1.

For a set of points $\{x^{(1)}, ..., x^{(n)}\}$, unit vector u, and scalar $\alpha \in \mathbf{R}$, we have the expanded norm

$$||x - \alpha u||^2 = ||x||^2 - 2\alpha\langle x, u\rangle + \alpha^2.$$

If x and u are fixed, then this is a quadratic form that has a minimum iff $\alpha = \langle x, u\rangle$. In this case, if $\mathbf{V} = \{\alpha u : \alpha \in \mathbf{R}\}$,

$$f_u(x) = argmin_{v\in\mathbf{V}}||x - v||^2 = \langle x, u\rangle u.$$

Now we examine the mean squared error between the projected points and the original points corresponding to the first principal component. Using the above, we get

$$\begin{aligned}
\sum_{i=1}^{n} ||x^{(i)} - f_u(x^{(i)})||^2 &= \sum_{i=1}^{n} \left(||x^{(i)}||^2 - 2\langle x^{(i)}, f_u(x^{(i)})\rangle + ||f_u(x^{(i)})||^2\right) \\
&= \sum_{i=1}^{n} \left(||x^{(i)}||^2 - 2\langle x^{(i)}, \langle x^{(i)}, u\rangle u\rangle + ||\langle x^{(i)}, u\rangle u||^2\right) \\
&= \sum_{i=1}^{n} \left(||x^{(i)}||^2 - 2\langle x^{(i)}, u\rangle^2 + \langle x^{(i)}, u\rangle^2\right) \\
&= \sum_{i=1}^{n} \left(||x^{(i)}||^2 - \langle x^{(i)}, u\rangle^2\right).
\end{aligned}$$

Therefore, we find that

$$argmin_{u:u^Tu=1} \sum_{i=1}^{n} |x^{(i)} - f_u(x^{(i)})|_2^2 = argmax_{u:u^Tu=1} \sum_{i=1}^{n} \langle x^{(i)}, u\rangle^2$$

which is the first principal component. This u is the unit vector that minimizes the MSE between projected points in setting up PCA.

2.a

$$\begin{aligned}
\ell(\theta^{(t+1)}) &= \alpha\ell_{\mathsf{sup}}(\theta^{(t+1)}) + \ell_{\mathsf{unsup}}(\theta^{(t+1)}) && \text{Definition} \\
&\geq \alpha\ell_{\mathsf{sup}}(\theta^{(t+1)}) + \sum_{i=1}^{n}\sum_{z^{(i)}} Q_i^{(t)}(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)}; \theta^{(t+1)})}{Q_i^{(t)}(z^{(i)})} && \text{Jensen's inequality} \\
&\geq \alpha \sum_{i=1}^{n} \log p(\tilde{x}^{(i)}, \tilde{z}^{(i)}; \theta^{(t+1)} + \sum_{i=1}^{n}\sum_{z^{(i)}} Q_i^{(t)}(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)}; \theta^{(t+1)})}{Q_i^{(t)}(z^{(i)})} && \text{Definition} \\
&\geq \sum_{i=1}^{n} \mathsf{ELBO}(x^{(i)}; Q_i^{(t)}, \theta^{(t)})
\end{aligned}$$

The final inequality follows from the fact that $\theta^{(t+1)}$ is chosen explicitly to be

$$\mathsf{argmax}_\theta \sum_{i=1}^{n}\sum_{z^{(i)}} Q_i^{(t)}(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)}; \theta^{(t+1)})}{Q_i^{(t)}(z^{(i)})}$$

by the definitions given by the problem statement. This is the same logic provided on page 7 of the lecture notes.

2.b

The only latent variables that we have to estimate in the E-step are $z^{(i)}$ because the supervised labels, $\tilde{z}^{(i)}$, are observed and therefore not unknown. We derive according to the following:

$$\begin{aligned} w_j^{(i)} &= \frac{p(z^{(i)} = j, x^{(i)}; \phi, \mu, \Sigma)}{p(x^{(i)}; \phi, \mu, \Sigma)} \\ &= \frac{p(z^{(i)} = j; \phi, \mu, \Sigma) p(x^{(i)} | z^{(i)} = j; \phi, \mu, \Sigma)}{\sum_{\ell=1}^{k} p(z^{(i)} = \ell; \phi, \mu, \Sigma) p(x^{(i)} | z^{(i)} = \ell; \phi, \mu, \Sigma)} \\ &= \frac{\frac{1}{(2\pi)^{d/2}|\Sigma_j|^{1/2}} \exp\left(-\frac{1}{2}(x^{(i)} - \mu_j)^T \Sigma_j^{-1}(x^{(i)} - \mu_j)\right) \phi_j}{\sum_{\ell=1}^{k} \frac{1}{(2\pi)^{d/2}|\Sigma_\ell|^{1/2}} \exp\left(-\frac{1}{2}(x^{(i)} - \mu_\ell)^T \Sigma_\ell^{-1}(x^{(i)} - \mu_\ell)\right) \phi_\ell}. \end{aligned}$$

Hence, we predict $z^{(i)}$ for $i = 1...n, j = 1...k$ in terms of x, z, μ, Σ, ϕ, and universal constants. This allows us to compute all terms and re-estimate $w_j^{(i)}$ in subsequent steps.

2.c

List the parameters which need to be re-estimated in the M-step:

$$\mu_j, \Sigma_j, \phi_j \text{ for } j = 1, ..., k.$$

In order to simplify derivation, it is useful to denote

$$w_j^{(i)} = Q_i^{(t)}(z^{(i)} = j),$$

and

$$\tilde{w}_j^{(i)} = \begin{cases} \alpha & \tilde{z}^{(i)} = j \\ 0 & \text{otherwise.} \end{cases}$$

We further denote $S = \Sigma^{-1}$, and note that because of chain rule of calculus, $\nabla_S \ell = 0 \Rightarrow \nabla_\Sigma \ell = 0$. So we choose to rewrite the M-step in terms of S and maximize it w.r.t S, and re-express the resulting solution back in terms of Σ.

Based on this, the M-step becomes:

$$\begin{aligned}
\phi^{(t+1)}, \mu^{(t+1)}, S^{(t+1)} &= \arg\max_{\phi,\mu,S} \sum_{i=1}^{n}\sum_{j=1}^{k} Q_i^{(t)}(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)}; \phi, \mu, S)}{Q_i^{(t)}(z^{(i)})} + \alpha \sum_{i=1}^{\tilde{n}} \log p(\tilde{x^{(i)}}, \tilde{z^{(i)}}; \phi, \mu, S) \\
&= \arg\max_{\phi,\mu,S} \sum_{i=1}^{n}\sum_{j=1}^{k} w_j^{(i)} \log \frac{\frac{1}{(2\pi)^{d/2}|\Sigma|_j^{1/2}} \exp\left(-\frac{1}{2}(x^{(i)} - \mu_j)^T S_j (x^{(i)} - \mu_j)\right) \phi_j}{w_j^{(i)}} \\
&+ \alpha \sum_{i=1}^{\tilde{n}}\sum_{j=1}^{k} \log \frac{1}{(2\pi)^{d/2}|\Sigma|_j^{1/2}} \exp\left(-\frac{1}{2}(\tilde{x^{(i)}} - \mu_j)^T S_j (\tilde{x^{(i)}} - \mu_j)\right) \phi_j.
\end{aligned}$$

Now, calculate the update steps by maximizing the expression within the argmax for each parameter (We will do the first for you).

ϕ_j: We construct the Lagrangian including the constraint that $\sum_{j=1}^{k} \phi_j = 1$, and absorbing all irrelevant terms into constant C:

$$\begin{aligned}
\mathcal{L}(\phi, \beta) &= C + \sum_{i=1}^{n}\sum_{j=1}^{k} w_j^{(i)} \log \phi_j + \sum_{i=1}^{\tilde{n}}\sum_{j=1}^{k} \tilde{w}_j^{(i)} \log \phi_j + \beta\left(\sum_{j=1}^{k} \phi_j - 1\right) \\
\nabla_{\phi_j} \mathcal{L}(\phi, \beta) &= \sum_{i=1}^{n} w_j^{(i)} \frac{1}{\phi_j} + \sum_{i=1}^{\tilde{n}} \tilde{w}_j^{(i)} \frac{1}{\phi_j} + \beta = 0 \\
&\Rightarrow \phi_j = \frac{\sum_{i=1}^{n} w_j^{(i)} + \sum_{i=1}^{\tilde{n}} \tilde{w}_j^{(i)}}{-\beta} \\
\nabla_\beta \mathcal{L}(\phi, \beta) &= \sum_{j=1}^{k} \phi_j - 1 = 0 \\
&\Rightarrow \sum_{j=1}^{k} \frac{\sum_{i=1}^{n} w_j^{(i)} + \sum_{i=1}^{\tilde{n}} \tilde{w}_j^{(i)}}{-\beta} = 1 \\
&\Rightarrow -\beta = \sum_{j=1}^{k}\left(\sum_{i=1}^{n} w_j^{(i)} + \sum_{i=1}^{\tilde{n}} \tilde{w}_j^{(i)}\right) \\
\Rightarrow \phi_j^{(t+1)} &= \frac{\sum_{i=1}^{n} w_j^{(i)} + \sum_{i=1}^{\tilde{n}} \tilde{w}_j^{(i)}}{\sum_{j=1}^{k}\left(\sum_{i=1}^{n} w_j^{(i)} + \sum_{i=1}^{\tilde{n}} \tilde{w}_j^{(i)}\right)} \\
&= \frac{\sum_{i=1}^{n} w_j^{(i)} + \sum_{i=1}^{\tilde{n}} \tilde{w}_j^{(i)}}{n + \alpha\tilde{n}}
\end{aligned}$$

μ_j: Next, derive the update for μ_j. Do this by maximizing the expression with the argmax above with respect to μ_j.

First, calculate the gradient with respect to μ_j:

$$\begin{aligned}\nabla_{\mu_j} &= -\nabla_{\mu_j}\left(\sum_{i=1}^{n}\sum_{j=1}^{k} w_j^{(i)}\frac{1}{2}(x^{(i)}-\mu_j)^T S_j(x^{(i)}-\mu_j) + \sum_{i=1}^{\tilde{n}}\sum_{j=1}^{k}\tilde{w}_j^{(i)}\frac{1}{2}(\tilde{x^{(i)}}-\mu_j)^T S_j(x^{(i)}-\mu_j)\right)\\ &= \frac{1}{2}\sum_{i=1}^{n} w_j^{(i)}\nabla_{\mu_j}(\mu_j^T S_j x^{(i)} - \mu_j^T S_j\mu_j) + \frac{1}{2}\sum_{i=1}^{\tilde{n}}\tilde{w}_j^{(i)}\nabla_{\mu_j}(\mu_j^T S_j \tilde{x^{(i)}} - \mu_j^T S_j\mu_j)\\ &= \sum_{i=1}^{n} w_j^{(i)}(S_j x^{(i)} - S_j\mu_j) + \sum_{i=1}^{\tilde{n}}\tilde{w}_j^{(i)}(S_j\tilde{x^{(i)}} - S_j\mu_j).\end{aligned}$$

Next, set the gradient to zero and solve for μ_j:

$$\begin{aligned}0 &= \sum_{i=1}^{n} w_j^{(i)}(S_j x^{(i)} - S_j\mu_j) + \sum_{i=1}^{\tilde{n}}\tilde{w}_j^{(i)}(S_j\tilde{x^{(i)}} - S_j\mu_j)\\ \implies \mu_j &= \frac{\sum_{i=1}^{n} w_j^{(i)}x^{(i)} + \sum_{i=1}^{n}\tilde{w}_j^{(i)}\tilde{x^{(i)}}}{\sum_{i=1}^{n} w_j^{(i)} + \sum_{i=1}^{\tilde{n}}\tilde{w}_j^{(i)}}.\end{aligned}$$

$\mathbf{\Sigma}_j$: Finally, derive the update for Σ_j via S_j. Again, Do this by maximizing the expression with the argmax above with respect to S_j.

.

First, calculate the gradient with respect to S_j:

$$\begin{aligned}\nabla_{S_j} &= \frac{1}{2}\nabla_{S_j}\left(\sum_{i=1}^{n}\sum_{j=1}^{k} w_j^{(i)}(x^{(i)}-\mu_j)^T S_j(x^{(i)}-\mu_j) + \frac{1}{2}\log|S_j^{-1}| + \sum_{i=1}^{\tilde{n}}\sum_{j=1}^{k}\tilde{w}_j^{(i)}(\tilde{x^{(i)}}-\mu_j)^T S_j(\tilde{x^{(i)}}-\mu_j) + \frac{1}{2}\log|S_j^{-1}| + C\right)\\ &= \sum_{i=1}^{n} w_j^{(i)}(x^{(i)}-\mu_j)^T(x^{(i)}-\mu_j) - 1/S_j^{-1} + \sum_{i=1}^{\tilde{n}}\tilde{w}_j^{(i)}(\tilde{x^{(i)}}-\mu_j)^T(\tilde{x^{(i)}}-\mu_j) - 1/S_j^{-1}.\end{aligned}$$

Next, set the gradient to zero and solve for S_j:

$$\begin{aligned}0 &= \sum_{i=1}^{n} w_j^{(i)}(x^{(i)}-\mu_j)^T(x^{(i)}-\mu_j) - 1/S_j^{-1} + \sum_{i=1}^{\tilde{n}}\tilde{w}_j^{(i)}(\tilde{x^{(i)}}-\mu_j)^T(x^{(i)}-\mu_j) - 1/S_j^{-1}\\ \implies S_j &= \left(\frac{\sum_{i=1}^{n} w_j^{(i)}(x^{(i)}-\mu_j)^T(x^{(i)}-\mu_j) + \sum_{i=1}^{\tilde{n}}\tilde{w}_j^{(i)}(\tilde{x^{(i)}}-\mu_j)^T(\tilde{x^{(i)}}-\mu_j)}{\sum_{i=1}^{n} w_j^{(i)} + \sum_{i=1}^{\tilde{n}}\tilde{w}_j^{(i)}}\right)^{-1}\end{aligned}$$

This results in the final set of update expressions:

$$\begin{aligned}\phi_j &:= \frac{\sum_{i=1}^{n} w_j^{(i)} + \sum_{i=1}^{\tilde{n}}\tilde{w}_j^{(i)}}{n+\alpha\tilde{n}}\\ \mu_j &:= \frac{\sum_{i=1}^{n} w_j^{(i)}x^{(i)} + \sum_{i=1}^{n}\tilde{w}_j^{(i)}\tilde{x^{(i)}}}{\sum_{i=1}^{n} w_j^{(i)} + \sum_{i=1}^{\tilde{n}}\tilde{w}_j^{(i)}}\\ \Sigma_j &:= \frac{\sum_{i=1}^{n} w_j^{(i)}(x^{(i)}-\mu_j)^T(x^{(i)}-\mu_j) + \sum_{i=1}^{\tilde{n}}\tilde{w}_j^{(i)}(\tilde{x^{(i)}}-\mu_j)^T(\tilde{x^{(i)}}-\mu_j)}{\sum_{i=1}^{n} w_j^{(i)} + \sum_{i=1}^{\tilde{n}}\tilde{w}_j^{(i)}}\end{aligned}$$

2.f

(i) The semi-supervised model converges faster. This is likely because the labeled data has a greater affect on the gradients of the model parameters.

(ii) As seen from the output plots, the unsupervised model has greater variability when confronted with different random initializations, while the semi-supervised EM gave consistent results. This is likely because the observed data in the latter gave the algorithm benchmark examples to work with, allowing it to converge on a consistent solution. The unsupervised EM predictions vary widely as the data become more sparsely spaced.

(iii) Based on stability alone, the semi-supervised EM gives better assignment predictions. The unsupervised implementation varies widely once the variance of the data distribution is high. Random noise in the initialization step did not affect the semi-supervised results.

3.a

The key technical idea of ICA is to use a non-rotationally invariant distribution and solve the log-likelihood of the unmixing matrix by using gradient descent to compute on W. To show why we cannot do this when g is the standard normal CDF, we attempt to mathematically break out $\ell(W)$ so that we can maximize.

$$\begin{aligned}
\ell(W) &= \sum_{i=1}^{n} \log p(x^{(i)}) \\
&= \sum_{i=1}^{n} \log p(Wx^{(i)})|W|) \\
&= \sum_{i=1}^{n} \log \left(\frac{1}{(2\pi)^{d/2}} \exp(-\frac{1}{2}(Wx^{(i)})^T(Wx^{(i)})|W|) \right) \\
&= \sum_{i=1}^{n} \left(\log |W| - \frac{1}{2}(x^{(i)})^T W^T W x^{(i)} - \frac{d}{2}\log(2\pi) \right).
\end{aligned}$$

Taking the gradient of the expanded log-likelihood of the unmixing matrix and setting it equal to zero, we get

$$\begin{aligned}
\nabla_W \ell(W) &= \sum_{i=1}^{n} \left(\nabla_W \log |W| - \nabla_W \frac{1}{2}(x^{(i)})^T W^T W x^{(i)} \right) \\
0 &= \sum_{i=1}^{n} ((W^{-1})^T - W x^{(i)}(x^{(i)})^T) \\
0 &= n(W^{-1})^T - W \sum_{i=1}^{n} x^{(i)}(x^{(i)})^T \\
0 &= n(W^{-1})^T - W X^T X \\
W^T W &= (\frac{1}{n} X^T X)^{-1}.
\end{aligned}$$

If we assume that the right-hand side of the resultant equation above is invertible, then we've actually arrived at a tautology. This is because $W^T W = I$ because W can be any orthogonal matrix and still lead to Gaussian outputs. Hence, ICA does not provide a unique solution for W in terms of X due to the lack of higher-order features to exploit. This is the ambiguity that exists for Gaussian sources.

3.b

Using the form of the log-likelihood of the unmixing matrix given in the notes, we derive the update rule by substituting the given PDF $f(s)$ as g'. We then take the gradient to represent the missing term of the update rule.

$$\begin{aligned}\nabla_W \ell(W) &= \sum_{i=1}^{n} \left(\nabla_W \log |W| + \sum_{j=1}^{d} \nabla_W \log g'(w_j^T x^{(i)}) \right) \\ &= (W^{-1})^T + \sum_{j=1}^{d} \nabla_W \log(\frac{1}{2} \exp(-|w_j^T x^{(i)}|)) \\ &= (W^{-1})^T - \frac{W^T X X^T}{|W^T X X^T|}.\end{aligned}$$

This gives the update rule

$$W := W + \alpha \left((W^{-1})^T - \frac{W^T X X^T}{|W^T X X^T|} \right).$$

4.a

To prove that β achieves zero cost in the least-squares cost function under the given conditions, we first assume that $X \in \mathbb{R}^{n \times d}$ with $n < d$. We also assume that $XX^T \in \mathbb{R}^{n \times n}$ is an invertible matrix, so β achieves zero cost if and only if

$$\beta = X^T(XX^T)^{-1}y + \zeta \tag{1}$$

such that ζ is in the subspace orthogonal to the data. For the linear model

$$J(\beta) = \frac{1}{2n}||X\beta - y||_2^2 \tag{2}$$

we say that β is a minimizer of (2) if $J(\beta) = 0$. Therefore,

$$X\beta - y = 0, \text{or} \beta = X^T y.$$

We define ζ to satisfy the statement $\zeta^T x^{(i)} = 0, \forall 1 \leq i \leq n$. For β of the form given in (1), we use equation (2) to get

$$\begin{aligned} ||X\beta - y||_2^2 &= ||XX^T(XX^T)^{-1}y + X\zeta - y||_2^2 \\ &= ||X\zeta||_2^2. \end{aligned}$$

Since we have assumed that ζ is in the subspace orthogonal to X, we have that

$$||X\beta - y||_2^2 = 0$$

and β is necessarily a minimizer of (2) that achieves zero cost.

4.b

We assume the same setup as 4(a) with $X \in \mathbb{R}^{n \times d}$ where $n < d$ and $XX^T \in \mathbb{R}^{n \times n}$ is an invertible matrix. To reiterate, β is of the form

$$\beta = X^T(XX^T)^{-1}y + \zeta \tag{3}$$

for some ζ in the subspace orthogonal to all the data. For a minimum norm solution of the form

$$\rho = X^T(XX^T)^{-1}y$$

we aim to show that

$$||\beta||_2^2 = ||\rho||_2^2 + ||\zeta||_2^2.$$

Using the definition given by (3), we expand the norm of β and cancel the terms containing $X^T\zeta$ which go to zero:

$$\begin{aligned} ||\beta||_2^2 = \beta^T\beta &= (X^T(XX^T)^{-1}y + \zeta)^T(X^T(XX^T)^{-1}y + \zeta) \\ &= X^T(XX^T)^{-1}yX^T(XX^T)^{-1}y + \zeta^T\zeta \\ &= ||\rho||_2^2 + ||\zeta||_2^2. \end{aligned}$$

Hence, for any vector β such that $J(\beta) = 0$ and $||\rho||_2^2 + ||\zeta||_2^2$, we have

$$||\beta||_2^2 \geq ||\rho||_2^2$$

because $||\zeta||_2^2 \geq 0$. This means that ρ is the minimum of the family of global minimizers, β (i.e. the minimum norm solution).

4.d

We assume the same setup as 4(a) with $X \in \mathbb{R}^{n \times d}$ where $n < d$ and $XX^T \in \mathbb{R}^{n \times n}$ is an invertible matrix. By way of induction, let $\beta^{(0)} = 0$. Then according to our update rule for gradient descent,

$$\beta^{(1)} = -\frac{\eta}{n} X^T (X\beta^{(0)} - y) = -\frac{\eta}{n} X^T y.$$

This shows that $\beta^{(1)}$ is a linear combination of $\{x^{(1)}, x^{(2)}, ..., x^{(n)}\}$ for $t \geq 0$. Furthermore, take some vector $v^{(t)}$ for $t \geq 0$. We use the definition of y to show

$$\begin{aligned} \beta^{(t+1)} = \beta^{(t)} - \frac{\eta}{n} X^T (X\beta^{(t)} - y) &= \beta^{(t)} - \frac{\eta}{n} X^T (X\beta^{(t)} - X^T v^{(t)}) \\ &= \beta^{(t)} - \frac{\eta}{n} X^T (X(\beta^{(t)} - v^{(t)})). \end{aligned}$$

Since $\beta^{(1)}$ is a linear combination of $\{x^{(1)}, x^{(2)}, ..., x^{(n)}\}$, the result above indicates that $(\beta^{(t)} - v^{(t)})$ is also a linear combination of the data. It follows that $\beta^{(t+1)}$ is also a linear combination, and by way of induction, we are able to write $\beta^{(t)} = X^T v^{(t)}$ for any $t \geq 0$.

Now let $\hat{\beta}$ be a solution to the gradient descent update rule with zero initialization such that $J(\hat{\beta}) = 0$. Likewise, if $\hat{\beta} = X^T v^{(t)}$ for some $v^{(t)}$, then we have that $\hat{\beta}$ is also a linear combination of $\{x^{(1)}, x^{(2)}, ..., x^{(n)}\}$ by our previous conclusion. $J(\hat{\beta}) = 0$ tells us that

$$0 = \frac{1}{2n} ||X\hat{\beta} - y||_2^2$$

must be true by the definition of y. Therefore, $\hat{\beta} = X^T v^{(t)}$ for some $v^{(t)}$ and $J(\hat{\beta})$, so we have $\hat{\beta} = \rho$ the minimum norm solution.

4.e

We assume that $X \in \mathbb{R}^{n \times d}$ and $y \in \mathbb{R}^n$ for $n < d$ and invertible XX^T. We also suppose that our model can be written as

$$f_{\theta,\phi}(x) = x^T(\theta^{\odot 2} - \phi^{\odot 2}). \tag{4}$$

Lastly, we assume the labels and the cost function take the forms

$$y^{(i)} = (x^{(i)})^T((\theta^*)^{\odot 2} - (\phi^*)^{\odot 2}) \tag{5}$$

$$J(\theta, \phi) = \frac{1}{4n}\sum_{i=1}^{n}(f_{\theta,\phi}(x^{(i)}) - y^{(i)})^2. \tag{6}$$

Substituting (4) and (5) into (6), we have

$$J(\theta, \phi) = \frac{1}{4n}\sum_{i=1}^{n}((x^{(i)})^T(\theta^{\odot 2} - \phi^{\odot 2}) - (x^{(i)})^T((\theta^*)^{\odot 2} - (\phi^*)^{\odot 2}))^2.$$

It is immediately clear that $J(\theta^*, \phi^*) = 0$ is a zero cost solution, but we can go a step further by taking the definition of the labeled data, $y^{(i)} = (x^{(i)})^T\beta*$ for some $\beta^* \in \mathbb{R}^d$. From the equation above, this means that $J(\theta, \phi) = 0$ when $\beta^* = (\theta^{\odot 2} - \phi^{\odot 2})$. In 4(a), we proved that infinitely many such β^* that minimize the linear model. By the same logic mapping $J(\theta, \phi)$ to $J(\beta)$, there exist infinitely many optimal solutions that minimize the quadratic model for the parameters θ and ϕ.

4.g

- All models give low training error regardless of the initialization
- The best validation error is given by $\alpha = 0.01$.

4.i

In general, SGD does not find a better solution. SGD yields lower validation error for the smallest batch size (i.e. batch size = 1), but not when compared at greater batch sizes against lower values of α for classic gradient descent. This is likely because noise in the training process induces implicit regularization since noise decreases with increasing batch size. SGD works better at lower batch sizes because stochasticity in the optimization process helps the optimizer to find a more generalized solution.

Documentation: I did not collaborate with anyone on this assignment.