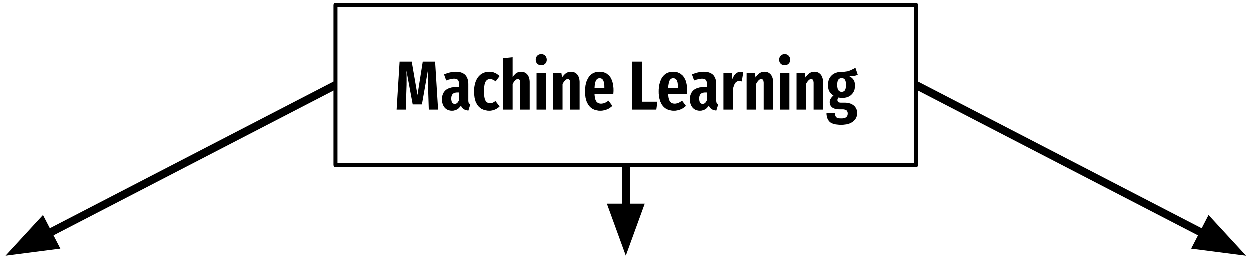
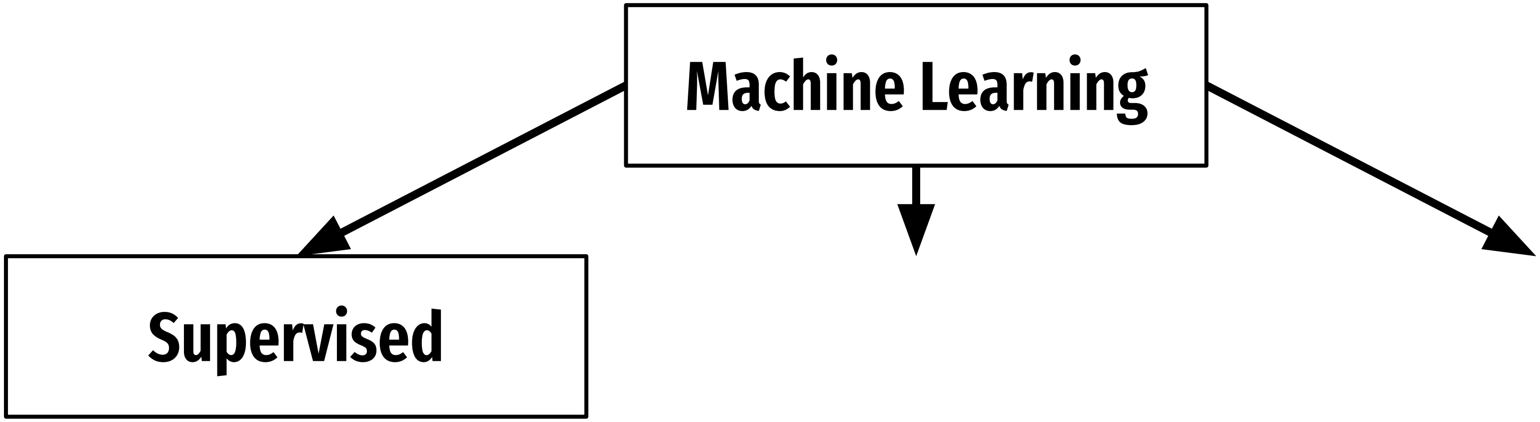
Basics of Machine Learning

Ismaël Lajaaiti

What is Machine Learning?


Machine Learning



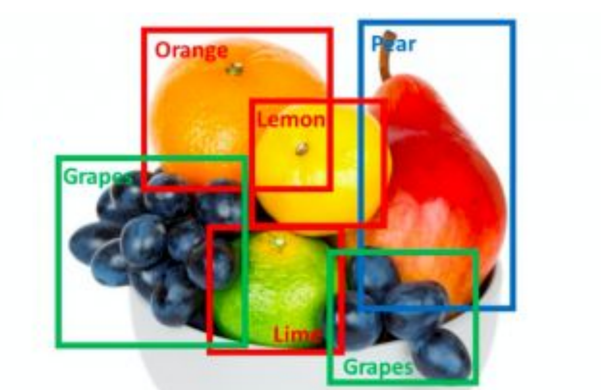
Train algorithm with labeled data

Machine Learning

Supervised

Train algorithm with labeled data

Example: Image classification

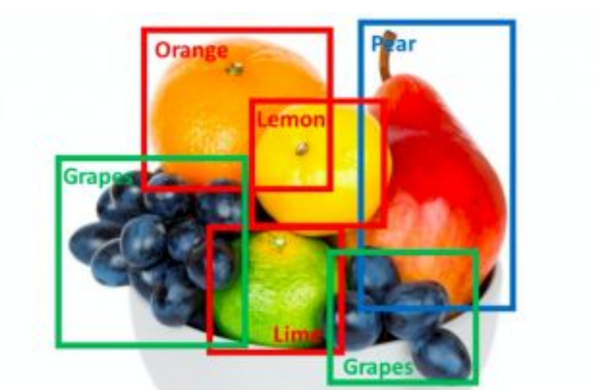


Machine Learning

Supervised

Train algorithm with labeled data

Example: Image classification



Unsupervised

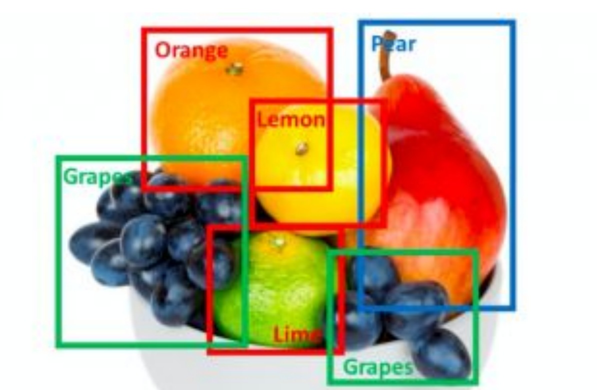
The algorithm discover the information by itself

Machine Learning

Supervised

Train algorithm with labeled data

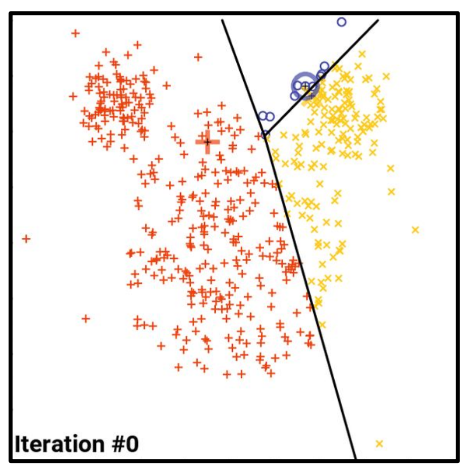
Example: Image classification



Unsupervised

The algorithm discover the information by itself

Example: K-means algorithm

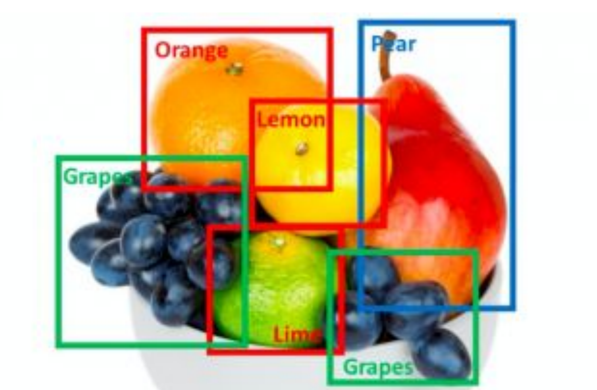


Machine Learning

Supervised

Train algorithm with labeled data

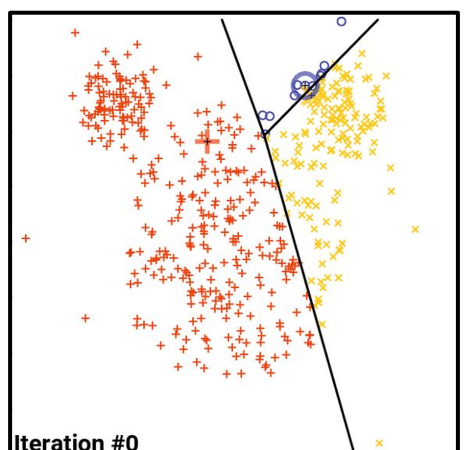
Example: Image classification



Unsupervised

The algorithm discover the information by itself

Example: K-means algorithm



Reinforcement

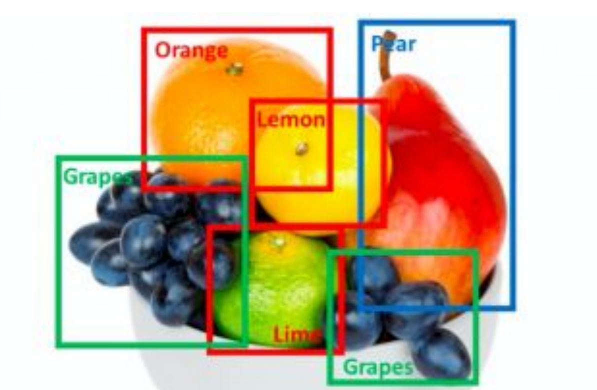
Simulate game-like situations, the algorithm get reward (for ‘good moves’) and penalties (for ‘bad moves’)

Machine Learning

Supervised

Train algorithm with labeled data

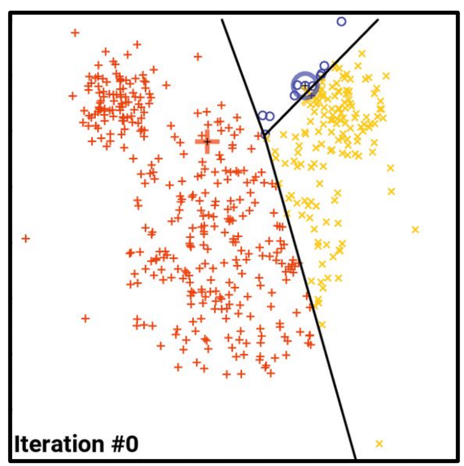
Example: Image classification



Unsupervised

The algorithm discover the information by itself

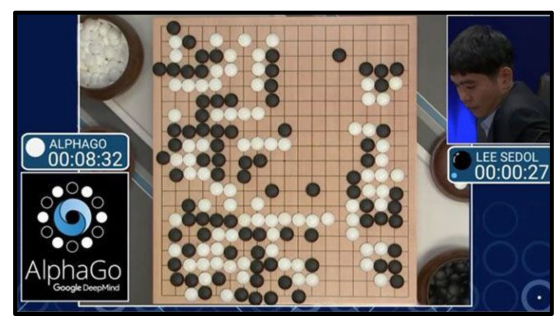
Example: K-means algorithm



Reinforcement

Simulate game-like situations, the algorithm get reward (for ‘good moves’) and penalties (for ‘bad moves’)

Example: AlphaGo

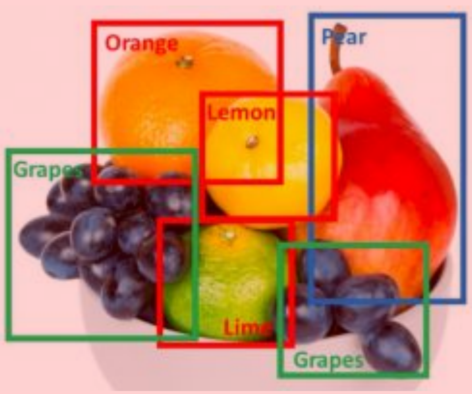


Machine Learning

Supervised

Train algorithm with labeled data

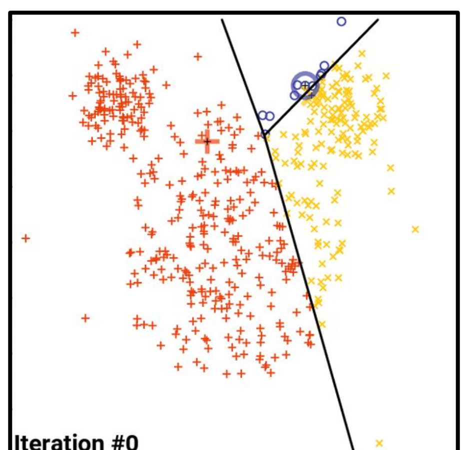
Example: Image classification



Unsupervised

The algorithm discover the information by itself

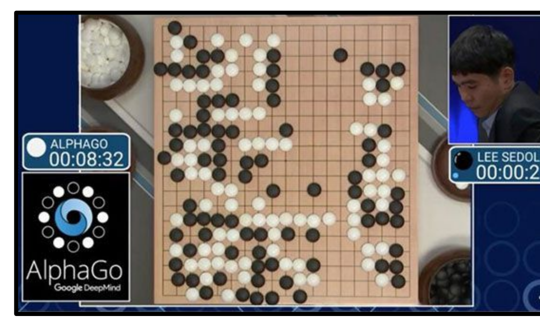
Example: K-means algorithm



Reinforcement

Simulate game-like situations, the algorithm get reward (for ‘good moves’) and penalties (for ‘bad moves’)

Example: AlphaGo



Supervised Learning

Supervised Learning

Classification

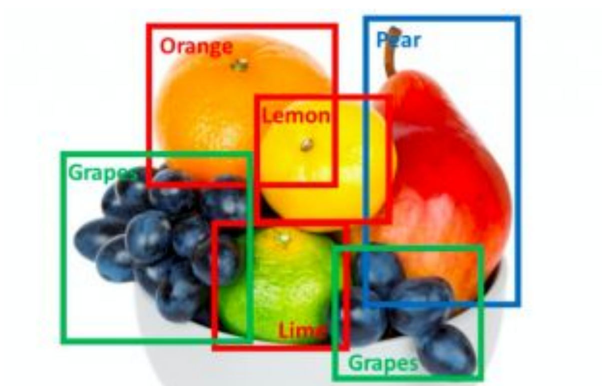
Predict **categorical** variable

Supervised Learning

Classification

Predict **categorical** variable

Example: *image classification* -
Output ∈ {“Orange”, “Grapes”, …}



Supervised Learning

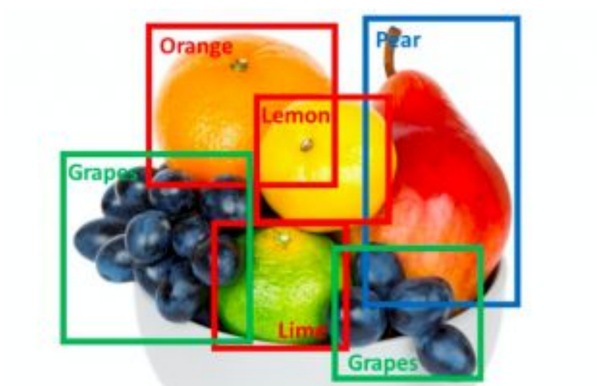
Classification

Regression

Predict **categorical** variable

Predict **quantitative** variable

Example: *image classification* -
Output ∈ {“Orange”, “Grapes”, …}

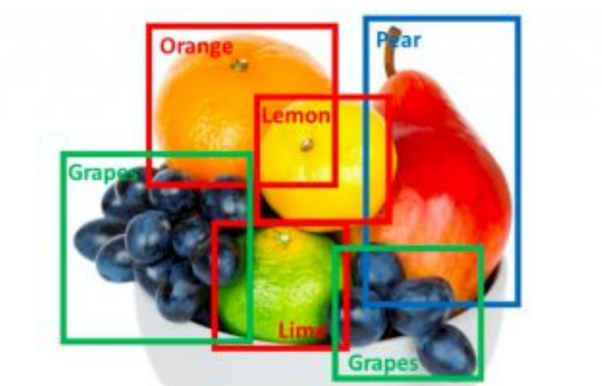


Supervised Learning

Classification

Predict **categorical** variable

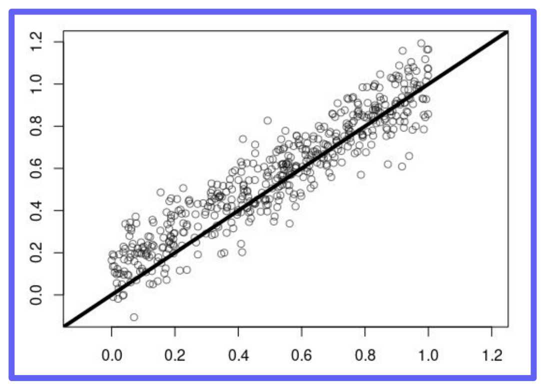
Example: *image classification* -
Output ∈ {“Orange”, “Grapes”, …}



Regression

Predict **quantitative** variable

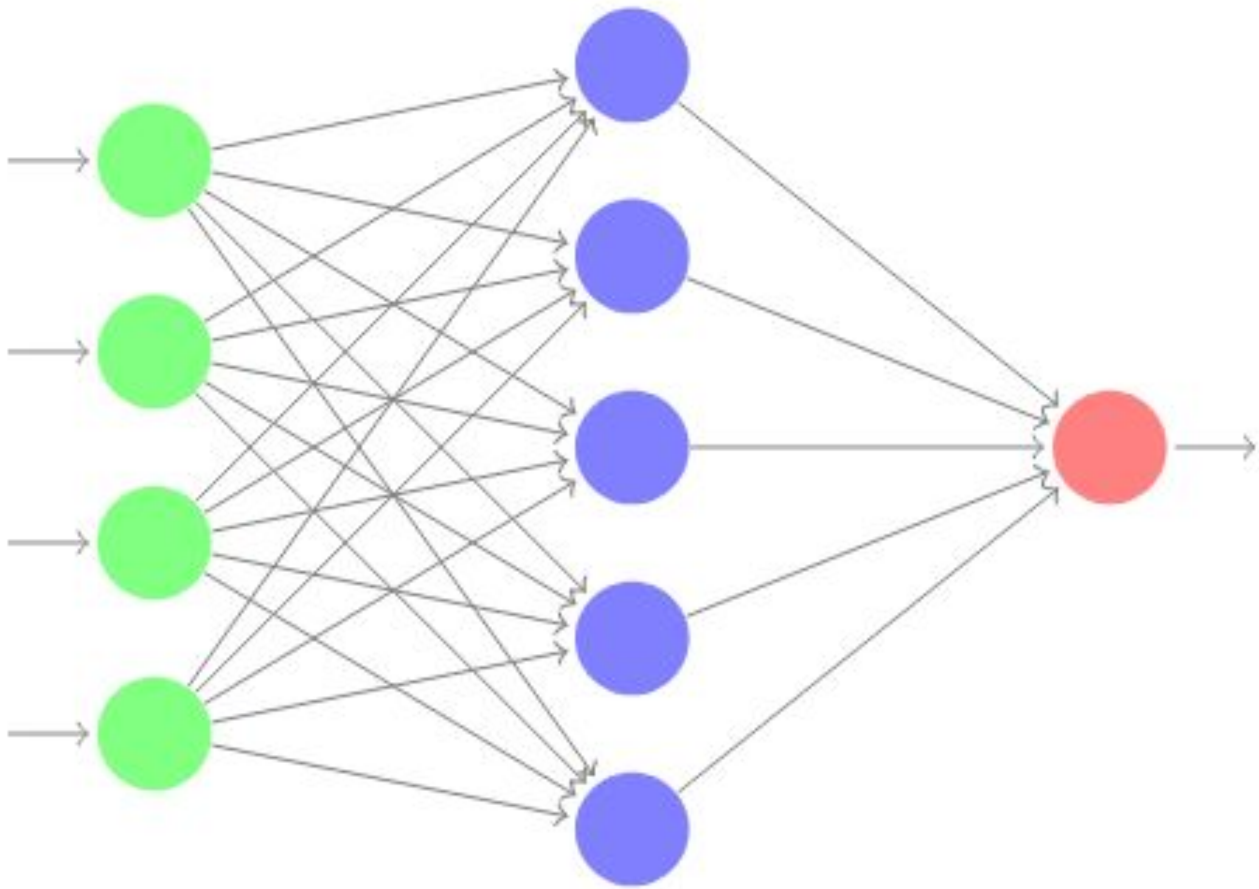
Example: *Guess the correlation* -
Predict correlation from scatter plot



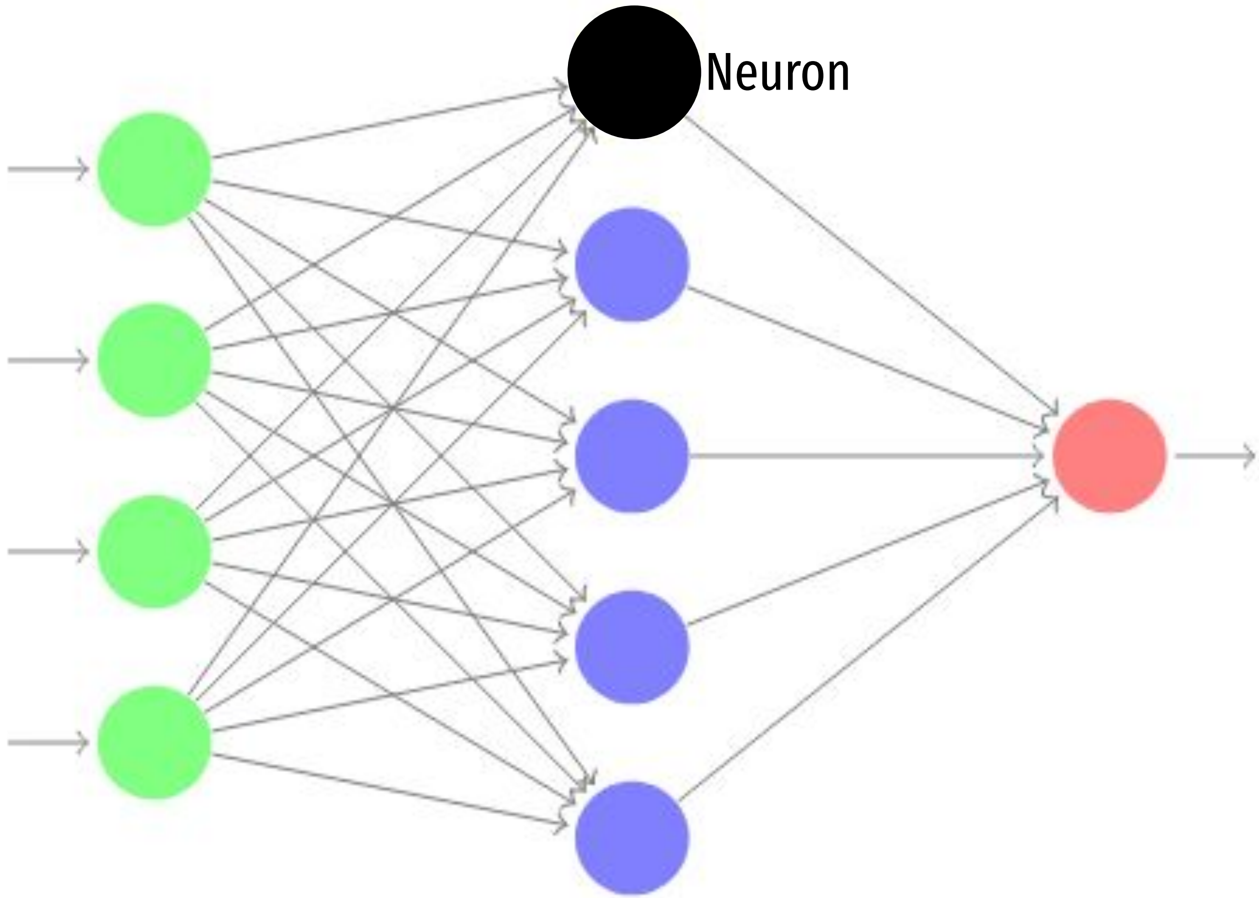
Scatter plot - ***input***

Cor(X,Y) = 0.9
Pred. corr. - ***output***

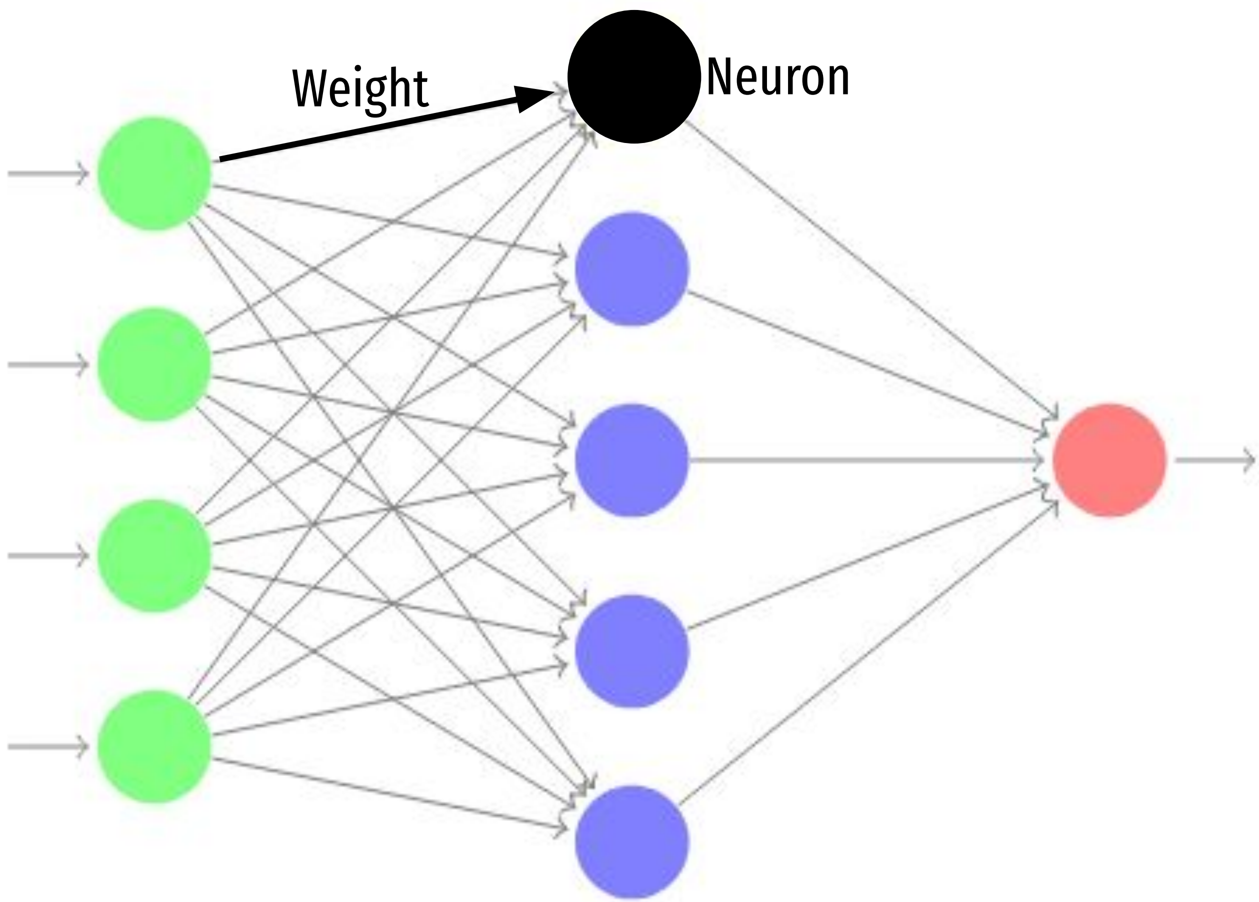
Neural networks



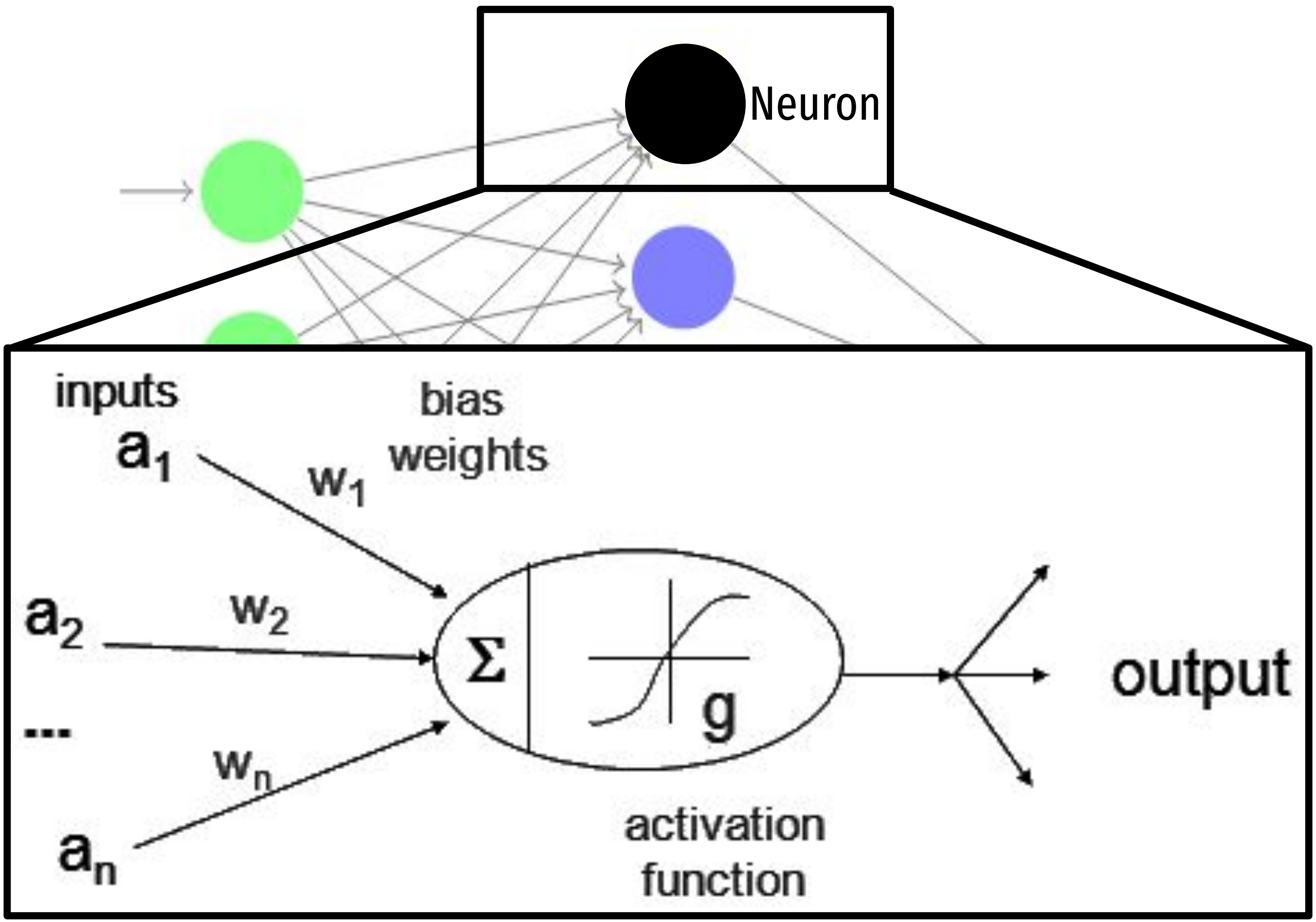
Neural networks



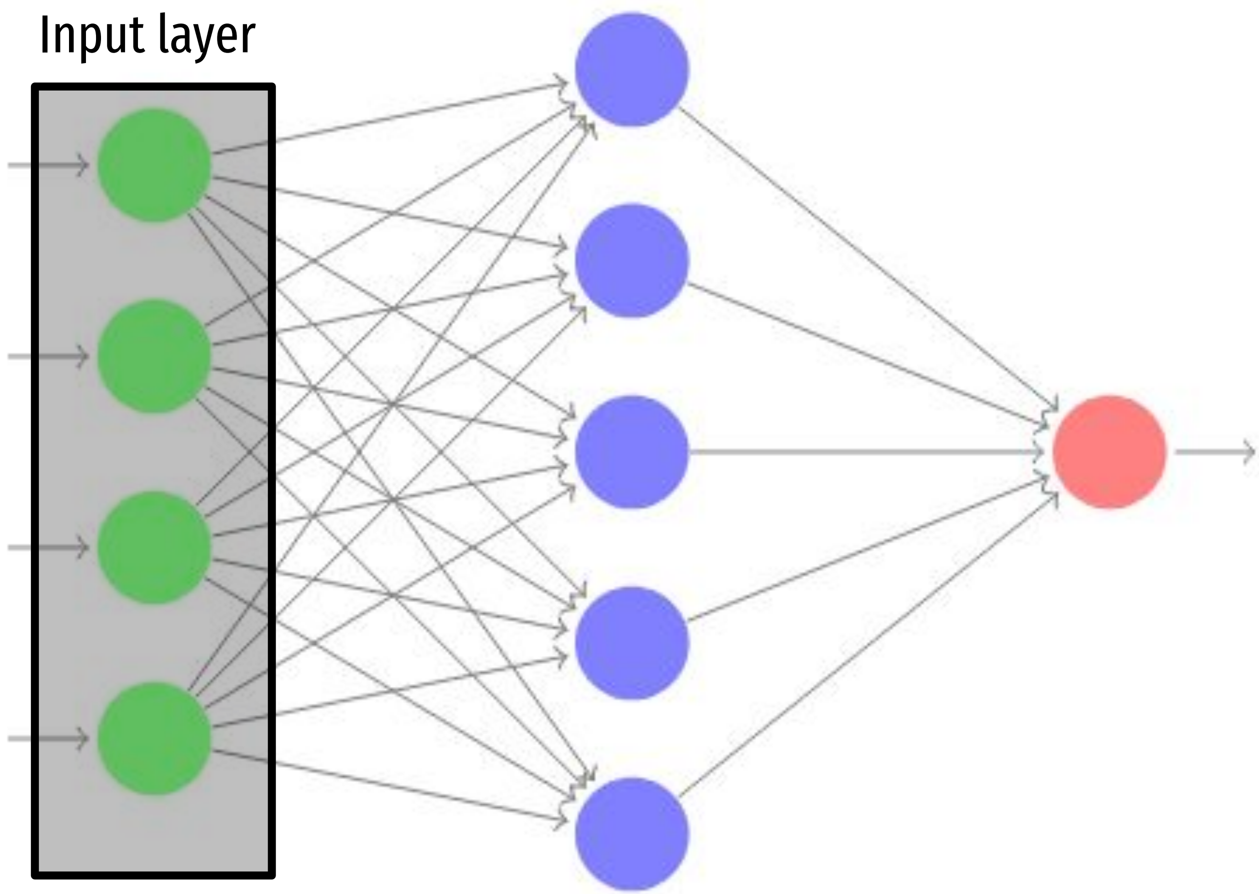
Neural networks



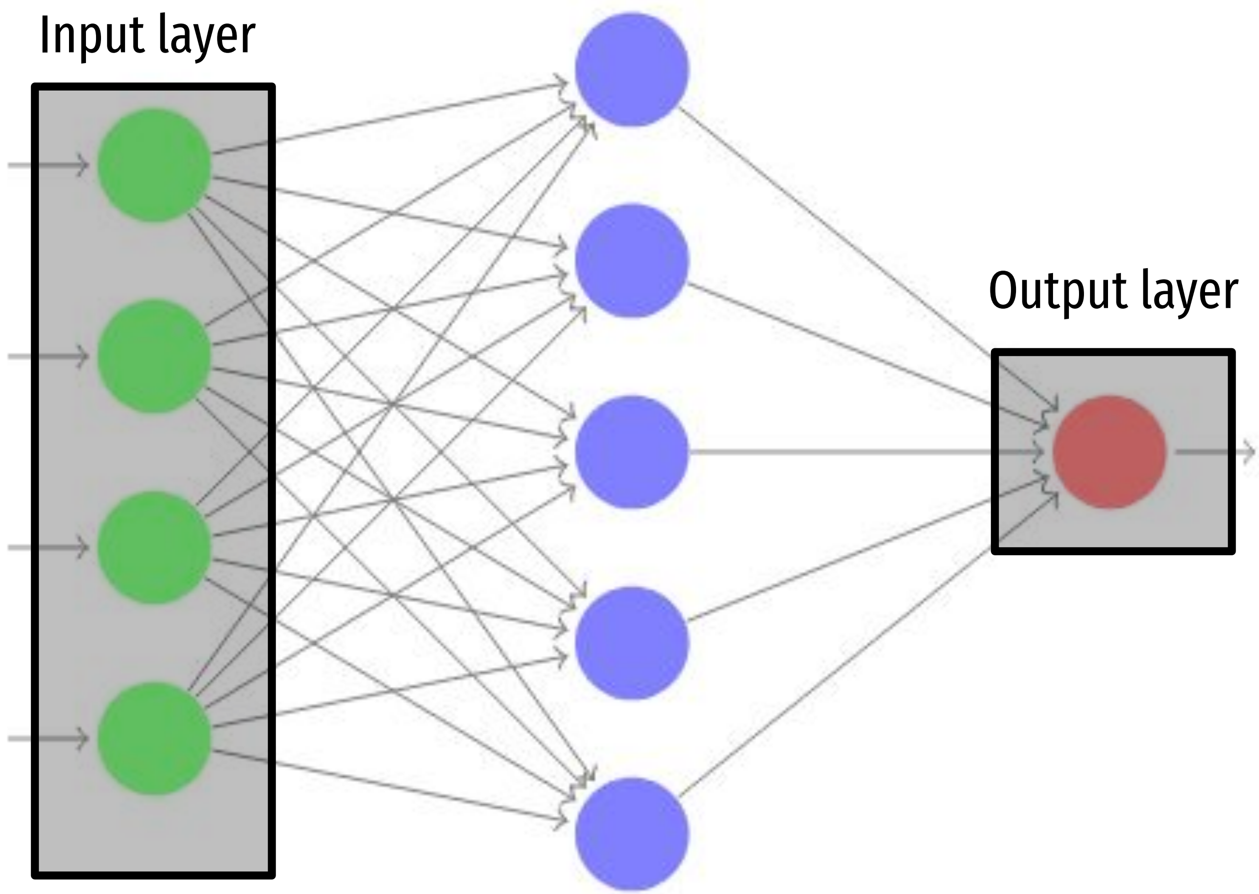
Neural networks



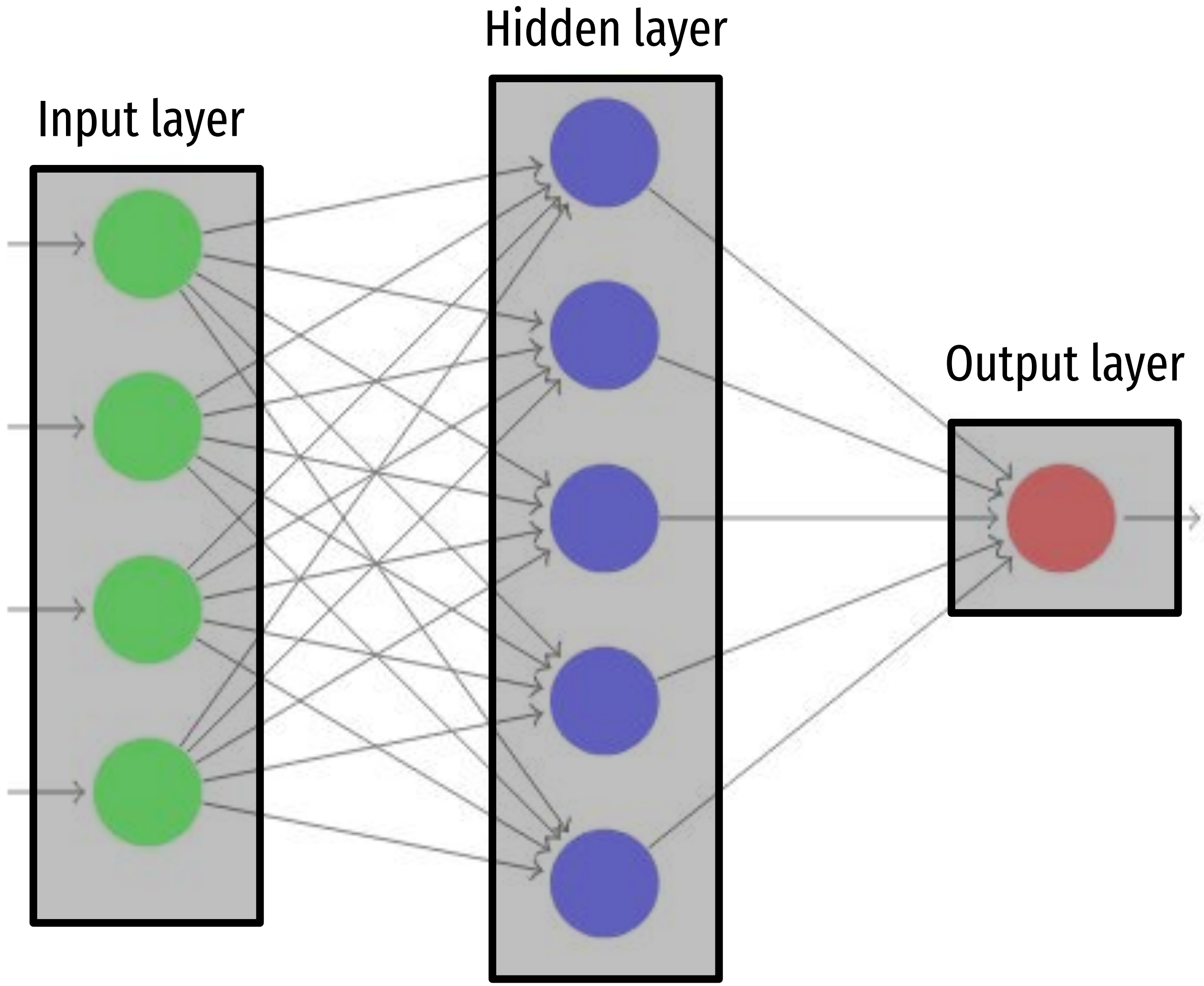
Neural networks



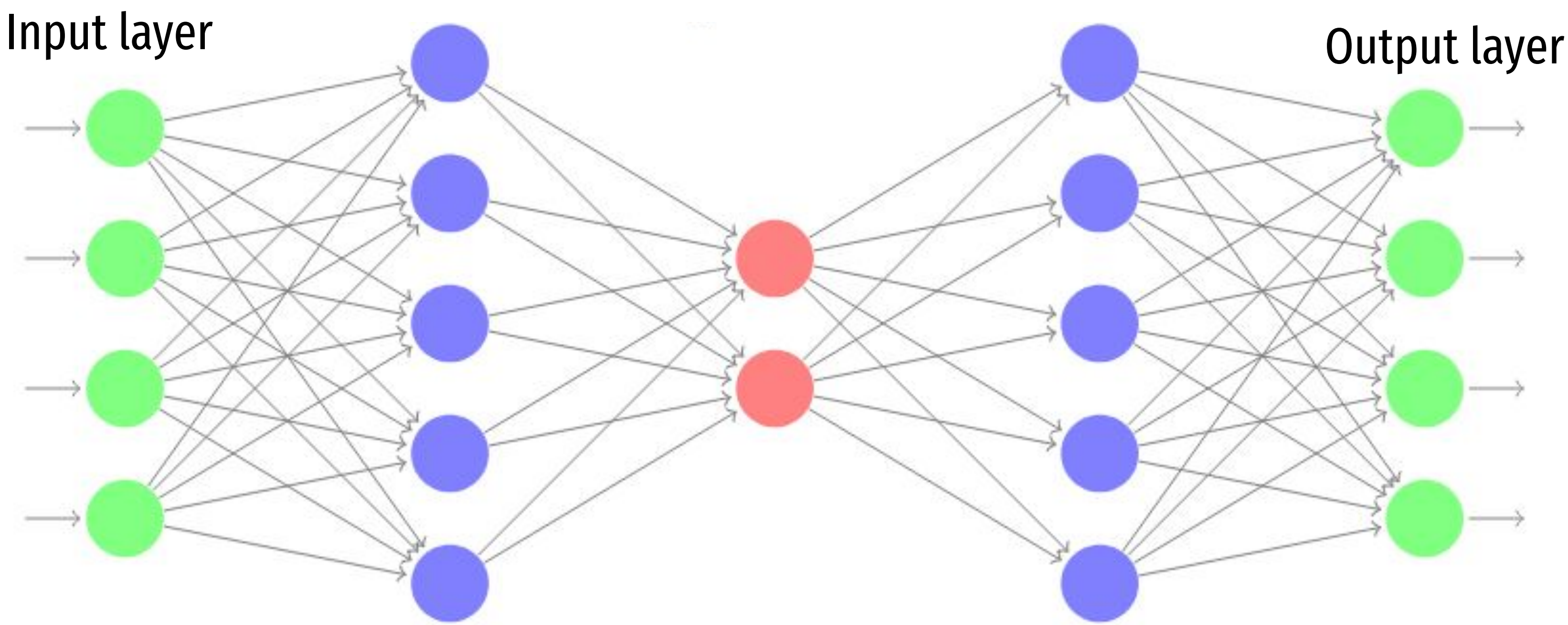
Neural networks



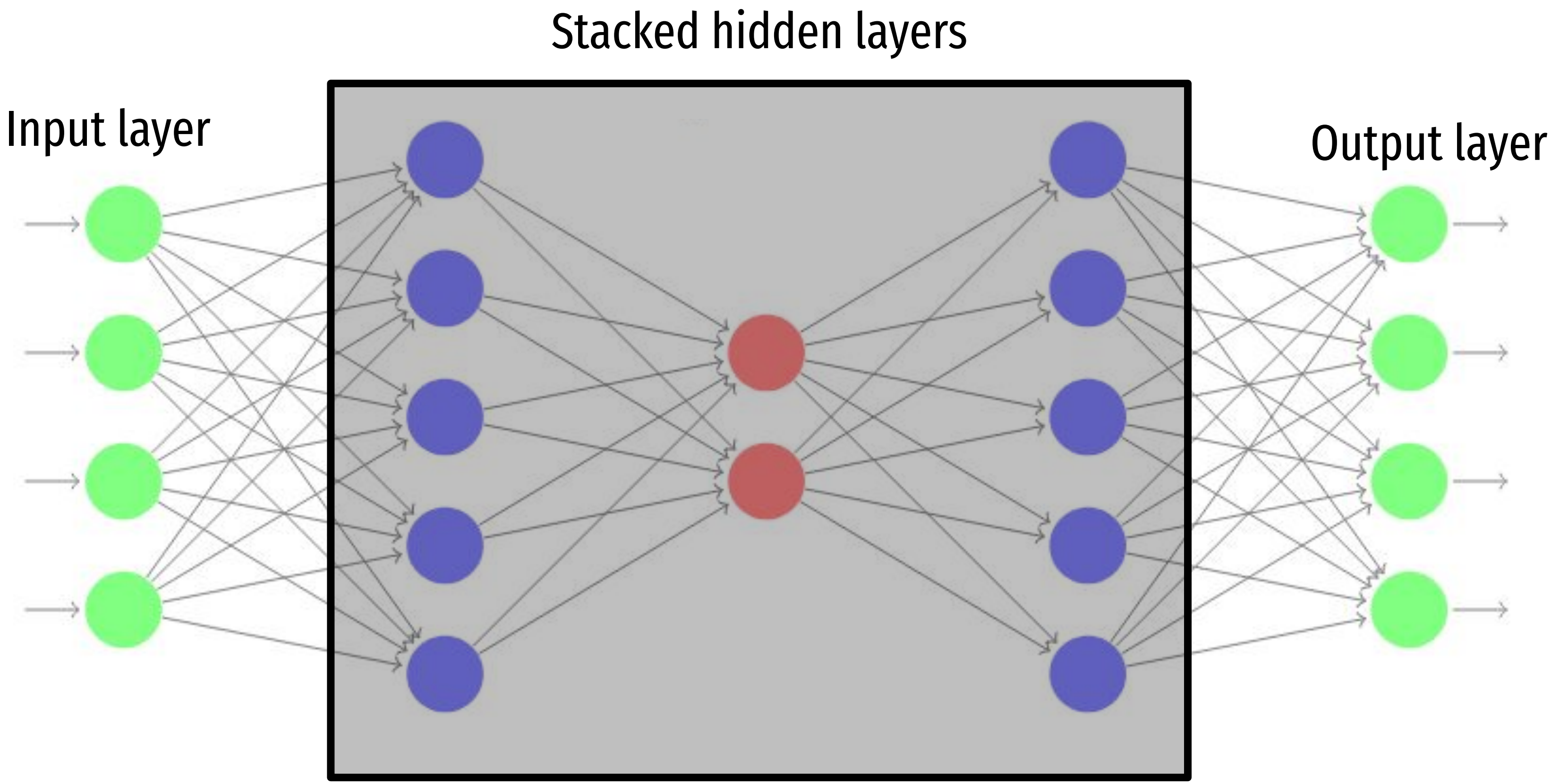
Neural networks



Deep neural network



Deep neural network



How does the neural network learn?

1. Define an objective: **minimize distance between predicted and expected value**

How does the neural network learn?

1. Define an objective: **minimize** $\underbrace{\textbf{distance between predicted and expected value}}_{\text{Loss function}}$

How does the neural network learn?

1. Define an objective: **minimize** $\underbrace{\textbf{distance between predicted and expected value}}_{\text{Loss function}}$

Example: *Mean Square Error (MSE)*

$$MSE = \Sigma_i (y_{i,pred} - y_{i,true})^2$$

How does the neural network learn?

1. Define an objective: $\underbrace{\textbf{minimize distance between predicted and expected value}}_{\text{Loss function}}$

   Example: *Mean Square Error (MSE)*

   $$MSE = \Sigma_i(y_{i,pred} - y_{i,true})^2$$

2. Optimize network to reach this objective

How does the neural network learn?

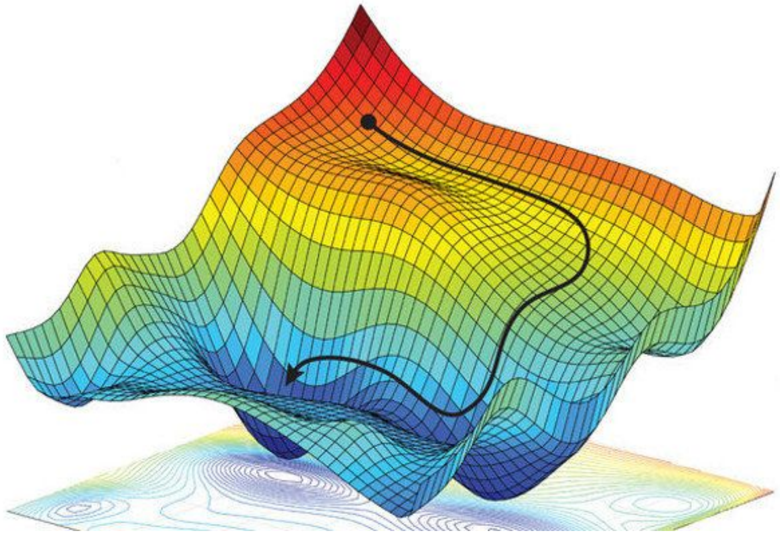
1. Define an objective: $\underbrace{\textbf{minimize distance between predicted and expected value}}_{\text{Loss function}}$

Example: *Mean Square Error (MSE)*

$$MSE = \Sigma_i(y_{i,pred} - y_{i,true})^2$$

2. Optimize network to reach this objective

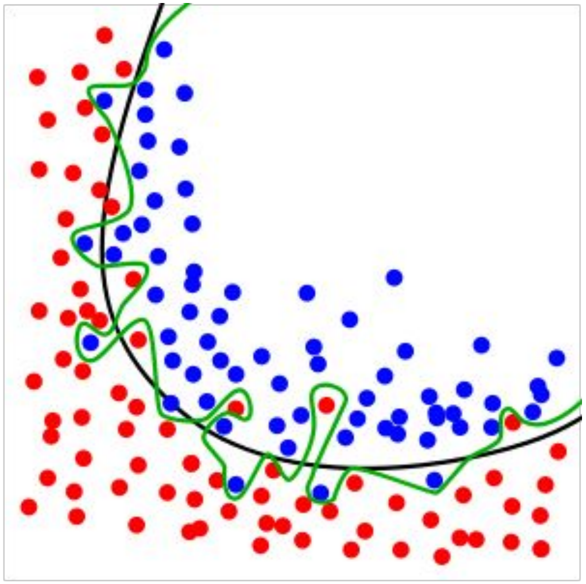
   Compute loss **gradient** vs. weights with **backward propagation**



How do we know that the neural network has learned correctly?

How do we know that the neural network has learned correctly?

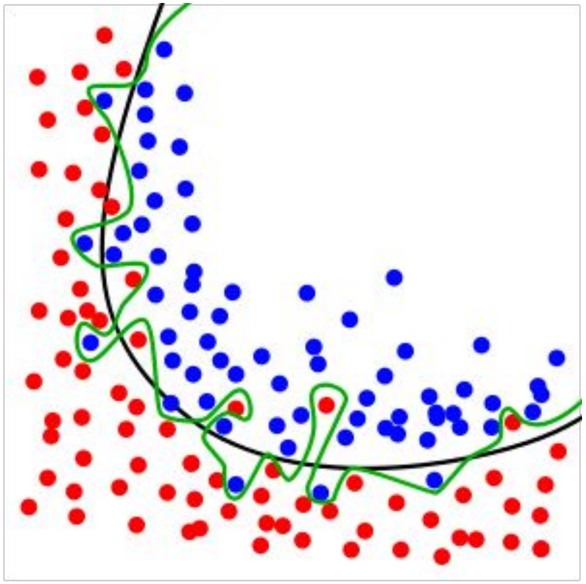
Avoid **overfitting** i.e. neural network learn train data 'by heart' and is not able to extrapolate to new data



How do we know that the neural network has learned correctly?

Avoid **overfitting** i.e. neural network learn train data ‘by heart’ and is not able to extrapolate to new data

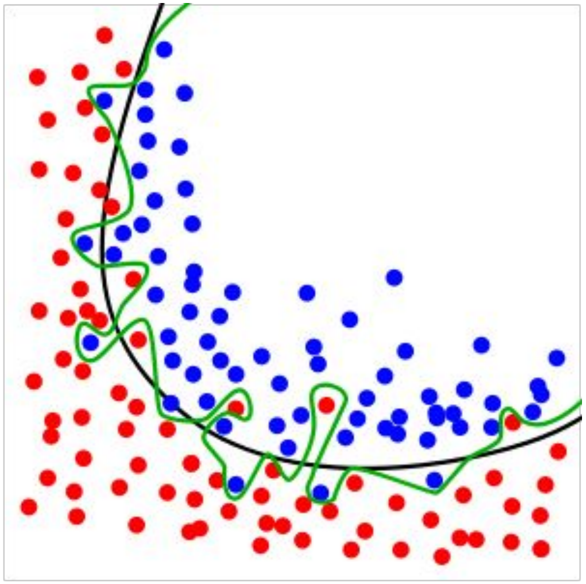
1. Split data set in two subsets: training set & validation set

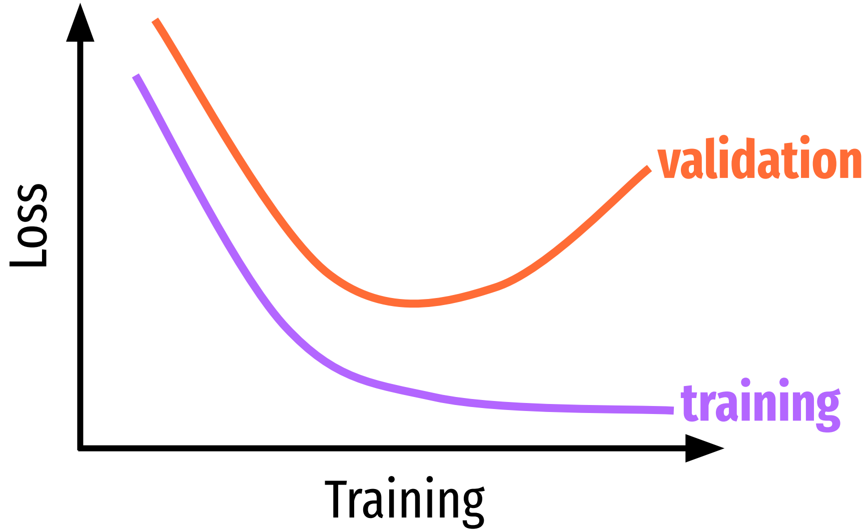


How do we know that the neural network has learned correctly?

Avoid **overfitting** i.e. neural network learn train data ‘by heart’ and is not able to extrapolate to new data

1. Split data set in two subsets: training set & validation set
2. Track train & validation losses through the training

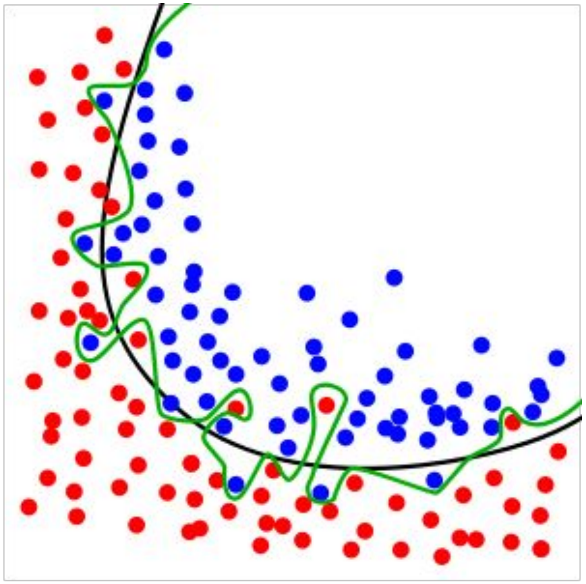


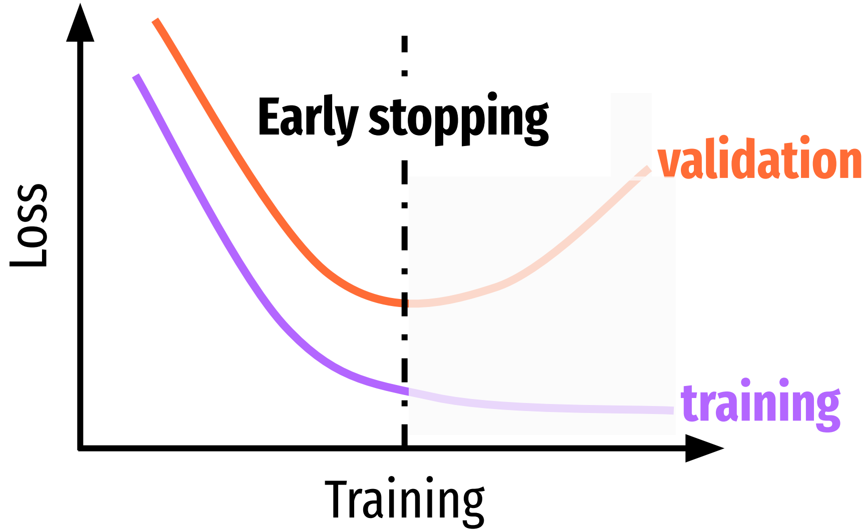


How do we know that the neural network has learned correctly?

Avoid **overfitting** i.e. neural network learn train data 'by heart' and is not able to extrapolate to new data

1. Split data set in two subsets: training set & validation set
2. Track train & validation losses through the training
3. Stop the training when validation loss stop to decrease = **early stopping**





Frameworks (in R)

TensorFlow

Torch

TensorFlow

Torch

Google

Facebook

TensorFlow

Google

Industry-focused

Torch

Facebook

Research-focused

TensorFlow

Google

Industry-focused

Easier to learn (Keras)

Torch

Facebook

Research-focused

Harder to learn

TensorFlow	Torch
Google	Facebook
Industry-focused	Research-focused
Easier to learn (Keras)	Harder to learn
Requires Python	Does not require Python

TensorFlow	Torch
Google	Facebook
Industry-focused	Research-focused
Easier to learn (Keras)	Harder to learn
Requires Python	Does not require Python
Good documentation	Poor documentation

TensorFlow	**Torch**
Google	Facebook
Industry-focused	Research-focused
Easier to learn (Keras)	Harder to learn
Requires Python	Does not require Python
Good documentation	Poor documentation

Examples of use

APPLICATION

Methods in Ecology and Evolution BRITISH ECOLOGICAL SOCIETY

Machine learning to classify animal species in camera trap images: Applications in ecology

Michael A. Tabak[1,2] | Mohammad S. Norouzzadeh[3] | David W. Wolfson[1] | Steven J. Sweeney[1] | Kurt C. Vercauteren[4] | Nathan P. Snow[4] | Joseph M. Halseth[4] | Paul A. Di Salvo[1] | Jesse S. Lewis[5] | Michael D. White[6] | Ben Teton[6] | James C. Beasley[7] | Peter E. Schlichting[7] | Raoul K. Boughton[8] | Bethany Wight[8] | Eric S. Newkirk[9] | Jacob S. Ivan[9] | Eric A. Odell[9] | Ryan K. Brook[10] | Paul M. Lukacs[11] | Anna K. Moeller[11] | Elizabeth G. Mandeville[2,12] | Jeff Clune[3] | Ryan S. Miller[1]

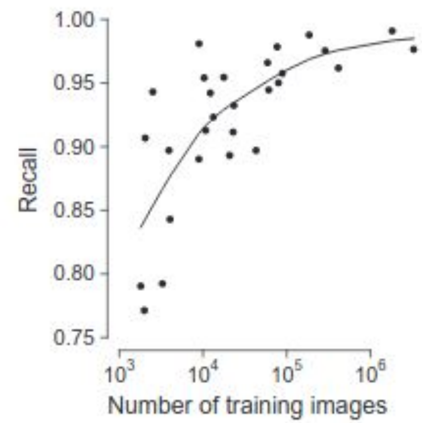
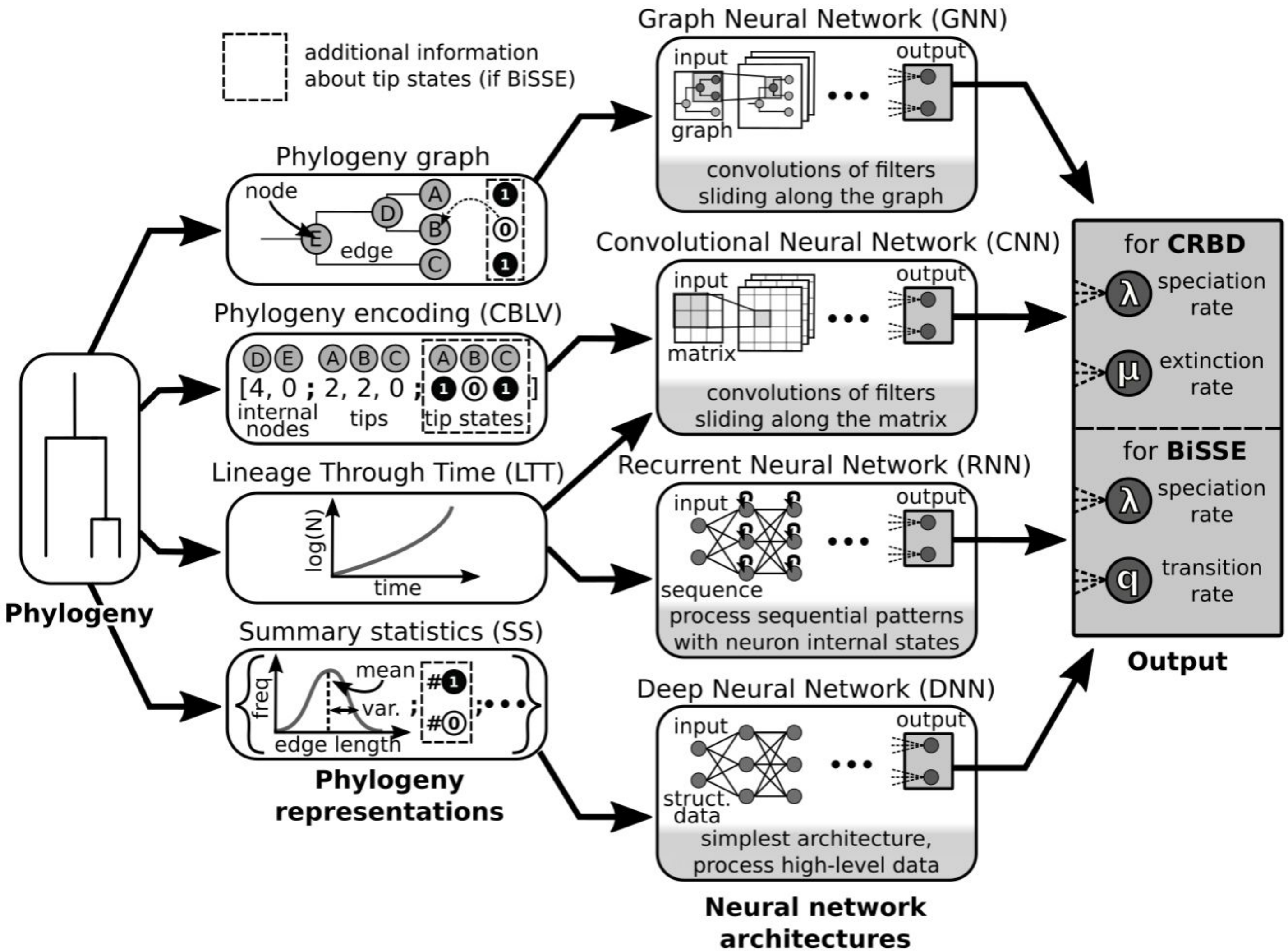


FIGURE 3 Model recall (the ability of the model to recognize species) increased with the size of the training dataset for that species. Points represent each species or group of species. The line represents the result of generalized additive models relating the two variables (see Appendix S9 for details)

Inferring speciation and extinction rates from phylogenies



Thanks!

Questions