CS771 End-semester Examination

RAKTIM MITRA

TOTAL POINTS

91.5 / 120

QUESTION 1

1 Problem 1 11 / 12

- 1 1
- 1 2
- 1 3
- 1 4
- 1 5
✓ - 1 6
- 1 7
- 1 8
- 1 9
- 1 10
- 1 11
- 1 12
✓ + 12 correct

QUESTION 2

Problem 2 24 pts

2.1 Problem 2.1 4 / 4

✓ + 4 Correct
+ 0 Incorrect

2.2 Problem 2.2 4 / 4

✓ + 4 Correct
+ 0 Incorrect

2.3 Problem 2.3 1 / 4

+ 4 Correct
+ 0 Incorrect
+ 1 Point adjustment

2.4 Problem 2.4 3 / 4

+ 0 Incorrect / Unattempted
+ 2 Expression left only as expansion
✓ + 3 Expression with K(x,x) = x
+ 4 Correct answer as L1 norm

2.5 Problem 2.5 4 / 4

+ 0 Incorrect / Not attempted
+ 2 Partially correct expression
✓ + 4 Full expression

2.6 Problem 2.6 4 / 4

+ 0 Incorrect / Unattempted
✓ + 2 Partially correct / Correct expression for earlier part
✓ + 2 Complete solution

QUESTION 3

Problem 3 36 pts

3.1 Problem 3.1 6 / 6

✓ + 2 Chain rule
✓ + 2 expression for df/dr
✓ + 2 final answer
+ 0 incorrect

3.2 Problem 3.2 6 / 6

✓ + 2 values of K
✓ + 2 values of distance
✓ + 2 Example
+ 0 incorrect

3.3 Problem 3.3 6 / 6

✓ + 4 Correct Answer for Matrix M
✓ + 2 Correct Answer for vector W
+ 0 No/Wrong Answer

3.4 Problem 3.4 2 / 6

✓ + 1 Correct expression for Gradient
✓ + 1 Correct expression for Hessian
+ 2 Correctly Applying Newton Method
+ 2 Correct Expression for W1,W2,W3
+ 1 Partial Answer (like not applying newton method properly or not writing expression for all W1,W2,W3 properly)
+ 0 No/ Wrong Answer

3.5 Problem 3.5 4.5 / 6

+ 0 Not attempted
+ 1 Gaussian prior, but incorrect parameters or constants
✓ + 1.5 Correct Prior
✓ + 2 correct likelihood with wrong constant, when error > epsilon
✓ + 1 partially correct likelihood (wrong constant), when error < epsilon
+ 1 one side correct likelihood when error> epsilon
+ 3 Correct likelihood, when error > epsilon
+ 1.5 Correct likelihood when error < epsilon
+ 0 Incorrect
- 0.5 Square missing
- 0.5 minus missing

3.6 Problem 3.6 5 / 6

✓ + 6 Correct
+ 0 Not attempted
+ 0 Incorrect
✓ - 1 w0 ignored

QUESTION 4

Problem 4 24 pts

4.1 Problem 4.1 10 / 12

✓ + 4 Correct P [y l , x , w, σ i]
✓ + 4 Correct P σ i l y i , x i , w]
✓ + 2 Correct Map solution expression
+ 2 Correct Map solution
✓ + 0 Not Attempted/ Incorrect
+ 1 Map Solution half correct

4.2 Problem 4.2 5 / 6

✓ + 6 Correct
+ 0 Not Attempted
+ 0 Incorrect
+ 2 Correct expression for posterior
+ 4 Correct Expression/Derivation for MAP
- 0.5 Incomplete/Inaccurate/Derivation steps missing
+ 1.5 Partially Correct (Has idea of posterior, formulated MAP)
+ 1 [Incomplete + right direction] Posterior step/MAP
✓ - 1 Errror in MAP/No derivation
+ 0.5 Right initial step [based on discretion specific to answer]

4.3 Problem 4.3 4 / 6

+ 6 Correct
+ 0 Not Attempted
+ 1 Partially correct [structure only]
+ 0 Incorrect
✓ + 4 Partially correct [structure]. Only one update expression correct.
- 1 Minor mistakes
- 0.5 Mistake in one update
- 1 Details missing

QUESTION 5

Problem 5 24 pts

5.1 Problem 5.1 6 / 8

+ 0 No Solution/ Wrong solution
✓ + 2 Part 1
+ 1 Part 2 Partial
✓ + 2 Part 2 complete
✓ + 2 Part 3
+ 2 Part 4
+ 1 Partial Marks

5.2 Problem 5.2 6 / 16

+ 0 Unanswered or insufficient details in answer
+ 2 Specifying the problem as a mixed regression/clustering problem
✓ + 2 Specifying latent variables
+ 8 Correct expressions for MLE/MAP estimates of latent and model variables
✓ + 4 Final algorithm
✓ + 2 Specifying the problem as a matrix completion problem
- 2 Point adjustment

Your technique gives me a separate user vector for each purchase. How do i recover the user vectors for the k members now?

Name: Raktim Mitra

Roll No.: 150562 Dept.: CSE

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

Instructions: *Total:* **120 marks**

1. This question paper contains a total of 8 pages (8 sides of paper). Please verify.
2. Write your name, roll number, department on **every side of every sheet** of this booklet.
3. Write final answers **neatly with a pen**. Pencil marks can get smudged and you may lose credit.
4. Do not give derivations/elaborate steps unless the question specifically asks you to provide these.

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.
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.
3. F — The k-means++ algorithm for clustering, initializes the cluster centers to k points in the dataset that are closest to each other.
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.
5. T — When working with large datasets, held-out validation is cheaper to execute as compared to k-fold cross validation.
6. T — The k-means++ algorithm cannot be used when performing kernel k-means clustering with a nonlinear Mercer kernel with an infinite dimensional feature map.
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.
8. F — The Power method can be used to solve the PCA problem but it cannot be used to solve the kernel PCA problem.
9. T — Solving the SVM problem is cheaper when using a linear kernel than it is when using the Gaussian kernel.
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.
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.
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.

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.

$$\phi(z) = \phi((x,y)) = \begin{bmatrix} x^2 \\ \sqrt{2}xy \\ y^2 \end{bmatrix}$$

Name: Raktim Mitra

Roll No.: 150562 Dept.: CSE

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

---

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?

I should agree with Dick (700 train, 300 val)
Reason: For tom, taking only 10 as validation may not capture all aspects of dataset, hence results may be faulty.
For Harry, only 10 training data will be very low for training and the model will surely heavily underfit.

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?

The Binary Classifier is an weak learner. So, Most prudent thing to do would be applying AdaBoost technique to improve its accuracy.

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 $\|\phi_{\text{int}}(\mathbf{x}) - \phi_{\text{int}}(\mathbf{y})\|_2^2$.

$\sum_{i=1}^{d} (x_i + y_i - 2\min\{x_i, y_i\})$

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 $\hat{\mathbf{w}}$.

$\Phi = \begin{bmatrix} \hat{\phi}(x^1) \\ \hat{\phi}(x^2) \\ \vdots \\ \hat{\phi}(x^n) \end{bmatrix} = G = G^T$ Hence, $\hat{w} = (G^T G + \lambda I_n)^{-1} G^T y = (G^2 + \lambda I_n)^{-1} G y$

Name: Raktim Mitra

Roll No.: 150562 Dept.: CSE

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 $\hat{\mathbf{W}} = \arg\min_{\mathbf{W} \in \mathcal{H}} \lambda \cdot \|\mathbf{W}\|_{\mathcal{H}}^2 + \sum_{i=1}^n (y^i - \langle \mathbf{W}, \phi_K(\mathbf{x}^i) \rangle)^2$, 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!

from problem 5 we got : $\hat{w}_g = (G^2 + \lambda I_n)^{-1} G y$

If $\lambda = 0$ and G is invertible, $\hat{w} = (G^2)^{-1} G y$

$= (GG)^{-1} G y = G^{-1} G^{-1} G y$

$= G^{-1} y \quad — (1)$

Here we get,

$\delta = (G + \lambda I_n)^{-1} y$

as $\lambda \Rightarrow 0$ and G invertible $\Rightarrow \delta = G^{-1} y \quad — (2)$

from 1 and 2 $\gamma_i = \delta_i \ \forall i \in [n]$ since $\gamma_i = \hat{w}_i$ and $\hat{w} = \delta$.

[Proved]

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 $\frac{df}{dr}$, and also evaluate $\frac{df}{dr}$ at $r = 2$.

Let, $p = f(z) = z_1 X + z_2 Y \qquad f : \mathbb{R}^2 \to \mathbb{R}^2$

$z = g(r) = [r^2, r^3] \qquad g : \mathbb{R} \to \mathbb{R}^2$

then, $\frac{df}{dr} = \frac{dp}{dr} = J^f . J^g \in \mathbb{R}^{2\times 1}$

$J^f = \begin{bmatrix} \frac{\partial f}{\partial z_1} & \frac{\partial f}{\partial z_2} \end{bmatrix} = \begin{bmatrix} x & Y \end{bmatrix} = \begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix}$

$J^g = \begin{bmatrix} \frac{\partial g_1}{\partial r} \\ \frac{\partial g_2}{\partial r} \end{bmatrix} = \begin{bmatrix} 2r \\ 3r^2 \end{bmatrix}$

hence, $\frac{df}{dr} = J^f J^g = \begin{bmatrix} 2r + 6r^2 \\ 2r + 3r^2 \end{bmatrix}$

at $r = 2$, $\frac{df}{dr} = \begin{bmatrix} 4 + 24 \\ 4 + 12 \end{bmatrix} = \begin{bmatrix} 28 \\ 16 \end{bmatrix}$

Name: Raktim Mitra

Roll No.: 150562 Dept.: CSE

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

---

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.

$\phi_K(X) : \mathbb{R}^2 \to \mathbb{R}^2$ linear. $\phi_K(X) = X$ so, $K(X,Y) = X^T Y$

$X = [0,1]^T \quad Y = [1,0]^T \quad Z = [2,2]^T$

$K(X,Y) = X^T Y = 0 \qquad \|\phi_K(X) - \phi_K(Y)\| \qquad \|\phi_K(X) - \phi_K(Z)\|$

$K(X,Z) = X^T Z = 2 \qquad = \|X - Y\| \qquad = \|X - Z\|$

$K(X,Y) < K(X,Z) \qquad = \sqrt{(-1)^2 + 1^2} \qquad = \sqrt{(-2)^2 + (-1)^2}$

$= \sqrt{2} \qquad = \sqrt{5}$

$\sqrt{2} < \sqrt{5} \Rightarrow \|\phi_K(X) - \phi_K(Y)\| < \|\phi_K(X) - \phi_K(Z)\|$

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.

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

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.

$g^t = \nabla f(w^t) + \nabla(r(w^t)) = \sum_{i=1}^{n} (\langle w^t, x^i \rangle - y^i) x^i + \cancel{8} W^t$ $\qquad (G)^T x^i) x^i \; d$

$H^t = \nabla^2 f(w^t) + \nabla^2(r(w^t)) = \sum_{i=1}^{n} x^i x^{iT} + I_{d\times d} = XX^T + I_d$

$W^0 = 0,$

$W^1 = 0 + \sum_{i=1}^{n} (x^i)^T (\langle W^t, x^i \rangle - y^i) W + W^T W$

Name: Raktim Mitra

Roll No.: 150562 Dept.: CSE

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 $\mathbb{P}[y^i \mid \mathbf{x}^i, \mathbf{w}]$ 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.

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

$$\mathbb{P}[y^i | x^i, w] = \begin{cases} e^{-c_1(|y^i - w^T x^i| - \epsilon)^2} & \text{if } |y^i - w^T x^i| > \epsilon \\ c_2 & \text{if } |y^i - w^T x^i| \leq \epsilon \end{cases}$$

c_1, c_2 constants determined by normalization.

$$\mathbb{P}[w] = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-\frac{\|w\|^2}{2\sigma^2}}$$

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 $\eta > 0$.

Let, initial w^0. We choose $\eta_1 \neq \eta_2$ $\eta_1, \eta_2 > 0$ and train perceptron for both.

~~Let~~ Base Case: Since w^0 is same 1st mistake will occur at same point for both.

Hypothesis: Let, 1st k mistakes are all same for both η_1 and η_2 ~~of~~ which is upon seeing $t-1$ data points.

Induction: then, $w_1^t = w_0 + \eta_1 \sum_{i=1}^{k} y^i x^i$ $\quad w_2^t = w_0 + \eta_2 \sum_{i=1}^{k} y^i x^i$

~~Since, $\eta_1, \eta_2 > 0$~~ t^{th} point comes.

$$\langle w_1^t, x^t \rangle = w_0^T x^t + \eta_1 \sum_{i=1}^{k} y^i x^{iT} x^t$$

$$\langle w_2^t, x^t \rangle = w_0^T x^t + \eta_2 \sum_{i=1}^{k} y^i x^{iT} x^t$$

We ~~know, $w_0^T x^k + \eta_1 \sum_{i=1}^{k-1} y^i x^{iT} x^k$ had same sign~~. Since, $w_0^T x^t$ part is same, and $\eta_1, \eta_2 > 0$ constant surely, $\langle w_1^t, x^t \rangle, \langle w_2^t, x^t \rangle$ have same sign. I.e. they will both either be wrong or right.

Hence, by induction proved, for all $\eta > 0$ same thing happens.

Name: Raktim Mitra

Roll No.: 150562 Dept.: CSE

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

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 $X = [\mathbf{x}^1, \ldots, \mathbf{x}^n], \mathbf{y} = [y^1, \ldots, y^n], \Sigma = \mathrm{diag}(\sigma_1^2, \ldots, \sigma_n^2)$ 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.

$\mathbb{P}[\sigma_i | y^i, x^i, w] = \dfrac{\mathbb{P}[y^i | x^i, \sigma_i, w]\, \mathbb{P}[\sigma_i]}{\mathbb{P}[y^i | x^i, w]}$ ← does not depend on σ_i → $\mathbb{P}[\sigma_i | Y^i, x^i, w]$

$= \mathbb{P}[Y^i | x^i, \sigma_i, w]$ $\sigma_i \in (0,1)$ (using given $\mathbb{P}[\sigma_i]$)

~~$\mathbb{P}[Y | X, \Sigma, w] = \left(\prod_{i=1}^{n} \mathbb{P}[y^i | x^i, \sigma_i, w]\right)\left(\prod_{i=1}^{n} \mathbb{P}[\sigma_i]\right)$~~

So for MAP, $\underset{\sigma_i}{\text{Arg Max}} \log \mathbb{P}[y^i | x^i, \sigma_i, w] + \log \mathbb{P}[\sigma_i]$ → Can remove this since $\log 1 = 0$, $\log 0 \to -\infty$

$\Rightarrow \underset{\sigma_i}{\text{Arg Max}} \log \mathbb{P}[y^i | x^i, \sigma_i, w]$

$= \underset{\sigma_i}{\arg\max} \log\left(\dfrac{1}{\sqrt{2\pi}\,\sigma_i} e^{-\frac{(y^i - \langle w, x^i \rangle)^2}{2\sigma_i^2}}\right)$

$= \underset{\sigma_i}{\arg\min} \left(c_1 \sigma_i + \dfrac{c}{2\sigma_i^2}\right)$ ⇒ derivative ⇒ $c_1 - \dfrac{c}{\sigma_i^3} = 0$

$\Rightarrow \sigma_i = \left(\dfrac{c}{c_1}\right)^{1/3}$ $\dfrac{c}{c_1} = \dfrac{(y^i - \langle w, x^i \rangle)^2}{\sqrt{2\pi}}$

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.

$\mathbb{P}[w | y^i, x^i, \sigma_i] = \dfrac{\mathbb{P}[y^i | w, x^i, \sigma_i]\, \mathbb{P}[w]}{\mathbb{P}[Y^i]}$ ← Ignore. $= \dfrac{1}{\sqrt{2\pi}\,\sigma_i} e^{-\frac{(y^i - \langle w, x^i \rangle)^2}{2\sigma_i^2}} \cdot \dfrac{1}{\sqrt{(2\pi)^d}} e^{-\frac{1}{2}\|w\|_2^2}$

Map of w

$\underset{w}{\arg\max}\, \mathbb{P}[w | y, X, \Sigma] = \underset{w}{\arg\max}\, \mathbb{P}[y | w, X, \Sigma] \cdot \mathbb{P}[w]$

$= \underset{w}{\arg\max} \left(\prod_{i=1}^{n} \mathbb{P}[y^i | w, X, \Sigma]\right) \cdot \mathbb{P}[w]$

$= \underset{w}{\arg\max} \dfrac{1}{(2\pi)^d |\Sigma|} e^{-\frac{1}{2}(Y - Xw)^T \Sigma^{-1} (Y - Xw)} \cdot \dfrac{1}{\sqrt{(2\pi)^d}} e^{-\frac{1}{2}\|w\|_2^2}$

$\Rightarrow \underset{w}{\arg\max} \log \mathbb{P}[w | y, X, \Sigma]$

$= \underset{w}{\arg\min} \dfrac{1}{2}(Y - Xw)^T \Sigma^{-1} (Y - Xw) + \frac{1}{2}\|w\|_2^2$

Solving, $w = (X^T \Sigma^{-1} X + I_n)^{-1} X^T \Sigma^{-1} Y$.

I have taken $X \in \mathbb{R}^{n \times d}$, i.e. $X = \begin{bmatrix} x^1 \\ x^2 \\ \vdots \\ x^n \end{bmatrix}$

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.

Name: Raktim Mitra

Roll No.: 150562 Dept.: CSE

1. Init Σ^0, W^0. $\Sigma^0 = \text{diag}(\sigma_1^0, \sigma_2^0, \dots, \sigma_n^0)$
2. for $i = 1, \dots, t$
   1. update Σ^t as: $\sigma_i^t = \arg\min_{\sigma_i} \mathbb{P}[\sigma_i \mid y^i, x^i, W^{t-1}]$
   2. update W^t as: $W^t = \arg\min_{W} \mathbb{P}[W \mid y, X, \Sigma^t]$
      $= (X^T \Sigma^{t-1} X + I_n)^{-1} X^T \Sigma^{t-1} Y$
3. Repeat until convergence.

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 $i = 1, \dots, n$. 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 $X = [\mathbf{x}^1, \dots, \mathbf{x}^n] \in \mathbb{R}^{d\times n}$ and $H = X^\top R^{-1} X \in \mathbb{R}^{n\times n}$ i.e. $H_{ij} = (\mathbf{x}^i)^\top R^{-1}\mathbf{x}^j$.

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

1. $\mathcal{L}(W, r, \alpha) = \frac{1}{2} W^T R W + \frac{1}{2}\|r\|^2 + \sum_{i=1}^{n} \alpha_i (y^i - \langle W, x^i \rangle - r_i)$
2. Simplified dual: $\max_{\alpha \in \mathbb{R}^n} \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j \underbrace{x^{iT} R^{-1} x^j}_{H_{ij}} + \frac{1}{2}\|\alpha\|^2 - \alpha^T y$
   $= \max_{\alpha \in \mathbb{R}^n} \frac{1}{2} \alpha^T H \alpha + \alpha^T \alpha - 2\alpha^T y = \max_{\alpha \in \mathbb{R}^n} \alpha^T (H + I_n)\alpha - 2\alpha^T y$
3. Solved $\alpha = 2(H + I_n)^{-1} y$
4. $\hat{W} = \sum_{i=1}^{n} \alpha_i R^{-1} x^i$ α_i is ith row of $2(H + I_n)^{-1} y$

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 $\mathbf{u}^1, \dots, \mathbf{u}^k \in \mathbb{R}^d$. 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.

Name: Raktim Mitra

Roll No.: 150562 Dept.: CSE

We will estimate this via matrix completion of

1st to ~~es~~ estimate for each ith purchase which member bought.

$\mathbb{P}[u^i | v^i, r^i] \cancel{\Rightarrow \mathbb{P}[r^i | u^i, v^i]\, \mathbb{P}[u^i] / \mathbb{P}[r^i | v^i]}$

$\Rightarrow$ Latent variable is u^i for each purchase $[v^i, r^i]$. $\forall i \in [n]$

$\Rightarrow$ Observed variable is v^i, r^i for $i = 1$ to n.

we want to compute UV^T. ~~where~~ (Matrix completion).

1st, Estimate u^i for each (v^i, r^i)

$$\mathbb{P}[u^j | v^i, r^i] = \mathbb{P}[r^i | u^j, v^i]\, \mathbb{P}[u^j] / \underbrace{\mathbb{P}[r^i | v^i]}_{\to \text{ignore}}$$

use Map: $\longrightarrow$ $\langle u, v \rangle + \epsilon \sim \mathcal{N}(0,1)$ prior $\longrightarrow$ take uniform.

So, $\arg\max_{\substack{u^j \in U \\ j \in [k]}} \cancel{\mathbb{P}[r^i | u^j, v^i]}$ $\mathbb{P}[r^i | u^j, v^i]$ gives u^j for all (v^i, r^i) $\forall i \in [n]$

Now, we have $U \times V$ matrix where we have ratings $u_i v_j$.

Now, by AltMin of LRMC:

We can compute low rank UV^T and hence the matrix.