1. In this question we consider the primal perceptron and dual perceptron algorithms and how they work given a concrete dataset and kernel. For the algorithms please refer to the pseudocode in programming project 4. In particular consider the dataset below where we have 3 features and 2 examples, and consider the kernel \( k(x,y) = (x \cdot y)^2 \).

\[
\begin{array}{ccc}
0.33 & 0.78 & 1.02 \quad (-) \\
0.44 & 1.25 & 0.88 \quad (+)
\end{array}
\]

(a) Note that because the algorithms initialize their weight vector to 0 they will predict (+) on the first example and therefore make a mistake. What is the hypothesis of the Primal Perceptron and Dual perceptron (vector \( w \) or \( \alpha \) respectively) after this first mistake? Please show any computation and explain your answer.

(b) Now use the answer to the previous part to show the prediction of the algorithms on the second example, and if the algorithm makes a mistake show its hypothesis after the corresponding update.

(c) For the dual perceptron show the expanded feature vector corresponding to the kernel and repeat the previous two parts, showing the results in terms of the sector \( w \) in the expanded space (you should get the same sums and predictions).

2. Recall that in class we have shown that \( k(x,y) = (x \cdot y + 1)^2 \) is a kernel. Now consider a new function \( f(x,y) = 4k(x,y) + 25 \). Prove that \( f() \) is a kernel by showing that it is an inner product, i.e., \( f(x,y) = \phi(x) \cdot \phi(y) \) for some \( \phi() \).

Note that you do not need to repeat the proof from class, but to use the fact that \( k(x,y) \) is an inner product for some space to prove the same for \( f(x,y) \).

3. In this question we consider the Naive Bayes algorithm and active learning. Consider the dataset below where the first 6 examples are labeled and the last 4 are not. Consider active learning with uncertainty sampling (where in the notation of the slides from lecture we pick the instance with the smallest \( p_1 \)).

\[
\begin{array}{ccc}
000 & + \\
011 & + \\
011 & - \\
110 & - \\
101 & + \\
100 & - \\
010 & \\
101 & \\
110 & \\
001 & \\
\end{array}
\]

(a) Show the hypothesis of Naive Bayes after learning from the 6 labelled examples.

(b) Which example will be picked by uncertainty sampling? explain your answer.
4. Consider a 3x3 Grid World domain in reinforcement learning, where there are two actions: Up (goes up with probability 0.5, down w.p. 0.2, right/left w.p. 0.15) and Right (goes right with probability 0.5, left w.p. 0.2, up/down w.p. 0.15). Assume that our $Q$ value estimates so far are as summarized in the following tables:

<table>
<thead>
<tr>
<th>States</th>
<th>$Q(s, \text{Up})$</th>
<th>$Q(s, \text{Right})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1, 2, 3</td>
<td>0.8, 1.5, 3.3</td>
<td>1.9, 1.3, 2.4</td>
</tr>
<tr>
<td>4, 5, 6</td>
<td>1.1, 0.6, 4.2</td>
<td>0.8, 0.2, 3.6</td>
</tr>
<tr>
<td>7, 8, 9</td>
<td>0.1, 2.9, 6.6</td>
<td>6.8, 2.6, 2.3</td>
</tr>
</tbody>
</table>

(a) Recall that $V(s) = \max_a Q(s, a)$ from which you can calculate the current value estimates of the states. Show the calculation of a Bellman update, as in the Value Iteration algorithm, applied to state 5. Please see algorithm description and related Equations 17.6 in [RN].

(b) Now consider the SARSA and Q-learning algorithms discussed in class whose update equations are $Q(s, a) = Q(s, a) + \alpha (r + Q(s', a') - Q(s, a))$ and $Q(s, a) = Q(s, a) + \alpha (r + [\max_{a'} Q(s', a')] - Q(s, a))$ respectively where SARSA updates according to the action followed in $s'$ but Q-learning updates for the best action in $s'$. Please see algorithm description and related Equations 21.8 and 21.9 in [RN]. Now consider the following trajectory where we have specified states, actions, and rewards, as well as the transitions that actually occur:

8, Right, $r=0$ --> 9, Up, $r=0.5$ --> 8, Up, $r=0.3$ --> 5, Up, $r=1.1$ --> 6, Up

Calculate and show the 4 value updates that are done in each algorithm on this trajectory.