ECE 133A HW 1

Lawrence Liu

October 10, 2022

Exercise A1.7

For the optimal coefficents we have:

$$
\begin{aligned}
J &= \frac{1}{n}||c_1\mathbf{1} + c_2a - b||^2 \\
&= \frac{1}{n}||(m_b - m_ac_2)\mathbf{1} + c_2a - b||^2 \\
&= \frac{1}{n}\sum_{k=1}^{n}((m_b - m_ac_2) + c_2a_k - b_k)^2 \\
&= \frac{1}{n}\sum_{k=1}^{n}(c_2(a_k - m_a) - (b_k - m_b))^2 \\
&= \frac{1}{n}\left(c_2^2\sum_{k=1}^{n}(a_k - m_a)^2 - c_2\sum_{k=1}^{n}(a_k - m_a)(b_k - m_b) + \sum_{k=1}^{n}(b_k - m_b)^2\right) \\
&= c_2^2 s_a^2 - \frac{2c_2}{n}(a - m_a\mathbf{1})^T(b - m_b\mathbf{1}) + s_b^2 \\
&= \rho^2 s_b^2 - 2\frac{\rho s_b}{s_a}\rho s_a s_b + s_b^2 \\
&= s_b^2 - \rho^2 s_b^2 \\
&= \boxed{(1 - \rho^2)s_b^2}
\end{aligned}
$$

Exercise A1.8

(a)

We want to maximize J with respect to c_1, so we take the derivative of J with respect to c_1 and find the value of c_1 that makes the derivative equal to zero:

$$\frac{\partial J}{\partial c_1} = \frac{2}{n} \sum_{k=1}^{n} \frac{c_1 + c_2 a_k - b_k}{1 + c^2} = 0$$

$$\frac{1}{n} \sum_{k=1}^{n} (c_1 + c_2 a_k - b_k) = 0$$

$$c_1 + \frac{1}{n} \sum_{k=1}^{n} (c_2 a_k - b_k) = 0$$

$$c_1 + c_2 m_a - m_b = 0$$

$$c_1 = m_b - c_2 m_a$$

(b)

We have for the optimal c_1,

$$J = \frac{||c_2(a - m_a \mathbf{1}) - (b - m_b \mathbf{1})||^2}{n(1 + c_2^2)}$$

$$= \frac{1}{n(1 + c_2^2)} \left(\sum_{k=1}^{n} (c_2 a_k - m_a c_2 - b_k + m_b)^2 \right)$$

$$= \frac{1}{(1 + c_2^2)} \left(c_2^2 s_a^2 + s_b^2 - 2\rho s_a s_b c_2 \right)$$

Taking the derivative with respect to c_2 and setting it equal to zero gives us that the optimal value of c_2 must satisfy:

$$\rho c_2^2 + \left(\frac{s_a}{s_b} - \frac{s_b}{s_a} \right) = 0$$

When $\rho \neq 0$ the optimal solution is one with the same sign as ρ, since that makes $-2\rho s_a s_b c_2$ negative.

(c)

TODO

Exercise A2.4

Expanding $(1 + A)^{n-1}$, we get

$$(1 + A)^{n-1} = \sum_{k=1}^{n-1} A^k$$

Since $A_{i,j}^k$ represents whether a path of length exists from j to i, (with $A_{i,j}^k > 0$ representing that such a path exitst) we can see that the sum of the A^k matrices for $1 \leq k \leq n - 1$ represents all the paths that can possibly exists. Therefore A is irreducible if it possible to travel to one node to any other node, ie that G_A is strongly connected.

Exercise A2.8

We can start by multiplying each x_i subvectors with B, each of these operations takes $(2n - 1)n$ flops, so for the n subvectors it will cost us $(2n - 1)n^2$ flops.

Now we get that for the ith output subvector $y_i = \sum_{j=1}^{n} A_{i,j} B x_j$. Multiplying Ai, j with Bx_j costs us n flops, therefore repeating this n times for $1 \leq i \leq n$ results in n^2 flops. Summing these vectors $A_{i,j} B x_j$ costs $n(n - 1)$ flops. So in total computing this subvector costs us $2n(n - 1)$ flops. We will need to repeat this for each of the n output subvectors, so it will cost use

$2n^2(n-1)$.

Therefore in total, this algorithm will cost us $2n^2(n-1) + (2n-1)n^2$ flops, so approximatley $4n^3$ flops for large n. Compared with $(2n^2-1)n^2$, or $2n^4$ approximatley for large n.