Solutions:

1. We can define ϕ as follows:

$$\phi(\boldsymbol{x}) = \begin{pmatrix} 1 \\ \sqrt{2}x_1 \\ \sqrt{2}x_2 \\ \vdots \\ \sqrt{2}x_d \\ \sqrt{2}x_1x_2 \\ \sqrt{2}x_1x_3 \\ \vdots \\ \sqrt{2}x_{d-1}x_d \\ x_1^2 \\ x_2^2 \\ \vdots \\ x_d^2 \end{pmatrix}.$$

Specifically, the first dimension is order 0 and has value 1, the next d dimensions are order 1 (with value $\sqrt{2}x_j$ for every dimension j of the original space), the next $d(d-1)/2$ dimensions are all second-order product pairs $\sqrt{2}x_ix_j$, and the last d dimensions are second order squared x_j^2 for each of the input space dimensions j. Thus, the total number of dimensions is $1 + d + d(d-1)/2 + d$, which is quadratic in the number of input space dimensions.

Now consider an inner product between two embedded vectors, $\phi(\boldsymbol{x})^T\phi(\boldsymbol{y})$. The first $d+1$ terms in the inner product equal

$$1 + \sum_j 2x_jy_j = 1 + 2\boldsymbol{x}^T\boldsymbol{y}.$$

The remaining terms in the inner product can be seen to be equal to

$$\begin{aligned} & \sum_{i,j,i\neq j} 2x_ix_jy_iy_j + x_1^2y_1^2 + ... + x_d^2y_d^2 \\ = & \sum_{i,j,i\neq j} 2x_iy_ix_jy_j + x_1^2y_1^2 + ... + x_d^2y_d^2 \\ = & (x_1y_1 + x_2y_2 + ... + x_dy_d)^2 \\ = & (\sum_j x_jy_j)^2 = (\boldsymbol{x}^T\boldsymbol{y})^2 \end{aligned}$$

The complete inner product is the sum of all these terms:

$$\phi(\boldsymbol{x})^T\phi(\boldsymbol{y}) = (\boldsymbol{x}^T\boldsymbol{y})^2 + 2\boldsymbol{x}^T\boldsymbol{y} + 1 = (\boldsymbol{x}^T\boldsymbol{y} + 1)^2.$$

2. In standard k-means, the mean $\boldsymbol{\mu}_j$ for cluster j (we will denote the set of points in cluster j as π_j) is

$$\boldsymbol{\mu}_j = \frac{1}{n_j} \sum_{\boldsymbol{x}_\ell \in \pi_j} \boldsymbol{x}_\ell.$$

If we map the data to another space via ϕ, the mean in the new space is given by

$$\boldsymbol{\mu}_j = \frac{1}{n_j} \sum_{\boldsymbol{x}_\ell \in \pi_j} \phi(\boldsymbol{x}_\ell).$$

Now the distance between a mapped point $\phi(\boldsymbol{x}_i)$ and the mean of a cluster in the new space is

$$\begin{aligned} \|\phi(\boldsymbol{x}_i) - \boldsymbol{\mu}_j\|_2^2 &= (\phi(\boldsymbol{x}_i) - \boldsymbol{\mu}_j)^T(\phi(\boldsymbol{x}_i) - \boldsymbol{\mu}_j) \\ &= \left(\phi(\boldsymbol{x}_i) - \frac{1}{n_j}\sum_{\boldsymbol{x}_\ell \in \pi_j} \phi(\boldsymbol{x}_\ell)\right)^T \left(\phi(\boldsymbol{x}_i) - \frac{1}{n_j}\sum_{\boldsymbol{x}_\ell \in \pi_j} \phi(\boldsymbol{x}_\ell)\right) \\ &= \phi(\boldsymbol{x}_i)^T\phi(\boldsymbol{x}_i) - \frac{2}{n_j}\sum_{\boldsymbol{x}_\ell \in \pi_j} \phi(\boldsymbol{x}_i)^T\phi(\boldsymbol{x}_\ell) + \frac{1}{n_j^2}\sum_{\boldsymbol{x}_\ell \in \pi_j}\sum_{\boldsymbol{x}_m \in \pi_j} \phi(\boldsymbol{x}_\ell)^T\phi(\boldsymbol{x}_m). \end{aligned}$$

Now the distance computation is expressed purely in terms of inner products between mapped vectors. Thus, if $\phi(\boldsymbol{x}_i)^T\phi(\boldsymbol{x}_\ell) = (\boldsymbol{x}_i^T\boldsymbol{x}_\ell + 1)^2$, then this distance can be computed via

$$(\boldsymbol{x}_i^T\boldsymbol{x}_i + 1)^2 - \frac{2}{n_j}\sum_{\boldsymbol{x}_\ell \in \pi_j} (\boldsymbol{x}_i^T\boldsymbol{x}_\ell + 1)^2 + \frac{1}{n_j^2}\sum_{\boldsymbol{x}_\ell \in \pi_j}\sum_{\boldsymbol{x}_m \in \pi_j} (\boldsymbol{x}_\ell^T\boldsymbol{x}_m + 1)^2.$$

3. When writing the k-means algorithm using only mapped inner products, we must keep a few things in mind. Since we never want to explicitly compute the mean of the mapped vectors, the initialization must be given as a partition (in k-means the initialization could be either an initialization of the partition of an initialization of the means). More generally, we never explicitly do the mean update step as in standard k-means, as that would involve working in the mapped space directly. Instead, we observe that the distance computation as given above uses the means implicitly.

   Therefore, the algorithm works as follows:

   (a) Initialize k clusters $\pi_1^{(0)}, ..., \pi_k^{(0)}$.

   (b) Set $t = 0$.

   (c) For each point $\boldsymbol{x}$, find its new cluster index as

   $$j^*(\boldsymbol{x}) = \mathrm{argmin}_j \|\phi(\boldsymbol{x}) - \boldsymbol{\mu}_j\|_2^2,$$

   using the expression given in the previous exercise.

(d) Compute the updated clusters as

$$\pi_j^{(t+1)} = \{\boldsymbol{x} | j^*(\boldsymbol{x}) = j\}.$$

(e) If not converged, set $t = t + 1$ and go to (c).

4. As discussed in lecture, the cluster decision boundary between two clusters in the original space will be given by the set of all points such that the squared Euclidean distance to one cluster is the same as the squared Euclidean distance to the other cluster. That is, $\|\boldsymbol{x} - \boldsymbol{\mu}_1\|_2^2 = \|\boldsymbol{x} - \boldsymbol{\mu}_2\|_2^2$. By expanding this out, we showed in class that this is given by

$$2(\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1)^T \boldsymbol{x} + (\boldsymbol{\mu}_1^T \boldsymbol{\mu}_1 - \boldsymbol{\mu}_2^T \boldsymbol{\mu}_2) = 0,$$

which is of the form $\boldsymbol{w}^T \boldsymbol{x} + b = 0$, i.e., a linear surface. Now, if instead we consider working in the mapped space, the cluster decision boundary will be given by the set of $\phi(\boldsymbol{x})$ such that $\|\phi(\boldsymbol{x}) - \boldsymbol{\mu}_1\|_2^2 = \|\phi(\boldsymbol{x}) - \boldsymbol{\mu}_2\|_2^2$ which, by a similar argument as in the original case, will be of the form $\boldsymbol{w}^T \phi(\boldsymbol{x}) + b = 0$. Based on the embedding computed for $\phi(\boldsymbol{x})$ earlier, this is exactly a quadratic function in the input space—it is a weighted combination of order 0, 1, and 2 terms of the original feature vector $\boldsymbol{x}$.