Logistic regression in TensorFlow

a) Open the notebook log_reg_challenger and run the fist 3 cells. We predict the the probability $p(y_i = 1|x_i)$ with the formula:

$$p(y_i = 1|x_i) = \frac{e^{(b+W'x_i)}}{1 + e^{(b+W'x_i)}} = \frac{1}{1 + e^{-(b+W'x_i)}}$$

Look at the predicted $p(y_i = 1|x_i)$ values with our given start parameters $W = -0.2$ and $b = 20$. What do you observe?

b) Now lets try to find better values for W and b. Lets assume W is given with -1. We want our probability $p(y_i = 1|x_i)$ to be 0.5. What is the value for b in this case.
Hint: at which x value should $p(y_i = 1|x_i)$ be 0.5, look at the data. $1 + e^{-(b+W'x_i)}$ must be 2.

c) Run the TensorFlow forward pass in cell 5 and optimize the values for W and b in cell 6.
Fetch the loss, W and b and print the final values.
Hint: You can't use the same names for the results of your fetches as you have used for the TensorFlow graph. See cell 5.

Multinomial Logistic Regression on MNIST dataset

a) Open the notebook Multinomial Logistic Regression. In this notebook we use multinomial logistic regression to predict the handwritten digits of the MNIST dataset.
We have 4000 examples with 784 pixelvalues and 10 classes. Run the fist 3 cells and explain the OneHot encodig. In TensorFlow we need to use OneHot encodig.

b) Write the missig TensorFlow code in cell 4 to do a matrix multiplication between x and w and then an addtion of b
$z = x * w + b$

c) Run the next two cells to store the graph and do a foreward pass of the untrained netword, look at the probability for each class of some examples.

d) Now lets train the model. We use a minibatch size of 128 and use the fist 2400 examples for the training.
The validation set will be the examples from 2400 to 3000. Write the code to get the loss and the probabilities of your validation set. Run the last cells to check the perfomace of the model and to get the probability of a random example of the validation set.