Good Luck

NAME______________________________

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>/20</td>
</tr>
<tr>
<td>2</td>
<td>/25</td>
</tr>
<tr>
<td>3</td>
<td>/25</td>
</tr>
<tr>
<td>4</td>
<td>/15</td>
</tr>
<tr>
<td>5</td>
<td>/15</td>
</tr>
<tr>
<td>TOTAL</td>
<td>/100</td>
</tr>
</tbody>
</table>
1) (20) Give concise answers to the following questions.

a) Consider a binary classification problem with \( m \) points in \( \mathcal{R}^2 \). We use a quadratic threshold classifier to separate points. Let \( m = 5 \). Find the growth number \( \Pi(5) \). Repeat for \( m = 6, m = 7, \) and \( m = 8 \). What is the VC dimension for quadratic threshold classifiers in \( \mathcal{R}^2 \).

b) Discuss why the Adaboost algorithm generalization test error continues to get smaller as the algorithm runs even after the training error has been reduced to zero.
c) Let \( K \) be the kernel matrix formed from inputs \( \phi(x(i)) \in \mathcal{R}^d, 1 \leq i \leq m \) where \( K(i,j) = \phi(x(i))^T \phi(x(j)) \). Show that the diagonals of \( K \) are all \( \geq 0 \).

d) Consider MCMC simulations using the Metropolis-Hastings algorithm. Show that this algorithm satisfied the detailed balance equations.
2) (25) Let $x(1) = (1, -1, -1)^T, d(1) = -1$, $x(2) = (-3, 1, 1)^T, d(2) = 1$, $x(3) = (-3, 1, -1)^T, d(3) = -1$, $x(4) = (1, 2, 1)^T, d(4) = -1$, and $x(5) = (-1, -1, 2)^T, d(5) = 1$ be five binary labeled training examples. These points are linearly separable.

a) Implement the Perceptron Learning algorithm showing the steps until a solution is reached.

b) Find the optimum margin classifier (support vectors, weights and threshold value) and the margin.

c) Find the Fisher Linear Discriminant solution (weights and threshold value). Hint: you can solve the problem using the Least Square SVM with positive points labeled as $m/m_1$ and negative points labeled as $-m/m_2$ where $m_1$ is the number of positively labeled points and $m_2$ is the number of negatively labeled points.
Consider the following regression problem where we try to approximate a noisy sinc function using Gaussian kernels. We observe 100 training samples of the noisy sinc function with
\[ d(k) = \text{sinc}(x(k)) + n(k) \]
where each input \( x(k) \) is drawn uniformly from \([-5, 5]\). All inputs are independent and identically distributed (iid). \( n(k) \) is zero mean Gaussian noise with variance 0.3. \( n(k) \) are iid and independent of all inputs.

Use the least squares support vector machine (LS SVM) to find an estimate \( y(x) = \sum_{j=1}^{100} \alpha(j)K(x, x(j)) + b \). Determine the \( \alpha(j) \) and threshold value \( b \) that minimizes
\[ J(w, b) = 0.5||w||^2 + 0.5C||e||^2 \]
where \( w \) is the associated weight vector with \( w = K\alpha \), \( e = d - y \) is the error vector for the 100 training examples, and \( C > 0 \) is the regularization parameter. \( K \) is the kernel matrix for the 100 training examples with \( K(x, z) = \exp(-||x - z||^2/(2\sigma^2)) \).

For a given \( C \) and \( \sigma \) find the LS SVM solution. Test this solution on 200 test examples with the target being \( d_{test}(x) = \text{sinc}(x) \) where \( x \) is drawn uniformly from \([-5, 5]\). Repeat this simulation 100 times to get the average mean squared error for given \( C \) and \( \sigma \).

Repeat simulations for different values of \( C \) and \( \sigma \) reporting on which values give smallest mean squared error.
Consider the following estimation problem. Let $D$ be a Gaussian random variable with mean $\mu$ and variance $\sigma^2_D$. We are given $m$ observations; $X(1), \ldots X(m)$. The $X(i), 1 \leq i \leq m$ are conditionally independent given $D = d$ and identically distributed Gaussian random variables with conditional mean $d$ and conditional variance $\sigma^2_X$.

Consider the posterior pdf $f_{D|X_1, \ldots, X_m}(d|x(1), \ldots, x(m))$. Find the posterior pdf, find the minimum mean squared error estimate of $D$ given the $m$ observations, and the resulting minimum mean squared error.
5) (15) Consider a single output unit with $n$ inputs and nonlinear activation function described by

$$y = \tanh(\beta s), \quad \beta > 0$$

where $s = w^T x$ with $x \in \mathbb{R}^n$ being the input. The weights are given by $w \in \mathbb{R}^n$.

a) Formulate the LMS learning rule for this nonlinear unit.

b) Consider an entropy based energy function described by

$$J(w) = \frac{1}{2} \sum_{k=1}^{m} \left[ (1 + d[k]) \log \left( \frac{1 + d[k]}{1 + y[k]} \right) + (1 - d[k]) \log \left( \frac{1 - d[k]}{1 - y[k]} \right) \right].$$

Formulate a learning rule based on the gradient estimate of this energy function. How does this algorithm compare to the LMS algorithm?