CS440/ECE448 Lecture 20
Exam 2 Review

4/7/2021

Mark Hasegawa-Johnson

Outline

- Perceptrons: Lecture 8
- Neural Networks: Lectures 9, 10
- Bayesian Networks: Lectures 13, 14
- HMMs: Lectures 15, 16

Lecture 12 (Autograd) was just an intro to MP3.

Lecture 17 (Vector semantics) is just a discussion of an interesting way to apply what you know about neural nets; there are no questions on the exam specifically addressing this lecture.

Perceptrons (Lecture 8)

- Classification, binary perceptron: $\hat{y} = \mathrm{sgn}(w^T x)$
- Learning, binary perceptron: if $y = \hat{y}$ then do nothing, else $w = w + \eta \mathrm{y} x$.
- Classification, multi-class perceptron: $\hat{y} = \mathrm{argmax}_{c=0}^{V-1}\left(w_c^T x\right)$
- Learning, multi-class perceptron: if $y = \hat{y}$ then do nothing, else $w_y = w_y + \eta x$, and $w_{\hat{y}} = w_{\hat{y}} - \eta x$.

Neural Nets (Lectures 9 and 10)

- Classification (forward-prop):

$$P(Y=y|x) \ = \ g_y^{(L)}(W^{(L)} \cdots g^{(2)}(W^{(2)} g^{(1)}(W^{(1)} x)) \cdots)$$

  - $g^{(l)}(\cdot)$ is an element-wise nonlinearity ($g_y^{(L)}(\cdot)$ is its y^{th} element).
  - $W^{(l)}$ is a weight matrix.
  - $e^{(l)} = W^{(l)} h^{(l-1)}$ is called the l^{th}-layer excitation.
  - $h^{(l)} = g^{(2)}\left(e^{(l)}\right)$ is the l^{th}-layer activation.
- Learning (back-prop):

$$\nabla_{W^{(l)}} \mathcal{L} = \left(\nabla_{W^{(l)}} e^{(l)}\right)\left(\nabla_{e^{(l)}} h^{(l)}\right) \cdots \left(\nabla_{h^{(L-1)}} e^{(L)}\right)\left(\nabla_{e^{(L)}} h^{(L)}\right)\left(\nabla_{h^{(L)}} \mathcal{L}\right)$$

- Learning (gradient descent):

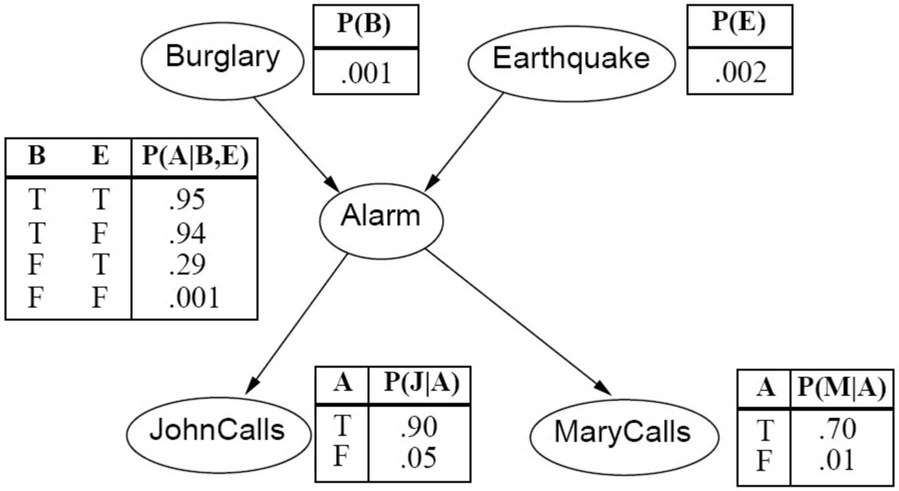
$$W^{(l)} \leftarrow W^{(l)} - \eta \nabla_{W^{(l)}} \mathcal{L}$$

Neural Nets (Lectures 9 and 10)

- Loss functions:
  - Mean-squared error (if y_i is a real vector): $\mathcal{L} = -\frac{1}{2n} \sum_{i=1}^{n} \left\| h^{(L)} - y_i \right\|^2$
  - Cross entropy (if y_i is a classification label): $\mathcal{L} = -\frac{1}{n} \sum_{i=1}^{n} \ln P(Y = y_i | x_i)$
- Nonlinearities:
  - Softmax: $\mathrm{softmax}_j(x) = \exp(x_j) / \sum_k \exp(x_k)$
  - Logistic Sigmoid: $\sigma(x) = 1/(1 + \exp(-x))$
  - ReLU: $\mathrm{ReLU}(x) = \max(0, x)$

Bayesian Networks (Lectures 13, 14)

- Variables in a Bayes net are **independent** if they have no common ancestors
  - If they have a common ancestor (e.g., J and M), they are not independent
  - If one is the ancestor of the other (e.g., B and M), they are not independent
- Variables in a Bayes net are **conditionally independent** given knowledge of:
  - Their common ancestors (e.g., J, M are c.i. given A), and
  - any variable that is a descendant of one, and an ancestor of the other (e.g., B, M are c.i. given A)



P(B)
.001

P(E)
.002

B	E	P(A|B,E)
T	T	.95
T	F	.94
F	T	.29
F	F	.001

A	P(J|A)
T	.90
F	.05

A	P(M|A)
T	.70
F	.01

Bayesian Network Learning (Lecture 14)

- Maximum Likelihood learning, given complete data:

$$P(B = b|A = a) = \frac{\text{count}(A = a, B = b)}{\text{count}(A = a)}$$

- Laplace Smoothing, given complete data:

$$P(B = b|A = a) = \frac{\text{count}(A = a, B = b) + k}{\text{count}(A = a) + k(1 + \#\text{distinct values of } B)}$$

- Expectation Maximization, given incomplete data:

$$P(B = b|A = a) = \frac{E[\text{count}(A = a, B = b)]}{E[\text{count}(A = a)]}$$

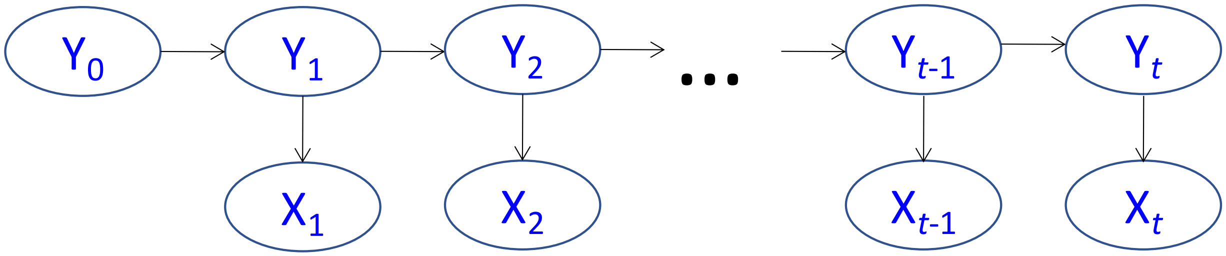
Hidden Markov Model (Lecture 15)

- A hidden Markov model assumes that both the state and the observation are Markov.
- **State Transitions:** the Markov assumption means that each state variable depends only on the preceding time step:

$$P(Y_t \mid Y_0, \ldots, Y_{t-1}) = P(Y_t \mid Y_{t-1})$$

- **Observation model:** the Markov assumption means that each state variable depends only on the current state:

$$P(X_t \mid Y_0, \ldots, Y_t, X_1, \ldots, X_{t-1}) = P(X_t \mid Y_t)$$



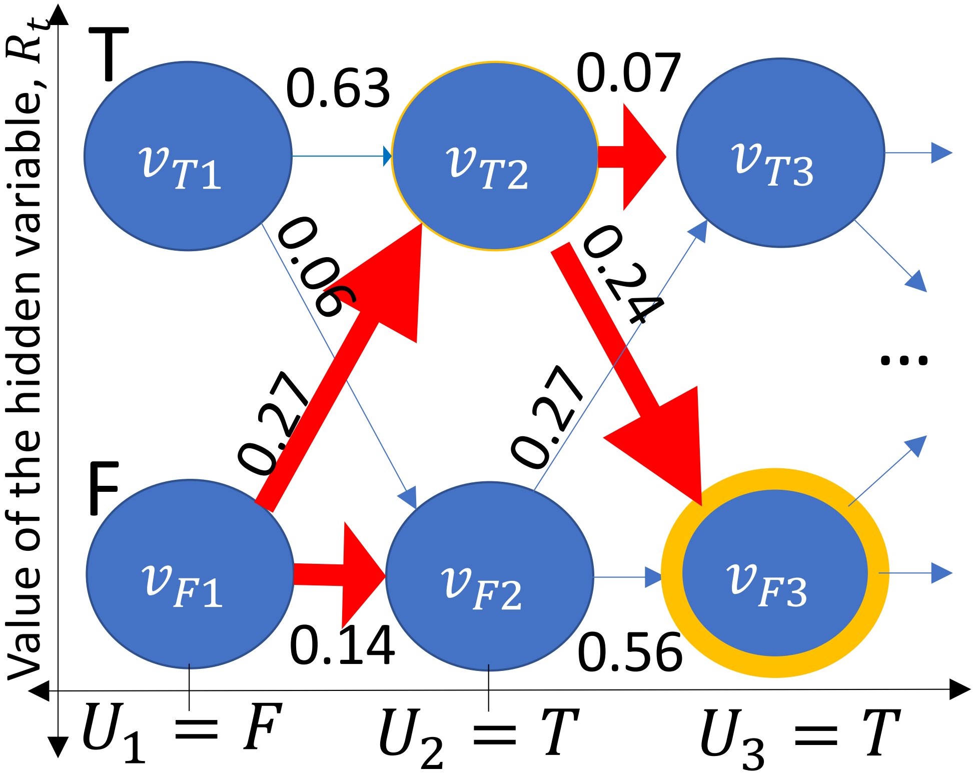
Viterbi Algorithm (Lecture 16)

Node Probability: Probability of the best path until node j at time t

$$v_{jt} = \max_{r_1,\ldots,r_{t-2},i} P(U_1 = u_1 \ldots, R_1 = r_1, \ldots, R_{t-1} = i, R_t = j)$$

Backpointer: which node, i, precedes node j on the best path?

$$i^*_{jt} = \operatorname*{argmax}_{r_1,\ldots,r_{t-2},i} P(U_1, = u_1 \ldots, R_1 = r_1, \ldots, R_{t-1} = i, R_t = j)$$



Some problems from the sample exam

- Perceptron: question 2
- Neural nets: question 6
- Bayesian networks: question 9
- HMMs: question 20

Question 2: Perceptron

A particular perceptron is initialized with the weights $w = [1,1,1]$ and the bias $b = 1$.

a) The perceptron undergoes one iteration of training, with learning rate $\eta = 1$, with the training token $x = [0.1,0.6,0.5]$, $y = -1$. After this one iteration of training, what are w and b?

b) The perceptron undergoes one more iteration of training, with learning rate $\eta = 1$, with the training token $x = [0.1,0.1,0.4]$, $y = 1$. After this additional iteration of training, what are w and b?

Question 2: Perceptron

A particular perceptron is initialized with the weights $w = [1,1,1]$ and the bias $b = 1$.

a) The perceptron undergoes one iteration of training, with learning rate $\eta = 1$, with the training token $x = [0.1,0.6,0.5]$, $y = -1$. After this one iteration of training, what are w and b?

Answer, part 1: did $y = \hat{y}$?

$$\hat{y} = \text{sgn}(w^T x + b) = \text{sgn}(0.1 + 0.6 + 0.5 + 1) = 1 \neq y$$

Answer, part 2: well, then,

$$w = w + \eta y x = w - x = [0.9,0.4,0.5]$$
$$b = b + \eta y = 1 - 1 = 0$$

Question 2: Perceptron

A particular perceptron is initialized with the weights $w = [1,1,1]$ and the bias $b = 1$.

b) The perceptron undergoes one more iteration of training, with learning rate $\eta = 1$, with the training token $x = [0.1,0.1,0.4]$, $y = 1$. After this one iteration of training, what are w and b?

Answer, part 1: did $y = \hat{y}$?

$$\hat{y} = \text{sgn}(w^T x + b) = \text{sgn}\big((0.9)(0.1) + (0.4)(0.1) + (0.5)(0.4) + 0\ \big) = 1$$
$$= y$$

Answer, part 2: well, then, w and b are unchanged. w is still $w = [0.9,0.4,0.5]$, and b is still $b = 0$.

Question 6: Neural Net

Cross-entropy is

$$\mathcal{L} = -\frac{1}{n}\sum_{i=1}^{n} \ln P(Y = y_i | x_i)$$

Suppose you have $n = 2$ training samples, $x_1 = [0.2,0.6]^T$, $y_1 = 1$, $x_2 = [-1.2,0.3]^T$, $y_2 = 0$. Suppose $P(Y = 1|x)$ is defined as $P(Y = 1|x) = \sigma(w^T x)$, where $\sigma(\cdot)$ is the logistic sigmoid function. This computation has already been performed so you already know that $P(Y = 1|x_1) = 0.2$, $P(Y = 1|x_2) = 0.9$. Find the gradient of $\mathcal{L}$ with respect to the vector w.

Question 6: Neural Net

Find the gradient of $\mathcal{L}$ with respect to the vector w:

$$\nabla_w \mathcal{L} = \left[\frac{\partial \mathcal{L}}{\partial w_1}, \frac{\partial \mathcal{L}}{\partial w_2}\right]^T$$

$$\frac{\partial \mathcal{L}}{\partial w_j} = \sum_{i=1}^{n} \frac{\partial \mathcal{L}}{\partial P(Y = y_i | x_i)} \frac{\partial P(Y = y_i | x_i)}{\partial w_j}$$

$$= \sum_{i=1}^{n} \left(-\frac{1}{n} \frac{1}{P(Y = y_i | x_i)}\right) \frac{\partial P(Y = y_i | x_i)}{\partial w_j} = -\frac{1}{2} \left(\frac{1}{0.2} \frac{\partial P(Y = 1 | x_1)}{\partial w_j} + \frac{1}{0.1} \frac{\partial P(Y = 0 | x_2)}{\partial w_j}\right)$$

Question 6: Neural Net

Find the gradient of $\mathcal{L}$ with respect to the vector w:

$$\frac{\partial \mathcal{L}}{\partial w_j} = -\frac{1}{2}\left(\frac{1}{0.2}\frac{\partial P(Y=1|x_1)}{\partial w_j} + \frac{1}{0.1}\frac{\partial P(Y=0|x_2)}{\partial w_j}\right)$$

$$P(Y=1|x_i) = \sigma(w^T x_i) = \frac{1}{1+\exp(-(w_1 x_{i1} + w_2 x_{i2}))}$$

$$\frac{\partial P(Y=1|x_1)}{\partial w_j} = \sigma(w^T x_1)\big(1-\sigma(w^T x_1)\big)x_{1j} = (0.2)(0.8)x_{1j}$$

$$\frac{\partial P(Y=0|x_2)}{\partial w_j} = \frac{\partial(1-P(Y=1|x_2))}{\partial w_j} = -\sigma(w^T x_2)\big(1-\sigma(w^T x_2)\big)x_{2j} = -(0.9)(0.1)x_{2j}$$

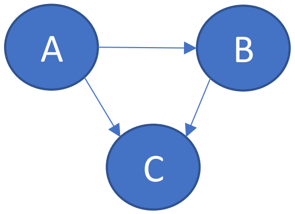
Question 6: Neural Net

Find the gradient of $\mathcal{L}$ with respect to the vector w:

$$\frac{\partial \mathcal{L}}{\partial w_j} = -\frac{1}{2}\Big((0.8)x_{1j} - (0.9)x_{2j}\Big)$$

$$= \begin{cases} -\frac{1}{2}\big((0.8)(0.2) - (0.9)(-1.2)\big) & j = 1 \\ -\frac{1}{2}\big((0.8)(0.6) - (0.9)(0.3)\big) & j = 2 \end{cases}$$

Bayesian networks: question 9



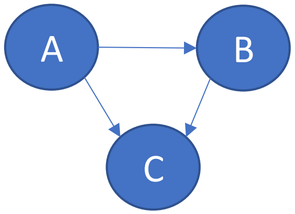
A, B, and C are binary random variables, whose dependencies are shown in the Bayes net above. Variable A has a probability $P(A = 1) = 0.3$. Conditional probabilities of the other two variables are given in the table below:

a	P(B=1|A=a)	P(C=1|A=a,B=0)	P(C=1|A=a,B=1)
0	0.2	0.4	0.9
1	0.8	0.3	0.7

a) What is $P(A = 1, B = 0, C = 1)$?

b) What is $P(A = 1|C = 1)$?

Bayesian networks: question 9



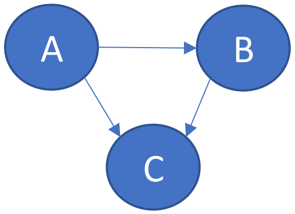
$P(A = 1) = 0.3$, and:

a	P(B=1|A=a)	P(C=1|A=a,B=0)	P(C=1|A=a,B=1)
0	0.2	0.4	0.9
1	0.8	0.3	0.7

a) What is $P(A = 1, B = 0, C = 1)$?

$$P(A = 1, B = 0, C = 1) = P(A = 1)P(B = 0|A = 1)P(C = 1|A = 1, B = 0)$$
$$= (0.3)(0.2)(0.3)$$

Bayesian networks: question 9



$P(A = 1) = 0.3$, and:

a	P(B=1|A=a)	P(C=1|A=a,B=0)	P(C=1|A=a,B=1)
0	0.2	0.4	0.9
1	0.8	0.3	0.7

a) What is $P(A = 1, B = 0, C = 1)$?

$$P(A = 1, B = 0, C = 1) = P(A = 1)P(B = 0|A = 1)P(C = 1|A = 1, B = 0) \\ = (0.3)(0.2)(0.3)$$

b) What is $P(A = 1|C = 1)$?

$$P(A = 1|C = 1) = \frac{P(A = 1, C = 1)}{P(A = 1, C = 1) + P(A = 0, C = 1)}$$

$$= \frac{(0.3)(0.2)(0.3) + (0.3)(0.8)(0.7)}{(0.3)(0.2)(0.3) + (0.3)(0.8)(0.7) + (0.7)(0.8)(0.4) + (0.7)(0.2)(0.9)}$$

HMMs: question 20

The University of Illinois Vaccavolatology Department has four professors, named Aya, Bob, Cho, and Dale. The building has only one key, so we take special care to protect it.

- Every day Aya goes to the gym, and on the days she has the key, 60% of the time she forgets it next to the bench press. When that happens one of the other three TAs, equally likely, always finds it since they work out right after.
- Bob likes to hang out at Einstein Bagels and 50% of the time he is there with the key, he forgets the key at the shop. Luckily Cho always shows up there and finds the key whenever Bob forgets it.
- Cho has a hole in her pocket and ends up losing the key 80% of the time somewhere on Goodwin street. However, Dale takes the same path to campus and always finds the key.
- Dale has a 10% chance to lose the key somewhere in the Vaccavolatology classroom, but then Cho picks it up.

The professors lose the key at most once per day, around noon (after losing it they become extra careful, for the rest of the day), and they always find it the same day in the early afternoon.

a) Let $X_t =$ the first letter of the name of the person who has the key ($X_t \in \{A, B, C, D\}$). Find the Markov transition probabilities $P(X_t | X_{t-1})$.

HMMs: question 20

- Every day Aya goes to the gym, and on the days she has the key, 60% of the time she forgets it next to the bench press. When that happens one of the other three TAs, equally likely, always finds it since they work out right after.

$$P(X_t = x|X_{t-1} = A) = \begin{cases} 0.4 & x = A \\ 0.2 & x \in \{B, C, D\} \end{cases}$$

- Bob likes to hang out at Einstein Bagels and 50% of the time he is there with the key, he forgets the key at the shop. Luckily Cho always shows up there and finds the key whenever Bob forgets it.

$$P(X_t = x|X_{t-1} = B) = \begin{cases} 0.5 & x = B \\ 0.5 & x = C \end{cases}$$

- Cho has a hole in her pocket and ends up losing the key 80% of the time somewhere on Goodwin street. However, Dale takes the same path to campus and always finds the key.

$$P(X_t = x|X_{t-1} = C) = \begin{cases} 0.2 & x = C \\ 0.8 & x = D \end{cases}$$

- Dale has a 10% chance to lose the key somewhere in the Vaccavolatology classroom, but then Cho picks it up.

$$P(X_t = x|X_{t-1} = D) = \begin{cases} 0.1 & x = C \\ 0.9 & x = D \end{cases}$$

HMMs: question 20

b) Sunday night Bob had the key (the initial state distribution assigns probability 1 to $X_0 = B$ and probability 0 to all other states). The first lecture of the week is Tuesday at 4:30pm, so one of the professors needs to open the building at that time. What is the probability for each professor to have the key at that time?

$$P(X_1 = x|X_0 = B) = \begin{cases} 0.5 & x = B \\ 0.5 & x = C \end{cases}$$

$$P(X_2 = y|X_0 = B) = \sum_x P(X_2 = y|X_1 = x)P(X_1 = x|X_0 = B)$$

$$= \begin{cases} (0.5)(0.5) & y = B \\ (0.5)(0.2) + (0.5)(0.5) & y = C \\ (0.5)(0.8) & y = D \end{cases}$$

Summary

- Perceptrons:

  If $y = \hat{y}$ then do nothing, else $w = w + \eta y x$.

- Neural Networks:

  $$W^{(l)} \leftarrow W^{(l)} - \eta \nabla_{W^{(l)}} \mathcal{L}$$

- Bayesian Networks:

  $$P(B, E, A, J, M) = P(B)P(E)P(A|B, E)P(J|A)P(M|A)$$

- HMMs:

  $$P(Y_1, X_1, Y_2, X_2) = P(Y_1)P(X_1|Y_1)P(Y_2|Y_1)P(X_2|Y_2)$$