Introduction to ML (CS771), Autumn 2020
Indian Institute of Technology Kanpur
Homework Assignment Number 1

Student Name: Sakshi
Roll Number: 180653
Date: October 30, 2020

QUESTION 1

A vector symbol $\mathbf{b}$, a symbol in blackboard font $\mathbb{R}$, a symbol in calligraphic font $\mathcal{A}$, some colored text

Given loss function:

$$L(\mathbf{w}) = \Sigma_{n=1}^{N} |y_n - \mathbf{w}^T \mathbf{x}_n| + \lambda \|\mathbf{w}\|_1$$

Let $f(\mathbf{w}) = \Sigma_{n=1}^{N} |y_n - \mathbf{w}^T \mathbf{x}_n|$ & $g(\mathbf{w}) = \lambda \|\mathbf{w}\|_1$. Thus, by **sum rule** of sub-gradients, we have:

$$\partial L(\mathbf{w}) = \partial f(\mathbf{w}) + \partial g(\mathbf{w})$$

Consider sub-gradient of $f(\mathbf{w}) = \Sigma_{n=1}^{N} |y_n - \mathbf{w}^T \mathbf{x}_n|$:

$$\partial f(\mathbf{w}) = -\sum_{n=1}^{N} \mathbf{x}_n \partial |t|, \qquad t = y_n - \mathbf{w}^T \mathbf{x}_n$$

$$\partial |t| = \begin{cases} 1 & , y_n - \mathbf{w}^T \mathbf{x}_n > 0 \\ -1 & , y_n - \mathbf{w}^T \mathbf{x}_n < 0 \\ [-1, 1] & , y_n - \mathbf{w}^T \mathbf{x}_n = 0 \end{cases}$$

Consider sub-gradient of $g(\mathbf{w}) = \lambda \|\mathbf{w}\|_1$:

$$\partial g(\mathbf{w})_d = \begin{cases} -\lambda & , \mathbf{w}_d < 0 \\ +\lambda & , \mathbf{w}_d > 0 \\ [-\lambda, \lambda] & , \mathbf{w}_d = 0 \end{cases}$$

Here, both $\partial f(\mathbf{w})$ and $\partial g(\mathbf{w})$ are D-dimensional.
Thus, $\partial L(\mathbf{w})$ can be obtained by summing $\partial f(\mathbf{w})$ and $\partial g(\mathbf{w})$ dimension-wise.

Introduction to ML (CS771), Autumn 2020
Indian Institute of Technology Kanpur
Homework Assignment Number 1

Student Name: Sakshi
Roll Number: 180653
Date: October 30, 2020

QUESTION 2

Given $\tilde{\mathbf{x}} = \mathbf{x} \circ \mathbf{m}$, and the new loss function as:

$$\begin{aligned} L(\mathbf{w}) &= \sum_{n=1}^{N} (y_n - \mathbf{w}^T \tilde{\mathbf{x}}_n)^2 \\ &= \sum_{n=1}^{N} (y_n^2 + (\mathbf{w}^T \tilde{\mathbf{x}}_n)^2 - 2y_n \mathbf{w}^T \tilde{\mathbf{x}}_n) \\ &= \sum_{n=1}^{N} (y_n^2 + (\sum_{i=1}^{D} w_i x_i m_i)^2 - 2y_n \sum_{i=1}^{D} (w_i x_i m_i)) \end{aligned}$$

Using: $E[m_i] = p$ $E[m_i^2] = p$ $\underset{i \neq j}{E}[m_i m_j] = p^2$ (independent R.V.s)

The expected value of this loss function can be found as following:

$$\begin{aligned} E[L(\mathbf{w})] &= E\big[\sum_{n=1}^{N} (y_n^2 + (\sum_{i=1}^{D} w_i x_i m_i)^2 - 2y_n \sum_{i=1}^{D} (w_i x_i m_i))\big] \\ &= \sum_{n=1}^{N} E\big[(y_n^2 + (\sum_{i=1}^{D} w_i x_i m_i)^2 - 2y_n \sum_{i=1}^{D} (w_i x_i m_i))\big] \\ &= \sum_{n=1}^{N} \big(E[y_n^2] + E[(\sum_{i=1}^{D} w_i x_i m_i)^2] - 2y_n E[\sum_{i=1}^{D} (w_i x_i m_i)]\big) \\ &= \sum_{n=1}^{N} \big(y_n^2 + E[\sum_{i=1}^{D} w_i^2 x_i^2 m_i^2 + 2\sum_{i=1}^{D} \sum_{j=i+1}^{D} (w_i x_i m_i w_j x_j m_j)] - 2y_n \sum_{i=1}^{D} (w_i x_i p)\big) \\ &= \sum_{n=1}^{N} \big(y_n^2 + \sum_{i=1}^{D} w_i^2 x_i^2 p + 2p^2 \sum_{i=1}^{D} \sum_{j=i+1}^{D} (w_i x_i w_j x_j)] - 2y_n p \sum_{i=1}^{D} (w_i x_i)\big) \\ &= \sum_{n=1}^{N} \big(y_n^2 + p\sum_{i=1}^{D} w_i^2 x_i^2 + 2p^2 \sum_{i=1}^{D} \sum_{j=i+1}^{D} (w_i x_i w_j x_j)] - 2y_n p \mathbf{w}^T \mathbf{x}_n\big) \\ &= \sum_{n=1}^{N} \big((y_n - p\mathbf{w}^T \mathbf{x}_n)^2 - p^2 (\mathbf{w}^T \mathbf{x}_n)^2 + p\sum_{i=1}^{D} w_i^2 x_i^2 + 2p^2 \sum_{i=1}^{D} \sum_{j=i+1}^{D} (w_i x_i w_j x_j)]\big) \\ &= \sum_{n=1}^{N} \big((y_n - p\mathbf{w}^T \mathbf{x}_n)^2 + p\sum_{i=1}^{D} w_i^2 x_i^2 - p^2 (\mathbf{w}^T \mathbf{x}_n)^2 + 2p^2 \sum_{i=1}^{D} \sum_{j=i+1}^{D} (w_i x_i w_j x_j)]\big) \end{aligned}$$

Using expansion for $(\mathbf{w}^T \mathbf{x}_n)^2$ for the 3^{rd} term, we get:

$$E[L(\mathbf{w})] = \sum_{n=1}^{N} \big((y_n - p\mathbf{w}^T \mathbf{x}_n)^2 + p\sum_{i=1}^{D} w_i^2 x_i^2 - p^2 \sum_{i=1}^{D} w_i^2 x_i^2\big)$$

$$
\begin{aligned}
&= \sum_{n=1}^{N} \left((y_n - p\mathbf{w}^T\mathbf{x}_n)^2 + (p - p^2) \sum_{i=1}^{D} w_i^2 x_i^2 \right) \\
&= \sum_{n=1}^{N} \left((y_n - p\mathbf{w}^T\mathbf{x}_n)^2 + (p - p^2) ||\mathbf{w}^T\mathbf{x}_n||_2^2 \right) \\
&= \sum_{n=1}^{N} \left((y_n - p\mathbf{w}^T\mathbf{x}_n)^2 + (1 - p) ||p\mathbf{w}^T\mathbf{x}_n||_2^2 \right)
\end{aligned}
$$

It can be clearly seen that in our expression of loss function, the second term is like regularizer term in loss function. Thus, this is equivalent to a regularized loss function where (1-p) equates to λ. Thus, p determines the extent of regularization in this loss function.

Introduction to ML (CS771), Autumn 2020
Indian Institute of Technology Kanpur
Homework Assignment Number 1

Student Name: Sakshi
Roll Number: 180653
Date: October 30, 2020

QUESTION 3

Given the squared loss error:

$$L[\mathbf{w}] = \sum_{n=1}^{N}\sum_{m=1}^{M}(y_{nm} - \mathbf{w}_m^T\mathbf{x}_n)^2 = TRACE[(\mathbf{Y}-\mathbf{XW})^T(\mathbf{Y}-\mathbf{XW})]$$

Defining $\mathbf{W} = \mathbf{BS}$, where $\mathbf{B}$ is $D \times K$ & $\mathbf{S}$ is $K \times M$, we have:

$$L[\mathbf{W}] = TRACE[(\mathbf{Y}-\mathbf{XBS})^T(\mathbf{Y}-\mathbf{XBS})]$$

Differentiating wrt $\mathbf{B}$ we get the following :

$$\begin{aligned}
\mathbf{B} &= \underset{\mathbf{B}}{arg\ min}\ TRACE[(\mathbf{Y}-\mathbf{XBS})^T(\mathbf{Y}-\mathbf{XBS})] \\
&= \underset{\mathbf{B}}{arg\ min}\ TRACE[\mathbf{Y}^T\mathbf{Y} - \mathbf{Y}^T\mathbf{XBS} - \mathbf{S}^T\mathbf{B}^T\mathbf{X}^T\mathbf{Y} + \mathbf{S}^T\mathbf{B}^T\mathbf{X}^T\mathbf{XBS}] \\
&\implies \frac{\partial}{\partial \mathbf{B}} TRACE[\mathbf{Y}^T\mathbf{Y} - \mathbf{Y}^T\mathbf{XBS} - \mathbf{S}^T\mathbf{B}^T\mathbf{X}^T\mathbf{Y} + \mathbf{S}^T\mathbf{B}^T\mathbf{X}^T\mathbf{XBS}] = 0 \\
&\implies 2(-\mathbf{X}^T\mathbf{YS}^T + \mathbf{X}^T\mathbf{XBSS}^T) = 0 \qquad \text{...matrix-cook book eq(101, 102, 116)} \\
&\implies \mathbf{B} = (\mathbf{X}^T\mathbf{X})^{-1}(\mathbf{X}^T\mathbf{YS}^T)(\mathbf{SS}^T)^{-1}
\end{aligned}$$

Differentiating wrt $\mathbf{S}$ we get the following :

$$\begin{aligned}
\mathbf{S} &= \underset{\mathbf{S}}{arg\ min}\ TRACE[(\mathbf{Y}-\mathbf{XBS})^T(\mathbf{Y}-\mathbf{XBS})] \\
&= \underset{\mathbf{S}}{arg\ min}\ TRACE[\mathbf{Y}^T\mathbf{Y} - \mathbf{Y}^T\mathbf{XBS} - \mathbf{S}^T\mathbf{B}^T\mathbf{X}^T\mathbf{Y} + \mathbf{S}^T\mathbf{B}^T\mathbf{X}^T\mathbf{XBS}] \\
&\implies \frac{\partial}{\partial \mathbf{S}} TRACE[\mathbf{Y}^T\mathbf{Y} - \mathbf{Y}^T\mathbf{XBS} - \mathbf{S}^T\mathbf{B}^T\mathbf{X}^T\mathbf{Y} + \mathbf{S}^T\mathbf{B}^T\mathbf{X}^T\mathbf{XBS}] = 0 \\
&\implies 2(-\mathbf{B}^T\mathbf{X}^T\mathbf{Y} + \mathbf{B}^T\mathbf{X}^T\mathbf{XBS}) = 0 \qquad \text{...matrix-cook book eq(102, 103, 116)} \\
&\implies \mathbf{S} = (\mathbf{B}^T\mathbf{X}^T\mathbf{XB})^{-1}(\mathbf{B}^T\mathbf{X}^T\mathbf{Y})
\end{aligned}$$

Taking $\mathbf{w}_1 = \mathbf{B}$ & $\mathbf{w}_2 = \mathbf{S}$, the ALT-OPT algorithm can be written as:

1. Initialize $\mathbf{S} = \mathbf{S}^{(0)}$ and $t = 0$.
2. $\mathbf{B}^{(t+1)} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\left(\mathbf{X}^T\mathbf{YS}^{(t)T}\right)\left(\mathbf{S}^{(t)}\mathbf{S}^{(t)T}\right)^{-1}$
3. $\mathbf{S}^{(t+1)} = \left(\mathbf{B}^{(t+1)T}\mathbf{X}^T\mathbf{XB}^{(t+1)}\right)^{-1}\left(\mathbf{B}^{(t+1)T}\mathbf{X}^T\mathbf{Y}\right)$
4. $t = t + 1$ & go to step 2 if not converged yet.

The sub-problems for solving $\mathbf{B}$ & $\mathbf{S}$ are not equally easy/difficult computationally. It can be clearly seen that the sub-problem for calculating $\mathbf{B}$ involves two matrix inversions of dimension $D \times D$ & $K \times K$ and thus is more difficult than calculating $\mathbf{S}$, which involves only one matrix inversion of dimension $K \times K$.

Introduction to ML (CS771), Autumn 2020
Indian Institute of Technology Kanpur
Homework Assignment Number 1

Student Name: Sakshi
Roll Number: 180653
Date: October 30, 2020

QUESTION 4

Given:

$$\hat{\mathbf{w}} = \underset{\mathbf{w}}{arg\ min}\ \frac{1}{2}(\mathbf{y}-\mathbf{X}\mathbf{w})^T(\mathbf{y}-\mathbf{X}\mathbf{w}) + \frac{\lambda}{2}\mathbf{w}^T\mathbf{w}$$

The loss function is:

$$L(\mathbf{w}) = \frac{1}{2}(\mathbf{y}-\mathbf{X}\mathbf{w})^T(\mathbf{y}-\mathbf{X}\mathbf{w}) + \frac{\lambda}{2}\mathbf{w}^T\mathbf{w}$$

Using the update equation given in the lecture slides, we have:

$$\mathbf{w}^{(t+1)} = \mathbf{w}^t - (\mathcal{H}^t(\mathbf{w}))^{-1}g(\mathbf{w})$$

where $g(\mathbf{w}) = \nabla L(\mathbf{w})$ & $\mathcal{H}(\mathbf{w}) = \nabla^2 L(\mathbf{w})$ is the Hessian matrix.

$$\begin{aligned}
L(\mathbf{w}) &= \frac{1}{2}\sum_{n=1}^{N}(y_n - \mathbf{w}^T\mathbf{x}_n)^2 + \frac{\lambda}{2}\mathbf{w}^T\mathbf{w} \\
g(\mathbf{w}) &= \frac{1}{2}\sum_{n=1}^{N}2(y_n - \mathbf{w}^T\mathbf{x}_n)(-\mathbf{x}_n) + \lambda\mathbf{w} \\
&= -\sum_{n=1}^{N}y_n\mathbf{x}_n + \sum_{n=1}^{N}\mathbf{x}_n(\mathbf{x}_n^T\mathbf{w}) + \lambda\mathbf{w} \\
&= -\mathbf{X}^T\mathbf{y} + \mathbf{X}^T\mathbf{X}\mathbf{w} + \lambda\mathbf{w} \\
&= -\mathbf{X}^T\mathbf{y} + (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})\mathbf{w} \\
\mathcal{H}(\mathbf{w}) &= \mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}
\end{aligned}$$

Placing the obtained values in the update equation, we get the Newton's update equation for each iteration as:

$$\mathbf{w}^{(t+1)} = \mathbf{w}^t - (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}(-\mathbf{X}^T\mathbf{y} + (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})\mathbf{w}^t)$$

Let $\mathbf{w}^t$ be the optimal weight vector $\mathbf{w}_{opt}$. Then $\mathbf{w}^{(t+1)} = \mathbf{w}^t$, which gives:

$$-(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}(-\mathbf{X}^T\mathbf{y} + (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})\mathbf{w}_{opt}) = 0$$

$$-\mathbf{X}^T\mathbf{y} + (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})\mathbf{w}_{opt} = 0$$

$$\mathbf{w}_{opt} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$$

It can be clearly seen that we have reached the closed form solution for $\mathbf{w}_{opt}$ in a single iteration of Newton's method. Thus, it takes only 1 iteration for Newton's method to converge.

Introduction to ML (CS771), Autumn 2020
Indian Institute of Technology Kanpur
Homework Assignment Number 1

Student Name: Sakshi
Roll Number: 180653
Date: October 30, 2020

QUESTION 5

The probability distribution for likelihood $\mathbb{P}[\mathbf{y}|\pi]$ is given as:

$$\mathbb{P}[\mathbf{y}|\pi] = \frac{N!}{\prod_{i=1}^{6} N_i!} \prod_{j=1}^{6} \pi_j^{N_j}$$

This is a Multinoulli distribution. For such likelihood, Dirichlet prior is an appropriate conjugate prior. The prior $\mathbb{P}[\pi]$ is given as:

$$\mathbb{P}[\pi] = c \times \prod_{i=1}^{6} \pi_i^{\beta_i - 1}$$

where c is a constant. The MAP solution is defined as:

$$\pi_{MAP} = \underset{\pi}{arg\ min}\Big(-\log \mathbb{P}[\mathbf{y}|\pi] - \log \mathbb{P}[\pi] \Big) \qquad given: \quad \sum_{i=1}^{6} \pi_i = 1 \quad \pi_i \geq 0$$

$$= \underset{\pi}{arg\ min}\Big(-\sum_{i=1}^{6} (N_i + \beta_i - 1) \log(\pi_i) \Big) \qquad \text{taking only } \pi \text{ dependent terms}$$

We can use Lagrange multipliers to solve this constrained optimization problem.

$$\mathcal{L}(\pi, \alpha, \lambda) = -\sum_{i=1}^{6} (N_i + \beta_i - 1) \log \pi_i - \sum_{i=1}^{6} \alpha_i \pi_i + \lambda (\sum_{i=1}^{6} \pi_i - 1)$$

The primal & dual are given as:

$$Primal: \underset{\pi}{arg\ min} \left[\underset{\alpha \geq 0, \lambda}{arg\ max}\ (\mathcal{L}(\pi, \alpha, \lambda)) \right] \qquad \text{solution for } \pi_{opt}$$

$$Dual: \underset{\alpha \geq 0, \lambda}{arg\ max} \left[\underset{\pi}{arg\ min}\ (\mathcal{L}(\pi, \alpha, \lambda)) \right] \qquad \text{solution for } \alpha_{opt}, \lambda_{opt}$$

In dual, the inner optimization is unconstrained. Thus, we solve it using 1^{st} order optimality:

$$-\frac{N_i + \beta_i - 1}{\pi_i} - \alpha_i + \lambda = 0 \qquad \text{diff. wrt } \pi_i$$

$$\implies (\pi_i)_{opt} = \frac{N_i + \beta_i - 1}{\lambda - \alpha_i}$$

Putting this value in the inner expression, we solve the outer optimization:

$$\alpha_{opt}, \lambda_{opt} = \underset{\alpha \geq 0, \lambda}{arg\ max} \left[-\sum_{i=1}^{6} (N_i + \beta_i - 1) \log \frac{N_i + \beta_i - 1}{\lambda - \alpha_i} - \sum_{i=1}^{6} \alpha_i \frac{N_i + \beta_i - 1}{\lambda - \alpha_i} + \lambda (\sum_{i=1}^{6} \frac{N_i + \beta_i - 1}{\lambda - \alpha_i} - 1) \right]$$

$$= \underset{\alpha \geq 0, \lambda}{arg\ max} \left[-\sum_{i=1}^{6} (N_i + \beta_i - 1) \log \frac{N_i + \beta_i - 1}{\lambda - \alpha_i} - \lambda \right] + \sum_{i=1}^{6} (N_i + \beta_i - 1)$$

For a particular λ_{opt}, α needs to be minimum to make $\lambda - \alpha_i \geq 0$ as large as possible.
This gives $\alpha_{opt} = \mathbf{0}$.
Now we have:

$$\lambda_{opt} = \underset{\lambda>0}{arg\ min} \left[-\sum_{i=1}^{6} (N_i + \beta_i - 1) \log \frac{N_i + \beta_i - 1}{\lambda} - \lambda \right]$$

$$\implies \left[-1 + \sum_{i=1}^{6} \frac{(N_i + \beta_i - 1)}{\lambda} \right] = 0 \qquad \text{diff. wrt } \lambda$$

$$\implies \lambda_{opt} = \sum_{i=1}^{6} (N_i + \beta_i - 1)$$

Thus, putting this value in expression of π_i, we get:

$$(\pi_i)_{MAP} = \frac{N_i + \beta_i - 1}{\sum_{i=1}^{6} (N_i + \beta_i - 1)}$$

The MAP estimate is better than MLE solution when we have a limited training data. To prevent overfitting in this case, we introduce a prior distribution on parameters (can be called pseudo-observations) which act as a regularizer to shift the estimate towards prior estimate. It can be seen as a compromise between prior estimate and MLE estimate. Also, when we have a lot of outliers in our training data, MAP solution is better(as it acts as a regularizer) and our solution is more inclined towards previous observations(prior).

The full posterior distribution is given as:

$$\mathbb{P}[\pi|\mathbf{y}] = \frac{\mathbb{P}[\mathbf{y}|\pi]\,\mathbb{P}[\pi]}{\mathbb{P}[\mathbf{y}]}$$

Clearly, $\mathbb{P}[\mathbf{y}]$ is independent of π. The posterior distribution is dependent only on the product of $\mathbb{P}[\mathbf{y}|\pi]$ & $\mathbb{P}[\pi]$. Also, because Multinoulli and Dirichlet distribution form a conjugate pair, we expect the posterior to be a Dirichlet distribution (prior distribution) as well. We have:

$$\mathbb{P}[\pi|\mathbf{y}] \propto \prod_{i=1}^{6} \pi_i^{N_i + \beta_i - 1}$$

This is a Dirichlet distribution with parameters: $\mu_i = N_i + \beta_i$.

Given only the posterior, it is not possible to obtain the MLE solution directly as we need to find π_{opt} that maximises $\mathbb{P}[\mathbf{y}|\pi]$. To find this, we need $\mathbb{P}[\pi]$ as well, i.e. the prior.
MAP solution is defined as the mode of the posterior distribution. For getting the MAP solution, we can simply take the mode of the posterior. Thus, it can be obtained directly without solving its optimization problem separately.
Thus, we can't get the MLE solution directly from the posterior distribution without solving its optimization problems separately. But, we can get the MAP solution directly from the posterior distribution.