3.

(a) There are 25 variables in each data. There are 2 possible classes, thus it could be represented as binary. First, I rearrange the dataset. Each observation is rearranged to 1 by 25 matrix. Then, I aggregate all of them in to one dataset. Thus, this would be 10 X 25 matrix. Because it is the classification problem, I used the logistic regression. Because it is classification problem, this model is using the sigmoid function, which could give almost 0 or 1 result. I used the gradient descent to find the best coefficients. The gradient could be calculated as

[prediction (use sigmoid function) - y]\*X

I updated the value by using 0.3 learning rate.

If we applied this beta to the trainset (X), it predicts the class perfectly.

I predict the unknown observations (Mystery), and they are classified as [B,A,B,A,B].

It makes sense because Class A tend to aggregated in upper left side of 5X5 matrix and Class B tend to aggregated in bottom right side of the matrix.

(b) The best way to avoid the overfitting problem is reducing the dimension. In this example, there are too many variables compare to the observations. Thus, like the method used in neural networks, compressing the 5 by 5 matrix's into 4 by 4 matrix (. For example, if I compressed it, the information would be the sum of the 2 by 2 matrix's elements (There are 16 2X2 matrixes). The changed matrix will be displayed as [[3,3,1,0],[2,3,1,0],[2,1,0,0],[1,0,0,0]]. Now, the dimension is reduced from 25 to 16. If we use the 3 by 3 matrix instead, then the dimension will be reduced to 9. However, if we reduce the dimension, it loses some information. However, for this case, because there are only 10 observations, it is better to reduce the dataset into 3X3 matrix.

(c) This case is classification problem with binary predictors. The best method in this situation is Bernoulli Bayes method. By using the prior probability, we could possibly expect the value. If the all predictors are independent, the formula for posterior probabilities would be P(class = A| X) = P(Class = A) \* Multiply while i = 1~ n [P(xi | Class=A)]

Thus, we could choose the class which has the biggest probability. This is the Bernoulli case, so we can compute p(x | class = A) = prob(class = A when xi) \*\* xi \* (1 - prob(class = A when xi) \*\* (1-xi)

prob(class = A when xi) can be calculated as the adding the 5 class A observations and divide it by 5. The issue here is that if we use 0 or 1, it would cause the error. So, I used 0.001 and 0.999 instead of 0 and 1. The result is same with the logistic regression's result, [B, A,B,A,B].