1.

a. Done in python

b. Done in Python

```
Epoch 1/5
1875/1875 [==============================] - 11s 6ms/step - loss: 0.1960 - accuracy: 0.9417
Epoch 2/5
1875/1875 [==============================] - 10s 5ms/step - loss: 0.0803 - accuracy: 0.9744
Epoch 3/5
1875/1875 [==============================] - 10s 5ms/step - loss: 0.0556 - accuracy: 0.9823
Epoch 4/5
1875/1875 [==============================] - 11s 6ms/step - loss: 0.0407 - accuracy: 0.9869
Epoch 5/5
1875/1875 [==============================] - 11s 6ms/step - loss: 0.0335 - accuracy: 0.9891
Evaluating MLP2 on test set 1
313/313 [==============================] - 1s 3ms/step - loss: 0.0875 - accuracy: 0.9766
Evaluating MLP2 on test set 2
313/313 [==============================] - 1s 3ms/step - loss: 2.3198 - accuracy: 0.5935
Epoch 1/5
1875/1875 [==============================] - 90s 48ms/step - loss: 0.1074 - accuracy: 0.9667
Epoch 2/5
1875/1875 [==============================] - 90s 48ms/step - loss: 0.0387 - accuracy: 0.9880
Epoch 3/5
1875/1875 [==============================] - 89s 48ms/step - loss: 0.0236 - accuracy: 0.9924
Epoch 4/5
1875/1875 [==============================] - 90s 48ms/step - loss: 0.0189 - accuracy: 0.9941
Epoch 5/5
1875/1875 [==============================] - 90s 48ms/step - loss: 0.0154 - accuracy: 0.9950
Evaluating CNN on test set 1
313/313 [==============================] - 4s 13ms/step - loss: 0.0361 - accuracy: 0.9895
Evaluating CNN on test set 2
313/313 [==============================] - 4s 13ms/step - loss: 0.4063 - accuracy: 0.9029
```

c. For test set 1, the accuracy was 0.9766 while for test set 2, the accuracy was 0.5935
d. For test set 1, the accuracy was 0.9895 while for test set 2, the accuracy was 0.9029
e. For me, the CNN proved to have a better accuracy on both tests in comparison to the Feedforward network. This is because for a CNN, we are using fewer parameters and have sparser connections between nodes which leads to our network being more efficient at classifying the images than a feedforward network which will need a lot more calculations due to the larger number of parameters it will take.

3. Here we use Bayes Rules to get the Posterior Model Probability.
Recall Bayes Rules is $P(\theta|D) = \frac{p(D|\theta)P(\theta)}{p(D)}$
$P(\theta_1) = P(\theta_2)$ and $P(\theta_3) = 3\,P(\theta_1)$.
Therefore: $5\,P(\theta_1)=1$
$P(\theta_1) = P(\theta_2) = 0.2$ and $P(\theta_3)=0.6$

P(D) $=\sum_{i=\{1,2,3\}} P(\theta_i)P(D|\theta_i) = P(\theta_1)P(D|\theta_1) + P(\theta_2)P(D|\theta_2) + P(\theta_3)P(D|\theta_3)$
= 0.2* 0.00084 + 0.2*0.00105 = 0.6*0.00007
= 0.00042

Therefore, the Posterior Probabilities for each model are
$p(\theta_1|D) = \frac{P(\theta_1)P(D|\theta_1)}{p(D)}$ = 0.2* 0.00084 / 0.00042 = 0.4

I have not collaborated with anyone

$p(\theta_2|D) = \frac{P(\theta_2)P(D|\theta_2)}{p(D)}$ = 0.2* 0.00105 / 0.00042 = 0.5
$p(\theta_3|D) = \frac{P(\theta_3)P(D|\theta_3)}{p(D)}$ = 0.6* 0.00007 / 0.00042 = 0.1

4. You would be better of selling as we first take the argmax $_\theta\ P(\theta|D)$ to get our selected θ which, based off the previous question, is θ_2 . Given theta two, the prediction we get is $p(y_{t+1} \mid y_t, \theta_2)$ = .4 . As 0.4<0.5, it would be better to sell.

5. Based on the posterior predictive distribution, we would be better off buying as to get the posterior predictive distribution, we first do the calculation $\sum_\theta p(\theta|D)p(y_{t+1}|y_t, \theta)$ = 0.75*0.4+0.5*0.5+0.1*0.6= 0.56. As 0.56>0.5, we would be better of buying.

I have not collaborated with anyone