Università
della
Svizzera
italiana

Institute of
Computational
Science
ICS

Machine Learning **2019**

Student: Sanchit Chiplunkar

Solution for Assignment 3

Due date: 27 November 2019, 23:59

Answer 1

Answer 1.a p(x) = $\sum_{i=1}^{i=K} \pi_i P(x; \lambda_i)$

Answer 1.b The value of γ_{nk} is : $\gamma_{nk} = \frac{\pi_k P(x_n, \lambda_k)}{\sum_{j=1}^{j=K} \pi_i P(x_n; \lambda_i)}$

Answer 1.c $\lambda_k = \sum_{i=0}^{i=N} \gamma_{nk} x_n / N_k$
Also $\pi_k = \frac{N_k}{N}$ and $N_k = \sum_{n=1}^{N} \gamma_{nk}$

Answer 2

Answer 2.a This is a Hidden Markov Model(HMM) as our States are Urn which are hidden from us because we are blindfolded and the only thing are Observe/ are told is the the colour of the balls. The Urns has initial probabilities of π shown below and some transition probability A while the Emission probability (B) is the probability of a Observing a colour given the Urn. All the 5 values are given. below
π = [0.5,.05]
S = ["Urn 1","Urn 2"]
O = ["Red","Yellow","Blue"]
A = ["Urn 1": ["Urn 1": 0.5, "Urn 2": 0.5], "Urn 2": ["Urn 1": 0.75, "Urn 2": 0.25]]
B = ["Urn 1": ["Blue": $\frac{5}{11}$, "Red": $\frac{2}{11}$, "Yellow": $\frac{4}{11}$],"Urn 2": ["Blue": 0.3 , "Red": 0.4, "Yellow": 0.3]]
Please Note: I could have also made the Matrix for A and B but was finding it difficult to make in latex with proper labeling and so thus made a list of it

Answer 2.b The code for this exercise is written in **ex2.py**. We have to Compute P($O_0 = Urn_1, O_1 = Urn_2, O_2 = Urn_1 | S_0 = Y, S_1 = R, S_2 = B$) using the Baye's formula this can be written as $\frac{P(O_0=Urn_1, O_1=Urn_2, O_2=Urn_1, S_0=Y, S_1=R, S_2=B)}{P(S_0=Y, S_1=R, S_2=B)}$. So we need to compute the joint probability and the total probability to compute our answer. JointProbablity is computed in the **jointprob function** and it's value is 0.0123 and the Total Probability is computed in the **compute function** using Dynamic Programming and it's value is 0.035315. Thus our Conditional Probablity or the Answer is just the division of two which is 0.351

Answer 2.c The Viterbi Algorithm is implemented in the viterbi.py python file the maximum probability that was achieved for the given problem is 0.01239 for the Urn in the following Order Urn 2,Urn 1,Urn 1.

Answer 3

The code is written in python file ex3.py which gives the correct answer 0.195 on running the program.

Answer 4

We are given 3 data points A: (-1,0);B: (0,1);C: (1,0). The loss to compute validation loss is Squared Error.

Answer 4.a If we leave out point A : (-1,0)
f(x) = .5 as (1-0)/2 = .5 and so our validation error for point A becomes = $(0.5-0)^2 = .25$
If we leave out point B : (0,1)
f(x) = 0 as (0-0)/2 = 0 and so our validation error for point B becomes = $(1)^2$= 1
If we leave out point C : (1,0)
f(x) = .5 as (1-0)/2 = .5 and so our validation error for point C becomes = $0.5-0)^2 = .25$ Total Validation Error is the average of the above three validation error which becomes (.25 + .25 + 1) /3 = 0.5

Answer 4.b In case of f(x) = ax+b when we leave out one data point we fit a line with the help of remaining two data point with the formula $\frac{y-y_a}{x-x_a} = \frac{x_a-x_b}{y_a-y_b}$ where A: (x_a, y_a) and B: (x_b, y_b) with $< x_i, y_i >$ being the coordinates of the data point.
Now When we leave out the point A: (-1,0) from the above formula f(x) = 1-x and so for point A f(x) = 1- (-1) = 2 Thus the Validation error become $(2-0)^2 = 4$
When we leave out the point B: (0,1) from the above formula f(x) = 0 and so for point B f(x) = 0 Thus the Validation error become $(1-0)^2 = 1$
When we leave out the point C: (1,0) from the above formula f(x) = 1+x and so for point C f(x) = 1 + 1 = 2 Thus the Validation error become $(2-0)^2 = 4$
Thus the validation error becomes. $\frac{4+4+1}{3} = 3$