Name: _____

Roll No.: _____ **Dept.**: _____

IIT Kanpur
CS771 Intro to ML
End-semester Examination
Date: November 17, 2017

Instructions:

Total: **120 marks**

Problem 1 (True or False: 12 X 1 = 12 marks). For each of the following simply write **T** or **F** in the box.

1. **F** The time it takes to make a prediction using a decision tree depends on the number of nodes in that tree. (The prediction time depends on the depth of the tree, more specifically the depth of the deepest leaf.)
2. **T** If $f(\mathbf{x})$ is a convex function for $\mathbf{x} \in \mathbb{R}^d$ and $g(\mathbf{x}) = \langle \mathbf{v}, \mathbf{x} \rangle + c$ for some fixed vector $\mathbf{v} \in \mathbb{R}^d$ and $c \in \mathbb{R}$, then $f + g$ is always a convex function. (The sum of two convex functions is always convex, and affine functions, such as g, are convex.)
3. **F** The k-means++ algorithm for clustering, initializes the cluster centers to k points in the dataset that are closest to each other. (The k-means++ algorithms chooses initial centers that are well seperated.)
4. **F** In CNNs, a larger pool size, e.g., max pooling a larger number of neurons together in a single pool, preserves more information about the output of the layer to which pooling is applied. (We lose all information in a max pooled set except the maximum value. Thus, large pool sizes throw away more information.)
5. **T** When working with large datasets, held-out validation is cheaper to execute as compared to k-fold cross validation. (k fold validation requires training k models which can be expensive if dataset size is large.)
6. **F** The k-means++ algorithm cannot be used when performing kernel k-means clustering with a nonlinear Mercer kernel with an infinite dimensional feature map. (Since distances $\|\cdot\|_{\mathcal{H}}$ can be computed in the RKHS $\mathcal{H}$ very easily, the k-means++ algorithm can be easily extended to kernelized settings.)
7. **F** If we learn a single model from a model class and find that the learnt model is overfitting to the data, then between bagging and boosting, boosting is better way to fix the problem. (To reduce variance, bagging is more effective. Boosting is usually used as a bias reduction technique.)
8. **F** The Power method can be used to solve the PCA problem but it cannot be used to solve the kernel PCA problem. (Both PCA and kernel PCA require finding eigenvectors/values of certain matrices for which the power method and the peeling technique are very effective.)
9. **T** Solving the SVM problem is cheaper when using a linear kernel than it is when using the Gaussian kernel. (Performing gradient/coordinate descent in the dual is more expensive with non-linear kernels. Solving the primal is mostly infeasible with non-linear kernels. With linear kernels, solving the primal is feasible, as well as dual descent is more efficient.)
10. **T** A neural network with a single hidden layer and a single output node with all nodes except input layer nodes using the sigmoid activation function will always learn a continuous function. (Since the sigmoid function is continuous and the operations performed by the network are only linear combination and sigmoid application, all of which preserve continuity of the final function, the final network will learn a continuous function.)
11. **T** For small scale recommendation problems, say with only 10 items to recommend from, we can cast the problem as 10 separate classification problems. (For small-scale problems, this is an acceptable approach and indeed, a widely used benchmark often known by the names "one-vs-all" or "binary relevance".)
12. **F** When interacting with a typical recommendation system, users usually tell the recommendation system what items they like and what items they do not like. (Users only reveal a subset of items that they do like. They seldom reveal which items they do not like.)

Name:

Roll No.: **Dept.**:

IIT Kanpur
CS771 Intro to ML
End-semester Examination
Date: November 17, 2017

Problem 2 (Ultra Short Answer: 6 x 4 = 24 marks)**.** Give your answers in the space provided only.

1. Write down below, a feature map corresponding to the Mercer kernel $K(\mathbf{z}^1, \mathbf{z}^2) = (\langle \mathbf{z}^1, \mathbf{z}^2 \rangle)^2$ where $\mathbf{z}^i = (x_i, y_i), i = 1, 2$ are 2D vectors. Note that maps will smaller dimensionality will get more credit.

   Solution:
   $$\phi_K(\mathbf{z}) = [x^2,\ y^2,\ \sqrt{2} \cdot xy]$$

2. I have 1000 data points which I wish to split into a training and a held-out validation set. Tom tells me to take 990 points as training and 10 as validation. Dick declares that dividing into 700 training points and 300 validation points is prefereable whereas Harry has heard that taking 10 training and 990 validation points works best. Which friend should I agree with? Why? Why should I disagree with the other two?

   Solution: You should trust Dick since this suggestion has a healthy number of training and validation points. Tom's suggestion has very few validation points so any decisions made using them would be low in confidence. Harry's solution will give very confident decisions but the models learnt will be bad since very few training points are used so we will be very confident that they are bad which is still useless.

3. My friend has trained a binary classifier which gets only 10% classification accuracy. What is the simplest thing I can do to boost the accuracy of this classifier to a more respectable level?

   Solution: Simply construct a new classifier which says red when your friend's classifier says green and vice versa. Your classification accuracy will jump to 90%.

4. Let K_{int} be the intersection kernel: for any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$, $K_{\text{int}}(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{d} \min\{\mathbf{x}_i, \mathbf{y}_i\}$. Let $\phi_{\text{int}} : \mathbb{R}^d \to \mathbb{R}^D$ be a feature map corresponding to K_{int}. Write down the expression for $\|\phi_{\text{int}}(\mathbf{x}) - \phi_{\text{int}}(\mathbf{y})\|_2^2$.

   Solution:
   $$\|\phi_{\text{int}}(\mathbf{x}) - \phi_{\text{int}}(\mathbf{y})\|_2^2 = \|\mathbf{x} - \mathbf{y}\|_1 = \sum_{i=1}^{d} |\mathbf{x}_i - \mathbf{y}_i|,$$
   since $\min\{x, x\} = x$ and $x + y - 2 \cdot \min\{x, y\} = |x - y|$.

5. I have a regression dataset $\{(\mathbf{x}^i, y^i)\}_{i \in [n]}$, $\mathbf{x}^i \in \mathcal{X}, y^i \in \mathbb{R}$ and a kernel $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$. Let $G \in \mathbb{R}^{n \times n}$ denote the Gram matrix with $G_{ij} = K(\mathbf{x}^i, \mathbf{x}^j)$. I perform landmarking with all training points as landmarks i.e. $\hat{\phi}(\mathbf{x}) = [K(\mathbf{x}, \mathbf{x}^1), \ldots, K(\mathbf{x}, \mathbf{x}^n)] \in \mathbb{R}^n$. Solve $\hat{\mathbf{w}} = \arg\min_{\mathbf{w} \in \mathbb{R}^n} \lambda \cdot \|\mathbf{w}\|_2^2 + \sum_{i=1}^{n} \left(y^i - \left\langle \mathbf{w}, \hat{\phi}(\mathbf{x}^i) \right\rangle \right)^2$ (i.e. ridge regression using the landmarked feature map $\hat{\phi}$) and write down the expression for $\hat{\mathbf{w}}$.

   Solution: Let $\Phi = [\hat{\phi}(\mathbf{x}^1), \ldots, \hat{\phi}(\mathbf{x}^n)]$. Then the solution to the ridge regression problem is
   $$\hat{\mathbf{w}} = (\Phi\Phi^\top + \lambda \cdot I_n)^{-1} \Phi \mathbf{y},$$
   where $\mathbf{y} = [y^1, \ldots, y^n]^\top$. However, notice that $\Phi = G$ and G is symmetric. So we have
   $$\hat{\mathbf{w}} = \left(G^2 + \lambda \cdot I_n\right)^{-1} G\mathbf{y}$$

Name:

Roll No.: **Dept.**:

IIT Kanpur
CS771 Intro to ML
End-semester Examination
Date: November 17, 2017

---

6. Note that the predictor we learnt in part 5 looks like $\left\langle \hat{\mathbf{w}}, \hat{\phi}(\mathbf{x}) \right\rangle = \sum_{i=1}^{n} \gamma_i \cdot K(\mathbf{x}, \mathbf{x}^i)$ where $\gamma_i = \hat{\mathbf{w}}_i$. Now let $\phi_K : \mathcal{X} \to \mathcal{H}$ be a feature map for the kernel K so that $K(\mathbf{x}^i, \mathbf{x}^j) = \langle \phi_K(\mathbf{x}^i), \phi_K(\mathbf{x}^j) \rangle$ for all $\mathbf{x}^i, \mathbf{x}^j \in \mathcal{X}$. Suppose we had instead solved $\hat{\mathbf{W}} = \arg\min_{\mathbf{W} \in \mathcal{H}} \lambda \cdot \|\mathbf{W}\|_{\mathcal{H}}^2 + \sum_{i=1}^{n} \left(y^i - \langle \mathbf{W}, \phi_K(\mathbf{x}^i) \rangle\right)^2$, i.e. performed kernel ridge regression on the dataset directly instead of landmarking then, as we saw in class, we would have obtained a predictor $\left\langle \hat{\mathbf{W}}, \phi_K(\mathbf{x}) \right\rangle = \sum_{i=1}^{n} \delta_i \cdot K(\mathbf{x}, \mathbf{x}^i)$ where $\boldsymbol{\delta} = [\delta_1, \ldots, \delta_n]^\top = (G + \lambda \cdot I)^{-1}\mathbf{y}$ where $\mathbf{y} = [y^1, \ldots, y^n]^\top$. Show that if G is invertible and we set $\lambda = 0$, then $\gamma_i = \delta_i$ for all $i \in [n]$. This means that kernel regression and landmarking-based regression will always learn the same predictor!

Solution: In the previous solution we saw that $\boldsymbol{\gamma} = \hat{\mathbf{w}} = (G^2 + \lambda \cdot I_n)^{-1} G\mathbf{y}$ where $\boldsymbol{\gamma} = [\gamma_1, \ldots, \gamma_n]^\top$. We have been given $\boldsymbol{\delta} = (G + \lambda \cdot I_n)^{-1}\mathbf{y}$. For $\lambda = 0$, since G is invertible, we have

$$\boldsymbol{\gamma} = \boldsymbol{\delta} = G^{-1}\mathbf{y}$$

Problem 3 (Short Answer: 6 x 6 = 36 marks)**.** For each of the problems, give your answer in space provided.

1. Let $\mathbf{x} = [1,\ 1]^\top, \mathbf{y} = [2,\ 1]^\top \in \mathbb{R}^2$ and let $f : \mathbb{R}^2 \to \mathbb{R}^2$ with $f(\mathbf{z}) = z_1 \cdot \mathbf{x} + z_2 \cdot \mathbf{y}$ for any $\mathbf{z} = [z_1, z_2]^\top \in \mathbb{R}^2$. Further, $\mathbf{z} = g(r) = [r^2,\ r^3]$ where $r \in \mathbb{R}$. Show how chain rule is applied here giving major steps of the calculation, write down the expression for $\dfrac{df}{dr}$, and also evaluate $\dfrac{df}{dr}$ at $r = 2$.

Solution: The chain rule gives us $\frac{df}{dr} = \frac{df}{d\mathbf{z}} \cdot \frac{d\mathbf{z}}{dr}$. We have $\frac{df}{d\mathbf{z}} = J_f$, the Jacobian of the function f which is

$$J_f = \begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix}$$

We also have $\frac{d\mathbf{z}}{dr} = J_g$, the Jacobian of the function g which is

$$J_g = \begin{bmatrix} 2r \\ 3r^2 \end{bmatrix}$$

This gives us $\frac{df}{dr} = J_f \cdot J_g = [(2r + 6r^2),\ (2r + 3r^2)]^\top$. At $r = 2$, we have $\frac{df}{dr}\big|_{r=2} = [28,\ 16]^\top$.

2. Give an example of a Mercer kernel $K : \mathbb{R}^2 \times \mathbb{R}^2 \to \mathbb{R}$ and three vectors $\mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathbb{R}^2$ such that $K(\mathbf{x}, \mathbf{y}) < K(\mathbf{x}, \mathbf{z})$ and $\|\phi_K(\mathbf{x}) - \phi_K(\mathbf{y})\|_{\mathcal{H}} < \|\phi_K(\mathbf{x}) - \phi_K(\mathbf{z})\|_{\mathcal{H}}$, where $\phi_K : \mathbb{R}^2 \to \mathcal{H}$ is the feature map for the kernel K. This means that the kernel thinks $\mathbf{x}$ and $\mathbf{y}$ are less similar than $\mathbf{x}$ and $\mathbf{z}$ but in the RKHS, $\mathbf{x}$ and $\mathbf{y}$ are closer than $\mathbf{x}$ and $\mathbf{z}$. You need to give the explicit form of the kernel, the three vectors, as well as values of $K(\mathbf{x}, \mathbf{y}), K(\mathbf{x}, \mathbf{z}), \|\phi_K(\mathbf{x}) - \phi_K(\mathbf{y})\|_{\mathcal{H}}, \|\phi_K(\mathbf{x}) - \phi_K(\mathbf{z})\|_{\mathcal{H}}$ for your construction.

Solution: Consider the linear kernel $K(\mathbf{x}, \mathbf{y}) = \langle \mathbf{x}, \mathbf{y} \rangle$ so that the RKHS is $\mathbb{R}^2$ itself and the feature map is identity $\phi : \mathbf{x} \mapsto \mathbf{x}$. Consider $\mathbf{x} = [1,\ 0]^\top, \mathbf{y} = [2,\ 0]^\top, \mathbf{z} = [10,\ 0]^\top$. We have $K(\mathbf{x}, \mathbf{y}) = 2 < 10 = K(\mathbf{x}, \mathbf{z})$ but we also have $\|\phi_K(\mathbf{x}) - \phi_K(\mathbf{y})\|_{\mathcal{H}} = \|\mathbf{x} - \mathbf{y}\|_2 = 1 < 9 = \|\mathbf{x} - \mathbf{z}\|_2 = \|\phi_K(\mathbf{x}) - \phi_K(\mathbf{z})\|_{\mathcal{H}}$.

3. Suppose $\phi : \mathbb{R}^2 \mapsto \mathbb{R}^4$ is a linear map i.e. $\phi(\mathbf{x}+\mathbf{y}) = \phi(\mathbf{x}) + \phi(\mathbf{y})$ and $\phi(c \cdot \mathbf{x}) = c \cdot \phi(\mathbf{x})$ for all $\mathbf{x}, \mathbf{y} \in \mathbb{R}^2, c \in \mathbb{R}$. Suppose $\phi([1,\ 1]) = [1,\ 1,\ 2,\ 1], \phi([1,\ 2]) = [1,\ 2,\ 3,\ 2]$, and $\phi([2,\ 0]) = [2,\ 0,\ 2,\ 0]$. Find the matrix $M \in \mathbb{R}^{4 \times 2}$ such that $\phi(\mathbf{x}) = M\mathbf{x}$ for all $\mathbf{x} \in \mathbb{R}^2$. Suppose I learn a model $\mathbf{W} = [2,\ 3,\ 1,\ 1] \in \mathbb{R}^4$. Find a model $\mathbf{w} \in \mathbb{R}^2$ such that $\langle \mathbf{w}, \mathbf{x} \rangle = \langle \mathbf{W}, \phi(\mathbf{x}) \rangle$ for all $\mathbf{x} \in \mathbb{R}^2$. Fill entries of M and $\mathbf{w}$ below.

Name:

Roll No.: **Dept.**:

IIT Kanpur
CS771 Intro to ML
End-semester Examination
Date: November 17, 2017

Solution: The third example tells us that $\phi([1,0]) = [1,\ 0,\ 1,\ 0]$. Subtracting this from the first example tells us that $\phi([0,1]) = [0,\ 1,\ 1,\ 1]$. Note that this does agree with the second example too. This gives us M. To get $\mathbf{w}$ simply observe that $\langle \mathbf{W}, \phi(\mathbf{x}) \rangle = \mathbf{W}^\top \phi(\mathbf{x}) = \mathbf{W}^\top M \mathbf{x} = (M^\top \mathbf{W})^\top \mathbf{x}$. Thus, $\mathbf{w} = M^\top \mathbf{W}$.

$$M = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \\ 0 & 1 \end{bmatrix} \qquad \mathbf{w} = \begin{bmatrix} 3 \\ 5 \end{bmatrix}$$

4. Consider the following ridge regression problem $\min_{\mathbf{w} \in \mathbb{R}^d} 0.5 \cdot \|\mathbf{w}\|_2^2 + 0.5 \cdot \sum_{i=1}^n (y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle)^2$. Denote $X = [\mathbf{x}^1, \ldots, \mathbf{x}^n] \in \mathbb{R}^{d \times n}, \mathbf{y} = [y^1, \ldots, y^n]^\top \in \mathbb{R}^n$. Write down the gradient and the Hessian of the objective function at an arbitrary point $\mathbf{w} \in \mathbb{R}^d$. Then start at $\mathbf{w}^0 = \mathbf{0}$ and execute the Newton method on this problem for 3 iterations. Write down expressions for the iterates $\mathbf{w}^1, \mathbf{w}^2, \mathbf{w}^3$ that you obtain.

Solution: The objective function is $\frac{1}{2} \cdot \|\mathbf{w}\|_2^2 + \frac{1}{2} \left\| X^\top \mathbf{w} - \mathbf{y} \right\|_2^2$. The gradient for this objective at a point $\mathbf{w} \in \mathbb{R}^d$ is $\mathbf{g} = \mathbf{w} + X(X^\top \mathbf{w} - \mathbf{y}) = (XX^\top + I)\mathbf{w} - X\mathbf{y}$. The Hessian of the objective is $H = (XX^\top + I)$. The Newton step executes $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t - H^{-1}\mathbf{g}^t$. Thus we have

$$\begin{aligned} \mathbf{w}^{t+1} &= \mathbf{w}^t - H^{-1}\mathbf{g}^t \\ &= \mathbf{w}^t - (XX^\top + I)^{-1}((XX^\top + I)\mathbf{w}^t - X\mathbf{y}) \\ &= \mathbf{w}^t - (XX^\top + I)^{-1}(XX^\top + I)\mathbf{w}^t + (XX^\top + I)^{-1}X\mathbf{y} \\ &= \mathbf{w}^t - \mathbf{w}^t + (XX^\top + I)^{-1}X\mathbf{y} \\ &= (XX^\top + I)^{-1}X\mathbf{y} \end{aligned}$$

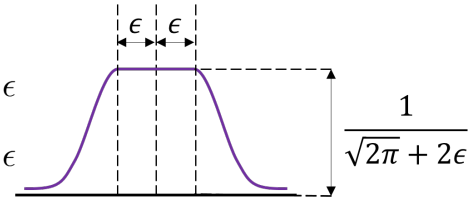
Thus, the next iterate $\mathbf{w}^{t+1}$ for Newton method is the same no matter what the starting point. Notice that the next iterate $\mathbf{w}^{t+1}$ is also the optimal model i.e. no matter what the starting point $\mathbf{w}^t$ is, Newton method reaches the optimal point in a single step. Thus we have $\mathbf{w}^1 = \mathbf{w}^2 = \mathbf{w}^3 = (XX^\top + I)^{-1}X\mathbf{y}$.

5. Recall the ϵ-insensitive loss defined as $\ell_\epsilon(y, \hat{y}) = 0$ if $|y - \hat{y}| \leq \epsilon$ and otherwise $\ell_\epsilon(y, \hat{y}) = (|y - \hat{y}| - \epsilon)^2$ where $\hat{y}, y \in \mathbb{R}$. Consider the following optimization problem with $\mathbf{x}^i \in \mathbb{R}^d, y^i \in \mathbb{R}$ and write down a likelihood distribution for $\mathbb{P}[y^i \,|\, \mathbf{x}^i, \mathbf{w}]$ and prior $\mathbb{P}[\mathbf{w}]$ such that $\hat{\mathbf{w}}$ is the MAP estimate for your model. Give explicit forms for the density functions but you need not calculate normalization constants.

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w} \in \mathbb{R}^d} \sum_{i=1}^n \ell_\epsilon(y^i, \langle \mathbf{w}, \mathbf{x}^i \rangle) + \|\mathbf{w}\|_2^2$$

Solution: Use a standard Gaussian prior $\mathbb{P}[\mathbf{w}] = \frac{1}{\sqrt{(2\pi)^d}} \exp\left(-\frac{1}{2}\|\mathbf{w}\|_2^2\right)$ and a likelihood with the following density function. The mode of the distribution corresponds to zero loss.

$$\mathbb{P}\left[y^i \,|\, \mathbf{x}^i, \mathbf{w}\right] = \begin{cases} \frac{1}{\sqrt{2\pi}+2\epsilon} & \text{if } \left|y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle\right| \leq \epsilon \\ \frac{1}{\sqrt{2\pi}+2\epsilon} \exp\left(-\frac{(|y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle| - \epsilon)^2}{2}\right) & \text{if } \left|y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle\right| > \epsilon \end{cases}$$



6. The perceptron algorithm makes the update $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t + \eta_t y^t \cdot \mathbf{x}^t$ when it misclassifies the t-th data point $(\mathbf{x}^t, y^t) \in \mathbb{R}^d \times \{-1, +1\}$. Show that if we decide to use a constant step length i.e. $\eta_t \equiv \eta$ for all t, it does not matter which value of η we choose so long as we choose a value $\eta > 0$. Specifically, show that the perceptron algorithm makes the same set of mistakes when using the constant step length η, for all $\eta > 0$.

Name:

Roll No.: Dept.:

IIT Kanpur
CS771 Intro to ML
End-semester Examination
Date: November 17, 2017

Solution: The result claimed in the question requires the initial model to be set to the zero vector. If the initial model is a non-zero vector then one can construct counter examples where the step length does matter. Credit will be given to both those who have shown the claimed result (possibly after assuming a zero initialization) as well as those who have given a valid counter example starting with a non-zero initialization. However, all such counter examples must use the same initialization no matter what the step length. If the initialization is allowed to be different for the different step lengths then the result becomes meaningless.

For now assume that $\mathbf{w}^0 = \mathbf{0}$. We will prove the result by induction. Since the initialization is the same, either the first point will be misclassified or will be correctly classified no matter what the step length. For the induction step, assume till time t, no matter what the step length used, the same set of examples $S_t = \{\tau < t : y^\tau \langle \mathbf{w}^{\tau-1}, \mathbf{x}^\tau \rangle < 0\}$ was misclassified in the past. Then, if the step length used is η, then we have $\mathbf{w}^t = \sum_{\tau \in S_t} \eta y^\tau \mathbf{x}^\tau$. Given the next data point $(y^{t+1}, \mathbf{x}^{t+1})$, this model will propose the following prediction $\langle \mathbf{w}^t, \mathbf{x}^{t+1} \rangle = \eta \cdot \sum_{\tau \in S_t} \langle \mathbf{x}^\tau, \mathbf{x}^{t+1} \rangle$. Note that the sign of this quantity does not depend on η so long as $\eta > 0$. This means either $y^{t+1} \langle \mathbf{w}^t, \mathbf{x}^{t+1} \rangle < 0$ or ≥ 0 irrespective of what step length was used. This proves the result.

Problem 4 (Long Answer: 12 + 6 + 6 = 24 marks)**.** Consider the problem of heteroscedastic regression, a curious variant of linear regression where the noise added to each data point comes from a different distribution! Let $\mathbf{x}^i \in \mathbb{R}^d, i = 1, \ldots, n$ denote the covariates/feature vectors. The responses are generated as $y^i = \langle \mathbf{w}, \mathbf{x}^i \rangle + \epsilon^i$, where the noise $\epsilon^i \sim \mathcal{N}(0, \sigma_i^2)$ for the i-th data point has variance σ_i^2 . We are shown $\{(\mathbf{x}^i, y^i)\}_{i \in [n]}$ but model $\{\sigma_i\}_{i \in [n]}$ as latent variables. Note that this is a discriminative model and $\mathbf{x}^i$ are not probabilistically modelled. You may find the shorthands $X = [\mathbf{x}^1, \ldots, \mathbf{x}^n], \mathbf{y} = [y^1, \ldots, y^n], \Sigma = \text{diag}(\sigma_1^2, \ldots, \sigma_n^2)$ to be helpful. Also, in all questions below, your expressions may have unspecified normalization constants. Give brief/concise derivations.

1. Derive an expression for $\mathbb{P}[\sigma_i \,|\, y^i, \mathbf{x}^i, \mathbf{w}]$ using the prior $\mathbb{P}[\sigma_i] = 1$ if $\sigma_i \in [0, 1]$ and $\mathbb{P}[\sigma_i] = 0$ otherwise. Then derive the MAP estimate for σ_i i.e. $\arg\max \; \mathbb{P}[\sigma_i \,|\, y^i, \mathbf{x}^i, \mathbf{w}]$ assuming the model $\mathbf{w}$ is known. For simplicity, assume $\mathbb{P}[\sigma_i \,|\, \mathbf{x}^i, \mathbf{w}] = \mathbb{P}[\sigma_i]$ i.e. $\mathbf{w}$ and $\mathbf{x}^i$ had nothing to do with the selection of σ_i.

   Solution: We have $\mathbb{P}[y^i \,|\, , \mathbf{x}^i, \mathbf{w}, \sigma_i] = \frac{1}{\sigma_i \sqrt{2\pi}} \exp\left(-\frac{(y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle)^2}{2\sigma_i^2} \right)$ due to the noise condition. Also we have, applying Bayes rule

$$\mathbb{P}\left[\sigma_i \,|\, y^i, \mathbf{x}^i, \mathbf{w}\right] = \frac{\mathbb{P}[y^i \,|\, \mathbf{x}^i, \mathbf{w}, \sigma_i] \cdot \mathbb{P}[\sigma_i \,|\, \mathbf{x}^i, \mathbf{w}]}{\mathbb{P}[y^i \,|\, \mathbf{x}^i, \mathbf{w}]} = \frac{\mathbb{P}[y^i \,|\, \mathbf{x}^i, \mathbf{w}, \sigma_i] \cdot \mathbb{P}[\sigma_i]}{\mathbb{P}[y^i \,|\, \mathbf{x}^i, \mathbf{w}]} = \begin{cases} 0, & \text{if } \sigma_i \notin [0,1] \\ \frac{\mathbb{P}[y^i \,|\, \mathbf{x}^i, \mathbf{w}, \sigma_i]}{\mathbb{P}[y^i \,|\, \mathbf{x}^i, \mathbf{w}]} & \text{if } \sigma_i \in [0,1] \end{cases}$$

   Thus, the MAP solution is arrived by solving

$$\arg\min_{\sigma \in [0,1]} \; \log \sigma + \frac{(y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle)^2}{2\sigma^2}$$

   Applying first order optimality tells us that the gradient of the objective vanishes at $|y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle|$. If this quantity is less than 1 (it is definitely non-negative) then this is the solution. Else observe that the function $\log x + \frac{a^2}{2x^2}$ is decreasing in the interval $(0, |a|)$. This means that if $|y^i - \langle \mathbf{w}, \mathbf{x}^i \rangle| > 1$, then the optimal solution is 1.

2. Derive an expression for $\mathbb{P}[\mathbf{w} \,|\, y^i, \mathbf{x}^i, \sigma_i]$ using a standard Gaussian prior $\mathbb{P}[\mathbf{w}] = \frac{1}{\sqrt{(2\pi)^d}} \exp(-\frac{1}{2} \|\mathbf{w}\|_2^2)$. Then derive the MAP estimate for $\mathbf{w}$ i.e. $\arg\max \; \mathbb{P}[\mathbf{w} \,|\, \mathbf{y}, X, \Sigma]$ assuming that $\{\sigma_i\}$ are known.

Name:

Roll No.: Dept.:

IIT Kanpur
CS771 Intro to ML
End-semester Examination
Date: November 17, 2017

Solution: We have

$$\mathbb{P}\left[\mathbf{w} \,|\, y^i, \mathbf{x}^i, \sigma_i\right] = \frac{\mathbb{P}\left[y^i \,|\, \mathbf{x}^i, \mathbf{w}, \sigma_i\right] \cdot \mathbb{P}\left[\mathbf{w}\right]}{\mathbb{P}\left[y^i, \sigma_i \,|\, \mathbf{x}^i\right]}$$

So, the MAP solution is

$$\arg\min_{\mathbf{w}\in\mathbb{R}^d} \frac{1}{2}\left\|\mathbf{w}\right\|_2^2 + \sum_{i=1}^n \frac{(y^i - \langle \mathbf{w}, \mathbf{x}^i\rangle)^2}{2\sigma_i^2} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \frac{1}{2}\left\|\mathbf{w}\right\|_2^2 + \frac{1}{2}(\mathbf{y} - X^\top \mathbf{w})^\top \Sigma^{-1}(\mathbf{y} - X^\top \mathbf{w})$$

First order optimality gives us the solution as $(X\Sigma^{-1}X^\top + \lambda I_n)^{-1}X\Sigma^{-1}\mathbf{y}$. Note that if all $\sigma_i = 1$ then the above just becomes the regular ridge regression solution.

3. Using the above estimates, give the pseudocode for an alternating optimization algorithm for estimating $\mathbf{w}$ that performs MAP-based hard assignments to the latent variables σ_i to solve the problem. Give precise update expressions in your pseudocode and not just vague statements.

Algorithm 1: HERO: HEterscedastic Regression via alternating Optimization

Input: Data points $\{(y^i, \mathbf{x}^i)\}_{i\in[n]}$

Let $\sigma_i^0 \leftarrow 1$ and denote $\Sigma^0 = \text{diag}((\sigma_1^0)^2, \ldots, (\sigma_n^0)^2)$ //Initialize
for $t = 1, 2, \ldots,$ **do**
  $\mathbf{w}^t \leftarrow (X\Sigma^{-1}X^\top + \lambda I_n)^{-1}X(\Sigma^{t-1})^{-1}\mathbf{y}$ //Update model
  $\sigma_i^t \leftarrow \max\left\{|y^i - \langle \mathbf{w}^t, \mathbf{x}^i\rangle|, 1\right\}$ //Update variances
  $\Sigma^t \leftarrow \text{diag}((\sigma_1^t)^2, \ldots, (\sigma_n^t)^2)$ //Just a shorthand
end for

Problem 5 (Long Answer: 8 + 16 = 24 marks)**.** For each of the problems, give your answer in space provided.

1. Let $R \in \mathbb{R}^{d\times d}$ be a symmetric, invertible matrix, $\mathbf{x}^i \in \mathbb{R}^d$, and $y^i \in \mathbb{R}$ for $i = 1, \ldots, n$. Using the same trick we used in class of introducing a new variable $\mathbf{r}_i = y^i - \langle \mathbf{w}, \mathbf{x}^i\rangle$ and corresponding constraints, solve the problem given below. Give 1) the Lagrangian, 2) the simplified dual optimization problem (with primal variables eliminated completely), 3) the dual solution and 4) the final primal solution $\hat{\mathbf{w}}$. Some shorthands you may find useful are $X = [\mathbf{x}^1, \ldots, \mathbf{x}^n] \in \mathbb{R}^{d\times n}$ and $H = X^\top R^{-1} X \in \mathbb{R}^{n\times n}$ i.e. $H_{ij} = (\mathbf{x}^i)^\top R^{-1}\mathbf{x}^j$.

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \frac{1}{2}\mathbf{w}^\top R\mathbf{w} + \frac{1}{2}\sum_{i=1}^n (y^i - \left\langle \mathbf{w}, \mathbf{x}^i\right\rangle)^2$$

Solution: The new optimization problem after introducing the dummy variables $\mathbf{r} \in \mathbb{R}^n$ is

$$\min_{\mathbf{w}\in\mathbb{R}^d, \mathbf{r}\in\mathbb{R}^n} \frac{1}{2}\mathbf{w}^\top R\mathbf{w} + \frac{1}{2}\left\|\mathbf{r}\right\|_2^2$$
$$\text{s.t. } \mathbf{r}_i = y^i - \left\langle \mathbf{w}, \mathbf{x}^i\right\rangle$$

The Lagrangian for this, upon introducing Lagrange variables $\boldsymbol{\alpha} \in \mathbb{R}^n$ is

$$\frac{1}{2}\mathbf{w}^\top R\mathbf{w} + \frac{1}{2}\left\|\mathbf{r}\right\|_2^2 + \sum_{i=1}^n \boldsymbol{\alpha}_i(y^i - \left\langle \mathbf{w}, \mathbf{x}^i\right\rangle - \mathbf{r}_i) = \frac{1}{2}\mathbf{w}^\top R\mathbf{w} + \frac{1}{2}\left\|\mathbf{r}\right\|_2^2 + \boldsymbol{\alpha}^\top(\mathbf{y} - X^\top\mathbf{w} - \mathbf{r})$$

The dual problem is

$$\max_{\boldsymbol{\alpha}\in\mathbb{R}^n} \min_{\mathbf{w}\in\mathbb{R}^d, \mathbf{r}\in\mathbb{R}^n} \frac{1}{2}\mathbf{w}^\top R\mathbf{w} + \frac{1}{2}\left\|\mathbf{r}\right\|_2^2 + \boldsymbol{\alpha}^\top(\mathbf{y} - X^\top\mathbf{w} - \mathbf{r})$$

Name:

Roll No.: Dept.:

IIT Kanpur
CS771 Intro to ML
End-semester Examination
Date: November 17, 2017

Applying first order optimality to the inner problem gives us $\mathbf{w} = R^{-1}X\boldsymbol{\alpha}$ and $\mathbf{r} = \boldsymbol{\alpha}$. Substituting these back gives us the simplified dual problem

$$\min_{\boldsymbol{\alpha}\in\mathbb{R}^n} \ \frac{1}{2}\boldsymbol{\alpha}^\top H\boldsymbol{\alpha} + \frac{1}{2}\left\|\boldsymbol{\alpha}\right\|_2^2 - \boldsymbol{\alpha}^\top \mathbf{y}$$

Applying first order optimality gives us the dual solution

$$\boldsymbol{\alpha} = (H+I)^{-1}\mathbf{y}$$

and using $\mathbf{w} = R^{-1}X\boldsymbol{\alpha}$, we get the primal solution

$$\hat{\mathbf{w}} = R^{-1}X(H+I)^{-1}\mathbf{y}$$

2. Flopkart.com has a customer who uses his account to make purchases for his entire family. There are k members in the family, each indexed by a vector $\mathbf{u}^1, \ldots, \mathbf{u}^k \in \mathbb{R}^d$. Each product on Flopkart.com is also indexed by a vector $\mathbf{v} \in \mathbb{R}^d$. It is known that the i-th member will give the product $\mathbf{v}$, a rating $r = \langle \mathbf{u}^i, \mathbf{v}\rangle + \epsilon$ where $\epsilon \sim \mathcal{N}(0,1)$. The customer has made n purchases with Flopkart. In the t-th purchase, the item $\mathbf{v}^t$ was purchased and a rating r^t was given to it but it is not known which member gave that rating. We have $\{(\mathbf{v}^t, r^t)\}_{t\in[n]}$ with us. Design an algorithm to estimate the user vectors corresponding to the k members of the family. Clearly specify what are the observed and latent variables in your model and give major steps of derivation whenever your algorithm uses a MAP/MLE/other estimate. Give pseudo code of your algorithm. Avoid very fine and unnecessary details e.g. application of first order optimality.

Solution: One way to model this problem is as a mixed regression problem with k-components. Denote $U = [\mathbf{u}^1, \ldots, \mathbf{u}^k] \in \mathbb{R}^{d\times k}, V = [\mathbf{v}^1, \ldots, \mathbf{v}^n] \in \mathbb{R}^{d\times n}, \mathbf{r} = [r^1, \ldots, r^n]^\top \in \mathbb{R}^n$. For each of the data points, $z^t \in [k]$ denotes the identity of the member who rated the item $\mathbf{v}^t$ and we denote $\mathbf{z} = [z^1, \ldots, z^n] \in [k]^n$. We use a Gaussian likelihood model for the ratings $\mathbb{P}\left[r^t \,|\, \mathbf{v}^t, z^t, U\right] = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(r^t - \left\langle \mathbf{u}^{z^t}, \mathbf{v}^t\right\rangle)}{2}\right)$ and assume that members are equally likely to rate items by assuming a uniform prior on the latent variables $\mathbb{P}\left[z^t = i\right] = \frac{1}{k}$ for all $i \in [k]$. We also assume a standard Gaussian prior on the user vectors $\mathbb{P}\left[\mathbf{u}^i\right] = \frac{1}{\sqrt{(2\pi)^d}} \exp\left(-\frac{1}{2}\left\|\mathbf{u}^i\right\|_2^2\right)$.

Given a setting of the latent variables z^t, we can find the MAP solution for the user vectors as follows

$$U = \arg\max \ \mathbb{P}\left[U \,|\, \mathbf{r}, V, \mathbf{z}\right] = \arg\max \ \mathbb{P}\left[\mathbf{r} \,|\, V, U, \mathbf{z}\right] \cdot \mathbb{P}\left[U\right]$$

The above reduces to a sequence of k ridge regression problems

$$\mathbf{u}^i = \arg\min \ \frac{1}{2}\left\|\mathbf{u}\right\|_2^2 + \frac{1}{2}\sum_{t:z^t=i}(r^t - \left\langle \mathbf{u}, \mathbf{v}^t\right\rangle)^2,$$

all of which have closed form solutions. Given an estimate of the user vectors, we can find the MAP assignments to the latent variables as follows

$$\mathbf{z} = \arg\max \ \mathbb{P}\left[\mathbf{z} \,|\, \mathbf{r}, V, U\right] = \arg\max \ \mathbb{P}\left[\mathbf{r} \,|\, \mathbf{z}, V, U\right] \cdot \mathbb{P}\left[\mathbf{z}\right]$$

Since we have assumed uniform priors, the above just becomes

$$z^t = \arg\max \ \mathbb{P}\left[\mathbf{r} \,|\, \mathbf{z}, V, U\right] = \arg\min_{i\in[k]} \ (r^t - \left\langle \mathbf{u}^i, \mathbf{v}^t\right\rangle)^2$$

Name:

Roll No.: **Dept.**:

IIT Kanpur
CS771 Intro to ML
End-semester Examination
Date: November 17, 2017

The above readily gives us a hard assignment alternating algorithm for this problem. Note that we may even perform soft assignment alternating optimization by applying the EM algorithm. That would assign a rating to every member of the family with different weights.

Algorithm 2: Family Feud

Input: Ratings $\{(r^t, \mathbf{v}^t)\}_{t\in[n]}$

Initialize $\mathbf{u}^{1,0}, \ldots, \mathbf{u}^{k,0}$
for $s = 1, 2, \ldots,$ **do**
  for $t = 1, \ldots, n$ **do**
    $z^{t,s} = \arg\min_{i\in[k]} (r^t - \langle \mathbf{u}^{i,s-1}, \mathbf{v}^t \rangle)^2$ //Reassign ratings to different members
  end for
  for $1 = 1, \ldots, k$ **do**
    Let $X^{i,s} = [\mathbf{v}^t]_{t:z^{t,s}=i}$ and $\mathbf{y}^{i,s} = [r^t]_{t:z^{t,s}=i}$ //Handy shorthand
    $\mathbf{u}^{i,s} = \left(X^{i,s}(X^{i,s})^\top + I_d\right)^{-1} X^{i,s}\mathbf{y}^{i,s}$ //Update member user models
  end for
end for