Visualization

December 23, 2017

1 Visualizations

```
In [ ]: import numpy as np
        import theano
        import theano.tensor as T
        import matplotlib.pyplot as plt
        from sklearn.utils import shuffle
        from theano.tensor.shared_randomstreams import RandomStreams
        from util import relu, error_rate, getKaggleMNIST, init_weights
        from unsupervised import DBN
        from rbm import RBM
```

```
In [ ]: Xtrain, Ytrain, Xtest, Ytest = getKaggleMNIST()
        #if loadfile:
        #    dbn = DBN.load(loadfile)
        #else:
        dbn = DBN([1000, 750, 500, 10]) # AutoEncoder is default
        dbn = DBN([1000, 750, 500, 10], UnsupervisedModel=RBM)
        dbn.fit(Xtrain, pretrain_epochs=1)

        if savefile:
            dbn.save(savefile)

        # first layer features
        # initial weight is D x M
        # W = dbn.hidden_layers[0].W.eval()
        # for i in xrange(dbn.hidden_layers[0].M):
        #     imgplot = plt.imshow(W[:,i].reshape(28, 28), cmap='gray')
        #     plt.show()
        #     should_quit = raw_input("Show more? Enter 'n' to quit\n")
        #     if should_quit == 'n':
        #         break

        # features learned in the last layer
        for k in xrange(dbn.hidden_layers[-1].M):
            # activate the kth node
            X = dbn.fit_to_input(k)
```

```python
imgplot = plt.imshow(X.reshape(28, 28), cmap='gray')
plt.show()
if k < dbn.hidden_layers[-1].M - 1:
    should_quit = raw_input("Show more? Enter 'n' to quit\n")
    if should_quit == 'n':
        break
```