Exercise 47-48

Question 1

The assumption *«The inputs/features distributed according to a normal/gaussian distribution»* does not fit with the ordinary least square linear regression.

Question 2

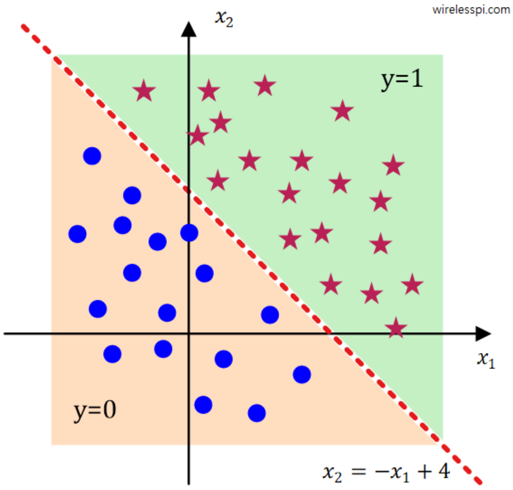
The mean squared error is convex since it has one global minimum and no local minima. This is because the MSE is a square function of the parameters.

Question 3

Logistic regression does not have a analytic closed form solution. With logistic regression, there is a need for iterative optimization for B until it converges, instead of a formula like OLS have.

Question 4

True, the logistic regression produces a linear decision boundary. Since the formula for logistic regression creates this output:



Figur 1: https://wirelesspi.com/logistic-regression-in-machine-learning

We see that the regression have a boundary where y= 0.5, creating a linear boundary between the y>0.5 and y<0.5 values.

Question 5

For binary classification, logistic regression is preferred over linear regression since they just produce numbers from 0 to 1, while linear regressions can produce –1 to 1, known as negative correlating. This makes logistic regression way more meaningful for probalility estimates, since -1 -> 1 simply does not make as much sense in this regard as 0 -> 1.

The linear regression model is not suitable since binary classification search for a “Yes or No” answer. While its great for housing prices, temperature etc where it can predict higher numbers, it is not so good for the smaller binary classification problems. For example if the true value is 10, and the linear regression gave 9 -> small error. In other words, linear regression treats “Yes or No” as normal numbers and not categories.

Question 6

The fourth option is not true, as the loss surface of a deep neural network can have many minimums, saddle points and flat regions.

Question 7

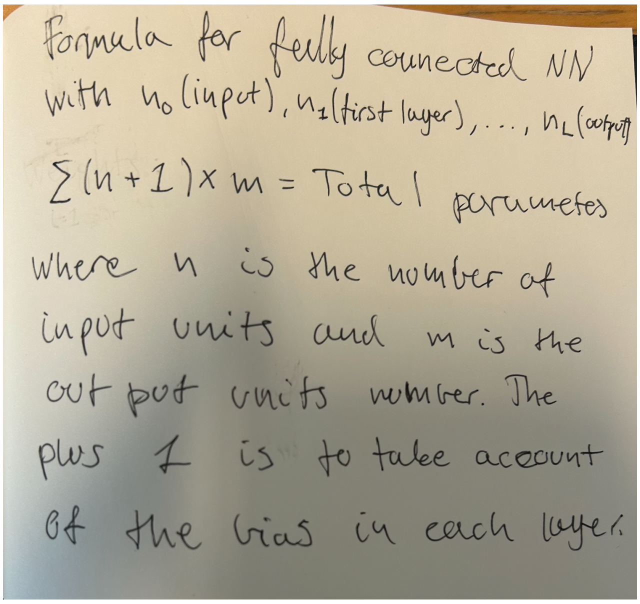
Sigmoid is usually best for neural network with 1-2 hidden layers. This is because the sigmoid activation function multiplies the gradients in each node of the layer with 0.25. So during backpropogation, it can lead to gradients that vanishes after 0.25 x 0.25 x 0.25…etc. The multiplication of 0.25 makes the gradients so small that they “vanishes” and become inconsequential for the learning, resulting in a model that learns very slow or not at all. For this reason, activation like ReLu are preferred for deeper networks.

The statement is true.

Question 8

As mentioned in Question 7, the vanishing gradient problem is a result of sigmoid multiplication and backpropagation. The most common technique to mitigate it is to use ReLU instead for deeper networks. But it is also possible to adjust the weights so that they’re not to small (shrinks to zero) or to big (explodes).

Question 9



Formula for fully connected NN with n_0 (input), n_1 (first layer), ..., n_L (output

$\sum (n+1) \times m$ = Total parameters

Where n is the number of input units and m is the out put units number. The plus 1 is to take account of the bias in each layer.