Indian Institute of Technology Kanpur
CS771 Introduction to Machine Learning, 2017-18-a

Assignment Number: 3
Student Name: Raktim Mitra
Roll Number: 150562
Date: November 15, 2017

QUESTION 1

Part 1:

We want to prove $\theta^{MLE} \epsilon$ arg $max_{\boldsymbol{\theta} \epsilon \Theta}\ Q_{\boldsymbol{\theta}^{MLE}}(\boldsymbol{\theta})$ with $\theta^{MLE} \epsilon$ arg $max_{\boldsymbol{\theta} \epsilon \Theta}\ P[X|\boldsymbol{\Theta}]$.

$$Q_{\boldsymbol{\theta}^t}(\boldsymbol{\theta}) = \sum_{i=1}^{n} \mathbb{E}_{\mathbf{z} \sim \mathbb{P}[\mathbf{z}|\mathbf{x}^i, \boldsymbol{\theta}^t]} log \mathbb{P}[\mathbf{x}^i, \mathbf{z}|\theta]$$

We can write: (by deviding with a term independent of $\boldsymbol{\theta}$)

$$\text{arg } max_{\boldsymbol{\theta} \epsilon \Theta} \sum_{i=1}^{n} \mathbb{E}_{\mathbf{z} \sim \mathbb{P}[\mathbf{z}|\mathbf{x}^i, \boldsymbol{\theta}^t]} log \mathbb{P}[\mathbf{x}^i, \mathbf{z}|\theta]$$
$$= argmax_{\boldsymbol{\theta} \epsilon \Theta} \sum_{i=1}^{n} \mathbb{E}_{\mathbf{z} \sim \mathbb{P}[\mathbf{z}|\mathbf{x}^i, \boldsymbol{\theta}^t]} log \frac{\mathbb{P}[\mathbf{x}^i, \mathbf{z}|\theta]}{\mathbb{P}[\mathbf{z}|\mathbf{x}^i, \boldsymbol{\theta}^t]}$$

Now we can use two results derived in lecture 16:

$$logP[X|\theta^{MLE}] = \sum_{i=1}^{n} \mathbb{E}_{\mathbf{z} \sim \mathbb{P}[\mathbf{z}|\mathbf{x}^i, \boldsymbol{\theta}^{MLE}]} log \frac{\mathbb{P}[\mathbf{x}^i, \mathbf{z}|\theta^{MLE}]}{\mathbb{P}[\mathbf{z}|\mathbf{x}^i, \boldsymbol{\theta}^{MLE}]}$$

and

$$logP[X|\theta] \geq \sum_{i=1}^{n} \mathbb{E}_{\mathbf{z} \sim \mathbb{P}[\mathbf{z}|\mathbf{x}^i, \boldsymbol{\theta}^{MLE}]} log \frac{\mathbb{P}[\mathbf{x}^i, \mathbf{z}|\theta]}{\mathbb{P}[\mathbf{z}|\mathbf{x}^i, \boldsymbol{\theta}^{MLE}]} \quad \forall \theta$$

Since, θ^{MLE} maximises $log \mathbb{P}[\mathbf{x}|\theta]$

$$logP[X|\theta^{MLE}] \geq \sum_{i=1}^{n} \mathbb{E}_{\mathbf{z} \sim \mathbb{P}[\mathbf{z}|\mathbf{x}^i, \boldsymbol{\theta}^{MLE}]} log \frac{\mathbb{P}[\mathbf{x}^i, \mathbf{z}|\theta]}{\mathbb{P}[\mathbf{z}|\mathbf{x}^i, \boldsymbol{\theta}^{MLE}]} \quad \forall \theta$$

i.e.

$$logP[X|\theta^{MLE}] \geq Q_{\boldsymbol{\theta}^t}(\boldsymbol{\theta})$$

Therefore, θ^{MLE} maximises $\geq Q_{\boldsymbol{\theta}^t}(\boldsymbol{\theta})$ too. i.e. $\theta^{MLE} \epsilon$ arg $max_{\boldsymbol{\theta} \epsilon \Theta}\ P[X|\boldsymbol{\Theta}]$. [Proved]

Indian Institute of Technology Kanpur
CS771 Introduction to Machine Learning, 2017-18-a

QUESTION

2

Assignment Number: 3
Student Name: Raktim Mitra
Roll Number: 150562
Date: November 15, 2017

Part 1:

For some partition $\{\Omega_i\}_{i=1,...,n}$ of $\mathbb{R}^d$ and weights $\{\mathbf{w}^i\}_{i=1,..,n}$, $f : \mathbb{R}^d \to \mathbb{R}$ is such that $\forall \mathbf{x} \epsilon \mathbb{R}^d$

$$f(\mathbf{x}) = \sum_{i=1}^{n} \mathbb{I}\{\mathbf{x}\epsilon\Omega_i\}.\langle \mathbf{w}^i, \mathbf{x}\rangle$$

The function $g(\mathbf{x}) = c.f(\mathbf{x})$ can be written as:

$$\begin{aligned} g(\mathbf{x}) &= c\sum_{i=1}^{n} \mathbb{I}\{\mathbf{x}\epsilon\Omega_i\}.\langle \mathbf{w}^i, \mathbf{x}\rangle \\ &= \sum_{i=1}^{n} \mathbb{I}\{\mathbf{x}\epsilon\Omega_i\}.\langle c\mathbf{w}^i, \mathbf{x}\rangle \end{aligned}$$

Thus for same partition $\{\Omega_i\}_{i=1,...,n}$ of $\mathbb{R}^d$ and weights $\{\mathbf{w}'^i\}_{i=1,..,n}$, where $\mathbf{w}'^i = c\mathbf{w}^i \ \forall i$
$\forall x \epsilon \mathbb{R}^d$ $g : \mathbb{R}^d \to \mathbb{R}$ can be written as :

$$g(\mathbf{x}) = \sum_{i=1}^{n} \mathbb{I}\{\mathbf{x}\epsilon\Omega_i\}.\langle \mathbf{w}'^i, \mathbf{x}\rangle$$

which means $g(x)$ is piecewise lineaer. [Proved]

Part 2:

Let, For some partition $\{\Omega_i^f\}_{i=1,...,n}$ of $\mathbb{R}^d$ and weights $\{\mathbf{w}_f^i\}_{i=1,..,n}$, $f : \mathbb{R}^d \to \mathbb{R}$ is such that $\forall \mathbf{x} \epsilon \mathbb{R}^d$

$$f(\mathbf{x}) = \sum_{i=1}^{n} \mathbb{I}\{\mathbf{x}\epsilon\Omega_i^f\}.\langle \mathbf{w}_f^i, \mathbf{x}\rangle$$

and Let, For some partition $\{\Omega_i^g\}_{i=1,...,n}$ of $\mathbb{R}^d$ and weights $\{\mathbf{w}_g^i\}_{i=1,..,n}$, $f : \mathbb{R}^d \to \mathbb{R}$ is such that $\forall \mathbf{x} \epsilon \mathbb{R}^d$

$$g(\mathbf{x}) = \sum_{i=1}^{n} \mathbb{I}\{\mathbf{x}\epsilon\Omega_i^g\}.\langle \mathbf{w}_g^i, \mathbf{x}\rangle$$

for $\mathbf{x} \ \epsilon \ \mathbb{R}^d$ let, $h(\mathbf{x}) = f(\mathbf{x}) + g(\mathbf{x})$ the,, since $\mathbf{x}$ can belong to only one of $\Omega_i^f s$ and one of $\Omega_i^g s$, we have to add only two terms. Say, $\mathbf{x}\epsilon\Omega_k^f$ and $\mathbf{x}\epsilon\Omega_l^g$ with weights w_f^k, w_g^l repectively.

$$\begin{aligned} \text{hence, } h(\mathbf{x}) &= \langle w_f^k, \mathbf{x}\rangle + \langle w_g^l, \mathbf{x}\rangle \\ &= \langle w_f^k + w_g^l, x\rangle \end{aligned}$$

Now we construct a new partition Ω^h for $h(\mathbf{x})$ using above observation:
For each partition Ω_i^f let, S_i be the set of $\Omega_j^g s$ such that $\Omega_i^f \cap \Omega_j^g \neq \phi$. For each element Ω_j^g of S_i add set $\Omega_i^f \cap \Omega_j^g$ as Ω_{ij}^h to Ω^h.

By the construction and by the disjointeness of each partition subsets, $\mathbf{x} \ \epsilon \ \Omega_{ij}^h \leftrightarrow \mathbf{x} \ \epsilon \ \Omega_i^f \& \ \mathbf{x} \ \epsilon \ \Omega_j^g$. hence we can define new weights $\mathbf{w}_h^{ij} = \mathbf{w}_f^i + \mathbf{w}_g^j$ for all $\Omega_{ij}^h \epsilon \Omega^h$.

Hence, for partition Ω^h , weights $\{\mathbf{w}_h^{ij}\}s$ for $\mathbf{x}\epsilon\mathbb{R}^d$, $h(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}$

$$h(\mathbf{x}) = \sum_{\Omega_{ij}^h \epsilon \Omega^h} \mathbb{I}\{\mathbf{x}\epsilon\Omega_{ij}^h\}.\langle \mathbf{w}_h^{ij}, \mathbf{x}\rangle$$

Hence. $h(\mathbf{x})$ is piecewise linear. [Proved]

Part 3:

Let, For some partition $\{\Omega_i\}_{i=1,..,n}$ of $\mathbb{R}^d$ and weights $\{\mathbf{w}^i\}_{i=1,..,n}$, $f : \mathbb{R}^d \to \mathbb{R}$ is such that $\forall \mathbf{x} \epsilon \mathbb{R}^d$

$$f(\mathbf{x}) = \sum_{i=1}^{n} \mathbb{I}\{\mathbf{x}\epsilon\Omega_i\}.\langle \mathbf{w}^i, \mathbf{x}\rangle$$

now, $g(\mathbf{x}) = f_{ReLU}(f(\mathbf{x})) = max(f(\mathbf{x}), 0)$, then for some $\mathbf{x}\epsilon\mathbb{R}^d$:

$$g(\mathbf{x}) = \sum_{i=1}^{n} \mathbb{I}\{\mathbf{x}\epsilon\Omega_i\}.max(\langle \mathbf{w}^i, \mathbf{x}\rangle, 0)$$
$$= \langle \mathbf{w}^k, \mathbf{x}\rangle when \langle \mathbf{w}^k, \mathbf{x}\rangle > 0 \; and = 0 \text{ when } \langle \mathbf{w}^k, \mathbf{x}\rangle \leq 0, \; x\epsilon\Omega_k$$

From above observation we can say that by applying ReLU we are basically passing a plane $\langle \mathbf{w}^k, \mathbf{x}\rangle = 0$ through Ω_k and assiging 0 to $h(\mathbf{x})$ for $\mathbf{x}$ lying on ≤ 0 side of the plane. In other words we can make $w^k = 0$ in those regions.

Therefore we create a new partition Ω' with $2n$ subsets $\{\Omega_{i1}, \Omega_{i2}\}_{i=1,..,n}$ where Ω_{i1} corresponds to set of $\mathbf{x}$'s such that $\langle \mathbf{w}^i, \mathbf{x}\rangle > 0$ and we define $\mathbf{w}'^{i1} = \mathbf{w}^i$, Ω_{i2} corresponds to set of $\mathbf{x}$'s such that $\langle \mathbf{w}^i, \mathbf{x}\rangle \leq 0$ and we define $\mathbf{w}'^{i2} = 0$. Then, we represent these new weights as $\{\mathbf{w}'^i\}_{i=1,2,..,2n}$. i.e. $\mathbf{w}^{jk} = \mathbf{w}'^{j*k}$. and also partition Ω' as $\{\Omega'_i\}_{i=1,2,..,2n}$. i.e. $\Omega_{jk} = \Omega'_{j*k}$

Now, for partition Ω' and weights $\{w'^i\}s$ $g(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}$ can be represented as:

$$g(\mathbf{x}) = \sum_{i=1}^{2n} \mathbb{I}\{\mathbf{x}\epsilon\Omega'_i\}.\langle \mathbf{w}'^i, \mathbf{x}\rangle$$

Hence, $g(x) = f_{ReLU}(f(\mathbf{x}))$ is piecewise linear. [Proved]

Part 4:

Claim: Any neural network with a ReLU activation function computes a piecewise linear function.

Proof by Induction:

Hypothesis

In a network with only ReLU activation function, if all neuron's in i^{th} layer outputs a piecewise linear activation function then so does $i + 1^{th}$ layer of the network.

Base Case

The input nodes may or may not have ReLU activation.

If, input nodes do not have ReLU activation then we have linear functions being output from each of them (they basically output $\mathbf{x}_i$ each) i.e. they are piecewise linear too.

Else, they output f_{ReLU} on a linear function which is again piecewise linear by part 3.

Induction Step

Each neuron in $i + 1^{th}$ layer ouputs $f_{ReLU}()$ applied on weighted sum of activation outputs of neuron's in previous layer it is connected to.

let, a neuron in $i + 1^{th}$ layer is connected to m neuron's in i^{th} layer. Let, activation outputs by them are $\{f_i\}_{i=1,..,m}$ and each connection edge has $\{\mathbf{w}^i\}_{i=1,..,m}$. The neuron in $i + 1^{th}$ layer therefore outputs $f_{ReLU}(\sum_{i=1}^{m} w^i.f_i)$

Now, since each f_i is piecewise linear, then by part 1 of the problem, so is $w^i.f_i$.(constant multiplication)

Now, by part 2 of the problem, $\sum_{i=1}^{m} w^i.f_i$ is also piecewise linear (sum of piecewise linear functions)

Now, by part 3 of problem, $f_{ReLU}(\sum_{i=1}^{m} w^i.f_i)$ is piecewise linear, since $\sum_{i=1}^{m} w^i.f_i$ is piecewise linear.

Therefore the neuron in $i + 1^{th}$ layer outputs piecewise linear function. This applies for all neurons in $i + 1^{th}$ layer.

Hence, by induction the claim holds. [Proved]

Part 5:

- The network has d input nodes, D hidden layer nodes on a single hidden layer with ReLU activation and one output node with ReLU activation.
- Each hidden layer node gets a input $\epsilon\ \mathbb{R}^d$. It outputs a piecewise linear function with 2 pieces(uses one ReLU on a linear function). So, basically each hidden node outputs a function which has two pieces basically signifying a hyperplane on $\mathbb{R}^d$.
- The piecewise linear function that is sent to output node as input is weighted sum of outputs by the hidden nodes. Weight multiplication does not change partition. The sum however, changes the partition and total number of partition now can be upper bounded by maximum number of regions D (d-1) dimensional hyperplanes can create in a d dimensional space. Let L_d^D be the number of regions. This is given by the recursion

  $$L_d^D = L_d^{D-1} + L_{d-1}^{D-1}$$

  We can solve it to get

  $$L_d^D = \sum_{i=0}^{d} \binom{D}{i}$$

  . Therefore input to output node is a piecewise linear function with at most L_d^D pieces.
- The output node applies another ReLU on it. Which may at max devide each piece into 2 pieces. Therefore, maximum number of pieces possible becomes

  $$2L_d^D = 2\sum_{i=0}^{d} \binom{D}{i}$$

Indian Institute of Technology Kanpur
CS771 Introduction to Machine Learning, 2017-18-a

Assignment Number: 3
Student Name: Raktim Mitra
Roll Number: 150562
Date: November 15, 2017

QUESTION 3

Algorithm 1: Kernel Perceptron:

Input: Receive a data point $(x^t, y^t)\ \epsilon\ \chi$ at each timestamp.

Initialize α with 0 length vector. //α will be of size t at timestamp t
We store older $(\mathbf{x}^i, y^i)$s to calculate update for incoming points.
while data points are incoming **do**
  At each timestep t:
  receive $(\mathbf{x}^t, y^t)$ and increase size of α by adding another 0 coordinate.
  $\hat{y} = \text{sgn} \sum_{j=1}^{t} \alpha_j y_j K(\mathbf{x}^j, \mathbf{x}^t)$
  if $\hat{y} \neq y_i$ **then**
    $\alpha_i \leftarrow \alpha_i + 1$
  end if
end while
predict y_{test} for a testpoint x_{test} by, $\hat{y} = \text{sgn} \sum_{j=1}^{t} \alpha_j y_j K(\mathbf{x}^t, \mathbf{x}^j)$
//from dual version of the perceptron

Indian Institute of Technology Kanpur
CS771 Introduction to Machine Learning, 2017-18-a

QUESTION

4

Assignment Number: 3
Student Name: Raktim Mitra
Roll Number: 150562
Date: November 15, 2017

Part 1:

$K(\mathbf{z}^1, \mathbf{z}^2) = (\langle \mathbf{z}^1, \mathbf{z}^2 \rangle + 1)^2 = (x_1x_2 + y_1y_2 + 1)^2 = x_1^2x_2^2 + y_1^2y_2^2 + 2x_1x_2y_1y_2 + 2x_1x_2 + 2y_1y_2 + 1$

We want to represent $K(\mathbf{z}^1, \mathbf{z}^2) = \langle \phi(\mathbf{z}^1), \phi(\mathbf{z}^2) \rangle$ we take $\mathcal{H}_K \equiv \mathbb{R}^6$ because $K(\mathbf{z}^1, \mathbf{z}^2)$ has 6 terms, to represent it as a scalar product, we need vectors with 6 dimensions.

Thus our construction follows from the expression :

$$\phi(\mathbf{z}) = \phi(x, y) = [x_1^2, y_1^2, \sqrt{2}xy, \sqrt{2}x, \sqrt{2}y, 1]$$

Clearly, $K(\mathbf{z}^1, \mathbf{z}^2) = \langle \phi(\mathbf{z}^1), \phi(\mathbf{z}^2) \rangle$. Hence K is Mercer. [Proved]

Part 2:

For given $f_{(A,\mathbf{b},c)} = \langle \mathbf{z}, A\mathbf{z} \rangle + \langle \mathbf{b}, \mathbf{z} \rangle + c$ where

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

and

$$\mathbf{b} = \begin{bmatrix} b_1 & b_2 \end{bmatrix}^T$$

Then, $f_{(A,\mathbf{b},c)} = a_{11}x^2 + (a_{21} + a_{12})xy + a_{22}y^2 + b_1x + b_2y + c$

Let, $\mathbf{w} = [w_1, w_2, w_3, w_4, w_5, w_6]^T$, then:

$$\langle \mathbf{w}, \phi(\mathbf{z}) \rangle = w_1x^2 + w_2y^2 + \sqrt{2}w_3xy + \sqrt{2}w_4x + \sqrt{2}w_5y + w_6$$

comparing $f_{(A,\mathbf{b},c)} = \langle \mathbf{w}, \phi(\mathbf{z}) \rangle$
we have $w_1 = a_{11}, w_2 = a_{22}, w_3 = \frac{a_{21}+a_{12}}{\sqrt{2}}, w_4 = \frac{b_1}{\sqrt{2}}, w_5 = \frac{b_2}{\sqrt{2}}, w_6 = c$

Therefore, required $\mathbf{w}$ is $[a_{11}, a_{22}, \frac{a_{21}+a_{12}}{\sqrt{2}}, \frac{b_1}{\sqrt{2}}, \frac{b_2}{\sqrt{2}}, c]^T$ [Answer]

Part 3:

For given $\mathbf{w} = [w_1, w_2, w_3, w_4, w_5, w_6]^T$, we know:

$$\langle \mathbf{w}, \phi(\mathbf{z}) \rangle = w_1x^2 + w_2y^2 + \sqrt{2}w_3xy + \sqrt{2}w_4x + \sqrt{2}w_5y + w_6$$

We want to construct $(A, \mathbf{b}, c)$ ϵ $\mathbb{R}^{2\times2} \times \mathbb{R}^2 \times \mathbb{R}$ where,

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

and

$$\mathbf{b} = \begin{bmatrix} b_1 & b_2 \end{bmatrix}^T$$

and $f_{(A,\mathbf{b},c)} = a_{11}x^2 + (a_{21} + a_{12})xy + a_{22}y^2 + b_1x + b_2y + c$

again comparing $f_{(A,\mathbf{b},c)} = \langle \mathbf{w}, \phi(\mathbf{z}) \rangle$
we have $= a_{11} = w_1, a_{22} = w_2, b_1 = \sqrt{2}w_4, b_2 = \sqrt{2}w_5, c = w_6$, aand as can be seen $w_3 = \frac{a_{21}+a_{12}}{\sqrt{2}}$, therefore a_{21}, a_{12} can have infinitely possible values such that the relation with w_3 holds. To make things symmetrical we can take $a_{21} = a_{12} = \frac{\sqrt{2}w_3}{2} = \frac{w_3}{\sqrt{2}}$

Hence required construction :

$$A = \begin{bmatrix} w_1 & \frac{w_3}{\sqrt{2}} \\ \frac{w_3}{\sqrt{2}} & w_2 \end{bmatrix}$$

and

$$\mathbf{b} = \begin{bmatrix} \sqrt{2}w_4 & \sqrt{2}w_5 \end{bmatrix}^T$$

and

$$c = w_6$$

Part 4:

- Let, $f_{A,b,c}$ minimises least square error on the right hand side. By, part 2 of the problem it can be converted to $\langle w_r, \phi(x)$.
- The solution for Kernel Ridge regression with regularization λ let the corresponding weight be w_l. By part 3, w_l corresponds to a quadratic function too. Let, L_l denote the total error on left side(kernel ridge side).

We can see :

$$||L_i - L_r|| = \sum_{j=1}^{n} (y^j - \langle w_i, \phi(\mathbf{z}^j) \rangle)^2 - (y^j - \langle w_r, \phi(\mathbf{z}^j) \rangle)^2$$

$$= \sum_{j=1}^{n} (2y^j - \langle w_i + w_r, \phi(z^j) \rangle)(\langle w_r - w_i, \phi(z)^j \rangle)$$

$$\leq ||w_i - w_r|| \sum_{i=1}^{n} (2y^j - \langle w_i + w_r, \phi(z^j) \rangle) ||\phi(z^j)||$$

Let, $X \epsilon \mathbb{R}^{n \times 6}$ and $y \epsilon \mathbb{R}^n$ contains our ϕ mapped input vectors.
We have known results for w_l and w_r :

$$w_r = (X^T X)^{-1} X^T y$$

and

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

We can do this because ϕ is not very high dimensional.
$X^T X$ is a psd matrix and can be written as UAU^T where A is diagonal matrix with $\geq$ entries $(a1, a_2, ...)$. then we have

$$w_l - w_m = U(diag(1/a_i - 1/(a_i + \lambda))U^T X^T y$$
$$\leq \lambda U(diag(1/a_i^2))U^T X^T y$$

i.e. $||w_l - w_m|| \to 0 as \lambda \to 0$ i.e. Diffrence between two losses tends to 0. [Proved]

QUESTION 5

Assignment Number: 3
Student Name: Raktim Mitra
Roll Number: 150562
Date: November 15, 2017

$P[\mathbf{X} \mid \boldsymbol{\mu}, \mathbf{W}, \sigma] = \mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}, \mathbf{C})$ where $\mathbf{C} = \mathbf{W}\mathbf{W}^T + \sigma^2 I_d$ The data log-likelihood is given as:

$$\begin{aligned}
L(\boldsymbol{\Theta}) &= log \prod_{i=1}^{n} P(\mathbf{X}^i \mid \boldsymbol{\mu}, \mathbf{W}, \sigma) \\
&= \sum_{i=1}^{n} log\ P(\mathbf{X}^i \mid \boldsymbol{\mu}, \mathbf{W}, \sigma) \\
&= \sum_{i=1}^{n} log \frac{1}{(2\pi)^{\frac{d}{2}} |\mathbf{C}|^{\frac{1}{2}}} exp(\frac{(\mathbf{X}^i - \boldsymbol{\mu})^T \mathbf{C}^{-1} (\mathbf{X}^i - \boldsymbol{\mu})}{2}) \\
&= \sum_{i=1}^{n} -\frac{d}{2} log 2\pi - \frac{1}{2}|\mathbf{C}| - \frac{(\mathbf{X}^i - \boldsymbol{\mu})^T \mathbf{C}^{-1} (\mathbf{X}^i - \boldsymbol{\mu})}{2} \\
&= -\frac{nd}{2} log 2\pi - \frac{n}{2}|\mathbf{C}| - \sum_{i=1}^{n} \frac{(\mathbf{X}^i - \boldsymbol{\mu})^T \mathbf{C}^{-1} (\mathbf{X}^i - \boldsymbol{\mu})}{2}
\end{aligned}$$

Differentiating w.r.t $\boldsymbol{\mu}$ suing $\frac{\partial \mathbf{x}^T \mathbf{A} \mathbf{x}}{\partial \mathbf{x}} = \mathbf{x}^T \mathbf{A} + \mathbf{x}^T \mathbf{A}^T$

$$\begin{aligned}
0 &= \sum_{i=1}^{n} (\mathbf{X}^i - \boldsymbol{\mu})^T (\mathbf{C}^{-1} + (\mathbf{C}^{-1})^T) \\
\implies 0 &= (\mathbf{C}^{-1} + (\mathbf{C}^{-1})^T) \sum_{i=1}^{n} (\mathbf{X}^i)^T - n\boldsymbol{\mu}^T
\end{aligned}$$

multiplying by $((\mathbf{C}^{-1} + (\mathbf{C}^{-1})^T))^{-1}$ since $\mathbf{C}$ is invertible

$$\begin{aligned}
\implies \boldsymbol{\mu}^T &= \frac{\sum_{i=1}^{n} (\mathbf{X}^i)^T}{n} \\
\implies \boldsymbol{\mu} &= \frac{\sum_{i=1}^{n} \mathbf{X}^i}{n}
\end{aligned}$$

i.e. MLE estimate for $\boldsymbol{\mu}$ is $\boldsymbol{\mu}^{MLE} = \frac{\sum_{i=1}^{n} \mathbf{X}^i}{n}$