To earn the extra credit, one of the following has to hold true. Please circle and sign.

A I spent 3 or more hours on the practice final.

B I spent fewer than 3 hours on the practice final, but I believe I have solved all the questions.

Signature: ________________________________

The normal instructions for the exam follow below:

- You have 3 hours.
- The exam is closed book, closed notes except for two double-sided cheat sheets.
- Non-programmable calculators only.
- Mark your answers ON THE EXAM ITSELF. If you are not sure of your answer you may wish to provide a brief explanation.

<table>
<thead>
<tr>
<th>First name</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Last name</td>
<td></td>
</tr>
<tr>
<td>SID</td>
<td></td>
</tr>
<tr>
<td>Login</td>
<td></td>
</tr>
</tbody>
</table>

For staff use only:

<table>
<thead>
<tr>
<th>Q1. Classification and Separating Hyperplanes</th>
<th>/11</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total</td>
<td>/11</td>
</tr>
</tbody>
</table>
Q1. [11 pts] Classification and Separating Hyperplanes

For this first part, we will be deciding what makes a good feature-mapping for different datasets, as well as finding feature weights that make the data separable.

![Figure 1: Sets of points separated into positive examples (x’s) and negative examples (o’s). In plot (a), the dotted line is given by \( f(x_1) = x_1^3 - x_1 \).](image)

We begin with a series of true/false questions on what kernels can separate the datasets given. We always assume a point \( x \) is represented without a bias term, so that \( x = [x_1 \ x_2] ^T \). We will consider the following four kernels:

A. The linear kernel \( K_{\text{lin}}(x,z) = x^T z = x \cdot z \).

B. The shifted linear kernel \( K_{\text{bias}}(x,z) = 1 + x^T z = 1 + x \cdot z \).

C. The quadratic kernel \( K_{\text{quad}}(x,z) = (1 + x^T z)^2 = (1 + x \cdot z)^2 \).

D. The cubic kernel \( K_{\text{cub}}(x,z) = (1 + x^T z)^3 = (1 + x \cdot z)^3 \).

(a) (i) \[true or false\] The kernel \( K_{\text{lin}} \) can separate the dataset in Fig. 1(b).

(ii) \[true or false\] The kernel \( K_{\text{bias}} \) can separate the dataset in Fig. 1(b).

(iii) \[true or false\] The kernel \( K_{\text{cub}} \) can separate the dataset in Fig. 1(b).

(iv) \[true or false\] The kernel \( K_{\text{lin}} \) can separate the dataset in Fig. 1(a).

(v) \[true or false\] The kernel \( K_{\text{quad}} \) can separate the dataset in Fig. 1(a).

(vi) \[true or false\] The kernel \( K_{\text{cub}} \) can separate the dataset in Fig. 1(a).

(b) [2 pts] Now imagine that instead of simply using \( x \in \mathbb{R}^2 \) as input to our learning algorithm, we use a feature mapping \( \phi : x \mapsto \phi(x) \in \mathbb{R}^k \), where \( k \gg 2 \), so that we can learn more powerful classifiers. Specifically, suppose that we use the feature mapping

\[
\phi(x) = [1 \ x_1 \ x_2 \ x_1^2 \ x_2^2 \ x_1^3 \ x_2^3]^T
\]  

so that \( \phi(x) \in \mathbb{R}^7 \). Give a weight vector \( w \) that separates the x points from the o points in Fig. 1(a), that is, \( w^T \phi(x) = w \cdot \phi(x) \) should be > 0 for x points and < 0 for o points.
(c) [1 pt] Using the feature mapping (1), give a weight vector $w$ that separates the $x$ points from the $o$ points in Fig. 1(b), assuming that the line given by $f(x_1) = ax_1 + b$ lies completely between the two sets of points.

Now it’s time to test your understanding of training error, test error, and the number of samples required to learn a classifier. Imagine you are learning a linear classifier of the form $\text{sign}(w^T \phi(x))$, as in the binary Perceptron or SVM, and you are trying to decide how many features to use in your feature mapping $\phi(x)$.

![Figure 2](image)

**Figure 2:** Number of samples required to learn a linear classifier $\text{sign}(w^T \phi(x))$ as a function of the number of features used in the feature mapping $\phi(x)$.

(d) [1 pt] Which of the plots (a), (b), and (c) in Fig. 2 is most likely to reflect the number of samples necessary to learn a classifier with good generalization properties as a function of the number of features used in $\phi(x)$?

![Figure 3](image)

**Figure 3:** Leftmost plot: training error of your classifier as a function of the number of features.

(e) [1 pt] You notice in training your classifier that the training error rate you achieve, as a function of the number of features, looks like the left-most plot in Fig. 3. Which of the plots (a), (b), or (c) in Fig. 3 is most likely to reflect the error rate of your classifier on a held-out validation set (as a function of the number of features)?