Introduction to deep learning

Machine learning & neural networks

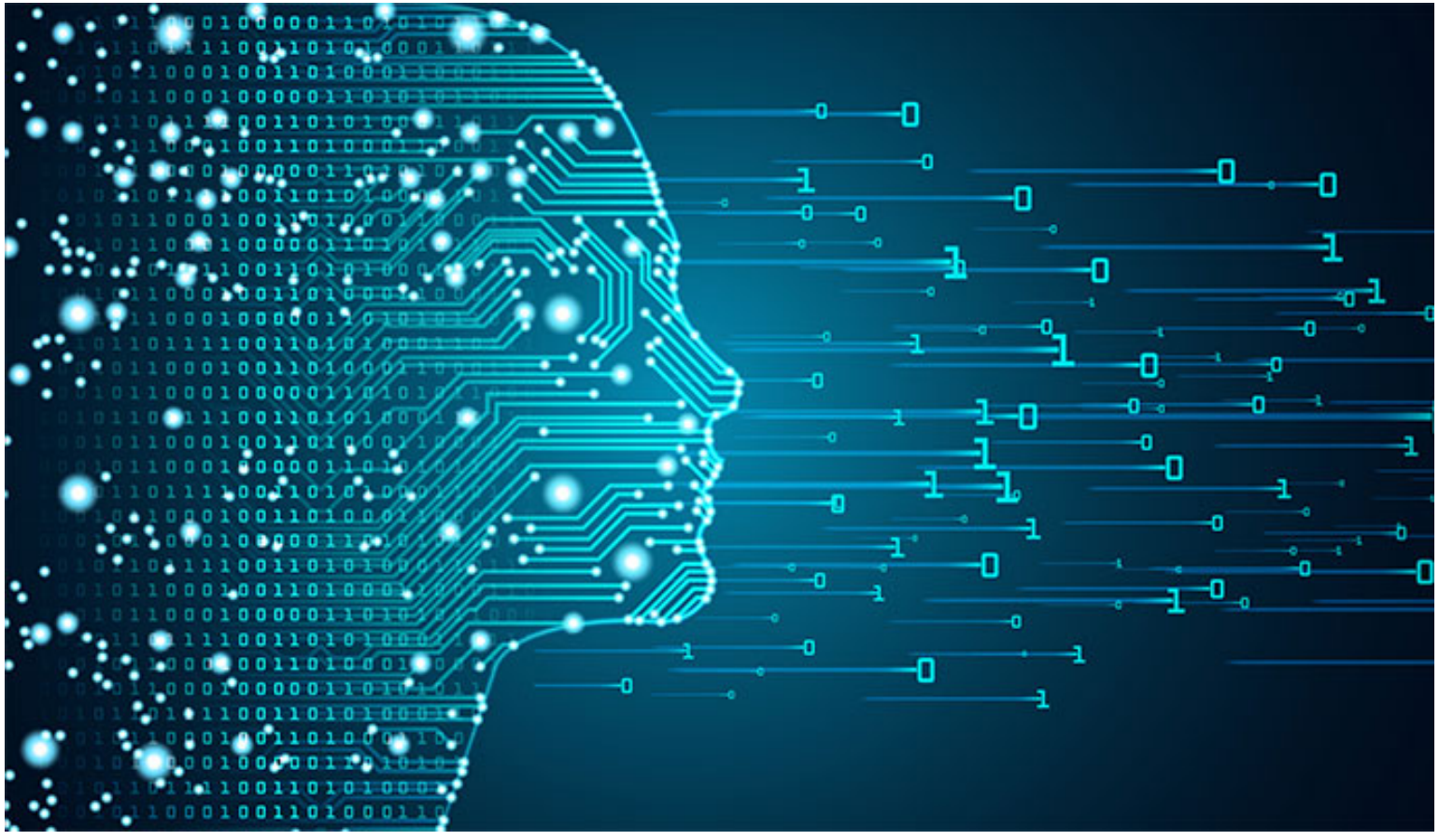
Anne Helby Petersen

Moving on to deep learning

Q: What is deep learning?

Moving on to deep learning

Q: What is deep learning?



A: Deep learning = neural network with more than 1 hidden layer.

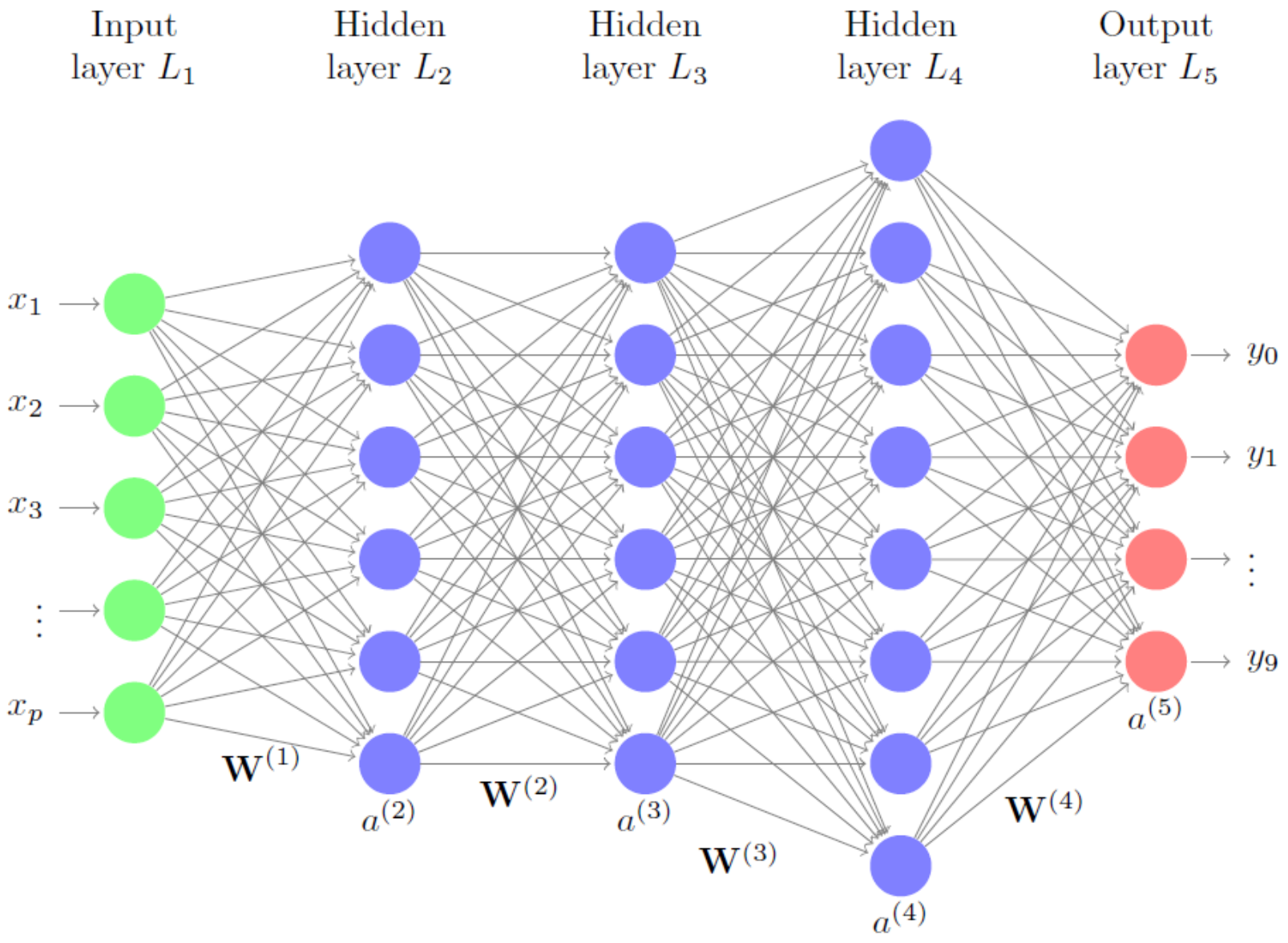
Moving on to deep learning

Q: What is deep learning?

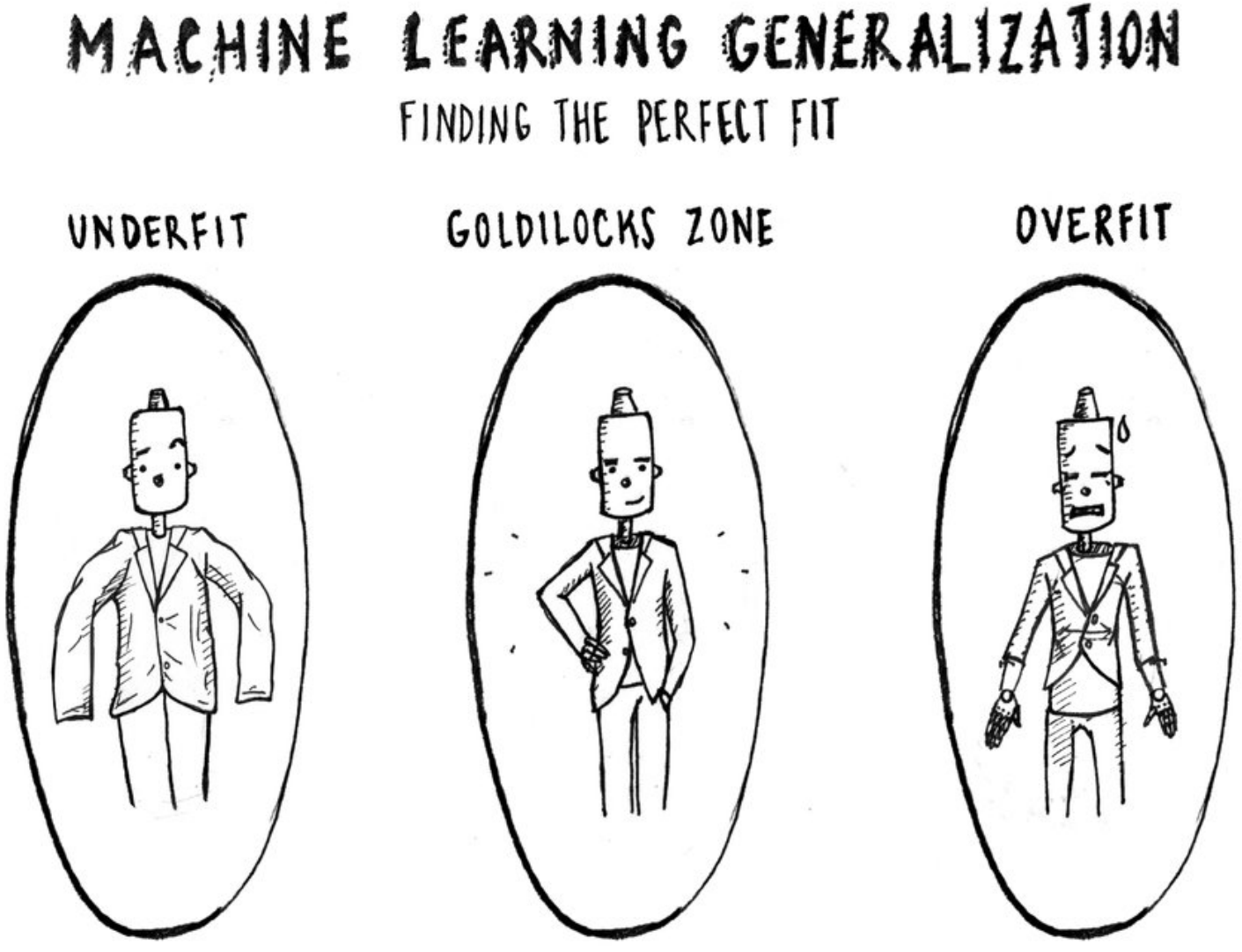
A: Deep learning = neural network with more than 1 hidden layer.

$\Rightarrow$ We did it already.

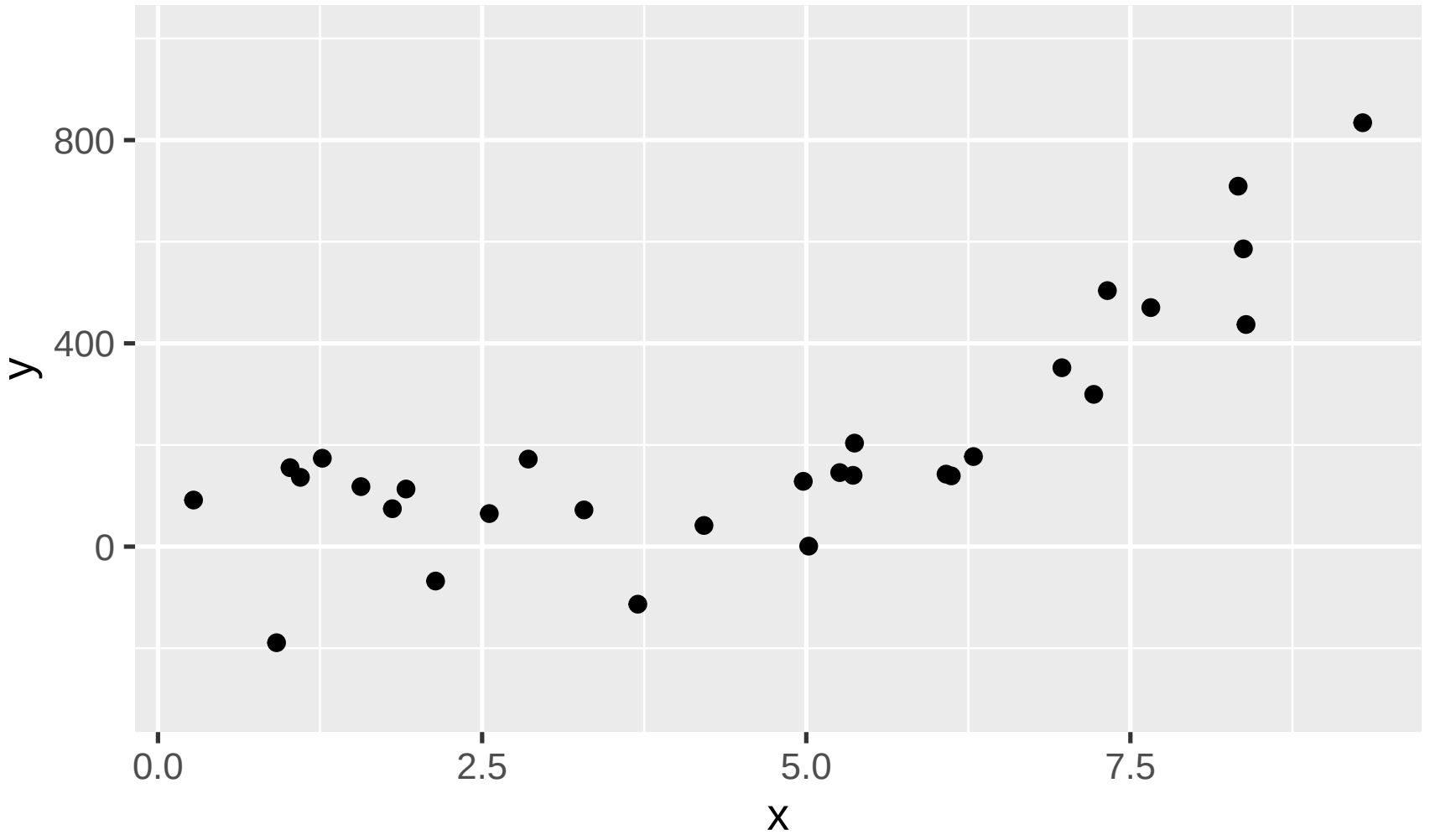
A deep neural network



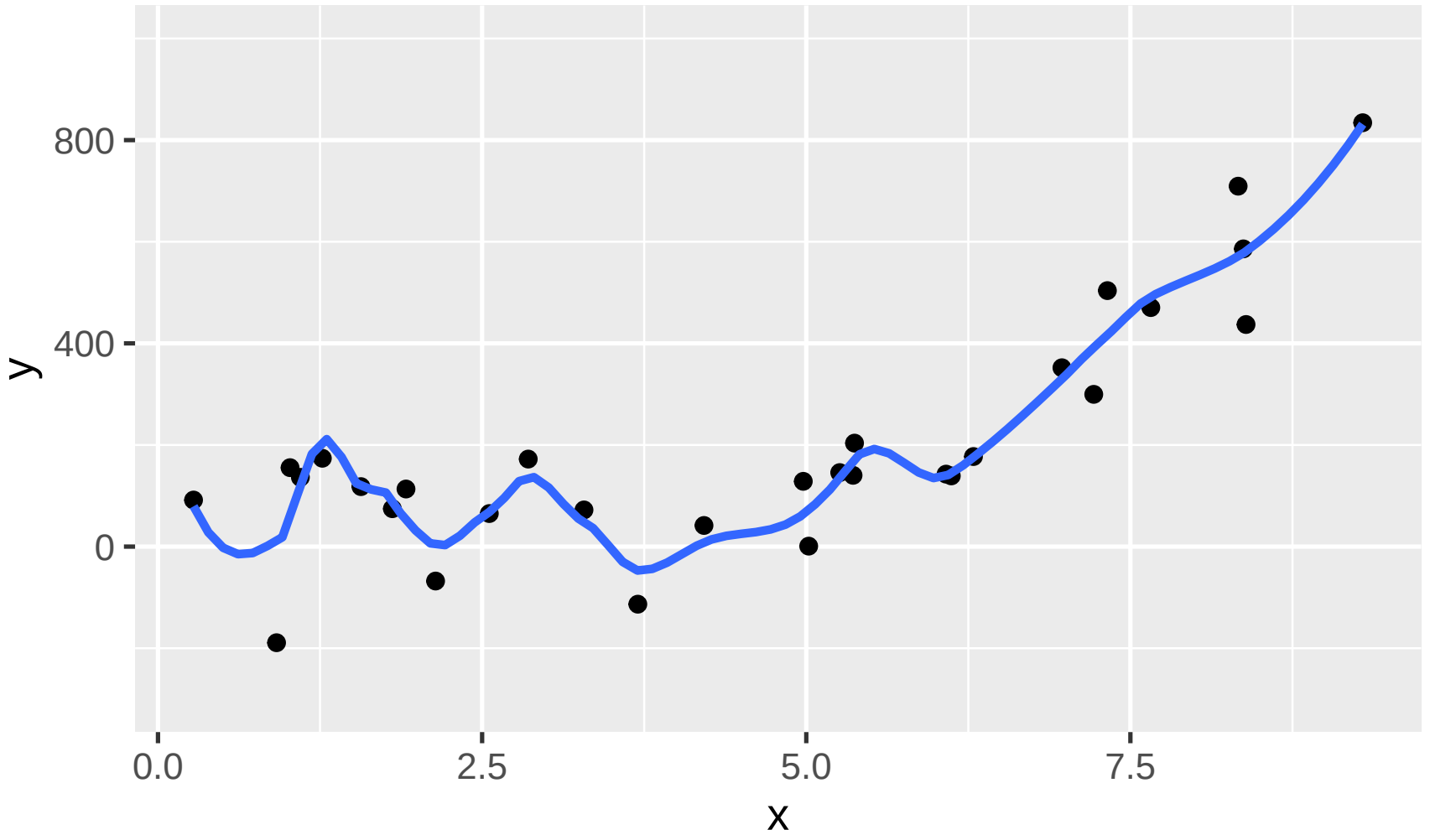
Overfitting and underfitting



Overfitting a curve



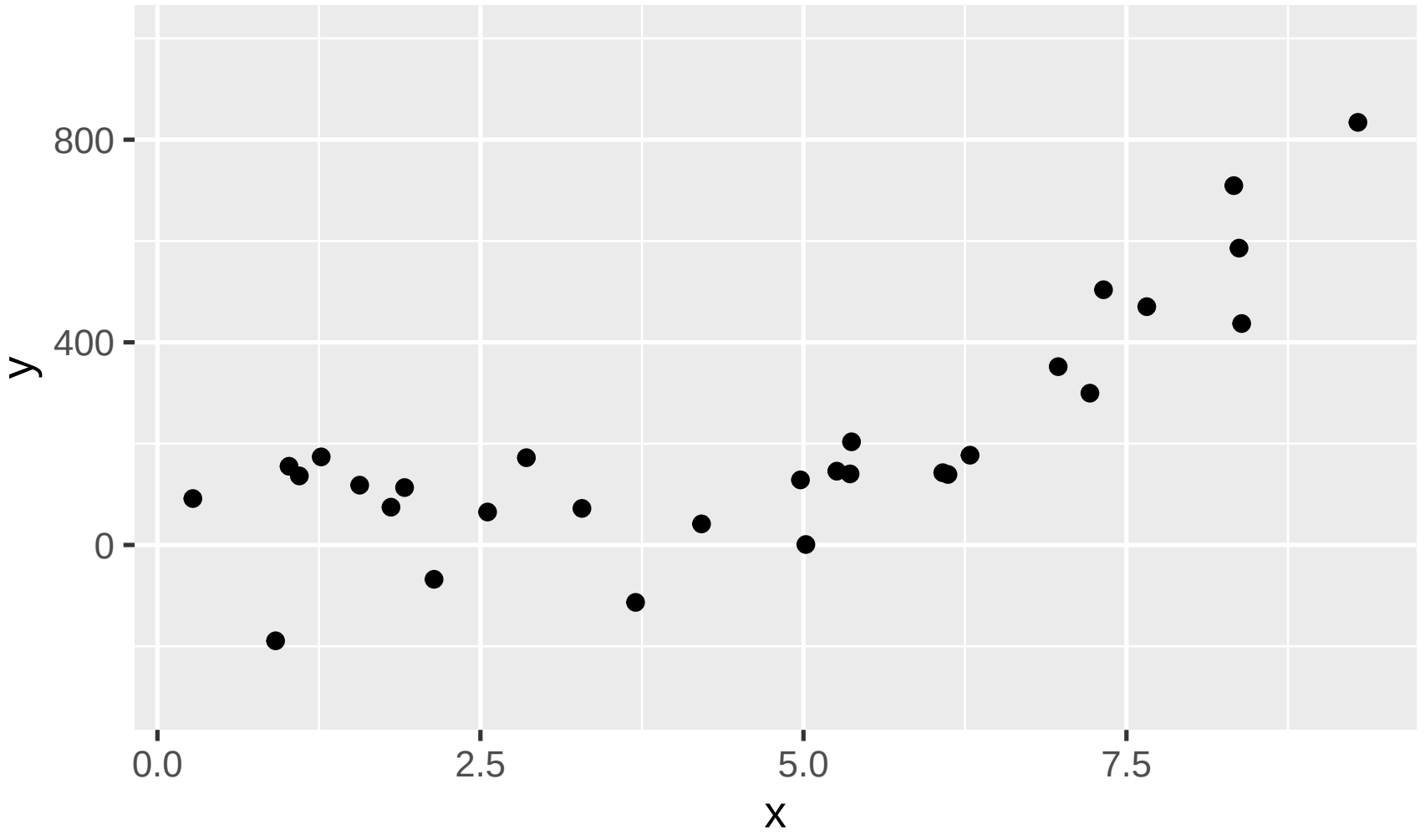
Overfitting a curve



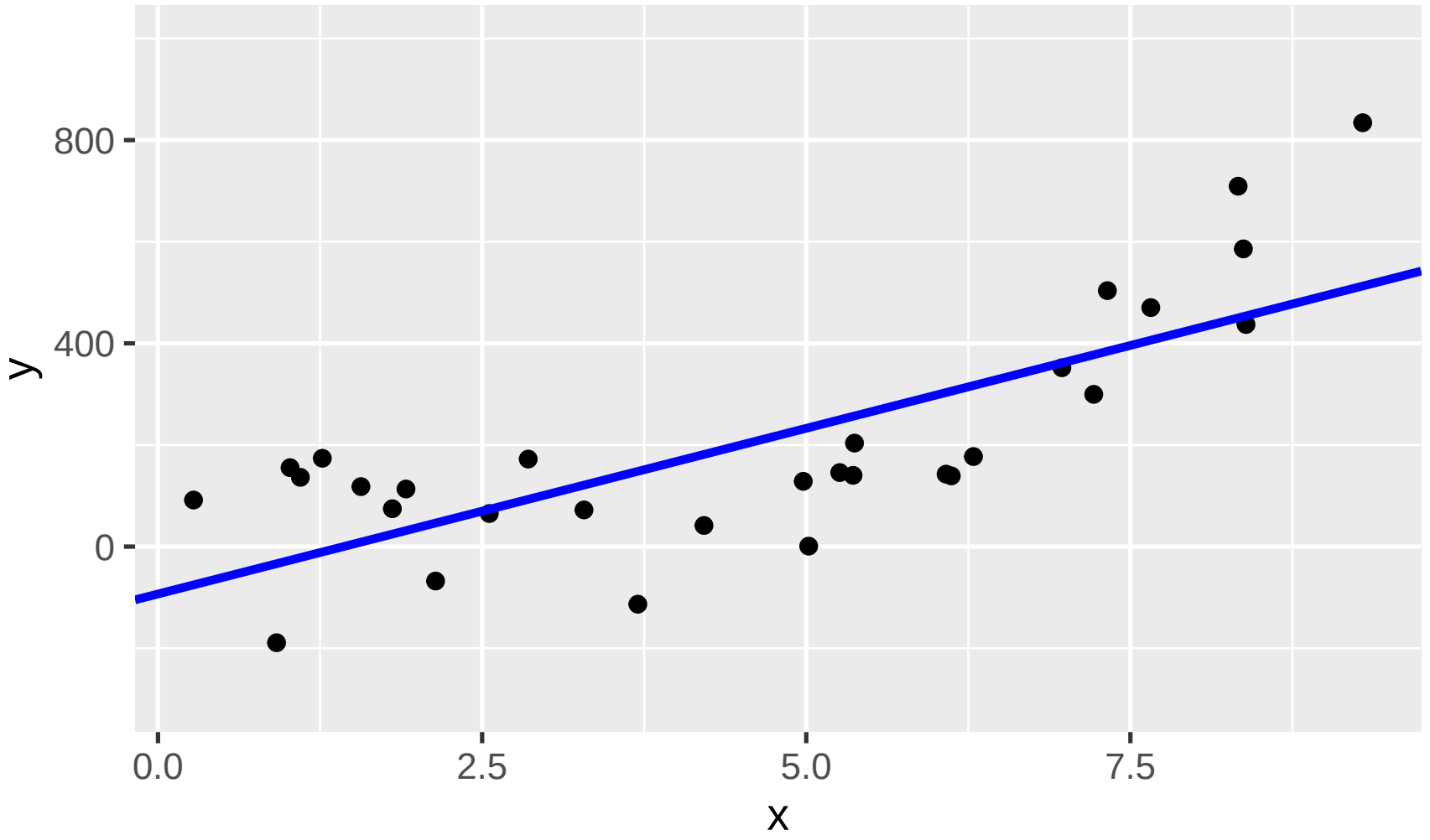
Overfitting a machine

- Building a machine that knows the training data *too well*.
- Interpreting noise in training data as actual information.
- Machine performs well on training data.
- **Machine performs bad on test data.**

Underfitting a curve



Underfitting a curve



Underfitting a machine

- Building a machine that doesn't really know the data.
- Overlooking important relationships in the data.
- Machine performs bad on training data.
- **Machine performs equally bad on test data.**

Getting the right balance

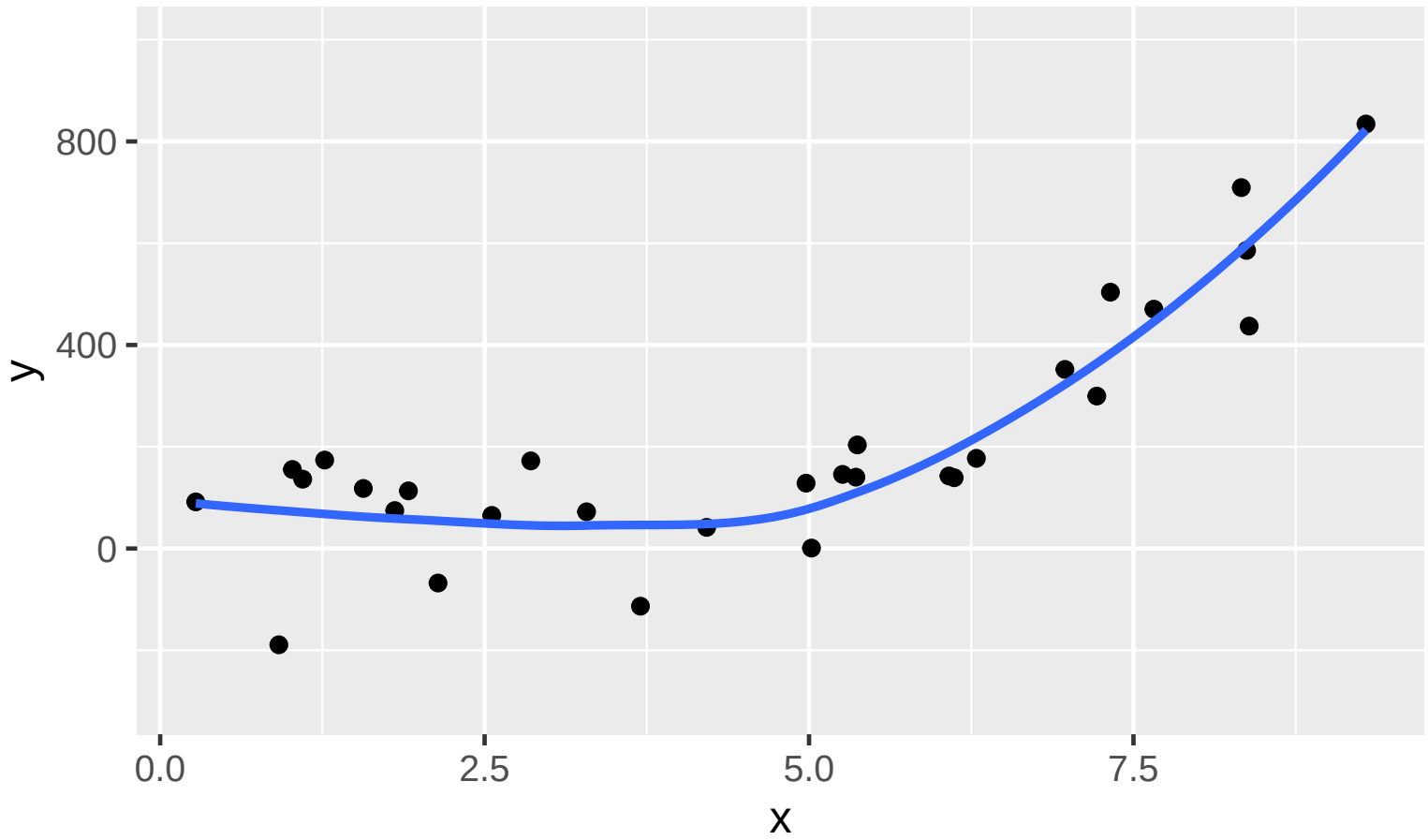
Two strategies in NNs:

1. Regularization (LASSO/ridge)
2. Drop out

Getting the right balance

Two strategies in NNs:

1. Regularization (LASSO/ridge)
2. Drop out



Regularization

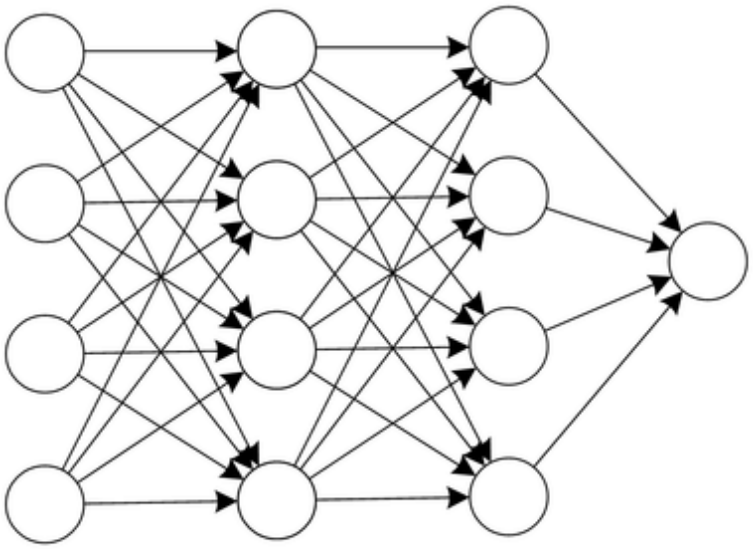
Idea: Shrink some of the weights to make sure not "too much" information is being used from training data.

Two types of regularization:

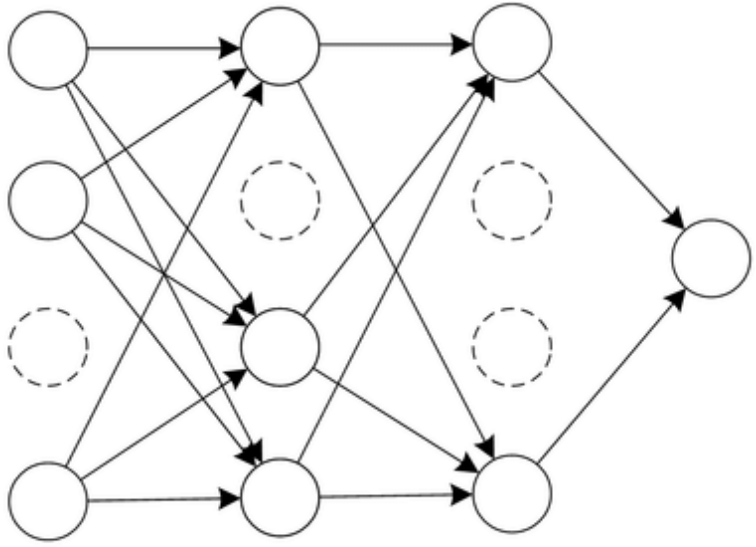
1. LASSO: Shrink weights using linear penalization $\Rightarrow$ some weights become zero
2. Ridge: Shrink weights using quadratic penalization $\Rightarrow$ no weights become zero (but they do become small)

Dropout

Idea: Set all weights from some randomly chosen nodes to zero (equivalently: drop random nodes from the network).



(a) Standard Neural Network



(b) Network after Dropout

Dropout intuition

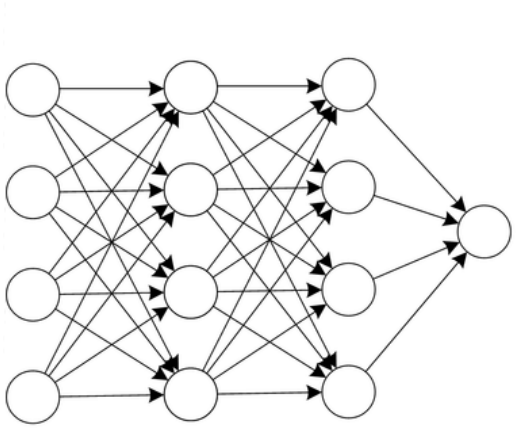
"Imagine a scenario, in a classroom, a teacher asks some questions but always same two kids are answering, immediately. Now, the teacher asks them to stay quiet for some time and let other pupils participate. This way other students get to learn better. Maybe they answer wrong, but the teacher can correct them (weight updates). This way the whole class (layer) learn about a topic better."

– some guy called Shubham Agrawal on *StackExchange.com*

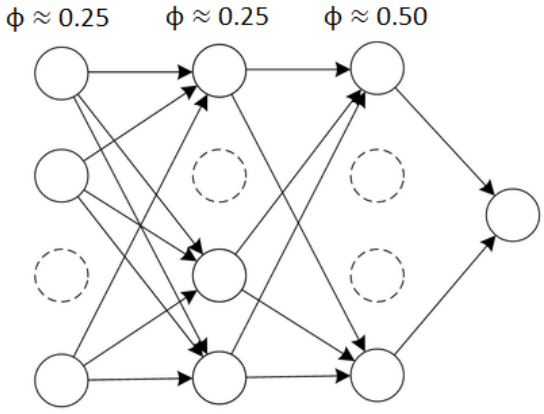
Dropout in practice

Procedure during training:

1. Choose one or more (or all) layers that will be subjected to dropout.
2. For each layer subjected to dropout, choose a dropout rate $\phi \in [0, 1)$.
3. The NN will then go through each of the nodes in the layer subject to dropout and set their "outbound" weights equal to zero with probability ϕ.



(a) Standard Neural Network



(b) Network after Dropout

Introducing dropout in Casper

Casper **without** dropout:

```
casper %>%
  layer_dense(units = 2, activation = 'sigmoid',
              input_shape = 91) %>%
  layer_dense(units = 2, activation = "softmax")
```

Introducing dropout in Casper

Casper **with** dropout ($\phi = 0.5$ here):

```
casper %>%
  layer_dense(units = 2, activation = 'sigmoid',
              input_shape = 91) %>%
   layer_dropout(0.5) %>%
  layer_dense(units = 2, activation = "softmax")
```

Dropout guidelines

Guidelines:

- Most effective when used on deep layers
- Combine with a large number of nodes in the network
- Needs more training: some say double the number of epochs

Try using dropout on your NNs

Go to the course website and find exercise session 4:

Exercise session 4

Machine learning & neural networks

Anne Helby Petersen

May 9, 2019

Overview

The goals of this exercise session is to:

- Try avoiding overfitting by use of dropout
- Get an idea of what model tuning does

4.1: Using dropout

We will use Brad from excerise 3.2. as an offset to see how dropout may help us in avoiding overfitting. If you didn't get to defining him, you can use this code as a starting point for Brad:

```
#define Brad and compile him
brad <- keras_model_sequential()
```