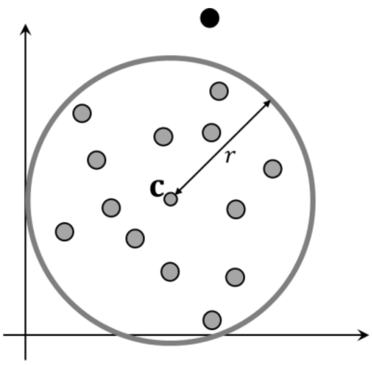
CS 771A: Intro to Machine Learning, IIT Kanpur				**Endsem Exam** (27 Apr 2025)
Name				**40 marks** Page **1** of **4**
Roll No		Dept.		

sInstructions:

1. This question paper contains 2 pages (4 sides of paper). Please verify.
2. Write your name, roll number, department in **block letters** with **ink** on **each page**.
3. Write your final answers neatly **with a blue/black pen**. Pencil marks may get smudged.
4. Don't overwrite/scratch answers especially in MCQ – ambiguous cases will get 0 marks.

Q1 (Anomalous Thresholds) Melba is designing an anomaly detection algorithm, using a neural network to embed all data points as 2D vectors and finding the mean vector $\mathbf{c}$. Melba now wants to find a radius threshold r such that given a test data point with 2D embedding $\mathbf{x}$, Melba can label it as $\hat{y} = +1$ (anomalous) if $\|\mathbf{x} - \mathbf{c}\|_2 > r$ else label it as $\hat{y} = -1$ (normal). Melba has the following test data containing normal and anomalous points. To simplify, the Euclidean distance of all test points from mean $\mathbf{c}$ and their ground truth label y ($y = +1$ for anomalous, $y = -1$ for normal) is given.



$\|\mathbf{x} - \mathbf{c}\|_2$	1.0	2.5	3.0	4.5	5.0	6.5	7.0	8.5	9.0	10.5
true label y	-1	-1	-1	$+1$	-1	-1	-1	$+1$	-1	$+1$

For each part, answer in the box and also justify briefly in space provided. **((1+1) x 3 = 6 marks)**

Is there a radius threshold that achieves perfect test accuracy i.e. i.e. $\mathbb{P}[\hat{y} = y] = 1$? Write **T** or **F**. Justify your answer briefly.

F

There exists no real number $\theta \in \mathbb{R}$ such that all anomalous data points all have distances more than θ and all normal points have distances less than θ.

Give (any) one radius threshold for which Melba's classifier achieves perfect true-positive rate (TPR) on test data i.e. $\mathbb{P}[\hat{y} = 1 | y = 1] = 1$. Justify your answer briefly.

4.49

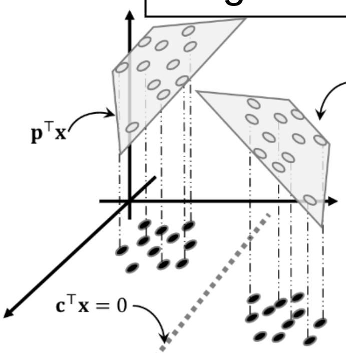
To achieve 100% TPR, a classifier must classify every anomalous data point as anomalous. Since the closest anomalous data point is distance 3 away from the center, any radius threshold strictly smaller than 4.5 will get 100% TPR (strictly smaller since Melba labels $\hat{y} = +1$ if $\|\mathbf{x} - \mathbf{c}\|_2 > r$ and not if $\|\mathbf{x} - \mathbf{c}\|_2 \geq r$).

Among radius thresholds that achieve perfect TPR, find (any) one that minimizes the false-positive rate (FPR) i.e. $\mathbb{P}[\hat{y} = 1 | y = -1]$ is smallest. Justify your answer briefly.

3.0

Radius thresholds that achieve 100% TPR must misclassify at least 4 normal points as anomalous. If the radius threshold drops below 3.0 it will classify 5 or 6 or 7 normal points as anomalous. Thus, the minimum possible FPR is $\frac{4}{7}$ for thresholds achieving 100% TPR – this is achieved by any radius threshold in the range $[3.0, 4.5)$.

Q2 (Probabilistic DT) Melbo wants to solve a mixed regression problem using two regression models $\mathbf{p}, \mathbf{q} \in \mathbb{R}^d$. A classifier $\mathbf{c} \in \mathbb{R}^d$ is also needed to decide which model to use at test time (see figure). For a data point $\mathbf{x} \in \mathbb{R}^d$, if $\mathbf{c}^\top \mathbf{x} \geq 0$, Melbo will predict $\mathbf{p}^\top \mathbf{x}$. If $\mathbf{c}^\top \mathbf{x} < 0$, predict $\mathbf{q}^\top \mathbf{x}$. Assume that bias is hidden inside the models. Note that this is nothing but a regression tree with one root and two leaves. **(4 x 4 = 16 marks)**


$\mathbf{q}^\top \mathbf{x}$
$\mathbf{p}^\top \mathbf{x}$
$\mathbf{c}^\top \mathbf{x} = 0$

Melbo has training data $(\mathbf{x}^i, y^i), i \in [N]$ where $\mathbf{x}^i \in \mathbb{R}^d, y^i \in \mathbb{R}$ but does not know which data point belongs to which model so Melba advises using latent variables. For each data point $i \in [N]$, Melbo uses a binary latent variable $z^i \in \{-1, +1\}$ and the following likelihoods: $\mathbb{P}[z^i \mid \mathbf{x}^i, \mathbf{p}, \mathbf{q}, \mathbf{c}] = \sigma(z^i \cdot \mathbf{c}^\top \mathbf{x}^i)$ and $\mathbb{P}[y^i \mid \mathbf{x}^i, z^i = +1, \mathbf{p}, \mathbf{q}, \mathbf{c}] = \frac{1}{\sqrt{\pi}} \exp\left(-\left(y^i - \mathbf{p}^\top \mathbf{x}^i\right)^2\right)$ and $\mathbb{P}[y^i \mid \mathbf{x}^i, z^i = -1, \mathbf{p}, \mathbf{q}, \mathbf{c}] = \frac{1}{\sqrt{\pi}} \exp\left(-\left(y^i - \mathbf{q}^\top \mathbf{x}^i\right)^2\right)$ where $\sigma(t) \overset{\text{def}}{=} \frac{1}{(1+\exp(-t))}$ is the sigmoid.

Give brief derivation of an expression for the exact likelihood $\mathbb{P}[y^i \mid \mathbf{x}^i, \mathbf{p}, \mathbf{q}, \mathbf{c}]$ by eliminating z^i.

The law of total probability gives us

$$\mathbb{P}[y^i \mid \mathbf{x}^i, \mathbf{p}, \mathbf{q}, \mathbf{c}] = \sum_{z^i \in \{-1,+1\}} \mathbb{P}[y^i \wedge z^i \mid \mathbf{x}^i, \mathbf{p}, \mathbf{q}, \mathbf{c}]$$

$$= \sum_{z^i \in \{-1,+1\}} \mathbb{P}[y^i \mid \mathbf{x}^i, z^i, \mathbf{p}, \mathbf{q}, \mathbf{c}] \cdot \mathbb{P}[z^i \mid \mathbf{x}^i, \mathbf{p}, \mathbf{q}, \mathbf{c}]$$

$$= \frac{1}{\sqrt{\pi}} \cdot \left(\sigma(\mathbf{c}^\top \mathbf{x}^i) \cdot \exp\left(-\left(y^i - \mathbf{p}^\top \mathbf{x}^i\right)^2\right) + \left(1 - \sigma(\mathbf{c}^\top \mathbf{x}^i)\right) \cdot \exp\left(-\left(y^i - \mathbf{q}^\top \mathbf{x}^i\right)^2\right)\right)$$

As exact MLE is hard, Melbo instead tries to solve $\underset{\mathbf{p},\mathbf{q}}{\arg\max}\, \underset{\{z^i\}}{\arg\max}\, \underset{\mathbf{c}}{\arg\max} \{\mathcal{L}(\mathbf{p}, \mathbf{q}, \mathbf{c}, \{z^i\})\}$ with $\mathcal{L}(\mathbf{p}, \mathbf{q}, \mathbf{c}, z^i) = \sum_{i \in [N]} \ln(\mathbb{P}[y^i \mid \mathbf{x}^i, z^i, \mathbf{p}, \mathbf{q}, \mathbf{c}]) + \sum_{i \in [N]} \ln(\mathbb{P}[z^i \mid \mathbf{x}^i, \mathbf{p}, \mathbf{q}, \mathbf{c}])$ using alternating optimization. **You are free to use simple operations like least squares, logistic regression directly**.

Step 1: Freeze $\mathbf{c}, \{z^i\}$ and give brief derivation on how to find $\arg\max_{\mathbf{p},\mathbf{q}} \mathcal{L}(\mathbf{p}, \mathbf{q}, \mathbf{c}, \{z^i\})$.

The second set of terms in $\mathcal{L}(\mathbf{p}, \mathbf{q}, \mathbf{c}, \{z^i\})$ i.e. $\sum_{i \in [N]} \ln(\mathbb{P}[z^i \mid \mathbf{x}^i, \mathbf{p}, \mathbf{q}, \mathbf{c}])$ do not participate here as these do not depend on $\mathbf{p}, \mathbf{q}$. For the first set of terms, the summation can be broken into terms that depend on $\mathbf{p}$ and that depend on $\mathbf{q}$ to give (after taking NLL)

$$\arg\max_{\mathbf{p},\mathbf{q}} \mathcal{L}(\mathbf{p}, \mathbf{q}, \mathbf{c}, \{z^i\}) = \arg\min_{\mathbf{p},\mathbf{q}} \sum_{i \in [N]} -\ln(\mathbb{P}[y^i \mid \mathbf{x}^i, z^i, \mathbf{p}, \mathbf{q}, \mathbf{c}])$$

$$= \arg\min_{\mathbf{p}} \sum_{z^i = +1} \left(y^i - \mathbf{p}^\top \mathbf{x}^i\right)^2 + \arg\min_{\mathbf{q}} \sum_{z^i = -1} \left(y^i - \mathbf{q}^\top \mathbf{x}^i\right)^2$$

The above can be solved by invoking a least squares solver twice, once over points with $z^i = +1$ to give us $\mathbf{p}$ and another time over points with $z^i = -1$ to give us $\mathbf{q}$.

Step 2: Freeze $\mathbf{p}, \mathbf{q}, \mathbf{c}$ and give brief derivation on how to find $\arg\max_{\{z^i\}} \mathcal{L}(\mathbf{p}, \mathbf{q}, \mathbf{c}, \{z^i\})$.

We show the process for a single z^i which can be repeated for all others.

$$\arg\max_{z^i} \mathcal{L}(\mathbf{p}, \mathbf{q}, \mathbf{c}, z^i) = \arg\min_{z^i} \{-\ln(\mathbb{P}[y^i \mid \mathbf{x}^i, z^i, \mathbf{p}, \mathbf{q}, \mathbf{c}]) - \ln(\mathbb{P}[z^i \mid \mathbf{x}^i, \mathbf{p}, \mathbf{q}, \mathbf{c}])\}$$

For $z^i = +1$, the objective takes the value $V_i^+ = \ln\sqrt{\pi} + (y^i - \mathbf{p}^\top \mathbf{x}^i)^2 + \ln(1 + \exp(-\mathbf{c}^\top \mathbf{x}^i))$

For $z^i = -1$, the objective takes the value $V_i^- = \ln\sqrt{\pi} + (y^i - \mathbf{q}^\top \mathbf{x}^i)^2 + \ln(1 + \exp(\mathbf{c}^\top \mathbf{x}^i))$

Thus, the optimal value of $z_i = \text{sign}(V_i^- - V_i^+)$ with ties broken arbitrarily.

Note: unlike the mixed regression case we discussed in lectures, where we chose the latent variable simply by looking at which model gave us smaller regression error, the choice here also must take care that the chosen latent variable value is easy to learn by the classifier $\mathbf{c}$.

Step 3: Freeze $\mathbf{p}, \mathbf{q}, \{z^i\}$ and give brief derivation on how to find $\arg\max_{\mathbf{c}} \mathcal{L}(\mathbf{p}, \mathbf{q}, \mathbf{c}, \{z^i\})$.

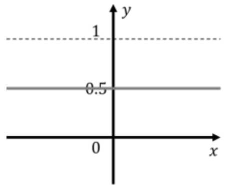
The first set of terms in $\mathcal{L}(\mathbf{p}, \mathbf{q}, \mathbf{c}, \{z^i\})$ i.e. $\sum_{i \in [N]} \ln(\mathbb{P}[y^i \mid \mathbf{x}^i, z^i, \mathbf{p}, \mathbf{q}, \mathbf{c}])$ do not participate here as these do not depend on $\mathbf{c}$. For the second set of terms, we get (after taking NLL)

$$\arg\max_{\mathbf{c}} \mathcal{L}(\mathbf{p}, \mathbf{q}, \mathbf{c}, \{z^i\}) = \arg\min_{\mathbf{c}} \sum_{i \in [N]} -\ln(\mathbb{P}[z^i \mid \mathbf{x}^i, \mathbf{p}, \mathbf{q}, \mathbf{c}])$$

$$= \arg\min_{\mathbf{c}} \sum_{i \in [N]} \ln(1 + \exp(-z^i \cdot \mathbf{c}^\top \mathbf{x}^i))$$

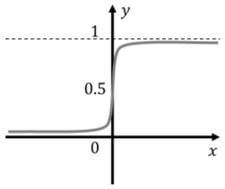
The above can be solved by invoking a logistic regression solver using z^i as the binary “labels”.

Q3 (Sharp Sigmoids) The sigmoid function becomes more expressive by introducing a bandwidth parameter B as $\sigma(x; B) \overset{\text{def}}{=} 1/(1 + \exp(-B \cdot x))$. For each of the following five curves, select the value of B that best generates that curve. **Shade only one circle in each part**. **(5 x 1 = 5 marks)**



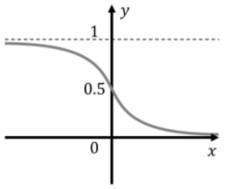
(a)

○ $B \to -\infty$
○ $B = -1$
● $B = 0$
○ $B = +1$
○ $B \to +\infty$



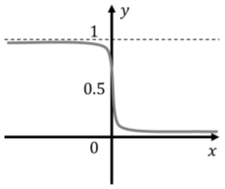
(b)

○ $B \to -\infty$
○ $B = -1$
○ $B = 0$
○ $B = +1$
● $B \to +\infty$



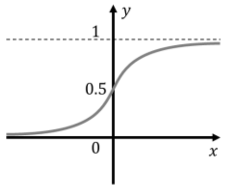
(c)

○ $B \to -\infty$
● $B = -1$
○ $B = 0$
○ $B = +1$
○ $B \to +\infty$



(d)

● $B \to -\infty$
○ $B = -1$
○ $B = 0$
○ $B = +1$
○ $B \to +\infty$



(e)

○ $B \to -\infty$
○ $B = -1$
○ $B = 0$
● $B = +1$
○ $B \to +\infty$

Q4 (Total Confusion) Melbu used a linear model $\text{sign}(\mathbf{w}^\top\mathbf{x} + b)$ to solve a binary classification problem with model vector $\mathbf{w} \in \mathbb{R}^d$ and bias $b \in \mathbb{R}$. The classifier was evaluated on 1000 test data points and the following confusion matrix was obtained. Note that y denotes the true label of a test point and $\hat{y}$ denotes the label predicted by the classifier. The entries in the confusion matrix show how many points of a particular class were classified in a particular manner by the classifier. There are only two classes namely $-1, +1$. Calculate the following quantities for the classifier based on its test performance (no derivations needed) **(4 x 1 + 2 = 6 marks)**

	$\hat{y} = 1$	$\hat{y} = -1$
$y = 1$	100	700
$y = -1$	100	100

Accuracy $\mathbb{P}[\hat{y} = y]$ 0.2

Recall $\mathbb{P}[\hat{y} = 1 | y = 1]$ 0.125

Precision $\mathbb{P}[y = 1 | \hat{y} = 1]$ 0.5

False omission rate $\mathbb{P}[\hat{y} \neq y | \hat{y} = -1]$ 0.875

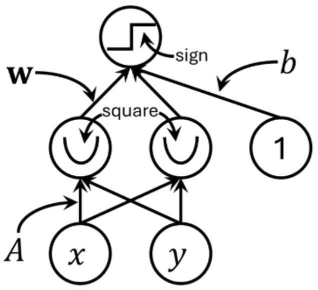
Melbu's classifier is not very good. Suggest a simple change to the model parameters so that the new classifier's accuracy goes up to at least 75% (maybe more). You are not allowed to (re)train on original or additional data or change the training algorithm. All you can do is make modifications directly to the model parameters $\mathbf{w}, b$ learnt by Melbu. Briefly justify your answer.

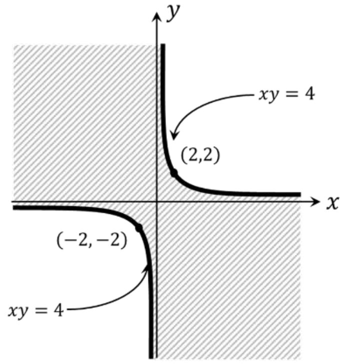
Set $\hat{\mathbf{w}} = -\mathbf{w}, \hat{b} = -b$ so that the predictions of the classifier are flipped. The new confusion matrix would look like the one given on the right and the new classifier would have 0.8 accuracy.

	$\hat{y} = 1$	$\hat{y} = -1$
$y = 1$	700	100
$y = -1$	100	100

In general, a model for a binary classification problem can always be massaged to give at least 0.5 accuracy. A model with 0.5 accuracy is actually useless – even random predictions assure that. More generally, for a C-class problem, an accuracy of $\frac{1}{C}$ is trivial – can you show this?

Q5 (Hyperbolic Networks) We wish to use a neural network with architecture shown on the left, to solve a binary classification problem shown on the right. The bold decision boundaries in the figure depict the hyperbola $xy = 4$. The NN has parameters a 2×2 matrix $A \in \mathbb{R}^{2\times2}$, a 2D vector $\mathbf{w} \in \mathbb{R}^2$ and a bias $b \in \mathbb{R}$. The output of the NN is $\text{sign}(\mathbf{w}^\top\phi(\mathbf{x}) + b)$ where $\phi(\mathbf{x}) = (A\mathbf{x})^2$ with square activation being applied coordinate-wise i.e. for a vector $\mathbf{v} = (v_1, v_2)$, the square activation is applied as $(\mathbf{v})^2 \overset{\text{def}}{=} (v_1^2, v_2^2)$. Find values of the parameters $A, \mathbf{w}, b$ so that the NN gives output $+1$ in the shaded region and -1 in the white region. Be careful to write value of A and not $A^\top$ (note that we have $\phi(\mathbf{x}) = (A\mathbf{x})^2$ not $(A^\top\mathbf{x})^2$). **Write only in the space provided.** **(7 marks)**





$$A = \begin{bmatrix} \boxed{1} & \boxed{1} \\ \boxed{1} & \boxed{-1} \end{bmatrix} \qquad \mathbf{w} = \begin{bmatrix} \boxed{-\frac{1}{4}} & \boxed{\frac{1}{4}} \end{bmatrix} \qquad b = \boxed{4}$$

Desired classifier is $\text{sign}(4 - xy)$. Multiple solutions are possible that exploit the fact that $xy = \frac{1}{4}((x+y)^2 - (x-y)^2)$. Another solution is $A = \frac{1}{2} \cdot \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}, \mathbf{w} = (-1,1), b = 4$