Name and Student ID:

Machine Learning BLG527E, Dec 25, 2014, 120mins, Final Exam (PART A)
Signature:
Duration: 120 minutes.
Closed books and notes. Write your answers neatly in the space provided for them. Write your name on each sheet. Good Luck!

Q1	Q2	Q3	Q4	Q5	Q6	Q7	TOTAL
20	25	10	5	5	15	20	100

QUESTIONS

Q1) [20pts]
Given the following HMM with N=2 states, M=3 observations from {Green,Red,Blue} for each state and the following parameters:

$\Pi = [0.2, 0.8]^T$ $\quad A = \begin{bmatrix} 0.6 & 0.4 \\ 0.5 & 0.5 \end{bmatrix}$ $\quad B = \begin{bmatrix} 0.2 & 0.3 & 0.5 \\ 0.1 & 0.5 & 0.4 \end{bmatrix}$

Given the following sequence of observations O={Red, Red}, Compute P(O|A,B, π)
Hint: The forward and backward variables in an HMM are calculated as follows:

$\alpha_t(i) \equiv P(O_1 \cdots O_t, q_t = S_i \mid \lambda)$

Initializa tion:

$\alpha_1(i) = \pi_i b_i(O_1)$

Recursion :

$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i) a_{ij} \right] b_j(O_{t+1})$

$P(O \mid \lambda) = \sum_{i=1}^{N} \alpha_T(i)$

$\beta_t(i) \equiv P(O_{t+1} \cdots O_T \mid q_t = S_i, \lambda)$

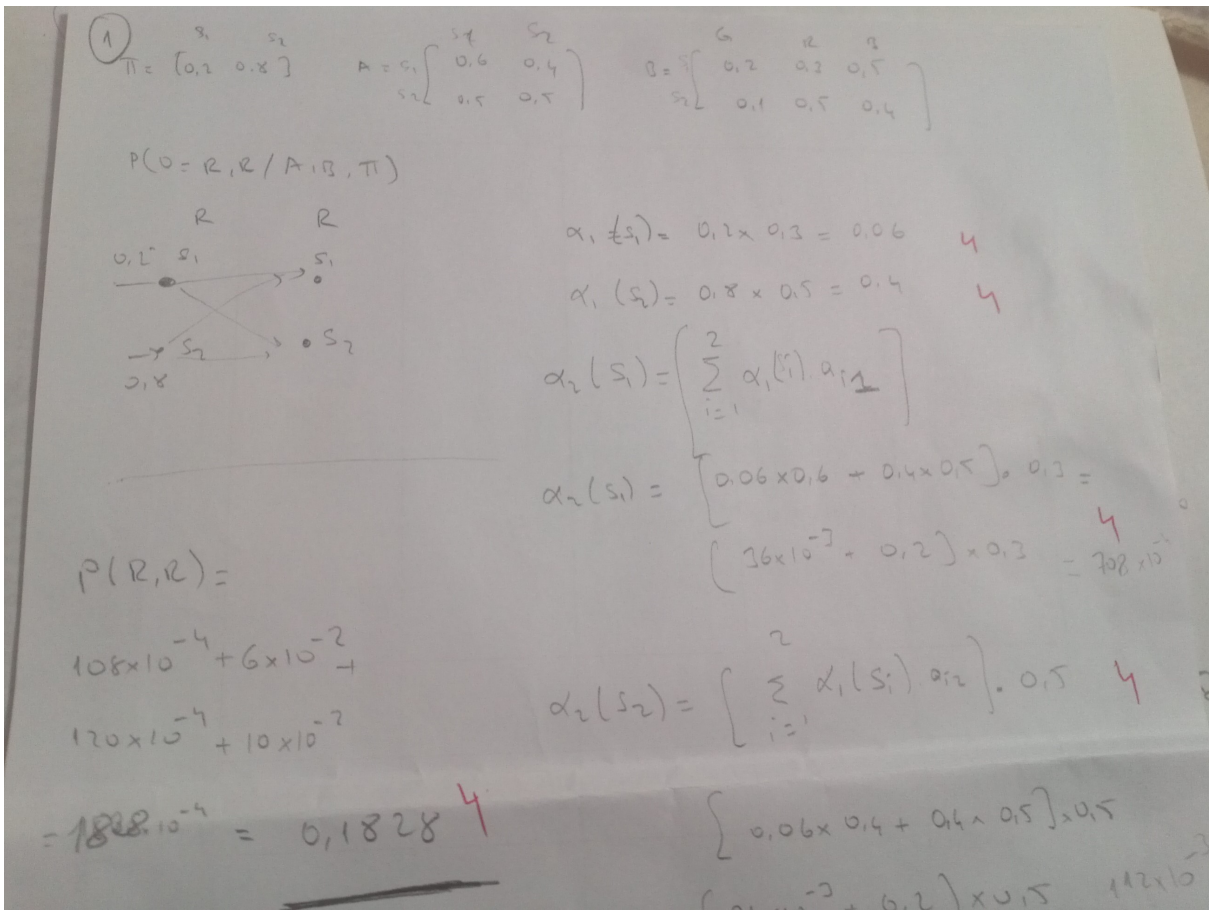
Initializa tion:

$\beta_T(i) = 1$

Recursion :

$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)$

Name and Student ID:



Q2) [25pts]

You need to classify the the following dataset using a linear SVM classifier.

X = [(-1, 1),
(1,-1),
(-1,-1),
(1,1)] ;

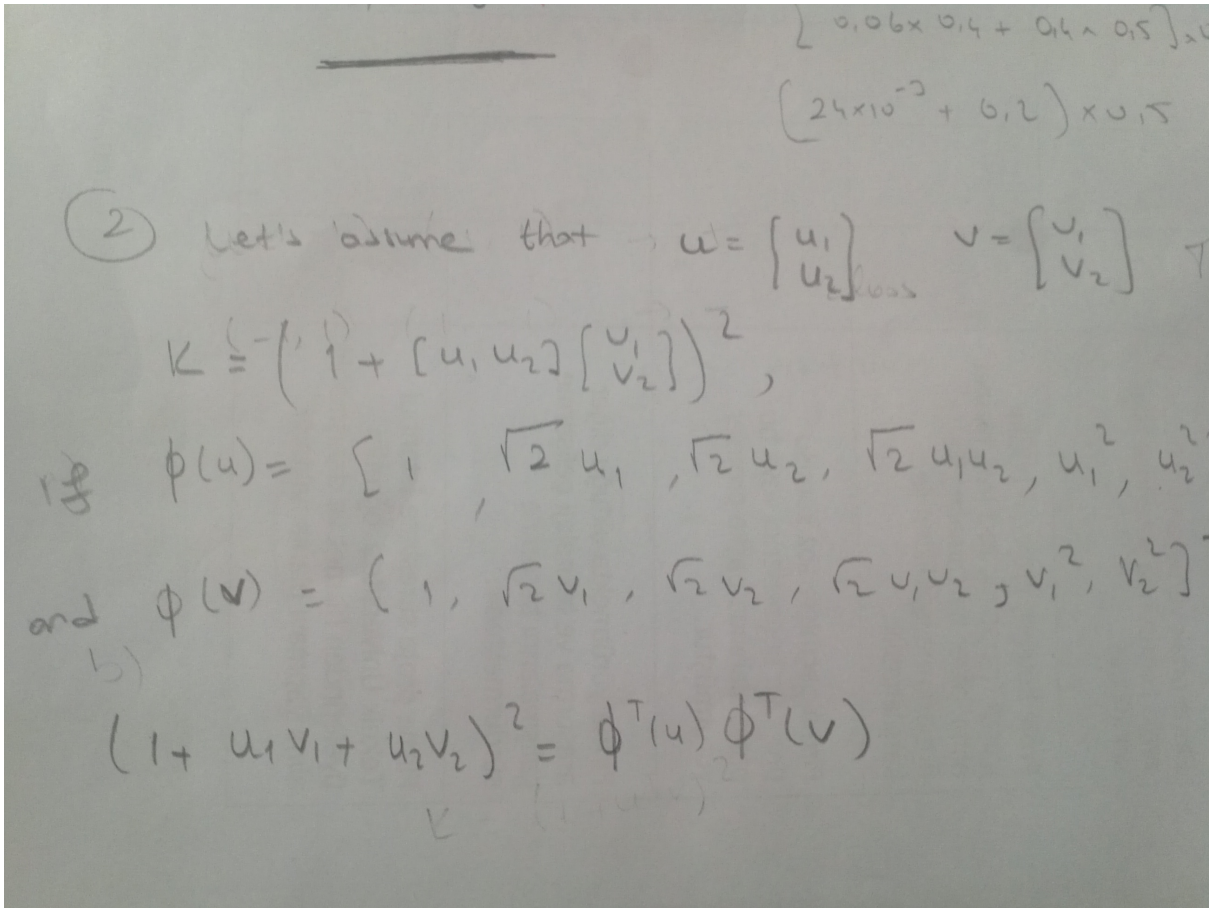
r=[-1,
-1,
1,
1]

Q2a) [10] A kernel function $K(\mathrm{u},\mathrm{v})$is a valid kernel if it can be written as a dot product in a transformed input space, i.e. K(u,v)=$K(\mathrm{u},\mathrm{v}) = \phi(\mathrm{u})^T\phi(\mathrm{v})$.

For 2 dimensional vectors u and v, show that the following is a valid kernel:

$K(\mathrm{u},\mathrm{v}) = (1+\mathrm{u}^T\mathrm{v})^2$

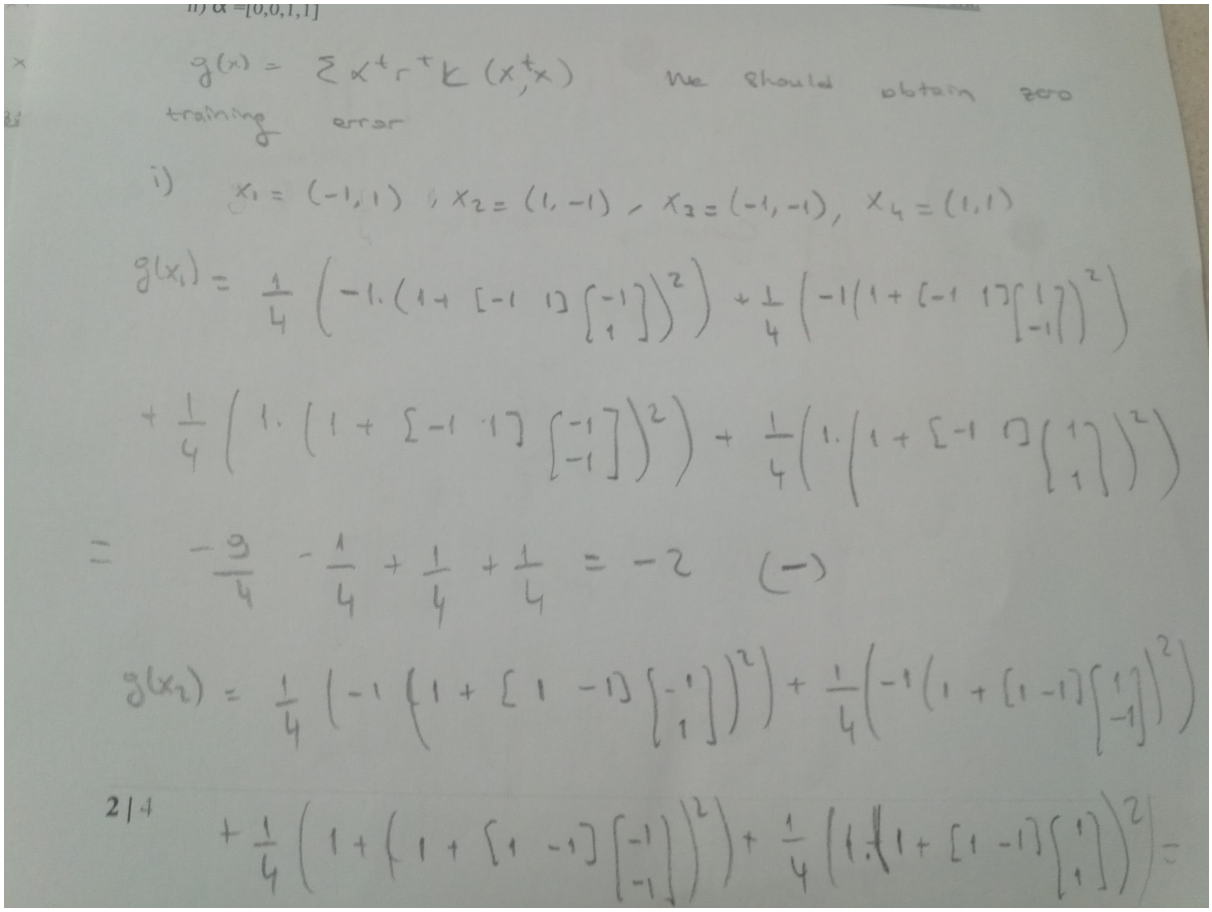
Name and Student ID:



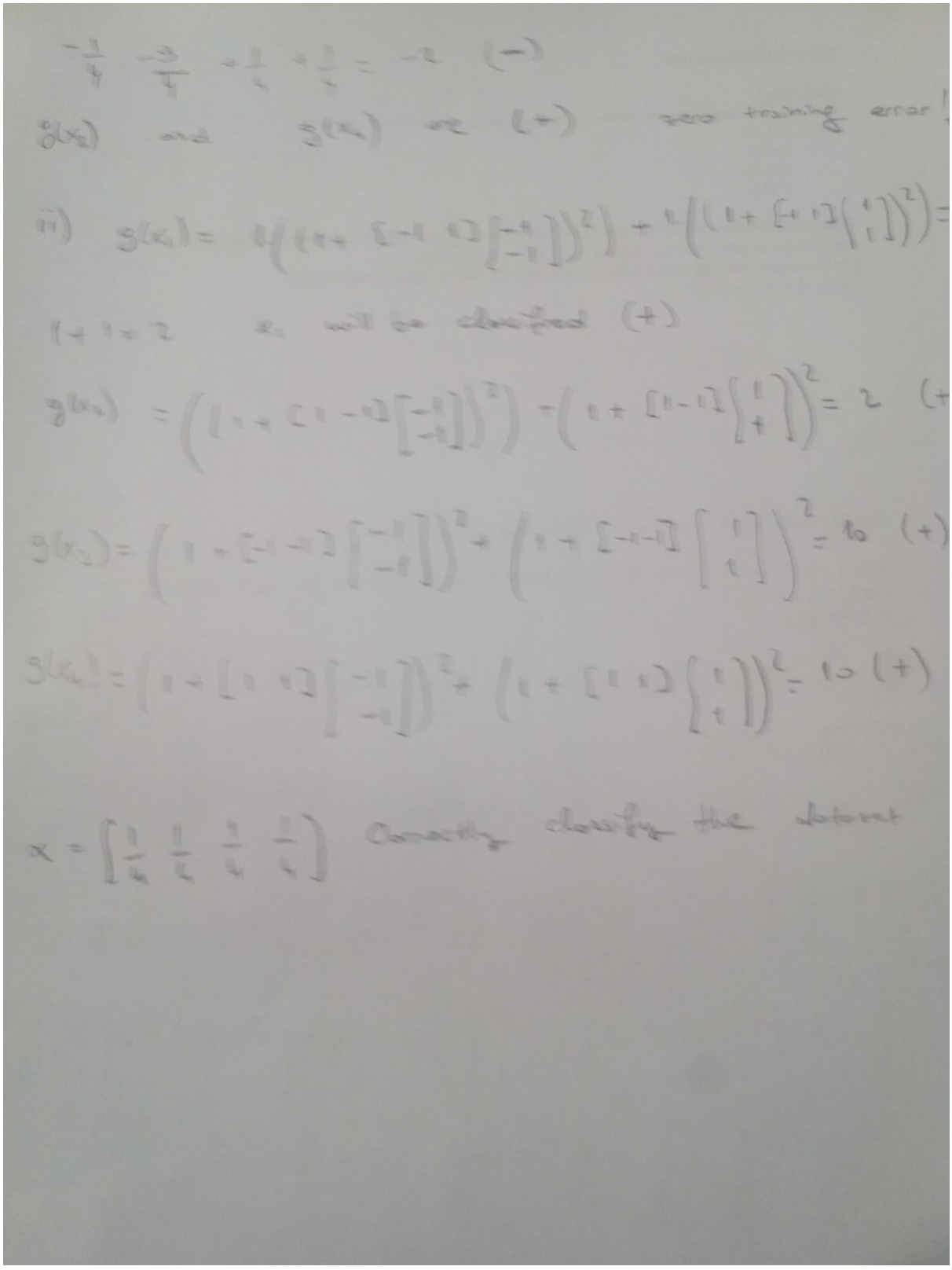
Q2b) [15pts] The SVM solution is the following function: $g(\mathbf{x}) = \sum_t \alpha^t r^t K(\mathbf{x}^t, \mathbf{x})$

Which one of the solutions for α^t, t=1,..,4 correctly classify the dataset X,r? Show your work.
i) α =[1/4, 1/4, 1/4, 1/4]
ii) α =[0,0,1,1]

Name and Student ID:



Name and Student ID:



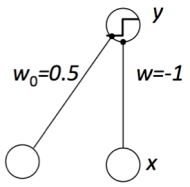
Name and Student ID:

Machine Learning BLG527E, Dec 25, 2014, 120mins, Final Exam (PART B)
Signature:
Duration: 120 minutes.
Closed books and notes. Write your answers neatly in the space provided for them. Write your name on each sheet. Good Luck!

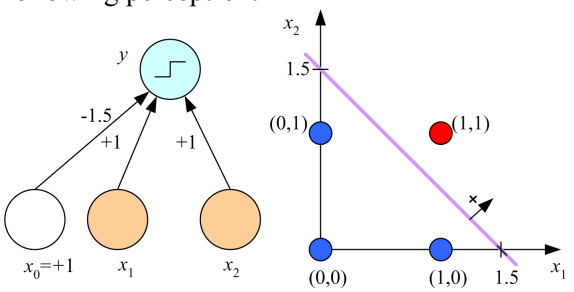
Q1	Q2	Q3	Q4	Q5	Q6	Q7	TOTAL
20	25	10	5	5	15	20	100

Q3) [10pts] How could you compute the Boolean NOT given by the following truth table:

x	NOT(x)	wx+w0	Sign(wx+w0)
0	1	0.5	1
1	0	-0.5	0



Hint: For example, Boolean AND can be computed using the following perceptron.

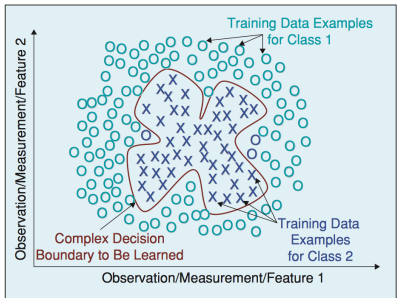


Q4) [5pts]
Given a test dataset from a two class problem with 90 instances from class A and 10 instances from class B, a classifier achieves 0.9 accuracy. If the confusion matrix contains TP=90, would you be happy to use the 0.9 accuracy classifier? Explain your answer.

	Predicted class	
True Class	Yes	No
Yes	TP: True Positive	FN: False Negative
No	FP: False Positive	TN: True Negative

We would not be happy with 0.9 accuracy, because this is an unbalanced dataset and a dummy classifier which always outputs Yes would still have 0.9 accuracy. We would need a classifier which also has as many correct TN values as possible. As the performance measure, instead of accuracy, F-measure, Precision or AUC should be used.

Q5) [5pts]
Given the following two class dataset, which classification methods we learned in the class would you use? Why?

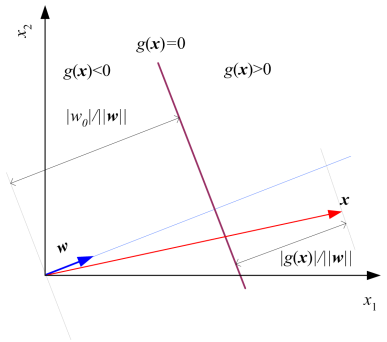


Since the boundary between the classes is nonlinear, we could not use a linear classifier. However, a generalized linear classifier (using nonlinear, for example Gaussian basis functions), SVM with a nonlinear kernel (Gaussian), k-nn, neural network, decision tree could be used.

Name and Student ID:

Q6) [15pts] Let $g(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + w_0$ be a linear discriminant function as shown in the figure.

Show that the distance between the separating hyperplane (g(x)=0) to the origin is $r_0 = \frac{|w_0|}{\|\mathbf{w}\|}$



As shown in the figure, distance between any point x and the separating hyperplane is |g(x)|/||w||. When x=0,

$g(\mathbf{x} = 0) = \mathbf{w}^T\mathbf{x} + w_0 = w_0$

Therefore, the distance between x=0 and the hyperplane is: |g(x)|/||w||=|w_0|/||w||

Q7) [20]

Q7a) Describe two methods that you could use to regularize a neural network.

Regularization: reducing the complexity to avoid overfitting

Weight decay: the error function has an additional term to penalize the large weights:

$$E' = E + \frac{\lambda}{2}\sum_i w_i^2 \qquad \Delta w_i = -\eta\frac{\partial E}{\partial w_i} - \lambda w_i$$

Weight elimination: After a neural network is trained, smaller weights are eliminated.

Cross validation: data is partitioned into training and validation set, training is stopped at the minimum of the error on the validation data.

Q7b) Why do we use a sigmoid function instead of the step function in a neural network?

In order to train a neural network, we need to use gradient descent which requires taking the derivative of the error with respect to the weights, which requires taking the derivative of the node outputs. The classification outputs of sigmoid and step function are exactly the same if classification is performed according to g(x)>0 for step and g(x)>0.5 for sigmoid. The step function is not differentiable and sigmoid is, therefore it is possible to learn the weights of a neural network with sigmoid units using gradient descent.

Q7c) Given a dataset with N=20 instances and a neural network to classify it, which cross validation method would you use to determine the number of hidden units in the neural network?

Since the number of instances is very small, we would use leave one out cross validation or bootstrapping. In leave one out method, we train the neural network using all instances but the ith one and we measure the validation performance on the ith instance. In bootstrapping, out of the available dataset of N instances, we generate a training dataset of N instances using random selection with replacement. The instances which do not occur in the training set are taken to be the validation set. The average of the validation errors is computed for each number of hidden unit units, the #hidden units resulting in the minimum average validation error is used.

Q7d) What is a conjugate prior distribution? Give an example.

In Bayesian estimation, we can estimate the posterior distribution of a parameter m given the dataset X as: p(m|X) α p(m)p(X|m)

A conjugate prior distribution (p(m)) is a distribution such that when it is multiplied with the likelihood it still has the same format.

For example, the conjugate prior for the mean of a Gaussian is another Gaussian.

For example, Dirichlet distribution is the conjugate prior for the parameter q of a multinomial distribution.