Problem. 1.

(A)

Proof.

$$K_{ij} = k(x_i, x_j) = \phi(x_i) \cdot \phi(x_j)$$

$$\begin{aligned} c^T K c &= \sum_i \sum_j c_i c_j K_{ij} \\ &= \sum_i \sum_j c_i c_j \phi(x_i) \cdot \phi(x_j) \\ &= (\sum_i c_i \phi(x_i)) \cdot (\sum_i c_i \phi(x_i)) \\ &= \|\sum_i c_i \phi(x_i)\|_2^2 \\ &\geq 0 \end{aligned}$$

□

a)

Proof.

$$\begin{aligned} k(x, y) &= \alpha k_1(x, y) + \beta k_2(x, y) \\ &= \langle \sqrt{\alpha}\phi_1(x), \sqrt{\alpha}\phi_1(y) \rangle + \langle \sqrt{\beta}\phi_2(x), \sqrt{\beta}\phi_2(y) \rangle \\ &= \langle [\sqrt{\alpha}\phi_1(x), \sqrt{\beta}\phi_2(x)], [\sqrt{\alpha}\phi_1(y), \sqrt{\beta}\phi_2(y)] \rangle \end{aligned}$$

□

b)

Proof. Let $f_i(x)$ be the ith feature value under the feature map ϕ_1, $g_i(x)$ be the ith feature value under the feature map ϕ_2.

Then:

$$\begin{aligned}
k(x,y) &= k_1(x,y)k_2(x,y) \\
&= (\phi_1(x)\cdot\phi_1(y))(\phi_2(x)\cdot\phi_2(y)) \\
&= (\sum_{i=0}^{\infty} f_i(x)f_i(y))(\sum_{j=0}^{\infty} g_j(x)g_j(y)) \\
&= \sum_{i,j} f_i(x)f_i(y)g_j(x)g_j(y) \\
&= \sum_{i,j} (f_i(x)g_j(x))(f_i(y)g_j(y)) \\
&= \langle \phi_3(x), \phi_3(y)\rangle
\end{aligned}$$

where ϕ_3 has feature $h_{i,j}(x) = f_i(x)g_j(x)$. □

c)
Since each polynomial term is a product of kernels with a positive coefficient, the proof follows from part a and b.

d)
By Taylor expansion:

$$e^x = \sum_{i=0}^{\infty} \frac{x^i}{i!}$$

Then the proof follows part c.

(B)

Proof. We wish to show that the kernel $k(\mathbf{x},\mathbf{y}) = \exp(-\frac{1}{2}\|\mathbf{x}-\mathbf{y}\|^2)$ can be written as an inner-product between some mapping ϕ on $\mathbf{x}$ and $\mathbf{y}$, in other words, $k(\mathbf{x},\mathbf{y}) = \langle\phi(\mathbf{x}),\phi(\mathbf{y})\rangle$. Assume that $\mathbf{x},\mathbf{y}\in\mathbb{R}^d$. Consider the formula for $\phi_{\mathbf{z}}(\mathbf{x}) = (\pi/2)^{-d/4}\exp(-\|\mathbf{x}-\mathbf{z}\|^2)$ which is an infinite dimensional function over $\mathbf{z}\in\mathbb{R}^d$ (rather than a finite dimensional vector with $\mathbf{z}$ being a discrete index as we did in class). Similarly, we have $\phi_{\mathbf{z}}(\mathbf{y}) = (\pi/2)^{-d/4}\exp(-\|\mathbf{y}-\mathbf{z}\|^2)$. Then, we define the kernel as $k(\mathbf{x},\mathbf{y}) = \langle\phi_{\mathbf{z}}(\mathbf{x}),\phi_{\mathbf{z}}(\mathbf{y})\rangle = \int_{\mathbf{z}}\phi_{\mathbf{z}}(\mathbf{x})\times\phi_{\mathbf{z}}(\mathbf{y})\mathbf{dz}$. This integral evaluates to

$$\begin{aligned}
k(\mathbf{x},\mathbf{y}) &= \int_{\mathbf{z}} (\pi/2)^{-d/4}\exp\left(-\|\mathbf{x}-\mathbf{z}\|^2\right)\times(\pi/2)^{-d/4}\exp\left(-\|\mathbf{y}-\mathbf{z}\|^2\right)\mathbf{dz} \\
&= (\pi/2)^{-d/2}\int_{\mathbf{z}}\exp\left(-\mathbf{x}^\top\mathbf{x}-\mathbf{z}^\top\mathbf{z}+2\mathbf{x}^\top\mathbf{z}\right)\exp\left(-\mathbf{y}^\top\mathbf{y}-\mathbf{z}^\top\mathbf{z}+2\mathbf{y}^\top\mathbf{z}\right)\mathbf{dz} \\
&= (\pi/2)^{-d/2}\exp\left(-\mathbf{x}^\top\mathbf{x}-\mathbf{y}^\top\mathbf{y}\right)\int_{\mathbf{z}}\exp\left(-2\mathbf{z}^\top\mathbf{z}+2\left(\mathbf{y}+\mathbf{x}\right)^\top\mathbf{z}\right)\mathbf{dz}
\end{aligned}$$

Define $\mathbf{r} = (\mathbf{y} + \mathbf{x})/2$ for short-hand and write...

$$
\begin{aligned}
k(\mathbf{x}, \mathbf{y}) &= (\pi/2)^{-d/2} \exp\left(-\mathbf{x}^\top \mathbf{x} - \mathbf{y}^\top \mathbf{y}\right) \int_{\mathbf{z}} \exp\left(-2\mathbf{z}^\top \mathbf{z} + 4\mathbf{r}^\top \mathbf{z}\right) \mathbf{dz} \\
&= (\pi/2)^{-d/2} \exp\left(-\mathbf{x}^\top \mathbf{x} - \mathbf{y}^\top \mathbf{y}\right) \exp\left(2\mathbf{r}^\top \mathbf{r}\right) \int_{\mathbf{z}} \exp\left(-2\mathbf{z}^\top \mathbf{z} + 4\mathbf{r}^\top \mathbf{z} - 2\mathbf{r}^\top \mathbf{r}\right) \mathbf{dz} \\
&= (\pi/2)^{-d/2} \exp\left(-\mathbf{x}^\top \mathbf{x} - \mathbf{y}^\top \mathbf{y}\right) \exp\left(2\mathbf{r}^\top \mathbf{r}\right) \int_{\mathbf{z}} \exp\left(-2\|\mathbf{z} - \mathbf{r}\|^2\right) \mathbf{dz} \\
&= (\pi/2)^{-d/2} \exp\left(-\mathbf{x}^\top \mathbf{x} - \mathbf{y}^\top \mathbf{y}\right) \exp\left(2\mathbf{r}^\top \mathbf{r}\right) (\pi/2)^{d/2} \\
&= \exp\left(-\mathbf{x}^\top \mathbf{x} - \mathbf{y}^\top \mathbf{y}\right) \exp\left(\frac{1}{2}\mathbf{x}^\top \mathbf{x} + \frac{1}{2}\mathbf{y}^\top \mathbf{y} + \mathbf{x}^\top \mathbf{y}\right) \\
&= \exp\left(-\frac{1}{2}\mathbf{x}^\top \mathbf{x} - \frac{1}{2}\mathbf{y}^\top \mathbf{y} + \mathbf{x}^\top \mathbf{y}\right) \\
&= \exp\left(-\frac{1}{2}\|\mathbf{x} - \mathbf{y}\|^2\right)
\end{aligned}
$$

$\square$

Problem. 2.

First, we normalize the data X, and then randomly split the dataset in to two half for cross validation. Then, we train SVM using polynomial kernel and RBF kernel with different parameters and costs. Testing errors are plotted in the following two graphs.

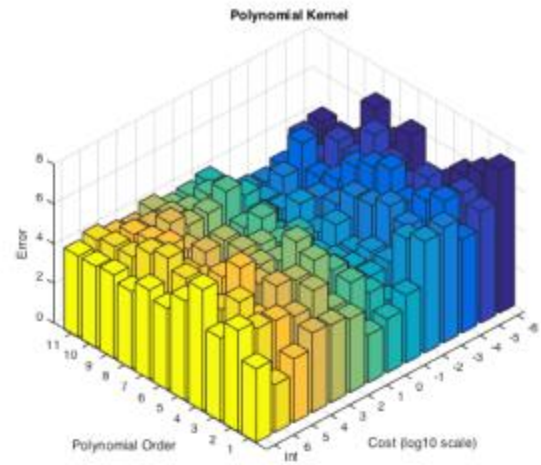

Figure 1: Testing errors using polynomial kernel with different orders and costs

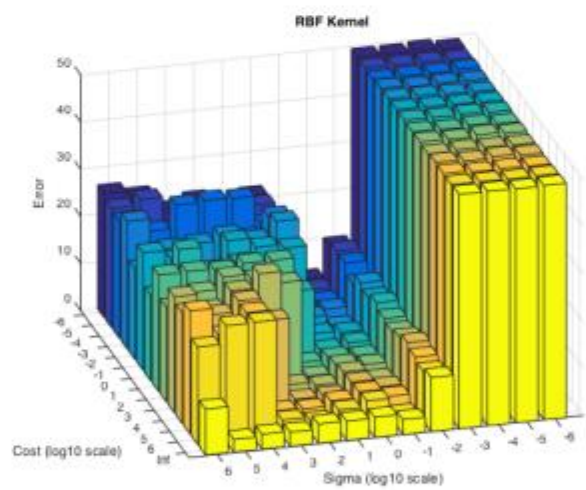

Figure 2: Testing errors using RBF kernel with different variances and costs

```
function question4

% load the dataset
load hw2-2015-dataset;

n = size(X,1);
idxtrain = randsample(n, n/2);

% split the dataset
idxtest  = setdiff(1:n, idxtrain);

trnX = X(idxtrain, :);
tstX  = X(idxtest,  :);

trnY = Y(idxtrain, :);
tstY  = Y(idxtest,  :);

% classify using polynomials
d_max = 50;
err_d = zeros(1, d_max);

global p1;
for d=1:d_max
    p1 = d;

    % train
    [nsv, alpha, bias] = svc(trnX,trnY,'poly',inf);

    % predict
    predictedY = svcoutput(trnX,trnY,tstX, 'poly',alpha,bias);

    % compute test error
    err_d(d)      = svcerror(trnX,trnY,tstX,tstY,'poly',alpha,bias);
end

% plot error vs polynomial degree
f = figure(1);
clf(f);
plot(1:d_max, err_d);
print(f, '-depsc', 'poly.eps');
```

```
% classify using rbfs
sigmas = .1:.1:2;
err_sigma = zeros(1, numel(sigmas));
for sigma_i=1:numel(sigmas)
    p1 = sigmas(sigma_i);

    % train
    [nsv, alpha, bias] = svc(trnX,trnY,'rbf',inf);

    % predict
    predictedY = svcoutput(trnX,trnY,tstX, 'rbf',alpha,bias);

    % compute test error
    err_sigma(sigma_i) = svcerror(trnX,trnY,tstX,tstY,'rbf',alpha,bias);
end

% plot error vs polynomial degree
f = figure(1);
clf(f);
plot(sigmas, err_sigma);
print(f, '-depsc', 'rbf.eps');
```

Problem. 3.

We first obtain the likelihood by **multiplying** the probability density function for each X_i. We then **simplify** this expression.

$$L(\alpha) = \prod_{i=1}^{n} f(x_i;\alpha) = \prod_{i=1}^{n} \alpha^{-2} x_i e^{-x_i/\alpha} = \alpha^{-2n}\left(\prod_{i=1}^{n} x_i\right)\exp\left(\frac{-\sum_{i=i}^{n} x_i}{\alpha}\right)$$

Instead of directly maximizing the likelihood, we instead maximize the **log**-likelihood.

$$\log L(\alpha) = -2n\log\alpha + \sum_{i=i}^{n}\log x_i - \frac{\sum_{i=i}^{n} x_i}{\alpha}$$

To maximize this function, we take a **derivative** with respect to α.

$$\frac{d}{d\alpha}\log L(\alpha) = \frac{-2n}{\alpha} + \frac{\sum_{i=i}^{n} x_i}{\alpha^2}$$

We set this derivative equal to **zero**, then **solve** for α.

$$\frac{-2n}{\alpha} + \frac{\sum_{i=i}^{n} x_i}{\alpha^2} = 0$$

Solving gives our *estimator*, which we denote with a **hat**.

$$\boxed{\hat{\alpha} = \frac{\sum_{i=i}^{n} x_i}{2n} = \frac{\bar{x}}{2}}$$

Using the given data, we obtain an *estimate*.

$$\hat{\alpha} = \frac{0.25 + 0.75 + 1.50 + 2.50 + 2.0}{2 \cdot 5} = \boxed{0.70}$$

(We should also verify that this point is a maxmimum, which is omitted here.)