Homework 4
Harshvardhan Kalra
2014043

The general steps performed in the assignment were:
i) Choosing an appropriate optimizer. I tried all optimizers provided with the Keras library, and chose Adadelta, as it gave the best results for the specified learning rates.
ii) Plotting the confusion matrix. The following site was used as reference for plotting the confusion matrix :
http://scikit-learn.org/stable/auto_examples/model_selection/plot_confusion_matrix.html.

The test accuracies were as follows:
Test accuracy for learning rate 0.1: 94.9%
Test accuracy for learning rate 0.3: 97.44%
Test accuracy for learning rate 0.5: 97.97%
Test accuracy for learning rate 0.7: 98.22%

Similarly, for the autoencoders, the following site was used as reference:
https://blog.keras.io/building-autoencoders-in-keras.html

The optimizer used was Adadelta for both the encoder and the trained model, and the results were as follows :
Test accuracy for learning rate 0.5: 85.78%
Test accuracy for learning rate 0.7: 90.13%
Test accuracy for learning rate 0.9: 91.81%
Test accuracy for learning rate 1.1: 92.58%
Test accuracy for learning rate 1.3: 93.09%

Please find the models and images in the corresponding folders.

1) $\tanh(a) = \frac{e^a - e^{-a}}{e^a + e^{-a}}$

Since $e^a > 0$, dividing numerator and denominator by e^a, we get

$$\tanh(a) = \frac{1-e^{-2a}}{1+e^{-2a}} = \frac{2-1-e^{-2a}}{1+e^{-2a}}$$

$$= \frac{2}{1+e^{-2a}} - \frac{1+e^{-2a}}{1+e^{-2a}}$$

$$= 2\sigma(2a) - 1$$

$$\Rightarrow \sigma(2a) = \frac{\tanh(a)+1}{2}$$

$$\Rightarrow \sigma(a) = \frac{\tanh(a/2)+1}{2}$$

Hence, $y_k(x,w) = \sigma\left(\sum_{j=1}^{m} w_{kj}^{(2)} h\left(\sum_{i=1}^{D} w_{ji}^{(1)} x_i + w_{j0}^{(1)}\right) + w_{k0}^{(2)}\right)$

$$= \frac{\tanh\left(\sum_{j=1}^{m} w_{kj}^{(2)} h\left(\sum_{i=1}^{D} w_{ji}^{(1)} x_i + w_{j0}^{(1)}\right) + w_{k0}^{(2)}\right)}{2}$$

$+1/2$ (Bias)

And therefore, there exists a corresponding neural net with a relabelling of classes computing the same function.

2) (i) $\theta_0 = -30$

→

X_1	X_2	X_1 and X_2
0	0	0
0	1	0
1	0	0
1	1	1

→ For both inputs being 1, the sigmoid should return 1, else 0.

Since $\sigma(x) \approx 1$ only when x is a large positive value,

$\sigma(1\cdot\theta_1 + 1\cdot\theta_2 - 30) = 1$

$\Rightarrow \theta_1 + \theta_2 - 30 > 0 \Rightarrow$ Using $\theta_1 = \theta_2 = 20$ as works as:

$\cancel{\sigma\theta}\ \sigma(20\cdot 0 + 20\cdot 0 - 30) = \sigma(20\cdot 1 - 30)$

$= -1$

ii) $X_1 \text{ NOR } X_2 = \text{NOT}(X_1 \text{ OR } X_2)$

$= \text{NOT} X_1 \text{ AND NOT} X_2$ (negating)

Since the ~~e~~ inputs are negated, flipping the weights from (i) will work. ~~However, the A~~

$\cancel{\Rightarrow \theta_1 = \theta_2 = 50, \theta_3 = 30}$

$\cancel{\Rightarrow \sigma(\theta_1 + \theta_2 x_2 + \theta_3 x_3) \text{ becomes:}}$

$\cancel{\sigma(-30) \text{ for } x_1 = x_2 = 0}$

$\Rightarrow \theta_1 = \theta_2 = -30,\ \theta_0 = 20$

$\Rightarrow \sigma(20) \approx 1 \quad (x_1 = x_2 = 0)$

$\sigma(20-30) = \sigma(-10) \approx 0 \quad (x_1 = 0 \text{ or } x_2 = 0$ but not both$)$

$\sigma(20-60) = \sigma(-40) \approx 0$

$\Rightarrow$ Hence the above weights emulate a nn for $X_1 \text{ NOR } X_2$.

(iii) $f = (X_1 \wedge X_2 \wedge X_3) \vee (X_2 \wedge X_4) \vee (X_1 \wedge X_4)' \vee (X_2 \wedge X_3)' \vee (X_1 \wedge X_2 \wedge X_3 \wedge X_4)$

$= (X_1 \wedge X_2 \wedge X_3) \vee (X_2 \wedge X_4) \vee (X_1' \vee X_4') \vee (X_2' \vee X_3') \vee (X_1 \wedge X_2 \wedge X_3 \wedge X_4)$

$= 1$ as last three terms reduce to $z \vee z'$, which is always ~~0~~ 1.