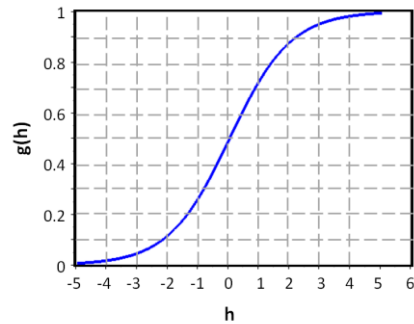
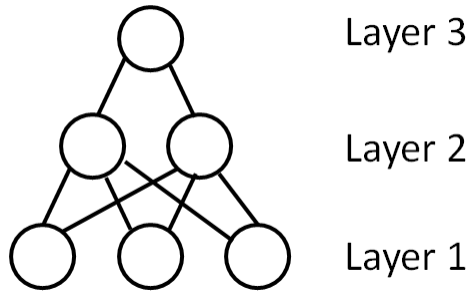


Final practice part 2

1. Consider the neural network below.



The initial weights are:

Layer 1:	$w_{1,1}^1 = -10$	$w_{1,2}^1 = 0$	$w_{1,3}^1 = -5$	$w_{1,4}^1 = 10$	$b_1^3 = 4$	Unit 1
	$w_{2,1}^1 = 20$	$w_{2,2}^1 = 0$	$w_{2,3}^1 = 10$	$w_{2,4}^1 = -5$	$b_1^3 = 4$	Unit 2
	$w_{3,1}^1 = 0$	$w_{3,2}^1 = -10$	$w_{3,3}^1 = 0$	$w_{3,4}^1 = 20$	$b_1^3 = 4$	Unit 3
Layer 2:	$w_{1,1}^2 = 5$	$w_{1,2}^2 = 10$	$w_{1,3}^2 = 0$	$b_1^3 = -2$		Unit 1
	$w_{2,1}^2 = 0$	$w_{2,2}^2 = -10$	$w_{2,3}^2 = 15$	$b_1^3 = -2$		Unit 2
Layer 3:	$w_{1,1}^3 = 10$	$w_{1,2}^3 = -20$	$b_1^3 = 5$			

Compute the output given the following inputs:

(a) Compute r_1^1, r_2^1, r_3^1 . Given the inputs: $x_1 = 5$ $x_2 = -10$ $x_3 = 10$ $x_4 = 0$

(b) Compute r_1^3 . Given the lower-layer outputs: $r_1^2 = 0.1$, $r_2^2 = 0.6$

(c) Compute r_2^2 . Given the lower-layer outputs: $r_1^1 = 0.1$, $r_2^1 = 0.3$, $r_3^1 = 0.6$

Sum inputs: $0.1 \times 0 + 0.3 \times -10 + 0.6 \times 15 - 2 = 0 - 3 + 9 - 2 = 4$

Use sigmoid $g(4)$: $r_2^2 = 0.99$

Compute the change in the specified weight based on the following input/outputs. In each case, presume the starting weight is as specified in the original list above. Assume $\epsilon = 1$

(d) Compute $\Delta w_{1,2}^3$. Given the layer 2 rates: $r_1^2 = 0.2$ and $r_2^2 = 0.8$; layer 3 rates: $r_1^3 = 0.1$; the desired output from r_1^3 is 1.0

$$\Delta w_{1,2}^3 = \epsilon(1 - r_1^3)(1 - r_1^2)r_1^2 r_2^2 = 1 \times (1 - 0.1) \times (1 - 0.2) \times 0.2 \times 0.8 = 1 \times 0.9 \times 0.8 \times 0.2 \times 0.8 = \mathbf{0.06}$$

(e) Compute $\Delta w_{1,2}^1$. Given the features: $x_1=10, x_2=-5, x_3=0, x_4=15$; $r_1^1=0.5, r_2^1=0.2, r_3^1=0.8$; delta values: $\delta_1^2 = -0.005, \delta_2^2 = 0.01$

2. For each of the following functions $f(x; h)$, compute the value of h that will maximize $f(x; h)$, assuming each function has a single maximum and no minimum.

(a) $f_1(\mathbf{x}; h) = \sum_i (-h^2 - 10hx_i + 12x_i^2)$

(b) $f_2(x; h) = e^{-(h^3+x^2)} = \exp(-(h^2 + x^2))$

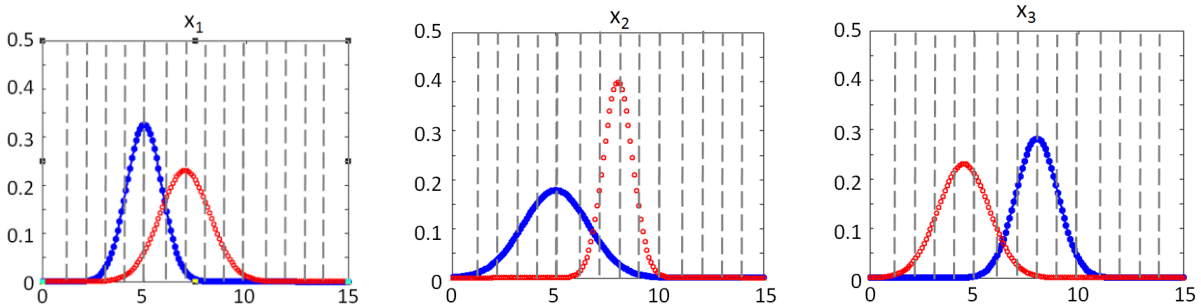
(c) $f_3(\mathbf{x}; h) = \prod_i 3h^{x_i}$

Set derivative equal to 0 ... or set derivative of log equal to 0
 $\log f_3 : \sum_i (\log 3 + x_i \log h)$

Set equal to 0: $\sum_i \frac{x_i}{h} = \frac{\sum_i x_i}{h} = 0$

As h approaches infinity, we approach 0
No actual final value for h that will make derivative exactly 0

3. Consider the following Gaussian likelihoods for features $x_1, x_2,$ and x_3 given **class = 1** (blue curves) or **class = 0** (red curves).



i. We wish to multiply these likelihoods together to compute $P(\mathbf{x} | y)$. Which type of classification is this:

- (a) Naïve Bayes Max-Posterior classification
- (b) Non-Naïve Bayes Max-Likelihood classification

- (c) Naïve Bayes Max-Posterior classification
- (d) Naïve Bayes Max-Likelihood classification
- (e) Support Vector Machine classification

ii. For the feature values below, which class is more probable (based on $P(\mathbf{x}|y)$ calculated from the plots above)?

(a) $x_1=5$ $x_2=7$ $x_3=6$

Class $y=1$: $P(x_1 | y=1) = 0.32$ $P(x_2 | y=1) = 0.1$ $P(x_3 | y=1) = 0.2$ -> total: 0.006
 Class $y=0$: $P(x_1 | y=0) = 0.05$ $P(x_2 | y=0) = 0.2$ $P(x_3 | y=0) = 0.1$ -> total: 0.001

Class $y=1$ most probable

(b) $x_1=8$ $x_2=8$ $x_3=6$

iii. Which class is more probable if we also incorporate the following prior:

$P(y=0) = 0.1$ $P(y=1) = 0.9$

to compute $P(y|\mathbf{x})$?

(a) $x_1=4$ $x_2=5$ $x_3=9$

(b) $x_1=6$ $x_2=7$ $x_3=7$

iv. Provide a prior that would make class 1 more probable if the \mathbf{x} values are:

$x_1=6$ $x_2=8$ $x_3=6$

Class $y=1$: $P(x_1 | y=1) = 0.2$ $P(x_2 | y=1) = 0.02$ $P(x_3 | y=1) = 0.05$ -> total: 0.0002
 Class $y=0$: $P(x_1 | y=0) = 0.15$ $P(x_2 | y=0) = 0.4$ $P(x_3 | y=0) = 0.1$ -> total: 0.006

Prior for $y=1$ ($P(y=1)$) needs to be at least 30x larger than prior for $y=0$:

e.g.,

$P(y=1) = 0.98$ $P(y=0) = 0.02$

4. Using each of the following kernel functions, compute the result of $K(\mathbf{x}^1, \mathbf{x}^2)$, for the specified input vectors.

$$K(\mathbf{c}, \mathbf{d}) = 2^{-(\mathbf{c}^T \mathbf{d} + 2)}$$

$$(a) \mathbf{c} = \begin{bmatrix} 4 \\ 0 \\ -2 \end{bmatrix} \quad \mathbf{d} = \begin{bmatrix} 0 \\ 0 \\ 1.5 \end{bmatrix}$$

$$(b) \mathbf{c} = \begin{bmatrix} 1 \\ 0.5 \\ -2 \end{bmatrix} \quad \mathbf{d} = \begin{bmatrix} 3 \\ -2 \\ -1 \end{bmatrix}$$

$$K(\mathbf{c}, \mathbf{d}) = (\mathbf{c}^T \mathbf{d} - 4)^2 + 10\mathbf{c}^T \mathbf{d}$$

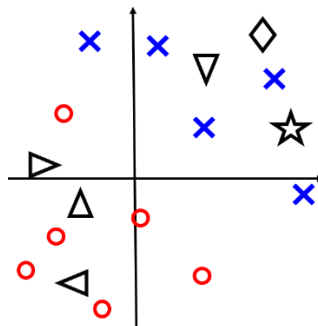
$$(c) \mathbf{c} = \begin{bmatrix} 0 \\ 3 \\ -2 \end{bmatrix} \quad \mathbf{d} = \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix}$$

$$\mathbf{c}^T \mathbf{d} = 0 + 0 - 4 = -4$$

$$(-4 - 4)^2 + 10 \times -4 = (-8)^2 - 40 = 64 - 40 = \mathbf{24}$$

$$(d) \mathbf{c} = \begin{bmatrix} -1 \\ 2 \\ -1 \end{bmatrix} \quad \mathbf{d} = \begin{bmatrix} 3 \\ 0 \\ -1 \end{bmatrix}$$

5. Consider the following training data. Red circles are class 0, blue x's are class 1, and all other shapes (triangles, stars, diamonds) are data points with known feature values but unknown labels.



Using the EM approach for learning, and assuming that we use a linear logistic classifier, how will the black triangles, diamonds, and star data points be used for learning? In the first round of EM, what y value do you expect each data point to be assigned, or no value at all?

First, could assign random y values to the non-circle and non-x shapes, then learn classifier, then in next round use learned separator/classifier to assign better guesses of classes.

6. Consider the classification problem with the following features and classes.

Class Person-type: Teenager, YoungProfessional, Adult, SeniorCitizen

Features:

Daily-time-online: 1-2 hours, 3-4 hours, 5-8 hours

Number-of-online-friends: 0-10, 10-50, 50-200, 200-1000

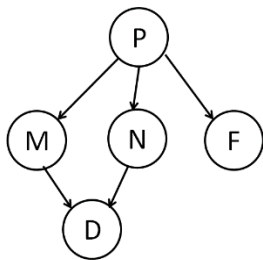
Favored content: News, SocialPosts, Education, Entertainment

Money-spent-online: None, \$1-\$50, \$50-\$100, \$100-\$500, \$500+

(a) How many parameters given Naïve Bayes a posteriori classification?

(b) How many parameters without Naïve Bayes (nor any Bayes net) likelihood classification?

(c) How many with the following Bayes nets likelihood classification?



Probability computed using $P(P)$, $P(M|P)$, $P(N|P)$, $P(F|P)$ and $P(D|M,N)$

$P(P) \leftarrow 4-1 = 3$ parameters \leftarrow ACTUALLY NOT NEEDED FOR LIKELIHOOD COMPUTATION

$P(M|P) = 4 \times (5-1) = 4 \times 4 = 16$ (# of dependents variable values \times # dependent var values)

$P(N|P) = 4 \times (4-1) = 4 \times 3 = 12$

$P(F|P) = 4 \times (4-1) = 4 \times 3 = 12$

$P(D|M,N) = (5 \times 4) \times (3-1) = 20 \times 2 = 40$

In total: $16 + 12 + 12 + 40 = 80$ or

Alternatively could be $4+16+12+12+40 = 84$

(d) What is the minimum number of training examples you would advise to use for the Bayes net from (c)?

Want training data to be at least double the number of parameters to learn (preferably 10x more or even greater). Take your answer from (b) and multiply by 2 (or 10). So 160, or 800.