Assignment 1

1 Part 1

You are supposed to implement a convolutional neural network for MNIST in TensorFlow. Your solution should contain the following features:

- it should contain both convolutional layers and fully connected layers,
- it should contain batch normalization for convolution, which you implement by basic tensor operations (without using tf.nn.moments) - it is fine if you use the train mode even for the test check,
- implementation should allow painless changes in the architecture, like adding new layers,
- you need to obtain at least 99.1% test accuracy reliably in front of a person checking your solution.

2 Part 2

In the second part of the problem the goal is to generate 10 images that give the highest probability for each of the output class. Hint: here the weights from Part 1 are frozen and they are considered as constants, whereas the values of pixels are now new variables and we are optimizing over those new variables.

This part should use a checkpoint with weights from Part 1. Visualize your obtained images.

3 Additional solution features (nonobligagory)

If you want, you can additionally implement:

- dropout - check how it works with and without batch normalization,
- data augmentation,
- try different learning algorithms and different learning rates.

4 Deadline

You should submit your solution by email by 23:59 on 09.05.2017 to `cygan@mimuw.edu.pl` with email title "Assignment 1 - Deep neural networks". Your code will be inspected either during lab session on 10.05 or 17.05.