בית ספר ״אפי ארזי״ למדעי המחשב המרכז הבינתחומי

The Efi Arazi school of computer science

The Interdisciplinary Center

סמסטר ב׳ תשע״ז

Spring 2018

מבחן מועד א בלמידה ממוכנת

Machine Learning Exam A

Lecturer: Prof Zohar Yakhini
Time limit: 3 hours
Additional material or calculators are not allowed in use!

Answer 5 out of 6 from the following question (each one is 20 points)

Good Luck!

מרצה: פרופ זהר יכיני
משך המבחן: 3 שעות
אין להשתמש בחומר עזר ואין להשתמש במחשבונים!

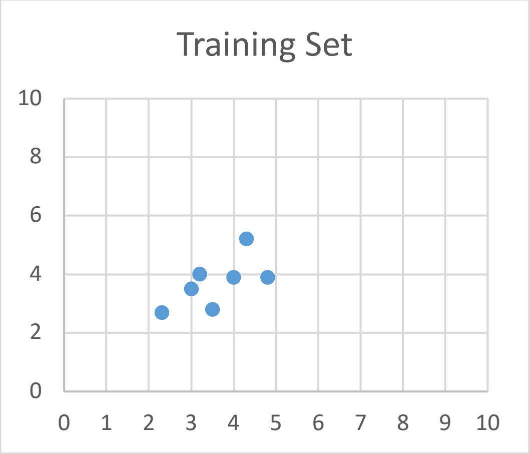
יש לענות על 5 מתוך 6 השאלות הבאות
לכל השאלות משקל שווה (20 נקודות)

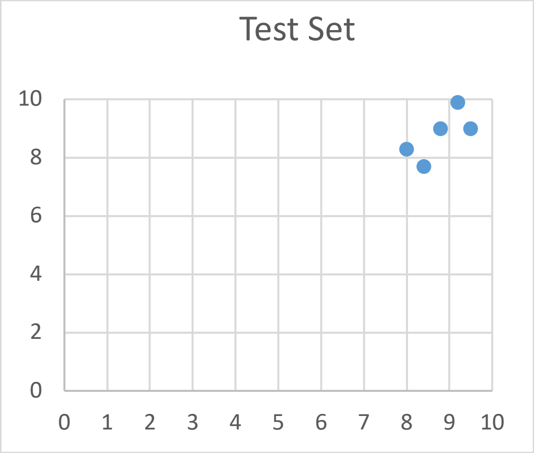
בהצלחה!

Question 1 (5 parts)

A. Write the formula for MSE (Mean Square Error) in the context of predicting the values of a function $y = f(x)$ (regression)

B. Consider the following training data and test data:





You are using one of the following approaches to predict y=f(x) using the training set:

a. Linear regression

b. 2-NN regression (regression using the closest 2 neighbors)

Which one of the two approaches has a smaller error on the **test** data?

C. How would you change the definition of MSE loss (as in A above) so that every instance can have a different weight, w_i, in computing the loss?

D. Write the pseudo code for linear regression, including the update step, using gradient descent in stochastic mode, as learned in class. How would you modify the algorithm to minimize the loss function you had defined in C?

E. Recall that linear regression seeks to find

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \ ||X\theta - y||_2^2$$

Find a matrix W to help you modify the above equation and define a pseudo-inverse solution for the function you defined in C.

Explain all your steps. In your solution, include the matrix used and the modified minimization task.

Question 1 (5 parts) – Solution

A. $MSE = \frac{1}{m}\sum_{i=1}^{m}(h(x_i) - y_i)^2$. Here, h is the prediction function and $y_i = f(x_i)$ are the observed values.

B. We can see the clear linear trend in both the training and test set.
This means the linear regression algorithm will have a low error on the test, in contrast the 2-nn algorithm will take the 2 points from the top right corner and their average ($y^i = \sim 4.5$) will be the prediction for each of the points in the test set which will yield a large error.

C. $WMSE = \frac{1}{m}\sum_{i=1}^{m} w_i(h(x_i) - y_i)^2$

D. Guess some initial values for $\theta_0, \theta_1, \ldots, \theta_n$

```
Repeat until error is small enough:
        for i=1..m
                for j=1 .. n
                        update $\theta_j = \theta_j - \alpha(< x_i, \theta > - y_i)x_{ij}$
```

And if we want to incorporate the weights:

$$\theta_j = \theta_j - \alpha(< x_i, \theta > - y_i)x_{ij}w_i$$

E.

$$W = \begin{bmatrix} \sqrt{w_1} & \cdots & \cdots & 0 \\ 0 & \sqrt{w_2} & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & \cdots & \sqrt{w_n} \end{bmatrix}$$

In words, diagonal matrix with the I entry of the diagonal is $\sqrt{w_i}$.

$$\theta^* = \underset{\theta}{\operatorname{argmin}} ||WX\theta - Wy||_2^2$$

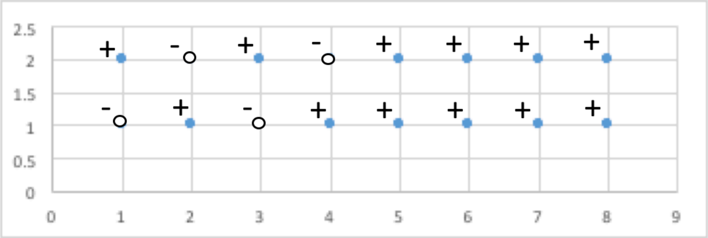
and then:

$$\theta^* = pinv(WX) \cdot Wy$$

Question 2 (4 parts)

Consider the following data table and its graphical representation. The data consists of instances with two numerical attributes, x1 and x2, coming from two classes “+” and “-“.
We use these data as training for learning a decision tree.
A tree of type T_N is a binary tree that uses Goodness of Split to perform splits and to grow up to height N or until no further split is possible.
For example:
T_1 will have one split (a root and two children)
T_2 will have one split at the root and then up to one split for each child.
* note: the tree doesn’t have to be symmetric.

instance	x1	x2	Value
1	1	2	+
2	2	1	+
3	3	2	+
4	4	1	+
5	5	1	+
6	5	2	+
7	6	1	+
8	6	2	+
9	7	1	+
10	7	2	+
11	8	1	+
12	8	2	+
13	1	1	-
14	2	2	-
15	3	1	-
16	4	2	-



A. Explain what an impurity function is and how Goodness of Split uses an impurity function φ to determine node splitting in a decision tree. Your explanation should use a clear formula.

B. Is it possible for a tree built using Goodness of Split to be of greater height than that of another tree built by splits that also get to pure leaves? If yes – provide an example. If not – clearly explain why.

C. Will the first split in a T_1 tree learned from training data be different from the first split in a T_3 learned from the same training data? Explain your answer.

D. When constructing trees of type T_1 and of type T_2 using the training data given above, what is the error obtained, evaluated by leave one out?
Which of the two approaches leads to a better result, as evaluated by leave one out?

Question 2 (4 parts) – Solution

A. Impurity measures the distance from perfect classification. The maximum value will be attained when the distribution is uniform, and the minimum value will be attained when there is perfect classification (all the instances in the node have the same class).
Goodness of Split checks the reduction in the impurity given the split feature. It calculates the impurity in the current node and then subtract the weighted average of the impurity in the children given the split feature.

$$Goodness\_of\_Split = \varphi(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} \varphi(S_v)$$

B. Yes. * Note that this was addresses during the semester and the we published an example.
For example (* We asked only for the example, the explanation wasn't mandatory. We did not expect you to provide the explanation):
Consider the following training data:

Instance No.	X1	X2	X3	Y
1	1	1	0	+
2	1	0	1	+
3	1	0	0	-
4	0	0	0	-
5	0	1	1	-

Where X1, X2 and X3 are binary attributes and Y is the target value (2 classes ⊕ and ⊖).
Recall that

$$Information\_Gain = Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v)$$

We calculate the information gain for each of the attributes, at the root node S. First – the entropy before splitting:

$$Entropy(S) = -\sum_{i=1}^{c} \frac{|S_i|}{|S|} \log \frac{|S_i|}{|S|} = -\frac{3}{5}\log\frac{3}{5} - \frac{2}{5}\log\frac{2}{5} = 0.97$$

And now, we consider the different split options.
Weighted average of the entropy according to X1:

$$-\sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v) = -\left(\frac{3}{5}\left(-\frac{2}{3}\log\frac{2}{3} - \frac{1}{3}\log\frac{1}{3}\right) + \frac{2}{5}\left(-\frac{2}{2}\log\frac{2}{2}\right)\right) = -0.55$$

Weighted average of the entropy according to X2:

$$-\sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v) = -\left(\frac{2}{5}\left(-\frac{1}{2}\log\frac{1}{2} - \frac{1}{2}\log\frac{1}{2}\right) + \frac{3}{5}\left(-\frac{1}{3}\log\frac{1}{3} - \frac{2}{3}\log\frac{2}{3}\right)\right) = -0.95$$

Weighted average of the entropy according to X3:

$$-\sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v) = -\left(\frac{2}{5}\left(-\frac{1}{2}\log\frac{1}{2} - \frac{1}{2}\log\frac{1}{2}\right) + \frac{3}{5}\left(-\frac{2}{3}\log\frac{2}{3} - \frac{1}{3}\log\frac{1}{3}\right)\right) = -0.95$$

Put it all together in the information gain formula we will get:

$$Information\_Gain(S, X1) = 0.97 - 0.55 = 0.42$$
$$Information\_Gain(S, X2) = 0.97 - 0.95 = 0.02$$
$$Information\_Gain(S, X3) = 0.97 - 0.95 = 0.02$$

And we will split according to the attribute X1.

We can see that there are 3 instances with X1=1. We can also see that the next single splitting step (according to X2 or X3) will not achieve perfect classification, due to the fact that for both X2 and X3 when the attribute value is 0 there is an instance with target value ⊖ and an instance with target value ⊕. Therefore, only after the third split we will achieve perfect separation.

Now check that by splitting on X2 first and then on X3 we achieve perfect classification with a shorter tree.

C. The first split will be the same in T_1 & T_3. The first split has no connection to the height of the tree. The instances & the features taken into account when deciding about the first split are the same in both cases and therefore the chosen feature will be the same.

D. Both T_1 & T_2 will have 8 errors in LOOCV. They both will make a wrong prediction on the 8 left instances and will predict correctly the 8 right instances. The extra split in T_2 doesn't help reduce the error for the 8 left instances as it will horizontally cut between them.

<u>Question 3 (5 parts)</u>

A. Find the minimum and the maximum of $x + 4y$ under the constraint $x^2 + 9y^2 = 1$

B. Given the following dataset (the XOR function):

X1	X2	Y
+1	+1	-1
+1	-1	+1
-1	+1	+1
-1	-1	-1

use the lemma below to show that it is not linearly separable. You do not need to prove the lemma.

<u>Lemma</u>

Assume that a linear classifier predicts the same $y \in \{-1, +1\}$ for some two points $z, z' \in \mathbb{R}^2$ (that is $h(z) = h(z')$). Then it will produce the same prediction for any intermediate point. That is:

$$\forall \alpha \in [0,1] \quad h\big((1-\alpha)z + \alpha z'\big) = y$$

C. Find a mapping φ into a space of a dimension of your choice that maps the dataset from Part B into a linearly separable dataset and define the linear classifier.

D. Consider the mapping $\varphi(x) = (1, x, x^2, x^3, \dots, x^N)$ for $x \in (-1, +1)$ (the open interval between -1 and +1) with some $N \in \mathbb{N}$. Show that a kernel function $K(x, y)$ exists for φ .

E. Show that the function $K(x, y) = \frac{1}{1-xy}$ for $x, y \in (-1, +1)$, is a kernel for some mapping into infinite dimensional space.

Question 3 (5 parts) – Solution

A.

$$L(x,y,\lambda) = x + 4y - \lambda(x^2 + 9y^2 - 1)$$

$$\nabla L_x = 1 - 2\lambda x = 0 \Rightarrow x = \frac{1}{2\lambda}$$

$$\nabla L_y = 4 - 18\lambda y = 0 \Rightarrow y = \frac{2}{9\lambda}$$

$$\nabla L_\lambda = x^2 + 9y^2 - 1 = 0$$

$$\frac{1}{4\lambda^2} + \frac{4}{9\lambda^2} = 1 \Rightarrow \lambda^2 = \frac{25}{36} \Rightarrow \lambda = \pm\frac{5}{6}$$

Then we get:

$$x = \pm\frac{3}{5}$$

$$y = \pm\frac{4}{15}$$

And therefore:

$max = \frac{5}{3}$ is attained at $\left(\frac{3}{5}, \frac{4}{15}\right)$ $min = -\frac{5}{3}$ is attained at $\left(-\frac{3}{5}, -\frac{4}{15}\right)$

Note that to determine max/min we only need to plug into $f(x) = x + 4y$.

B. Choose $\alpha = 0.5$ and the first and last points in the data to see that a linear classifier that perfectly predicts the data must classify the origin as −1. Likewise, use $\alpha = 0.5$ and the second and third points to show that the origin must also be classified as +1 to derive a contradiction.

C. $\varphi(x_1, x_2) = (x_1, x_2, x_1x_2)$.
One possible separator has weights $(w_1, w_2, w_3, b) = (0,0,-1,0)$, resulting in the predictor
$h(x_1, x_2) = sgn(< w, \varphi(x_1, x_2) >) = sgn(< (0,0,-1,0), (x_1, x_2, x_1x_2, 1) >) = sgn(-x_1x_2)$.

D. $\langle\varphi(x), \varphi(y)\rangle = 1 + xy + x^2y^2 + \cdots + x^Ny^N = 1 + xy + (xy)^2 + \cdots + (xy)^N$

$$= \frac{(xy)^{N+1} - 1}{xy - 1}$$

E. Taking φ from the previous section with $N = \infty$ we get:

$$\frac{1}{1 - xy} = \lim_{N\to\infty} \frac{(xy)^{N+1} - 1}{xy - 1} = \langle\varphi(x), \varphi(y)\rangle$$

Question 4 (4 parts)

Given an instance set S, we want to divide S into k groups (preform clustering).
The k-means-outlier algorithm is a variant of k-means given by the pseudocode below:

```
Initialize $c_1, …, c_k$ randomly
Loop:
    Assign all n samples to their closest $c_i$ and create k clusters $S_1, …, S_k$
    For each cluster $S_i$ ($1 \le i \le k$) define a new $c_i$:
            $b_i$ = the center of the cluster (average point)
            if $|S_i| > 2$ (if the number of samples in $S_i$ is larger than 2):
                    x = the sample with the highest distance from $b_i$
                    $c_i$ = the center of the cluster without x
            else:
                    $c_i = b_i$
Until no change in $c_1, …, c_k$
Return $c_1, …, c_k$
```

A. What is the function that the standard k-means algorithm seeks to minimize? Provide a formula.

B. Consider an instance set S of size 5 with 2 features as shown in the table. Run both the standard k-means algorithm and the k-manes-outlier algorithm with $k = 2$ and with the starting centers at $c_1 = (0, 0)$ and $c_2 = (2,2)$.
You might want to draw the points on a 2-dimensional grid. At each iteration write down the new centers and which center each point is assigned to. No need to show all intermediate calculations.

instance	x_1	x_2
p_1	1	0
p_2	0	1
p_3	2	1
p_4	1	2
p_5	6	6

C. Does the k-means-outlier algorithm converge? If so, prove it. If not, show an example where the algorithm fails to converge.

D. A version of a fuzzy-k-means algorithm goes according to the following pseudocode:

```
Initialize $c_1, …, c_k$ randomly
Loop:
        Calculate a distance vector for each sample with distances to each cluster
        For each sample j, convert the distance vector to probability vector

            $w_j = (w_{j1}, …, w_{jk})$
        For each cluster $S_i$ ($1 \le i \le k$) define a new center $c_i$:

            $c_i = \frac{\sum_{j=1}^{n} w_{ji} x_j}{\sum_{j=1}^{n} w_{ji}}$
Until no change in $c_1, …, c_k$
Return $c_1, …, c_k$
```

Write a suggested pseudo code for a fuzzy version of the k-means-outlier algorithm.

Question 4 (4 parts) – Solution

A. k-means minimizes, over all possible selections of centroids μ_i , $i = 1 \dots k$, the total distances of all points to their cluster centroids – the geographic mean of the cluster member points. Formally:

$$J = \sum_{i=1}^{k} \sum_{x \in S_i} \|x - \mu_i\|_2^2$$

B. k-means:

$c_1 = (0, 0)$
$c_2 = (2, 2)$

instance	x1	x2	C(p)
p1	1	0	1
p2	0	1	1
p3	2	1	2
p4	1	2	2
p5	6	6	2

$c_1 = (0.5, 0.5)$
$c_2 = (3, 3)$

instance	x1	x2	C(p)
p1	1	0	1
p2	0	1	1
p3	2	1	1
p4	1	2	1
p5	6	6	2

$c_1 = (1, 1)$
$c_2 = (6, 6)$

instance	x1	x2	C(p)
p1	1	0	1
p2	0	1	1
p3	2	1	1
p4	1	2	1
p5	6	6	2

k-means-outlier:

$c_1 = (0, 0)$
$c_2 = (2, 2)$

instance	x1	x2	C(p)
p1	1	0	1
p2	0	1	1
p3	2	1	2
p4	1	2	2
p5	6	6	2

$c_1 = (0.5, 0.5)$
$c_2 = (1.5, 1.5)$

instance	x1	x2	C(p)
p1	1	0	1
p2	0	1	1
p3	2	1	2
p4	1	2	2
p5	6	6	2

* Both the colors as well as the column C(p) indicate the cluster assignment of every point.

C. The k-means-outlier algorithm may not converge. For example:

instance	x1	x2
p1	0	0
p2	2	3
p3	3	2
p4	5.5	5.5
p5	8.5	9.5
p6	9.5	8.5

Where we initiate the centers to be:
$c_1 = (1, 1)$ and $c_2 = (6,6)$.

D.

```
Initialize $c_1, …, c_k$ randomly
Loop:
    Calculate a distance vector for each sample with distances to each cluster
    For each sample j, convert the distance vector to a probability vector
        $w_j = (w_{j1}, …, w_{jk})$
    For each cluster $S_i$ ($1 \leq i \leq k$) define a new $c_i$:
        $b_i$ = the center of the cluster (average point):
            $b_i = \frac{\sum_{j=1}^{n} w_{ji} x_j}{\sum_{j=1}^{n} w_{ji}}$
        if $|S_i| > 2$ (if the number of samples in $S_i$ is larger than 2):
            x = the sample with the highest distance from $b_i$
            r = index(x)
            $c_i$ = the center of the cluster without x:
                $c_i = \frac{\sum_{j \neq r} w_{ji} x_j}{\sum_{j \neq r} w_{ji}}$
        else:
            $c_i = b_i$
Until no change in c1, …, ck
Return $c_1, …, c_k$
```

Question 5 (4 parts)

A. Give an example of an instance space X and a binary hypothesis space H on X, such that:

$$VC(H) = 2018$$

B. Recall the 3 sample complexity bounds given in class:

- $m \geq \frac{1}{\varepsilon}\left(\ln|H| + \ln\frac{1}{\delta}\right)$
- $m \geq \frac{1}{\varepsilon^2}\left(\ln 2|H| + \ln\frac{1}{\delta}\right)$
- $m \geq \frac{1}{\varepsilon}\left(8 \cdot VC(H)\log_2\frac{13}{\epsilon} + 4\log_2\frac{2}{\delta}\right)$

Consider an instance space $X = [0,1] \times [0,1]$.

Let $N \in \mathbb{N}$ and $N \geq 2$, we define the set $\mathrm{A} := \{\frac{1}{N}, \frac{2}{N}, \dots, 1\}$ and the following hypothesis spaces:

- $H_1 = \{h\colon h(x_1, x_2) = +1 \Leftrightarrow x_1 \in [0, a] \wedge x_2 \in [0, a],\ a \in A\}$
- $H_2 = \{h\colon h(x_1, x_2) = +1 \Leftrightarrow x_1 \in [0, a] \wedge x_2 \in [0, b],\ a, b \in A\}$
- $H_3 = \{h\colon h(x_1, x_2) = +1 \Leftrightarrow x_1 \in [0, a] \wedge x_2 \in [0, b],\ a, b \in [0,1]\}$

You can view each hypothesis as an axis aligned rectangle with a vertex at the origin. (i.e. an instance is classified as positive iff it's inside the rectangle with vertices [(0,0), (0,a), (a,b), (0,b)])

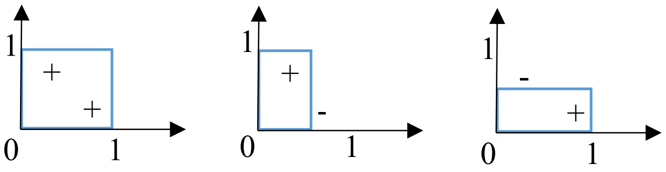
For each of the following cases, use one of the above bounds to compute the number of instances needed to guarantee an error of less than 0.1 with probability at least 95%:

1. When trying to learn a concept in H_3 using H_2.
2. When trying to learn a concept in H_1 using H_2.
3. When trying to learn a concept in H_3 using H_3.

Question 5 (4 parts) – Solution

A. There are several answers to this question. Few of them are:
   - H = all the linear classifier. $X = \mathbb{R}^{2017}$. We saw in class that linear classifiers have VC=d+1.
   - H = n-intervals where $n \leq 1009$. $X = \mathbb{R}$. We saw in class that the hypothesis space of n-intervals in $\mathbb{R}$ has VC=2n. Therefore, for $X = \mathbb{R}$ we have. VC(1009 intervals)=2018.

B. First, observe that $|H_1| = N, |H_2| = N^2, |H_3| = \infty$.
   1. Possibly $c \notin H_2$ and $|H_2|$ is finite, therefore we will use the second bound:
   $$m \geq 100(\ln 2 + 2\ln N + \ln 20)$$
   2. $c \in H_1 \subseteq H_2$, and $|H_2|$ is finite, therefore we will use the first bound:
   $$m \geq 10(2\ln N + \ln 20)$$
   3. Since $|H_3| = \infty$, we must use the third bound and compute the VC dimension. All the points that have coordinates outside the square [0,0] x [1,1] can't be classified as positive. Our instance space contains only points in this square.
   $VC \geq 2$:
   We can shutter the instance $\{(0.25, 0.75), (0.75, 0.25)\}$.
   $$a = b = 1 \begin{cases}(0.25, 0.75) \to + \\ (0.75, 0.25) \to +\end{cases}$$
   $$a = 0.5, b = 1 \begin{cases}(0.25, 0.75) \to + \\ (0.75, 0.25) \to -\end{cases}$$
   $$a = 1, b = 0.5 \begin{cases}(0.25, 0.75) \to - \\ (0.75, 0.25) \to +\end{cases}$$
   $$a = b = 0 \begin{cases}(0.25, 0.75) \to - \\ (0.75, 0.25) \to -\end{cases}$$
   In a picture:



   $VC < 3$:
   The left and the bottom edges are fixed to the axes. Given 3 points in the square [0,0] x [1,1] lets denote the point with the max x_1 among the 3 values of x_1 as z_1 and the point with the max x_2 among the 3 values of x_2 as z_2. There are 2 options:
   The first is that $z_1 = z_2$, in which case if this point is positive and the 2 others are negative, there is no hypothesis that can separate the positive from the negative (the rectangle will contain the 2 others points as it need to reach up to $z_1 = z_2$ on the upper right side).
   The second option is that $z_1 \neq z_2$. Still, when z_1 and z_2 are positive and the third point is negative, there is no hypothesis that can separate the positive from the negative (the rectangle will contain the third point as its upper edge reaches up to z_2 and its right edge reaches right to z_1).
   We got that $VC = 2$ and then:
   $$m \geq 10(16\log_2 130 + 4\log_2 40)$$

Question 6 (4 parts)

Consider a quality test that consists of measuring two quantitative features x1 and x2. We know, based on long-term measurement history, that the class-conditional probability density functions for these features are given below (G and B denote the two classes G =Good and B =Bad).

For the first feature

$$f(x_1|G) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(x-2)^2}{2}\right)$$
$$f(x_1|B) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(x-5)^2}{2}\right)$$

and, for the second feature

$$f(x_2|G) = \begin{cases} \frac{1}{3} & 0 \leq x_2 \leq 1 \\ \frac{2}{3} & 2 \leq \mathrm{x}_2 \leq 3 \\ 0 & otherwise \end{cases}$$

$$f(x_2|B) = \begin{cases} e^2 & 3 - \frac{1}{e^2} \leq x_2 \leq 3 \\ 0 & otherwise \end{cases}$$

Sketching out these distributions might help your intuition.

A. What would the ML prediction be in the following (separate) two cases?
   1. We have measured, for a certain product, the value $x_2 = 2.9$
   2. We have measured, for a different product, the value $x_1 = 3$

B. Assuming a prior P(B), clearly state the Naïve Bayes MAP formula for this quality test. What is the minimal prior P(B) for which the Naïve Bayes MAP prediction, in Case A2 above, would be B?

C. Assume that for A2 we also measured $x_2 = 2.99$
   In this case what is the minimal prior P(B) for which the Naïve Bayes MAP prediction would be B?

D. For a third case we measured $x_1 = 6$ and $x_2 = 0.975(3 - \frac{1}{e^2})$. Assume that P(B) = 0.9. What is the MAP prediction in this case? Do you think that this represents a bias or a shortcoming of the classification approach? How would you remedy this problem?

<u>Question 6 (4 parts) – Solution</u>

A.

1. $f(x_2 = 2.9|G) = \frac{2}{3} < e^2 = f(x_2 = 2.9|B) \Rightarrow Prediction\ will\ be\ B$
2. $f(x_1 = 3|G) = \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}\right) > \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{4}{2}\right) = f(x_1 = 3|B) \Rightarrow$ $Prediction\ will\ be\ G$

B. The naïve Bayes MAP approach will select the class C (either G or B) that maximizes the aposteriori probability of the observed data. It also assumes that x1 and x2 are independent given the class. It therefore maximizes:
P(C|x1,x2)= P(C)p(x1,x2|C) = P(C)p(x1|C)p(x2|C)

We want P(B)p(x1|B) >= P(G)p(x1|G)
So, setting P(B)=p and P(G) = 1-p, we get:

$$pf(x_1|B) \geq (1-p)f(x_1|G)$$

Which leads to:

$$p\big(f(x_1|B) + f(x_1|G)\big) \geq f(x_1|G)$$

And then to:

$$p \geq \frac{f(x_1|G)}{f(x_1|G) + f(x_1|B)} = \frac{\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1^2}{2}\right)}{\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1^2}{2}\right) + \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{(-2)^2}{2}\right)} = \\ = \frac{e^{-\frac{1}{2}}}{e^{-\frac{1}{2}} + e^{-2}}$$

C. We want P(B)p(x1|B) p(x2|B) >= P(G)p(x1|G) p(x2|G)
So, setting P(B)=p and P(G) = 1-p, we get:

$$pf(x_1|B)f(x_2|B) \geq (1-p)f(x_1|G)(x_2|G)$$

Which leads to:

$$p\big(f(x_1|B)f(x_2|B) + f(x_1|G)(x_2|G)\big) \geq f(x_1|G)(x_2|G)$$

And then to:

$$p \geq \frac{f(x_1|G)f(x_2|G)}{f(x_1|G)f(x_2|G) + f(x_1|B)f(x_2|B)} =$$

$$\frac{\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1^2}{2}\right)\frac{2}{3}}{\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1^2}{2}\right)\frac{2}{3} + \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{(3-5)^2}{2}\right)e^2} = \frac{\frac{2}{3}e^{-\frac{1}{2}}}{\frac{2}{3}e^{-\frac{1}{2}} + 1}$$

D. Since $f\left(x_2 = 0.975\left(3 - \frac{1}{e^2}\right)\middle|B\right) = 0$ The MAP prediction will be G, as $f\left(x_2 = 0.975\left(3 - \frac{1}{e^2}\right)\middle|G\right)$ is not zero.
This is not reasonable as the x_1 observation is strongly indicative of B and x_2 observation is very close to the boundary.
To remedy this, we would like to artificially modify $f(x_2|B)$ to be:

$$f(x_2|B) = \begin{cases} e^2 & 3 - \frac{1}{e^2} \leq x_2 \leq 3 \\ \varepsilon & \text{Otherwise} \end{cases}$$

For some small $\varepsilon > 0$.
* This answer will receive full credit.

However –
F as defined above is NOT a probability density function (why?).
Take any two functions g(x) and h(x) so that:

$$g(x) = \int_{-\infty}^{3-\frac{1}{e^2}} g(t)dt = 1$$

$$h(x) = \int_{3}^{\infty} h(t)dt = 1$$

And then our correction is:

$$f(x_2|B) = \begin{cases} e^2 - e^2\varepsilon & 3 - \frac{1}{e^2} \le x_2 \le 3 \\ \frac{\varepsilon}{2} g(x_2) & x_2 < 3 - \frac{1}{e^2} \\ \frac{\varepsilon}{2} h(x_2) & x_2 > 3 \end{cases}$$