**Mid-term**
EECS 545: Machine Learning
Winter, 2018

**Name**:

**UM uniqname**:

- **Closed books. One sheet of paper of notes is allowed. No computers or calculators.**
  - Showing your work makes partial credit possible.
    If you write nothing at all, it's hard to justify any score but zero.
  - Feel free to use the backs of the sheets for scratch paper.
  - Write clearly. If we can't read your writing, it will be marked wrong.
- This course operates under the rules of the College of Engineering Honor Code. Your signature endorses the pledge below. **After** you finish your exam, please sign below:
  *I have neither given nor received aid on this examination, nor have I concealed any violations of the Honor Code.*

DO NOT WRITE BELOW THIS LINE

| **Problem** | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | Total |
|---|---|---|---|---|---|---|---|---|---|
| **Points** | | | | | | | | | |
| **Max Points** | 24 | 6 | 12 | 6 | 6 | 6 | 6 | 6 | 72 |

**Question 1 [24pts]** YOU MUST FILL OUT THE SCANTRON SHEET FOR THIS QUESTION (No need for explanations here unless you feel the question is ambiguous and want to justify your answer).

1. In general the error on the training set is a better estimate of the generalization error than the error on the test set.

   FALSE

2. The perceptron algorithm finds the maximum margin classifier if the data is linearly separable.

   FALSE

3. Locally-weighted linear regression can produce nonlinear fits to the data.

   TRUE

4. In nearest neighbor regression, using more neighbors generally leads to more complex functions than using fewer neighbors.

   FALSE

5. In polynomial regression with a fixed data set, as one increases the degree of the polynomial used, the expected mean-squared error on the test set strictly decreases.

   FALSE

6. Linear decision boundaries for classification are optimal (minimum misclassification error on training set) only if the underlying data is truly linearly separable?

   FALSE

7. Quadratic discriminant analysis as an approach to classification cannot be **applied** if the true class-conditional density for each class is *not* Gaussian.

   FALSE

8. Logistic Regression is a method for doing classification problems.

   TRUE

9. In Classification, for data D and a class/hypothesis C, which of the following best describes $\sum_c P(C = c|D = d) = 1$

   ALWAYS TRUE

10. In Classification, for data D and a class/hypothesis C, which of the following best describes $\sum_c P(D = d|C = c) = 1$

    ALWAYS FALSE

11. In Classification, choose the most specific relation that holds between the following equations (i.e., replace the "?" by one of the given choices). Note that "depends" is the least specific relation. Assume all probabilities involved are non-zero.
    $P(C = c|D = d)$ ? $P(C = c)$

    "depends"

12. In Classification, choose the most specific relation that holds between the following equations (i.e., replace the "?" by one of the given choices). Note that "depends" is the least specific relation. Assume all probabilities involved are non-zero.
    $P(C = c|D = d)$ ? $P(D = d|C = c)P(C = c)$

    $\geq$

**Question 2 [6pts]** (Regression) Suppose we have a training set $\{(x_i, t_i); i = 1, \cdots, N\}$ of $N$ independent examples, but in which the $t_i$'s were observed with different variances. Specifically, suppose that

$$p(t_i|x_i; w) = \frac{1}{\sqrt{2\pi}\sigma_i} \exp\Big( - \frac{(t_i - w^T x_i)^2}{2(\sigma_i)^2}\Big).$$

In words, $t_i$ has mean $w^T x_i$ and variance $(\sigma_i)^2$ (where the $\sigma_i$'s are fixed, **known** constants). **Show/Prove** that finding the maximum likelihood estimate of $w$ reduces to solving a weighted linear regression problem. State clearly what the weights ($r_i$'s) are in terms of the $\sigma_i$'s.

The log-likelihood is given by

$$\sum_{i=1}^{N} \log\, p(t_i|x_i; w) = -\sum_{i=1}^{N} \log\, (\sqrt{2\pi}\sigma_i) - \sum_{i=1}^{N} \frac{(t_i - w^T x_i)^2}{2\sigma_i^2}$$

The maximum-likelihood parameter estimates are given by

$$\text{argmax}_w \sum_{i=1}^{N} \log\, p(t_i|x_i; w) = \text{argmin}_w \sum_{i=1}^{N} \frac{(t_i - w^T x_i)^2}{2\sigma_i^2} = \text{argmin}_w \sum_{i=1}^{N} \frac{1}{2\sigma_i^2}(w^T x_i - t_i)^2$$

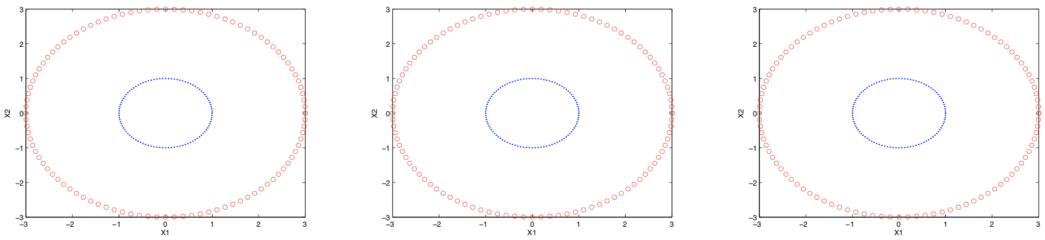
which is a weighted linear regression problem where the linear model is given by $y(x) = w^T x$ and $\frac{1}{2\sigma_i^2}$ represent the weights.

**Question 3 [12pts]** In class we learnt that SVM can be used to classify linearly inseparable data by transforming it to a higher dimensional space with a kernel $K(x,z) = \phi(x)^T\phi(z)$, where $\phi(x)$ is a feature mapping. Let $K_1$ and $K_2$ be $R^n \times R^n$ kernels, $K_3$ be a $R^d \times R^d$ kernel and $c \in R^+$ be a positive constant. $\phi_1 : R^n \to R^d$, $\phi_2 : R^n \to R^d$, and $\phi_3 : R^d \to R^d$ are feature mappings of $K_1$, $K_2$ and $K_3$ respectively. **Explain** how to use $\phi_1$ and $\phi_2$ to obtain the following kernels.

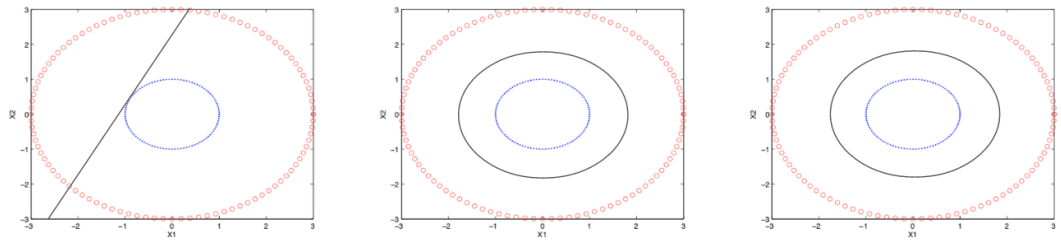
**a** (2 pts) $K(x,z) = cK_1(x,z)$ (2pt; -1 if explicit $\phi(x)$ is missing) Ans: $\phi(x) = \sqrt{c}\phi_1(x)$

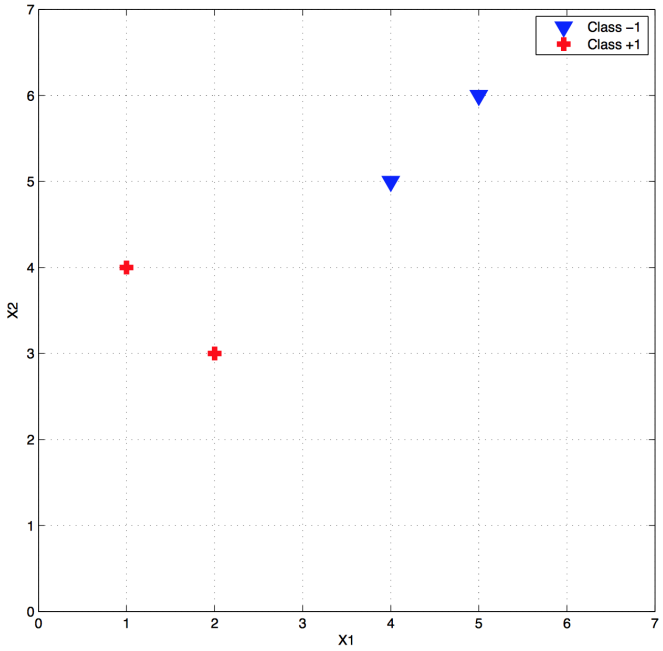
**b** (3 pts) $K(x,z) = K_1(x,z)K_2(x,z)$

(3pt; -2 if explicit $\phi(x)$ is missing) Ans: $\phi(x)$ will be a vector of all elements in the outer product given by $\phi_1(x)\phi_2(x)'$. Lose all points if you answered $\phi_1(x)\phi_2(x)$ with explanation of your symbols, since it's a valid matrix multiplication.



**c** (3 pts) You are given the above 3 identical plots, which illustrates a dataset with two classes (the inner ring of data is one class, the outer ring of data is the other class). Both classes have equal number of instances in the data set. Draw the decision boundary when you train an SVM classifier with linear, polynomial (order 2) and RBF kernels respectively in the plots going from left to right.

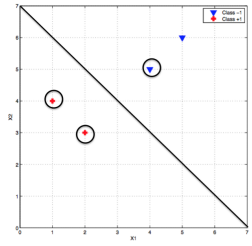





**d** (4pts) Support vector machines learn a decision boundary leading to the largest margin from both classes. You are training SVM (with no slack) on a tiny dataset with 4 points shown in Figure above. This dataset consists of two examples with class label $-1$ (denoted with plus), and two examples with class label $+1$ (denoted with triangles).

Circle the support vectors and draw the approximate decision boundary in Figure above.

Figure 1:



**Question 4 [6pts]** For a classification problem with $K$ classes, suppose our goal is to make as few misclassifications as possible. Let the input $x$ be continuous rather than discrete (the dimensionality of $x$ is not important here). We need a rule that assigns each value of $x$ to one of the classes. Such a rule would divide the input space into decision regions $R_k$ such that all points in $R_k$ are assigned to class $C_k$.

a Write down the expression for the probability that the rule is correct.

$$Pr(correct) = \sum_{k=1}^{K} \int_{R_k} p(x, C_k) dx$$

b What rule would minimize misclassification probability, i.e., maximize probability of being correct in part 'a' above. Justify/Prove your answer.

$Pr(correct)$ is maximized when the regions $R_k$ are chosen such that each x is assigned to the class for which $p(x, C_k)$ is largest. Because $p(x, C_k) = p(C_k|x)p(x)$ and $p(x)$ is common to all terms, each x should be assigned to the class having the largest posterior probability.

**Question 5 [6 pts]** (Geometry of Margin Calculation)

In our discussion of max-margin classifiers, we consider a linear discriminant function of the form:

$$y(\mathbf{x}) = \mathbf{w}^T\phi(\mathbf{x}) + b$$

The decision boundary is the hyperplane specified by $y(\mathbf{x}) = 0$. In class we stated that the signed perpendicular distance of $\phi(\mathbf{x})$ from the decision boundary is given by

$$\frac{t(\mathbf{w}^T\phi(\mathbf{x}) + b)}{||\mathbf{w}||}.$$

**Prove** this result here.

Consider a signed measurement of the perpendicular distance r of $\phi(\mathbf{x})$ from the decision surface: r has the same sign with $y(\mathbf{x})$, and its absolute value equals the perpendicular distance. Let $\phi(\mathbf{x})$ be an arbitrary point and $\phi(\mathbf{x})_\perp$ be $\phi(\mathbf{x})$'s orthogonal projection onto the decision surface, so that

$$\phi(\mathbf{x}) = \phi(\mathbf{x})_\perp + r\frac{\mathbf{w}}{||\mathbf{w}||}$$

then we have:

$$\mathbf{w}^T\phi(\mathbf{x}) + b = \mathbf{w}^T\phi(\mathbf{x})_\perp + b + r\frac{\mathbf{w}^T\mathbf{w}}{||\mathbf{w}||}$$

Since $\phi(\mathbf{x}_\perp)$ is on the decision surface, then $\mathbf{w}^T\phi(\mathbf{x})_\perp + b = 0$. We have:

$$r = \frac{\mathbf{w}^T\phi(\mathbf{x}) + b}{||\mathbf{w}||}$$

If we assume data set is linearly separable in feature space by $\mathbf{w}$ and $b$, so that the function $y(\mathbf{x}) = \mathbf{w}^T\phi(\mathbf{x}) + b$ satisfies $y(\mathbf{x}_n) > 0$ for points having $t_n = +1$, and $y(\mathbf{x}_n) < 0$ for points having $t_n = -1$. Thus the perpendicular distance is given by

$$\frac{t(\mathbf{w}^T\phi(\mathbf{x}) + b)}{||\mathbf{w}||}$$

**Question 6 [6 pts]** The following questions require a yes/no accompanied by one sentence of explanation, or a reasonably short answer (usually at most a few sentences). To discourage random guessing, no credit will be given for answers without a correct explanation.

Suppose you are using logistic regression (trained with maximum likelihood), and the classifier is performing poorly (has unacceptably high error) on the test set but performing well on the training set.

Determine whether each of the following is a reasonable step to take to try to get it up to an acceptable level of performance? (Yes/No) In either case, give a one-sentence justification.

**(i)** Increase the regularization parameter.

Yes. Because the symptom is that of overfitting and increasing regularization parameter can decrease overfitting.

**(ii)** Decrease the regularization parameter.

No. Decreasing the regularization parameter could cause more overfitting.

**(iii)** Use Newton's method instead of gradient descent.

No. Using Newton's method is to fix optimization, which is not directly related to fixing overfitting. It is possible that gradient descent may not have converged, so Newton's method may provide a solution that better optimizes for the training data (note that logistic regression is a convex problem, and you will obtain the same solution with both gradient descent or Newton'5s method after convergence); however, this may cause even more overfitting to the training data.

**(iv)** Use feature selection to reduce the number of features.

Yes. By selecting informative features, the redundant or irrelevant features (that may result in overfitting) will be removed from the feature set.

**(v)** Get more training examples. In this case, please explain (1) whether the training error will likely to increase or decrease; and (2) whether the test error will likely to increase or decrease. (For simplicity, we assume that both the training and testing examples were randomly sampled from an IID distribution.)

Yes. By adding more training examples, the training error will likely to increase, because it increases the difficulty of fitting (i.e., to more diverse patterns). In contrast, the testing error will likely to decrease, since the model has seen more training data and may cover the data space more densely and thus generalize better.

**Question 7 [6 pts]**

**(i)** (Cross-Validation) Imagine you have $K$ different regression-models with different complexities (e.g., first-order polynomial, second-order polynomial with squared-error loss functions) and you have a large data set $D$. Describe the cross-validation procedure to select among the regression-models and describe what final score you would report as your final result.

1pt for the partition of data set D : you need have a test set untouched by the training and validation process.
1pt for correct cross-validation procedure. You can divide the data set into training, validation and test set, or use L-folded validation, and select the regression model with lowest validation ratio.
1pt for providing the final score : after model selection, use both the training set and validation set to train the selected regression model, and report the test accuracy on testing set.

**(ii)** (MLE) Suppose repeatedly tossing a coin with unknown probability of heads $\mu$ produces the data sequence $HHTTTTHTHT$. What is the maximum likelihood estimate of $\mu$? Show your derivation.

The outcome of a coin toss is distributed as $t \sim \text{Ber}(t; \mu)$.

Our data consists of 10 coin tosses $\mathcal{D} = \{t_i\}_{i=1}^{10}$

The data log-likelihood is given by

$$\begin{aligned}
\log\, P(\mathcal{D}) &= \sum_{i=1}^{10} \log\, \text{Ber}(t_i|\mu) \\
&= \sum_{i=1}^{10} \log\, \mu * I[t_i = H] + \log\, (1-\mu) * I[t_i = T] \\
&= \log\, \mu \sum_{i=1}^{10} I[t_i = H] + \log\, (1-\mu) \sum_{i=1}^{10} I[t_i = T] \\
&= 4\log\, \mu + 6\log\, (1-\mu)
\end{aligned}$$

$$\frac{d\log\, P(\mathcal{D})}{d\mu} = \frac{4}{\mu} - \frac{6}{1-\mu} = \frac{4-10\mu}{\mu(1-\mu)} = 0 \Rightarrow \hat{\mu} = 0.4$$

$$\frac{d^2\log\, P(\mathcal{D})}{d\mu^2} = -\frac{4}{\mu^2} - \frac{6}{(1-\mu)^2} < 0 \text{ for } \mu \in (0,1)$$

Thus, $\mu_{MLE} = 0.4$

**Question 8 [6 pts]** Consider using a logistic regression model $p(y = 1|x; w) = g(w^T x)$ where $g$ is the sigmoid function, and let a training set $\{(x^i, y^i); i = 1, ..., N\}$ be given as usual. The maximum likelihood estimate of the parameters $w$ is given by

$$w_{ML} = \arg\max_w \Pi_{i=1}^N p(y^i|x^i; w)$$

If we wanted to regularize logistic regression, then we might put a prior on the parameters. Suppose we chose the prior $w \sim \mathcal{N}(0, \tau^2 I)$ (here, $\tau > 0$, and $I$ is the identity matrix of the appropriate size), and then found the MAP estimate as

$$w_{MAP} = \arg\max_w \Pi_{i=1}^N p(y^i|x^i; w)p(w).$$

Prove that $||w_{MAP}||_2 \leq ||w_{ML}||_2$.

By contradiction, assume that

$$\|\mathbf{w}_{MAP}\|_2 > \|\mathbf{w}_{ML}\|$$

Then, we have that

$$\begin{aligned} p(\mathbf{w}_{MAP}) &= \frac{1}{(2\pi)^{\frac{n+1}{2}}|\sigma^2 I|^{\frac{1}{2}}} e^{(-\frac{1}{2\sigma^2}\|\mathbf{w}_{MAP}\|_2^2)} \\ &< \frac{1}{(2\pi)^{\frac{n+1}{2}}|\sigma^2 I|^{\frac{1}{2}}} e^{(-\frac{1}{2\sigma^2}\|\mathbf{w}_{ML}\|_2^2)} \\ &= p(\mathbf{w}_{ML}) \end{aligned}$$

This yields

$$\begin{aligned} p(\mathbf{w}_{MAP}) \prod_{i=1}^N p(y^{(i)}|\mathbf{x}^{(i)}, \mathbf{w}_{MAP}) &< p(\mathbf{w}_{ML}) \prod_{i=1}^N p(y^{(i)}|\mathbf{x}^{(i)}, \mathbf{w}_{MAP}) \\ &\leq p(\mathbf{w}_{ML}) \prod_{i=1}^N p(y^{(i)}|\mathbf{x}^{(i)}, \mathbf{w}_{ML}) \end{aligned}$$

where the last inequality holds since $\mathbf{w}_{ML}$ was chosen to maximize $\prod_{i=1}^N p(y^{(i)}|\mathbf{x}^{(i)}; \mathbf{w})$. However, this result gives us a contradiction, since $\mathbf{w}_{MAP}$ was chosen to maximize $\prod_{i=1}^N p(y^{(i)}|\mathbf{x}^{(i)}; \mathbf{w})p(\mathbf{w})$.