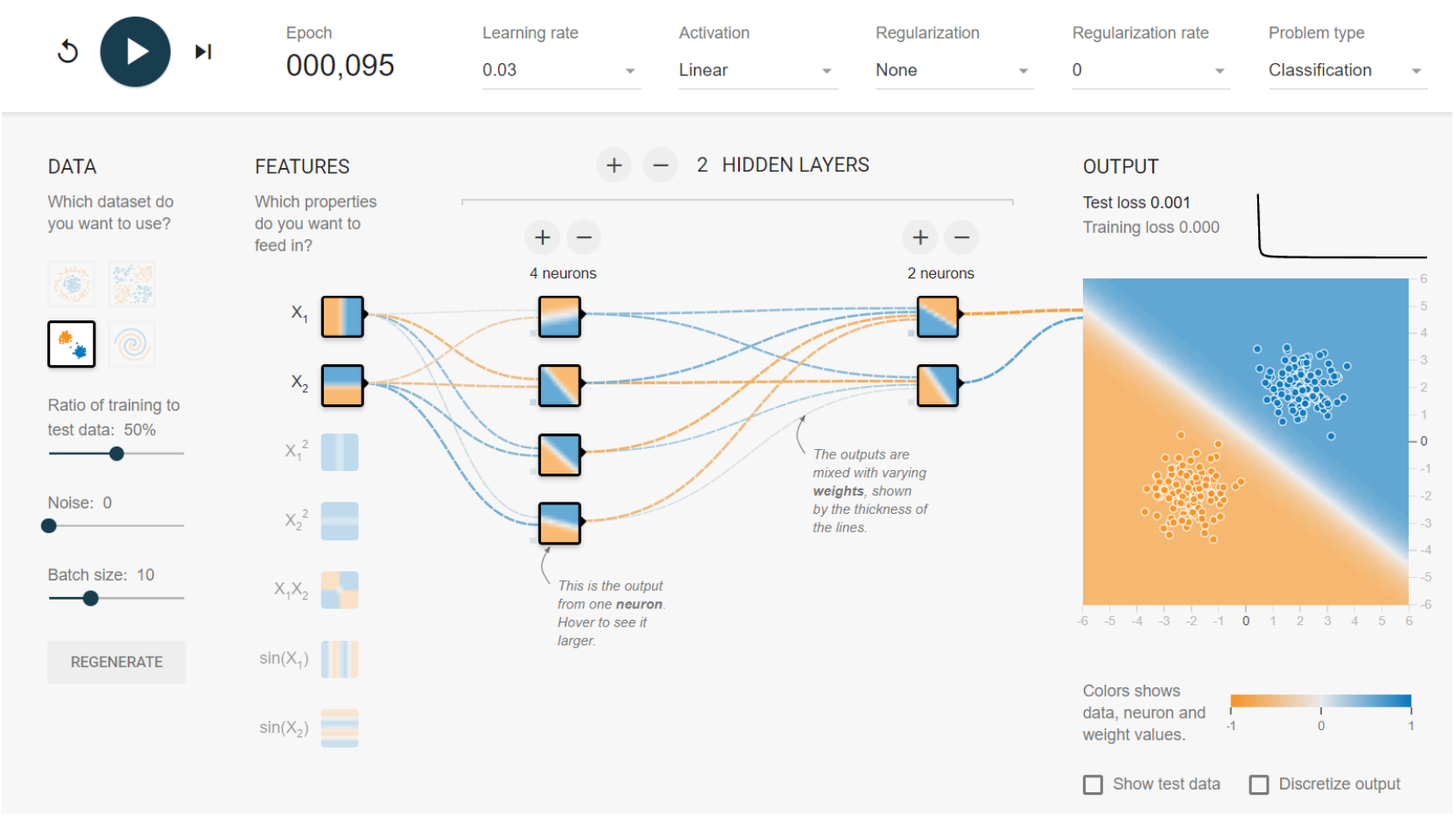
Machine Learning Using Tensorflow

Week2:
Basis of Deep Learning (1)

Shu-Ting Pi, PhD
UC Davis

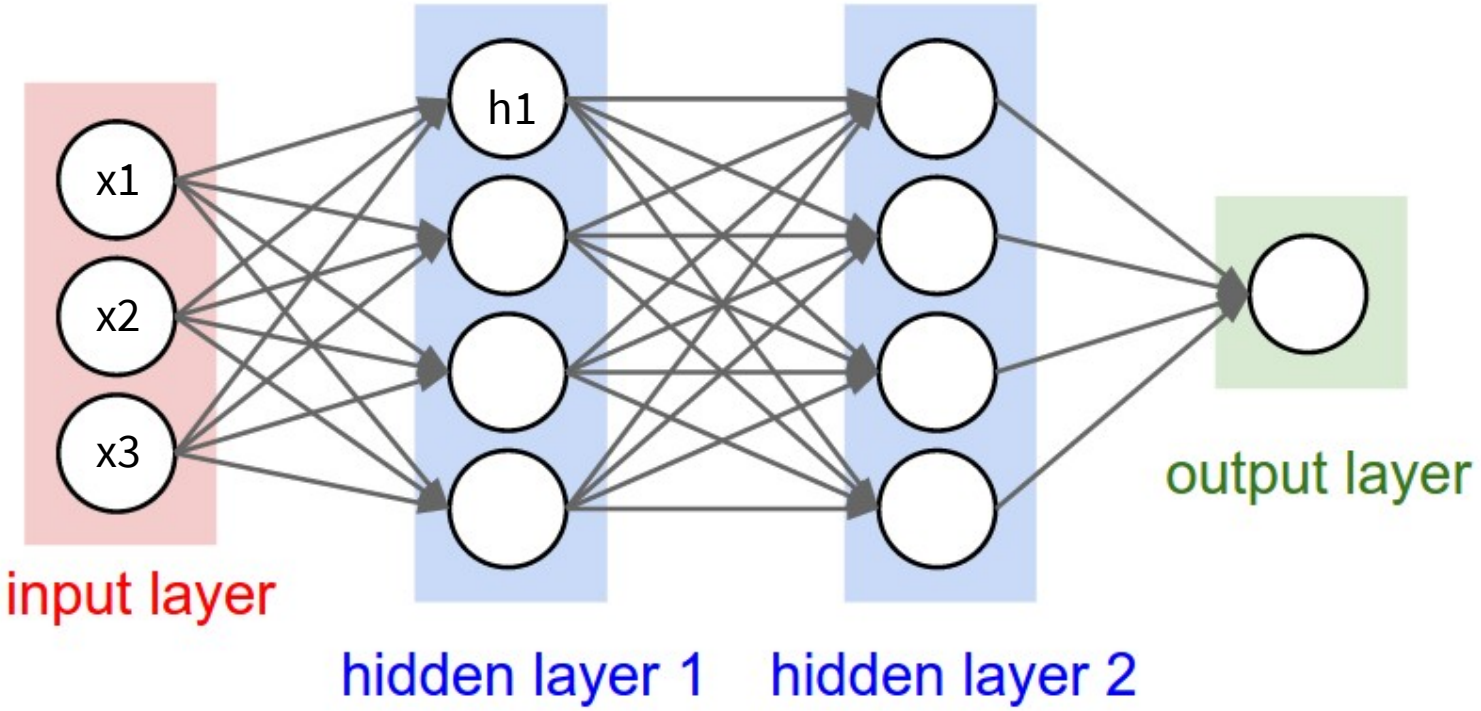
TensorFlow ™

Tensorflow Playground



TF playground is particular useful to give you some hints to fine tune the parameters of your network.

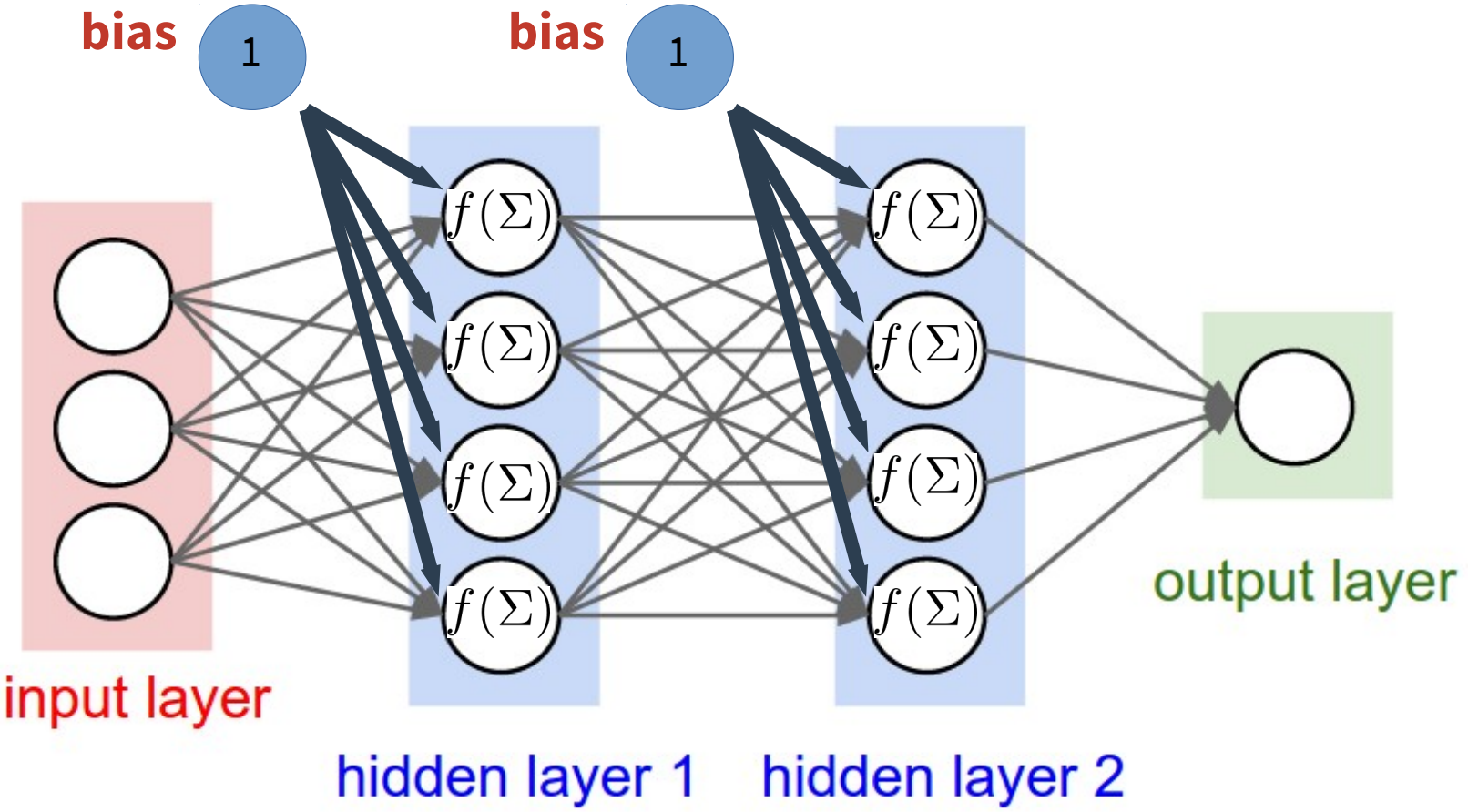
linear neural network



$$h1 = x1 \times w1 + x2 \times w2 + x3 \times w3$$

- Only works for linear separable data
- multilayer structure is meaningless !

Nonlinear neural network

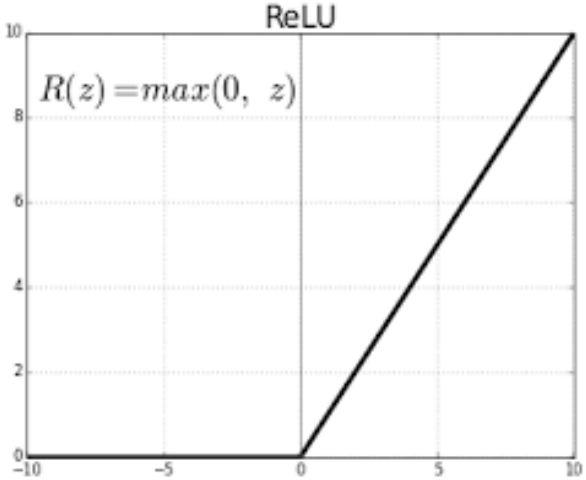


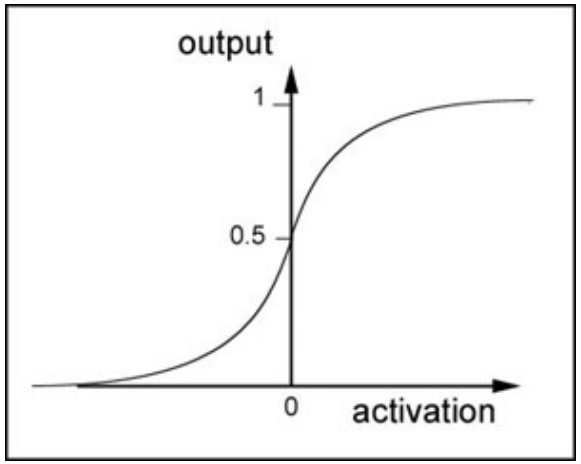
Biases: sensitivity of the node

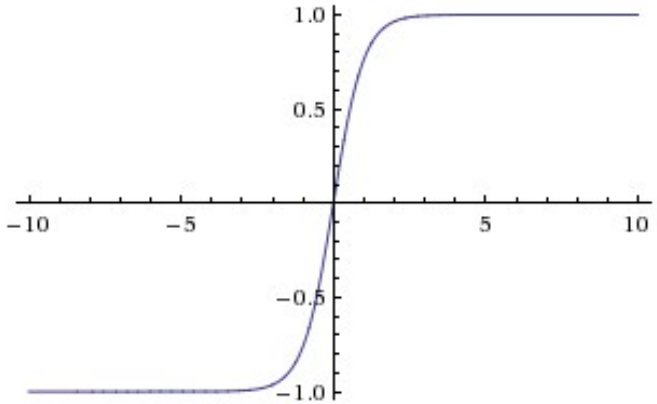
Activation function: nonlinearlize the inputs

Activation functions

How to nonlinearize? Think about human neural network!





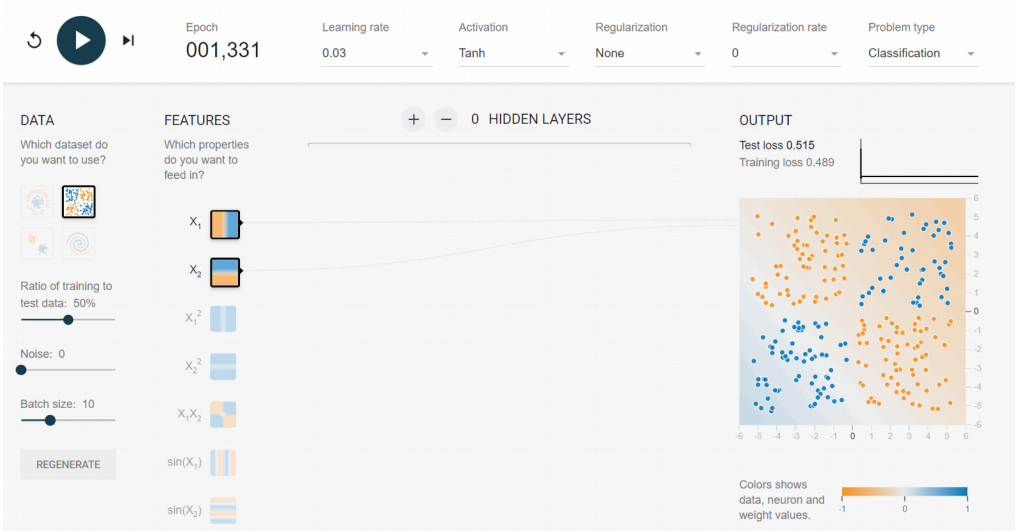


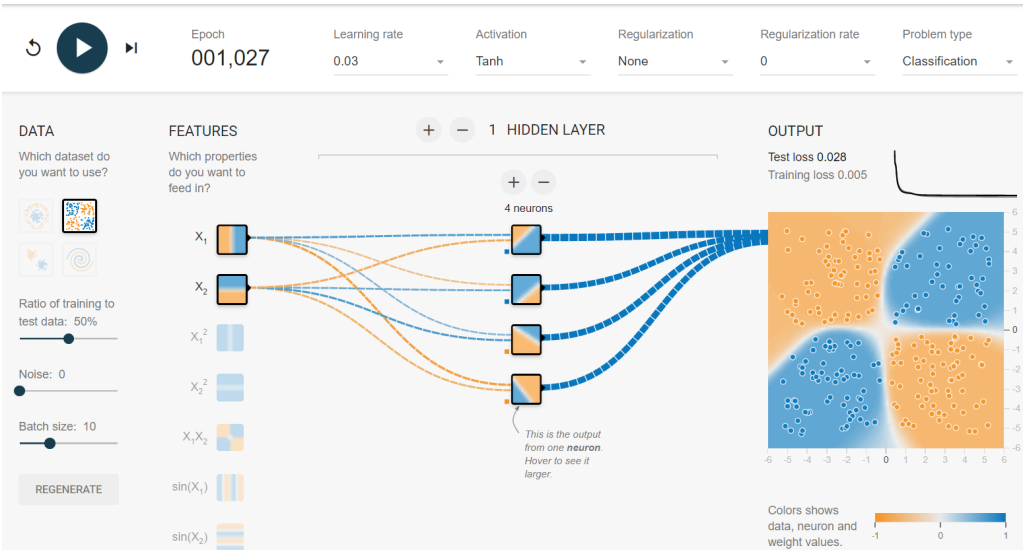
Relu **Sigmoid** **tanh**

Neural nodes become "active" after the input if higher than a threshold and become "numb" if input is too high (sigmoid & tanh).

Why multilayer?

It is proven that single layer NN can not solve “XOR” problem!



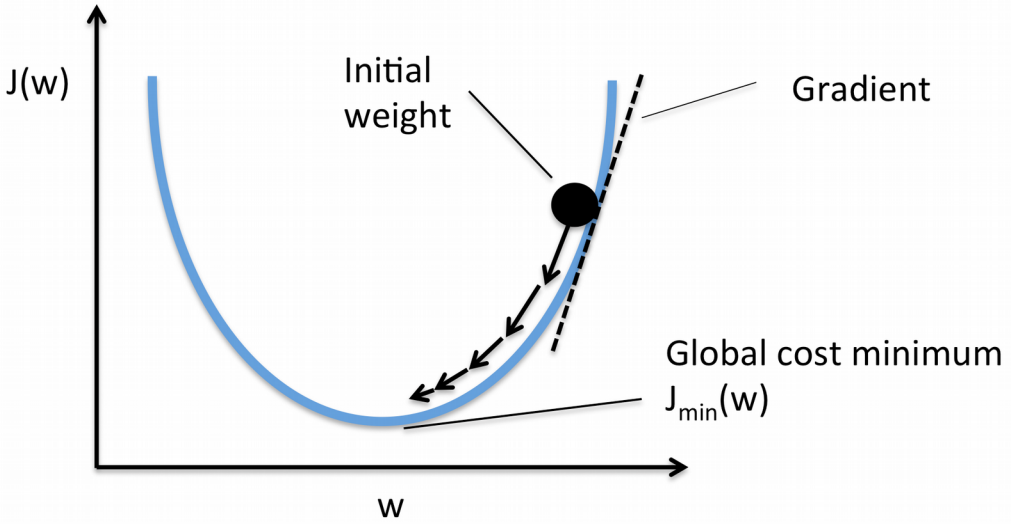


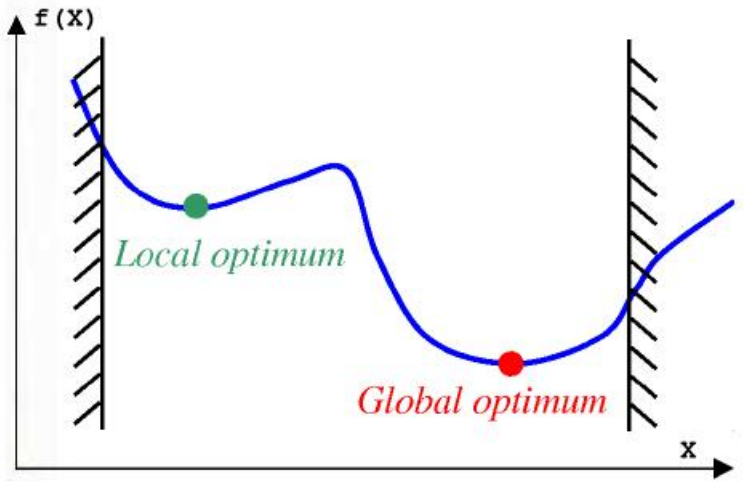
Don't dig into the math, let's prove it using tensorflow playground!

Cross Entropy

Loss functions

Optimizer





- `tf.train.Optimizer`
- ★ `tf.train.GradientDescentOptimizer`
- `tf.train.AdadeltaOptimizer`
- `tf.train.AdagradOptimizer`
- `tf.train.AdagradDAOptimizer`
- ★ `tf.train.MomentumOptimizer`
- ★ `tf.train.AdamOptimizer`
- `tf.train.FtrlOptimizer`
- `tf.train.ProximalGradientDescentOptimizer`
- `tf.train.ProximalAdagradOptimizer`
- **Alpha go!** `tf.train.RMSPropOptimizer`

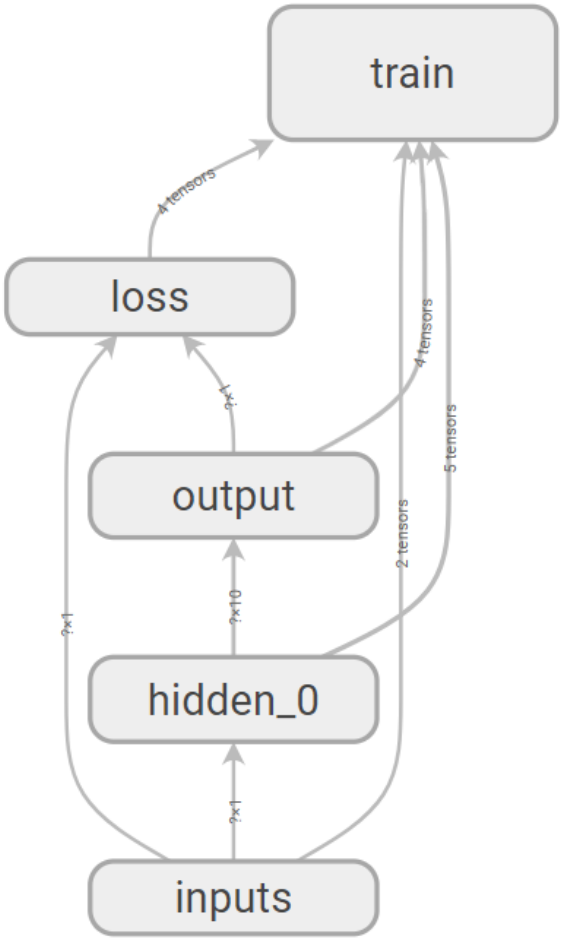
How does tensorflow work?

Construct computation graph

- tf.Variable
- tf.constant
- tf.placeholder
- tf.layers.dense
- tf.train.GradientDescentOptimizer

Make tensors flow

- tf.global_variables_initializer()
- tf.Session()
- tf.Session.run(, feed_dict={})



Let's do it !

Steps

- generate raw data, say y=x^p+x0
- Define