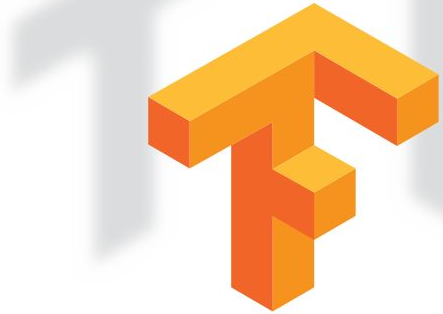


TensorFlow Tutorial

Sheng Lundquist

Introduction

C

```c
float input[100];
//Initialize array
for(int i = 0; i < 100; i++){
    input[i] = (float) 1;
}

//Multiply array by 2 element wise
for(int i = 0; i < 100; i++){
    input[i] = input[i] * 2;
}
```

Numpy

```python
import numpy as np
input = np.ones([100])
input = input * 2
```

Matlab

```matlab
input = ones(100, 1);
input = input .* 2;
```

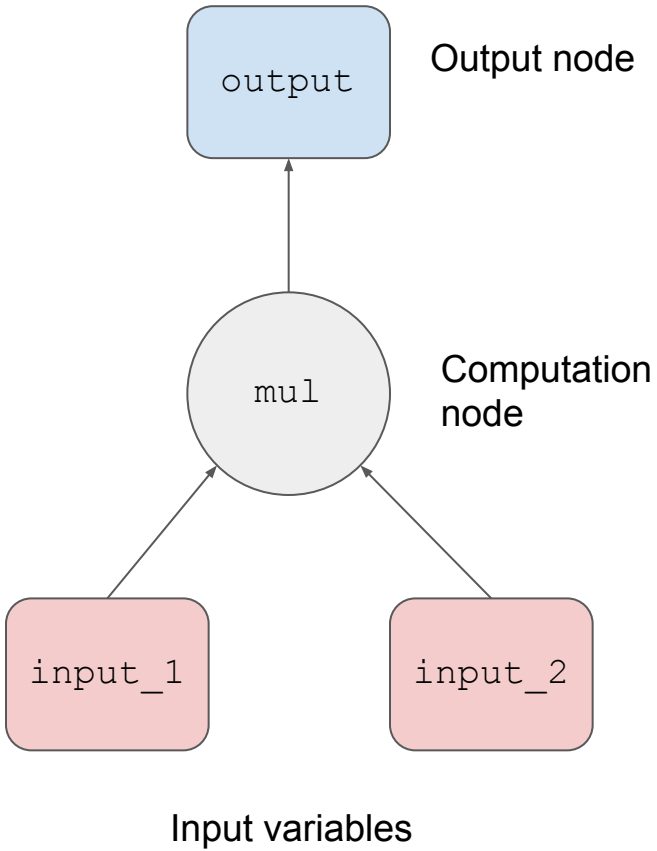
TensorFlow

- Build graph

```
import tensorflow as tf
input_1 = tf.ones([100])
input_2 = 2
output = tf.multiply(input_1, input_2)
>>> <tf.Tensor 'Mul:0' shape=(100,) dtype=float32>
```

- Evaluate graph

```
sess = tf.session()
sess.run(output)
>>> array([ 2.,  2.,  2.,  2.,  2.,  2.,  2.,  2.,  2.,
2.,  2.,  2.,  2., ...])
```



Key Advantages

Hardware abstraction

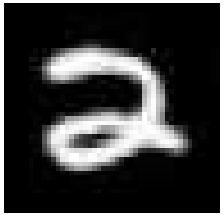
Optimizations on graph

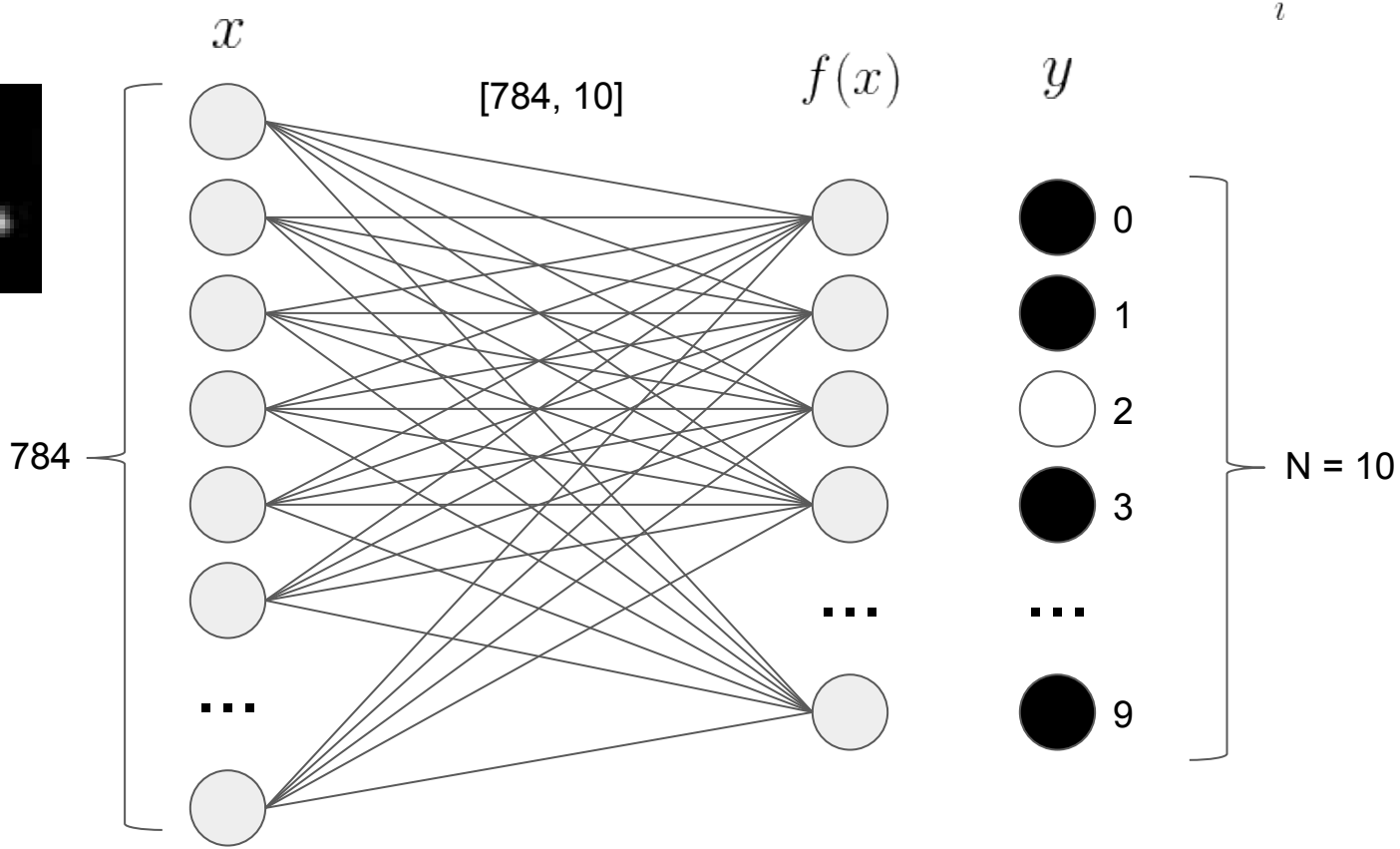
Automatic gradient calculations

Online visualization tools

Task

$$L = \frac{1}{2N}\sum_{i}^{N}(y_i - f(x)_i)^2$$





Building Graph for SLP on MNIST

```
import tensorflow as tf
```

Placeholders

```
import tensorflow as tf
inImage = tf.placeholder(tf.float32, shape=[None, 784]) #Input image is 28 x 28
gt = tf.placeholder(tf.float32, shape=[None, 10])
```

Variables

```
import tensorflow as tf
inImage = tf.placeholder(tf.float32, shape=[None, 784])
gt = tf.placeholder(tf.float32, shape=[None, 10])
W = tf.Variable(tf.zeros([784,10]))
b = tf.Variable(tf.zeros([10]))
```

Operation

```
import tensorflow as tf
inImage = tf.placeholder(tf.float32, shape=[None, 784])
gt = tf.placeholder(tf.float32, shape=[None, 10])
W = tf.Variable(tf.zeros([784,10]))
b = tf.Variable(tf.zeros([10]))
est = tf.matmul(inImage, W) + b
```

Operation

```
import tensorflow as tf
inImage = tf.placeholder(tf.float32, shape=[None, 784])
gt = tf.placeholder(tf.float32, shape=[None, 10])
W = tf.Variable(tf.zeros([784,10]))
b = tf.Variable(tf.zeros([10]))
est = tf.matmul(inImage, W) + b
loss = tf.reduce_mean(tf.square(gt - est))/2
```

Optimizer

```
import tensorflow as tf
inImage = tf.placeholder(tf.float32, shape=[None, 784])
gt = tf.placeholder(tf.float32, shape=[None, 10])
W = tf.Variable(tf.zeros([784,10]))
b = tf.Variable(tf.zeros([10]))
est = tf.matmul(inImage, W) + b
loss = tf.reduce_mean(tf.square(gt - est))/2
opt = tf.train.GradientDescentOptimizer(0.1).minimize(loss)
```

Initialization

```
import tensorflow as tf
inImage = tf.placeholder(tf.float32, shape=[None, 784])
gt = tf.placeholder(tf.float32, shape=[None, 10])
W = tf.Variable(tf.zeros([784,10]))
b = tf.Variable(tf.zeros([10]))
est = tf.matmul(inImage, W) + b
loss = tf.reduce_mean(tf.square(gt - est))/2
opt = tf.train.GradientDescentOptimizer(0.1).minimize(loss)
sess = tf.Session()
sess.run(tf.global_variables_initializer())
```

Evaluation

```
import tensorflow as tf
inImage = tf.placeholder(tf.float32, shape=[None, 784])
gt = tf.placeholder(tf.float32, shape=[None, 10])
W = tf.Variable(tf.zeros([784,10]))
b = tf.Variable(tf.zeros([10]))
est = tf.matmul(inImage, W) + b
loss = tf.reduce_mean(tf.square(gt - est))/2
opt = tf.train.GradientDescentOptimizer(0.1).minimize(loss)
sess = tf.Session()
sess.run(tf.global_variables_initializer())
for i in range(1000):
   batch_input, batch_gt = mnist.train.next_batch(256)
   sess.run(opt, feed_dict={inImage: batch_input, gt: batch_gt})
```

Evaluation

```
import tensorflow as tf
inImage = tf.placeholder(tf.float32, shape=[None, 784])
gt = tf.placeholder(tf.float32, shape=[None, 10])
W = tf.Variable(tf.zeros([784,10]))
b = tf.Variable(tf.zeros([10]))
est = tf.matmul(inImage, W) + b
loss = tf.reduce_mean(tf.square(gt - est))/2
opt = tf.train.GradientDescentOptimizer(0.1).minimize(loss)
sess = tf.Session()
sess.run(tf.global_variables_initializer())
for i in range(1000):
   batch_input, batch_gt = mnist.train.next_batch(256)
   sess.run(opt, feed_dict={inImage: batch_input, gt: batch_gt})
   print("Loss on step", i, ":", sess.run(loss, feed_dict={inImage:batch_input, gt:batch_gt}))
```

Demo

Code

https://github.com/slundqui/TFTutorial