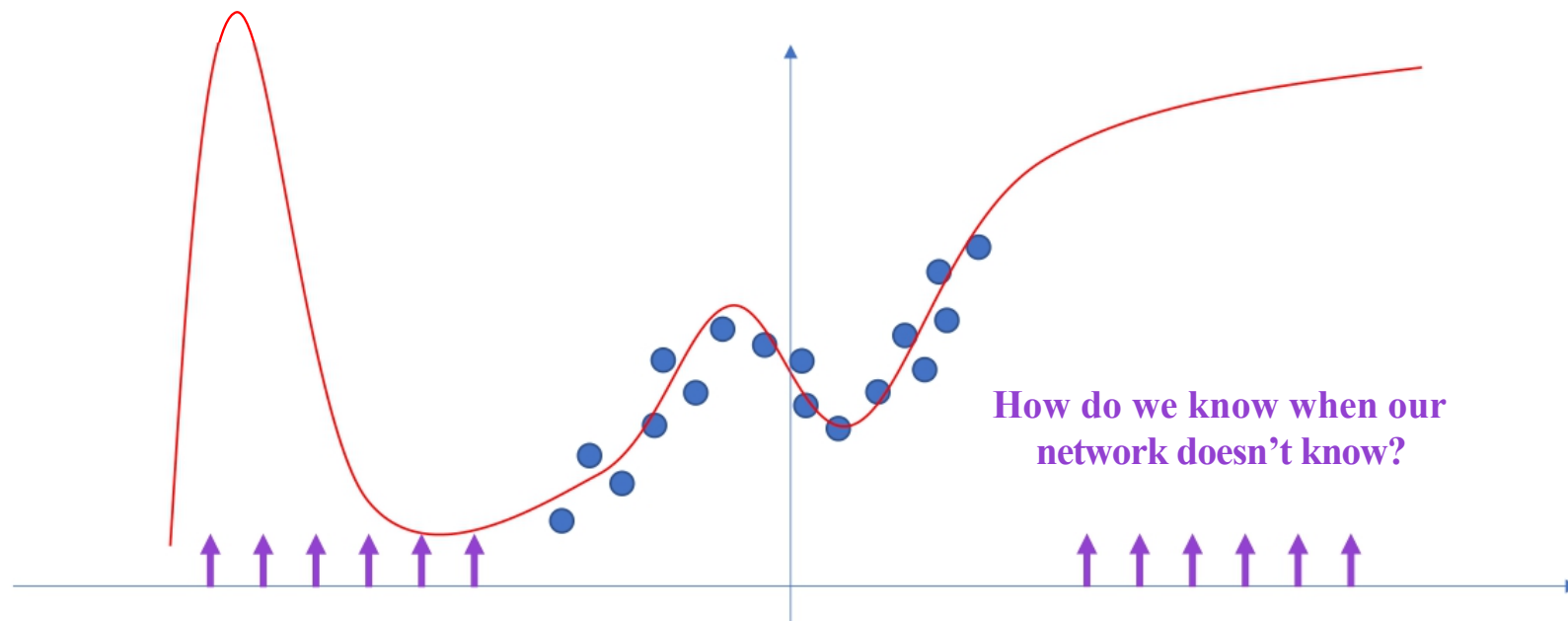
Neural Networks as Function Approximators

Neural networks are **excellent** function approximators

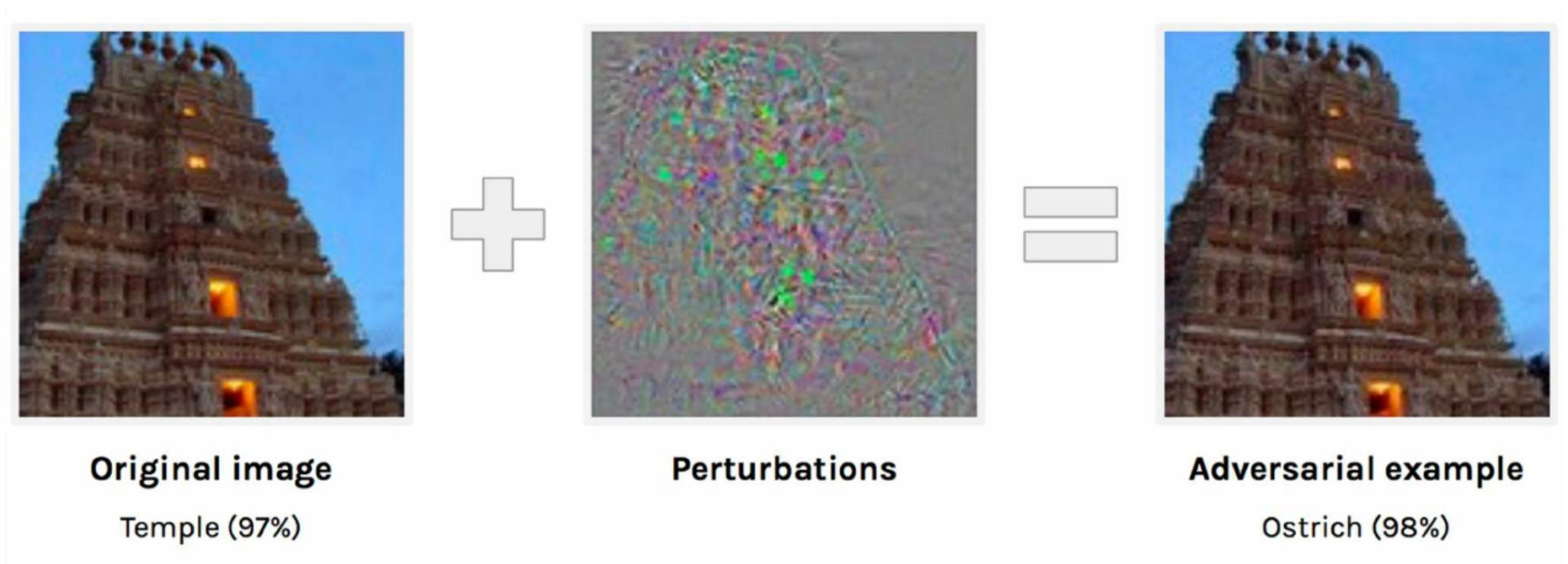


Neural Networks as Function Approximators

Neural networks are **excellent** function approximators
...when they have training data

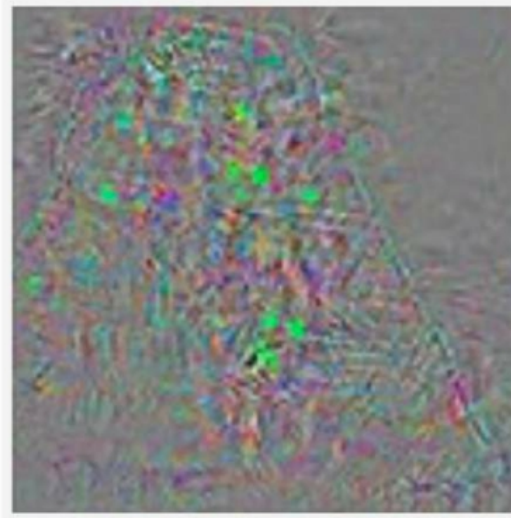


Adversarial Attacks on Neural Networks



Despois. “Adversarial examples and their implications” (2017).

Adversarial Attacks on Neural Networks



Perturbations

Adversarial Attacks on Neural Networks

Remember:

We train our networks with gradient descent

$$\theta \leftarrow \theta - \eta \frac{\partial J(\theta, x, y)}{\partial \theta}$$

“How does a small change in weights decrease our loss”

Adversarial Attacks on Neural Networks

Remember:

We train our networks with gradient descent

$$\theta \leftarrow \theta - \eta \frac{\partial J(\theta, x, y)}{\partial \theta}$$

“How does a small change in weights decrease our loss”

Adversarial Attacks on Neural Networks

Remember:

We train our networks with gradient descent

$$\theta \leftarrow \theta - \eta \frac{\partial J(\theta, x, y)}{\partial \theta}$$

Fix your image x ,
and true label y

“How does a small change in weights decrease our loss”

Adversarial Attacks on Neural Networks

Adversarial Image:

Modify image to increase error

$$x \leftarrow x + \eta \frac{\partial J(\theta, x, y)}{\partial x}$$

“How does a small change in the input increase our loss”

Goodfellow et al. *NIPS* (2014)

Adversarial Attacks on Neural Networks

Adversarial Image:

Modify image to increase error

$$x \leftarrow x + \eta \frac{\partial J(\theta, x, y)}{\partial x}$$

“How does a small change in the input increase our loss”

Adversarial Attacks on Neural Networks

Adversarial Image:

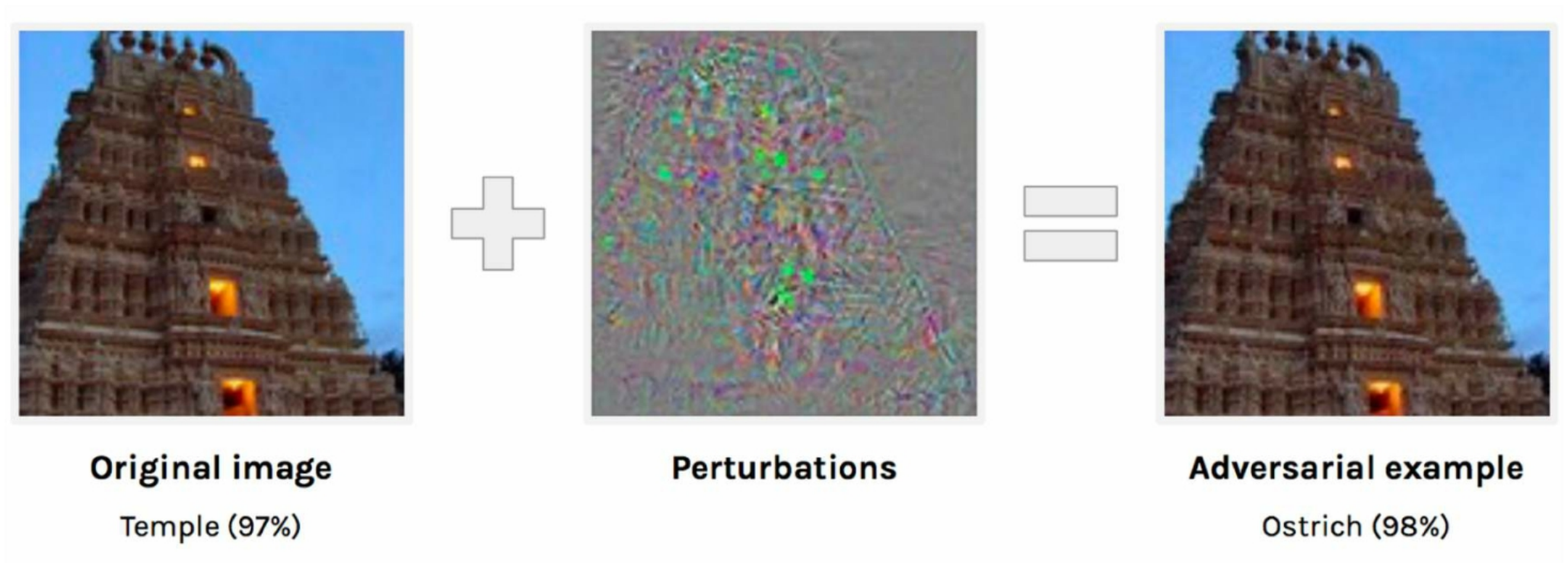
Modify image to increase error

$$x \leftarrow x + \eta \frac{\partial J(\theta, x, y)}{\partial x}$$

Fix your weights θ ,
and true label y

“How does a small change in the input increase our loss”

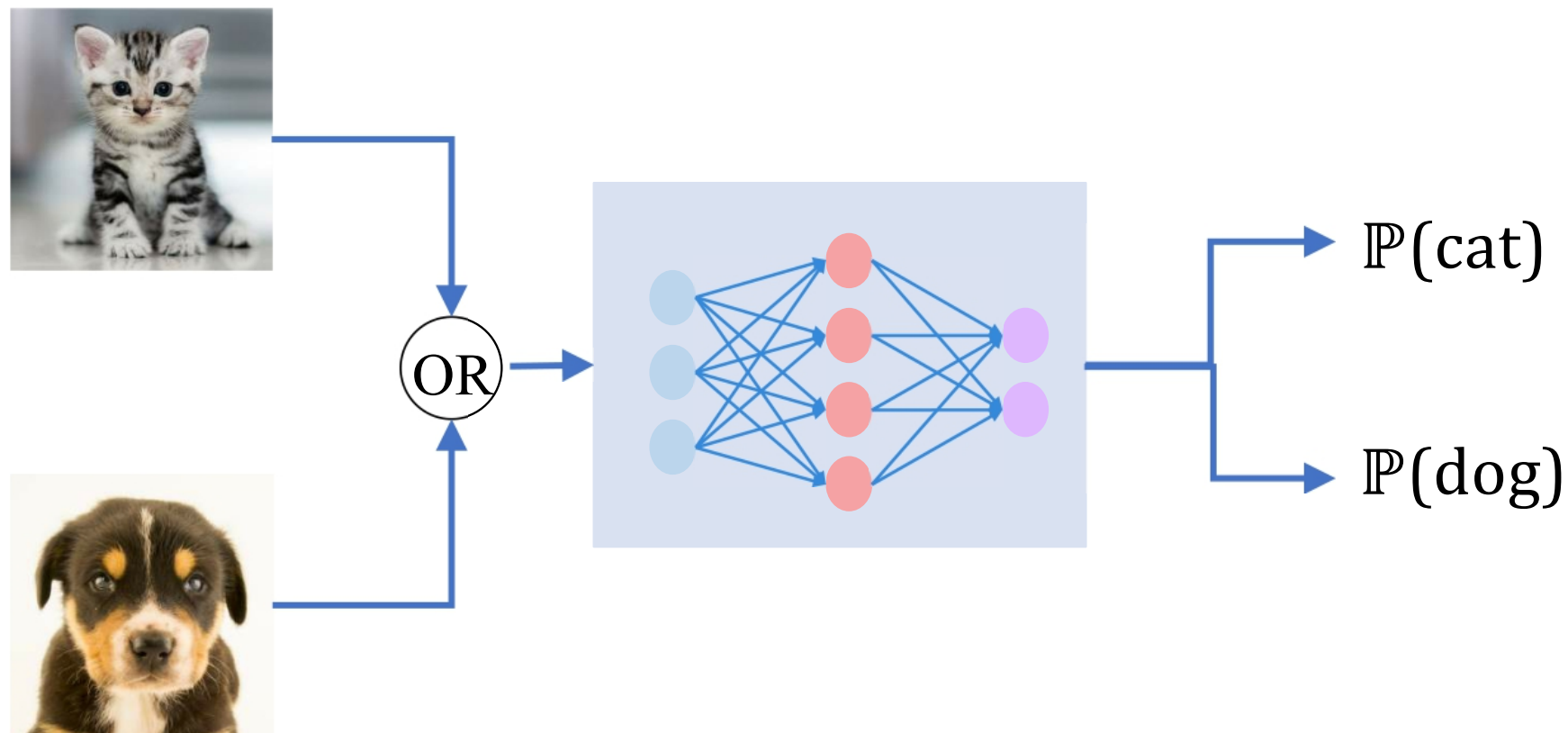
Adversarial Attacks on Neural Networks



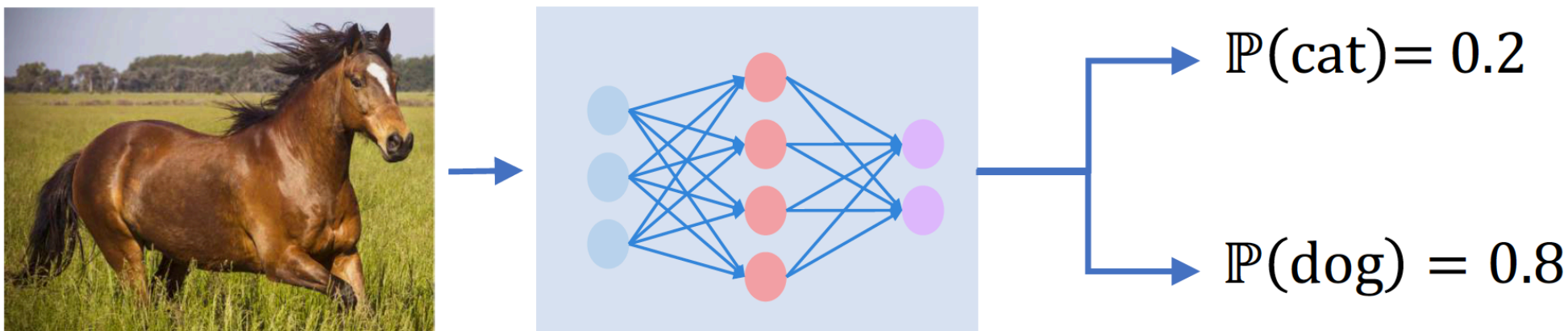
Despois. “Adversarial examples and their implications” (2017).

Bayesian Deep Learning

Why Care About Uncertainty?



Why Care About Uncertainty?



Remember: $\mathbb{P}(\text{cat}) + \mathbb{P}(\text{dog}) = 1$

Bayesian Deep Learning for Uncertainty

Network tries to learn output, $\mathbf{Y}$, directly from raw data, $\mathbf{X}$

Find mapping, f , parameterized by weights $\boldsymbol{\theta}$ such that
$$\min \mathcal{L}(\mathbf{Y}, f(\mathbf{X}; \boldsymbol{\theta}))$$

Bayesian neural networks aim to learn a posterior over weights,
$$\mathbb{P}(\boldsymbol{\theta}|\mathbf{X}, \mathbf{Y}):$$

$$\mathbb{P}(\boldsymbol{\theta}|\mathbf{X}, \mathbf{Y}) = \frac{\mathbb{P}(\mathbf{Y}|\mathbf{X}, \boldsymbol{\theta})\mathbb{P}(\boldsymbol{\theta})}{\mathbb{P}(\mathbf{Y}|\mathbf{X})}$$

Bayesian Deep Learning for Uncertainty

Network tries to learn output, $\mathbf{Y}$, directly from raw data, $\mathbf{X}$

Find mapping, f , parameterized by weights $\boldsymbol{\theta}$ such that
$$\min \mathcal{L}(\mathbf{Y}, f(\mathbf{X}; \boldsymbol{\theta}))$$

Bayesian neural networks aim to learn a posterior over weights,
$$\mathbb{P}(\boldsymbol{\theta}|\mathbf{X}, \mathbf{Y}):$$

Intractable!
$$\mathbb{P}(\boldsymbol{\theta}|\mathbf{X}, \mathbf{Y}) = \frac{\mathbb{P}(\mathbf{Y}|\mathbf{X}, \boldsymbol{\theta})\mathbb{P}(\boldsymbol{\theta})}{\mathbb{P}(\mathbf{Y}|\mathbf{X})}$$

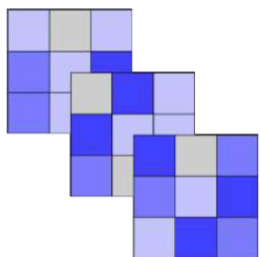
Elementwise Dropout for Uncertainty

Evaluate T stochastic forward passes through the network $\{\theta_t\}_{t=1}^T$

Dropout as a form of stochastic sampling $z_{w,t} \sim \text{Bernoulli}(p) \quad \forall w \in \theta$

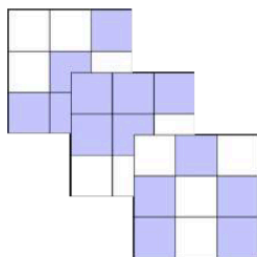
Unregularized Kernel

θ



Bernoulli Dropout

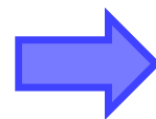
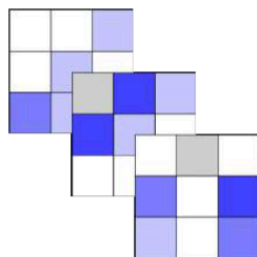
$z_{\theta,t}$



=

Stochastic Sampled

θ_t



$$\mathbb{E}(\hat{Y}|\mathbf{X}) = \frac{1}{T} \sum_{t=1}^T f(\mathbf{X}|\theta_t)$$

$$\text{Var}(\hat{Y}|\mathbf{X}) = \frac{1}{T} \sum_{t=1}^T f(\mathbf{X})^2 - \mathbb{E}(\hat{Y}|\mathbf{X})^2$$

Gal and Ghahramani, *ICML*, 2016.

Amini, Soleimany, et al., *NIPS Workshop on Bayesian Deep Learning*, 2017.



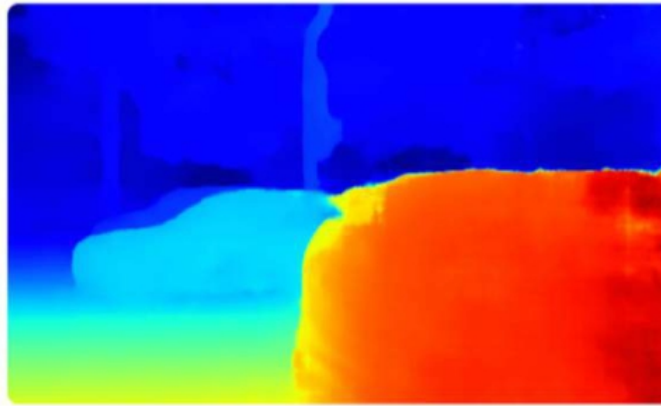
Massachusetts
Institute of
Technology

6.S191 Introduction to Deep Learning
introtodeeplearning.com

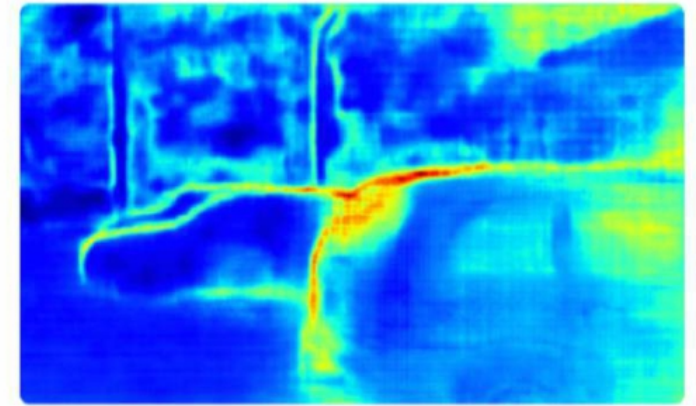
Model Uncertainty Application



Input image



Predicted Depth



Model Uncertainty

Kendall, Gal, *NIPS*, 2017.

Neural Network Limitations...

- Very **data hungry** (eg. often millions of examples)
- **Computationally intensive** to train and deploy (tractably requires GPUs)
- Easily fooled by **adversarial examples**
- Can be subject to **algorithmic bias**
- Poor at **representing uncertainty** (how do you know what the model knows?)
- Uninterpretable **black boxes**, difficult to trust
- **Finicky to optimize**: non-convex, choice of architecture, learning parameters
- Often require **expert knowledge** to design, fine tune architectures

The End