# CS229 Machine Learning Problem Set #3 Solution

Roy C.K. Chan

**Problem 1(a).** Let $l^{(i)} = (o^{(i)} - y^{(i)})^2$ be the loss associated with the $i^{th}$ example. Note that $l^{(i)}$ depends on $w_{1,2}^{[1]}$ in the following manner:

$$w_{1,2}^{[1]} \to z_2^{[1](i)} \to h_2^{(i)} \to z^{[2](i)} \to o^{(i)} \to l^{(i)}\,,$$

where $z_2^{[1](i)}, z^{[2](i)}$ are standard notations (representing values after affine transformation but before non-linearity in a neuron) in class. Therefore, by the chain rule of calculus, we have

$$\begin{aligned}\frac{\partial l^{(i)}}{\partial w_{1,2}^{[1]}} &= \frac{\partial l^{(i)}}{\partial o^{(i)}}\frac{\partial o^{(i)}}{\partial z^{[2](i)}}\frac{\partial z^{[2](i)}}{\partial h_2^{(i)}}\frac{\partial h_2^{(i)}}{\partial z_2^{[1](i)}}\frac{\partial z_2^{[1](i)}}{\partial w_{1,2}^{[1]}} \\ &= 2o^{(i)}o^{(i)}(1-o^{(i)})w_2^{[2]}h_2^{(i)}(1-h_2^{(i)})x_1^{(i)}\,,\end{aligned}$$

where we denote $h_2^{(i)} \triangleq \sigma(w_{0,2}^{[1]} + w_{1,2}^{[1]}x_1^{(i)} + w_{2,2}^{[1]}x_2^{(i)})$ to simplify expressions, and the last equality comes from the derivative of the sigmoid function, i.e., $\sigma' = \sigma(1-\sigma)$. (See lecture notes or PS1 Problem 1(a) for details.) Hence, the gradient descent update is given by

$$w_{1,2}^{[1]} := w_{1,2}^{[1]} - \frac{\alpha}{m}\sum_{i=1}^{m} 2o^{(i)}o^{(i)}(1-o^{(i)})w_2^{[2]}h_2^{(i)}(1-h_2^{(i)})x_1^{(i)}\,.$$

**Problem 1(b).** Consider the triangular region of negative examples (class 0). From the figure, for an example $(x_1, x_2)$ to fall into this region, it has to satisfy the following three inequalities:

$$\begin{aligned} x_1 &\geq 0.25\,, \\ x_2 &\geq 0.25\,, \\ x_2 &\leq -x_1 + 4\,, \end{aligned}$$

which is the same as

$$\begin{aligned} 4x_1 - 1 &\geq 0\,, \\ 4x_2 - 1 &\geq 0\,, \\ -x_1 - x_2 + 4 &\geq 0\,. \end{aligned}$$

(A Very Minor) Remark: There are some negative examples really close to the line $-x_1 - x_2 + 4 = 0$, so it may be better, in terms of "maximizing geometric margin", to choose $-x_1 - x_2 + 4.1 \geq 0$ or $-x_1 - x_2 + 4.2 \geq 0$ as the third inequality, but I decided to use a nicer round number as shown above.

Therefore, we choose

$$w_{0,1}^{[1]} = -1,\ w_{1,1}^{[1]} = 4,\ w_{2,1}^{[1]} = 0\,, \tag{1}$$
$$w_{0,2}^{[1]} = -1,\ w_{1,2}^{[1]} = 0,\ w_{2,2}^{[1]} = 4\,, \tag{2}$$
$$w_{0,3}^{[1]} = 4,\ w_{1,3}^{[1]} = -1,\ w_{2,3}^{[1]} = -1\,. \tag{3}$$

With such weights and the step function $f$, examples inside the triangular region will output $[1\ 1\ 1]^T$ in the hidden layer, which is used as the inputs of the output layer. Then, we choose

$$w_0^{[2]} = 2,\ w_1^{[2]} = -1,\ w_2^{[2]} = -1,\ w_3^{[2]} = -1\,, \tag{4}$$

so that, after this affine transformation, only those points inside the triangle will give a negative number, i.e., -1, whereas points outside the region give a non-negative number, i.e., either 0 or 1 (but 2 is impossible). With the non-linearity step function $f(x) = 1\{x \geq 0\}$, this neural network achieves 100% accuracy on this dataset.

**Problem 1(c).** No. With the activation functions in the hidden layer being the linear function $f(x) = x$, the outputs of the hidden layer (that is, the inputs of the output layer) will be linear in $x_1$ and $x_2$, so this gives a linear decision boundary, but the dataset is not linearly separable. Indeed, it becomes the perceptron algorithm with features $x_1, x_2$ (and the intercept).

**Problem 2.** The derivation of EM for MAP estimation is almost identical to that for MLE shown in class. Consider

$$\begin{aligned}
\log \prod_{i=1}^{m} p(x^{(i)}|\theta)p(\theta) &= \sum_{i=1}^{m} \log p(x^{(i)}|\theta) + \log p(\theta) \\
&= \sum_{i=1}^{m} \log \sum_{z^{(i)}} p(x^{(i)}, z^{(i)}|\theta) + \log p(\theta) \\
&= \sum_{i=1}^{m} \log \sum_{z^{(i)}} Q_i(z^{(i)}) \frac{p(x^{(i)}, z^{(i)}|\theta)}{Q_i(z^{(i)})} + \log p(\theta) \\
&\geq \sum_{i=1}^{m} \sum_{z^{(i)}} Q_i(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)}|\theta)}{Q_i(z^{(i)})} + \log p(\theta)\,,
\end{aligned}$$

where $Q_i$ denotes some distribution over $z_i$ for each $i$ (that is $\sum_z Q_i(z) = 1, Q_i(z) \geq 0$), and the last inequality holds by the fact the logarithmic function is (strictly) concave and Jensen's inequality. To make the bound tight for a particular value of $\theta$, we make the inequality above hold with equality, and require that

$$\frac{p(x^{(i)}, z^{(i)}|\theta)}{Q_i(z^{(i)})} = c\,,$$

for some constant $c$ that does not depend on $z^{(i)}$. It can be easily shown that $c = p(x^{(i)}|\theta)$, which means

$$Q_i(z^{(i)}) = p(z^{(i)}|x^{(i)}, \theta)\,. \tag{5}$$

Therefore, in the E-step, we compute $Q_i$'s according to (**??**), given the training examples $x^{(i)}$'s and the current values of $\theta$'s. And in the M-step, we update $\theta$'s by setting

$$\theta := \arg\max_{\theta} \left( \sum_{i=1}^{m} \sum_{z^{(i)}} Q_i(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)}|\theta)}{Q_i(z^{(i)})} + \log p(\theta) \right). \tag{6}$$

Note that this M-step is tractable because it requires maximizing a linear combination of $\log p(x, z|\theta)$ and $\log p(\theta)$ (plus some terms that do not involve $\theta$).

Next, we prove that $\prod_{i=1}^{m} p(x^{(i)}|\theta)p(\theta)$ monotonically increases with each iteration of the algorithm. Suppose the parameters start out as $\theta^{(t)}$ at some iteration $t$, and we set $Q_i^{(t)}(z^{(i)}) = p(z^{(i)}|x^{(i)}, \theta^{(t)})$ in the E-step, after $\theta^{(t)}$ is updated to $\theta^{(t+1)}$ in the M-step, we have

$$\begin{aligned}
\sum_{i=1}^{m} \log p(x^{(i)}|\theta^{(t+1)})p(\theta^{(t+1)}) &\geq \sum_{i=1}^{m} \sum_{z^{(i)}} Q_i^{(t)}(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)}|\theta^{(t+1)})}{Q_i^{(t)}(z^{(i)})} + \log p(\theta^{(t+1)}) \\
&\geq \sum_{i=1}^{m} \sum_{z^{(i)}} Q_i^{(t)}(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)}|\theta^{(t)})}{Q_i^{(t)}(z^{(i)})} + \log p(\theta^{(t)}) \\
&= \sum_{i=1}^{m} \log p(x^{(i)}|\theta^{(t)})p(\theta^{(t)}) \,.
\end{aligned}$$

The first inequality comes from Jensen's inequality as discussed above (with $Q_i = Q_i^{(t)}$, $\theta = \theta^{(t+1)}$), the second inequality holds since $\theta^{(t+1)}$ solves the maximization problem (**??**) given that $Q_i = Q_i^{(t)}$, and the last equality is true because we have chosen $Q_i^{(t)}(z^{(i)}) = p(z^{(i)}|x^{(i)}, \theta^{(t)})$ in the E-step to make the Jensen's inequality hold with equality at $\theta^{(t)}$.

**Problem 3(a)(i)(ii).** Consider the joint density

$$\begin{aligned}
p(y^{(pr)}, z^{(pr)}, x^{(pr)}) &= p(y^{(pr)}, z^{(pr)})p(x^{(pr)}|y^{(pr)}, z^{(pr)}) \\
&= p(y^{(pr)})p(z^{(pr)})p(x^{(pr)}|y^{(pr)}, z^{(pr)}) \,,
\end{aligned}$$

where the first equality is by the chain rule of probability, and the second equality is by independence between $y^{(pr)}$ and $z^{(pr)}$. As a product of Gaussian densities, we know that the joint density $p(y^{(pr)}, z^{(pr)}, x^{(pr)})$ is a multivariate Gaussian. It suffices to specify its mean vector and covariance matrix.

First, consider the mean vector. We have $E(y^{(pr)}) = \mu_p, E(z^{(pr)}) = \nu_r$. We also re-write $x^{(pr)} = y^{(pr)} + z^{(pr)} + \epsilon^{(pr)}$, where $\epsilon^{(pr)} \sim \mathcal{N}(0, \sigma^2)$ is independent Gaussian noise. So,

$$\begin{aligned}
E(x^{(pr)}) &= E(y^{(pr)} + z^{(pr)} + \epsilon^{(pr)}) \\
&= E(y^{(pr)}) + E(z^{(pr)}) + E(\epsilon^{(pr)}) \\
&= \mu_p + \nu_r + 0 \\
&= \mu_p + \nu_r \,.
\end{aligned}$$

Then, consider the covariance matrix. We have $Var(y^{(pr)}) = \sigma_p^2, Var(z^{(pr)}) = \tau_r^2$, and $Cov(y^{(pr)}, z^{(pr)}) = Cov(z^{(pr)}, y^{(pr)}) = 0$ because of independence between $y^{(pr)}$ and $z^{(pr)}$. Also,

$$\begin{aligned}
Var(x^{(pr)}) &= Var(y^{(pr)} + z^{(pr)} + \epsilon^{(pr)}) \\
&= Var(y^{(pr)}) + Var(z^{(pr)}) + Var(\epsilon^{(pr)}) \\
&= \sigma_p^2 + \tau_t^2 + \sigma^2 \,, \\
Cov(y^{(pr)}, x^{(pr)}) &= Cov(x^{(pr)}, y^{(pr)}) \\
&= Cov(y^{(pr)}, y^{(pr)} + z^{(pr)} + \epsilon^{(pr)}) \\
&= Cov(y^{(pr)}, y^{(pr)}) + 0 + 0 \\
&= \sigma_p^2 \,.
\end{aligned}$$

Note that we make use of indenpence between $y^{(pr)}$, $z^{(pr)}$ and $\epsilon^{(pr)}$ in the above calculations. Similarly, we have $Cov(z^{(pr)}, x^{(pr)}) = Cov(x^{(pr)}, z^{(pr)}) = \tau_r^2$. Therefore,

$$\begin{bmatrix} y^{(pr)} \\ z^{(pr)} \\ x^{(pr)} \end{bmatrix} \sim \mathcal{N}\left( \begin{bmatrix} \mu_p \\ \nu_r \\ \mu_p + \nu_r \end{bmatrix}, \begin{bmatrix} \sigma_p^2 & 0 & \sigma_p^2 \\ 0 & \tau_r^2 & \tau_r^2 \\ \sigma_p^2 & \tau_r^2 & \sigma_p^2 + \tau_r^2 + \sigma^2 \end{bmatrix} \right).$$

Using the rules for conditioning on subsets of jointly Gaussian random variables, we know that $Q_{pr}(y^{(pr)}, z^{(pr)}) = p(y^{(pr)}, z^{(pr)} | x^{(pr)})$ is Gaussian with

$$\mu_{pr} = \begin{bmatrix} \mu_{pr,y} \\ \mu_{pr,z} \end{bmatrix} = \begin{bmatrix} \mu_p + \frac{\sigma_p^2}{\sigma_p^2 + \tau_r^2 + \sigma^2}(x^{(pr)} - \mu_p - \nu_r) \\ \nu_r + \frac{\tau_r^2}{\sigma_p^2 + \tau_r^2 + \sigma^2}(x^{(pr)} - \mu_p - \nu_r) \end{bmatrix}, \tag{7}$$

$$\Sigma_{pr} = \begin{bmatrix} \Sigma_{pr,yy} & \Sigma_{pr,yz} \\ \Sigma_{pr,zy} & \Sigma_{pr,zz} \end{bmatrix} = \frac{1}{\sigma_p^2 + \tau_r^2 + \sigma^2} \begin{bmatrix} \sigma_p^2(\tau_r^2 + \sigma^2) & -\sigma_p^2\tau_r^2 \\ -\sigma_p^2\tau_r^2 & \tau_r^2(\sigma_p^2 + \sigma^2) \end{bmatrix}. \tag{8}$$

**Problem 3(b).** Let $\theta = \{\mu_p, \nu_r, \sigma_p^2, \tau_r^2\}$ be the collection of parameters we want to estimate. In the M-step,

$$
\begin{aligned}
\theta &= \arg\max_\theta \sum_{p=1}^P \sum_{r=1}^R E_{(y^{(pr)}, z^{(pr)}) \sim Q_{pr}} \log p(y^{(pr)}, z^{(pr)}, x^{(pr)}; \theta) \\
&= \arg\max_\theta \sum_{p=1}^P \sum_{r=1}^R E \log \left( \frac{1}{\sqrt{2\pi}\sigma_p} e^{-\frac{(y^{(pr)}-\mu_p)^2}{2\sigma_p^2}} \cdot \frac{1}{\sqrt{2\pi}\tau_r} e^{-\frac{(z^{(pr)}-\nu_r)^2}{2\tau_r^2}} \cdot \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x^{(pr)}-y^{(pr)}-z^{(pr)})^2}{2\sigma^2}} \right) \\
&= \arg\max_\theta \sum_{p=1}^P \sum_{r=1}^R E \left( \log \frac{1}{\sigma_p \tau_r \sigma} - \frac{(y^{(pr)}-\mu_p)^2}{2\sigma_p^2} - \frac{(z^{(pr)}-\nu_r)^2}{2\tau_r^2} - \frac{(x^{(pr)}-y^{(pr)}-z^{(pr)})^2}{2\sigma^2} \right) \\
&= \arg\max_\theta \sum_{p=1}^P \sum_{r=1}^R E \left( \log \frac{1}{\sigma_p \tau_r} - \frac{(y^{(pr)}-\mu_p)^2}{2\sigma_p^2} - \frac{(z^{(pr)}-\nu_r)^2}{2\tau_r^2} \right) \\
&= \arg\max_\theta \sum_{p=1}^P \sum_{r=1}^R E \left( \log \frac{1}{\sigma_p \tau_r} - \frac{\big((y^{(pr)}-\mu_{pr,y}) + (\mu_{pr,y}-\mu_p)\big)^2}{2\sigma_p^2} - \frac{\big((z^{(pr)}-\mu_{pr,z}) + (\mu_{pr,z}-\nu_r)\big)^2}{2\tau_r^2} \right) \\
&= \arg\max_\theta \sum_{p=1}^P \sum_{r=1}^R \left( \log \frac{1}{\sigma_p \tau_r} - \frac{\Sigma_{pr,yy} + (\mu_{pr,y}-\mu_p)^2}{2\sigma_p^2} - \frac{\Sigma_{pr,zz} + (\mu_{pr,y}-\nu_r)^2}{2\tau_r^2} \right) \\
&= \arg\max_\theta \sum_{p=1}^P \sum_{r=1}^R \left( \log \frac{1}{\sigma_p} + \log \frac{1}{\tau_r} - \frac{\Sigma_{pr,yy} + \mu_{pr,y}^2 - 2\mu_{pr,y}\mu_p + \mu_p^2}{2\sigma_p^2} - \frac{\Sigma_{pr,zz} + \mu_{pr,y}^2 - 2\mu_{pr,y}\nu_r + \nu_r^2}{2\tau_r^2} \right).
\end{aligned}
$$

In the above calculations, the third and fourth equalities hold by dropping terms that do not depend on $\theta$, while the sixth equality is obtained by (i) expanding the numerators; (ii) taking expectations w.r.t. the distribution $(y^{(pr)}, z^{(pr)}) \sim Q_{pr}$; and (iii) noticing that $E(y^{(pr)} - \mu_{pr,y})^2 = Var(y^{(pr)}) = \Sigma_{pr,yy}$, $E(z^{(pr)} - \mu_{pr,y})^2 = Var(z^{(pr)}) = \Sigma_{pr,zz}$, and the cross terms vanish because $E(y^{(pr)}) = \mu_{pr,y}$ and $E(z^{(pr)}) = \mu_{pr,z}$.

For each $p = 1, \cdots, P, r = 1, \cdots, R$, differentiate w.r.t. $\mu_p, \nu_r, \sigma_p, \tau_r$, and set the derivatives to zeros,

$$
-\frac{1}{2\sigma_p^2} \sum_{r=1}^R (2\mu_p - 2\mu_{pr,y}) = 0 \quad \Longrightarrow \quad \mu_p = \frac{1}{R} \sum_{r=1}^R \mu_{pr,y}\,, \tag{9}
$$

$$
-\frac{1}{2\tau_r^2} \sum_{p=1}^P (2\nu_r - 2\mu_{pr,z}) = 0 \quad \Longrightarrow \quad \nu_r = \frac{1}{P} \sum_{p=1}^P \mu_{pr,z}\,, \tag{10}
$$

$$
\sum_{r=1}^R \Big( -\frac{1}{\sigma_p} + \frac{\Sigma_{pr,yy} + \mu_{pr,y}^2 - 2\mu_{pr,y}\mu_p + \mu_p^2}{\sigma_p^3} \Big) = 0 \quad \Longrightarrow \quad \sigma_p^2 = \frac{1}{R} \sum_{r=1}^R \Big( \Sigma_{pr,yy} + (\mu_{pr,y} - \mu_p)^2 \Big), \tag{11}
$$

$$
\sum_{p=1}^P \Big( -\frac{1}{\tau_r} + \frac{\Sigma_{pr,zz} + \mu_{pr,z}^2 - 2\mu_{pr,z}\nu_r + \nu_r^2}{\tau_r^3} \Big) = 0 \quad \Longrightarrow \quad \tau_r^2 = \frac{1}{P} \sum_{p=1}^P \Big( \Sigma_{pr,zz} + (\mu_{pr,z} - \nu_r)^2 \Big). \tag{12}
$$

Hence, the E-step is to compute $\mu_{pr}, \Sigma_{pr}$ using equations (**??**), (**??**) for each $p, r$; while the M-step is to compute $\mu_p, \nu_r, \sigma_p^2, \tau_r^2$ using the above four equations (in this order) for each $p, r$.

Remark: At the $t^{th}$ (current) update of EM algorithm, the conditional distribution $Q_{pr}^{(t)}$ actually depends on $\theta$, as indicated by equations (**??**) and (**??**). However, those parameter values are computed in the M-step of the $(t-1)^{th}$ (previous) update. Therefore, after the E-step of the current update, $Q_{pr}^{(t)}$ is used as a fixed distribution (that does not depend on $\theta$) in the current M-step.

**Problem 4(a).** To prove nonnegativity,

$$\begin{aligned}
\forall P, Q \quad KL(P||Q) &= \sum_x P(x) \log \frac{P(x)}{Q(x)} \\
&= \sum_x P(x) \Big[ -\log \frac{Q(x)}{P(x)} \Big] \\
&= \mathbb{E}_{X \sim P} \Big[ -\log \frac{Q(X)}{P(X)} \Big] \\
&\geq -\log \mathbb{E}_{X \sim P} \Big[ \frac{Q(X)}{P(X)} \Big] \\
&= -\log \sum_x P(x) \frac{Q(x)}{P(x)} \\
&= -\log \sum_x Q(x) \\
&= -\log 1 \\
&= 0\,,
\end{aligned}$$

where the inequality is due to Jensen's inequality and the fact that $f(x) = -\log x$ is a (strictly) convex function, and the second last equality holds because $Q(x)$ is a probability mass function over $x$ such that it sums to one.

Obviously, $P = Q \implies KL(P||Q) = 0$, so it suffices to prove that $KL(P||Q) = 0 \implies P = Q$. $KL(P||Q) = 0$ implies the above inequality holds with equality, which means

$$\frac{Q(X)}{P(X)} = C\,,$$

for some constant $C$. Rearranging terms and summing over x, we have

$$\begin{aligned}
\sum_x Q(x) &= C \sum_x P(x) \\
\implies C &= 1\,, \\
\implies P &= Q\,.
\end{aligned}$$

Hence, $KL(P||Q) = 0$ if and only if $P = Q$.

**Problem 4(b).** We start from the RHS.

$$
\begin{aligned}
& KL(P(X)||Q(X)) + KL(P(Y|X)||Q(Y|X)) \\
&= \sum_x P(x)\log\frac{P(x)}{Q(x)} + \sum_x P(x)\Big(\sum_y P(y|x)\log\frac{P(y|x)}{Q(y|x)}\Big) \\
&= \sum_x P(x)\log\frac{P(x)}{Q(x)} + \sum_x P(x)\Big(\sum_y \frac{P(x,y)}{P(x)}\log\Big[\frac{P(x,y)}{P(x)}\cdot\frac{Q(x)}{Q(x,y)}\Big]\Big) \\
&= \sum_x P(x)\log\frac{P(x)}{Q(x)} + \sum_x\sum_y P(x,y)\Big(\log\frac{P(x,y)}{Q(x,y)} + \log\frac{Q(x)}{P(x)}\Big) \\
&= \sum_x P(x)\log\frac{P(x)}{Q(x)} + \sum_x\sum_y P(x,y)\log\frac{P(x,y)}{Q(x,y)} + \sum_x\Big(\sum_y P(x,y)\Big)\log\frac{Q(x)}{P(x)} \\
&= \sum_x P(x)\log\frac{P(x)}{Q(x)} + \sum_x\sum_y P(x,y)\log\frac{P(x,y)}{Q(x,y)} - \sum_x P(x)\log\frac{P(x)}{Q(x)} \\
&= \sum_x\sum_y P(x,y)\log\frac{P(x,y)}{Q(x,y)} \\
&= \sum_{x,y} P(x,y)\log\frac{P(x,y)}{Q(x,y)} \\
&= KL(P(X,Y)||Q(X,Y))\,.
\end{aligned}
$$

Note that, in the fourth last line, we use the facts that (i) $\sum_y P(x,y) = P(x)$ and (ii) $\log\frac{Q(x)}{P(x)} = -\log\frac{P(x)}{Q(x)}$.

**Problem 4(c).** With the training set $\{x^{(i)}; i = 1, \cdots, m\}$ and the empirical distribution $\hat{P}(x) = \frac{1}{m}\sum_{i=1}^m 1\{x^{(i)} = x\}$,

$$
\begin{aligned}
\arg\min_\theta KL(\hat{P}||P_\theta) &= \arg\min_\theta \sum_x \hat{P}(x)\log\frac{\hat{P}(x)}{P_\theta(x)} \\
&= \arg\min_\theta \Big(\sum_x \hat{P}(x)\log\hat{P}(x) - \sum_x \hat{P}(x)\log P_\theta(x)\Big) \\
&= \arg\min_\theta \Big(-\sum_x \hat{P}(x)\log P_\theta(x)\Big) \\
&= \arg\max_\theta \sum_x \hat{P}(x)\log P_\theta(x) \\
&= \arg\max_\theta \sum_x \frac{1}{m}\sum_{i=1}^m 1\{x^{(i)} = x\}\log P_\theta(x) \\
&= \arg\max_\theta \sum_{i=1}^m \sum_x 1\{x^{(i)} = x\}\log P_\theta(x) \\
&= \arg\max_\theta \sum_{i=1}^m \log P_\theta(x^{(i)})\,,
\end{aligned}
$$

where the third equality holds because we drop the terms that do not depend on $\theta$, and the second last equality is obtained by (i) dropping the positive constant factor $\frac{1}{m}$, and (ii) interchanging the order of summations (one can always interchange the order of a finite sum and an infinite sum, as a special case of the Fubini's Theorem).

**Problem 5(a)(b)(c).** See the files "kmeans.py" and "Q5.ipynb".

**Problem 5(d).** The original image needs 24 bits per pixel, whereas the compressed image requires only 4 bits ($2^4 = 16$ colors) per pixel to store which cluster that pixel belongs to. Therefore, we compress the image by approximately a factor of $24/4 = 6$. (There are of course some overhead costs of storing those centroids, but they are minimal given an image size of $128 \times 128$ or $512 \times 512$.)