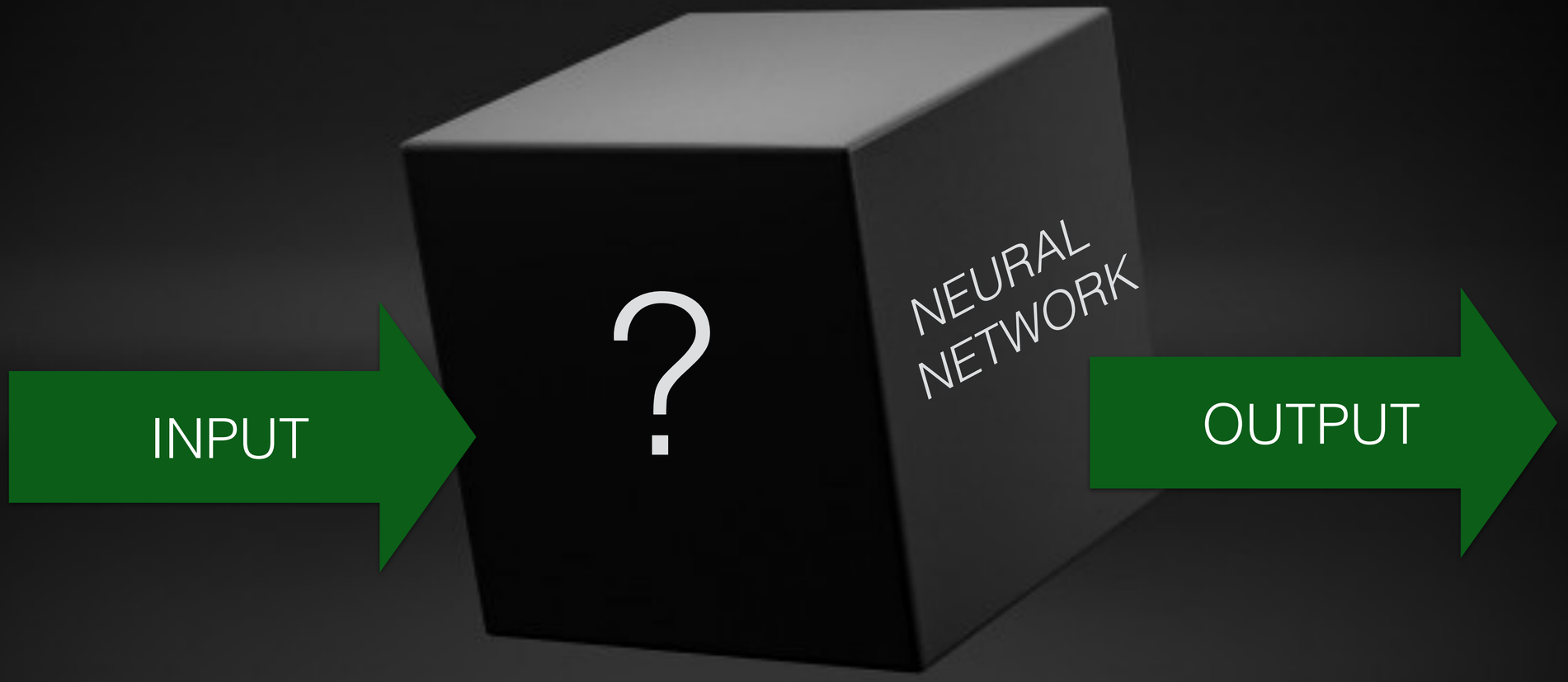
UNIVERSITY
OF MIAMI

PEEKING UNDER THE HOOD OF AN ARTIFICIAL NEURAL NETWORK

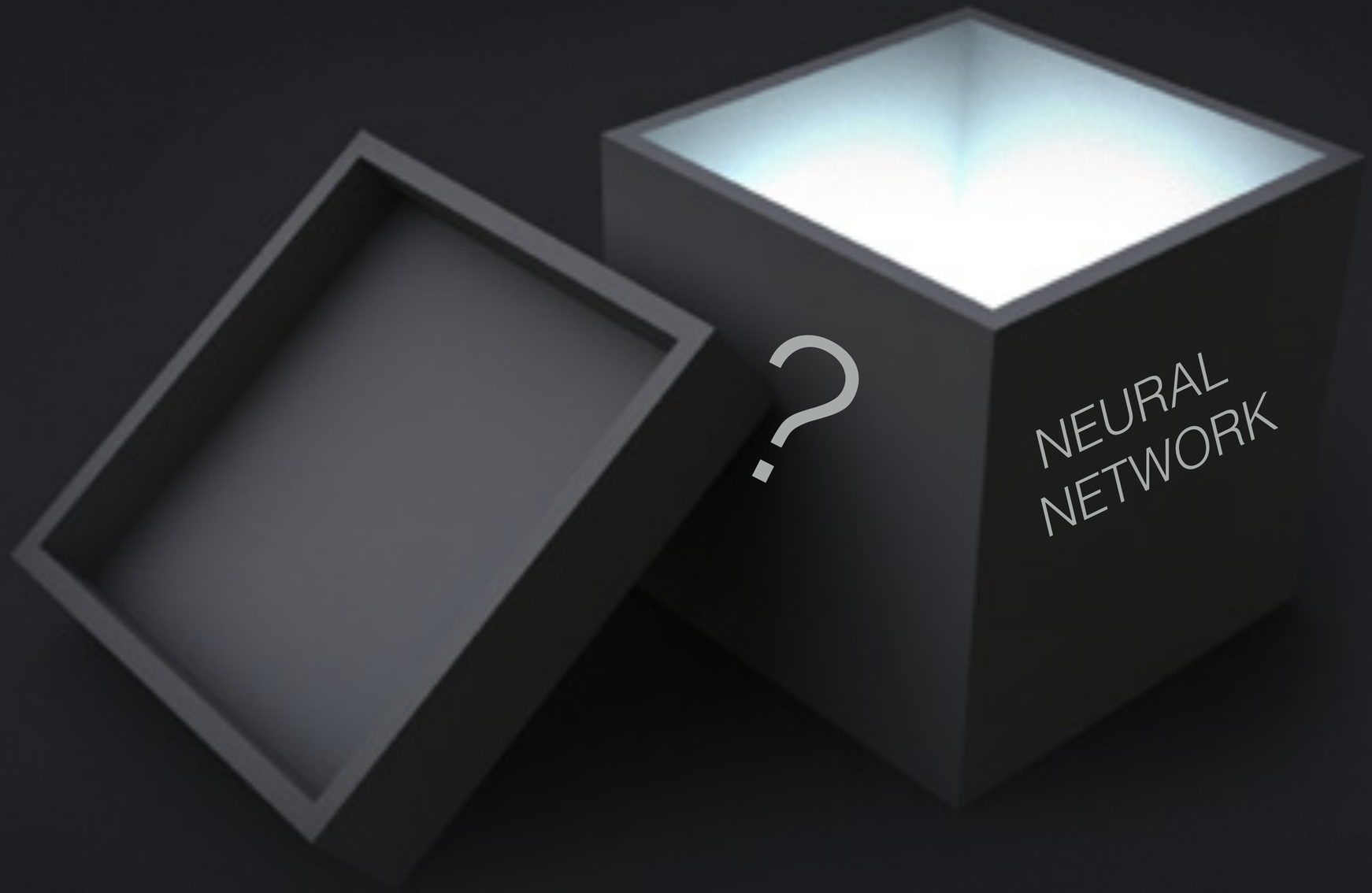
Matt Grossi
matthew.grossi@rsmas.miami.edu

October 26, 2018 • RSMAS Lunch Bytes Seminar

The “black box” objection

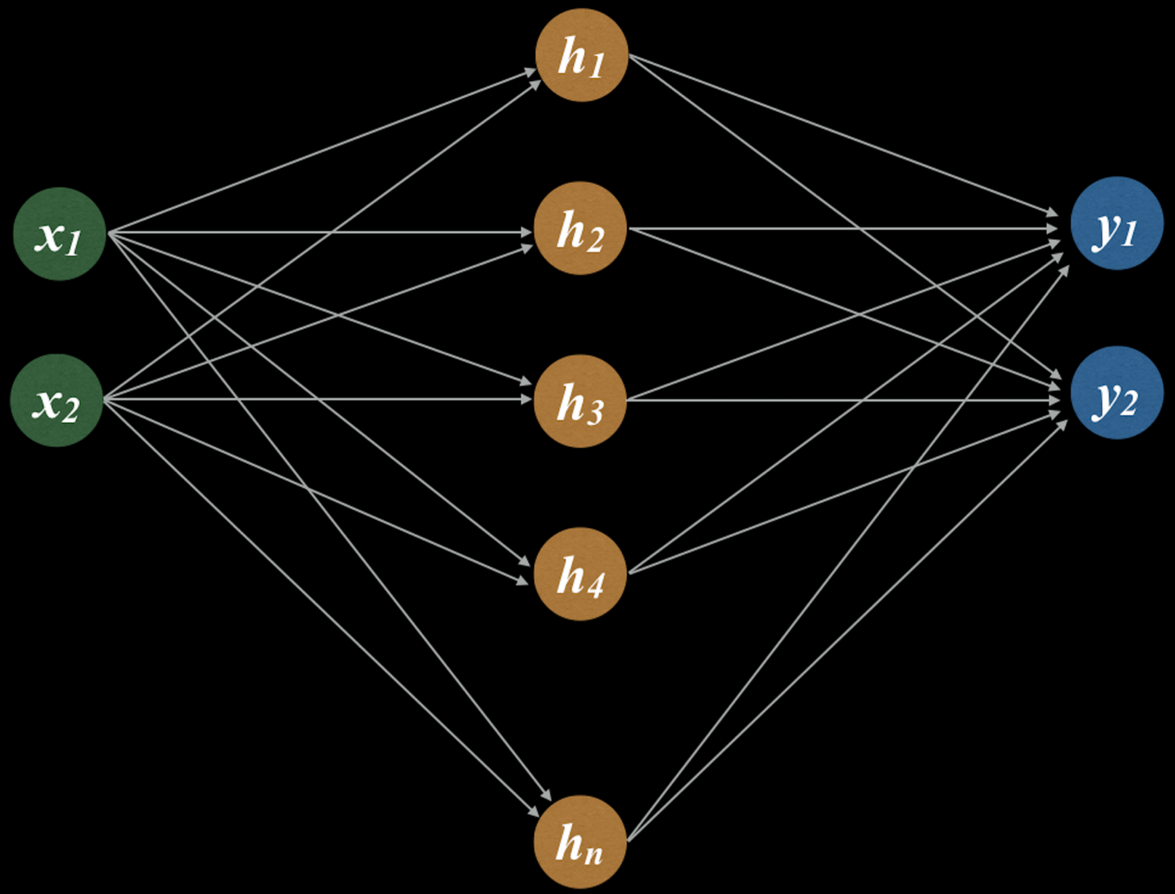


What’s inside?



What’s inside?

What's inside a neural network?



A *multilayer perceptron*.

Think *brain…*

…with a vast network of *interconnected neurons*.

How do we **learn**?

How do we **teach**?

LiveSlides web content

To view

Download the add-in.

liveslides.com/download

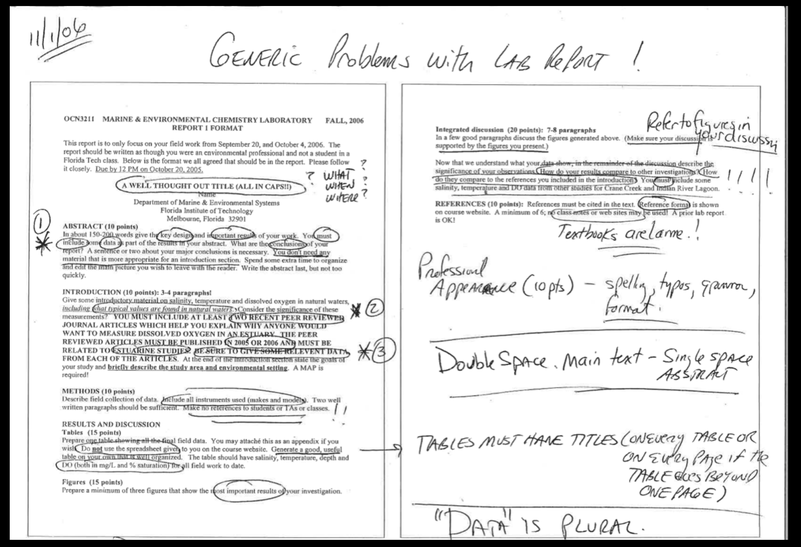
Start the presentation.

The conventional teaching process (for better or for worse!)

1. Known information is communicated to the learner

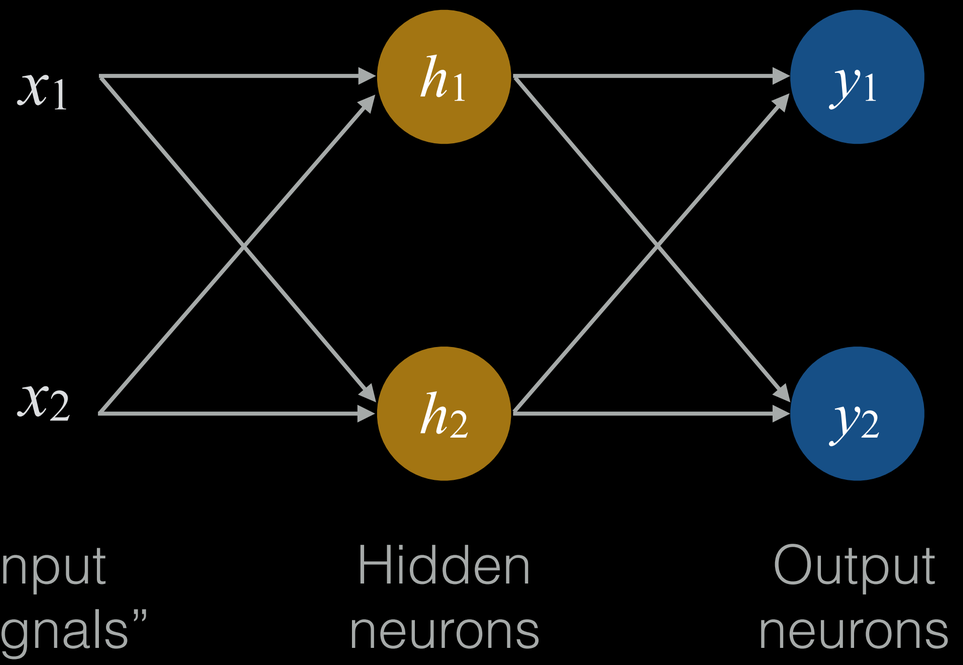
2. Student communicates the information back

3. Responses are assessed, errors are corrected



4. Repeat until the task is mastered perfectly (hopefully!)

What goes on inside?



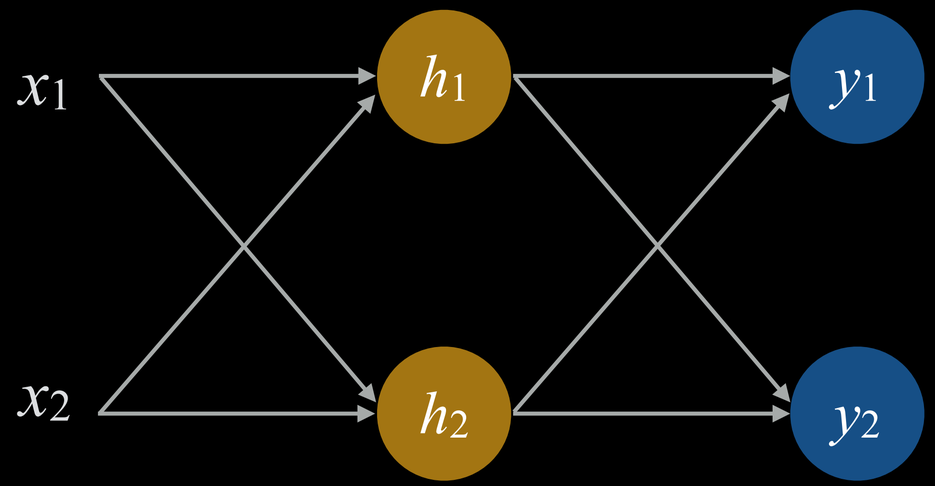
A simple neural network

The network is passed a signal of information.

A decision is made about the information (*e.g.*, to what class does it belong?) — ***output***

The decision is compared to the known information. If the network erred, small adjustments are made to its "thinking process."

What goes on inside?



Input "signals"

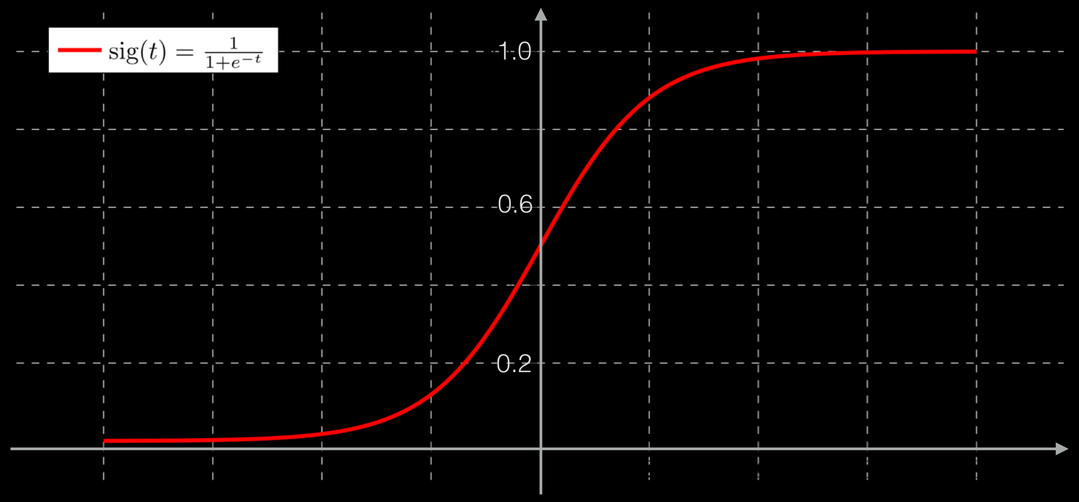
Hidden neurons

Output neurons

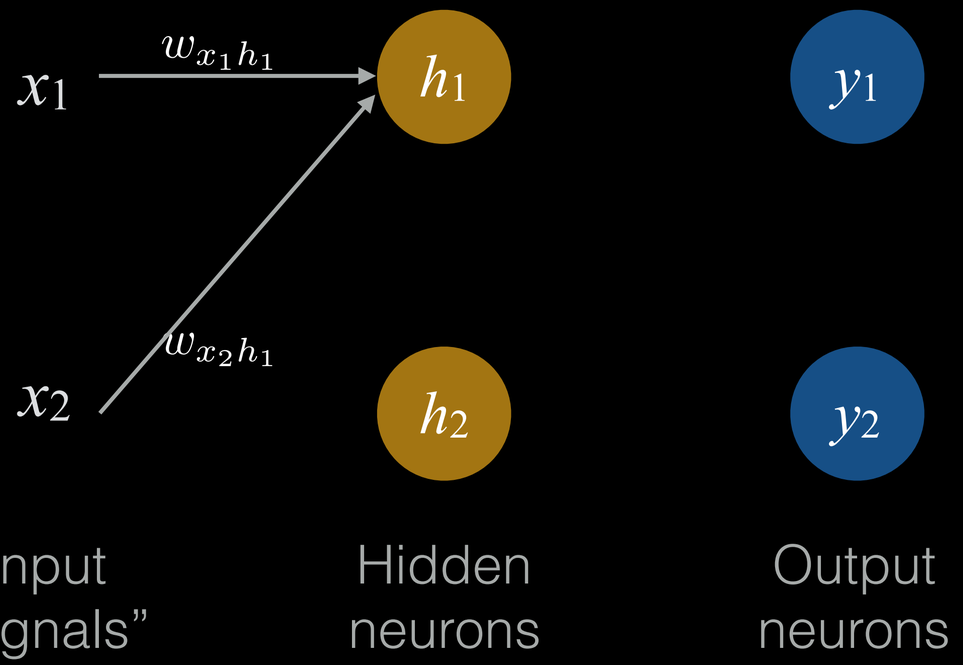
A simple neural network

Each neuron receives as input a *weighted sum* (Σ) from the previous layer and passes it to a ***transfer function***, for example:

$$f(\Sigma) = \frac{1}{1 + e^{-\Sigma}}$$



What goes on inside?



A simple neural network

Consider the first hidden neuron, h_1:

1. Calculate the neuron's **input**:

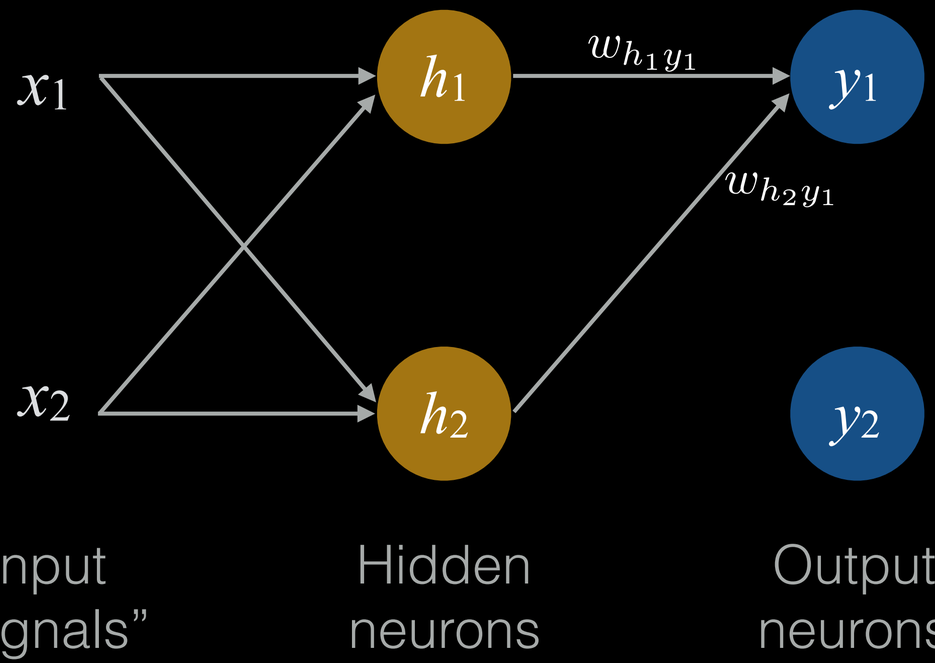
$$\Sigma_{h_1} = x_1 w_{x_1 h_1} + x_2 w_{x_2 h_1}$$

2. Calculate the neuron's **output**:

$$\frac{1}{1 + e^{-\Sigma_{h_1}}} = \frac{1}{1 + e^{-x_1 w_{x_1 h_1} - x_2 w_{x_2 h_1}}}$$

This gets passed to the output neurons, y_1 and y_2

What goes on inside?



A simple neural network

Now the first output neuron, y_1:

3. Calculate the neuron's input:

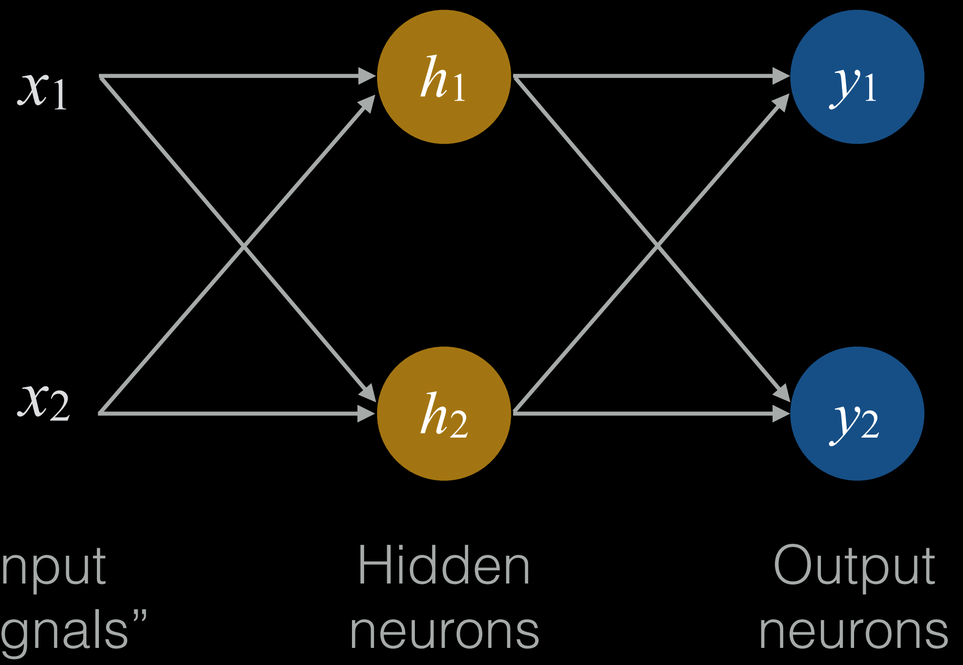
$$\Sigma_{y_1} = h_1 w_{h_1 y_1} + h_2 w_{h_2 y_1}$$

4. Calculate the neuron's output:

$$y_1 = \frac{1}{1 + e^{-h_1 w_{h_1 y_1} - h_2 w_{h_2 y_1}}}$$

This process is called ***forward propagation***.

What goes on inside?



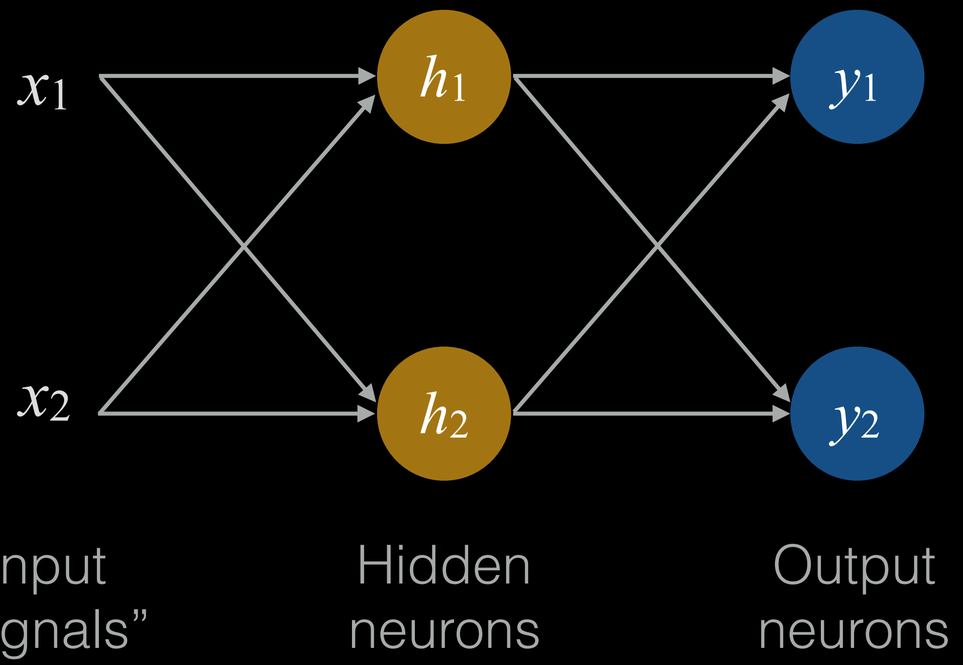
A simple neural network

The output is directly dependent on values of the *weights*.

Goal: ***Find the best values of the weights to maximize the model's performance.***

Next, compare the output to the expected (target) values.

What goes on inside?



A simple neural network

Back propagation of error:

5. Determine each neuron's "responsibility" for the overall error. Call this δ.

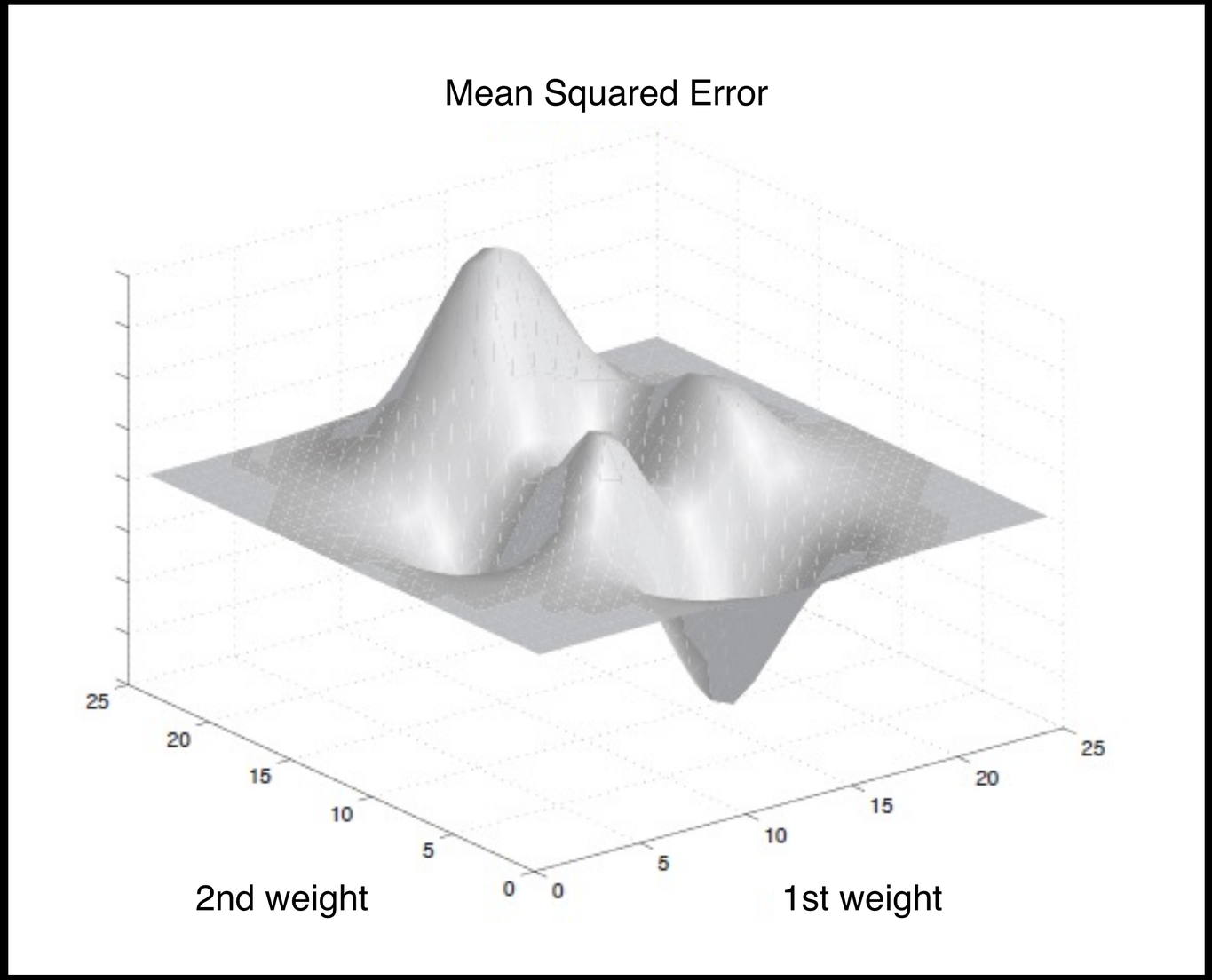
6. Adjust each weight accordingly:

   Output layer: $w = w + \eta\delta h$

   Hidden layer: $w = w + \eta\delta x$

   with learning rate η<0.1

What just happened?



From Kubat (2017), *An Introduction to Machine Learning*

What about the black box objection?

A neural network can theoretically approximate any function, but studying its structure won't provide any insights into the structure of the function being approximated.

What *can* we learn?

1. Are data separable? In other words, is there any learnability?
2. Which attributes are relevant? Which are irrelevant?
3. How are the data being transformed?

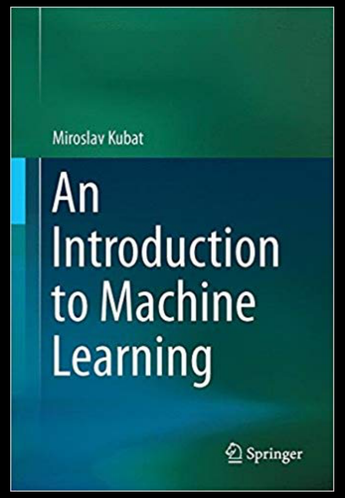
What can neural networks be used for?

Classification problems
Prediction of continuous variables
Forecasts

As long as:
1. The data are able to be transformed into linearly separable classes
2. You have enough data to train adequately
3. You don't care about the fine details of the function. In other words, performance is most important.

If you want to *explore* your data set, a neural network may not be your best tool, or at least not at first!

References



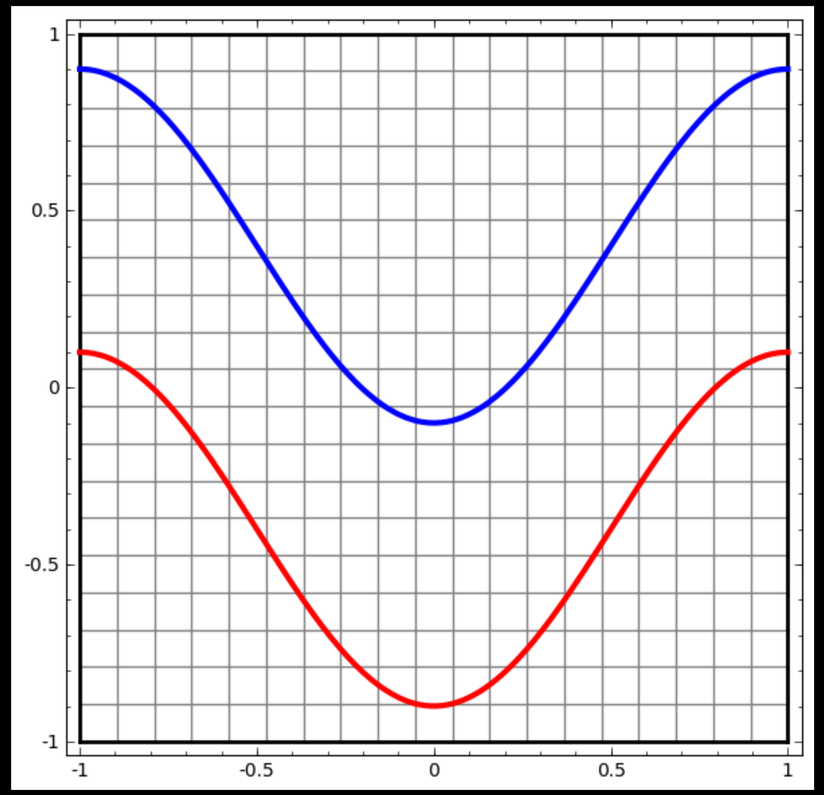
<- Kubat, M. (2017) *An Introduction to Machine Learning* (2nd ed.), Cham: Springer International Publishing.

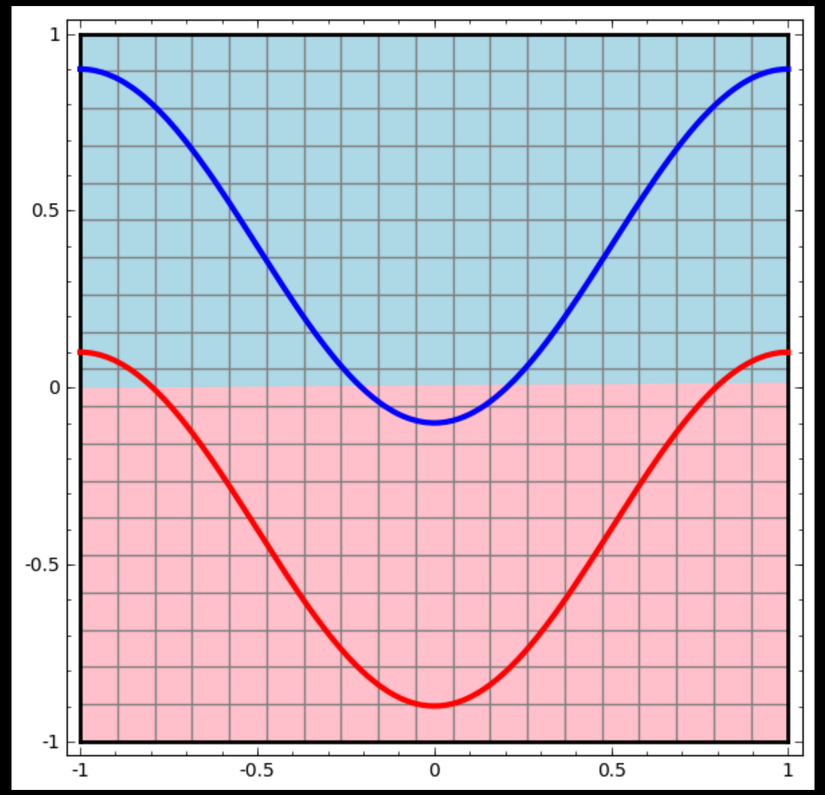
Nielsen, M. A. (2015) *Neural Networks and Deep Learning*, Determination Press, available http://neuralnetworksanddeeplearning.com/index.html

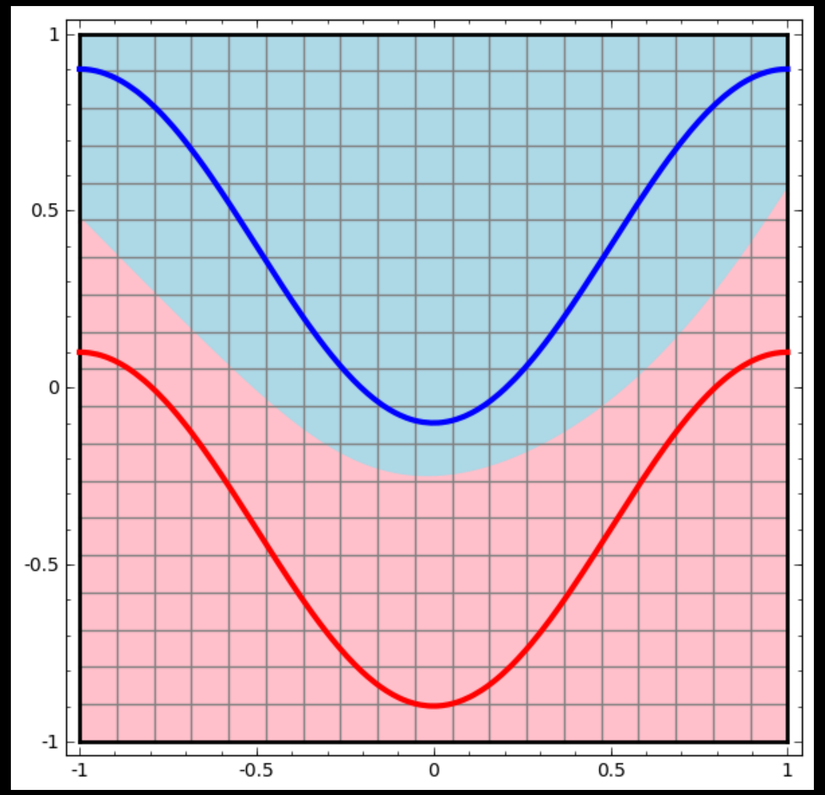
Olah, C. (2014) “Neural Networks, Manifolds, and Topology,” Colah’s blog, available http://colah.github.io/posts/2014-03-NN-Manifolds-Topology/.

Extras…

What just happened?

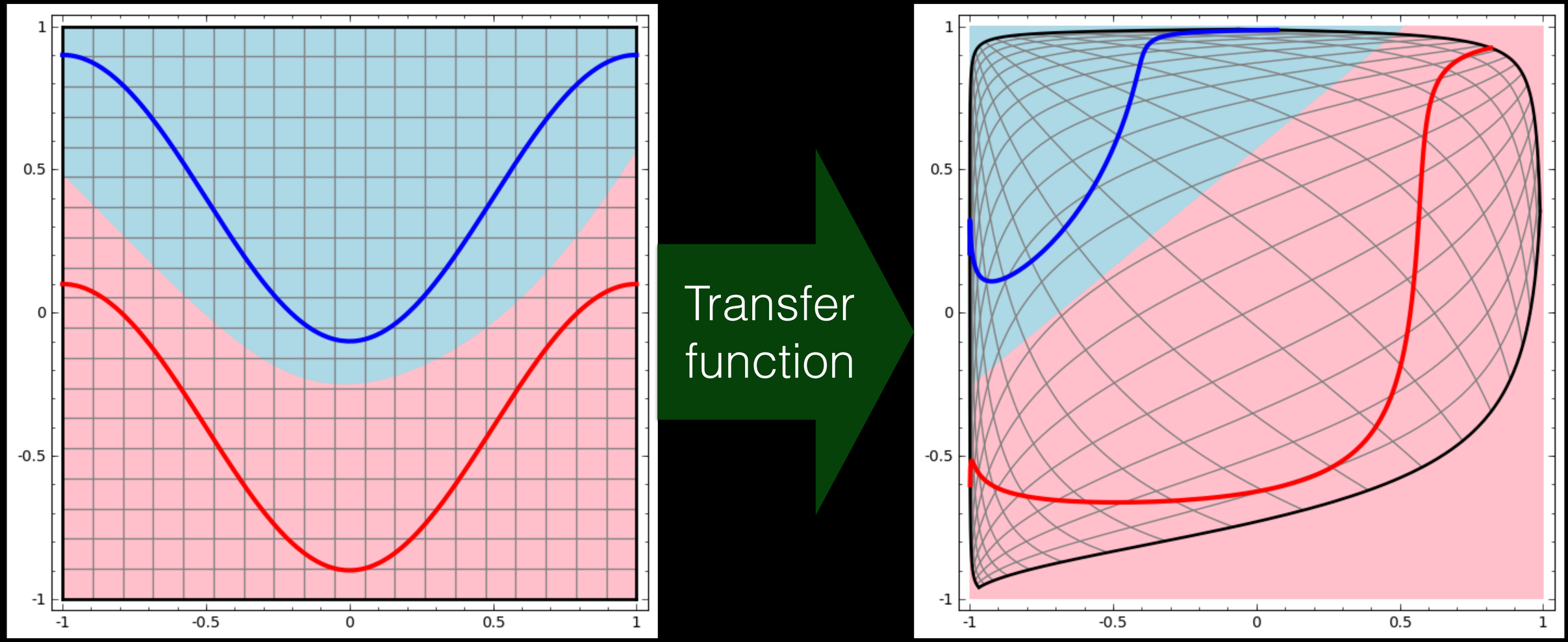






From http://colah.github.io/posts/2014-03-NN-Manifolds-Topology/

What just happened?



From http://colah.github.io/posts/2014-03-NN-Manifolds-Topology/