CS771 Mid-sem Exam

BHAVY KHATRI

TOTAL POINTS

42.5 / 80

QUESTION 1

1 True/False 6 / 10

✓ **+ 1 pts** **Q1 correct**

✓ **+ 1 pts** **Q2 correct**

✓ **+ 1 pts** **Q3 correct**

+ 1 pts Q4 correct

✓ **+ 1 pts** **Q6 correct**

✓ **+ 1 pts** **Q5 correct**

+ 1 pts Q7 correct

✓ **+ 1 pts** **Q8 correct**

+ 1 pts Q9 correct

+ 1 pts Q10 correct

QUESTION 2

2 S2Q1 2 / 3

+ 3 pts Correct values for z's and mu's in each iteration. Says that the algorithm has converged

✓ **+ 2 pts** **Some minor errors in calculations but the overall approach is correct.**

+ 0 pts Not attempted, or wrong answer.

QUESTION 3

3 S2Q2 3 / 3

+ 0.5 pts Puts linear SVM first

+ 0.5 pts Puts kernelized Perceptron last

✓ **+ 1.5 pts** **All correctly ranked**

✓ **+ 1 pts** **Justification part mentions support vectors (sparsity of the alpha's) and says that kernelized SVM would be faster than kernelized Perceptron due to this.**

✓ **+ 0.5 pts** **Overall justification also correct**

+ 0 pts Not attempted, or incorrect answer.

QUESTION 4

4 S2Q3 3 / 3

✓ **+ 1 pts** **Mentions that we will need L binary classifiers, or some other approach that makes some sense (but must use binary classifiers only).**

✓ **+ 2 pts** **Clearly mentions what each of the L binary classifiers does.**

+ 1 pts Mentions that we can represent output vector using 2^L classes and learn 2^L binary classifiers (one for each class).

+ 2 pts Clearly mentions what each of 2^L binary classifier does.

+ 0 pts Wrong approach

QUESTION 5

5 S2Q4 1 / 3

+ 1 pts The answer clearly reflects that the number of features to be tested at levels 1, 2, 3, .. is D, D-1, D-2, ...

✓ **+ 1 pts** **The answer clearly reflects that the number of IG calculations at levels 1, 2, 3, 4, ... is 1, 2, 4, 8, and so on.**

+ 1 pts Overall expression also correct, i.e., of the form D+2*(D-1) + 2^2*(D-1) + ...

+ 0 pts Not attempted

QUESTION 6

6 S2Q5 1 / 3

+ 2 pts Correct feature mapping.

+ 1 pts Correct weights of the corresponding linear model.

- 0.5 pts Some calculation mistakes

+ 0 pts Not answered or incorrect.

+ 1 **Point adjustment**

Tried to expand the expression but solution not fully correct. Weights not specified either.

QUESTION 7

7 S2Q6 3 / 3

✓ **+ 1 pts Correct computational cost for OVA**

✓ **+ 1 pts Correct computational cost for AVA**

✓ **+ 1 pts Correct comparison + shows that no value of K makes OVA faster.**

+ 0 pts Incorrect

- 0.5 pts Some calculation mistakes

+ 0.5 pts Partially correct AVA rxpression

+ 0.5 pts Partially correct OVA rxpression

+ 0.5 pts Comparison is partially correct

QUESTION 8

8 S2Q7 2.5 / 3

✓ **+ 3 pts Using SVDD inequality, gets the expression of OCSVM inequality and mentions correct expression for \tau**

+ 3 pts Using a more elaborate argument (e.g., construting Lagrangian etc), somehow, gets to the correct solution.

+ 1 pts Some partial attempt but doesn't solve the problem fully

+ 0 pts Incorrect/ not attempted

✓ **- 0.5 pts Minor calculation mistake/ steps missing**

+ 2 pts Right approach gone awry/ reasoning unclear

QUESTION 9

9 S2Q8 3 / 3

✓ **+ 2 pts Writes the problem as a sum of two squared losses and says that the second term is like an L2 regularizer.**

+ 1 pts Writes the problem as sum of two squared losses (one on real data, one on fake data)

+ 1 pts Writes the correct solution for the optimal w and recognizes that the fake data terms is like the L2 regularizer on w.

✓ **+ 1 pts Gives the correct expression for \tilde{X} and \tilde{y}**

- 0.5 pts Doesn't mention what \tilde{y} is

+ 0 pts Not answered or totally incorrect

QUESTION 10

10 S3Q1 0 / 6

+ 2 pts Uses KKT condition and uses the fact that \alpha_s will be nonzero for the support vector and the [1 - y_s(w'x_s+b)] term will be zero.

+ 2 pts Uses the above fact and tries to get the expression for b

+ 2 pts Gets the correct expression for b

+ 1 pts Minor mistakes for part (3) of the rubric but shows some effort.

✓ **+ 0 pts Did not answer/Answered Incorrectly**

QUESTION 11

11 S3Q2 3 / 6

✓ **+ 1 pts Writing the absolute loss as squared loss divided by the absolute loss**

+ 1 pts Explanation of why expression of s_n makes sense

✓ **+ 2 pts Writing the objective as a weighted least squared problem**

+ 2 pts Alternating optimization that clearly mentions that s_n's will be estimated given w and w will be estimated given s_n's.

+ 0 pts Not attempted/Wrong

QUESTION 12

12 S3Q3 6 / 6

✓ **+ 6 pts Obtained correct expression with proper steps**

+ 4.5 pts Obtained correct expression but lack all the steps

+ 3 pts Mentions that y=0 when m=0 and all the m_k's are zero and tries to find p(y=0|λ_1, . . . , λ_K), to get p(y=1|λ_1, . . . , λ_K).

+ 2 pts Mentions that y=0 when m=0 and tries to find p(y=0|λ_1, . . . , λ_K), to get p(y=1|λ_1, . . . , λ_K)

+ 1 pts Tries to find p(y=1|λ_1, . . . , λ_K) directly, and the steps are correct till p(y=1|λ_1, . . . , λ_K) = p(m > 0|λ_1, . . . , λ_K)

+ 1 pts Tries to find p(y=1|λ_1, . . . , λ_K) directly, and the steps are correct till p(y=1|λ_1, . . . , λ_K) = p(\sum_{k=1}^K m_k > 0|λ_1, . . . , λ_K)

+ 0 pts Not attempted or Incorrect

+ **0 pts** Not clearly stated about: "sum of all m_k's = 0 iff all m_k's=0" and directly writes p(Sum of m_k=0) = Product of p(m_k=0).

- **0.5 pts** Some argument is wrong or lack of clarity in steps

QUESTION 13

13 S3Q4 0 / 6

+ **1 pts** Mentions how z_n is updated

+ **1 pts** Correct (stochastic) gradient w.r.t. \mu_k

+ **2 pts** Correct use of gradient in getting the update equation of \mu_k

+ **1 pts** Mentions that \mu_k will move towards x_n

+ **1 pts** Mentions that none of the other means move.

✓ + **0 pts** **No marks**

QUESTION 14

14 S3Q5 5 / 6

✓ + **1 pts** **Writes the posterior as a product of gamma prior and gaussian likelihood**

✓ + **1 pts** **Splits the likelihood into product of N terms**

✓ + **2 pts** **Simplifies the product and brings it to the form of gamma distribution**

✓ + **2 pts** **Clearly says that the posterior is available in closed form, and says that the posterior is a gamma (mentioning the conjugacy is not necessary)**

+ **0 pts** Unattempted or incorrect

- **1 pts** Incorrectly identifies distribution or doesn't mention it

- 1 Point adjustment

Incorrect expressions

QUESTION 15

15 S3Q6 4 / 6

✓ + **1 pts** **Correct Gradient**

✓ + **1 pts** **Correct Hessian**

✓ + **1 pts** **Correct Newton's step**

✓ + **1 pts** **Fully simplified and correct form of final update and shows that the Newton method converges in 1 iteration.**

+ **2 pts** Correctly mentions why Newton's method converges in one iteration (basically, a quadratic function)

+ **0 pts** Incorrect or incomplete or not attempted

QUESTION 16

16 S4Q1 0 / 10

+ **3 pts** Correctly writing the objective with the constraint

+ **1 pts** Correctly writing the Lagrangian

+ **2 pts** Taking derivative of Lagrangian w.r.t. w and solving for it

+ **2 pts** Correctly solving for alpha and getting the final expression for w using the optimal value of alpha

+ **1 pts** Partially correct solution for alpha with some minor mistakes

+ **2 pts** Verifying that the new w satisfies the constraint

✓ + **0 pts** **Not attempted or incorrect.**

- **1 pts** Using sum over N instead of a single example

Name: BHAVY KHATRI

Roll No.: 150186 Dept.: MTH

IIT Kanpur
CS771 Intro to ML
Mid-semester Examination
Date: September 20, 2018

Total: **80 marks**

Instructions:

1. This question paper contains a total of 8 pages (8 sides of paper). Please verify.
2. Please 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. **Important:** Please do not give derivations/elaborate steps unless specifically asked for it. Feel free to use standard results (e.g., solution of least squares regression) without deriving them from scratch. If needed, you may use the personal rough space for more detailed derivations.
5. The last page of the question paper lists some formulae if you need them.

Section 1 (True or False: 10 X 1 = 10 marks). For each of the following simply write **T** or **F** in the box.

1. [T] The prediction cost (time taken to predict the label for a test input) for 1-nearest neighbors classification is higher as compared to that of prototype based classification.
2. [T] A least squares regression problem with ℓ_1 norm regularizer on the weight vector $\boldsymbol{w}$ will have a globally optimal solution.
3. [F] The softmax regression based discriminative model for classification with K classes would require learning $K-1$ probability distributions.
4. [F] Changing the Perceptron update rule from $\boldsymbol{w}^{(t+1)} = \boldsymbol{w}^{(t)} + y_n \boldsymbol{x}_n$ to $\boldsymbol{w}^{(t+1)} = \boldsymbol{w}^{(t)} + \gamma y_n \boldsymbol{x}_n$ would effectively learn the same hyperplane separator.
5. [F] The size of kernel induced feature mapping ϕ of a polynomial kernel with degree $d \geq 2$ is the same for any value of d.
6. [F] If the MAP objective has a unique optima then the predictive distribution computed using the MAP estimate and computed by averaging over the full posterior will be the same.
7. [T] For linear/logistic regression, it is not possible to regularize different entries of the weight vector differently if using a zero mean Gaussian prior over the weight vector.
8. [T] Increasing the parameter C in soft-margin SVM objective $\frac{||\boldsymbol{w}||^2}{2} + C\sum_{n=1}^{N} \xi_n$ tends to increase the ℓ_2 squared norm of $\boldsymbol{w}$.
9. [T] Running a linear model on landmark based features or kernel random features would always be faster than a linear model on the original features.
10. [T] The SGD algorithm for a binary linear classification model would update the weight vector only when the current weight vector mispredicts the chosen training example.

Section 2 (8 problems: 8 x 3 = 24 marks). Write your answers precisely and concisely in the provided box.

1. Consider a data set with 5 points $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3, \boldsymbol{x}_4, \boldsymbol{x}_5\}$ in two dimensions: $\{(1,0), (-1,0), (0,1), (3,0), (3,1)\}$ Run <u>two</u> iterations of K-means with initial points at $\mu_1 = (-1,0)$ and $\mu_2 = (3,1)$. What are the assignments z_1, z_2, z_3, z_4, z_5 and the centers μ_1 and μ_2 at each iteration? Has the algorithm converged?

$\{z_1, z_2, z_3\}$ will correspond to μ_1 i.e. $z_1 = z_2 = z_3 =$ 1st Class

$\{z_4, z_5\}$ will correspond to μ_2 i.e. $z_4 = z_5 =$ 2nd class.

In both the iterations $\mu_1 = \left(0, \frac{1}{3}\right)$ & $\mu_2 = \left(\frac{3}{2}, \frac{1}{2}\right)$

Yes the algorithm has converged

Name: BHAVY KHATRI

Roll No.: 150186 Dept.: MTH

IIT Kanpur
CS771 Intro to ML
Mid-semester Examination
Date: September 20, 2018

2. Rank the following methods in terms of the ~~time~~ test-time prediction cost: linear SVM, kernelized Perceptron, and kernelized SVM (fastest first, slowest last), and briefly justify your answer.

fastest, mid, slowest = (linear SVM, Kernelized SVM, Kernelized Perceptron)

As SVM depends only on the support vectors so it is very easy to predicts as prediction will be solely based on support vectors.

3. Consider multi-label classification given training data $\{(\boldsymbol{x}_n \boldsymbol{y}_n)\}_{n=1}^N$. Here each output $\boldsymbol{y}_n \in \{0,1\}^L$, i.e., a binary vector of length L. Briefly describe an approach to learn a multi-label classification model using this data if you only have access to a *binary classification* algorithm $\mathcal{B}$ (e.g., a binary SVM).

Use Binary classification Algorithm B, L times to classify each y_i, $i = 1, 2, \ldots L$. For each (x_n, y_i) pair learn the necessary weight vector (or function).

4. Consider learning a decision tree, given some training data where each input has D binary-valued features. Let's assume that we will not test any feature that has been tested at one of the previous levels (but we can possibly test a feature at multiple nodes at the same level). How many information gain calculations would be needed to construct the full decision tree (i.e., assuming no pruning)? Just give the basic expression; no need to try simplifying it too much to get a more compact expression.

Suppose at each node we split it into K children., for simplicity assume $K=2$

Then total no. of nodes $= 1 + 2 + \ldots 2^{D-1} = 2^D - 1 = O(2^D)$

5. Consider a binary classification dataset with two-dimensional inputs, where each input is of the form $\boldsymbol{x} = (x_1, x_2)$. Suppose the decision boundary is given by $\frac{(x_1-1)^2}{2} + \frac{(x_2-2)^2}{3} = 1$. Note that this is a nonlinear boundary (equation of an ellipse). Write down a mapping $\phi(\boldsymbol{x})$ that will make it possible to separate the two classes using a linear separator. Also write down the weights of this linear model.

$$\frac{x_1^2}{2} - x_1 + \frac{x_2^2}{3} + \frac{4}{3} + \frac{1}{2} - \frac{4x_2}{3} - 1$$

$$\phi : \mathbb{R}^2 \longrightarrow \mathbb{R}^3.$$

$$\phi(x_1, x_2) = [x_1, x_2, x_1^2 + x_2^2]$$

Name: BHAVY KHATRI

Roll No.: 150186 Dept.: MTH

IIT Kanpur
CS771 Intro to ML
Mid-semester Examination
Date: September 20, 2018

---

6. Consider solving multi-class classification by reducing it to binary SVM classification problems using One-vs-All (OvA) and All-Pairs (note: all-pairs is also called "All vs All" or AvA). Suppose we have K classes and M examples per class ($N = KM$ examples total). Typical binary SVM takes roughly N^2 time to train on N examples. For what values of K is it computationally cheaper to use OVA instead of AVA?

Each OVA takes $O(K^2M^2)$ time, Total K classes. Total OVA Time = $O(K^3M^2)$

Each AVA takes $O(M^2)$, Total $O(K^2)$ pairs Total AvA time = $O(K^2M^2)$

More precisely Total OVA time = $K^2M^2 \times K = K^3M^2$

Total AVA time = $2M^2 \times \frac{K(K-1)}{2} = (K^2-K)M^2$.

For no value of k, OVA is computationally cheaper than AVA.

7. Consider the hard-margin versions of the SVDD problem (left) and the one-class SVM problem (right).

$$\min_{\boldsymbol{c}\in\mathbb{R}^D, R\in\mathbb{R}} R^2 \qquad \min_{\boldsymbol{c}\in\mathbb{R}^D, \tau\in\mathbb{R}} \frac{||\boldsymbol{c}||^2}{2} - \tau$$

$$\text{subject to } ||\boldsymbol{x}_n - \boldsymbol{c}||^2 \leq R^2 \qquad \text{subject to } \boldsymbol{c}^\top \boldsymbol{x}_n \geq \tau + \frac{||\boldsymbol{x}_n||^2}{2}$$

Show that these two problems are equivalent for some value of τ. What is that value τ?

note that $(x_n - c)^T(x_n - c) \leq R^2 \Rightarrow \frac{x_n^T x_n}{2} + \frac{c^T c}{2} - R^2 \leq c^T x_n$.

taking $\boxed{\tau = \frac{\|c\|^2}{2} - R^2}$ will change the ~~value of~~ 2nd problem into first problem.

8. Consider a least squares regression problem with N examples $(\mathbf{X}, \boldsymbol{y}) = \{(\boldsymbol{x}_n, y_n)\}_{n=1}^N$ and regression weight vector $\boldsymbol{w} \in \mathbb{R}^D$. Assume no regularization on $\boldsymbol{w}$. However, suppose we add another M "fake" examples $(\tilde{\mathbf{X}}, \tilde{\boldsymbol{y}}) = \{(\tilde{\boldsymbol{x}}_m, \tilde{y}_m)\}_{m=1}^M$. Show that using these additional fake examples is equivalent to using an ℓ_2 regularizer. For what values of $(\tilde{\mathbf{X}}, \tilde{\boldsymbol{y}})$, it will correspond to an ℓ_2 regularizer $\lambda \boldsymbol{w}^\top \boldsymbol{w}$?

new weight vector $w' = (X^TX + \tilde{X}^T\tilde{X})^{-1}(X^Ty + \tilde{X}^T\tilde{y})$

choose $\tilde{X}$ and $\tilde{y}$ such that $\boxed{\tilde{X}^T\tilde{X} = \lambda I_D}$

and $\boxed{\tilde{X}^T\tilde{y} = 0}$

Name: BHAVY KHATRI

Roll No.: 150186 Dept.: MTH

IIT Kanpur
CS771 Intro to ML
Mid-semester Examination
Date: September 20, 2018

Section 3 (6 problems: 6 x 6 = 36 marks). Write your answers precisely and concisely in the provided box.

1. Assuming hard-margin SVM, show that, given the solution for the dual variables α_n's, the bias term $b \in \mathbb{R}$ can be computed as $b = y_s - t_s$ where s can denote the index of *any* of the support vectors, and t_s is a term that requires computing a summation defined over all the support vectors. (Hint: Use KKT conditions)

2. Show that we can rewrite regression with *absolute* loss function $|y_n - \boldsymbol{w}^\top \boldsymbol{x}_n|$ as a *reweighted* least squares objective where the squared loss term for each example $(\boldsymbol{x}_n, y_n)$ is multiplied by an importance weight $s_n > 0$. Write down the expression for s_n, and briefly explain why this expression for s_n makes intuitive sense. Given N examples $\{(\boldsymbol{x}_n, y_n\}_{n=1}^N$, briefly outline the steps of an optimization algorithm that estimates the unknowns ($\boldsymbol{w}$ and the importance weights $\{s_n\}_{n=1}^N$) for this reweighted least squares problem.

$L(w) = \sum_{n=1}^{N} |y_n - w^T x_n| = \sum_{n=1}^{N} \frac{1}{|y_n - w^T x_n|} (y_n - w^T x_n)^2 = \sum_{n=1}^{N} s_n (y_n - w^T x_n)^2$

$s_n \propto |y_n - w^T x_n| \neq 0$ // $\quad s_n = \frac{1}{|y_n - w^T x_n|}$, $s_n > 0$, $n = 1, 2 \ldots N$

s_n is the inverse of quantity of misprediction for every input.

This problem reduce to constrained optimization problem w.r.t. w, s_n.

$L(w, s_n) = \sum_{n=1}^{N} s_n (y_n - w^T x_n)^2 \iff L'(w, s_n) = \sum_{n=1}^{N} s_{n_i} \frac{(y_n - w^T x_n)^2}{} + \sum_{i=1}^{N} \alpha_i (-s_n)$

$s_n > 0 \quad n = 1, 2 \ldots N$

$-s_n < 0$ $\quad \alpha_i \geq 0$

Use lagrangian to solve this problem.

$\hat{w}_0 = \underset{\alpha_i}{\text{argmax}} \; \underset{w}{\text{argmin}} \; L'(w, s_n)$

Since $\frac{1}{|y_n - w^T x_n|}$ is

You can also

Name: BHAVY KHATRI

Roll No.: 150186 Dept.: MTH

IIT Kanpur
CS771 Intro to ML
Mid-semester Examination
Date: September 20, 2018

3. Consider the following way to generate a binary random number y: Draw K count-valued random variables $m_k \sim \text{Poisson}(\lambda_k)$, $k = 1, \ldots, K$. Define $m = \sum_{k=1}^{K} m_k$, and generate y as $y = \mathbb{I}[m \geq 0]$, where $\mathbb{I}[.]$ is indicator function. Derive the expression for $p(y = 1|\lambda_1, \ldots, \lambda_K)$.

$P(y=1|\lambda_1 -- \lambda_k) = P(m>0 \mid \lambda_1 -- \lambda_k) = 1 - P(m=0 \mid \lambda_1 -- \lambda_k)$

[Note that $m = \sum_{k=1}^{k} m_k = 0$ iff $m_k = 0 \ \forall \ k = 1, -- k$]

$P(y=1|\lambda_1 -- \lambda_k) = 1 - P(m_1 = 0, m_2 = 0 \ --- \ m_k = 0)$

$= 1 - \prod_{k=1}^{k} P(m_k = 0) = 1 - \prod_{k=1}^{k} e^{-\lambda_k}$

$P(y=1|\lambda_1 -- \lambda_k) = 1 - e^{-\sum_{k=1}^{k} \lambda_k}$ (Answer)

4. Suppose we wish to do an "online" K-means by performing an SGD-style optimization on the K-means objective $\sum_{n=1}^{N} ||\boldsymbol{x}_n - \mu_{z_n}||^2 = \sum_{n=1}^{N} \sum_{k=1}^{K} z_{nk} ||\boldsymbol{x}_n - \mu_k||^2$. Assume mini-batch size = 1, i.e., you get one randomly chosen example $\boldsymbol{x}_n$. Assume learning rate η. What will be the SGD update equations for the cluster means $\mu_1, \ldots, \mu_K$? In which direction does each mean move as a result of these SGD updates?

ame: BHAVY KHATRI

oll No.: 150186 Dept.: MTH

IIT Kanpur
CS771 Intro to ML
Mid-semester Examination
Date: September 20, 2018

5. Consider N scalar-valued observations $x_1, \ldots, x_N$ from a Gaussian $\mathcal{N}(\mu, \lambda^{-1})$. Suppose the mean μ is known and precision λ is unknown. Assume a gamma prior on λ, i.e., $p(\lambda) = \text{Gamma}(\lambda|a,b) = \frac{1}{\Gamma(a)} b^a \lambda^{a-1} \exp(-b\lambda)$. Compute the posterior distribution of λ, i.e., $p(\lambda|x_1, \ldots, x_N)$. Is the posterior available in closed form? If yes, why, and what's the name of this distribution? If no, why not? (Hint/suggestion: Try computing the posterior first, before answering yes or no).

$$P(\lambda | x_1 \ldots x_N) = \frac{P(x_1, x_2 \ldots x_n | \lambda) \times P(\lambda)}{P(x_1 \ldots x_n)} = \frac{P(x_1, x_2 \ldots x_n | \lambda) P(\lambda)}{\int P(x_1 \ldots x_n | \lambda) P(\lambda) d\lambda}$$

$$P(x_1, x_2, x_n | \lambda) = \left(\sqrt{\frac{\lambda}{2\pi}}\right)^n e^{-\lambda \sum_{n=1}^{N} (x_n - u)^2}$$

$$P(\lambda | x_1 \ldots x_N) \propto P(x_1 \ldots x_n | u) P(\lambda) \propto \lambda^{a-1} \exp(-\lambda (b + \sum_{n=1}^{N} (x_n - u)^2)$$

So Posterior follows $\Gamma(a, b + \sum_{n=1}^{N} (x_n - u)^2)$

Yes posterior is available in closed form as we can easily compute the multiplication of both terms.

6. Consider linear regression with squared loss $\mathcal{L}(\boldsymbol{w}) = \sum_{n=1}^{N} (y_n - \boldsymbol{w}^\top \boldsymbol{x}_n)^2$ (no regularizer), and apply Newton's method to find the optimal $\boldsymbol{w}$. Write down the expression for the weight update at each iteration. What is the minimum number of iterations that Newton's method will take to converge? Is that what you would expect Newton's method to do for *this* problem? If yes, why? If no, why not?

$$\nabla(f(w^{(t)}) = -2(X^T y - X^T X w) = 2(-X^T y + X^T X w)$$

$$\nabla^2(f(w^{(t)})) = 2X^T X$$

$$w^{(t+1)} = w^{(t)} - (2X^T X)^{-1} 2(-X^T y + X^T X w^{(t)})$$

$$w^{(t+1)} = (X^T X)^{-1} X^T y$$

It will converge at the end of 1 iteration. We need to compute $w^{(t+1)}$ only for the one time.

Yes this is what I expect from Newton's method because the Closed form solution of the problem is also $(X^T X)^{-1} X^T y$

Name: BHAVY KHATRI

Roll No.: 150186 Dept.: MTH

IIT Kanpur
CS771 Intro to ML
Mid-semester Examination
Date: September 20, 2018

Section 4 (1 problem: 10 marks). Write your answers precisely and concisely in the provided box.

1. The mistake-driven Perceptron update rule, after Percentron makes a mistake on $(\boldsymbol{x}_n, y_n)$, updates the weight vector as $\boldsymbol{w}^{(t)} = \boldsymbol{w}^{(t-1)} + y_n \boldsymbol{x}_n$. This update will modify the weight vector (hyperplane separator) in such a way that it becomes *less incorrect* on the example $(\boldsymbol{x}_n, y_n)$, but not necessarily *correct*. For example, if $y_n = 1$ then ${\boldsymbol{w}^{(t)}}^\top \boldsymbol{x}_n$ will become less negative than ${\boldsymbol{w}^{(t-1)}}^\top \boldsymbol{x}_n$ (but not necessarily positive). Likewise, if $y_n = -1$ then ${\boldsymbol{w}^{(t)}}^\top \boldsymbol{x}_n$ will become less positive than ${\boldsymbol{w}^{(t-1)}}^\top \boldsymbol{x}_n$ (but not necessarily negative).

   Let's try to design a variant of Perceptron that guarantees that, after the update, the new weight vector $\boldsymbol{w}^{(t)}$ will be *definitely* be correct on the example $(\boldsymbol{x}_n, y_n)$, i.e., $y_n {\boldsymbol{w}^{(t)}}^\top \boldsymbol{x}_n \geq 0$. At the same time, let's not make the new weight vector $\boldsymbol{w}^{(t)}$ drift too far from the current weight $\boldsymbol{w}^{(t-1)}$, i.e., we want to keep the squared ℓ_2 distance $||\boldsymbol{w}^{(t)} - \boldsymbol{w}^{(t-1)}||^2$ as small as possible.

   Formulate the above as an optimization problem and solve it to derive the updates for this variant of the Perceptron. Also verify that your obtained expression for $\boldsymbol{w}^{(t)}$ does satisfy the constraint $y_n {\boldsymbol{w}^{(t)}}^\top \boldsymbol{x}_n \geq 0$.

(if needed, you may continue the answer in the box on the next page)

Name: BHAVY KHATRI

Roll No.: 150186 Dept.: MTH

IIT Kanpur
CS771 Intro to ML
Mid-semester Examination
Date: September 20, 2018

Some formulae you might need

- Gaussian PDF: $\mathcal{N}(x|\mu, \lambda^{-1}) = \sqrt{\frac{\lambda}{2\pi}} \exp(-\frac{\lambda}{2}(x-\mu)^2)$
- Poisson PMF: $\text{Poisson}(x|\lambda) = \frac{\lambda^x \exp(-\lambda)}{x!}$
- Hessian of a scalar-valued function $\mathcal{L}(\boldsymbol{w})$ w.r.t. $\boldsymbol{w} \in \mathbb{R}^D$ is $\mathbf{H} = \frac{\partial^2 \mathcal{L}(\boldsymbol{w})}{\partial \boldsymbol{w} \partial \boldsymbol{w}^\top} = \frac{\partial}{\partial \boldsymbol{w}} \left[\frac{\partial \mathcal{L}(\boldsymbol{w})}{\partial \boldsymbol{w}}\right]^\top = \frac{\partial \boldsymbol{g}^\top}{\partial \boldsymbol{w}}$
- Derivatives - linear form: $\frac{\partial \boldsymbol{x}^\top \boldsymbol{a}}{\partial \boldsymbol{x}} = \frac{\partial \boldsymbol{a}^\top \boldsymbol{x}}{\partial \boldsymbol{x}} = \boldsymbol{a}$, quadratic form: $\frac{\partial}{\partial \boldsymbol{x}}(\boldsymbol{x}-\boldsymbol{s})^\top \mathbf{W}(\boldsymbol{x}-\boldsymbol{s}) = 2\mathbf{W}(\boldsymbol{x}-\boldsymbol{s})$