Date: Tuesday, September 27, 2022. 8:00 am - 9:00 am. Max Marks: 30
Instructions:
This exam is closed notes/books/resources. There are 5 questions. Start answer to each question on a new page. You need to justify all your answers. Answers with insufficient justification may not full points.
Questions:

1. Let a training dataset be given as $D = \{(x^{(i)}, y^{(i)})\}_{i=1}^{m}$. Let $\{D_1, D_2 \cdots D_k\}$ denote K datasets (each containing m examples) constructed by bootstrapping, i.e., each of the m examples in the dataset D_k is obtained by sampling uniformly at random (with replacement) from the of original training data. Consider learning K decision trees on these bootstrapped datasets, and let the corresponding learned trees be given by the set $\{T_1, \cdots T_2, \cdots, T_K\}$. The set of these decision trees can be thought of as a Random Forest, as discussed in the class. Consider a test point x, and let $T_k(x)$ denote the prediction of tree T_k on x. Assume a binary classification problem (i.e., $y \in \{0, 1\}$), and assume that each tree outputs the probability of assigning label 1 to the input example. Let the prediction of the random forest constructed as above be given as $T_{avg}(x) = \frac{1}{K}\sum_k T_k(x)$. Since each T_k is constructed by learning over D_k which in turn is constructed randomly using the process of bootstrapping, the prediction of T_k on x, denoted as $T_k(x)$, itself a random variable. Let $E[T_k(x)]$ denote the expected value this random variable, and $Var(T_k(x))]$ denote its variance. Further, since $T_{avg}(x)$ is average of random variables $T_k(x)$'s, it is also a random variable. Let $E[T_{avg}(x)]$, $Var[T_{avg}(x)]$ denote the average and variance of $T_{avg}(x)$, respectively.

   (a) **(2 points)** Show that $E[T_{avg}(x)] = E[T_k(x)]$.

   (b) **(3 points)** Derive the expression for $Var[T_{avg}(x)]$ as a function of $Var[T_k(x)]$. Show that $Var[T_{avg}(x)] \leq Var[T_k(x)]$.

2. Consider a variation of linear regression, called locally weighted linear regression, where error corresponding to each point is given a different weight. Specifically, given a dataset $\{x^{(i)}, y^{(i)}\}_{i=1}^{m}$, the error function for locally weight linear regression is given as: $J_{lw}^{x}(\theta) = \frac{1}{m}\sum_{i=1}^{m} w_i(y^{(i)} - \theta^T x^{(i)})^2$, where $w^{(i)} = exp(-\gamma||x^{(i)} - x||^2)$ for some value of $\gamma \geq 0$. Note that in this case, we define a different error function for each x, and hence, will learn a different regression model for each test input x.

   (a) **(3 points)** For any given input x, show that $J_{lw}^{x}(\theta)$ is a convex function of θ. For what value(s) of γ does the locally weighted linear regression reduces to the standard linear regression? Why?

   (b) **(2 points)** Let f denote the true target function which is linear. Let each observed y value in the training data be obtained by adding a random iid Gaussian noise to the true target value. Consider learning a linear regression model, and a locally weighted linear regression model [1]. In the limit of very low amount of data, and finite (fixed) $\gamma > 0$, do you expect the standard linear regression to do better than locally weighted linear regression, or vice-versa? What happens when the amount of training data increases? Explain.

   (c) **(1 points)** Next assume that f is quadratic, and assume the same kind of noise distribution in the y's observed in the training data as in the part above. Assume that you have large amounts of data available for training. Which of the two models do you expect do better, and why?

3. (a) **(3 points)** Show that Mutual Information is a symmetric measure, i.e., if X, Y are random variables, with some joint distribution defined over them, then $MI(X, Y) = MI(Y, X)$. Recall that Mutual Information of Y with respect to X is the difference between entropy of Y, $H(Y)$, and the conditional entropy of Y given X, $H(Y|X)$.

[1] You can fix some arbitrary (test) input point x for this analysis if this helps

(b) **(3 points)** Let $\phi : R^n \rightarrow R^N$ correspond to a feature transformation. Let $K : \mathcal{R}^n \times \mathcal{R}^n \rightarrow \mathcal{R}$ be a Kernel function corresponding to feature transformation ϕ, i.e., K is defined such that $K(x^{(i)}, x^{(j)}) = \phi(x^{(i)})^T \phi(x^{(j)})$, $\forall x^{(i)}, x^{(j)} \in \mathcal{R}^n$. Given a set of m points $\{(x^{(i)})\}_{i=1}^m$ in the instance space, $x^{(i)} \in \mathcal{R}^n$, let K_M denote the kernel matrix for the kernel function K. Show that K_M is positive semi-definite (do not forget to show that K_M is symmetric).

4. Consider the problem of text classification, and assume the multinoulli event model as discussed in class, where we assume a feature for every word position, and that parameters of the multinoulli distributions at each word position are tied, i.e., $\theta_{jl|k} = \theta_{j'l|k}$, $\forall j, j', l, k$, where $\theta_{jl|k}$ is the multinoulli parameter denoting the probability of the l^{th} word occurring at j^{th} position in document of class k (similarly for $\theta_{j'l|k}$). Let the training data points be given as $\{(x^{(i)}, y^{(i)})\}_{i=1}^m$. Let $n^{(i)}$ denote the number of words in the i^{th} and $|V|$ be the size of the vocabulary. Let r denote the total number of target labels. Assume any other symbols are as used in the class.

   (a) **(5 points)** Show that

$$\frac{\theta_{jl|k}}{\theta_{lj|k}} = \frac{\sum_{i=1}^{m} \sum_{k=1}^{r} \sum_{j'=1}^{n^{(i)}} \sum_{l=1}^{|V|} \mathbb{1}\{y^{(i)} = k\} \mathbb{1}\{x_{j'}^{(i)} = l\}}{\sum_{i=1}^{m} \sum_{k=1}^{r} \sum_{j'=1}^{n^{(i)}} \sum_{l=1}^{|V|} \mathbb{1}\{y^{(i)} = k\} \mathbb{1}\{x_{j'}^{(i)} = l'\}}$$

   (b) **(1 points)** As a result (or otherwise), show that

$$\theta_{jl|k} = \frac{\sum_{i=1}^{m} \sum_{k=1}^{r} \sum_{j'=1}^{n^{(i)}} \sum_{l=1}^{|V|} \mathbb{1}\{y^{(i)} = k\} \mathbb{1}\{x_{j'}^{(i)} = l\}}{\sum_{i=1}^{m} \sum_{k=1}^{r} \sum_{j'=1}^{n^{(i)}} \mathbb{1}\{y^{(i)} = k\} 1}$$

5. Assume a set of training examples given as $\{(x^{(i)}, y^{(i)})\}_{i=1}^m$. Consider learning a linear classifier, with parameters w, b, by optimizing the following objective:

$$\min_{w,b} \frac{1}{2} w^T w + C \sum_i max[0, 1 - y^{(i)}(w^T x^{(i)} + b)] \quad (1)$$

   (a) **(3 points)** Show that optimising the above objective gives exactly the same hyperplane, as learned by the soft-margin SVM classifier.

   (b) **(1 points)** Can you directly apply gradient descent over the above objective to learn a hyperplane? Why or why not?

   (c) **(3 points)** Consider a convex function $f(\theta)$ ($\theta \in \mathcal{R}^n$), such that the partial derivative $\frac{\delta f}{\delta \theta_j}$ is not defined at a finite number of values of θ_j but left and right partial derivatives exist at θ_j. For instance, for $f(\theta) = \sum_j |\theta_j|$, the partial derivative with respect to θ_j does not exist at $\theta_j = 0$. But both left and right partial derivatives are well defined. We define the j^{th} component of the sub-gradient of a function $f(\theta)$ at θ_j (irrespective of whether function is partially differentiable at θ_j or not) to be a value α, such that α lies between the left and right partial derivatives at θ_j. In the above example, we could define the ((j^{th}) component of) sub-gradient at $\theta_j = 0$ to be 0 (right and left partial derivatives are -1 and 1 respectively). In fact, in this case, α could be any value such that $-1 \leq \alpha \leq 1$.

      i. Show that notion of sub-gradients generalizes the notion of gradients over convex functions which are non-differentiable at a finite number of points.

      ii. Compute the sub-gradient of the alternate SVM objective defined in Equation 1 above with respect to parameters w and b. Use the computed sub-gradient to provide the parameter update rule for the gradient descent algorithm for optimizing this alternative objective.