COMP5318 - Lab 13 Solutions

June 2018

1 Question 1

By the definition of eigenvalue and eigenvector, we have

$$\mathbf{M}\mathbf{v} = \lambda\mathbf{v}, \tag{1}$$

where λ is the eigenvalue and $\mathbf{v}$ is the eigenvector. The equation could be written as $(\mathbf{M} - \lambda\mathbf{I})\mathbf{v} = \mathbf{0}$, which is equivalent to $\det(\mathbf{M} - \lambda\mathbf{I}) = 0$.

$$\begin{aligned}\det(\mathbf{M} - \lambda\mathbf{I}) &= \det(\begin{bmatrix} 3-\lambda & 2 \\ 2 & 6-\lambda \end{bmatrix}) \\ &= (3-\lambda)(6-\lambda) - 4 \\ &= \lambda^2 - 9\lambda + 14 = 0. \end{aligned} \tag{*}$$

By solving the equation (*), we will get $\lambda_1 = 2$ and $\lambda_2 = 7$.

For $\lambda_1 = 2$, we have $\mathbf{M} - \lambda\mathbf{I} = \begin{bmatrix} 1 & 2 \\ 2 & 4 \end{bmatrix}$, so $\mathbf{v}_1 = \begin{bmatrix} -2 \\ 1 \end{bmatrix}$;

For $\lambda_2 = 7$, we have $\mathbf{M} - \lambda\mathbf{I} = \begin{bmatrix} -4 & 2 \\ 2 & -1 \end{bmatrix}$, so $\mathbf{v}_2 = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$.

2 Question 2

Suppose two events:

X: a person is guilty or innocent. $X = 0$ stands for innocent, and $X = 1$ stands for guilty.

Y: a DNA match occurs or not. $Y = 0$ means non-match, and $Y = 1$ means match.

Suppose N people in one town, so

$$P(Y = 1|X = 1) = 1, \tag{2}$$

$$P(Y = 1|X = 0) = \frac{1}{10^5}, \tag{3}$$

$$P(X = 1) = \frac{1}{N}. \tag{4}$$

The probability when there is a DNA match but the defendant is innocent is $P(X = 0|Y = 1)$. By using *Bayes' rule*,

$$\begin{aligned} P(X = 0|Y = 1) &= \frac{P(Y = 1|X = 0)P(X = 0)}{P(Y = 1|X = 0)P(X = 0) + P(Y = 1|X = 0)P(X = 0)} \\ &= \frac{\frac{1}{10^5} \times (1 - \frac{1}{N})}{\frac{1}{10^5} \times (1 - \frac{1}{N}) + 1 \times \frac{1}{N}} \\ &= \frac{N - 1}{10^5 + N - 1} \approx \frac{N}{10^5 + N}. \end{aligned} \tag{*}$$

From the equation (*), we have the following discussions:

1) if $N \leq 10^3$, then the probability is less than 1%. This means it is very improbable to match one innocent person when the population is small.

2) if $N \geq 10^5$, then the probability is larger than 50%. This means it is very probable to match one innocent person when the population is large enough.

3 Question 3

3(a)

$$\begin{aligned} p(X, Z|Y) &= \frac{p(X, Y, Z)}{p(Y)} \\ &= \frac{p(Z|Y)p(Y|X)p(X)}{p(Y)} \\ &= p(Z|Y)\frac{P(Y|X)p(X)}{p(Y)} \\ &= p(Z|Y)P(X|Y) \quad \text{(using Bayes rule)} \end{aligned}$$

3(b)

Assuming a Bernoulli variable, and visualising the conditional operations as coin flips conditional on other coin flips, observe the directed graphical model: $\textcircled{X} \rightarrow \textcircled{Y} \rightarrow \textcircled{Z}$ where $\textcircled{X} = p(X)$, $\textcircled{Y} = P(Y|X)$, and $\textcircled{Z} = p(Z|Y)$. So, X requires 1, Y requires 2, Z requires 2 giving 5 parameters total.

4 Question 4

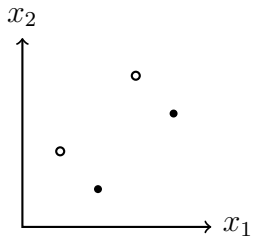
We will solve question 4 by graphing and then computing the solution to a simplified constrained optimisation problem.

4.1 Graphing

The training points are given by the table in Figure 1a. Plotting them we get a plot like the one in Figure 1b. The solution to a hard-margin SVM learning problem, like the one

x_1	x_2	y
1	2	+1
3	4	+1
2	1	-1
4	3	-1

(a) Training set

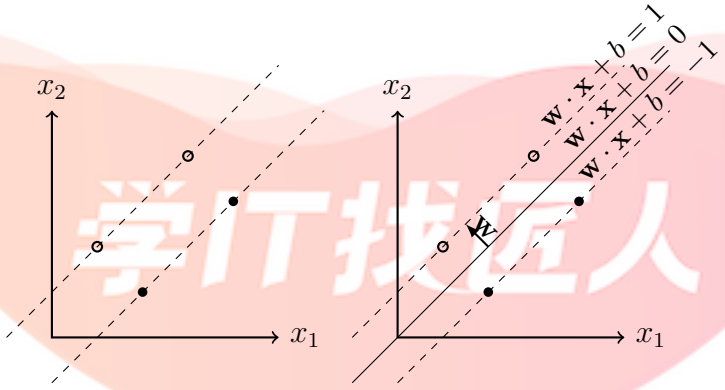


(b) Plot

Figure 1: Training set and plot with the training samples. Filled circles represent samples with negative (-1) labels and empty circles represent the positive-labelled (+1) ones.

in this question, is only possible if the groups of points of each class are linearly separable. Fortunately, by the plot in Figure 1b, we can see that that's the case for this question.

The first step is to find the support vectors, which are the points that delimit the classes boundaries. In our simple linearly separable case, we can visualise that the sample points provided by the training set are all support vectors. Then we can trace a line through each group of support vectors to determine the margin of our SVM problem, as plotted in Figure 2a. In fact, we could even have way more examples for each class. As long as the extra sample points lie beyond the dashed lines in Figure 2a, i.e. the margins, those extra points would not alter the solution to this hard-margin SVM problem, which can be solely determined by the support vectors.



(a) Margin (b) Hyper-planes

Figure 2: Graphing the SVM solution

By the SVM formulation, we are trying to find a hyper-plane $\mathbf{w}\cdot\mathbf{x}+b$ which can maximally separate the points of the two classes, i.e. the *maximum-margin hyper-plane*. Despite the complicated name, in our case, that's simply tracing a parallel line in the middle between the two margin borderlines, as shown in Figure 2b.

4.2 Computing the solution

The vector $\mathbf{w}$ solving the SVM problem is such that the hyper-plane determined by the support vectors of the positive-labelled class, $y = +1$, is $\mathbf{w} \cdot \mathbf{x} + b = 1$, and any point $\mathbf{x}$ in the same class lying beyond this plane would be assigned $\mathbf{w} \cdot \mathbf{x} + b > 1$. The same works for the negative-labelled points, $y = -1$, with $\mathbf{w} \cdot \mathbf{x} + b \leq -1$. We also have that $\mathbf{w} \cdot \mathbf{x} + b = 0$ at the maximum-margin hyper-plane.

By the plot in Figure 2b, we can see that the line $\mathbf{w} \cdot \mathbf{x} + b = 0$ intersects the origin, $x_1 = x_2 = 0$. From that, setting $\mathbf{x} = 0$, it is easy to see that the only way it can happen is with $b = 0$, regardless of $\mathbf{w}$.

To compute $\mathbf{w}$ we need to find a vector perpendicular to the hyper-plane $\mathbf{w} \cdot \mathbf{x} + b = 0$. Since $b = 0$ and our line is a diagonal, with a 45^o slope, we have that $\mathbf{w} = [u, v]$ is some vector with $u = -v$, so that the line is $ux_1 = ux_2$. However, by the formulation the solution's $\mathbf{w}$ is not any perpendicular vector. It is a vector of minimum norm satisfying the constraints defined by the margin, $y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1$. Formulating that for the support vectors, we have:

$$u - 2u \geq +1 \tag{5}$$
$$3u - 4u \geq +1 \tag{6}$$
$$2u - u \leq -1 \tag{7}$$
$$4u - 3u \leq -1 \tag{8}$$
$$\implies u \leq -1 \tag{9}$$

Back to the main formulation, we want:

$$\min\|\mathbf{w}\| = \min\sqrt{u^2 + v^2} = \min\sqrt{2}|u| = \min|u| \ , \text{ subject to: } u \leq -1 \ . \tag{10}$$

Therefore, the solution is $u = -1, v = 1$. In summary:

$$\mathbf{w} = [-1, 1] \ , \tag{11}$$
$$b = 0 \ . \tag{12}$$

5 Question 5

For more details, please see Murphy 8.3.6

In logistic regression, we prefer MAP estimation to MLE estimation. We may overfit if we use MLE, placing too much probability mass on our training data and creating solutions that do not generalise well to test data.

Even in data rich settings, we prefer regularised models (in similar vein to regularising linear regression models with methods such as ridge regression, lasso, elastic net etc). Optimising with respect to MLE may lead to brittle solutions that do not generalise well. Murphy explains in his book:

Suppose we have linearly separable data. MLE is obtained when $\|\mathbf{w}\| \to \infty$, which is an infinitely steep sigmoid function $\mathbf{w}^2\mathbf{x} > \mathbf{w}_0 \cdots$ linear threshold unit).

This assigns maximal amount of probability mass to the training data, and accordingly will overfit and not generalise well. To prevent this, we can use l2 regularisation.

6 Question 6

The posterior mean is given as

$$(\mathbf{W}^T\mathbf{W} + \sigma^2\mathbf{I})^{-1}\mathbf{W}^T(\mathbf{x} - \mu) \tag{13}$$

where

$$\mathbf{W} = \mathbf{U}_M(\mathbf{L}_M - \sigma^2\mathbf{I})^{1/2}\mathbf{R} \tag{14}$$

using the ML solution. See lecture slides.

Since $\mathbf{R}$ is an arbitrary orthogonal matrix we set it to the identity matrix, $\mathbf{I}$.

Substituting $\mathbf{W} = ...$ into the first equation and setting the limit, $\sigma \to 0$ we get

$$(\mathbf{W}^T\mathbf{W})^{-1}\mathbf{W}^T(\mathbf{x} - \mu) \tag{15}$$

Focusing on $(\mathbf{W}^T\mathbf{W})^{-1}$, remembering that $\sigma = 0$ and $\mathbf{R} = \mathbf{I}$

$$\begin{aligned}(\mathbf{W}^T\mathbf{W})^{-1} &= ((\mathbf{U}_M\mathbf{L}_M^{1/2})^T(\mathbf{U}_M\mathbf{L}_M^{1/2}))^{-1} \\ &= ((\mathbf{L}_M^{1/2})^T\mathbf{U}_M^T)(\mathbf{U}_M\mathbf{L}_M^{1/2}))^{-1} \\ &= ((\mathbf{L}_M^{1/2})^T\mathbf{L}_M^{1/2})^{-1} \\ &= \mathbf{L}_M^{-1}\end{aligned}$$

Line 2-3, the columns of each $\mathbf{U}_M$ comes from SVD and forms a set of orthonormal vectors so $\mathbf{U}_M^T\mathbf{U}_M = \mathbf{I}$. Line 3-4, $\mathbf{L}_M$ is a diagonal matrix of eigen values corresponding to each eigen vector. Since diagonal matrix is symmetric, $\mathbf{L}_M^T = \mathbf{L}_M$.

Going back to Equation 15, we have

$$\begin{aligned}(\mathbf{W}^T\mathbf{W})^{-1}\mathbf{W}^T(\mathbf{x} - \mu) &= \mathbf{L}_M^{-1}\mathbf{W}(\mathbf{x} - \mu) \\ &= \mathbf{L}_M^{-1}((\mathbf{U}_M\mathbf{L}_M^{1/2})^T(\mathbf{x} - \mu) \\ &= \mathbf{L}_M^{-1}\mathbf{L}_M^{1/2}\mathbf{U}_M^T(\mathbf{x} - \mu) \\ &= \mathbf{L}_M^{-1/2}\mathbf{U}_M^T(\mathbf{x} - \mu)\end{aligned}$$

which is the equation for each point transformed by normal PCA see equation 12.24 in bishop

7 Question 7

Since the question asks to add Gaussian noise to each data point, let $x_{ij} = x_{ij} + \epsilon_{ij}$ where i is the data point and j is the feature of that data point. Therefore

$$y_i(x, w) = w_o + \sum_{j=1}^{D} w_j(x_{ij} + \epsilon_{ij}) \tag{16}$$

Substituting that into the cost function,

$$E_D(w) = \frac{1}{2}\sum_{i=1}^{N}\Big[w_o + \sum_{j=1}^{D} w_j(x_{ij} + \epsilon_{ij}) - t_i\Big]^2$$
$$= \frac{1}{2}\sum_{i=1}^{N}\Big[w_o + \sum_{j=1}^{D} w_j x_{ij} - t_i + \sum_{j=1}^{D} w_j \epsilon_{ij}\Big]^2$$

Let $A_i = w_o + \sum_{j=1}^{D} w_j x_{ij} - t_i$ and $B_i = \sum_{j=1}^{D} w_j \epsilon_{ij}$

$$E_D(w) = \frac{1}{2}\sum_{i=1}^{N}\Big[A_i + B_i\Big]^2 \tag{17}$$
$$= \frac{1}{2}\sum_{i=1}^{N}\Big[A_i^2 + 2A_iB_i + B_i^2\Big] \tag{18}$$
$$= \frac{1}{2}\sum_{i=1}^{N} A_i^2 + \sum_{i=1}^{N} A_iB_i + \frac{1}{2}\sum_{i=1}^{N} B_i^2 \tag{19}$$

Taking the expectation of the cost function

$$\mathbb{E}[E_D(w)] = \mathbb{E}\left[\frac{1}{2}\sum_{i=1}^{N} A_i^2 + \sum_{i=1}^{N} A_iB_i + \frac{1}{2}\sum_{i=1}^{N} B_i^2\right]$$
$$= \mathbb{E}\left[\frac{1}{2}\sum_{i=1}^{N} A_i^2\right] + \mathbb{E}\left[\sum_{i=1}^{N} A_iB_i\right] + \mathbb{E}\left[\frac{1}{2}\sum_{i=1}^{N} B_i^2\right]$$

Now we need to simplify each of these expectation terms. Looking at each of the individual components

$$\mathbb{E}\left[\frac{1}{2}\sum_{i=1}^{N} A_i^2\right] = \frac{1}{2}\sum_{i=1}^{N}\Big[(w_o + \sum_{j=1}^{D} w_j x_{ij} - t_i)^2\Big]$$

Since each of the terms w x and t are fixed w.r.t to the cost function. Note that this is our original cost function.

$$\mathbb{E}\left[\frac{1}{2}\sum_{i=1}^{N} 2A_iB_i\right] = \mathbb{E}\left[\sum_{i=1}^{N} A_iB_i\right]$$
$$= \mathbb{E}\left[\sum_{i=1}^{N} A_i \sum_{j=1}^{D} w_j\epsilon_{ij}\right]$$
$$= \sum_{i=1}^{N} A_i \sum_{j=1}^{D} w_j \mathbb{E}\Big[\epsilon_{ij}\Big]$$
$$= 0$$

since $\mathbb{E}\Big[\epsilon_{ij}\Big] = 0$

$$\begin{aligned}
\frac{1}{2}\sum_{i=1}^{N} B_i^2 &= \frac{1}{2}\sum_{i=1}^{N}\Big[\sum_{j=1}^{D} w_j\epsilon_{ij}\Big]^2 \\
&= \frac{1}{2}\sum_{i=1}^{N}\Big[\sum_{j=1}^{D}\sum_{k=1}^{D} w_j w_k\epsilon_{ij}\epsilon_{ik}\Big] \\
&= \frac{1}{2}\sum_{i=1}^{N}\Big[\sum_{j=1}^{D} w_j^2\sigma^2\Big] \\
&= \frac{1}{2}\sigma^2\sum_{j=1}^{D} w_j^2
\end{aligned}$$

We take the expectation to get to the third line since.

Bringing it all back together, the expected loss function is given as

$$\mathbb{E}\Big[E_D\Big] = \frac{1}{2}\sum_{i=1}^{N}\Big[(w_o + \sum_{j=1}^{D} w_j x_{ij} - t_i)^2\Big] + \frac{N}{2}\sigma^2\sum_{j=1}^{D} w_j^2 \tag{20}$$

You may have noticed that the regulariser term increases with the number of data points. This is not true. Generally the amount of regularisation should decrease as N increases.

To remove the dependence of N on the regularisation term, in this situation, we can instead minimise the mean of squares error instead of the sum of squares error, so the loss becomes

$$E_D(w) = \frac{1}{2N}\sum_{i=1}^{N}(y(x_i, w) - t_i)^2 \tag{21}$$

i.e. we divide by a factor of N. By doing this, we make the regularisation term independent of the number of data points.

$$\mathbb{E}\Big[E_D\Big] = \frac{1}{2N}\sum_{i=1}^{N}\Big[(w_o + \sum_{j=1}^{D} w_j x_{ij} - t_i)^2\Big] + \frac{1}{2}\sigma^2\sum_{j=1}^{D} w_j^2 \tag{22}$$