Name and Student ID: Signature:

Machine Learning BLG527E, Jan 10, 2013, Final Exam.

1	2	3	4	Total

Duration: 120 minutes.

Write your answers neatly in the space provided for them. Write your name on each sheet. Books and notes are closed. Good Luck!

QUESTIONS

QUESTION1) [30 points, 5 points each] (use at most 3 sentences per question, use of formulas, drawings etc. to better express yourself is encouraged.)

a) Compare Adaboost and bagging.

Bagging draws K bootstrap samples giving each training instance equal weight, trains K classifiers and takes the average/majority (regression/classification) of those K classifiers as the output.

In Adaboost, the bootstrap sample for classifier i (i=1..K) is drawn based on the performance of previous classifiers, an instance which has been misclassified has more probability of being included in the t'th bootstrap sample, weighted average of classifiers, based on how they perform on the whole dataset, is taken to produce the final classifier.

b) What is ROC and AUC?

A point on the ROC (Receiver operating characteristics) curve measures the classifier's hit rate (y axis, hit rate = TP / (TP+FN) = sensitivity = hit rate) and false alarm rate) versus false alarm rate (x axis, FP / (FP+TN) = 1 - Specificity) for a certain value of the threshold on the output. If classifier g's ROC curve is always higher than classifier f's ROC curve, we can say that classifier g is better. AUC (Area under –ROC- Curve) is the area under the ROC curve and summarizes the classifier performance into a single number for all operating thresholds. AUC also enables comparison of two classifiers which may be better at different operating thresholds.

c) How do you use momentum and adaptive learning rate for training a multilayer perceptron?

Momentum is used to speed up learning if weight change for a certain dimension is always positive or negative: $$\Delta w_i^t = -\eta \frac{\partial E^t}{\partial w_i} + \alpha \Delta w_i^{t-1}$$ *Momentum requires that the previous weight change values are stored.*

Adaptive learning rate solves the difficulty of choosing the learning rate for a particular learning problem. The learning rate is additively increased if the error keeps decreasing, it is multiplied by a constant<1 if the error has increased: $$\Delta\eta = \begin{cases} +a & \text{if } E^{t+\tau} < E^t \\ -b\eta & \text{otherwise} \end{cases}$$

d) What is the difference between k-means clustering and Gaussian Mixture Model (GMM) clustering?

k-means clustering assumes spherical shaped, non overlapping clusters.

GMM assumes that each cluster has a Gaussian shape with a different mean and covariance, and a weighted sum of these probabilities gives the whole distribution. Each instance has a probabilty of belonging to each of the Gaussians. GMM parameters (means, covariances and mixing coefficients are learned using the EM algorithm).

e) Compare backward and forward feature selection.

In backward feature selection, initially using all features a classifier is trained. The feature whose removal results in the best validation error is removed and feature selection is continued in this manner. In forward feature selection the first classifier is

trained using a single feature with the best validation error and a single feature whose addition to the already selected set of features results in the best validation error is added at each step. Backward feature selection is more expensive since more classifiers with more features are trained, on the other hand, backward feature selection does not separate features which work well together.

f) Compare Mahalabonis and Euclidean distance.

$$-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T \Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu})$$

Mahalabonis Distance(x, μ):
Euclidean Distance(x, μ): $(x-\mu)^T(x-\mu)$
Mahalabonis distance takes into account the shape (i.e. covariance) of the dataset distribution when computing the distance between two points.

QUESTION2) [40 points, 10 points each] (use at most 10 sentences per question, use of formulas, drawings etc. to better express yourself is encouraged.)

a) Assume that you have K classifiers which are independent and ith classifier has generalization error of E_i. Show that the average of these classifiers has less generalization error. Why, in practice, classifier averaging doesn't always result in less generalization error?

Let dj show the output of classifier j and y the average classifier.

$$y = \sum_{j=1}^{L} w_j d_j$$

$$w_j \geq 0 \text{ and } \sum_{j=1}^{L} w_j = 1$$

If d_j are iid, Bias does not change, variance decreases by K, therefore, generalization error (i.e. mse, test error on unseen instances) which is sum of bias2 + variance decreases:

$$E[y] = E\left[\sum_j \frac{1}{L} d_j\right] = \frac{1}{L} L \cdot E[d_j] = E[d_j]$$

$$\mathrm{Var}(y) = \mathrm{Var}\left(\sum_j \frac{1}{L} d_j\right) = \frac{1}{L^2}\mathrm{Var}\left(\sum_j d_j\right) = \frac{1}{L^2} L \cdot \mathrm{Var}(d_j) = \frac{1}{L}\mathrm{Var}(d_j)$$

*In practice this may not work though, because classifiers **may not be independent** of each other. Because most of the time classifiers are trained on the same dataset or using the same features etc.*

b) What are the three basic problems that you can solve using an hmm? Explain each of problem and how it is solved briefly.

λ:parameters of the HMM, λ=(A,B,π)

1. *Evaluation: Given λ, and O, calculate P (O | λ)*
   Solved using sum of the alpha parameters for the states corresponding to the last observation in sequence O.
2. *State sequence: Given λ, and O, find Q^* such that*
   $P(Q^* | O, \lambda) = \max_Q P(Q | O, \lambda)$
   Solved using theViterbi algorithm which finds the most probable state sequence.
3. *Learning: Given $X=\{O^k\}_k$, find λ^* such that*
   $P(X | \lambda^*) = \max_\lambda P(X | \lambda)$

Solved using the Baum-Welch algorithm which is an EM based algorithm.

c) How does 5x2 cross validation work? Under which circumstances do you use leave one out, 10-fold or 5x2 cross validation?

5x2 cross validation randomly partitions the whole dataset into two subsets of equal size, uses one subset for training and the other one for testing and vice-versa. Repeats this procedure 5 times, obtaining 10 validation errors.
Based on we have a very small number of (10s), moderate amount of (100s) or a lot (>1000s) instances, we could use leave one out, 10-fold or 5x2 cross validation.

d) How do you use Parzen windows for density estimation, classification and regression?

Let the Gaussian Kernel function be : $K(u) = \frac{1}{\sqrt{2\pi}} \exp\left[-\frac{u^2}{2}\right]$

Density Estimation:

$$\hat{p}(x) = \frac{1}{Nh} \sum_{t=1}^{N} K\left(\frac{x - x^t}{h}\right)$$

Classification:

$$\hat{p}(\boldsymbol{x} \mid C_i) = \frac{1}{N_i h^d} \sum_{t=1}^{N} K\left(\frac{\boldsymbol{x} - \boldsymbol{x}^t}{h}\right) r_i^t \quad \hat{P}(C_i) = \frac{N_i}{N}$$

$$g_i(\boldsymbol{x}) = \hat{p}(\boldsymbol{x} \mid C_i)\hat{P}(C_i) = \frac{1}{Nh^d} \sum_{t=1}^{N} K\left(\frac{\boldsymbol{x} - \boldsymbol{x}^t}{h}\right) r_i^t$$

Regression:

$$\hat{g}(x) = \frac{\sum_{t=1}^{N} K\left(\frac{x - x^t}{h}\right) r^t}{\sum_{t=1}^{N} K\left(\frac{x - x^t}{h}\right)}$$

QUESTION3) [20points]

	Actual	$g_1(x)$	$g_2(x)$	$g_3(x)$	$g_4(x)$	$g_5(x)$	$f_1(x)$	$f_2(x)$	$f_3(x)$	$f_4(x)$	$f_5(x)$
Train	0	0	0	0	0	0	0	0	0	1	1
	1	0	1	0	0	1	1	1	1	0	0
	0	0	0	1	1	0	0	1	1	0	0
	1	0	1	0	1	0	1	0	1	0	1
Test	?	1	1	1	1	1	1	0	1	0	1
	?	0	0	0	0	0	0	1	0	1	0
	?	0	1	0	1	0	1	0	1	0	1
	?	1	1	1	1	1	0	1	0	1	0

You are given a dataset which contains 4 training inputs and output, 4 test inputs. You have two different types of classifiers f() and g(), you train and test 5 different configurations of f() and g() (for example you might choose different parameters of the classifiers). The training and test outputs that are obtained for each input are shown in the table. Would you choose a classifier of kind f() or g() for this task? Clearly explain the reason behind your choice?

[**Hint:** if M samples (data sets) $X_i=\{x^t_i, r^t_i\}$, $i=1,...,M$ are used to fit $g_i(x)$, $i=1,...,M$

$$\text{Bias}^2(g) = \frac{1}{N} \sum_t \left[\bar{g}(x^t) - f(x^t)\right]^2$$

$$\text{Variance}(g) = \frac{1}{NM} \sum_t \sum_i \left[g_i(x^t) - \bar{g}(x^t)\right]^2$$

$$\bar{g}(x) = \frac{1}{M} \sum_i g_i(x)$$

Name and Student ID: Signature:

]

	Actual	$g_1(x)$	$g_2(x)$	$g_3(x)$	$g_4(x)$	$g_5(x)$	gbar(x)	sum((gi(x)-gbar(x))*(gi(x)-gbar(x))	$f_1(x)$	$f_2(x)$	$f_3(x)$	$f_4(x)$	$f_5(x)$	fbar(x)	sum((fi fbar(x) fbar(x)
Train	0	0	0	0	0	0	0	0	0	0	0	1	1	0.4	1.2
	1	0	1	0	0	1	0.4	1.2	1	1	1	0	0	0.6	1.2
	0	0	0	1	1	0	0.4	1.2	0	1	1	0	0	0.4	1.2
	1	0	1	0	1	0	0.4	1.2	1	0	1	0	1	0.6	1.2
Test	?	1	1	1	1	1	1	0	1	0	1	0	1	0.6	1.2
	?	0	0	0	0	0	0	0	0	1	0	1	0	0.4	1.2
	?	0	1	0	1	0	0.4	1.2	1	0	1	0	1	0.6	1.2
	?	1	1	1	1	1	1	0	0	1	0	1	0	0.4	1.2

ANSWER3:

We can compute the bias and variance of classifier f() and g() as follows and then use bias^2+variance as an estimate of mean square error on unseen test instances:

	Bias^2 (Train)	*Var(Train)*	*Bias^2+VarTrain*	*Var(Train+Test)*	*Bias^2+VarTest*
g	*0.22*	*0.18*	*0.4*	*0.12*	*0.34*
f	*0.16*	*0.24*	*0.4*	*0.24*	*0.4*

Since you do not know the test actual outputs, you can compute the bias only on the training set. However, the variance can be computed on train test or train+test set.
Based on how you compute the variance you may answer this question differently:
If you compute the variance based on the training instances only:
Both f and g have the same bias^2+variance, which is 0.4, therefore we can choose either one of them.
If you compute the variance based on the training+test instances:
g has smaller bias^2+variance (0.34) then f (0.4), therefore, you should choose g.

QUESTION4) [10points]

Compute the change in v and w (i.e. Δv and Δw) if you modified the error function for a neural network as follows.

$$E(\mathbf{W},\mathbf{v}|\mathcal{X}) = \frac{1}{2}\sum_t \left(r^t - y^t\right)^2 + \| v \|^2$$

Hint: Without the regularization term, you would have:

$$y_i = \mathbf{v}_i^T \mathbf{z} = \sum_{h=1}^{H} v_{ih} z_h + v_{i0} \quad z_h = \text{sigmoid}\left(\mathbf{w}_h^T \mathbf{x}\right)$$

$$\Delta v_h = \sum_t \left(r^t - y^t\right) z_h^t$$

$$\begin{aligned}\Delta w_{hj} &= -\eta \frac{\partial E}{\partial w_{hj}} = -\eta \sum_t \frac{\partial E}{\partial y^t} \frac{\partial y^t}{\partial z_h^t} \frac{\partial z_h^t}{\partial w_{hj}} \\ &= -\eta \sum_t -\left(r^t - y^t\right) v_h z_h^t \left(1 - z_h^t\right) x_j^t \\ &= \eta \sum_t \left(r^t - y^t\right) v_h z_h^t \left(1 - z_h^t\right) x_j^t\end{aligned}$$

ANSWER4:

Let *E'* denote the new error function. $E' = E + \sum_h v_h^2$

$$\Delta v_h = -\eta \frac{\partial E'}{\partial v_h} = -\eta \frac{\partial (E + \sum_h v_h^2)}{\partial v_h} = -\eta (\sum_t \left(r^t - y^t\right) z_h^t + v_h)$$

Since, $\sum_h v_h^2$ is not a function of w_{hj}, the computation of Δw_{hj} remains the same:

$$\Delta w_{hj} = -\eta \frac{\partial E'}{\partial w_{hj}} = -\eta \frac{\partial (E + \sum_h v_h^2)}{\partial w_{hj}} = -\eta \frac{\partial E}{\partial w_{hj}} = \eta \sum_t \left(r^t - y^t\right) v_h z_h^t \left(1 - z_h^t\right) x_j^t$$