Machine Learning BLG527E, June 8, 2017, 120mins, Final Exam, Duration: 120 minutes.
Closed books and notes. Use a separate page for each question and write the question no at the top. Write your name on each sheet and sign. Good Luck!

QUESTIONS

Q1) [25pts]
Given the following HMM with N=2 states, M=3 observations from {Red, Green, Blue} for each state and the following parameters:

$$\Pi=[0.1,0.9]^T \qquad A=\begin{bmatrix} 0.6 & 0.4 \\ 0.1 & 0.9 \end{bmatrix} \qquad B=\begin{matrix}\text{Red} & \text{Green} & \text{Blue}\end{matrix}\begin{bmatrix} 0.1 & 0.4 & 0.5 \\ 0.3 & 0.4 & 0.3 \end{bmatrix}$$

Given the following sequence of observations O={Blue, Red}, **and the knowledge that the first state is S_2,** compute P(O|A,B, π).
Hint: The forward variables in an HMM are calculated as follows:
$\alpha_t(i) \equiv P(O_1 \cdots O_t, q_t = S_i \mid \lambda)$

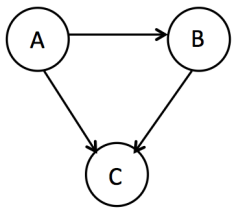
Initializa tion :

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

Recursion :

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

Answer1) Since first state is known to be S2, α1(1)=0*0.5=0 α1(2)=1*0.3=0.3
α2(1)=(0*0.6+0.3*0.1)*0.1=0.003
α2(2)=(0*0.4+0.3*0.9)*0.3=0.081
P(O|A,B, π)= α2(1)+ α2(2)=0.084



P(A)=0.8
P(B|A)=0.1
P(B|~A)=0.5
P(C|A,B)=0.2
P(C|~A,B)=0.5
P(C|A,~B)=0.4
P(C|~A,~B)=0.2

Q2) ___0.37_____ For the graphical model given above.
Q2a) [10pts] P(C|B)=?
Q2b) [10pts] Is A independent of C given B? Explain why or why not.

Answer2a)
compute: P(B)=P(B|A)*P(A)+P(B|~A)*P(~A)=0.1*0.8+0.5*0.2=0.18
P(C|B)= P(C|A,B)*P(A|B)+P(C|~A,B)*P(~A|B)=
= P(C|A,B)*P(B|A)*P(A)/P(B)+ P(C|~A,B)* P(B|~A)*P(~A)/P(B)
=0.2*0.1*0.8/0.18+0.5*0.5*0.2/0.18=**0.37**

Answer2b)
P(A,C|B)=P(A,C,B)/P(B)=P(A)*P(B|A)*P(C|A,B)/P(B)=0.8*0.1*0.2/0.18=0.089
P(A|B)=P(B|A)*P(A)/P(B)=0.1*0.8/0.18=0.44
To check for independence:
P(A,C|B)=?P(A|B)*P(C|B)
0.089 =? 0.44 * 0.37 = 0.1628
0.089 ≠ 0.1628
Therefore, **A is not independent of C given B.**

Q3a) [10pts] 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. $K(\mathrm{u},\mathrm{v}) = \phi(\mathrm{u})^T\phi(\mathrm{v})$. For vectors u and v, if K_1 and K_2 are valid kernels, **show that the following is a valid kernel:**
$K(\mathrm{u},\mathrm{v}) = a^2 * K_1(\mathrm{u},\mathrm{v}) + b^2 * K_2(\mathrm{u},\mathrm{v})$

Q3b) [10pts] You have a **regression** problem with one dimensional inputs and you know the following **hint** about your target function f(x) = f(-x). You are given a training data set with inputs $X=\{x_1,...,x_N\}$ and outputs $r=\{r_1,...,r_N\}$ and will train an MLP **g(x,w)** with input x and weights **w**. **Give details of how to compute the weight w* that minimizes both the training error and the hint error?**

Answer3a)
Since K_1 and K_2 are valid kernels, there existss transformation functions ϕ_1 and ϕ_2 such that:
$K_1(\mathrm{u},\mathrm{v}) = \phi_1(\mathrm{u})^T\phi_1(\mathrm{v})$ and $K_2(\mathrm{u},\mathrm{v}) = \phi_2(\mathrm{u})^T\phi_2(\mathrm{v})$
$K(\mathrm{u},\mathrm{v}) = a^2 * K_1(\mathrm{u},\mathrm{v}) + b^2 * K_2(\mathrm{u},\mathrm{v}) = a^2\phi_1(\mathrm{u})^T\phi_1(\mathrm{v}) + b^2\phi_2(\mathrm{u})^T\phi_2(\mathrm{v}) = \phi_3(\mathrm{u})^T\phi_3(\mathrm{v})$
where $\phi_3(\mathrm{u}) = [a\phi_1(\mathrm{u}), b\phi_2(\mathrm{u})]$

Answer3b) Define the error function to minimize as:

$$E(w) = \frac{1}{N}\sum_{t=1}^{N}(g(x_t, w) - r_t)^2 + \frac{\lambda}{N}\sum_{t=1}^{N}\left(g(x_t, w) - g(-x_t, w)\right)^2$$

The second part of the function could also be computed over some random instances of x.

Dtart with small random initial w and update them as $w^{t+1} = w^t - \eta \left.\frac{dE}{dw}\right|_{w^t}$.

Take the derivative and compute the derivatives of g(x,w) using backpropagation algorithm.

$$\frac{dE}{dw} = \frac{2}{N}\sum_{t=1}^{N}(g(x_t, w) - r_t)\frac{dg(x_t, w)}{dw} + \frac{2\lambda}{N}\sum_{t=1}^{N}\left(g(x_t, w) - g(-x_t, w)\right)\left(\frac{dg(x_t, w)}{dw} - \frac{dg(-x_t, w)}{dw}\right)$$

Q4) [20pts]

Q4a) Write down two differences between logistic regression classifier versus a 1 hidden layer and 1 output neural network used for classification.

Q4b) Write down two differences between Bagging versus Adaboost.

Q4c) Why do we use a sigmoid function instead of the step function in a neural network?

Q4d) What is a conjugate prior distribution? Give an example.

Answer4a)

-Logistic regression classifier has only 1 layer, neural network has two layers, therefore, logistic regression classifier can only compute a linear function of the inputs and then pass it through a nonlinearity (sigmoid) to map it to [0:1] range.

-By inspecting the weight assigned for each input, functions implemented by logistic regression can be implemented by humans.

-Logistic regression takes less time to train

-Since it is a simpler function, logistic regression is less likely to overfit then a hidden neural network.

Answer4b)

-Bagging assigns constant 1/N for probability of selection of each training instance, Adaboost changes the probability of selection of each instance based on how much error is made on that instance.

- Weight of each classifier output is constant ın bagging where as it depends on classifier performance in Adaboost.

Answer4c)

We need to compute $\frac{dE}{dw}$ when minimizing an error function E, which is a function of the neural network output, when the output is the step function it is not continuous and differentiable at argument 0. On the other hand when the output is sigmoid, it is continuous and differentiable for all inputs.

Answer4d)

Let X be the observations and θ be parameters of a distribution, $p_0(\theta)$ be the prior distribution of θ. We can compute the posterior distribution of θ after observing X as:

$p(\theta|X) = \frac{p(X|\theta)p_0(\theta)}{p(X)}$. Conjugate prior distributions are distributions p such that shape of the prior $p_0(\theta)$ and the posterior $p(\theta)$ are the same. For example, for estimation of the mean of the normal distribution, normal distribution is a conjugate prior.

Q5)[15pts]

	Fold1	**Fold2**	**Fold3**
Algo1	**0.14**	**0.16**	**0.14**
Algo2	**0.1**	**0.11**	**0.09**

You experimented with Algo1 and Algo2 on 3 folds and obtained the following errors. Use ANOVA to determine if Error(Algo1)=Error(Algo2) at significance level α=0.1

Hint: $\frac{SSb/(L-1)}{SSw/(L(K-1))} \sim F_{L-1,L(K-1)}$ when there are L algorithms on K folds

$F_{0.1,1,4} = 4.55$

Answer5)

m1=0.157, m2=0.1 m = (m1+m2)/2 = 0.123, L=2, K=3

SSb= 0.00327 // $SSb \equiv K\sum_j (m_j - m)^2$ SSw= 0.00047 // $SSw \equiv \sum_j \sum_i (X_{ij} - m_j)^2$

Statistic given in the hint = (0.00327/1) /(0.00047/4)=28

28 > 4.55, so we can not accept that Error(Algo1)=Error(Algo2) at this significance level.