Rio Sonoyama
CS146, Homework4

1.

(a) First, we expand $J(\tilde{W})$:

$$\begin{aligned}
J(\tilde{W}) &= \frac{1}{2}\operatorname{tr}\left((\tilde{X}\tilde{W} - T)^T(\tilde{X}\tilde{W} - T)\right) \\
&= \frac{1}{2}\operatorname{tr}\left((\tilde{W}^T\tilde{X}^T - T^T)(\tilde{X}\tilde{W} - T)\right) \\
&= \frac{1}{2}\operatorname{tr}\left(\tilde{W}^T\tilde{X}^T\tilde{X}\tilde{W} - \tilde{W}^T\tilde{X}^T T - T^T\tilde{X}\tilde{W} + T^T T\right) \\
&= \frac{1}{2}\operatorname{tr}\left(\tilde{W}^T\tilde{X}^T\tilde{X}\tilde{W} - 2T^T\tilde{X}\tilde{W} + T^T T\right) \\
&= \frac{1}{2}\operatorname{tr}(\tilde{W}^T\tilde{X}^T\tilde{X}\tilde{W}) - \operatorname{tr}(2T^T\tilde{X}\tilde{W}) + \operatorname{tr}(T^T T) \qquad \text{because tr is a linear operator}
\end{aligned}$$

To find the closed form solution, we take derivative of $J(\tilde{W})$ with respect to $\tilde{W}$ and set it equal to 0.

$$\begin{aligned}
&\frac{\partial}{\partial\tilde{W}}\operatorname{tr}(T^T\tilde{X}\tilde{W}) = (T^T\tilde{X})^T = \tilde{X}^T T \\
&\frac{\partial}{\partial\tilde{W}}\operatorname{tr}(\tilde{W}^T\tilde{X}^T\tilde{X}\tilde{W}) = ((\tilde{X}^T\tilde{X})^T + \tilde{X}^T\tilde{X})\tilde{W} = 2\tilde{X}^T\tilde{X}\tilde{W} \\
&\frac{\partial}{\partial\tilde{W}}\operatorname{tr}(T^T T) = 0
\end{aligned}$$

So using above 3 equations,

$$\frac{\partial J(\tilde{W})}{\partial\tilde{W}} = 2\tilde{X}^T\tilde{X}\tilde{W} - 2\tilde{X}^T T = 0 \implies \tilde{X}^T\tilde{X}\tilde{W} = \tilde{X}^T T.$$

By multiplying $(\tilde{X}^T\tilde{X})^{-1}$ for the left, we have

$$\tilde{W} = (\tilde{X}^T\tilde{X})^{-1}\tilde{X}^T T.$$

(b) To show $J(\tilde{W})$ has a unique minimum, we are going to show that the second derivative of $J(\tilde{W})$ with respect to $\tilde{W}$ is positive semi-definite.

$$\frac{\partial^2 J(\tilde{W})}{\partial\tilde{W}^2} = \frac{\partial}{\partial\tilde{W}}(2\tilde{X}^T\tilde{X}\tilde{W} - 2\tilde{X}^T T) = 2(\tilde{X}^T\tilde{X})^T.$$

But since $\tilde{X}$ is a metric whose nth row is $\tilde{x}_n^T$,

$$\tilde{X}^T\tilde{X} = \begin{pmatrix}\tilde{x}_1 & \tilde{x}_2^T & \cdots & \tilde{x}_n\end{pmatrix}\begin{pmatrix}\tilde{x}_1 \\ \tilde{x}_2 \\ \vdots \\ \tilde{x}_n\end{pmatrix} = \sum_{i=1}^{n}\tilde{x}_i^T\tilde{x}_i.$$

Thus for any $z \in \mathbb{R}^D$,

$$z^T\tilde{X}^T\tilde{X}z = z^T\left(\sum_{i=1}^{n}\tilde{x}_i^T\tilde{x}_i\right)z = \sum_{i=1}^{n} z^T\tilde{x}_i^T\tilde{x}_i z = \sum_{i=1}^{n}(\tilde{x}_i z)^T(\tilde{x}_i z).$$

But for any i, $(\tilde{x}_i z)^T(\tilde{x}_i z) \geq 0$ (by an axiom of inner product). Thus,

$$z^T \tilde{X}^T \tilde{X} z = \sum_{i=1}^{n} (\tilde{x}_i z)^T(\tilde{x}_i z) \geq 0.$$

Hence, $\tilde{X}^T \tilde{X}$ is positive semi-definite, therefore, $J(\tilde{W})$ has a unique minimum.

2. Notice that $K(x_1, x_2) = \langle \phi(x_1), \phi(x_2) \rangle$ for some $\phi(\,.\,)$. But since $\phi(\,.\,)$ is a d-dimensional vector ($d \in \mathbb{N}$), we have

$$\langle \phi(x_1), \phi(x_2) \rangle^2 \leq \langle \phi(x_1), \phi(x_1) \rangle \cdot \langle \phi(x_2), \phi(x_2) \rangle.$$

by Cauchy-Schwartz inequality. Thus,

$$K(x_1, x_2)^2 \leq K(x_1, x_1) K(x_2, x_2).$$

3.

(a) Since K_1 and K_2 are symmetric and sum of symmetric matrices are also symmetric, K is symmetric. Also, because K_1 and K_2 are positive semi-definite, for any z, $z^T K_1 z \geq 0$ and $z^T K_2 z \geq 0$. Thus, for any z,

$$z^T K z = z^T (K_1 + K_2) z = z^T K_1 z + z^T K_2 z \geq 0.$$

Hence, K is positive semi-definite.
Because K is symmetric and positive semi-definite, it is a valid kernel.

(b) In general, for any symmetric matrices A and B of the same size, matrix multiplication is commutative (i.e. $AB = BA$). Also, since K_1 and K_2 are symmetric, $K_1^T = K_1$ and $K_2^T = K_2$. Using these two facts, we have

$$K^T = (K_1 K_2)^T = K_2^T K_1^T = K_2 K_1 = K_1 K_2 = K \implies K^T = K.$$

Hence, K is symmetric.
Now for any z,

$$\begin{aligned}
z^T K z &= z^T K_1 K_2 z \\
&= \sum_{i,k} z_i z_k K_1(i,k) K_2(i,k) \\
&= \sum_{i,k} z_i z_k \langle \phi^{(1)}(i), \phi^{(1)}(k) \rangle \langle \phi^{(2)}(i), \phi^{(2)}(k) \rangle \\
&= \sum_{i,k} z_i z_k \left(\sum_l \phi_l^{(1)}(i) \phi_l^{(1)}(k) \right) \left(\sum_m \phi_m^{(2)}(i) \phi_m^{(2)}(k) \right) \\
&= \sum_{l,m} \left(\sum_i z_i \phi_l^{(1)}(i) \phi_m^{(2)}(i) \right) \left(\sum_k z_j \phi_l^{(1)}(k) \phi_m^{(2)}(k) \right) \\
&= \sum_{l,m} \left(\sum_i z_i \phi_l^{(1)}(i) \phi_m^{(2)}(i) \right)^2 \geq 0
\end{aligned}$$

Thus, $z^T K z \geq 0$, hence K is positive semi-definite.
Because K is symmetric and positive semi-definite, it is a valid kernel.

(c) Expand K as follows:

$$K = \exp(K_1(i,j)) = \sum_{k}^{\infty} \frac{K_1(i,j)^k}{k!}.$$

Notice that if we expand the summation, each term has a product of K_1s. Since multiplying a kernel by positive scalar ($k! > 0$ for all k) does not affect the conditions to be kernel (i.e. scalar multiple of symmetric matrix is still symmetric and positive scalar multiple of a positive semi-definite matrix is still positive semi-definite), each term in the summation is a valid kernel by (b). Now since sum of kernels are still kernel (from part (a)), we can conclude that K is a valid kernel.

4. To find b_i, we can use support vectors. For a support vector i, we have

$$y^{(i)}(w^T x^{(i)} + b_i) = 1 \implies y^{(i)} = w^T x^{(i)} b_i.$$

Also, from $\dfrac{\partial L}{\partial w} = 0$, we have

$$w = \sum_i \alpha_i y^{(i)} x^{(i)}.$$

By combining these two equalities, for kth data point, we have

$$b_k = y^{(k)} - w^T x^{(k)} = y^{(k)} - \left(\sum_i \alpha_i y^{(i)} x^{(i)} \right) x^{(k)}.$$

But for non support vectors j, $\alpha_j = 0$. Thus, if we let $\mathcal{S}$ be the set of indexes of data points having $\alpha_j \neq 0$,

$$\sum_i \alpha_i y^{(i)} x^{(i)} = \sum_{i \in \mathcal{S}} \alpha_i y^{(i)} x^{(i)}.$$

Thus,

$$b_k = y^{(k)} - \left(\sum_{i \in \mathcal{S}} \alpha_i y^{(i)} x^{(i)} \right) x^{(k)} = y^{(k)} - \sum_{i \in \mathcal{S}} \alpha_i y^{(i)} \langle x^{(i)}, x^{(k)} \rangle.$$

Now, for any data point k that satisfies $0 < \alpha_k < C$ (i.e. any support vector k), we can find b_k. So, we take the average of such bs to find the final b. Thus, if we let $\mathcal{M}$ be the set of indexes of data points having $0 < \alpha_k < C$,

$$b = \frac{1}{N_{\mathcal{M}}} \sum_{k \in \mathcal{M}} b_k = \frac{1}{N_{\mathcal{M}}} \sum_{k \in \mathcal{M}} \left(y^{(k)} - \sum_{i \in \mathcal{S}} \alpha_i y^{(i)} \langle x^{(i)}, x^{(k)} \rangle \right),$$

which is what we wanted to show.

5.

(a) Let $w = (-0.1, -1)$.

Then, classification on the given data set yields:

$$
\begin{aligned}
f(x_1) &= \text{sign}(w_1 x_1 + w_0) = \text{sign}((-1)\cdot(-1) + (-0.1)) = \text{sign}(0.9) = +\\
f(x_2) &= \text{sign}(w_1 x_2 + w_0) = \text{sign}((-1)\cdot(0) + (-0.1)) = \text{sign}(-0.1) = -\\
f(x_3) &= \text{sign}(w_1 x_3 + w_0) = \text{sign}((-1)\cdot(1) + (-0.1)) = \text{sign}(-1.1) = -\\
f(x_4) &= \text{sign}(w_1 x_4 + w_0) = \text{sign}((-1)\cdot(-3) + (-0.1)) = \text{sign}(2.9) = +\\
f(x_5) &= \text{sign}(w_1 x_5 + w_0) = \text{sign}((-1)\cdot(-2) + (-0.1)) = \text{sign}(1.9) = +\\
f(x_6) &= \text{sign}(w_1 x_6 + w_0) = \text{sign}((-1)\cdot(3) + (-0.1)) = \text{sign}(-3.3) = -
\end{aligned}
$$

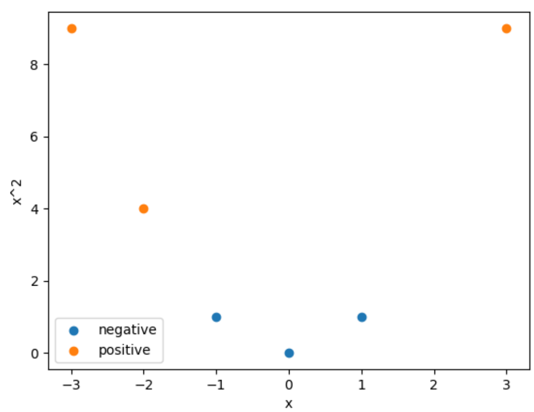
So, 4 points are correctly classified and 2 points are misclassified. Thus, accuracy is $\frac{2}{3}$.

(b) $K(x, z) = xz(1 + xz) = xz + x^2z^2$. Thus, $\phi(x) = (x, x^2)$.

(c) By applying $\phi(\,.\,)$, we have

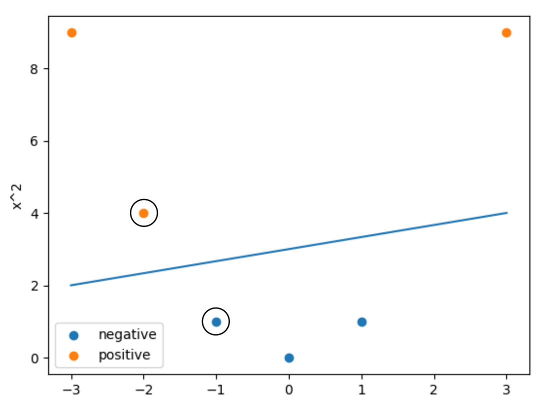
$$
\begin{aligned}
\phi(x_1) &= (-1,1)\\
\phi(x_2) &= (0,0)\\
\phi(x_3) &= (1,1)\\
\phi(x_4) &= (-3,9)\\
\phi(x_5) &= (-2,4)\\
\phi(x_6) &= (3,9)
\end{aligned}
$$

The plot of the points are shown below:

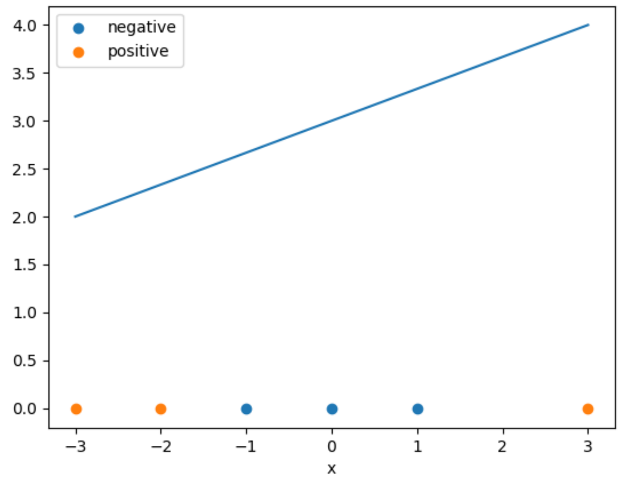


Clearly, the points are linearly separable in the induced feature space

(d) $w = (-9, -1, 3)$ gives the hyperplane shown below (Note: x and y axises are differently scaled so the boundary might not look orthogonal to the line that passes through support vectors):



(e)



(f) We use support vectors to find w and b. From part (d), we know x_1 and x_5 are support vectors. From the constraint $\sum_i \alpha_i y_i = 0$, we have $\alpha_1 - \alpha_5 = 0 \implies \alpha_1 = \alpha_5$.

So,

$$
\begin{aligned}
L(\alpha) &= \alpha_1 + \alpha_5 - \frac{1}{2}\left(\alpha_1^2 K(1,1) + \alpha_5^2 K(5,5) - 2\alpha_1\alpha_5 K(1,5)\right) \\
&= 2\alpha_1 - \frac{\alpha_1^2}{2}(2 + 20 - 12) \\
&= 2\alpha_1 - 5\alpha_1^2
\end{aligned}
$$

To find α_1, take derivative and set it equal to 0:

$$\frac{dL}{d\alpha_1} = 2 - 10\alpha_1 = 0 \implies \alpha_1 = \frac{1}{5}.$$

Now using this α_1,

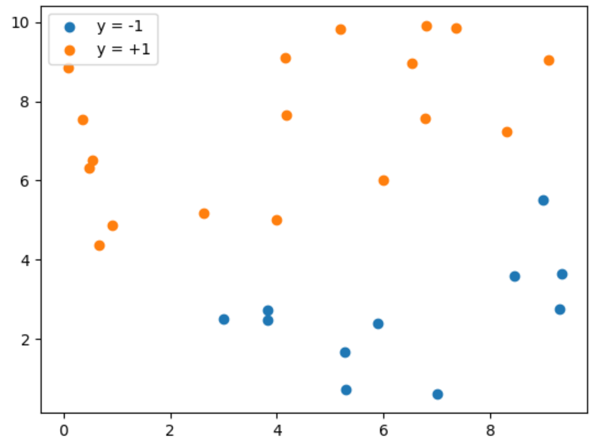
$$w = \alpha_1 y_1 \phi(x_1) + \alpha_5 y_5 \phi(x_5) = \left(-\frac{1}{5}, \frac{3}{5}\right)$$

$$b = y_5 - w^T \phi(x_5) = -\frac{9}{5}.$$

The values we obtained here are $\alpha = \frac{1}{5}$ multiple of what we had in part (d).

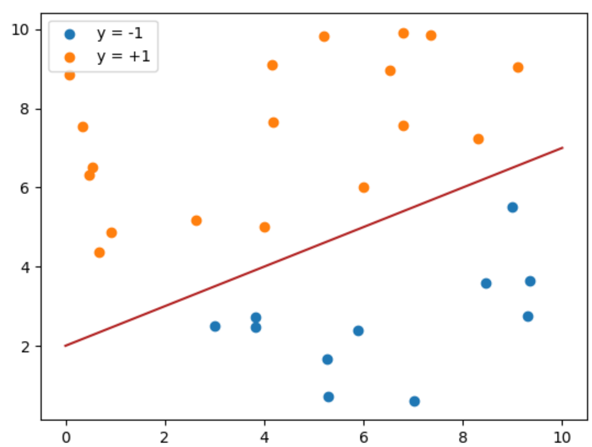
6.

(a) The plot is shown below:



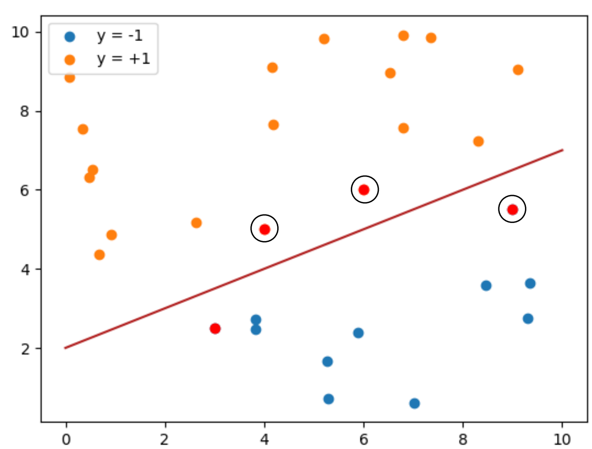
Clearly, the data is linearly separable.

(b) The plot and value of w and b are shown below:




```
w = [-0.5  1. ]
b = -2.0000000001022613
```

(c) There are 4 support vectors. The plot and nonzero α_is are shown below:




```
alpha[25] = 0.3825457131270906
alpha[26] = 0.24245428687288206
alpha[27] = 0.4601371393813008
alpha[28] = 0.16486286061867184
```

Python script for Q6:

```python
import numpy as np
import cvxpy as cp
import matplotlib.pyplot as plt

# Read data
data = np.genfromtxt('Data.csv', delimiter=',')
y_positive = data[data[:, 2] > 0]
y_negative = data[data[:, 2] < 0]

y = data[:, 2]
x = data[:, :2]


# Prime problem
w = cp.Variable(data.shape[1] - 1)
b = cp.Variable()
objective = cp.Minimize(0.5 * cp.norm(w, 2))
constraints = [y[i] * (w.T * x[i] + b) >= 1 for i in range(y.shape[0])]
prob = cp.Problem(objective, constraints)
prob.solve()
w_p = w.value
b_p = b.value


# Dual problem
temp = []
for i in range (y.shape[0]):
    temp.append(y[i] * x[i])
P = np.dot(np.array(temp),np.array(temp).T) + 1e-13 * np.eye(y.shape[0])

a = cp.Variable(y.shape[0])
objective = cp.Maximize(cp.sum(a) - 0.5 * cp.quad_form(a, P))
constraints = [i >= 0 for i in a] + [a.T * y == 0]
prob = cp.Problem(objective, constraints)
prob.solve()
a_d = a.value
idx = np.argwhere(a_d >= 1e-9)

# Plot
plt.scatter(y_negative[:, 0], y_negative[:, 1], label='y = -1')
plt.scatter(y_positive[:, 0], y_positive[:, 1], label='y = +1')

for i in range(y.shape[0]):
    if [i] in idx:
        plt.scatter(x[i][0], x[i][1], color='red')
n = np.linspace(0, 10)
plt.plot(n,(-w_p[0] * n - b_p) / w_p[1], c='firebrick')

plt.legend()
#plt.show()
#plt.savefig('6_c')
exit()
```