Tensorflow Tutorial

Lesson 6

- MNIST Part 1

MNIST

- Handwritten digits
  - http://yann.lecun.com/exdb/mnist/
- Tensorflow has a function to download the dataset:

```
#Include the example
from tensorflow.examples.tutorials.mnist import input_data

#Download example and encode labels with ONE-HOT Coding
mnist = input_data.read_data_sets("MNIST_data/",one_hot=True)
```

- With „one-hot“ coding of numbers, for easier computation

- Images are 28 x 28 (but stored 1D)

One-Hot-Coding:

0 1000000000
1 0100000000
2 0010000000
3 0001000000
4 0000100000
5 0000010000
6 0000001000
7 0000000100
8 0000000010
9 0000000001

MNIST

- Mnist now holds all the needed data
  - mnist.train → Trainings data
  - mnist.validation → Validation data
  - mnist.test → Test data
- After choosing from these three above the images or the labels can be accesed via
  - *.labels
  - *.images

Function to debug or look at the data

```
def debugData(part,idx=10):

    #Look at an example image
    x = part.images[idx]
    x = x.reshape([28,28])

    y = part.labels[idx]

    return x,y
```

Look at Image

- Image is stored as float32 , means that the pixel values are between 0 and 255

- To look at the image in the command line it must be scaled to 0 to 255 with PIL

```
#Get random image to evaluate
x_train,y_train = debugData(mnist.train,idx=np.random.randint(0,5000))

#Show the example data (Execute in command line to see image)
Image.fromarray(np.asarray(x_train*255,dtype=np.uint8))
print(y_train)
```

```
from PIL import Image
import numpy as np

#Include the example
from tensorflow.examples.tutorials.mnist import input_data

#Download example and encode labels with ONE-HOT Coding
mnist = input_data.read_data_sets("MNIST_data/",one_hot=True)

#Get random example data
x_train,y_train = debugData(mnist.train,idx=np.random.randint(0,5000))
x_val,y_val = debugData(mnist.validation,idx=np.random.randint(0,5000))
x_test,y_test  = debugData(mnist.test,idx=np.random.randint(0,5000))

#Show the example data
Image.fromarray(np.asarray(x_train*255,dtype=np.uint8))
print(y_train)

Image.fromarray(np.asarray(x_val*255,dtype=np.uint8))
print(y_val)

Image.fromarray(np.asarray(x_test*255,dtype=np.uint8))
print(y_test)
```