1 EM for supervised learning

a) The log likelihood function is given by

$$\begin{aligned}\ell(\phi, \theta_0, \theta_1) &= \log \prod \exp\left(\frac{-\left(y^{(i)} - \theta_z^T x^{(i)}\right)^2}{2\sigma^2}\right) g\left(\phi^T x^{(i)}\right)^z \left(1 - g\left(\phi^T x^{(i)}\right)\right)^{1-z} \\ &= \sum \frac{-\left(y^{(i)} - \theta_z^T x^{(i)}\right)^2}{2\sigma^2} + z \log g\left(\phi^T x^{(i)}\right) + (1-z)\log\left(1 - g\left(\phi^T x^{(i)}\right)\right).\end{aligned}$$

With $z = 0$, this becomes

$$\ell_0\left(\theta_0, \phi\right) = \sum \frac{-\left(y^{(i)} - \theta_0^T x^{(i)}\right)^2}{2\sigma^2} + \log\left(1 - g\left(\phi^T x^{(i)}\right)\right).$$

Differentiating with respect to θ_0, we get

$$\frac{\partial \ell_0}{\theta_{0,k}} = \sum -\left(y^{(i)} - \theta_0^T x^{(i)}\right) x_k^{(i)}$$

Setting equal to zero and writing in matrix notation, this becomes

$$X_0^T\left(Y_0 - X\theta_0\right) = 0,$$

where X_0, Y_0 are the rows in X, Y that correspond to $z = 0$. Solving for θ_0 yields the familiar normal equations,

$$\theta_0 = \left(X_0^T X_0\right)^{-1} X_0^T Y_0.$$

The derivation for θ_1 is identical.

Turning to ϕ, we remove terms from the likelihood that do not contain ϕ, to get

$$f\left(\phi\right) = \sum z \log g\left(\phi^T x^{(i)}\right) + (1-z)\log\left(1 - g\left(\phi^T x^{(i)}\right)\right).$$

Taking the kth partial derivative with respect to ϕ,

$$\begin{aligned}\frac{\partial f}{\partial \phi_k} &= \sum\left(\frac{z^{(i)}}{g\left(\phi^T x^{(i)}\right)} - \frac{\left(1 - z^{(i)}\right)}{1 - g\left(\phi^T x^{(i)}\right)}\right) g\left(\phi^T x^{(i)}\right)\left(1 - g\left(\phi^T x^{(i)}\right)\right) x_k^{(i)} \\ &= \sum\left(z^{(i)} - g\left(\phi^T x^{(i)}\right)\right) x_k^{(i)}.\end{aligned}$$

In matrix form, $\nabla f_\phi = X^T H$, where $H = Z - G$, $Z_i = z^{(i)}$, and $H_i = g\left(\phi^T x^{(i)}\right)$.

To derive the Hessian of f, we take the jth partial derivative of ∇f_ϕ, to get

$$\frac{\partial \nabla f_\phi}{\partial \phi_j} = -\sum g\left(\phi^T x^{(i)}\right)\left(1 - g\left(\phi^T x^{(i)}\right)\right) x_k^{(i)} x_j^{(i)}.$$

In matrix notation, this is $X^T D X$, where D is a diagonal matrix with $D_{ii} = g\left(\phi^T x^{(i)}\right)\left(g\left(\phi^T x^{(i)}\right) - 1\right)$.

b) In the E-step, we calculate $p\left(z^{(i)} \mid y^{(i)}, x^{(i)}, \varphi\right)$, where $\varphi$is a catch-all for the parameters in the model. Using Bayes' Theorem, this becomes

$$p\left(z^{(i)} = j \mid y^{(i)}, x^{(i)}, \varphi\right) = \frac{p\left(y^{(i)} \mid x^{(i)}, z(i) = j, \varphi\right) p\left(z^{(i)} = j \mid x^{(i)}, \varphi\right)}{\sum_j p\left(y^{(i)} \mid x^{(i)}, z(i) = j, \varphi\right) p\left(z^{(i)} = j \mid x^{(i)}, \varphi\right)} = w_j^{(i)}.$$

For the M-step, we first write the likelihood as

$$\ell\left(\theta_0, \theta_1, \phi\right) = \prod \sum w_j^{(i)} p\left(y^{(i)} \mid x^{(i)}, z^{(i)} = j, \theta_0, \theta_1, \phi\right).$$

Taking logs and substituting, this becomes

$$\begin{aligned}
\log \ell(\theta_0, \theta_1, \phi) &= \sum_{i=1}^{m} \sum_{j=0}^{1} w_j^{(i)} \log \exp\left(-\left(y^{(i)} - \theta_j^T x^{(i)}\right)^2\right) \\
&= -\sum_{i=1}^{m}\left[w_0^{(i)}\left(y^{(i)} - \theta_0^T x^{(i)}\right)^2 + w_1^{(i)}\left(y^{(i)} - \theta_1^T x^{(i)}\right)^2\right].
\end{aligned}$$

Taking the kth partial derivitative of θ_0, we get

$$\frac{\partial \log \ell\left(\theta_0, \theta_1, \phi\right)}{\partial \theta_{0,k}} = -\sum_{i=1}^{m} w_0^{(i)}\left(y^{(i)} - \theta_0^T x^{(i)}\right) x_k^{(i)}.$$

In matrix notation this becomes $X^T W_0 \left(Y - X\theta_0\right)$, where W_0 is a diagonal matrix with $W_{i,i} = w_0^{(i)}$. Solving for $\theta_{0,}$ we get the familiar normal equations for weighted regression,

$$\theta_0 = \left(X_0^T W_0 X_0\right)^{-1} X_0^T W_0 Y_0.$$

The derivation for θ_1 is identical.

Turning to ϕ, the log likelihood is the same form as the logistic regression above, but with z replaced by w. Therefore, the gradient and Hessian are of the same form as above. In particular, the gradient is $X^T H$, where H is a diagonal matrix with $H_{i,i} = w^{(i)} - g\left(\phi^T x^{(i)}\right)$, and the Hessian is $X^T D X$, where D is a diagonal matrix with $D_{ii} = g\left(\phi^T x^{(i)}\right)\left(g\left(\phi^T x^{(i)}\right) - 1\right)$.

2 Factor analysis and PCA

a) First calculate the means of z and x. By definition, $\mathbb{E}[z] = 0$. We can write x as

$$x = Uz + \epsilon,$$

where $\epsilon \sim N\left(0, \sigma^2 I\right)$. Calculating the expectation of the right side gives

$$\mathbb{E}[Uz + \epsilon] = U\mathbb{E}[z] + \mathbb{E}[\epsilon] = U \cdot 0 + 0 = 0.$$

By definition, $Var[z] = 1$, so $\Sigma_{zz} = 1$. To get Σ_{zx}, we calculate

$$\begin{aligned}
\mathbb{E}\left[(z - \mathbb{E}[z])(x - \mathbb{E}[x])^T\right] &= \mathbb{E}\left[z(Uz + \epsilon)^T\right] \\
&= \mathbb{E}\left[z\left(z^T U^T + \epsilon^T\right)\right] \\
&= \mathbb{E}\left[z\left(z^T U^T + \epsilon^T\right)\right] \\
&= \mathbb{E}\left[zz^T\right] U^T + \mathbb{E}\left[z^T \epsilon\right] \\
&= U^T.
\end{aligned}$$

It follows that $\Sigma_{xz} = U$. Finally, to get Σ_{xx}, we calculate

$$\begin{aligned}
\mathbb{E}\left[(x - \mathbb{E}[x])(x - \mathbb{E}[x])^T\right] &= \mathbb{E}\left[(Uz + \epsilon)(Uz + \epsilon)^T\right] \\
&= \mathbb{E}\left[Uzz^T U^T + Uz\epsilon^T + \epsilon\epsilon^T\right] \\
&= UU^T + \sigma^2 I.
\end{aligned}$$

The joint distribution of z and x is therefore given by

$$\begin{bmatrix} z \\ x \end{bmatrix} \sim N\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & U^T \\ U & UU^T + \sigma^2 I \end{bmatrix}\right).$$

To find $\mu_{z|x}$ and $\Sigma_{z|x}$, we plug the values above into the general formulas for the conditional mean and variance of multivariate normal variables. For $\mu_{x|z}$, we have

$$\begin{aligned}
\mu_{z|x} &= \mu_z + \Sigma_{zx}\Sigma_{xx}^{-1}(x - \mu_x) \\
&= U^T \left[UU^T + \sigma^2 I\right]^{-1} x \\
&= \left[U^T U + \sigma^2 I\right]^{-1} U^T x.
\end{aligned}$$

For $\Sigma_{z|x}$, we have

$$\begin{aligned}
\Sigma_{z|x} &= \Sigma_{zz} - \Sigma_{zx}\Sigma_{xx}^{-1}\Sigma_{xz} \\
&= 1 - U^T \left[UU^T + \sigma^2 I\right]^{-1} U \\
&= 1 - \left[U^T U + \sigma^2 I\right]^{-1} U^T U.
\end{aligned}$$

b) For the E-step, $Q_i\left(z^{(i)}\right)$ is equal to a multivariate normal distribution with parameters $\mu_{z|x}$ and $\Sigma_{z|x}$. For the M-step, we observe that this problem is equivalent to the general factor anaysis problem, with $\mu = 0$ and $\Lambda = \sigma^2 I$. We can therefore plug the values above into the M updated step for factor analysis to get

$$U = \left(\sum \left(x^{(i)}\right) \mu^T_{z^{(i)}|x^{(i)}}\right)\left(\sum \mu_{z^{(i)}|x^{(i)}} \mu^T_{z^{(i)}|x^{(i)}} + \Sigma_{z^{(i)}|x^{(i)}}\right)^{-1}.$$

c) As $\sigma^2 \to 0$, $\mu_{z|x} \to \left[U^T U\right]^{-1} U^T x = U^{-1}x$, and $\Sigma_{z|x} \to 1 - \left[U^T U\right]^{-1} U^T U = 1$. Now we can re-write the M step update in matrix form and substitute these values for $\mu_{z|x}$ and $\Sigma_{z|x}$ as

$$\begin{aligned} U &= XX^T U^{-T}\left(U^{-1} XX^T U^{-T}\right)^{-1} \\ &= U^T U^{-T}\left(U^{-1} U^T U^{-T}\right)^{-1} \\ &= U. \end{aligned}$$

The first line is the translation of the update step into matrix notation. The second line follows from the fact that $X^T X = \Sigma_X$ (because $\mu_x = 0$), and $\Sigma_{xx} = U$, as derived above. So the M step update leaves U unchanged, as required.

3 PCA and ICA for Natural Images

I skipped this problem because it was too tedious to convert the Matlab/Octave code into Python or R.

4 Convergence of Policy Iterations

a) From the definition of B, we can write

$$\begin{aligned} B\left(V_1\right) - B\left(V_2\right) &= \gamma \sum_{s' \in S} P_{s,\pi(s)}\left(s'\right)\left[V_2\left(s'\right) - V_1\left(s'\right)\right] \\ &\geq \gamma\delta \\ &\geq 0, \end{aligned}$$

where $\delta = min_{s \in S}\left\{V_2\left(s\right) - V_1\left(s\right)\right\} \geq 0$, by assumption.

b) Let $s^* = \arg\max_{s \in S}\left\{V\left(s\right) - V^\pi\left(s\right)\right\}$. Let

$$f\left(s\right) = \gamma \sum_{s' \in S} P_{s,\pi(s)}\left(s'\right)\left[V\left(s'\right) - V^\pi\left(s'\right)\right]$$

for $s \in S$. Then from the hint, it follows that $f\left(s\right) \leq V\left(s^*\right) - V^\pi\left(s^*\right)$. Intuitively, what the hint says is that the average of a set of numbers is less than (or equal) to the largest number in the set. The result follows.

c) Per the hint, we'll first show that $V^\pi(s) \leq B^{\pi'}(V^\pi)(s)$. Subtracting, we get

$$
\begin{aligned}
B^{\pi'}(V^\pi)(s) - V^\pi(s) &= \gamma \sum_{s' \in S} P_{s,\pi'(s)}(s')V^\pi(s') - P_{s,\pi(s)}(s')V^\pi(s') \\
&\geq 0,
\end{aligned}
$$

where the inequality follows from the definition of π'.

Let $B_n^{\pi'}$ indicate the application of $B^{\pi'}$ n times. From part a) above, it follows that

$$V^\pi(s) \leq B^{\pi'} V^\pi(s) \leq B_2(V^{\pi'})(s) \leq B_n(V^{\pi'})(s).$$

In addition, as $n \to \infty, B(V^{\pi'})(s) \to V^{\pi'}(s)$. The result follows.

d) Assuming a finite number of states, $|S|$, and actions, $|A|$, there are a finite number of policies. We showed in part c) that the policies are monotonically improving, so we will stop improving our policy after at most $|S|^{|A|}$ iterations. Optimality follows from the same logic as we used in part c).

5 Reinforcement Learning: The Mountain Car

Didn't do this one. Again, too tedious to translate Matlab into Python or R.