Q1. Perceptron and Kernels

A kernel is a mapping $K(x, y)$ from pairs vectors in $\mathbb{R}^d$ into the real numbers such that $K(x, y) = \Phi(x) \cdot \Phi(y)$ where $\Phi$ is a mapping from $\mathbb{R}^d$ into $\mathbb{R}^D$ where $D$ is possibly different from $d$ and even infinite. We say that a mapping $K(x, y)$ for which such $\Phi$ exists is a valid kernel.

(a) The following binary class data has two features, $A$ and $B$.

<table>
<thead>
<tr>
<th>Index</th>
<th>A</th>
<th>B</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2.</td>
<td>0</td>
<td>3</td>
<td>-1</td>
</tr>
<tr>
<td>3.</td>
<td>1</td>
<td>-1</td>
<td>1</td>
</tr>
<tr>
<td>4.</td>
<td>3</td>
<td>0</td>
<td>-1</td>
</tr>
<tr>
<td>5.</td>
<td>-1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>6.</td>
<td>0</td>
<td>-3</td>
<td>-1</td>
</tr>
<tr>
<td>7.</td>
<td>-1</td>
<td>-1</td>
<td>1</td>
</tr>
<tr>
<td>8.</td>
<td>-3</td>
<td>0</td>
<td>-1</td>
</tr>
</tbody>
</table>

(i) Select all true statements:

- This data is linearly separable.
- This data is linearly separable if we use a feature map $\phi((A, B)) = (A^2, B^2, 1)$.
- There exists a kernel such that this data is linearly separable.
- For all datasets in which no data point is labeled in more than one distinct way, there exists a kernel such that the data is linearly separable.
- For all datasets, there exists a kernel such that the data is linearly separable.
- For all valid kernels, there exists a dataset with at least one point from each class that is linearly separable under that kernel.
- None of the above.

We will be running both the primal (normal) binary (not multiclass) perceptron and dual binary perceptron algorithms on this dataset. We will initialize the weight vector $w$ to $(1, 1)$ for the primal perceptron algorithm. Accordingly, we will initialize the $a$ vector to $(1, 0, 0, 0, 0, 0, 0, 0)$ for the dual perceptron algorithm with the kernel $K(x, y) = x \cdot y$. Pass through the data using the indexing order provided. There is no bias term.

Write your answer in the box provided. Show your work outside of the boxes to have a chance at receiving partial credit.

(ii) What is the first misclassified point?

Point 2.

(iii) For the primal perceptron algorithm, what is the weight vector after the first weight update?

The weight vector after the first weight update will be:

$$w = (1, 1) - (0, 3) = (1, -2)$$ (1)
For your convenience, the data is duplicated on this page.

<table>
<thead>
<tr>
<th>Index</th>
<th>A</th>
<th>B</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2.</td>
<td>0</td>
<td>3</td>
<td>-1</td>
</tr>
<tr>
<td>3.</td>
<td>1</td>
<td>-1</td>
<td>1</td>
</tr>
<tr>
<td>4.</td>
<td>3</td>
<td>0</td>
<td>-1</td>
</tr>
<tr>
<td>5.</td>
<td>-1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>6.</td>
<td>0</td>
<td>-3</td>
<td>-1</td>
</tr>
<tr>
<td>7.</td>
<td>-1</td>
<td>-1</td>
<td>1</td>
</tr>
<tr>
<td>8.</td>
<td>-3</td>
<td>0</td>
<td>-1</td>
</tr>
</tbody>
</table>

(iv) For the dual perceptron algorithm, what is the \( \alpha \) vector after the first weight update?

\[
\alpha = (1, -1, 0, 0, 0, 0, 0, 0)
\]  

(2)

(v) What is the second misclassified point?

Point 4.

(vi) For the primal perceptron algorithm, what is the weight vector after the second weight update?

\[
w = (1, -2) - (3, 0) = (-2, -2)
\]  

(3)

(vii) For the dual perceptron algorithm, what is the \( \alpha \) vector after the second weight update?

\[
\alpha = (1, -1, 0, -1, 0, 0, 0, 0)
\]  

(4)

(b) Consider the following kernel function: \( K(x, y) = (x \cdot y)^2 \) where \( x, y \in \mathbb{R}^2 \). Find a valid \( \Phi \) map for this kernel. That is, find a vector-to-vector function \( \phi \) such that \( \phi(x) \cdot \phi(y) = K(x, y) = (x \cdot y)^2 \). Show work to have a chance at receiving partial credit. Any precise answer format is acceptable.

Expanding \((x \cdot y)^2 = (x_1 y_1 + x_2 y_2)^2 = x_1^2 y_1^2 + 2x_1 y_1 x_2 y_2 + x_2^2 y_2^2\) so the mapping

\[
\Phi(x) = [x_1^2, \sqrt{2} x_1 x_2, x_2^2]
\]  

is valid.

(c) We have \( n \) data points, \( \{(x_i, y_i)\}_{i=1}^n \), with \( x_i \in \mathbb{R}^d \) and \( y_i \in \{1, 2, \ldots, M\} \). That is, they are labelled as belonging to one of \( M \) classes. We will run the multiclass perceptron algorithm with an RBF kernel:
\[ K(x_i, x_j) = \exp(-\|x_i - x_j\|^2) \]  

(5)

Denote the dual weights at time \( t \) as \( \alpha_y^{(t)} = (\alpha_{y,1}^{(t)}, \ldots, \alpha_{y,K}^{(t)}) \) for all classes \( y = 1, \ldots, M \).

(i) What is the right value for \( K \), the dimension of each of the dual weight vectors?

- \( n \)
- \( M \)
- \( M + n \)

(ii) Assume that for some \( t \), and for all \( y \), \( \alpha_{y}^{(t)} \) has only one nonzero entry. This single nonzero entry equals one. All the nonzero entries occur at different indices for different \( y \). Describe the decision regions in \( \mathbb{R}^d \) for the \( M \) classes in terms of distances between points.

The nonzero entries of \( \alpha_{y}^{(t)} \) correspond to \( M \) points in the training data. Call these points the class centers. Each of these will also correspond to some class. Not necessarily the training point class.

Any new query point \( x \in \mathbb{R}^d \) will be labeled as the class corresponding to the closest class center.
Q2. Decision Trees

You are given a dataset for training a decision tree. The goal is to predict the label (+ or -) given the features A, B, and C.

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>label</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>+</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>+</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>+</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>-</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>-</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>+</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>-</td>
</tr>
</tbody>
</table>

First, consider building a decision tree by greedily splitting according to information gain.

(a) Which features could be at the root of the resulting tree? Select all possible answers.

- [ ] A
- [ ] B
- [x] C

A and C yield maximal information gain at the root.

(b) How many edges are there in the longest path of the resulting tree? Select all possible answers.

- [ ] 1
- [ ] 2
- [x] 3
- [ ] 4
- [ ] None of the above

Regardless of the choice of the feature at the root, the resulting tree needs to consider all 3 features in a path, so there are 3 edges in that path.

Now, consider building a decision tree with the smallest possible height.

(c) Which features could be at the root of the resulting tree? Select all possible answers.

- [ ] A
- [x] B
- [ ] C

The optimal decision tree first splits on B. For the B=0 branch, the next split is on A; for the B=1 branch, the next split is on C.

(d) How many edges are there in the longest path of the resulting tree? Select all possible answers.

- [ ] 1
- [x] 2
- [ ] 3
- [ ] 4
- [ ] None of the above

As can be seen from the answer to part (c), the optimal tree has two edges per path from the root to any leaf.