FINAL EXAM

CMU 10-601B: MACHINE LEARNING (FALL 2016)

Dec. 7, 2016

Name: _____

Andrew ID: _____

START HERE: Instructions

- This exam has 21 pages and 7 Questions (page one is this cover page). Check to see if any pages are missing. Enter your name and Andrew ID above.
- You are allowed to use one page of notes, front and back.
- Electronic devices are not acceptable.
- Some of the questions are True/False or Multiple Choice with no explanation required. In this case, we will give partial credit if you supply an incorrect choice, but a partially correct justification.
- Note that the questions vary in difficulty. Make sure to look over the entire exam before you start and answer the easier questions first.

Question	Points	Extra Credit	Score
1	20	0	
2	20	0	
3	20	0	
4	10	0	
5	10	3	
6	20	0	
7	20	3	
Total	120	6	

1 Before the midtem [20 pts]

Circle True or False for the questions below. **If your answer is False, provide a One line justification.**

1. **[2 pts]** The full set of parameters for a multivariate Gaussian distribution can be estimated by computing the mean and variance of each variable separately.

   Circle one: True False
   One line justification (only if False):

   Solution: False. The multivariate Gaussian distribution also captures covariances between variables.

2. **[2 pts]** The ID3 algorithm is a method to train decision trees that is guaranteed to find the optimal decision tree.

   Circle one: True False
   One line justification (only if False):

   Solution: False. Finding the optimal decision tree is NP-hard.

3. **[2 pts]** Consider the linear regression model $y = w^T x + \epsilon$. Assuming $\epsilon \sim \mathcal{N}(0, \sigma^2)$ and maximizing the conditional log-likelihood is equivalent to minimizing the sum of squared errors $\|y - w^T x\|_2^2$.

   Circle one: True False
   One line justification (only if False):

   Solution: True. The squared error term comes from the squared term in the Gaussian distribution.

4. **[2 pts]** In logistic regression, adding a regularization term typically results in a decrease in the training error.

   Circle one: True False
   One line justification (only if False):

   Solution: False. Regularization prevents overfitting — overfitting leads to trivially low training error.

5. **[2 pts]** When dealing with probabilistic models for text classification, one typically prefers MLE estimates over MAP estimates.

Circle one: True False
One line justification (only if False):

Solution: False. Text classification benefits from including a prior—especially for unseen words in the training data.

6. **[2 pts]** The perceptron is an example of a generative classifier.

   Circle one: True False
   One line justification (only if False):

   Solution: False. The perceptron does not model $P(X|Y)$.

7. **[2 pts]** There is at least one set of 5 points in $\mathcal{R}^4$ that can be shattered by the hypothesis set of hyperplanes in $\mathcal{R}^4$.

   Circle one: True False
   One line justification (only if False):

   Solution: True. The VC dimension of a d-dimensional hyperplane is $d + 1$, which in this case would be 5.

8. **[2 pts]** With an infinite supply of training data, the trained Naïve Bayes classifier is an optimal classifier.

   Circle one: True False
   One line justification (only if False):

   Solution: False. NB does not model correlations between features for a given class, which could increase classification performance.

9. **[2 pts]** The support vectors for a soft margin SVM include the points within the margin as well as those that are incorrectly classified.

   Circle one: True False
   One line justification (only if False):

   Solution: True.

10. **[2 pts]** $K(x, y) = 1 + x^T y$ is a valid kernel function.

    Circle one: True False
    One line justification (only if False):

Solution: True. Map x to $\phi(x)$ by appending a 1 to x. Do the same for $\phi(y)$. Thus, $\phi(x)^T\phi(y) = 1 + x^T y$ is a valid kernel.

2 Clustering and Lloyd's Algorithm [20 pts]

2.1 Multiple choice

1. **[2 pts]** Which of the following statements is true about clustering using Lloyd's algorithm? **Select all that apply. No justification necessary.**

   (a) Lloyd's algorithm always converges to a global optimum.

   (b) The choice of initialization greatly affects the performance of the algorithm.

   (c) With random initialization, increasing the selected number of clusters k makes it more likely that the initialization is better.

   (d) Lloyd's method is proved to converge only if initialized with k-means++.

   Solution: b.

2. **[2 pts]** Say you want to use k-means clustering to partition N data points, where $1 < k < N$. Which of the following is a sensible way to pick the value of k? **Select all that apply. No justification necessary.**

   (a) Choose k such that the k-means cost function is minimized on held out data.

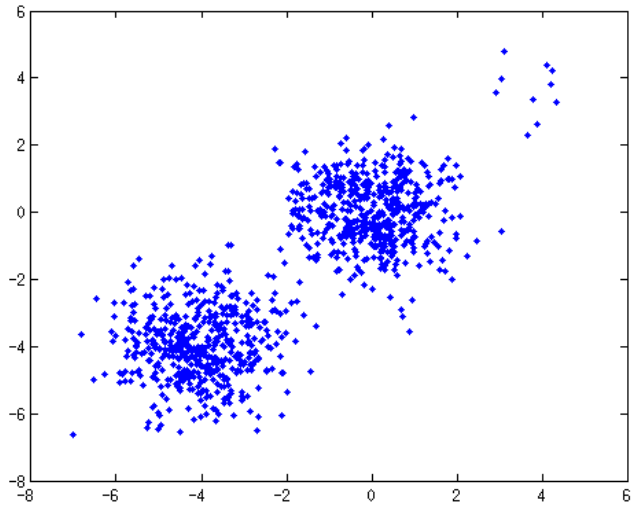
   (b) Choose k such that $k, k+1, \ldots, N$ have similar values of the cost function but that $1, \ldots, k-1$ have much higher costs, where all costs are evaluated from training data.

   (c) Set $k = \frac{n}{2}$.

   (d) Set $k = \log(\frac{n}{2})$.

   Solution: a and b.

2.2 Initialization Techniques



1. **[5 pts]** Recall the three different initialization techniques discussed in class for initializing cluster centers before running Lloyd's algorithm. For the distribution of data (plotted above), which method would give us the best initialization? Assume that the number of clusters $k = 2$. Please provide an explanation for why the technique you chose would produce better initial cluster centers than the other two.

Initialization method:

Short explanation: **Solution:** (c) k-means++. k-means++ will ignore the top-right outliers, and likely choose a center for each big cluster. Random initialization has the potential to choose two centers in the same big cluster, making clustering with Lloyd's algorithm more difficult. Meanwhile, furthest point initialization would likely choose a center in one of the big clusters and the other center as one of the top-right outliers, again making clustering with Lloyd's algorithm more difficult.

2. **[5 pts]** A 10601B TA has brilliantly come up with another initialization method for the cluster centers. The method chooses the next cluster center by randomly selecting a datapoint such that datapoints that have smaller distances to the previously-chosen cluster centers have a higher chance of being selected. Mathematically, the probability of choosing a cluster center $P(c_j = X_i)$ is proportional to 1 divided by the minimum distance between data point X_i and the previous centers $\min_{\ell<j} \frac{1}{\|c_\ell - X_i\|_2}$, where $c_j \neq c_\ell$. However, the instructors question the brilliance of this method.

List one advantage and one disadvantage of such a method:

Solution: This method is the flip of k-means++. One advantage is that it will avoid outliers, since points are not chosen substantially far from the other cluster centers. However, a

disadvantage is that if the data have one cluster far away from other clusters but has few datapoints, this cluster will likely be missed with initialization. This is why k-means++ weights datapoints that are further away more likely to be chosen. Other advantages/disadvantages are acceptable.

2.3 Hierarchical Clustering

Consider a data set of English words, in which all words have the same length. The distance measure between pairs of data points (words) is defined as the number of substitutions in letters required to transform one string to another. This is a modified version of the edit distance[1] between the two strings. For example, the distance between `baba` and `bebe` is 2, since we have to substitute two letters (`a` and `a` with `e` and `e`) in the first word to transform it into the second.

Consider the following words:
{`litter`, `lister`, `kitten`, `mitten`, `bitter`, `booted`}

These words have the following modified edit distances in the distance matrix:

	`litter`	`lister`	`kitten`	`mitten`	`bitter`	`booted`
`litter`	0	1	2	2	1	4
`lister`	1	0	3	3	2	4
`kitten`	2	3	0	1	2	4
`mitten`	2	3	1	0	2	4
`bitter`	1	2	2	2	0	3
`booted`	4	4	4	4	3	0

We now perform bottom-up hierarchical clustering using single or complete linkage, and at an intermediate stage, we have the following three clusters:

C1 {`litter`, `lister`}
C2 {`kitten`, `mitten`}
C3 {`bitter`, `booted`}

[6 pts] Which cluster will be merged with **C1** in the next iteration? **Circle the correct option, and then justify your answer.**

(a) **C2** with either single or complete linkage

(b) **C3** with either single or complete linkage

(c) **C2** with single-linkage or **C3** with complete-linkage

(d) **C3** with single-linkage or **C2** with complete-linkage

Justification:
Solution: Solution: (d). Single-linkage combines clusters with the smallest pairwise distance. Complete-linkage combines clusters with the smallest (across cluster pairs) of the largest pairwise distance (across word pairs between two clusters).

[1] While edit distance typically includes additions, deletions and substitutions as the possible edit operations, in this problem we count only substitutions.

3 Active learning [20 pts]

3.1 True/False

Circle one answer. No justification necessary.

1. **[2 pts]** Assume the problem is realizable. Consider a concept class H and a sequence of training instances $\{(x_t, y_t)\}_{t=1}^n$. Let $H_t \subseteq H$ be the version space after the learner has received the t-th instance. Then $H_t \neq \varnothing$, for all $t = 1, \ldots, n$.

   Circle one: True False

   Solution: True.

2. **[2 pts]** Assume the problem is realizable. Consider a concept class H and a sequence of training instances $\{(x_t, y_t)\}_{t=1}^n$. Let $H_t \subseteq H$ be the version space after the learner has received the t-th instance. After all n examples have been observed, we have $|H_n| = 1$.

   Circle one: True False

   Solution: False.

3. **[2 pts]** Let $X = \{(x_t, y_t)\}_{t=1}^n$ be a set of n training instances and H be the hypothesis space. Assume our current version space is $H' \subseteq H$. Pick another new sample (x_{t+1}, y_{t+1}). Let $\hat{H} \subseteq H$ be the version space of X after querying x_{t+1}, i.e., the version space of $X \cup \{(x_{t+1}, y_{t+1})\}$. Then it must be that $\hat{H} \neq H'$.

   Circle one: True False

   Solution: False.

4. **[2 pts]** In the agnostic case, it is possible that none of the version spaces in the hypothesis space H are consistent with all the training instances $\{(x_t, y_t)\}_{t=1}^n$.

   Circle one: True False

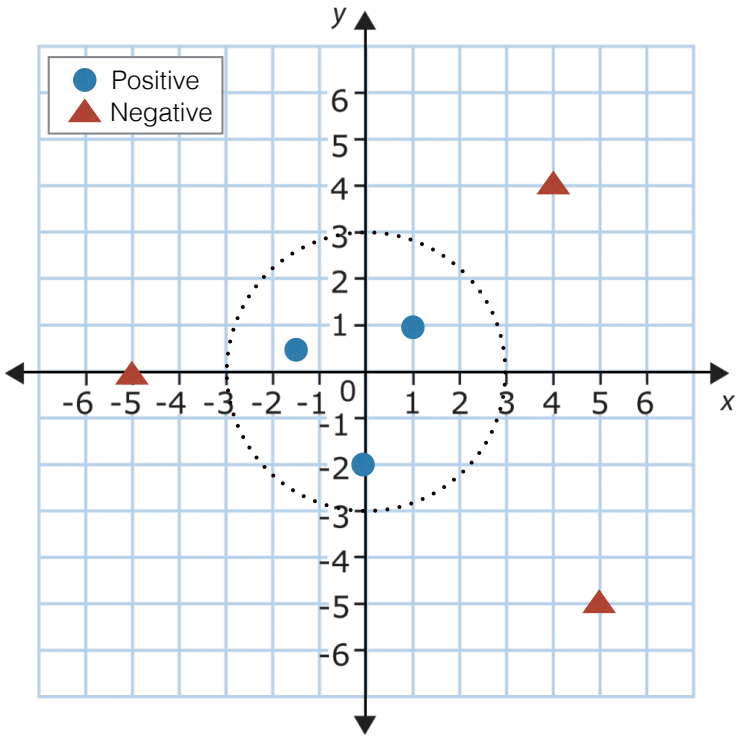
   Solution: True.

3.2 Short Answer

Provide short answers to the following questions.
Consider a binary classification problem where the sample space is $\Omega = \{(\mathbf{x}, y) : \mathbf{x} \in \mathbb{R}^2, y \in \{+1, -1\}\}$. Let the hypothesis space $H = \{a | a \geq 0\}$, where the decision rule is given by:

$$h_{\mathbf{a}}(\mathbf{x}) = \begin{cases} +1 & \|\mathbf{x}\|_2 < a, \\ -1 & \text{otherwise} \end{cases}$$

In this problem we assume that H is realizable. Suppose that you receive the sequence of 6 training instances shown in the figure below. An example hypothesis (with $a = 3$) is shown as the black dashed circle, where any point inside the circle is labelled as positive (+1).



1. **[4 pts]** Write down the version space $H_1 \subseteq H$ that is consistent with all the instances.

   Short answer:

   Solution: $H_1 = \{h_a(x) : 2 \leq a \leq 5\}$

2. **[4 pts]** Suppose we receive a new instance ($\mathbf{x}_7$ = (3.2, -2.4), $y_7 = +1$). Does this sample lie in the disagreement region DIS(H_1)? Write down the new version space $H_2 \subseteq H$ that is consistent with all the instances including the new instance.

   Circle one: Yes or no
   Short answer:

   Solution: Yes. Note that $\sqrt{3.2^2 + (-2.4)^2} = 4$. So, $H_2 = \{h_a(x) : 4 \leq a \leq 5\}$

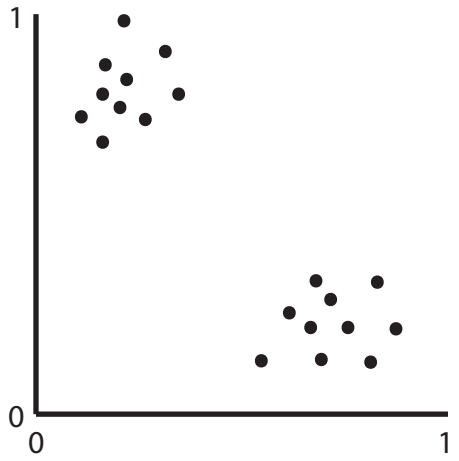
3. **[4 pts]** Which of the following unlabelled samples will change the version space, H_1?
   Circle all that apply:

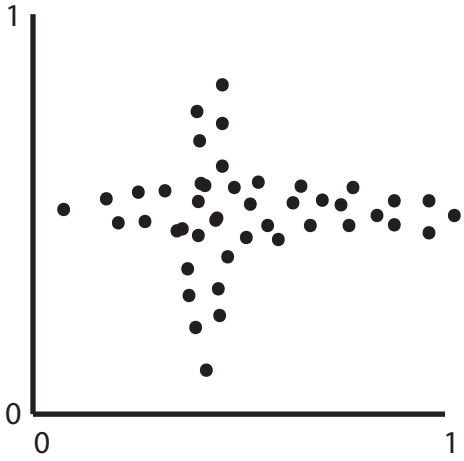
   (a) (0.0, 3.0)
   (b) (0.0, 4.0)
   (c) (0.0, 6.0)
   (d) (6.0, 6.0)
   (e) None of the above, since it depends on the labels of the samples.

   Solution: a and b

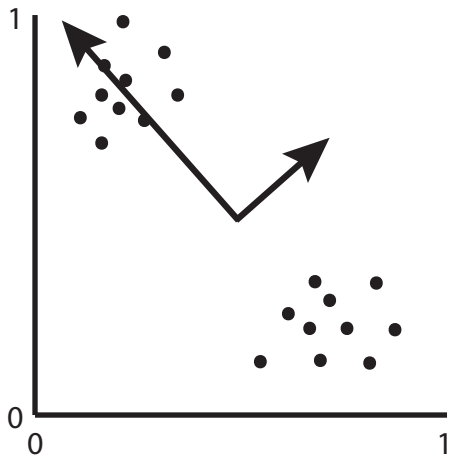
4 Principal Component Analysis [10 pts]

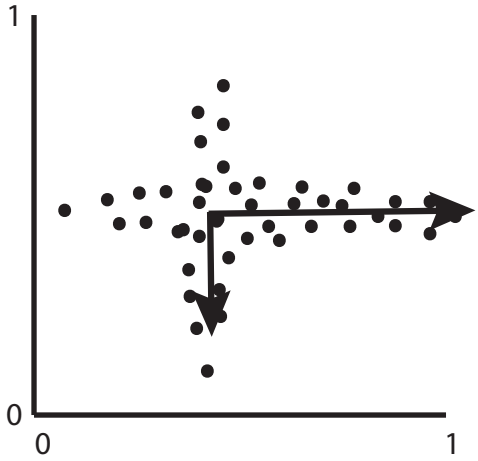
1. **[5 pts]** Consider the following two plots of data. Draw arrows from the mean of the data to denote the direction and relative magnitudes of the principal components.





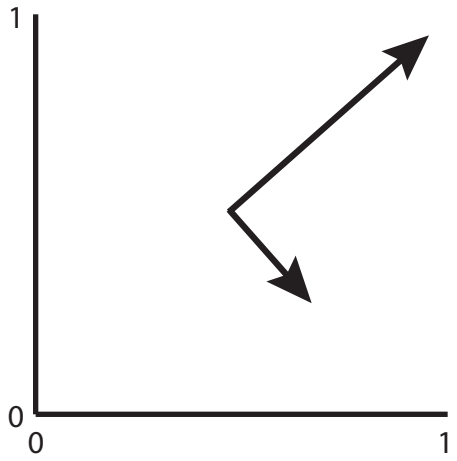
Solution:

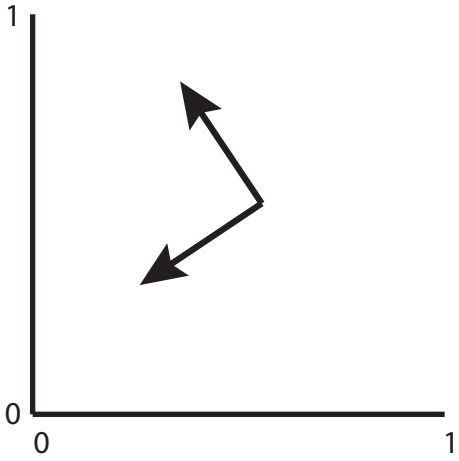


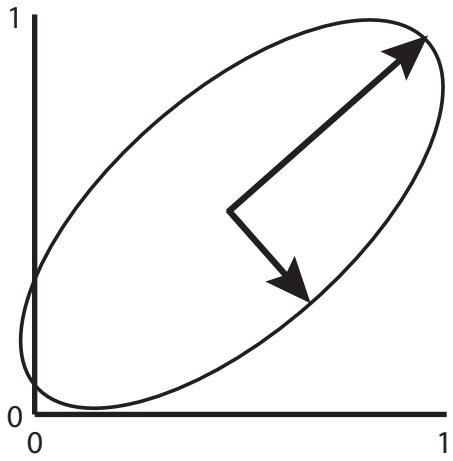


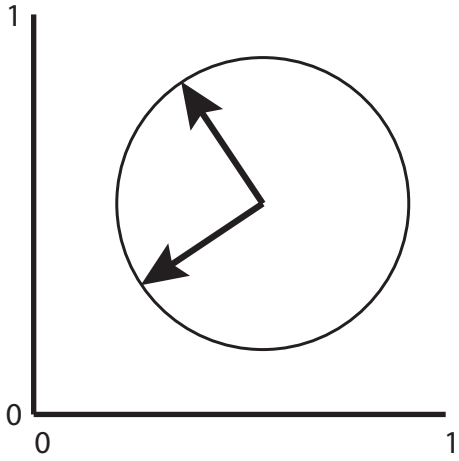
2. **[5 pts]** Now consider the following two plots, where we have drawn only the principal components. Draw the data ellipse or place data points that could yield the given principal components for each plot. Note that for the right hand plot, the principal components are of equal magnitude.

Solution:


1
0
0
1


1
0
0
1


1
0
0
1


1
0
0
1

5 Boosting and Semi-supervised learning [10 + 3 pts]

1. **[3 pts]** What condition must a weak learner satisfy in order for boosting to work?
Short answer:

   Solution: The weak learner must classify above chance performance.

2. **[3 pts]** After an iteration of training, AdaBoost more heavily weights which data points to train the next weak learner? (Provide an intuitive answer with no math symbols.)
Short answer:

   Solution: The data points that are incorrectly classified by weak learners trained in previous iterations are more heavily weighted.

3. **[4 pts]** In class, we talked about three different semi-supervised methods. List two of them.
Short answer:

   Solution: all three (only need to choose two): Transductive SVM, co-training, and graph-based methods.

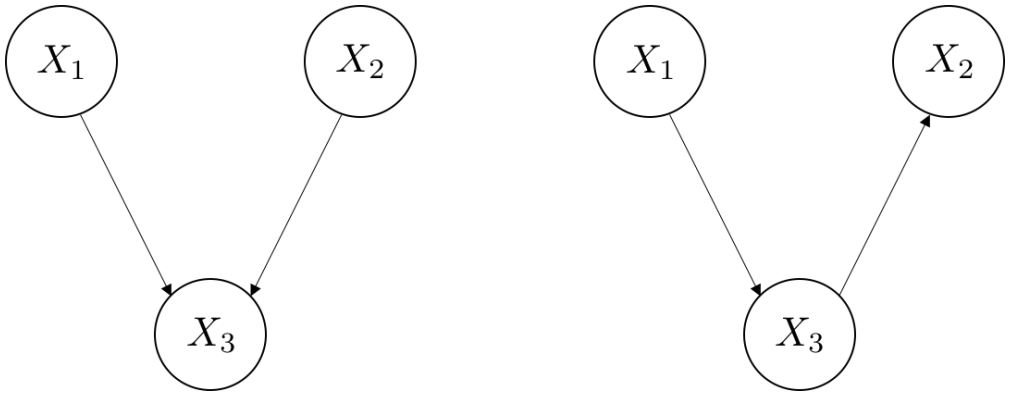
4. **[3 pts extra credit]** Do you think that a deep neural network is nothing but a case of boosting? Why or why not? Impress us.
Answer:

   Solution: Both viewpoints can be argued. One may view passing a linear combination through a nonlinear function as a weak learner (e.g., logistic regression), and that the deep neural network corrects for errors made by these weak learners in deeper layers. Then again, every layer of the deep neural network is optimized in a global fashion (i.e., all weights are updated simultaneously) to improve performance, which could possibly capture dependencies which boosting could not.

   Almost all coherent answers should be accepted, with full points to those who strongly argue their position with ML ideas.

6 Graphical Models [20 pts]

1. Consider the following two Bayesian networks.



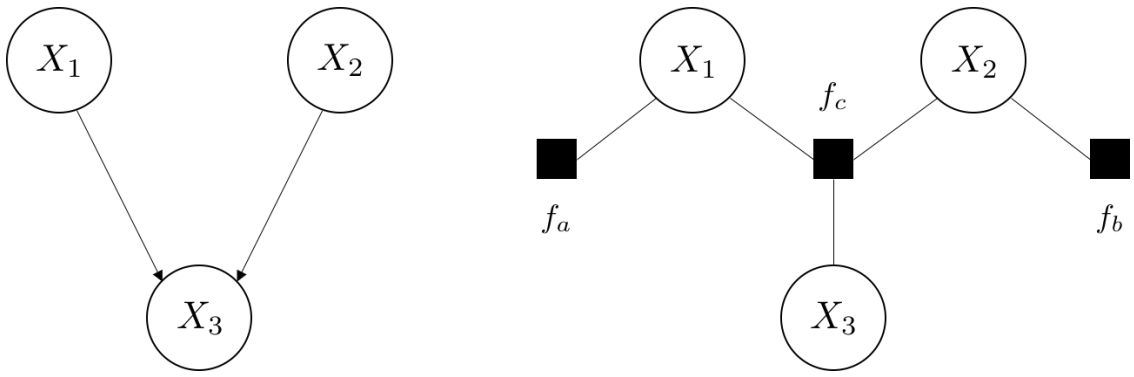
[3 pts] Do they have the same set of independence/conditional independence assumptions?

Circle one: Yes No

Please explain briefly in one sentence.

Solution: No. Given X_3, X_1 is conditionally independent from X_2 for the graph on the right, but given X_3, X_1 is *not* conditionally independent from X_2 for the graph on the left.

2. Consider the following Bayesian network and factor graph. Note the ordering of f_a, f_c, and f_b.



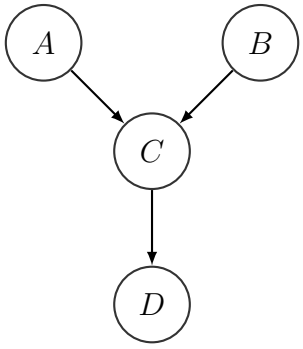
[3 pts] Can the Bayesian network on the left be represented by the factor graph on the right?

Circle one: Yes No

If so, please explicitly write out f_a, f_b, f_c in terms of the probability density functions (e.g., $P(X_1, X_2, X_3)$) such that the joint distribution of the factor graph factorizes in the same way as implied by the Bayesian network. If not, please give an example where the relationship between the variables X_1, X_2 and X_3 in the Bayesian network cannot be represented by the factor graph.
Answer:

Solution: Yes. Let $f_a(X_1) = P(X_1)$, $f_b(X_2) = P(X_2)$, and $f_c(X_1, X_2, X_3) = P(X_3|X_1, X_2)$.

3. Consider the following graphical model. Write out the graph's joint probability in a form that utilizes as many independence/conditional independence assumptions contained in the graph as possible.



[4 pts] Answer: P(A,B,C,D) =

Solution: $P(A, B, C, D) = P(A)P(B)P(C|A, B)P(D|C)$

4. Now assume that each random variable in the graphical model of question 3 is binary.

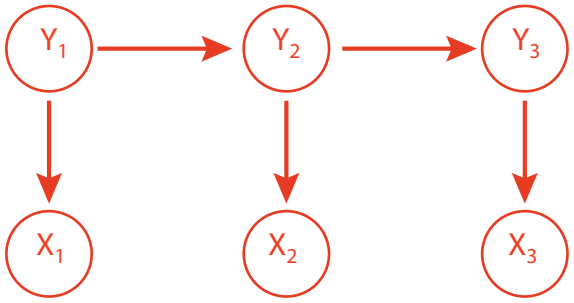
   (i) **[2 pts]** What is the minimum number of total parameters needed to model the joint distribution given the independencies/conditional independencies of the graphical structure? Assume that each binary random variable follows a Bernoulli distribution or a conditional Bernoulli distribution (e.g., $X|\{Y = y, Z = z\} \sim$ Bernoulli(p_{yz}), where p_{yz} depends on y and z).
   Answer:
   Solution: $1 + 1 + 4 + 2 = 8$

   (ii) **[2 pts]** If no graphical structure is given and we model the full joint distribution $P(A, B, C, D)$ explicitly using a large table, such that we could model any probability distribution. What is the minimum number of parameters needed to model the joint distribution?
   Answer: Solution: $2^4 - 1 = 15$

5. Recall that both the Hidden Markov Model (HMM) and Linear-Chain Conditional Random Fields (CRF) can be used to model sequential data with local dependence structures. In this question, let Y_t be the hidden state at time t, X_t be the observation at time t, $\mathbf{Y}$ be all the hidden states, and $\mathbf{X}$ be all the observations.

   (i) **[2 pts]** Draw the HMM as a Bayesian network where the observation sequence has length 3 (i.e., $t = 1, 2, 3$), labelling nodes with Y_1, Y_2, Y_3 and X_1, X_2, X_3.

   Solution:



   (ii) **[2 pts]** Write out the factorized joint distribution of $P(\mathbf{X}, \mathbf{Y})$ using the independencies/conditional independencies assumed by the HMM graph, using terms Y_1, Y_2, Y_3 and X_1, X_2, X_3.
   $P(\mathbf{X}, \mathbf{Y}) =$

   Solution:

   $P(\mathbf{X}, \mathbf{Y}) = P(Y_1)P(Y_2|Y_1)P(Y_3|Y_2)\prod_{t=1}^{3} P(X_t|Y_t)$

   (iii) **[2 pts]** Which of the following statements about HMM and CRF are **NOT** true? **Select all that apply. No justification needed.**

   (a) In both HMM and CRF, the objective of learning is to maximize the joint log likelihood $\log p(\mathbf{X}, \mathbf{Y}; \theta)$ of the training data.

   (b) Unlike HMM, CRF does NOT require its factors to be probability distributions.

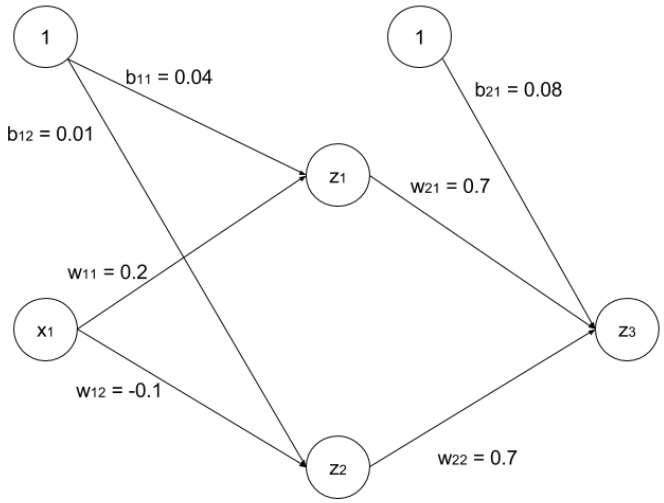
   (c) The predictive performance of CRF tends to be better than that of HMM in real-world data where the generative process is unknown.

   (d) In general, we can perform maximum likelihood estimation in closed form for HMM, but not CRF.

   Solution: (a). The CRF is not able to generate X, but rather is a discriminative method (optimizing for $P(Y|X, \theta)$).

7 Neural Nets, Deep Learning, and Boosting [20+3 pts]

7.1 Neural Networks



Consider the neural network architecture shown above for a 2-class $(0, 1)$ classification problem. The values for weights and biases are shown in the figure. We define:

$a_1 = w_{11}x_1 + b_{11}$
$a_2 = w_{12}x_1 + b_{12}$
$a_3 = w_{21}z_1 + w_{22}z_2 + b_{21}$
$z_1 = \text{relu}(a_1)$
$z_2 = \text{relu}(a_2)$
$z_3 = \sigma(a_3), \sigma(x) = \frac{1}{1+e^{-x}}$

Use this information to answer the questions that follow.

1. **[6 pts]** For $x_1 = 0.3$, compute z_3, in terms of e. **Show all work.**
   $z_3 =$

   Solution: $z_3 = \frac{1}{1+e^{-0.15}}$

2. **[2 pts]** To which class does the network predict the given data point ($x_1 = 0.3$), i.e., $\hat{y} =?$ Note that $\hat{y} = 1$ if $z_3 > \frac{1}{2}$, else $\hat{y} = 0$.
   Circle one: 0 1

   Solution: $\hat{y}(x_1 = 0.3) = 1$

3. **[6 pts]** Perform backpropagation on the bias b_{21} by deriving the expression for the gradient of the loss function $L(y, z_3)$ with respect to the bias term b_{21}, $\frac{\partial L}{\partial b_{21}}$, in terms of the partial derivatives $\frac{\partial \alpha}{\partial \beta}$, where α and β can be any of $L, z_i, a_i, b_{ij}, w_{ij}, x_1$ for all valid values of i, j. Your backpropagation algorithm should be as explicit as possible—that is, make sure each partial derivative $\frac{\partial \alpha}{\partial \beta}$ cannot be decomposed further into simpler partial derivatives. Do *not* evaluate the partial derivatives.

   $\frac{\partial L}{\partial b_{21}} =$

   Solution: $\frac{\partial L}{\partial b_{21}} = \frac{\partial L}{\partial z_3} \frac{\partial z_3}{\partial a_3} \frac{\partial a_3}{\partial b_{21}}$

4. **[6 pts]** Perform backpropagation on the bias b_{12} by deriving the expression for the gradient of the loss function $L(y, z_3)$ with respect to the bias term b_{12}, $\frac{\partial L}{\partial b_{12}}$, in terms of the partial derivatives $\frac{\partial \alpha}{\partial \beta}$, where α and β can be any of $L, z_i, a_i, b_{ij}, w_{ij}, x_1$ for all valid values of i, j. Your backpropagation algorithm should be as explicit as possible—that is, make sure each partial derivative $\frac{\partial \alpha}{\partial \beta}$ cannot be decomposed further into simpler partial derivatives. Do *not* evaluate the partial derivatives.

   $\frac{\partial L}{\partial b_{12}} =$

   Solution: $\frac{\partial L}{\partial b_{12}} = \frac{\partial L}{\partial z_3} \frac{\partial z_3}{\partial a_3} \frac{\partial a_3}{\partial z_2} \frac{\partial z_2}{\partial a_2} \frac{\partial a_2}{\partial b_{12}}$

7.2 [3 pts extra credit]

If the size of the input to a max-pooling layer with feature window (k) = 3 and stride (s) = 2 is 10×10, what is the size of the output? The input is padded with 0s (i.e., $p = 1$ and the padded input has size $k + 2p \times k + 2p$).

Solution: The size of the output is 5×5.

Use this page for scratch work