Section 1. Fill in the blanks

Instructions:

- Fill in the blanks in the questions with appropriate answers.
- Your answers must be in a **highly simplified** form. For e.g., if the correct answer is $\int_a^b x^2 \, dx \geq 0$, then you need to write the answer as $b \geq a$. Answers that are not simplified will NOT get any credit.
- To make the questions precise, the type of answer to be filled in the blank is clearly mentioned after the blank. For example, if a blank is followed by:
  - [[MathExpr]], then you need to write an answer that is a valid mathematical expression. For e.g., $3x - 4y + x^3$, (or) $\neq 0$, (or) $trace(M)$ etc.
  - [[Term]], then you need to write an answer that is an English phrase representing a well-defined object/concept or a well-known theorem name. For e.g., equilateral triangle (or) Spectral theorem etc.
  - [[T/F/M]], then the only choices for answer are 'T', 'F', and 'M'. While 'T', 'F' represents that the previous sentence is 'true', 'false' respectively; 'M' represents that the given information is in-sufficient to decide whether the previous sentence is true or false (basically 'M' handles the undecidable case). If [[T/F]] is given, then the only choices for answer are 'T', and 'F'.

Questions:

1. A function $k : \mathcal{Z} \times \mathcal{Z} \mapsto \mathbb{R}$ is defined as a kernel over $\mathcal{Z}$ iff every gram matrix induced by the function, k, is symmetric [[Term]] and positive-semi-definite [[Term]]. (1mark)

2. Given that tractability and computational effort are not of concern, finding the regularization hyper-parameter C in the SVM that minimizes cross-validation error over all possible values of C is a better model selection algorithm than finding the same among few (say, 10) possible values of C. F [[T/F]]. (1mark)

3. 1-NN classifier achieves an error on training set that is not greater than that with 5-NN classifier. T [[T/F/M]]. 1-NN classifier achieves a cross-validation error that is not greater than that with 5-NN classifier. M [[T/F/M]]. In the limit $m \to \infty$ (m is no. training examples), the true (but unknown) probability of misclaffication with 1-NN will be $\geq$ than that with 5-NN classifier. T [[T/F/M]]. (2marks)

4. Logistic loss, $l : \mathbb{R} \mapsto \mathbb{R}^+$, given by $l(z) \equiv$ $\log(1+e^{-z})$ $\forall\, z \in \mathbb{R}$ [[MathExpr]], is a convex function because $l''(z) = e^{-z}/(1+e^{-z})^2 > 0 \;\forall\, z \in \mathbb{R}$ [[MathExpr]]. (2marks)

5. Let $\mathcal{D} = \left\{ \left(\begin{bmatrix} 1 \\ 0 \end{bmatrix}, 1 \right), \left(\begin{bmatrix} -1 \\ 0 \end{bmatrix}, 1 \right), \left(\begin{bmatrix} 0 \\ 1 \end{bmatrix}, -1 \right), \left(\begin{bmatrix} 0 \\ -1 \end{bmatrix}, -1 \right), \right\}$ be the training dataset. The SVM formulation with polynomial kernel of degree three i.e., $k(x,z) = \left(1 + x^\top z\right)^3$ was solved and the dual variables for the training data points turned out to be $\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = \frac{1}{6}$. Then the prediction function of this SVM will be $g\left(\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \right) = \text{sign}(x_1^2 - x_2^2)$ [[MathExpr]]. (2.5marks)

6. Consider a set $\mathcal{Z} = \{z_1, z_2, z_3\}$ and a function $k : \mathcal{Z} \times \mathcal{Z} \mapsto \mathbb{R}$. One of the gram matrices induced by this function, k, is $\begin{bmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$. Then this function k is a valid kernel over $\mathcal{Z}$. T [[T/F/M]]. (2marks)

7. The function $k : \mathbb{R} \times \mathbb{R} \mapsto \mathbb{R}$ defined by $k(x,z) \equiv \cos(x - z)$ is a valid kernel because there exists a $\phi : \mathbb{R} \mapsto \mathbb{R}^2$ defined by $\phi(x) \equiv$ $[\cos(x) \quad \sin(x)]^\top$ [[MathExpr]] such that $k(x,z) = \phi(x)^\top \phi(z)$. (2marks)

Section 2. Analytical Questions

1. Let $\mathcal{D} = \{x_1, \ldots, x_m\} \subset \mathcal{X}$ be the training data and let $\phi : \mathcal{X} \mapsto \mathbb{R}^n$ be a feature map. There is a machine learning algorithm known as Principal Component Analysis (PCA), whose key steps are summarized below:

   (a) Form the correlation matrix, C, of the feature representations of the training points i.e., $C = \frac{1}{m}\sum_{i=1}^{m} \phi(x_i)\phi(x_i)^\top$.

   (b) Obtain the Eigen Value Decomposition (EVD) of C i.e., $C = VDV^\top$, where columns of V are eigen vectors of C and entries in the diagonal matrix D are the corresponding eigen values[1].

   (c) Construct a function $\psi : \mathcal{X} \mapsto \mathbb{R}^n$ defined by $\psi(x) \equiv V^\top \phi(x)$.

   Given that the primary goal in PCA is to construct the function ψ (as defined above), your job is to convince me that for constructing the function ψ it is enough to provide $\mathcal{D}$ and a kernel function $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ such that $k(x, z) = \phi(x)^\top \phi(z)\ \forall\ x, z \in \mathcal{X}$. Hence, the feature map ϕ is NOT explicitly needed. In short, your job is to kernelize PCA algorithm. However, you are not allowed to write an entirely free form answer. You need to answer by providing details of the following steps[2], which will kernelize PCA:

   (a) Firstly, prove that every eigen vector of C can be written as a linear combination of the feature representations of the training points. (2.5marks)

Note that trivial eigenvalues of C won't affect us.

Let v_j, d_j be the j^{th} eigenvector, value pair of C (non-trivial)

We have, $Cv_j = d_j v_j \Leftrightarrow \sum_{i=1}^{m} \varphi(x_i)\left(\frac{\varphi(x_i)^\top v_j}{m d_j}\right) = v_j$

$\Leftrightarrow v_j = \sum_{i=1}^{m} \alpha_{ij}\varphi(x_i) = X\alpha_j$,

where, $\alpha_{ij} = \frac{\varphi(x_i)^\top v_j}{m d_j}$, i^{th} column of X is $\varphi(x_i)$, i^{th} entry of α_j is α_{ij}.

[1] Given a matrix $M_{n\times n}$, a pair $v \in \mathbb{R}^n \ni \|v\| = 1$, $\alpha \in \mathbb{R}$ is called an eigen vector and eigen value pair iff $Mv = \alpha v$.

[2] Please use the spaces provided for writing your answers. You are free to attempt a later part of a question, even though you are not able to prove an earlier part. For e.g., you may attempt part (b) of this question by assuming the result to be proved in part (a) etc.

(b) Show that[3] all the coefficients of the linear combinations (leading to the eigen vectors of C) can themselves be obtained from EVD of some other matrix, which can be computed given $k, \mathcal{D}$ (the matrix does NOT explicitly involve ϕ). Clearly write down the matrix and describe how the coefficients shall be obtained from its EVD.

(5marks)

We have, $\frac{1}{m} X X^T (X\alpha_j) = d_j (X\alpha_j)$

$\Leftrightarrow \frac{K^2}{m} \alpha_j = d_j K \alpha_j$ ($\because$ premultiply by X^T, which is full-row rank)

$\Leftrightarrow K\alpha_j = (md_j)\alpha_j$

K is gram-matrix induced by k on D.

Also, $\|v_j\| = 1 \Rightarrow 1 = v_j^T v_j = \alpha_j^T K \alpha_j = md_j \|K_j\|^2 \Rightarrow \|\alpha_j\| = \frac{1}{\sqrt{md_j}}$.

Hence, the matrix whose EVD gives coeff. is K. And, the coeff. are the eigenvectors of K with norm $= \frac{1}{\sqrt{e_j}}$, $e_j = md_j$, the eigenvalue.

(c) Write down a formula for $\psi(x)$ involving these coefficients, kernel function and $\mathcal{D}$. Again, your answer must NOT explicitly involve ϕ.

(2.5marks)

$$\psi(x) = V^T \varphi(x) = \begin{bmatrix} v_1^T \varphi(x) \\ \vdots \\ v_m^T \varphi(x) \end{bmatrix}$$

$$= \begin{bmatrix} \sum_{i=1}^{m} \alpha_{i1} k(x_i, x) \\ \vdots \\ \sum_{i=1}^{m} \alpha_{im} k(x_i, x) \end{bmatrix}.$$

[3] For convenience sake, you may assume that the gram matrix induced by k on $\mathcal{D}$ is invertible.

2. The following is a proposal for a simple movie recommender system for a given user:

   (a) Show user a few pairs of movies he has seen and ask him to compare them according to his liking. This will give training data of the following form $\mathcal{D} = \{(x_{11}, x_{12}), \ldots, (x_{m1}, x_{m2})\}$, where a pair of movies (x_{j1}, x_{j2}) represents the fact that the user likes the movie x_{j1} more than the movie x_{j2}. Note that the the number of unique movies in $\mathcal{D}$ may be far less[4] than $2m$.

   (b) Using this training data, train an SVM-kind of model that gives a preference score for any movie, $x \in \mathcal{X}$. Needless to say, it makes sense only if the predicted scores are close to those implicitly hidden in the brain of the user after he watches that movie.

   (c) Every Thursday night send out an email to the user giving a ranked list of movies (as per the predicted scores) going to be released in that week.

   Ofcourse, as a ML researcher, you are least interested in steps 1 and 3. Your job is to provide details of step 2. To ease your job, I will specify a few things in the SVM-kind of model. Let $f : \mathcal{X} \mapsto \mathbb{R}$ represent the function that predicts the preference score. Since it is an SVM-kind of model, we restrict f to linear functions of the form: $f(x) = w^\top \phi(x)$, $x \in \mathcal{X}$, where $\phi : \mathcal{X} \mapsto \mathbb{R}^n$ is a feature map that captures key factors in a movie that may influence the user's preference: for e.g., $\phi_1(x)$ is the name of the lead-role actress, $\phi_2(x)$ is the name of the lead-role actor, $\phi_3(x)$ is the genre of the movie etc. Intuitively, then w_1 will represent how important the lead-role actress is for that movie being preferred by the user, and so on.

   Write down the final optimization problem that formalizes the training algorithm in the space below. Note that your formalism must ensure that you are employing a "sparse" loss like hinge-loss and you

[4] It is natural to imagine that there might be inconsistencies in his ranking. For example, if the user just received a good grade in CS725, then he might rank sci-fi movies higher that day, and rank comedy movies higher otherwise. There may be explicit inconsistencies too, in the sense that he might give a pair of movies with both orderings! Note that though these appear as inconsistencies, they actual might be very useful signals indicating that he has NO preference either way.

This is SAME as SVM with $\varphi(x_i) \to \delta(x_i)$, $b=0$

are indeed "maximizing margin". (2.5marks)

$$\min_{w\in\mathbb{R}^n,\,\xi\in\mathbb{R}^m} \ \frac{1}{2}\|w\|^2 + \frac{C}{m}\sum_{j=1}^{m}\xi_j$$

$$\text{s.t.} \quad w^T\underbrace{\left(\varphi(x_{j1})-\varphi(x_{j2})\right)}_{\text{denote by } \delta(x_j)} \geq 1-\xi_j,\ \xi_j\geq 0,\ \forall j=1\text{ to }m.$$

Write down a Lagrange dual of the problem in the space below. Simplify as much as possible. (2.5marks)

Special case of derivation in lecture

$$\max_{\alpha\in\mathbb{R}^m} \ 1^T\alpha - \frac{1}{2}\alpha^T Q\alpha,$$

$$\text{s.t.} \quad 0\leq\alpha\leq\frac{C}{m}.$$

Q is matrix with ij entry as $\delta(x_i)^T\delta(x_j)$

$$= k(x_{i1},x_{j1}) + k(x_{i2},x_{j2}) - k(x_{i1},x_{j2}) - k(x_{i2},x_{j1})$$

Write down all the optimality conditions in the space below. Simplify as much as possible. (2.5marks)

Special case of derivation in lecture

Most simplified:

$$\begin{cases} \alpha_i = 0 \Rightarrow \sum_{j=1}^{m}\alpha_j Q_{ij} \geq 1 \\ 0<\alpha_i<\frac{C}{m} \Rightarrow \sum_{j=1}^{m}\alpha_j Q_{ij} = 1 \\ \alpha_i = \frac{C}{m} \Rightarrow \sum_{j=1}^{m}\alpha_j Q_{ij} \leq 1. \end{cases}$$

In case one wants to maintain primals too, here are additional conds:

$$w = \sum_{i=1}^{m}\alpha_i\delta(x_i), \quad \xi_i = \begin{cases} 0 & \text{if } \alpha_i < \frac{C}{m} \\ 1-\sum_{j=1}^{m}\alpha_j Q_{ij} & \text{if } \alpha_i = C/m. \end{cases}$$