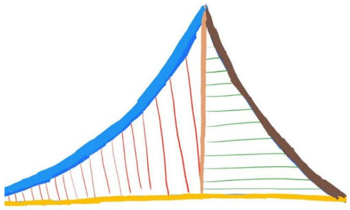
MACHINE LEARNING

Through Practise

G. V. V. Sharma



Contents

Introduction

This book introduces machine learning through simple examples

Chapter 1

Least Squares

1.0.1 Find the shortest distance between the lines

$\overrightarrow{r} = (\hat{i} + 2\hat{j} + \hat{k}) + \lambda(\hat{i} - \hat{j} + \hat{k})$ and

$\overrightarrow{r} = 2\hat{i} - \hat{j} - \hat{k} + \mu(2\hat{i} + \hat{j} + 2\hat{k})$

1.0.2 Find the shortest distance between the lines

$\frac{x+1}{7} = \frac{y+1}{-6} = \frac{z+1}{1}$ and $\frac{x-3}{1} = \frac{y-5}{-2} = \frac{z-7}{1}$ **Solution:** The given lines can be written as

$$\mathbf{x} = \begin{pmatrix} -1 \\ -1 \\ -1 \end{pmatrix} + \lambda_1 \begin{pmatrix} 7 \\ -6 \\ 1 \end{pmatrix} \tag{1.1}$$

$$\mathbf{x} = \begin{pmatrix} 3 \\ 5 \\ 7 \end{pmatrix} + \lambda_2 \begin{pmatrix} 1 \\ -2 \\ 1 \end{pmatrix} \tag{1.2}$$

$$\mathbf{x_1} = \begin{pmatrix} -1 \\ -1 \\ -1 \end{pmatrix}, \mathbf{x_2} = \begin{pmatrix} 3 \\ 5 \\ 7 \end{pmatrix}, \mathbf{m_1} = \begin{pmatrix} 7 \\ -6 \\ 1 \end{pmatrix}, \mathbf{m_2} = \begin{pmatrix} 1 \\ -2 \\ 1 \end{pmatrix} \tag{1.3}$$

We first check whether the given lines are skew. The lines

$$\mathbf{x} = \mathbf{x_1} + \lambda_1 \mathbf{m_1}, \, \mathbf{x} = \mathbf{x_2} + \lambda_2 \mathbf{m_2} \tag{1.4}$$

intersect if

$$\mathbf{M}\lambda = \mathbf{x_2} - \mathbf{x_1} \tag{1.5}$$

$$\mathbf{M} \triangleq \begin{pmatrix} \mathbf{m_1} & \mathbf{m_2} \end{pmatrix} \tag{1.6}$$

$$\boldsymbol{\lambda} \triangleq \begin{pmatrix} \lambda_1 \\ -\lambda_2 \end{pmatrix} \tag{1.7}$$

$$\tag{1.8}$$

Here we have,

$$\mathbf{M} = \begin{pmatrix} 7 & 1 \\ -6 & -2 \\ 1 & 1 \end{pmatrix} \mathbf{x_2} - \mathbf{x_1} = \begin{pmatrix} 4 \\ 6 \\ 8 \end{pmatrix} \tag{1.9}$$

We check whether the equation (1.10) has a solution

$$\begin{pmatrix} 7 & 1 \\ -6 & -2 \\ 1 & 1 \end{pmatrix} \boldsymbol{\lambda} = \begin{pmatrix} 4 \\ 6 \\ 8 \end{pmatrix} \tag{1.10}$$

the augmented matrix is given by,

$$\left(\begin{array}{cc|c} 7 & 1 & 4 \\ -6 & -2 & 6 \\ 1 & 1 & 8 \end{array}\right) \xleftrightarrow[R_3 \leftarrow R_3 - \frac{1}{7} R_1]{R_2 \leftarrow R_2 + \frac{6}{7} R_1} \tag{1.11}$$

$$\left(\begin{array}{cc|c} 7 & 1 & 4 \\ 0 & -\frac{8}{7} & \frac{66}{7} \\ 0 & \frac{6}{7} & -\frac{52}{7} \end{array}\right) \xleftrightarrow{R_3 \leftarrow R_3 + \frac{3}{4} R_2} \tag{1.12}$$

$$\left(\begin{array}{cc|c} 2 & 3 & 1 \\ 0 & -\frac{7}{2} & \frac{1}{2} \\ 0 & 0 & -\frac{5}{14} \end{array}\right) \tag{1.13}$$

The rank of the matrix is 3. So the given lines are skew. The closest points on two skew lines defined by (1.4) are given by

$$\mathbf{M}^\top \mathbf{M} \boldsymbol{\lambda} = \mathbf{M}^\top \left(\mathbf{x_2} - \mathbf{x_1}\right) \tag{1.14}$$

$$\implies \begin{pmatrix} 7 & -6 & 1 \\ 1 & -2 & 1 \end{pmatrix} \begin{pmatrix} 7 & 1 \\ -6 & -2 \\ 1 & 1 \end{pmatrix} \boldsymbol{\lambda} = \begin{pmatrix} 7 & -6 & 1 \\ 1 & -2 & 1 \end{pmatrix} \begin{pmatrix} 4 \\ 6 \\ 8 \end{pmatrix} \tag{1.15}$$

$$\implies \begin{pmatrix} 86 & 20 \\ 20 & 6 \end{pmatrix} \boldsymbol{\lambda} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{1.16}$$

The augmented matrix of the above equation (1.16) is given by,

$$\left(\begin{array}{cc|c} 86 & 20 & 0 \\ 20 & 6 & 0 \end{array}\right) \xleftrightarrow{R_2 \leftarrow R_2 - \frac{10}{43} R_1} \left(\begin{array}{cc|c} 86 & 20 & 0 \\ 0 & \frac{58}{43} & 0 \end{array}\right) \xleftrightarrow[R_2 \leftarrow \frac{43}{58} R_2]{R_1 \leftarrow \frac{1}{86}\left(R_1 - \frac{430}{29} R_2\right)} \tag{1.17}$$

$$\left(\begin{array}{cc|c} 1 & 0 & 0 \\ 0 & 1 & 0 \end{array}\right) \tag{1.18}$$

yielding

$$\begin{pmatrix} \lambda_1 \\ -\lambda_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{1.19}$$

The closest points $\mathbf{A}$ on line l_1 and $\mathbf{B}$ on line l_2 are given by,

$$\mathbf{A} = \mathbf{x_1} + \lambda_1 \mathbf{m_1} = \begin{pmatrix} -1 \\ -1 \\ -1 \end{pmatrix} \tag{1.20}$$

$$\mathbf{B} = \mathbf{x_2} + \lambda_2 \mathbf{m_2} = \begin{pmatrix} 3 \\ 5 \\ 7 \end{pmatrix} \tag{1.21}$$

The minimum distance between the lines is given by

$$\|\mathbf{B}-\mathbf{A}\| = \left\|\begin{pmatrix}4\\6\\8\end{pmatrix}\right\| = 2\sqrt{29} \tag{1.22}$$

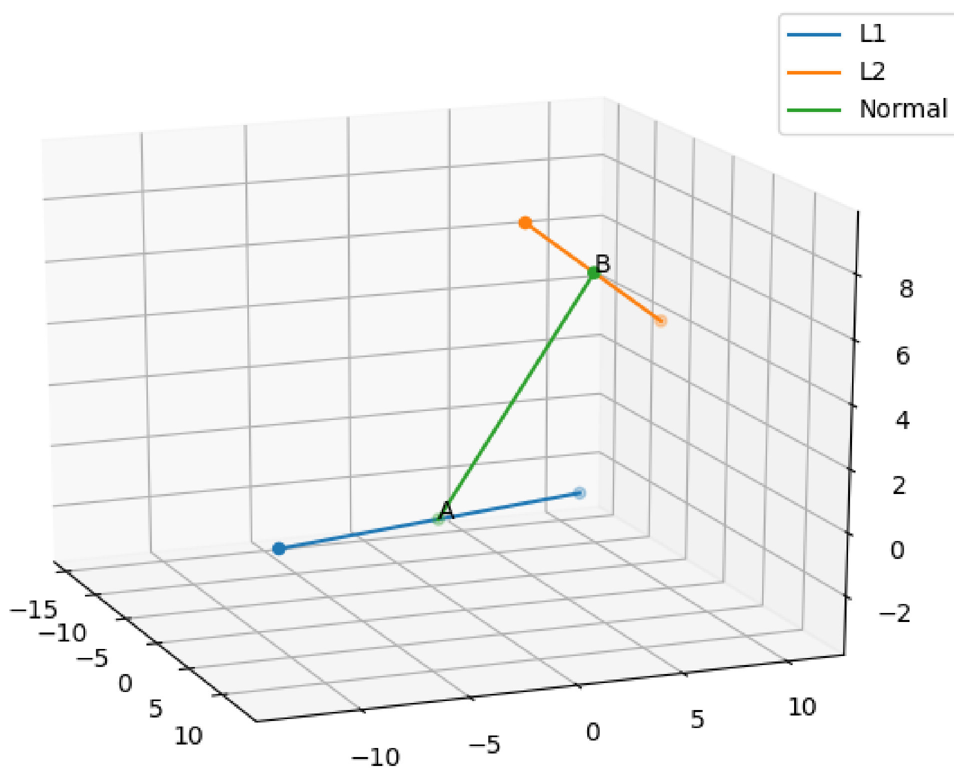


Figure 1.1:

1.0.3 Find the shortest distance between the lines whose vector equations are

$$\mathbf{x} = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} + \lambda_1 \begin{pmatrix} 1 \\ -3 \\ 2 \end{pmatrix} \tag{1.23}$$

and

$$\mathbf{x} = \begin{pmatrix} 4 \\ 5 \\ 6 \end{pmatrix} + \lambda_2 \begin{pmatrix} 2 \\ 3 \\ 1 \end{pmatrix} \tag{1.24}$$

Solution: In this case,

$$\mathbf{x_1} = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} \quad \mathbf{x_2} = \begin{pmatrix} 4 \\ 5 \\ 6 \end{pmatrix} \quad \mathbf{m_1} = \begin{pmatrix} 1 \\ -3 \\ 2 \end{pmatrix} \quad \mathbf{m_2} = \begin{pmatrix} 2 \\ 3 \\ 1 \end{pmatrix} \tag{1.25}$$

To check whether (A.3) has a solution in λ, we use the augmented matrix.

$$\begin{pmatrix} 1 & 2 & 3 \\ -3 & 3 & 3 \\ 2 & 1 & 3 \end{pmatrix} \xleftrightarrow{R_2 \leftarrow R_2 + 3R_1} \begin{pmatrix} 1 & 2 & 3 \\ 0 & 9 & 12 \\ 2 & 1 & 3 \end{pmatrix} \tag{1.26}$$

$$\xleftrightarrow{R_3 \leftarrow R_3 - 2R_1} \begin{pmatrix} 1 & 2 & 3 \\ 0 & 9 & 12 \\ 0 & -3 & -3 \end{pmatrix} \tag{1.27}$$

$$\xleftrightarrow{R_3 \leftarrow 3R_3 + R_2} \begin{pmatrix} 1 & 2 & 3 \\ 0 & 9 & 12 \\ 0 & 0 & 3 \end{pmatrix} \tag{1.28}$$

Clearly, the rank of this matrix is 3, and therefore, the lines are skew. Substituting from (1.25) in (A.6) and forming the augmented matrix,

$$\begin{pmatrix} 14 & -5 & 0 \\ -5 & 14 & 18 \end{pmatrix} \xleftrightarrow{R_1 \leftarrow R_1 + R_2} \begin{pmatrix} 9 & 9 & 18 \\ -5 & 14 & 18 \end{pmatrix} \tag{1.29}$$

$$\xleftrightarrow{R_1 \leftarrow \frac{R_1}{9}} \begin{pmatrix} 1 & 1 & 2 \\ -5 & 14 & 18 \end{pmatrix} \tag{1.30}$$

$$\xleftrightarrow{R_2 \leftarrow R_2 + 5R_1} \begin{pmatrix} 1 & 1 & 2 \\ 0 & 19 & 28 \end{pmatrix} \tag{1.31}$$

$$\xleftrightarrow{R_1 \leftarrow 19R_1 - R_2} \begin{pmatrix} 19 & 0 & 10 \\ 0 & 19 & 28 \end{pmatrix} \tag{1.32}$$

$$\xleftrightarrow[R_2 \leftarrow \frac{R_2}{9}]{R_1 \leftarrow \frac{R_1}{19}} \begin{pmatrix} 1 & 0 & \frac{10}{19} \\ 0 & 1 & \frac{28}{19} \end{pmatrix} \tag{1.33}$$

$$\implies \lambda = \frac{1}{19} \begin{pmatrix} 10 \\ 28 \end{pmatrix} \tag{1.34}$$

Hence, using (A.5) and substituing into (A.7) and (A.8),

$$\mathbf{A} = \frac{1}{19} \begin{pmatrix} 29 \\ 8 \\ 77 \end{pmatrix} \quad \mathbf{B} = \frac{1}{19} \begin{pmatrix} 20 \\ 11 \\ 86 \end{pmatrix} \tag{1.35}$$

Thus, the required distance is

$$\|\mathbf{B} - \mathbf{A}\| = \frac{\sqrt{9^2 + 3^2 + (-9)^2}}{19} = \frac{3}{\sqrt{19}} \tag{1.36}$$

The situation is depicted in Fig. 1.2.

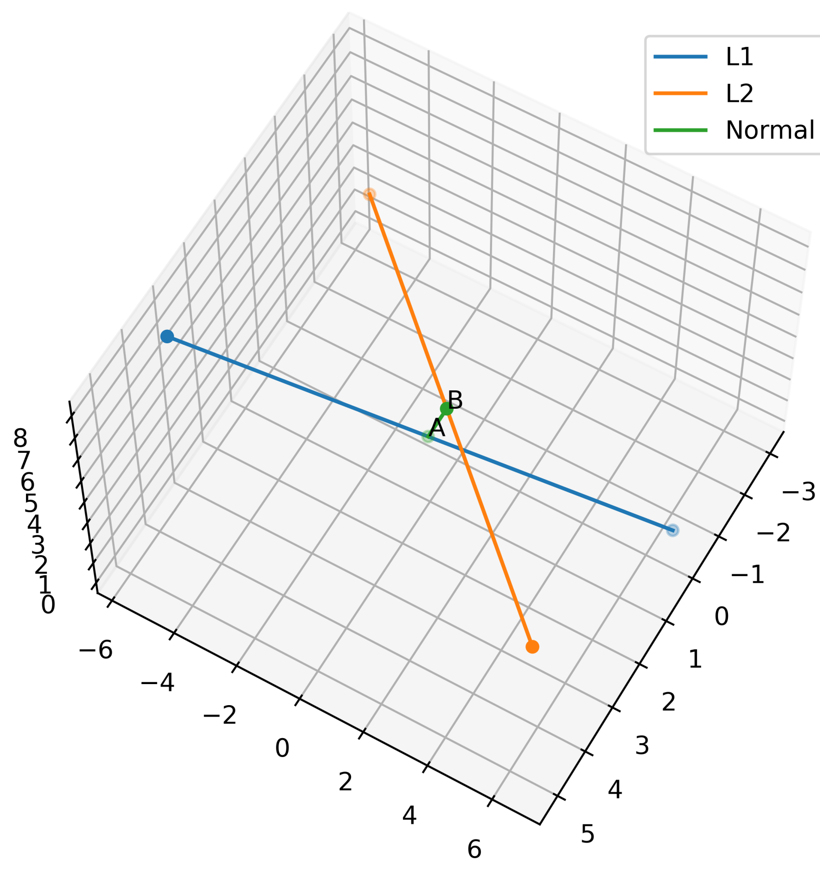


Figure 1.2: AB is the required shortest distance.

1.0.4

1.0.5 Find the shortest distance between the lines l_1 and l_2 whose vector equations are $\overrightarrow{r} = \hat{i} + \hat{j} + \lambda(2\hat{i} - \hat{j} + \hat{k})$ and $\overrightarrow{r} = 2\hat{i} + \hat{j} - \hat{k} + \mu(3\hat{i} - 5\hat{j} + 2\hat{k})$. **Solution:** The givne lines can be written in vector form as

$$\mathbf{x} = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} + \lambda_1 \begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix}, \mathbf{x} = \begin{pmatrix} 2 \\ 1 \\ -1 \end{pmatrix} + \lambda_2 \begin{pmatrix} 3 \\ -5 \\ 2 \end{pmatrix} \tag{1.37}$$

$$\implies \mathbf{x_1} = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}, \mathbf{x_2} = \begin{pmatrix} 2 \\ 1 \\ -1 \end{pmatrix}, \mathbf{m_1} = \begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix}, \mathbf{m_2} = \begin{pmatrix} 3 \\ -5 \\ 2 \end{pmatrix} \tag{1.38}$$

We first check whether the given lines are skew. The lines

$$\mathbf{x} = \mathbf{x_1} + \lambda_1 \mathbf{m_1}, \mathbf{x} = \mathbf{x_2} + \lambda_2 \mathbf{m_2} \tag{1.39}$$

intersect if

$$\mathbf{M}\boldsymbol{\lambda} = \mathbf{x_2} - \mathbf{x_1} \tag{1.40}$$

$$\mathbf{M} \triangleq \begin{pmatrix} \mathbf{m_1} & \mathbf{m_2} \end{pmatrix} \tag{1.41}$$

$$\boldsymbol{\lambda} \triangleq \begin{pmatrix} \lambda_1 \\ -\lambda_2 \end{pmatrix} \tag{1.42}$$

$$\tag{1.43}$$

Here we have,

$$\mathbf{M} = \begin{pmatrix} 2 & 3 \\ -1 & -5 \\ 1 & 2 \end{pmatrix}, \mathbf{x_2} - \mathbf{x_1} = \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix} \tag{1.44}$$

We check whether the equation (1.45) has a solution

$$\begin{pmatrix} 2 & 3 \\ -1 & -5 \\ 1 & 2 \end{pmatrix} \boldsymbol{\lambda} = \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix} \tag{1.45}$$

The augmented matrix is given by,

$$\left(\begin{array}{cc|c} 2 & 3 & 1 \\ -1 & -5 & 0 \\ 1 & 2 & -1 \end{array}\right) \xleftrightarrow[R_3 \leftarrow R_3 - \frac{1}{2}R_1]{R_2 \leftarrow R_2 + \frac{1}{2}R_1} \left(\begin{array}{cc|c} 2 & 3 & 1 \\ 0 & -\frac{7}{2} & \frac{1}{2} \\ 0 & \frac{1}{2} & -\frac{3}{2} \end{array}\right) \tag{1.46}$$

$$\xleftrightarrow{R_3 \leftarrow R_3 + 7R_2} \left(\begin{array}{cc|c} 2 & 3 & 1 \\ 0 & -\frac{7}{2} & \frac{1}{2} \\ 0 & 0 & -10 \end{array}\right) \tag{1.47}$$

The rank of the matrix is 3. So the given lines are skew. The closest points on two skew lines defined by (1.39) are given by

$$\mathbf{M}^{\top}\mathbf{M}\boldsymbol{\lambda} = \mathbf{M}^{\top}\left(\mathbf{x_2} - \mathbf{x_1}\right) \tag{1.48}$$

$$\implies \begin{pmatrix} 2 & -1 & 1 \\ 3 & -5 & 2 \end{pmatrix} \begin{pmatrix} 2 & 3 \\ -1 & -5 \\ 1 & 2 \end{pmatrix} \boldsymbol{\lambda} = \begin{pmatrix} 2 & -1 & 1 \\ 3 & -5 & 2 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix} \tag{1.49}$$

$$\implies \begin{pmatrix} 6 & 13 \\ 13 & 38 \end{pmatrix} \boldsymbol{\lambda} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \tag{1.50}$$

The augmented matrix of the above equation (1.50) is given by,

$$\left(\begin{array}{cc|c} 6 & 13 & 1 \\ 13 & 38 & 1 \end{array}\right) \xleftrightarrow{R_2 \leftarrow R_2 - \frac{13}{6} R_1} \left(\begin{array}{cc|c} 6 & 13 & 1 \\ 0 & \frac{59}{6} & -\frac{7}{6} \end{array}\right) \tag{1.51}$$

$$\xleftrightarrow{R_1 \leftarrow R_1 - \frac{78}{59} R_2} \left(\begin{array}{cc|c} 6 & 0 & \frac{150}{59} \\ 0 & \frac{59}{6} & -\frac{7}{6} \end{array}\right) \tag{1.52}$$

So, we get

$$\begin{pmatrix} \lambda_1 \\ -\lambda_2 \end{pmatrix} = \begin{pmatrix} \frac{25}{59} \\ -\frac{7}{59} \end{pmatrix} \tag{1.53}$$

The closest points $\mathbf{A}$ on line l_1 and $\mathbf{B}$ on line l_2 are given by,

$$\mathbf{A} = \mathbf{x_1} + \lambda_1 \mathbf{m_1} = \frac{1}{59}\begin{pmatrix} 109 \\ 34 \\ 25 \end{pmatrix} \tag{1.54}$$

$$\mathbf{B} = \mathbf{x_2} + \lambda_2 \mathbf{m_2} = \frac{1}{59}\begin{pmatrix} 139 \\ 24 \\ -45 \end{pmatrix} \tag{1.55}$$

The minimum distance between the lines is given by,

$$\|\mathbf{B} - \mathbf{A}\| = \left\| \frac{1}{59}\begin{pmatrix} 30 \\ -10 \\ -70 \end{pmatrix} \right\| = \frac{10}{\sqrt{59}} \tag{1.56}$$

See Fig. 1.3.

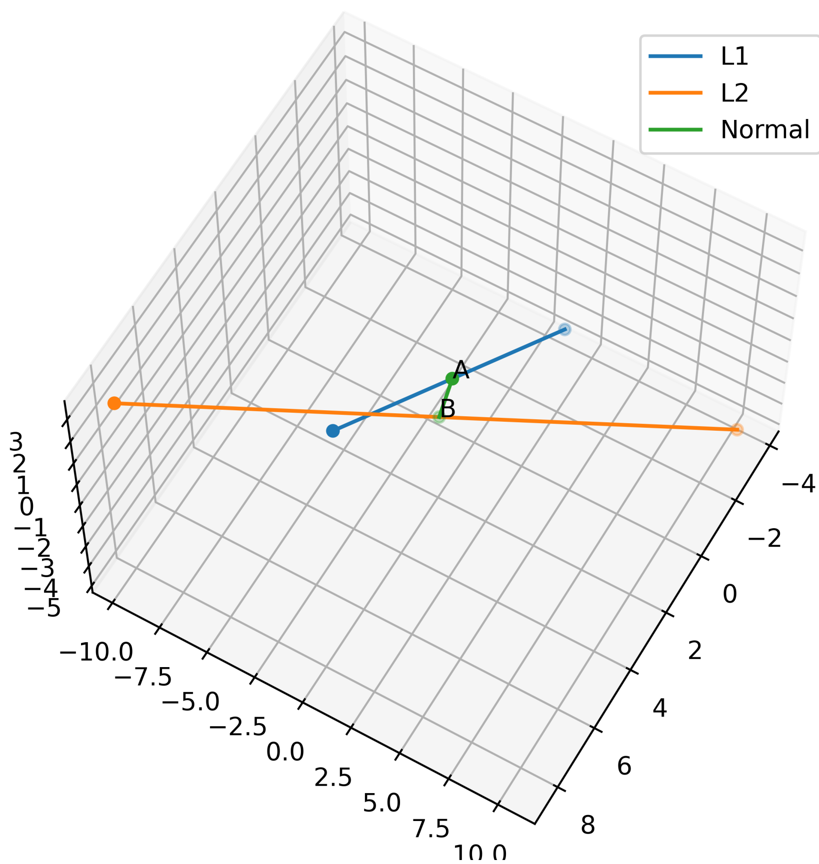


Figure 1.3:

Chapter 2

PT-100

This chapter illustrates the modeling of the voltage-temperature characteristics of the PT-100 RTD (Resistance Temperature Detector) using the least squares method.

2.1. Training Data

The training data gathered by the PT-100 to train the Arduino is shown in Table 2.1.

Temperature (°C)	**Voltage (V)**
66	1.85
27	1.76
2	1.66
23	1.72
56	1.82
34	1.76
33	1.75
31	1.74

Table 2.1: Training data.

The C++ source `codes/data.cpp` was used along with *platformio* to drive the Arduino. The effective schematic circuit diagram is shown in Figure 2.1.

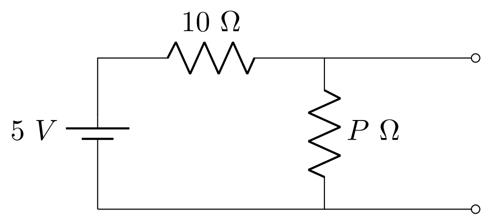


Figure 2.1: Schematic Circuit Diagram to Measure the Output of PT-100 (P).

2.2. Model

For the PT-100, we use the Callendar-Van Dusen equation

$$V(T) = V(0)\left(1 + AT + BT^2\right) \tag{2.1}$$

$$\implies c = \mathbf{n}^\top \mathbf{x} \tag{2.2}$$

where

$$c = V(T),\ \mathbf{n} = V(0)\begin{pmatrix}1\\A\\B\end{pmatrix},\ \mathbf{x} = \begin{pmatrix}1\\T\\T^2\end{pmatrix} \tag{2.3}$$

For multiple points, (2.2) becomes

$$\mathbf{X}^\top \mathbf{n} = \mathbf{C} \tag{2.4}$$

where

$$\mathbf{X} = \begin{pmatrix} 1 & 1 & \dots & 1 \\ T_1 & T_2 & \dots & T_n \\ T_1^2 & T_2^2 & \dots & T_n^2 \end{pmatrix} \tag{2.5}$$

$$\mathbf{C} = \begin{pmatrix} V(T_1) \\ V(T_2) \\ \vdots \\ V(T_n) \end{pmatrix} \tag{2.6}$$

and $\mathbf{n}$ is the unknown.

2.3. Solution

We approximate $\mathbf{n}$ by using the least sqaures method. The Python code `codes/lsq.py` solves for $\mathbf{n}$.

The calculated value of $\mathbf{n}$ is

$$\mathbf{n} = \begin{pmatrix} 1.6547 \\ 3.199 \times 10^{-3} \\ -3.9599 \times 10^{-6} \end{pmatrix} \tag{2.7}$$

The approximation is shown in Fig. 2.2.

Thus, the approximate model is given by

$$\begin{aligned} V(T) = 1.6547 + \left(3.199 \times 10^{-3}\right) T \\ - \left(3.9599 \times 10^{-6}\right) T^2 \end{aligned} \tag{2.8}$$

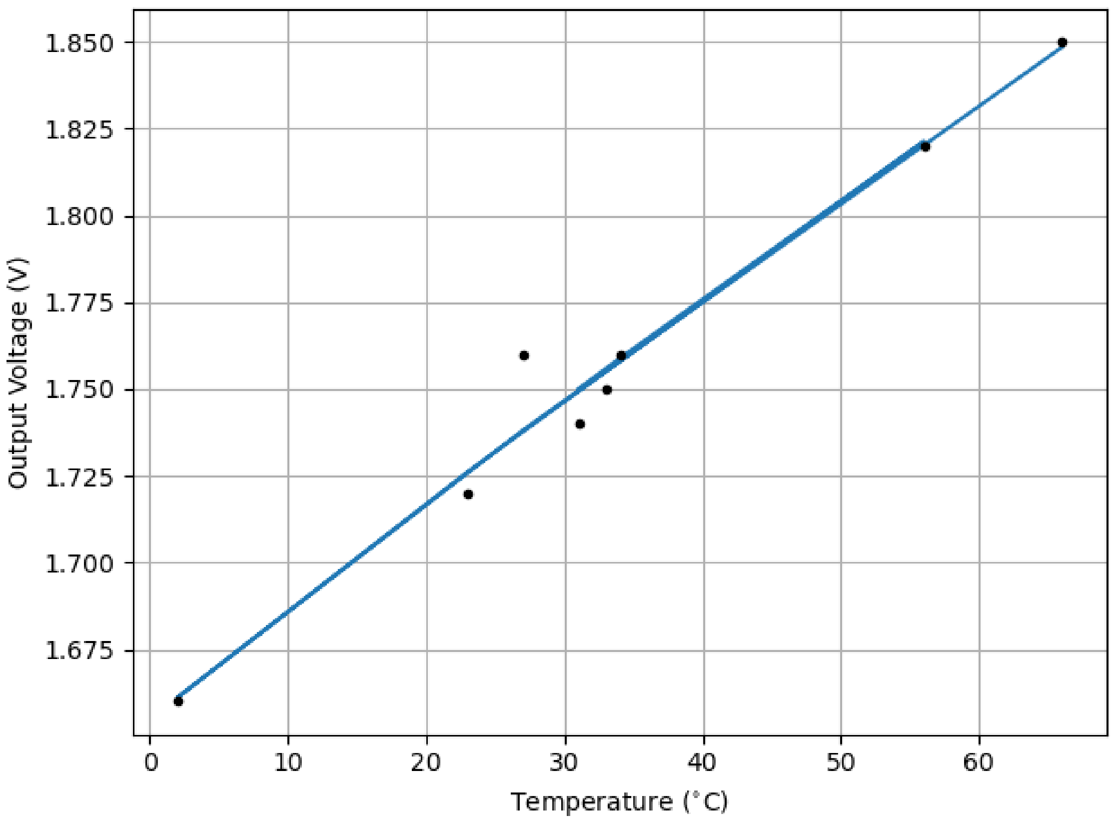


Figure 2.2: Training the model.

Notice in (2.8) that the coefficient of T^2 is negative, and hence the governing function is strictly concave. Hence, we cannot use gradient descent methods to solve this problem.

2.4. Validation

The validation dataset is shown in Table 2.2. The results of the validation are shown in Fig. 2.3.

Temperature (°C)	**Voltage (V)**
4	1.67
25	1.73
61	1.83
35	1.77

Table 2.2: Validation data.

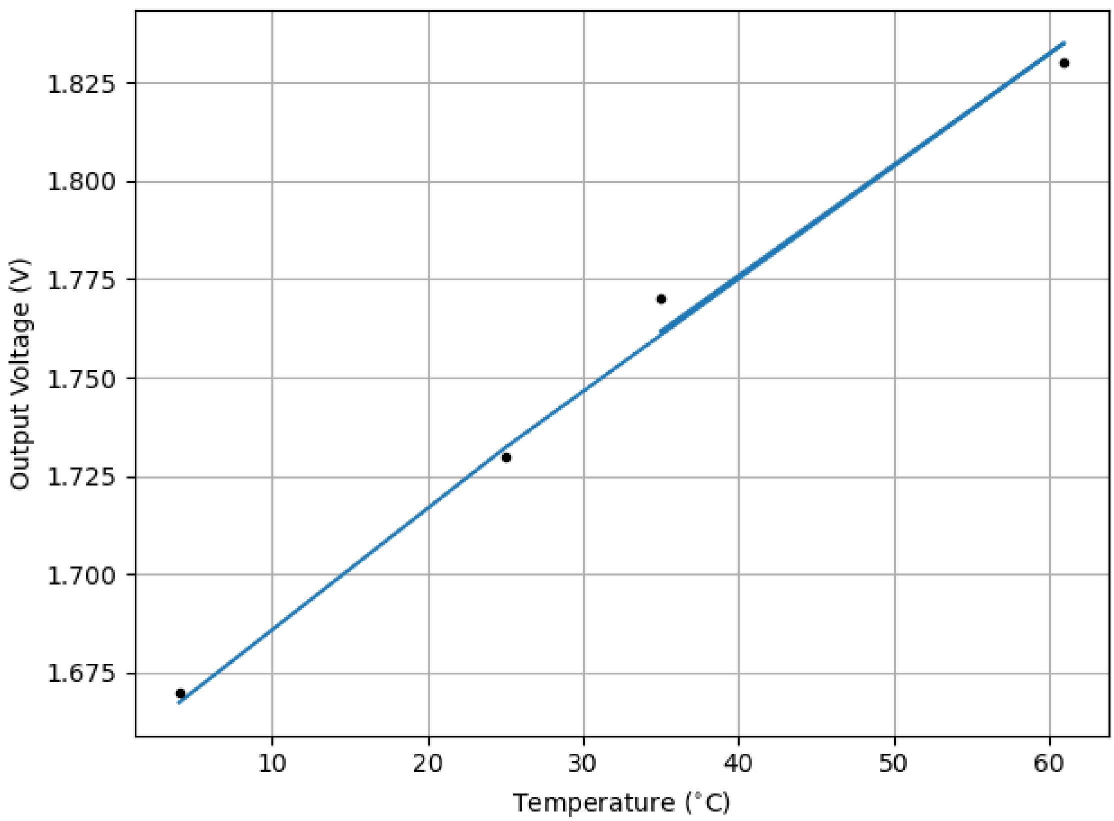


Figure 2.3: Validating the model.

Chapter 3

K-Means Method

3.1. Introduction

We test the utility of the K-means algorithm in assigning grades as compared to estimating the grades using the standard normal distribution.

We consider the scores of $N = 94$ students who have taken a course in the Indian Institute of Technology, Hyderabad (IITH) as our dataset.

3.2. Fitting a Gaussian Curve

Since N is not very large, given the scores of each student $x_i,\ 1 \leq i \leq N$, we can compute the population mean and population variance as

$$\mu = E\left[x\right] \tag{3.1}$$

$$\sigma^2 = E\left[(x-\mu)^2\right] \tag{3.2}$$

We assume that the scores $x \sim N\left(\mu, \sigma^2\right)$. Thus, we compute the Z-scores as

$$Z = \frac{x - \mu}{\sigma} \tag{3.3}$$

The grades are assigned as per Table 3.1.

Interval	**Grade**
$(-\infty$, -3]	F
(-3, -2]	D
(-2, 1]	C
(-1, 0]	B-
(0, 1]	B
(1, 2]	A-
(2, 3]	A
(3,∞)	A+

Table 3.1: Grading Scheme.

The Python code `codes/grades_norm.py` takes the given input population dataset `marks.xlsx` and assigns grades appropriately. The grades are output to `grades.xlsx`.

3.3. K-Means Clustering

K-Means clustering is an unsupervised classification model, which attempts to cluster unlabeled data in order to gain more structure from it.

We frame this requirement as an optimization problem. For a set of data points $\{\mathbf{x}_i\}_{i=1}^N$ and means $\{\mu_i\}_{i=1}^K$, we define for $1 \leq n \leq N,\ 1 \leq k \leq K$,

$$r_{nk} \triangleq \begin{cases} 1 & \arg\min_j \|\mathbf{x}_n - \mu_j\| = k \\ 0 & \text{otherwise} \end{cases} \tag{3.4}$$

Thus, we need to find points μ_k minimizing the cost function

$$J \triangleq \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \left\| \mathbf{x}_n - \mu_k \right\|^2 \tag{3.5}$$

Clearly, (3.5) is a quadratic function of μ_k. Differentiating with respect to μ_k and setting the derivative to zero, we get

$$\sum_{n=1}^{N} 2\mu_k r_{nk} \left(\mathbf{x}_n - \mu_k \right) = 0 \tag{3.6}$$

$$\implies \mu_{\mathbf{k}} = \frac{\sum_{n=1}^{N} r_{nk} \mathbf{x}_n}{\sum_{n=1}^{N} r_{nk}} = \frac{\mathbf{X}\mathbf{r}_k}{\mathbf{1}^\top \mathbf{r}_k} \tag{3.7}$$

where

$$\mathbf{X} \triangleq \begin{pmatrix} \mathbf{x}_1 & \mathbf{x}_2 & \ldots & \mathbf{x}_n \end{pmatrix} \tag{3.8}$$

$$\mathbf{r}_k \triangleq \begin{pmatrix} r_{1k} & r_{2k} & \ldots & r_{nk} \end{pmatrix}^\top \tag{3.9}$$

$$\mathbf{1} \triangleq \begin{pmatrix} 1 & 1 & \ldots & 1 \end{pmatrix}^\top \tag{3.10}$$

From (3.7), we see that the optimum is attained when μ_k is set to the expectation of the $\mathbf{x}_n$ with respect to r_{nk}.

Thus, the K-means algorithm is essentially an *EM algorithm*, where each iteration consists of two steps.

1. *E Step*: Calculate the K-expected values

$$\tilde{\mu_k} \triangleq \frac{\sum_{n=1}^{N} r_{nk} \mathbf{x}_n}{\sum_{n=1}^{N} r_{nk}} \tag{3.11}$$

   for $1 \leq k \leq K$.

2. *M Step*: Assign $\mu_k \leftarrow \tilde{\mu_k}$ for $1 \leq k \leq K$.

3.4. Results

The grade distribution using each method is shown in Fig. 3.1 and Fig. 3.2. Based on the

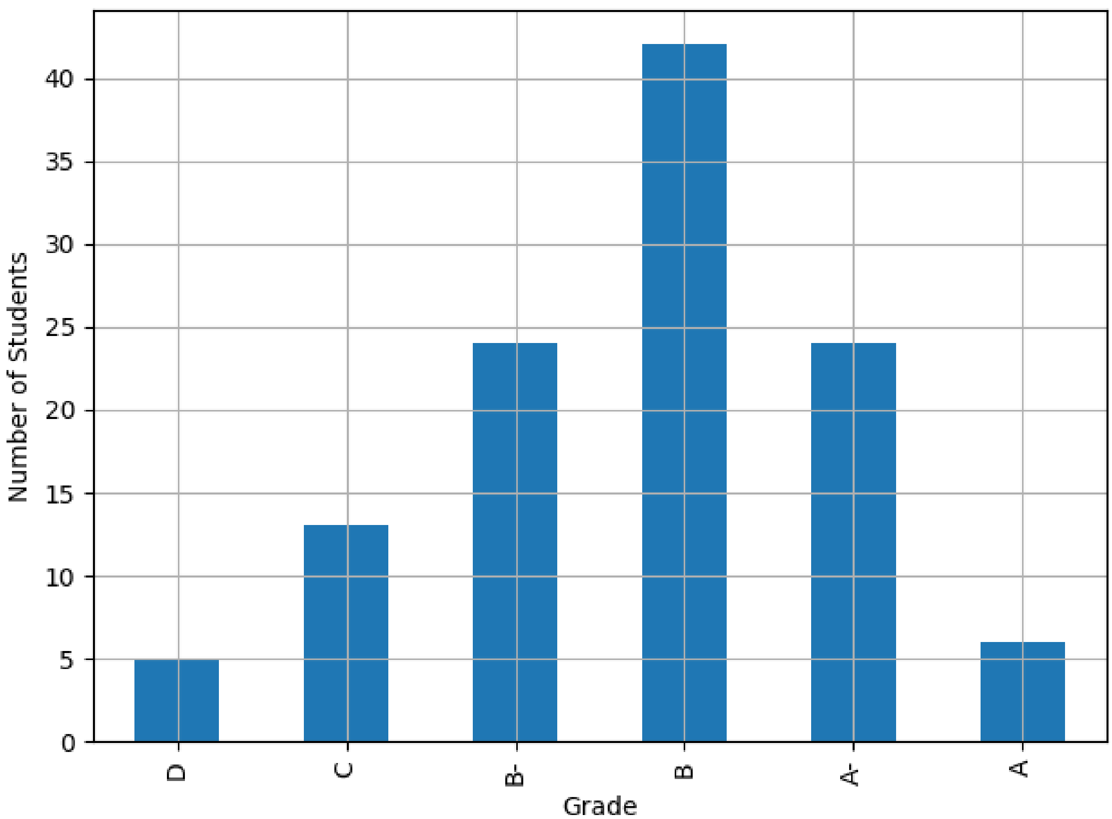


Figure 3.1: Grade distribution using a Gaussian curve.

results, we can make the following observations:

1. Grading using the Gaussian distribution would lead to many students failing the course, while this is not the case using the K-means algorithm.

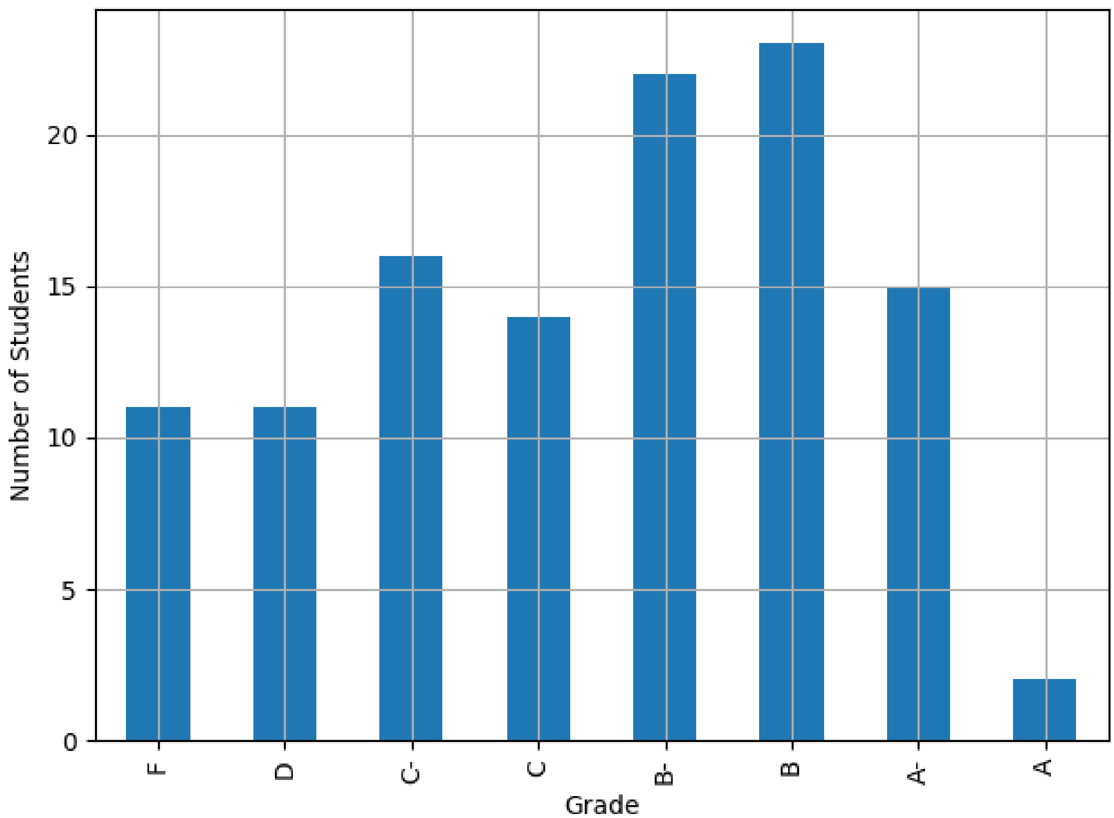


Figure 3.2: Grade distribution using the K-means algorithm.

2. Using the Gaussian distribution is quite unfair, since there could be students with quite similar marks but with a difference in grade, just because they lie on either side of a predefined boundary.

3. The K-means algorithm allows for better decision boundaries, depending on how skewed the performance of the students is, accordingly to the difficulty of the course.

4. Unlike the Gaussian distribution, the K-means algorithm can be used for a fairer assignment of the grades, no matter how skewed the performance of students in a course is.

Chapter 4

Beacon Tracking

This chpater demonstrates the use of machine learning in beacon tracking using an unmanned ground vehicle (UGV) and a WiFi-enabled microcontroller such as the ESP32.

4.1. Assets

1. UGV chassis with DC motors
2. ESP32 microcontroller with Type-B USB cable
3. L293D Motor Driver IC
4. Breadboard and Jumper Wires
5. Android phone
6. (Optional) USB 2.0/3.0 Hub

4.2. Procedure

1. Make the connections as per the wiring diagram in Fig. 4.1.

2. Connect the ESP32 board to your Android Phone.

3. Generate the firmware by entering the following commands.

```
$ cd codes
$ pio run
```

4. Go to ArduinoDroid and select

```
Actions – Upload – Upload Precompiled
```

and choose the firmware file at

```
codes/.pio/build/firmware.hex
```

5. Now put the phone at a reasonable distance from the UGV with no obstacles in the way and then turn on the hotspot. The UGV should travel towards the phone and stop near it.

4.3. Working

4.3.1. Underlying Principles

1. To estimate (radial) distance to beacon, we use its signal strength. For WiFi, this is the **Received Signal Strength Indicator** (RSSI).

2. The RSSI (R dBm) at distance d metres is given by

$$R(d) = R(1) - 10\log_{10}(d) \tag{4.1}$$

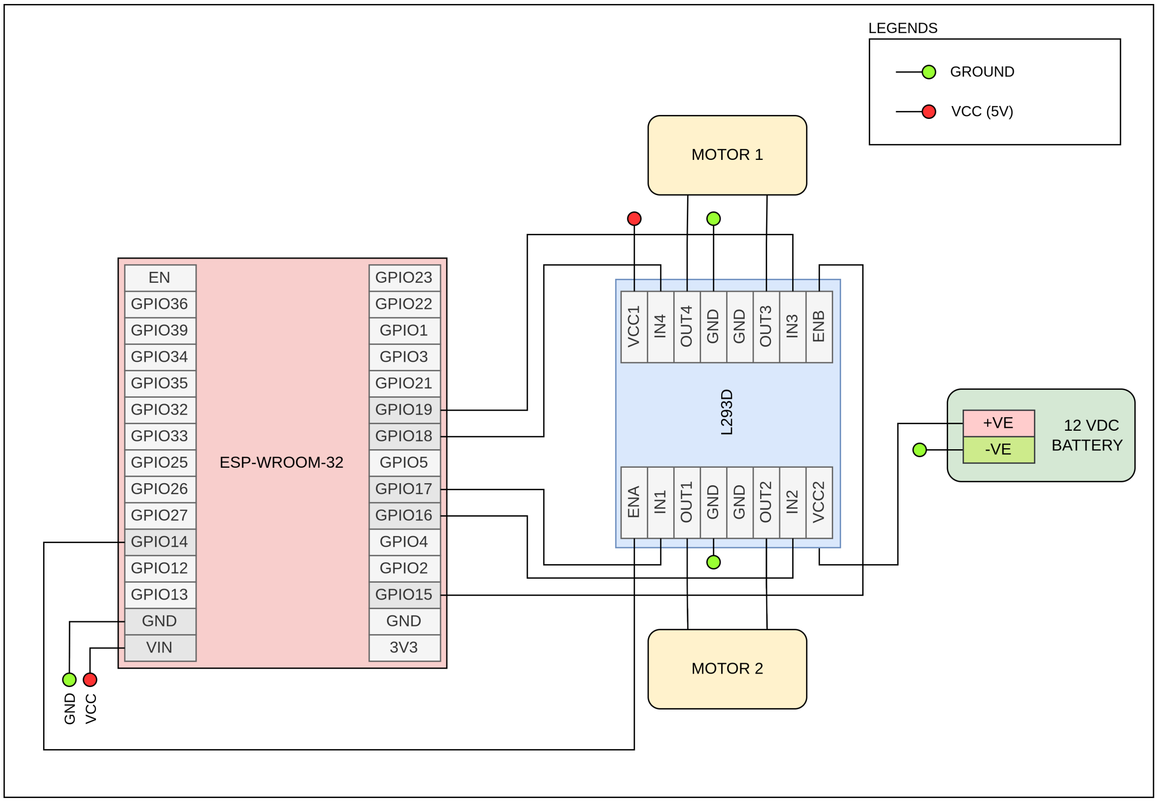


Figure 4.1: Wiring Diagram for Beacon Tracking.

3. Clearly, $R(d)$ is a convex function. Hence, we can use gradient descent.

4.3.2. Algorithm Description

Please note that this is a generic description of the algorithm employed. Refer to

ugv−beacon/codes/src/main.cpp

for a more verbose implementation.

1. If the UGV is close enough to the beacon, *terminate*.

2. Take measurements at various points on a straight line.

3. Based on these measurements, decide the next move of the UGV, and recurse till the UGV is close enough to the beacon.

4.4. Observations

1. The UGV eventually converges close to the beacon (here, the hotspot).

2. However, if there are a lot of nearby obstacles, the UGV may not converge close to the location of the beacon. It may either get physically blocked by the beacon or the signal interference may be too high.

Appendix A

Three Dimensions

A.1. The lines

$$\mathbf{x} = \mathbf{x_1} + \lambda_1 \mathbf{m_1} \tag{A.1}$$

$$\mathbf{x} = \mathbf{x_2} + \lambda_2 \mathbf{m_2} \tag{A.2}$$

intersect if

$$\mathbf{M}\boldsymbol{\lambda} = \mathbf{x_2} - \mathbf{x_1} \tag{A.3}$$

where

$$\mathbf{M} \triangleq \begin{pmatrix} \mathbf{m_1} & \mathbf{m_2} \end{pmatrix} \tag{A.4}$$

$$\boldsymbol{\lambda} \triangleq \begin{pmatrix} \lambda_1 \\ -\lambda_2 \end{pmatrix} \tag{A.5}$$

A.2. The closest points on two skew lines are given by

$$\mathbf{M}^\top \mathbf{M}\boldsymbol{\lambda} = \mathbf{M}^\top (\mathbf{x_2} - \mathbf{x_1}) \tag{A.6}$$

Solution: For the lines defined in (A.1) and (A.2), Suppose the closest points on both lines are

$$\mathbf{A} = \mathbf{x_1} + \lambda_1 \mathbf{m_1} \tag{A.7}$$

$$\mathbf{B} = \mathbf{x_2} + \lambda_2 \mathbf{m_2} \tag{A.8}$$

Then, AB is perpendicular to both lines, hence

$$\mathbf{m_1}^\top (\mathbf{A} - \mathbf{B}) = 0 \tag{A.9}$$

$$\mathbf{m_2}^\top (\mathbf{A} - \mathbf{B}) = 0 \tag{A.10}$$

$$\implies \mathbf{M}^\top (\mathbf{A} - \mathbf{B}) = \mathbf{O} \tag{A.11}$$

Using (A.7) and (A.8) in (A.11),

$$\mathbf{M}^\top (\mathbf{x_1} - \mathbf{x_2} + \mathbf{M}\boldsymbol{\lambda}) = \mathbf{0} \tag{A.12}$$

$$\tag{A.13}$$

yielding A.6.