Exercise 3

Machine Learning I

2A-1.

Short way:

Prerequisites

The simplest most high-level way is to use predefined rules for matrix differentiation. The rules for this algebra are laid out in the “Matrix Cookbook” on page 8 and page 10.

The book can be found here:

https://www.math.uwaterloo.ca/~hwolkowi/matrixcookbook.pdf

Some important rules form this book (let $\mathbf{x}$ be a column vector):

$$\frac{\partial \boldsymbol{x}^T\mathbf{a}}{\partial \mathbf{x}} = \frac{\partial \mathbf{a}^T\mathbf{x}}{\partial \mathbf{x}} = \mathbf{a}$$

$$\frac{\partial}{\partial \mathbf{x}}\boldsymbol{x}^T\mathbf{A}\mathbf{x} = \mathbf{x}^T(\mathbf{A}^T + \mathbf{A})$$

$$\frac{\partial \mathbf{a}^T\mathbf{X}^{-1}\mathbf{b}}{\partial \mathbf{X}} = -\mathbf{X}^{-\boldsymbol{T}}\mathbf{a}\mathbf{b}^{\boldsymbol{T}}\mathbf{X}^{-\boldsymbol{T}}$$

$$\frac{\partial \ln|\det \mathbf{X}|}{\partial \mathbf{X}} = (\mathbf{X}^{-1})^T = (\mathbf{X}^T)^{-1}.$$

We have:

$$L(\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \prod_{n=1}^{N} (2\pi)^{-\frac{d}{2}}|\Sigma|^{-\frac{1}{2}}e^{-\frac{1}{2}(\mathbf{x}_n-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{x}_n-\boldsymbol{\mu})} = (2\pi)^{-\frac{dN}{2}}|\Sigma|^{-\frac{N}{2}}e^{-\frac{1}{2}\sum_{n=1}^{N}(\mathbf{x}_n-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{x}_n-\boldsymbol{\mu})}$$

As usual, we take the logarithmic transform:

$$\ln L(\boldsymbol{\mu}, \boldsymbol{\Lambda}) = -\frac{dN}{2}\log 2\pi + \frac{N}{2}\log|\boldsymbol{\Lambda}| - \frac{1}{2}\sum_{n=1}^{N}(\mathbf{x}_n - \boldsymbol{\mu})^T\boldsymbol{\Lambda}(\mathbf{x}_n - \boldsymbol{\mu}).$$

Calculation of the mean:

First, let us calculate out the brackets in the previous exponent:

$$\sum_{n=1}^{N}(\mathbf{x}_n - \boldsymbol{\mu})^T\boldsymbol{\Lambda}(\mathbf{x}_n - \boldsymbol{\mu}) = \sum_{n=1}^{N}\mathbf{x}_n^T\boldsymbol{\Lambda}\mathbf{x}_n - \mathbf{x}_n^T\boldsymbol{\Lambda}\boldsymbol{\mu} - \boldsymbol{\mu}^T\boldsymbol{\Lambda}\mathbf{x}_n + \boldsymbol{\mu}^T\boldsymbol{\Lambda}\boldsymbol{\mu}.$$

Now apply the rules of matrix differentiation algebra:

$$\begin{aligned}
\frac{\partial}{\partial \boldsymbol{\mu}} \ln L(\boldsymbol{\mu}, \boldsymbol{\Lambda}) &= -\frac{1}{2}\frac{\partial}{\partial \boldsymbol{\mu}}\left[\sum_{n=1}^{N} \mathbf{x}_n^T \boldsymbol{\Lambda} \mathbf{x}_n - \mathbf{x}_n^T \boldsymbol{\Lambda} \boldsymbol{\mu} - \boldsymbol{\mu}^T \boldsymbol{\Lambda} \mathbf{x}_n + \boldsymbol{\mu}^T \boldsymbol{\Lambda} \boldsymbol{\mu}\right] \\
&= -\frac{1}{2}\left[\sum_{n=1}^{N} -\mathbf{x}_n^T \boldsymbol{\Lambda} - \mathbf{x}_n^T \boldsymbol{\Lambda} + \boldsymbol{\mu}^{\boldsymbol{T}} \left(\underbrace{\boldsymbol{\Lambda}^{\boldsymbol{T}} + \boldsymbol{\Lambda}}_{=2\boldsymbol{\Lambda}\ \boldsymbol{due\ symmetry}} \right)\right] \\
&= -\frac{1}{2}\left[\sum_{n=1}^{N} -2\mathbf{x}_n^T \boldsymbol{\Lambda} + 2\boldsymbol{\mu}^T \boldsymbol{\Lambda}\right] \\
&= \sum_{n=1}^{N} [\mathbf{x}_n^T \boldsymbol{\Lambda} - \boldsymbol{\mu}^T \boldsymbol{\Lambda}].
\end{aligned}$$

Now we can solve for $\boldsymbol{\mu}$:

$$\begin{aligned}
& \sum_{n=1}^{N} [\mathbf{x}_n^T \boldsymbol{\Lambda} - \boldsymbol{\mu}^T \boldsymbol{\Lambda}] &&= 0 \\
\Leftrightarrow \quad & \sum_{n=1}^{N} \mathbf{x}_n^T \boldsymbol{\Lambda} &&= \sum_{n=1}^{N} \boldsymbol{\mu}^T \boldsymbol{\Lambda} \\
\Leftrightarrow \quad & \frac{1}{N}\sum_{n=1}^{N} \mathbf{x}_n^T &&= \boldsymbol{\mu}^T \\
\Leftrightarrow \quad & \frac{1}{N}\sum_{n=1}^{N} \mathbf{x}_n &&= \boldsymbol{\mu}
\end{aligned}$$

Calculation of Variance:

This one is more involved. But by just utilizing the rules of the Matrix Cookbook, we quickly get to a solution. Note: Now I use $\boldsymbol{\Sigma}$ instead of $\boldsymbol{\Lambda}$ for convenience.

$$\begin{aligned}
\frac{\partial}{\partial \boldsymbol{\mu}} \ln L(\boldsymbol{\mu}, \boldsymbol{\Sigma}) &= -\frac{N}{2}\boldsymbol{\Sigma}^{-\mathrm{T}} - \frac{1}{2}\sum_{n=1}^{N} [-\boldsymbol{\Sigma}^{-\boldsymbol{T}} (\mathbf{x}_n - \boldsymbol{\mu})(\mathbf{x}_n - \boldsymbol{\mu})^{\boldsymbol{T}} \boldsymbol{\Sigma}^{-\boldsymbol{T}}] \\
&= -\frac{N}{2} \underbrace{\boldsymbol{\Sigma}^{-1}}_{symmetry} - \frac{1}{2}\sum_{n=1}^{N} \left[- \underbrace{\boldsymbol{\Sigma}^{-1}}_{symmetry} (\mathbf{x}_n - \boldsymbol{\mu})(\mathbf{x}_n - \boldsymbol{\mu})^{\boldsymbol{T}} \underbrace{\boldsymbol{\Sigma}^{-1}}_{symmetry}\right]
\end{aligned}$$

Now solve for $\boldsymbol{\Sigma}^{-1}$:

$$
\begin{aligned}
& -\frac{N}{2}\underbrace{\boldsymbol{\Sigma}^{-1}}_{symmetry} -\frac{1}{2}\sum_{n=1}^{N}\left[-\underbrace{\boldsymbol{\Sigma}^{-1}}_{symmetry}(\mathbf{x}_n-\boldsymbol{\mu})(\mathbf{x}_n-\boldsymbol{\mu})^{\boldsymbol{T}}\underbrace{\boldsymbol{\Sigma}^{-1}}_{symmetry}\right] && = 0 \quad |\cdot\boldsymbol{\Sigma}\, left\,|\cdot 2 \\
\Leftrightarrow\quad & -N+\sum_{n=1}^{N}[(\mathbf{x}_n-\boldsymbol{\mu})(\mathbf{x}_n-\boldsymbol{\mu})^{\boldsymbol{T}}\boldsymbol{\Sigma}^{-1}] && = 0 \quad |\cdot\boldsymbol{\Sigma}\, right \\
\Leftrightarrow\quad & -N\boldsymbol{\Sigma}+\sum_{n=1}^{N}[(\mathbf{x}_n-\boldsymbol{\mu})(\mathbf{x}_n-\boldsymbol{\mu})^{\boldsymbol{T}}] && = 0 \\
\Leftrightarrow\quad & \frac{1}{N}\sum_{n=1}^{N}[(\mathbf{x}_n-\boldsymbol{\mu})(\mathbf{x}_n-\boldsymbol{\mu})^{\boldsymbol{T}}] && = \boldsymbol{\Sigma}
\end{aligned}
$$

It appears that the original task that says

$$\boldsymbol{\Sigma} = \frac{1}{N}\mathbf{x}_n\mathbf{x}_n^{\boldsymbol{T}}$$

is wrong, as the $\boldsymbol{\mu}$ is also part other solutions:

https://stats.stackexchange.com/questions/351549/maximum-likelihood-estimators-multivariate-gaussian

2A-3.

Direct calculation:

$N(x;\mu_1,\sigma_1^2)N(x;\mu_2,\sigma_2^2) = (2\pi)^{-\frac{1}{4}}\sigma_1\sigma_2 e^{-\frac{1}{2}\left[\frac{(x-\mu_1)^2}{\sigma_1^2}+\frac{(x-\mu_2)^2}{\sigma_2^2}\right]}$.

Calculation of the exponent and completing the square:

$$
\begin{aligned}
\frac{(x-\mu_1)^2}{\sigma_1^2}+\frac{(x-\mu_2)^2}{\sigma_2^2} &= \frac{x^2-2\mu_1 x+\mu_1^2}{\sigma_1^2}+\frac{x^2-2\mu_2 x+\mu_2^2}{\sigma_2^2} \\
&= \frac{(\sigma_2^2+\sigma_1^2)x^2-2x(\sigma_2^2\mu_1+\sigma_1^2\mu_2)+\sigma_2^2\mu_1^2+\sigma_1^2\mu_2^2}{\sigma_1^2\sigma_2^2} \\
&= \frac{x^2-2x\frac{(\sigma_2^2\mu_1+\sigma_1^2\mu_2)}{(\sigma_2^2+\sigma_1^2)}+\frac{\sigma_2^2\mu_1^2}{(\sigma_2^2+\sigma_1^2)}+\frac{\sigma_1^2\mu_2^2}{(\sigma_2^2+\sigma_1^2)}}{\frac{\sigma_1^2\sigma_2^2}{(\sigma_2^2+\sigma_1^2)}} \\
&\propto \frac{\left(x-\frac{(\sigma_2^2\mu_1+\sigma_1^2\mu_2)}{(\sigma_2^2+\sigma_1^2)}\right)^2}{\frac{\sigma_1^2\sigma_2^2}{(\sigma_2^2+\sigma_1^2)}}
\end{aligned}
$$

Let

$$
\sigma^2=\frac{\sigma_1^2\sigma_2^2}{(\sigma_2^2+\sigma_1^2)}
$$

$$
\mu=\frac{(\sigma_2^2\mu_1+\sigma_1^2\mu_2)}{(\sigma_2^2+\sigma_1^2)}
$$

Now insert into the original equation:

$$
\begin{aligned}
N(x;\mu_1,\sigma_1^2)N(x;\mu_2,\sigma_2^2) &\propto (2\pi)^{-\frac{1}{4}}\sigma_1\sigma_2 e^{-\frac{1}{2}\left[\frac{(x-\mu)^2}{\sigma^2}\right]} \\
&\propto (2\pi\sigma^2)^{-\frac{1}{2}}e^{-\frac{1}{2}\left[\frac{(x-\mu)^2}{\sigma^2}\right]} \\
&= N\left(\frac{(\sigma_2^2\mu_1+\sigma_1^2\mu_2)}{(\sigma_2^2+\sigma_1^2)},\frac{\sigma_1^2\sigma_2^2}{(\sigma_2^2+\sigma_1^2)}\right)
\end{aligned}
$$

2A-4.

As usual, we are taking the partial derivate $\frac{\partial SE(\mathbf{w})}{\partial w_i}$:

$$
\frac{\partial}{\partial w_i}0.5\sum_{n=1}^{N} r_n(w^T\phi(x_n)-t_n)^2 \quad = \sum_{n=1}^{N}\phi_i(x_n)r_n(w^T\phi(x_n)-t_n).
$$

If we tried to solve this for w_i, we would encounter dependencies on the other $w_j{}'s$. This is an indicator that it would make sense to convert the equation into matrix form and treat the entire derivative as a solution to a system of d equations.

Converting each term step by step leads to:

$$\sum_{n=1}^{N} \phi_i(x_n) r_n (w^T \phi(x_n) - t_n) \quad = \left(r_1\phi_i(x_1), \dots, r_n\phi_i(x_n)\right)[(\mathbf{w}^T \mathbf{\Phi}^T)^T - \mathbf{t}].$$

where

$$\mathbf{\Phi} = \begin{pmatrix} \phi_1(x_1) & \dots & \phi_n(x_1) \\ \dots & \dots & \dots \\ \phi_1(x_n) & \dots & \phi_n(x_n) \end{pmatrix}$$

is the design matrix. For the entire vector, this would lead to:

$$\frac{\partial SE(\mathbf{w})}{\partial \mathbf{w}} = \begin{pmatrix} r_1\phi_1(x_1) & \dots & r_n\phi_1(x_n) \\ \dots & \dots & \dots \\ r_1\phi_d(x_1) & \dots & \dots r_n\phi_d(x_n) \end{pmatrix} [(\mathbf{w}^T \mathbf{\Phi}^T)^T - \mathbf{t}] = \mathbf{0}.$$

Solving for $\mathbf{w}$ and utilizing the Penrose inverse A^+:

$$\begin{aligned} \mathbf{w} \quad &= \left[\begin{pmatrix} r_1\phi_1(x_1) & \dots & r_n\phi_1(x_n) \\ \dots & \dots & \dots \\ r_1\phi_d(x_1) & \dots & \dots r_n\phi_d(x_n) \end{pmatrix} \mathbf{\Phi}\right]^+ \begin{pmatrix} r_1\phi_1(x_1) & \dots & r_n\phi_1(x_n) \\ \dots & \dots & \dots \\ r_1\phi_d(x_1) & \dots & \dots r_n\phi_d(x_n) \end{pmatrix} \mathbf{t} \\ &= \left(\sum_{n=1}^{N} r_n \phi(x_n)\phi(x_n)^T\right)^+ \left(\sum_{n=1}^{N} r_n t_n \phi(x_n)\right). \end{aligned}$$