Deep Learning

Next Generation Neural Networks

Tuesday 14th January

Dr. Nicholas Cummins

Topics to be covered

- Spiking Neural Networks
- Neural Turing Machines
- Progressive Neural Networks
- Residual Networks
- Squeeze Nets
- Bayesian Neural Networks

Spiking Neural Networks

- https://towardsdatascience.com/spiking-neural-networks-the-next-generation-of-machine-learning-84e167f4eb2b
- https://medium.com/@amissinato/neuromorphic-computers-and-spiking-neural-networks-the-new-generation-of-machine-learning-8ccd39c29956
- https://arxiv.org/pdf/1804.08150.pdf
- https://www.frontiersin.org/articles/10.3389/fnins.2018.00774/full

What is it?

Biologically realistic deep neural network

Core Idea

Event Based Input

SNNs processes time information depending on the events

Neurons have a binary activation function

How does it work?

- Often sparsely connected NN
- Activation Function based on thresholds
- Learning is based on spike timing between pairs of directly connected neurons
- Through training threshold is modified

Uses Cases:

Pattern recognition (medical diagnosis)

Image and audio processing

Handwritten digit recognition

Etc.

Advantages

Hardware and energy friendly

Disadvantages

Gradient based optimisation techniques can’t be applied, because activation functions are non-derivative

Inefficient training algorithms lead to longer training times

Neural Turing Machines

- https://distill.pub/2016/augmented-rnns/#neural-turing-machines
- https://medium.com/towards-artificial-intelligence/neural-turing-machines-eaada7e7a6cc
- https://arxiv.org/pdf/1410.5401.pdf
- https://arxiv.org/ftp/arxiv/papers/1904/1904.05061.pdf

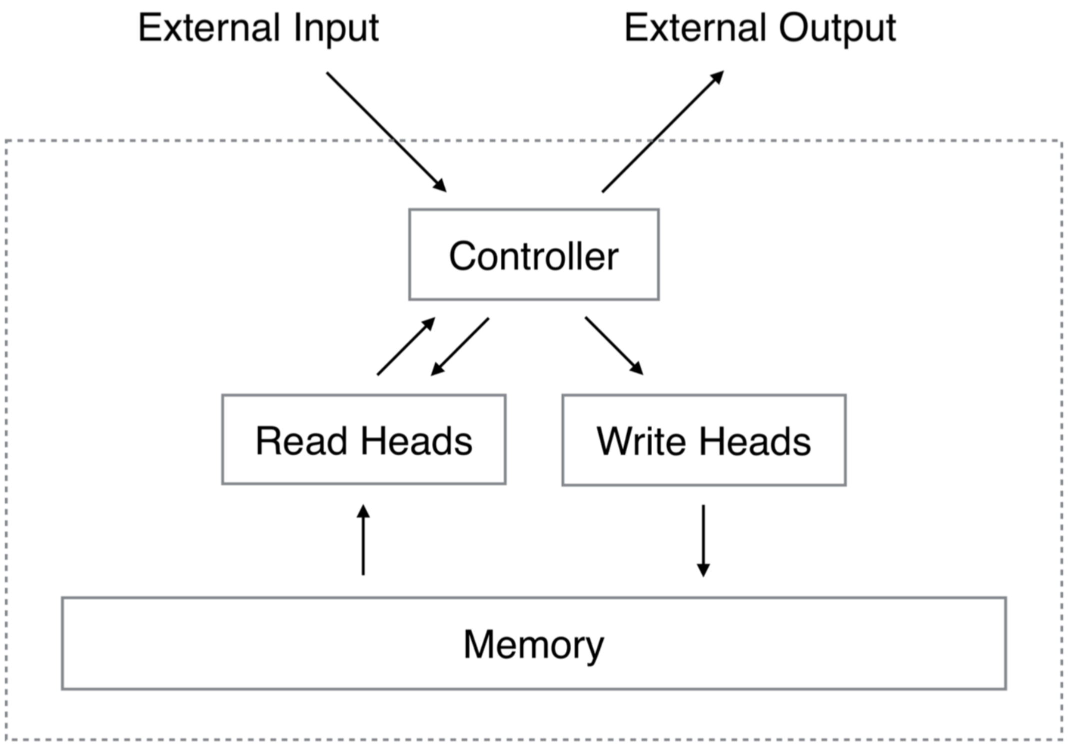
What is it?

A neural network attached to a memory matrix
utilizing attention mechanisms to read and write data.

Core Idea:

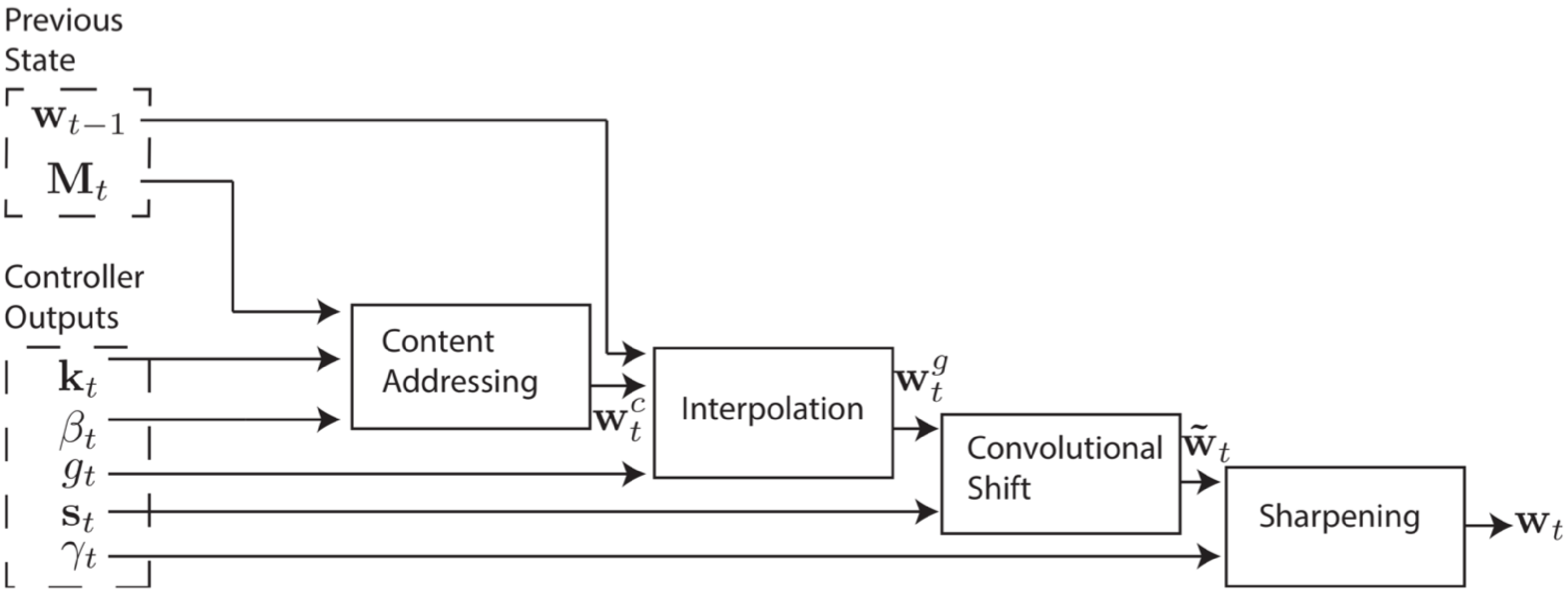
Solve tasks, that require remembering long sequences

How does it work?



Universität Augsburg University

How does it work?



Use Cases:

- Sequence Copying Tasks
- Associative Recall Tasks
- Sorting

Likely to outperform conventional architectures in tasks that are fundamentally algorithmic that cannot be learned by finding a decision boundary

Advantages

- Fewer parameters required for a certain set of problems (compared to LSTM)
- Reading/Writing is visualizable

Disadvantages

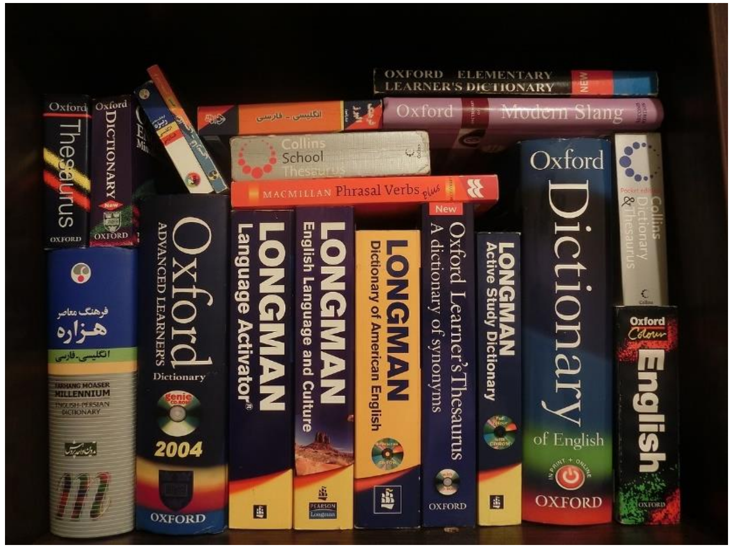
- Only good for a certain set of tasks – outperformed in others

Progressive Neural Networks

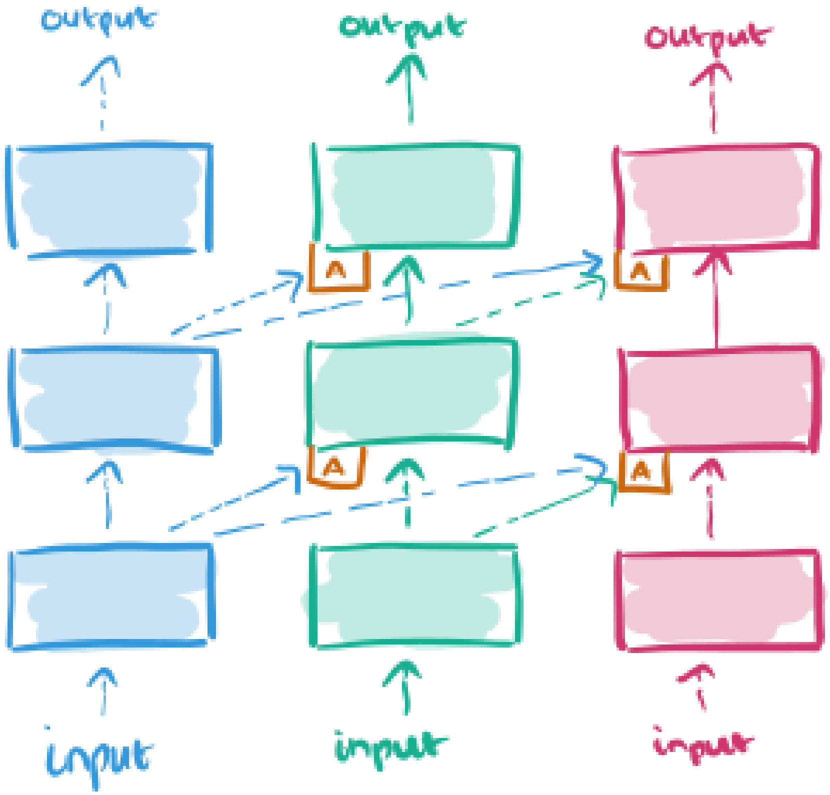
- https://towardsdatascience.com/what-are-progressive-neural-networks-b7b4f8de603
- https://blog.acolyer.org/2016/10/11/progressive-neural-networks/
- https://arxiv.org/pdf/1606.04671.pdf

These modelling decisions are informed by our desire to:

- solve K independent tasks at the end of training
- accelerate learning via transfer when possible
- avoid catastrophic forgetting



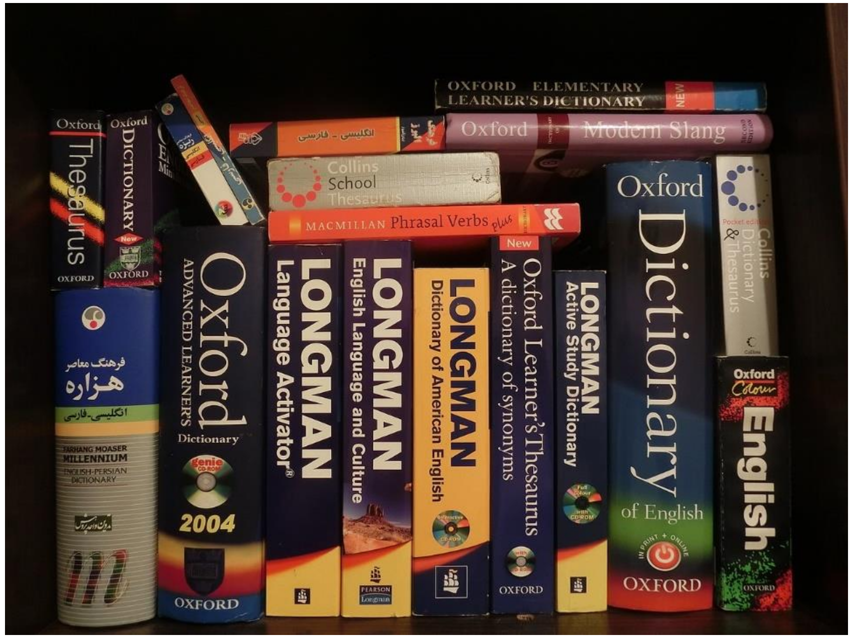
Lateral Transfer



- Learn 1. task in 1. column (blue)
- Freeze 1. column weights
- The outputs of layer *l* in task 1 becomes additional inputs to layer *l+1* in the new column

Use Case

- learn multiple tasks, in sequence
- enabling transfer
- being immune to catastrophic forgetting



Pro Cons

Advantages

High positive transfer

Disadvantages

Immunity to catastrophic forgetting prevents any ‘skills’ a network learns on subsequent tasks being used to improve performance on previous tasks.

Aims

Perform any task based on previous knowledge based on other tasks

Residual Networks

- https://towardsdatascience.com/an-overview-of-resnet-and-its-variants-5281e2f56035
- https://medium.com/analytics-vidhya/understanding-and-implementation-of-residual-networks-resnets-b80f9a507b9c
- https://arxiv.org/pdf/1512.03385.pdf
- https://arxiv.org/pdf/1605.06431.pdf

What is it?

Residual Networks

Core Idea

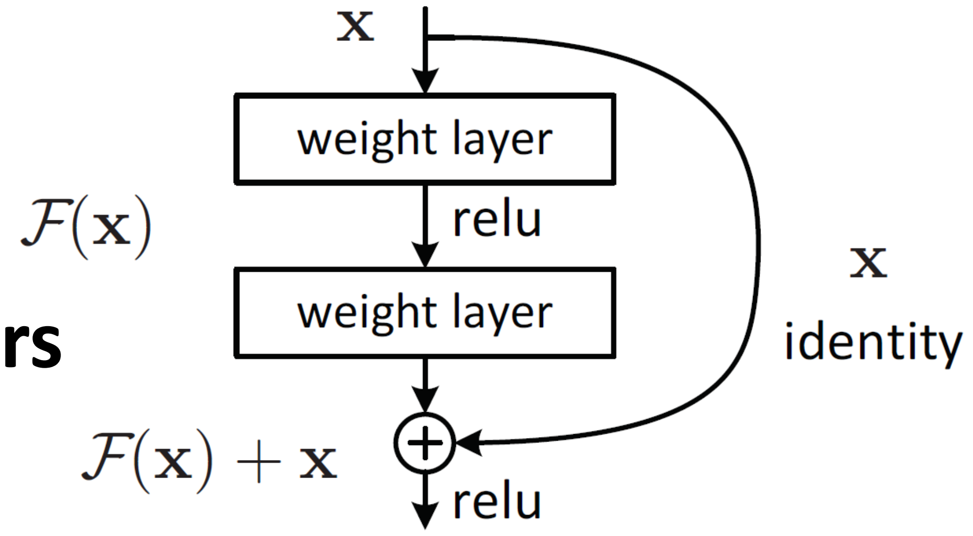
Deeper Networks

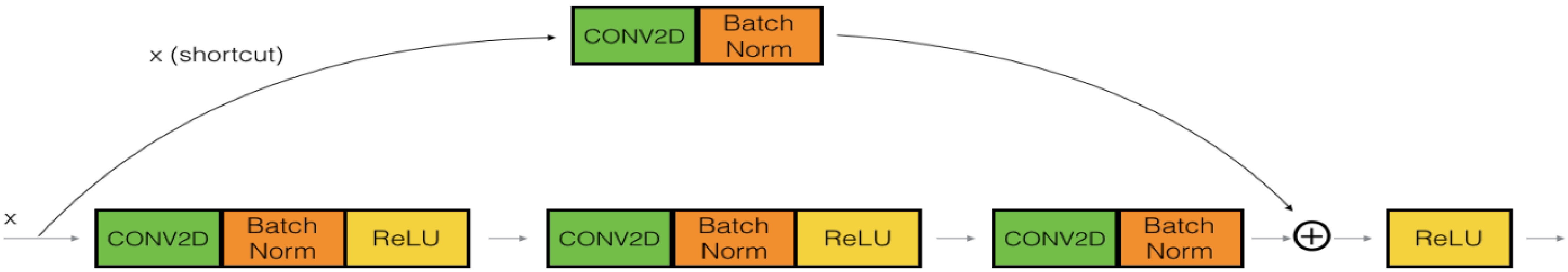
➔Vanishing gradient

➔Skipping layers

How does it work?

- **Adding identity from previous layers**
- **Weight = 0 ➔ Unused layer**
- **Convolutional layers to fit dimensions**





Uses Cases:

- Image classification (1000 classes)
- Deep Neural Networks

Advantages

- Learning with many layers
- Self-optimizing performance by skipping layers

Disadvantages

- Does not resolve vanishing gradient

SqueezeNet

- https://towardsdatascience.com/review-squeezenet-image-classification-e7414825581a
- https://medium.com/@smallfishbigsea/notes-of-squeezenet-4137d51feef4
- https://arxiv.org/pdf/1602.07360.pdf
- https://arxiv.org/pdf/1803.10615.pdf

What is it?

- Novel Convolutional Deep Neural Network Architecture

Core Idea

- Reduce parameters and maintain good accuracy (like AlexNet)

How does it work?

- Replace 3x3 filters with 1x1 filters
  -> 1/9 of computation
- Decrease the number of input channels to 3x3 filters by using 1x1 filters as bottleneck layers
- Downsample late in the network to keep a big feature map

- Firemodule (squeeze / bottleneck and expand)

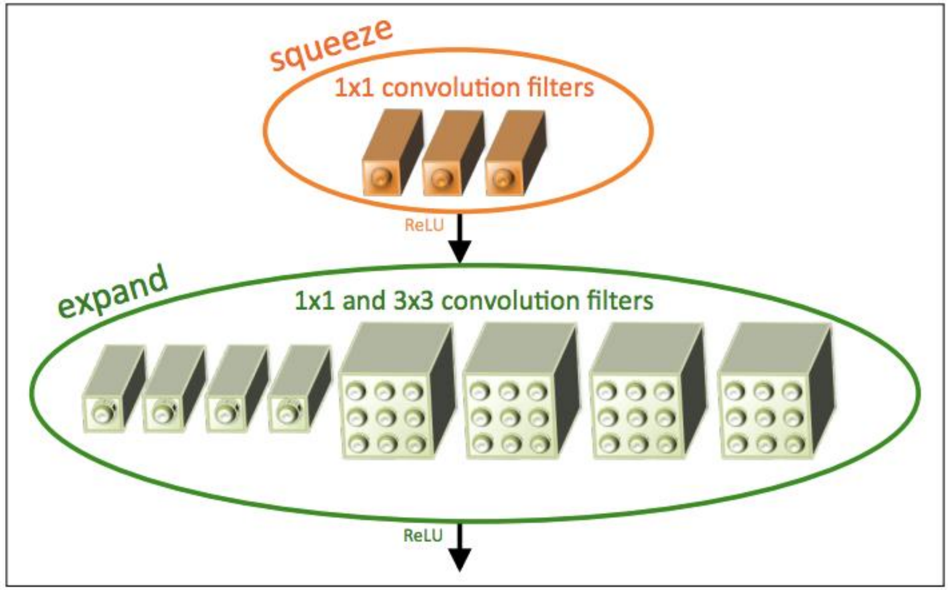


Figure 1: Microarchitectural view: Organization of convolution filters in the **Fire module**. In this example, $s_{1x1} = 3$, $e_{1x1} = 4$, and $e_{3x3} = 4$. We illustrate the convolution filters but not the activations.

Uses Cases:

- Image Classification
- Fine-grained object recognition
- Logo identification in images
- Generating sentences about images

Advantages

- More efficient distributed training
- Less overhead when exporting new models to clients
- Less memory / bandwidth
- Embedded deployment on small hardware resources

Disadvantages

- No guarantees that it will work for every classification problem

Bayesian Neural Networks

- https://towardsdatascience.com/bayesian-neural-networks-in-10-mins-in-tfp-c735ec99384f
- https://towardsdatascience.com/making-your-neural-network-say-i-dont-know-bayesian-nns-using-pyro-and-pytorch-b1c24e6ab8cd
- https://arxiv.org/ftp/arxiv/papers/1801/1801.07710.pdf

What is it?

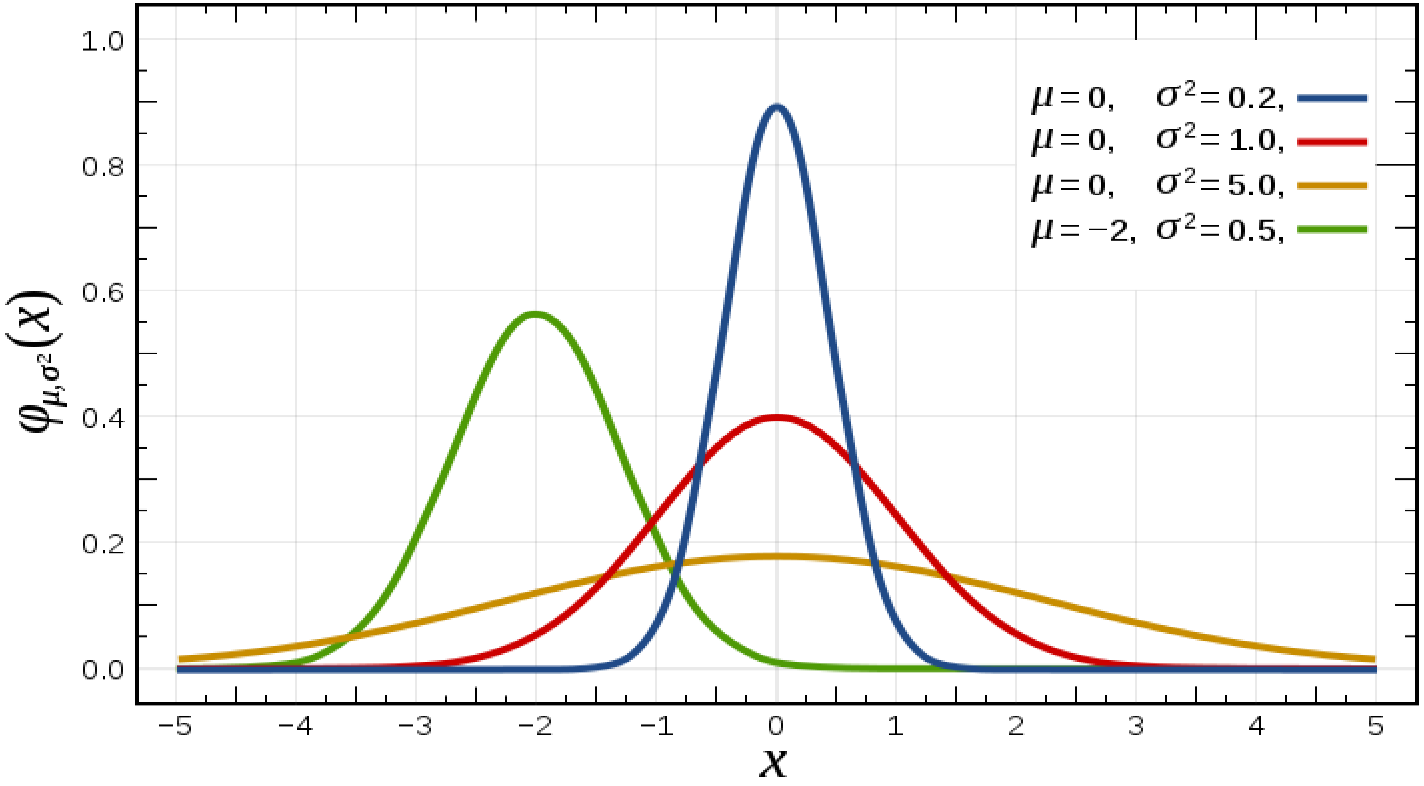
BNNs are FF-Neural Nets where the weights and biases are expressed by distributions instead of numbers

Core Idea

Weights are sampled. → Different predictions for multiple passes (for on input)

How does it work?

Learn the parameters of the distributions instead of single scalar values. This can be done by gradient based optimizers.



Source:
https://en.wikipedia.org/wiki/File:Normal_Distribution_PDF.svg

Uses Cases:

Classification: Inputs that are alien to all classes can be passed multiple times. This way we can measure the confidence.

High var. in the outputs → Image classified as unknown.

Low var. in the outputs → Image is classified as the most likely class.

Advantages

We can identify data that doesn't belong to any class.

Disadvantages

We will have to do multiple passes (computationally more expensive)

Sources (like on the Slides):

- https://towardsdatascience.com/bayesian-neural-networks-in-10-mins-in-tfp-c735ec99384f
- https://towardsdatascience.com/making-your-neural-network-say-i-dont-know-bayesian-nns-using-pyro-and-pytorch-b1c24e6ab8cd
- https://arxiv.org/ftp/arxiv/papers/1801/1801.07710.pdf