Problem Solutions

e-Chapter 8

Pierre Paquay

Problem 8.1

The two separation constraints are

$$(w^T x_+ + b) \geq 1 \text{ and } -(w^T x_- + b) \geq 1;$$

by adding these two constraints, we get that

$$w^T(x_+ - x_-) \geq 2.$$

Then, the Cauchy-Schwarz inequality gives us the following inequalities

$$2 \leq w^T(x_+ - x_-) \leq |w^T(x_+ - x_-)| \leq ||w||||x_+ - x_-||;$$

consequently, we get that

$$||w|| \geq \frac{2}{||x_+ - x_-||}.$$

Since we seek to minimize $||w||$, we choose w^* such that

$$||w^*|| = \frac{2}{||x_+ - x_-||}.$$

In this case, as we want w^* to satisfy both constraints, we may note that

$$2 \leq w^{*T}(x_+ - x_-) \leq |w^{*T}(x_+ - x_-)| \leq ||w^*||||x_+ - x_-|| = 2.$$

This means that

$$|w^{*T}(x_+ - x_-)| = ||w^*||||x_+ - x_-||,$$

which can only happen when $w^* = k(x_+ - x_-)$. Since, we have already established that

$$||w^*|| = \frac{2}{||x_+ - x_-||},$$

we choose k to be

$$k = \frac{2}{||x_+ - x_-||^2}.$$

Now, we may write that

$$w^* = \frac{2(x_+ - x_-)}{||x_+ - x_-||^2}.$$

It remains to determine the value of b^*. To do that we fix the following equality

$$2\left(\frac{(x_+ - x_-)}{||x_+ - x_-||^2}\right)^T x_+ + b^* = 1;$$

which gives us that

$$\begin{aligned} b^* &= 1 - 2\frac{x_+^T x_+ - x_-^T x_+}{||x_+ - x_-||^2} \\ &= \frac{x_-^T x_- - x_+^T x_+}{||x_+ - x_-||^2} \\ &= \frac{||x_-||^2 - ||x_+||^2}{||x_+ - x_-||^2}. \end{aligned}$$

It is now easy to verify that (w^*, b^*) satisfies both constraints and minimizes $||w||$, and therefore gives us the optimal hyperplane.

Problem 8.2

In this case, the constraints are

$$-b \geq 1,\ -(-w_2 + b) \geq 1,\ (-2w_1 + b) \geq 1.$$

If we combine the first and the third ones, we get $w_1 \leq -1$. The quantity we seek to minimize is

$$\frac{1}{2}w^T w = \frac{1}{2}(w_1^2 + w_2^2) \geq \frac{1}{2}(1 + 0) \geq \frac{1}{2},$$

where we have equality when $w_1 = -1$ and $w_2 = 0$; consequently, we choose $w^* = (-1, 0)$. With this in mind, the third constraint becomes

$$1 \leq -2w_1^* + b = 2 + b \Leftrightarrow b \geq -1;$$

so we choose $b^* = -1$. It is now easy to verify that (w^*, b^*) satisfies both constraints and minimizes $||w||$, and therefore gives us the optimal hyperplane. The margin in this case is given by $1/||w^*|| = 1$.

Problem 8.3

(a) We begin by computing the Lagrangian, we get

$$\begin{aligned} \mathcal{L}(\alpha) &= \frac{1}{2}\sum_n \sum_m y_n y_m \alpha_n \alpha_m x_n^T x_m - \sum_n \alpha_n \\ &= \frac{1}{2}(8\alpha_2^2 - 4\alpha_2\alpha_3 - 6\alpha_2\alpha_4 - 4\alpha_2\alpha_3 + 4\alpha_3^2 + 6\alpha_3\alpha_4 - 6\alpha_4\alpha_2 + 6\alpha_3\alpha_4 + 9\alpha_4^2) - \alpha_1 - \alpha_2 - \alpha_3 - \alpha_4 \\ &= 4\alpha_2^2 + 2\alpha_3^2 + \frac{9}{2}\alpha_4^2 - 4\alpha_2\alpha_3 - 6\alpha_2\alpha_4 + 6\alpha_3\alpha_4 - \alpha_1 - \alpha_2 - \alpha_3 - \alpha_4. \end{aligned}$$

Concerning the constraints, we have that

$$0 = \sum_n y_n \alpha_n = -\alpha_1 - \alpha_2 + \alpha_3 + \alpha_4,$$

or equivalently

$$\alpha_1 + \alpha_2 = \alpha_3 + \alpha_4$$

with $\alpha_1, \alpha_2, \alpha_3, \alpha_4 \geq 0$.

(b) If we replace α_1 with $\alpha_3 + \alpha_4 - \alpha_2$, we obtain

$$\mathcal{L}(\alpha) = 4\alpha_2^2 + 2\alpha_3^2 + \frac{9}{2}\alpha_4^2 - 4\alpha_2\alpha_3 - 6\alpha_2\alpha_4 + 6\alpha_3\alpha_4 - 2\alpha_3 - 2\alpha_4.$$

(c) Now, we fix α_3 and α_4 and we take the derivative of $\mathcal{L}(\alpha)$ with respect to α_2, this gives us that

$$\frac{\partial \mathcal{L}}{\partial \alpha_2} = 8\alpha_2 - 4\alpha_3 - 6\alpha_4.$$

By setting the previous expression to 0, we get that

$$\alpha_2 = \frac{\alpha_3}{2} + \frac{3\alpha_4}{4},$$

and also that

$$\alpha_1 = -\alpha_2 + \alpha_3 + \alpha_4 = \frac{\alpha_3}{2} + \frac{\alpha_4}{4}.$$

These expressions are valid since they are both greater or equal to 0, and obviously

$$\alpha_1 + \alpha_2 = \alpha_3 + \alpha_4.$$

(d) It remains to replace α_2 by its new expression (in (c)), we get that

$$\begin{aligned}
\mathcal{L}(\alpha) &= 4\left(\frac{\alpha_3}{2} + \frac{3\alpha_4}{4}\right)^2 + 2\alpha_3^2 + \frac{9}{2}\alpha_4^2 - 4\left(\frac{\alpha_3}{2} + \frac{3\alpha_4}{4}\right)\alpha_3 - 6\left(\frac{\alpha_3}{2} + \frac{3\alpha_4}{4}\right)\alpha_4 + 6\alpha_3\alpha_4 - 2\alpha_3 - 2\alpha_4 \\
&= \alpha_3^2 + (3\alpha_4 - 2)\alpha_3 + \frac{9}{4}\alpha_4^2 - 2\alpha_4 \\
&= \left(\alpha_3 + \frac{3\alpha_4 - 2}{2}\right)^2 + \frac{9}{4}\alpha_4^2 - 2\alpha_4 - \frac{(3\alpha_4 - 2)^2}{4} \\
&= \left(\alpha_3 + \frac{3\alpha_4 - 2}{2}\right)^2 + \alpha_4 - 1 \geq -1.
\end{aligned}$$

The minimum of the Lagrangian is attained when $\alpha_3 = 1$ and $\alpha_4 = 0$, in this case we also have

$$\alpha_1 = \frac{\alpha_3}{2} + \frac{\alpha_4}{4} = \frac{1}{2}$$

and

$$\alpha_2 = \frac{\alpha_3}{2} + \frac{3\alpha_4}{4} = \frac{1}{2}.$$

Problem 8.4

We have

$$X = \begin{pmatrix} 0 & 0 \\ 2 & 2 \\ 2 & 0 \end{pmatrix} \text{ and } y = \begin{pmatrix} -1 \\ -1 \\ 1 \end{pmatrix}.$$

The Lagrangian is equal to

$$\begin{aligned}
\mathcal{L}(\alpha) &= \frac{1}{2}\sum_n \sum_m y_n y_m \alpha_n \alpha_m x_n^T x_m - \sum_n \alpha_n \\
&= 4\alpha_2^2 - 4\alpha_2\alpha_3 + 2\alpha_3^2 - \alpha_1 - \alpha_2 - \alpha_3 \\
&= 2(\alpha_1^2 - \alpha_1) + 2(\alpha_2^2 - \alpha_2) \geq -\frac{1}{2} - \frac{1}{2} \geq -1;
\end{aligned}$$

and the constraints are $\alpha_3 = \alpha_1 + \alpha_2$ with $\alpha_1, \alpha_2, \alpha_3 \geq 0$. The minimum of the Lagrangian is attained when $\alpha_1 = \alpha_2 = 1/2$ which gives us $\alpha_3 = \alpha_1 + \alpha_2 = 1$. Then, the optimal Lagrange multipliers are

$$\alpha_1^* = \frac{1}{2},\ \alpha_2^* = \frac{1}{2},\ \text{and } \alpha_3^* = 1.$$

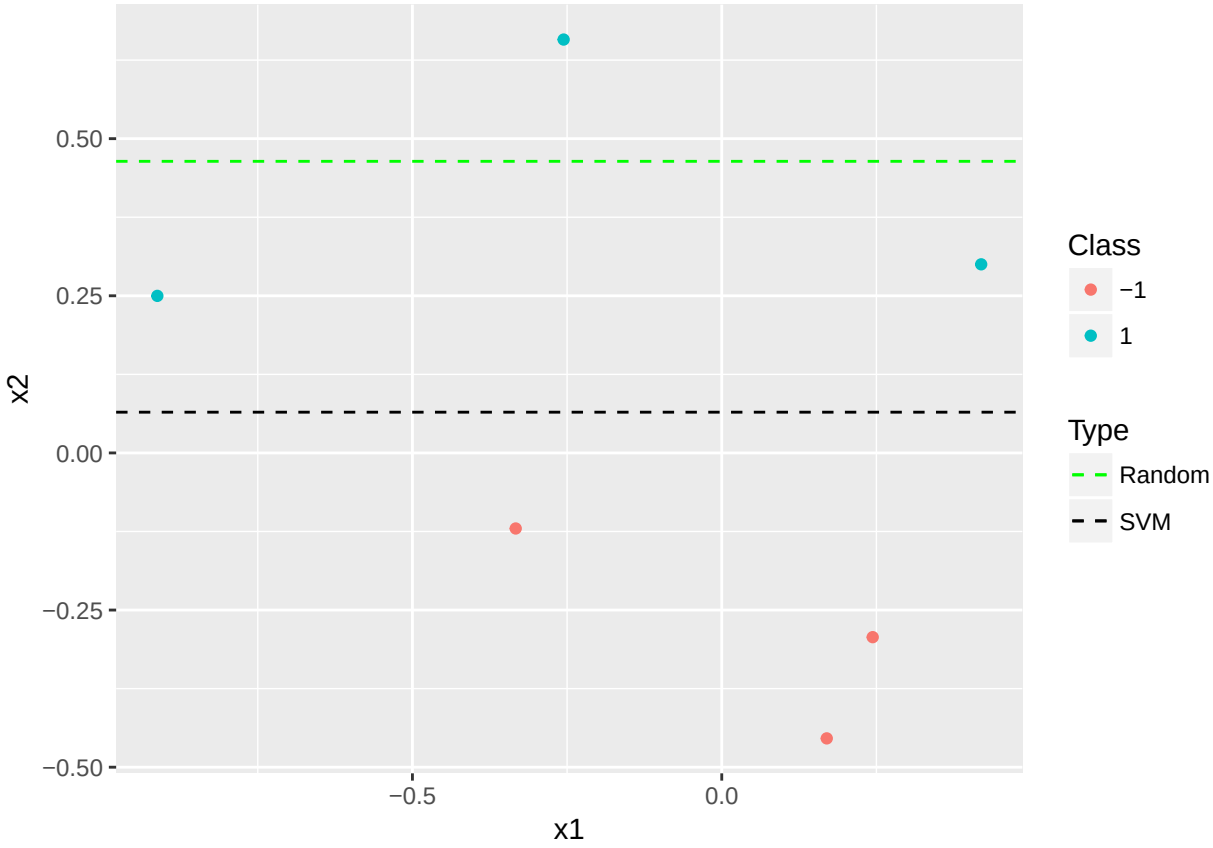
Problem 8.5

(a) Below, we generate three data points uniformly in the upper half of the input space and three data points in the lower half. We also obtain g_{random} and g_{SVM}.

```
set.seed(101)

f <- function(D) {

  return(sign(D$x2))
}

g <- function(D, a) {

  return(sign(D$x2 - a))
}

dataset_gen <- function() {
  D1 <- data.frame(x1 = runif(3, min = -1, max = 1), x2 = runif(3, min = 0, max = 1))
  D2 <- data.frame(x1 = runif(3, min = -1, max = 1), x2 = runif(3, min = -1, max = 0))
  D <- rbind(D1, D2)
  D <- cbind(D, y = as.factor(f(D)))

  return(D)
}

a_random <- function() {
  a <- runif(1, min = -1, max = 1)

  return(a)
}

a_SVM <- function(D) {
  min_pos <- min(D[D$x2 > 0, ]$x2)
  max_neg <- max(D[D$x2 < 0, ]$x2)
  a <- (min_pos + max_neg) / 2

  return(a)
}

D <- dataset_gen()

a_r <- a_random()
a_s <- a_SVM(D)
```

(b) Here, we create a plot of our data and of our two hypotheses.

```
ggplot(D, aes(x1, x2, col = y)) + geom_point() +
  guides(colour = guide_legend(title = "Class")) +
  geom_hline(aes(yintercept = a_r, linetype = "Random"), colour = "green") +
  geom_hline(aes(yintercept = a_s, linetype = "SVM"), colour = "black") +
  scale_linetype_manual(name = "Type", values = c(2, 2),
                        guide = guide_legend(override.aes = list(color = c("green", "black"))))
```

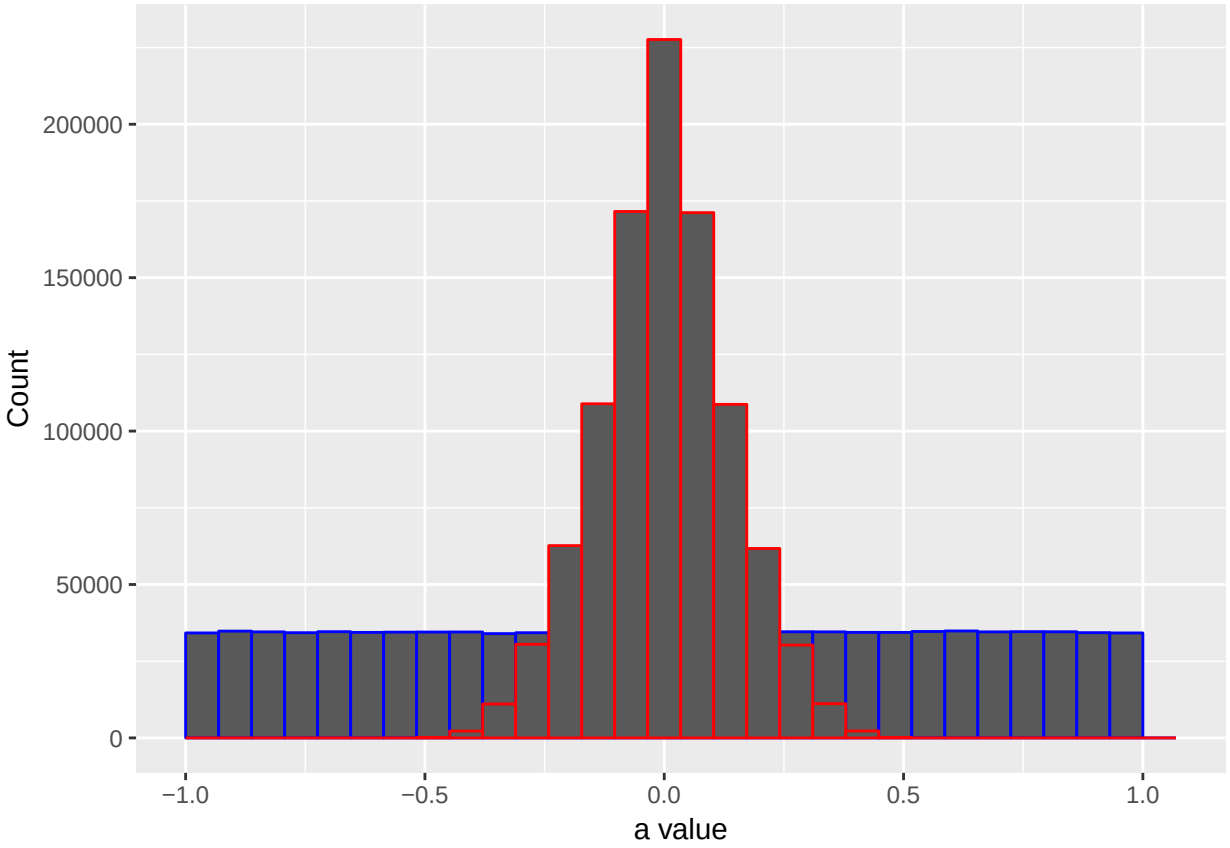


(*c*) Now, we repeat (*a*) for a million data sets to obtain one million random and SVM hypotheses.

```
N <- 1000000
vec_a_r <- rep(NA, N)
vec_a_s <- rep(NA, N)
for (i in 1:N) {
  D <- dataset_gen()
  a_r <- a_random()
  a_s <- a_SVM(D)
  vec_a_r[i] <- a_r
  vec_a_s[i] <- a_s
}

a_repeat <- data.frame(vec_a_r, vec_a_s)
```

(*d*) Below, we give a histogram of the values of a_{random} and another histogram of the values of a_{SVM}.

```
ggplot(a_repeat, aes(x = vec_a_r)) + geom_histogram(colour = "blue") +
  geom_histogram(aes(x = vec_a_s), colour = "red") +
  xlab("a value") + ylab("Count")
```



The two histograms are pretty different : the histogram of a_{random} is clearly uniform on $[-1, 1]$ which was expected; on the other hand, the histogram of a_{SVM} is symmetric and centered around 0 which was expected as well since our data set contains three points in the upper half and three points in the lower half, consequently the distribution of a_{SVM} was going to be centered around 0.

(e) Finally, we estimate the bias and variance for the two algorithms. Here, we cannot use any computation trick, so we have to use the full computation to get our estimates. We begin by estimating the biases of the two algorithms.

```
Ni <- 1000
Nj <- 1000

bias_r.x <- numeric(Nj)
bias_s.x <- numeric(Nj)
for (i in 1:Ni) {
  g_hat_r <- numeric(Ni)
  g_hat_s <- numeric(Ni)
  x.new <- data.frame(x1 = runif(1, min = -1, max = 1), x2 = runif(1, min = -1, max = 1))
  for (j in 1:Nj) {
    D <- dataset_gen()
    a_r <- a_random()
    a_s <- a_SVM(D)
    g_hat_r <- c(g_hat_r, g(x.new, a_r))
    g_hat_s <- c(g_hat_s, g(x.new, a_s))
  }
  g_bar_r <- mean(g_hat_r)
  bias_r.x <- c(bias_r.x, (g_bar_r - f(x.new))^2)
  g_bar_s <- mean(g_hat_s)
```

```
  bias_s.x <- c(bias_s.x, (g_bar_s - f(x.new))^2)
}
bias_r <- mean(bias_r.x)
bias_s <- mean(bias_s.x)
```

We get a bias of 0.2855502 for our random algorithm and a bias of 0.1571874 for SVM; the bias for SVM is clearly better than the bias for our random algorithm which is not surprising. Now, we take a look at the variances.

```
Ni <- 100
Nj <- 100

var_r.x <- numeric(Nj)
var_s.x <- numeric(Nj)
for (i in 1:Ni) {
  g_hat_r <- numeric(Ni)
  diff_hat_r <- numeric(Ni)
  g_hat_s <- numeric(Ni)
  diff_hat_s <- numeric(Ni)
  x.new <- data.frame(x1 = runif(1, min = -1, max = 1), x2 = runif(1, min = -1, max = 1))
  for (j in 1:Nj) {
    D <- dataset_gen()
    a_r <- a_random()
    g_hat_r <- c(g_hat_r, g(x.new, a_r))
    a_s <- a_SVM(D)
    g_hat_s <- c(g_hat_s, g(x.new, a_s))
  }
  g_bar_r <- mean(g_hat_r)
  g_bar_s <- mean(g_hat_s)
  for (j in 1:Nj) {
    D <- dataset_gen()
    a_r <- a_random()
    diff_hat_r <- c(diff_hat_r, (g(x.new, a_r) - g_bar_r)^2)
    a_s <- a_SVM(D)
    diff_hat_s <- c(diff_hat_s, (g(x.new, a_s) - g_bar_s)^2)
  }
  var_r.x <- c(var_r.x, mean(diff_hat_r))
  var_s.x <- c(var_s.x, mean(diff_hat_s))
}
var_r <- mean(var_r.x)
var_s <- mean(var_s.x)
```

We get a variance of 0.1884158 for our random algorithm and a variance of 0.0929888 for SVM; here the variance for SVM is clearly better as well. So, it seems that SVM is much better in our present case.

Problem 8.6

To find the stationary point of $\sum_{n=1}^{N} ||x_n - \mu||^2$, we compute its gradient and we equal it to 0, we get

$$\begin{aligned}\nabla_\mu \sum_{n=1}^{N} ||x_n - \mu||^2 &= \nabla_\mu \sum_{n=1}^{N} (x_n - \mu)^T (x_n - \mu) \\ &= \nabla_\mu \left(\sum_{n=1}^{N} (x_n^T x_n - 2x_n^T \mu + \mu^T \mu) \right) \\ &= 2N\mu - 2\sum_{n=1}^{N} x_n = 0.\end{aligned}$$

This gives us

$$\mu = \frac{1}{N} \sum_{n=1}^{N} x_n.$$

To check that it is actually a minimum, we need to take a look at the Hessian matrix which is

$$Hess(\mu) = 2NI_N;$$

this is a positive definite matrix which means that our stationary point is in fact a minimum.

Problem 8.7

(a) We have that

$$\left\| \sum_{n=1}^{N} y_n x_n \right\|^2 = \left(\sum_{n=1}^{N} y_n x_n \right)^T \left(\sum_{m=1}^{N} y_m x_m \right) = \sum_{m,n} y_n y_m x_n^T x_m.$$

(b) Since $y_n \in \{-1, 1\}$, when $n = m$, we obviously have $y_n y_m = +1$ which gives us immediately $\mathbb{E}[y_n y_m] = 1$ when $n = m$. Now, if $n \neq m$, we may write that

$$\begin{aligned}\mathbb{P}[y_n y_m = 1] &= \mathbb{P}[y_n = 1, y_m = 1] + \mathbb{P}[y_n = -1, y_m = -1] \\ &= \frac{N/2}{N} \cdot \frac{N/2 - 1}{N - 1} + \frac{N/2}{N} \cdot \frac{N/2 - 1}{N - 1} \\ &= \frac{N - 2}{2(N - 1)}.\end{aligned}$$

In this case, we have that

$$\begin{aligned}\mathbb{E}[y_n y_m] &= (+1)\mathbb{P}[y_n y_m = 1] + (-1)\mathbb{P}[y_n y_m = -1] \\ &= \frac{N - 2}{2(N - 1)} - \left(1 - \frac{N - 2}{2(N - 1)} \right) \\ &= \frac{-1}{N - 1}.\end{aligned}$$

Consequently, we get

$$\mathbb{E}[y_n y_m] = \begin{cases} 1 \text{ if } n = m \\ \frac{-1}{N-1} \text{ if } n \neq m \end{cases}.$$

(c) We may write that

$$\begin{aligned}
\mathbb{E}\left[\left\|\sum_{n=1}^{N} y_n x_n\right\|^2\right] &= \mathbb{E}\left[\sum_{m,n} y_n y_m x_n^T x_m\right] \\
&= \sum_{m,n} \mathbb{E}[y_n y_m] x_n^T x_m \\
&= \sum_n \underbrace{\mathbb{E}[y_n y_m]}_{=1} x_n^T x_m + \sum_{m \neq n} \underbrace{\mathbb{E}[y_n y_m]}_{=-1/(N-1)} x_n^T x_m \\
&= \frac{N}{N-1} \sum_n x_n^T x_m - \underbrace{(\frac{1}{N-1} \sum_n x_n^T x_n + \frac{1}{N-1} \sum_{m \neq n} x_n^T x_m)}_{=1/(N-1)\sum_{m,n} x_n^T x_m} \\
&= \frac{N}{N-1} \sum_n x_n^T x_m - \frac{1}{N-1} \sum_{m,n} x_n^T x_m.
\end{aligned}$$

We may also write that

$$\begin{aligned}
\frac{N}{N-1} \sum_{n=1}^{N} ||x_n - \overline{x}||^2 &= \frac{N}{N-1} \sum_n (x_n^T x_n - 2 x_n^T \overline{x} + \overline{x}^T \overline{x}) \\
&= \frac{N}{N-1} \left(\sum_n x_n^T x_n - \underbrace{N \overline{x}^T \overline{x}}_{=(N/N^2)\sum_{m,n} x_n^T x_m} \right) \\
&= \frac{N}{N-1} \sum_n x_n^T x_n - \frac{1}{N-1} \sum_{m,n} x_n^T x_m.
\end{aligned}$$

This proves that

$$\mathbb{E}\left[\left\|\sum_{n=1}^{N} y_n x_n\right\|^2\right] = \frac{N}{N-1} \sum_{n=1}^{N} ||x_n - \overline{x}||^2.$$

(d) The Problem 8.6 gives us that

$$\sum_n ||x_n - \overline{x}||^2 \leq \sum_n ||x_n - \mu||^2$$

for all μ; so we get that

$$\sum_n ||x_n - \overline{x}||^2 \leq \sum_n \underbrace{||x_n||^2}_{\leq R^2} \leq NR^2.$$

(e) From parts (b) and (c), we have that

$$\begin{aligned}
\mathbb{E}\left[\left\|\sum_{n=1}^{N} y_n x_n\right\|^2\right] &= \frac{N}{N-1} \sum_{n=1}^{N} ||x_n - \overline{x}||^2 \\
&\leq \frac{N}{N-1} NR^2 = \frac{N^2 R^2}{N-1}.
\end{aligned}$$

Now, let us assume that

$$\mathbb{P}\left[\left\|\sum_n y_n x_n\right\| \leq \frac{NR}{\sqrt{N-1}}\right] = 0,$$

or equivalently

$$\mathbb{P}\left[\left\|\sum_n y_n x_n\right\| > \frac{NR}{\sqrt{N-1}}\right] = 1.$$

In this case, we may write that

$$\underbrace{\mathbb{E}\left[\left\|\sum_{n=1}^N y_n x_n\right\|^2\right]}_{>N^2R^2/(N-1)} > \frac{N^2R^2}{N-1},$$

which is impossible. We conclude by writing that

$$\mathbb{P}\left[\left\|\sum_n y_n x_n\right\| \leq \frac{NR}{\sqrt{N-1}}\right] > 0.$$

Problem 8.8

(a) We begin by assuming that the y_n are numbered in such a way that $y_1, \cdots, y_k = +1$ and $y_{k+1}, \cdots, y_N = -1$. We know that

$$\rho||w|| \leq y_n(w^T x_n + b)$$

for $n = 1, \cdots, N$; if we sum this expression for $n = 1, \cdots, k$ and for $n = k+1, \cdots, N$, we get that

$$\underbrace{\sum_{n=1}^k \rho||w||}_{=k\rho||w||} \leq \sum_{n=1}^k y_n(w^T x_n + b) = \sum_{n=1}^k y_n w^T x_n + kb$$

and

$$\underbrace{\sum_{n=k+1}^N \rho||w||}_{=(k+1)\rho||w||} \leq \sum_{n=k+1}^N y_n(w^T x_n + b) = \sum_{n=k+1}^N y_n w^T x_n - (k+1)b.$$

By multiplying the first expression by $k+1$ and the second one by k, we may write that

$$\begin{aligned}\underbrace{(k+1)k\rho||w|| + k(k+1)\rho||w||}_{=2k(k+1)\rho||w||} &\leq (k+1)\left(\sum_{n=1}^k y_n w^T x_n + kb\right) + k\left(\sum_{n=k+1}^N y_n w^T x_n - (k+1)b\right) \\ &= (k+1)\sum_{n=1}^k y_n w^T x_n + k\sum_{n=k+1}^N y_n w^T x_n.\end{aligned}$$

This means that we have

$$\begin{aligned} 2\rho||w|| &\leq \underbrace{\frac{1}{k}}_{=l_n}\sum_{n=1}^{k} y_n w^T x_n + \underbrace{\frac{1}{k+1}}_{=l_n}\sum_{n=k+1}^{N} y_n w^T x_n \\ &= w^T \sum_{n=1}^{N} l_n y_n x_n \\ &\leq ||w|| \cdot \left\| \sum_{n=1}^{N} l_n y_n x_n \right\|, \end{aligned}$$

the last inequality is due to the Cauchy-Schwarz inequality. Consequently, we now have that

$$2\rho \leq \left\| \sum_{n=1}^{N} l_n y_n x_n \right\|$$

since $w \neq 0$.

(b) (i) We have immediately that

$$\left\| \sum_{n=1}^{N} l_n y_n x_n \right\|^2 = \sum_{n=1}^{N}\sum_{m=1}^{N} l_n l_m y_n y_m x_n^T x_m.$$

(ii) When $m = n$, we may write that

$$\begin{aligned} \mathbb{E}[l_n l_m \underbrace{y_n y_m}_{=1}] &= \frac{1}{k^2}\underbrace{\mathbb{P}[y_n = +1]}_{=k/N} + \frac{1}{(k+1)^2}\underbrace{\mathbb{P}[y_n = -1]}_{=(k+1)/N} \\ &= \frac{1}{kN} + \frac{1}{(k+1)N} = \frac{1}{k(k+1)}. \end{aligned}$$

(iii) When $m \neq n$, we may write that

$$\begin{aligned} \mathbb{E}[l_n l_m y_n y_m] &= \frac{1}{k^2}\underbrace{\mathbb{P}[y_n = +1, y_m = +1]}_{=k(k-1)/N(N-1)} + \frac{1}{(k+1)^2}\underbrace{\mathbb{P}[y_n = -1, y_m = -1]}_{=k(k+1)/N(N-1)} - \frac{2}{k(k+1)}\underbrace{\mathbb{P}[y_n = +1, y_m = -1]}_{=k(k+1)/N(N-1)} \\ &= \frac{(k-1)(k+1) + k^2 - 2k(k+1)}{k(k+1)N(N-1)} = \frac{-1}{k(k+1)(N-1)}. \end{aligned}$$

(iv) We have that

$$\begin{aligned} \mathbb{E}\left[\left\| \sum_{n=1}^{N} l_n y_n x_n \right\|^2 \right] &= \sum_{m,n} \mathbb{E}[l_n l_m y_n y_m] x_n^T x_m \\ &= \frac{1}{k(k+1)}\sum_{n} x_n^T x_n - \frac{1}{k(k+1)(N-1)}\sum_{m,n} x_n^T x_m \\ &= \frac{1}{k(k+1)(N-1)}[(N-1)\sum_{n} x_n^T x_n - \sum_{m \neq n} x_n^T x_m] \\ &= \frac{1}{k(k+1)(N-1)}[N\sum_{n} x_n^T x_n - \sum_{m,n} x_n^T x_m]; \end{aligned}$$

and also (see Problem 8.7) that

$$\begin{aligned}\frac{N}{(N-1)k(k+1)}\sum_n ||x_n-\overline{x}||^2 &= \frac{N}{(N-1)k(k+1)}[\sum_n x_n^T x_n - \frac{1}{N}\sum_{m,n} x_n^T x_m] \\ &= \frac{1}{k(k+1)(N-1)}[N\sum_n x_n^T x_n - \sum_{m,n} x_n^T x_m].\end{aligned}$$

Then, we can conclude that

$$\mathbb{E}\left[\left\|\sum_{n=1}^N l_n y_n x_n\right\|^2\right] = \frac{N}{(N-1)k(k+1)}\sum_n ||x_n-\overline{x}||^2.$$

(v) In a similar way as Problem 8.6, we may write that

$$\mathbb{E}\left[\left\|\sum_{n=1}^N l_n y_n x_n\right\|^2\right] = \frac{N}{(N-1)k(k+1)}\underbrace{\sum_n ||x_n-\overline{x}||^2}_{\leq \sum_n ||x_n||^2 \leq NR^2} \leq \frac{N^2R^2}{(N-1)k(k+1)}.$$

Then, if we assume that

$$\mathbb{P}\left[\left\|\sum_{n=1}^N l_n y_n x_n\right\| \leq \frac{NR}{\sqrt{k(k+1)(N-1)}}\right] = 0,$$

or equivalently that

$$\mathbb{P}\left[\left\|\sum_{n=1}^N l_n y_n x_n\right\| > \frac{NR}{\sqrt{k(k+1)(N-1)}}\right] = 1,$$

we get that

$$\mathbb{E}\left[\left\|\sum_{n=1}^N l_n y_n x_n\right\|^2\right] > \frac{N^2R^2}{k(k+1)(N-1)},$$

which is impossible. So we are now able to conclude that there exists a labeling with k labels being $+1$ for which

$$\begin{aligned}\left\|\sum_{n=1}^N l_n y_n x_n\right\| &\leq \frac{NR}{\sqrt{k(k+1)(N-1)}} \\ &= \frac{2NR}{(N-1)\sqrt{N+1}}.\end{aligned}$$

(c) From points (a) and (b) and for a specific labeling, we have that

$$2\rho \leq \left\|\sum_{n=1}^N l_n y_n x_n\right\| \leq \frac{2NR}{(N-1)\sqrt{N+1}},$$

this can be written as

$$\frac{R^2}{\rho^2} \geq \frac{(N-1)^2(N+1)}{N^2} = N - 1 - \frac{1}{N} + \frac{1}{N^2} \geq N - 1 - \frac{1}{N}.$$

In conclusion, we get that

$$N \leq \frac{R^2}{\rho^2} + \frac{1}{N} + 1.$$

Problem 8.9

Let g be the maximum margin classifier for the data set $\mathcal{D}$, this means that g may be constructed from $\alpha^* = (\alpha_1^*, \cdots, \alpha_N^*)$ solution to the following problem

$$\min_{\alpha} \underbrace{\sum_{m=1}^{N}\sum_{n=1}^{N} y_m y_n \alpha_m \alpha_n x_m^T x_n - \sum_{n=1}^{N} \alpha_n}_{=f(\alpha_1,\cdots,\alpha_N)}$$

subject to

$$\sum_{n=1}^{N} y_n \alpha_n = 0 \text{ and } \alpha_n \geq 0$$

for $n = 1, \cdots, N$. Now, let us assume that we remove the point (x_N, y_N) from $\mathcal{D}$ such as $\alpha_N^* = 0$; in this case, we obtain a new problem on our altered data set $\mathcal{D}^-$ which is

$$\min_{\alpha} \underbrace{\sum_{m=1}^{N-1}\sum_{n=1}^{N-1} y_m y_n \alpha_m \alpha_n x_m^T x_n - \sum_{n=1}^{N-1} \alpha_n}_{=f(\alpha_1,\cdots,\alpha_{N-1},0)}$$

subject to

$$\sum_{n=1}^{N-1} y_n \alpha_n = 0 \text{ and } \alpha_n \geq 0$$

for $n = 1, \cdots, N-1$. It is easy to see that $(\alpha_1^*, \cdots, \alpha_{N-1}^*)$ is still feasible for our new problem, we have

$$\sum_{n=1}^{N-1} y_n \alpha_n^* = \sum_{n=1}^{N} y_n \alpha_n^* = 0$$

and $\alpha_n^* \geq 0$ for $n = 1, \cdots, N-1$. Now, we consider $\alpha' = (\alpha_1', \cdots, \alpha_{N-1}')$ another solution to our new problem such that

$$f(\alpha_1', \cdots, \alpha_{N-1}', 0) < f(\alpha_1^*, \cdots, \alpha_{N-1}^*, 0).$$

First, we notice that

$$f(\alpha_1^*, \cdots, \alpha_{N-1}^*, 0) \geq \min_{\alpha} f(\alpha_1, \cdots, \alpha_{N-1}, 0);$$

then, we have that

$$f(\alpha_1^*, \cdots, \alpha_{N-1}^*, 0) = f(\alpha_1^*, \cdots, \alpha_N^*) = \min_{\alpha} f(\alpha_1, \cdots, \alpha_N) \leq \min_{\alpha} f(\alpha_1, \cdots, \alpha_{N-1}, 0).$$

Combining the inequalities above, we obtain

$$f(\alpha_1^*, \cdots, \alpha_{N-1}^*, 0) = \min_{\alpha} f(\alpha_1, \cdots, \alpha_{N-1}, 0);$$

thus we have proven that $(\alpha_1^*, \cdots, \alpha_{N-1}^*)$ is solution to our new problem. Consequently, we also get that

$$f(\alpha_1', \cdots, \alpha_{N-1}', 0) < f(\alpha_1^*, \cdots, \alpha_{N-1}^*, 0) = \min_{\alpha} f(\alpha_1, \cdots, \alpha_{N-1}, 0)$$

which is impossible. This means that $(\alpha_1^*, \cdots, \alpha_{N-1}^*)$ is the unique solution to our new problem. Thus, the optimal hyerplane for our new problem is defined by

$$w^* = \sum_{n=1}^{N-1} y_n \alpha_n^* x_n = \sum_{n=1}^{N} y_n \alpha_n^* x_n$$

and

$$b^* = y_s - w^{*T} x_s$$

where $\alpha_s \neq 0$ which is exactly the optimal hyperplane for our former problem. This means that g is also the maximum margin separator for $\mathcal{D}^-$.

Problem 8.10

We consider all the points (x_n, y_n) $(n = 1 \cdots, S)$ which are on the boundary of the optimal fat-hyperplane defined by (b^*, w^*); these points are characterized by the condition

$$y_n(w^{*T}x_n + b^*) = 1.$$

Certain points among these are what we call essential support vectors (those with $\alpha_n > 0$); if we put these equalities in matrix form, we get

$$A_S u^* = c$$

where

$$A_S = \begin{pmatrix} y_1 & y_1 x_1^T \\ \vdots & \vdots \\ y_S & y_S x_S^T \end{pmatrix}, \; u^* = \begin{pmatrix} b^* \\ w^* \end{pmatrix} \text{ and } c = \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}.$$

Thus, the vector u^* is determined by the matrix $A_S \in \mathbb{R}^{S \times (d+1)}$ only, which means that only points on the boundary are actually needed to compute u^*. Moreover, we know that only the rows of matrix A_S which are linearly independant are needed to compute u^*, these are exactly our essential support vectors and there are $\text{rank}(A_S) = r \leq d + 1$ of them. We also know that

$$E_{cv} = \frac{1}{N} \sum_{n=1}^{N} e_n$$

where e_n is the leave-one-out error for (x_n, y_n). Since removing any data point that is not an essential support vector results in the same separator, and since (x_n, y_n) was classified correctly before removal, it will remain correct after removal; we get that $e_n = 0$ for these non essential support vectors, and $e_n \leq 1$ for the support vectors. Consequently, we may write that

$$E_{cv} = \frac{1}{N} \sum_{n=1}^{N} e_n \leq \frac{r}{N} \leq \frac{d+1}{N}.$$

Problem 8.11

(a) The result obtained in Problem 1.3 gives us that

$$T \leq \frac{R^2 ||w^*||^2}{\rho'^2}$$

where w^* is an optimal set of weights obtained in this case by the SVM algorithm, $R = \max_n ||x_n||$, and $\rho' = \min_n y_n(w^{*T}x_n)$. Since, by construction of w^*, we have that $||w^*|| = 1/\rho$ and $\rho' = \min_n y_n(w^{*T}x_n) = 1$. Consequently, we get that

$$T \leq \frac{R^2 ||w^*||^2}{\rho'^2} = \frac{R^2}{\rho^2}.$$

(b) Since R^2/ρ^2 is the maximum number of updates for PLA, it is obvious that at most R^2/ρ^2 different points are visited during the course of this algorithm.

(c) The PLA algorithm generates its final solution by taking a linear combination of the visited points, so deleting the points that have not been visited does not affect this final solution.

(d) Since deleting the points that have not been visited does not affect the final solution, the corresponding e_n will be equal to 0, thus

$$E_{cv}(PLA) = \frac{1}{N} \sum_{n=1}^{N} e_n \leq \frac{\#\text{visited points}}{N} \leq \frac{R^2}{N\rho^2}.$$

Problem 8.12

The soft-margin optimization problem is

$$\min_{w,b,\xi} \frac{1}{2} w^T w + C \sum_{n=1}^{N} \xi_n$$

subject to

$$y_n(w^T x_n + b) \geq 1 - \xi_n \text{ and } \xi_n \geq 0$$

for $i = 1, \cdots, N$; and the optimization problem of Problem 3.6 (c) is

$$\min_{w,\xi} \sum_{n=1}^{N} \xi_n$$

subject to

$$y_n(w^T x_n) \geq 1 - \xi_n \text{ and } \xi_n \geq 0$$

for $i = 1, \cdots, N$. First, we may note that the constraints are identical for these two problems (by taking into account the difference in notation). Then obviously if $C \to \infty$, the corresponding term will render the term in $w^T w$ obsolete, so in this case we only need to minimize $\sum_n \xi_n$ which is equivalent to the soft-margin optimization problem.

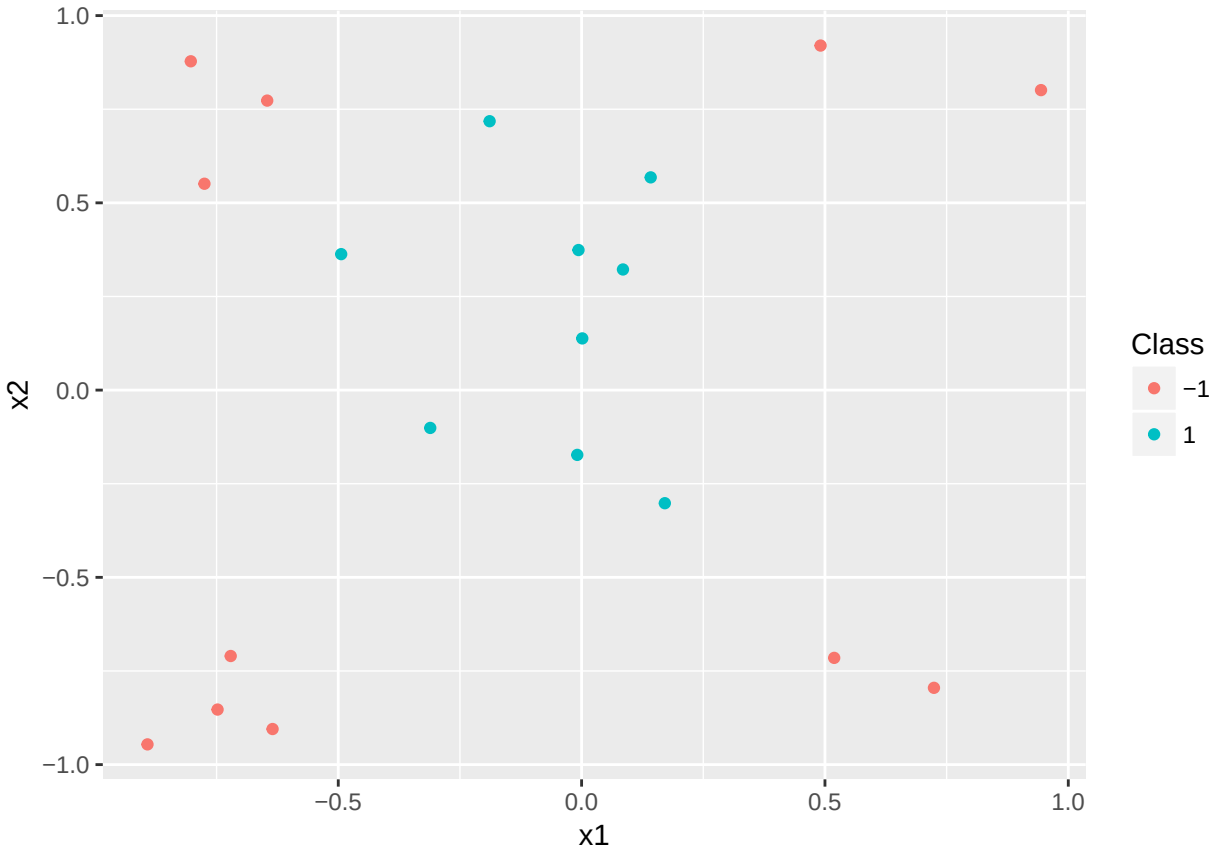
Problem 8.13

(a) We use our data set with the 2nd and 3rd order polynomial transforms Φ_2 and Φ_3 and the pseudo-inverse algorithm for linear regression to get weights $\tilde{w}$ for our final hypothesis in $\mathcal{Z}$-space

$$g(x) = \text{sign}(\tilde{w}^T \Phi(x) + \tilde{b}).$$

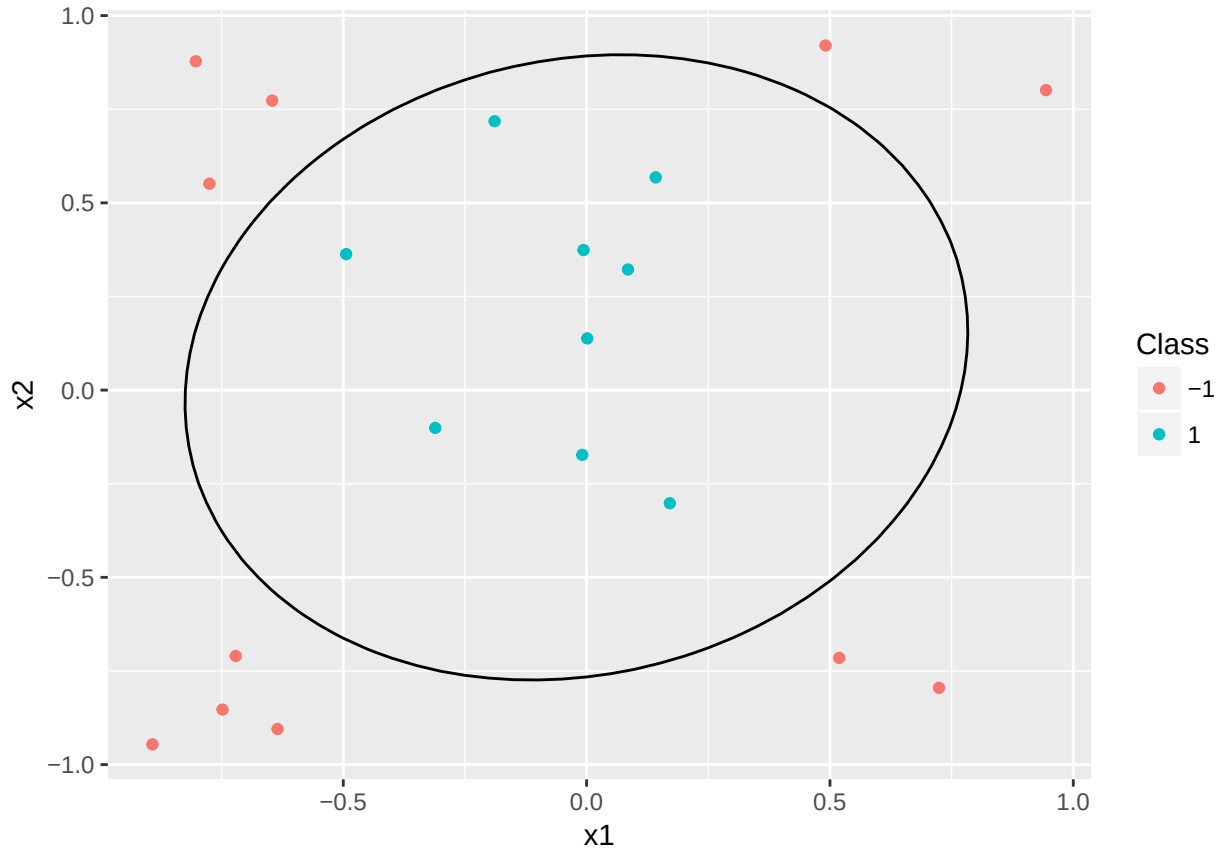
We also plot our data set.

```
X <- matrix(c(-0.494, 0.363, -0.311, -0.101, -0.0064, 0.374, -0.0089, -0.173, 0.0014,
              0.138, -0.189, 0.718, 0.085, 0.32208, 0.171, -0.302, 0.142, 0.568, 0.491,
              0.920, -0.892, -0.946, -0.721, -0.710, 0.519, -0.715, -0.775, 0.551,
              -0.646, 0.773, -0.803, 0.878, 0.944, 0.801, 0.724, -0.795, -0.748,
              -0.853, -0.635, -0.905), ncol = 2, byrow = TRUE)
D <- as.data.frame(X)
colnames(D) <- c("x1", "x2")
y <- c(rep(1, 9), rep(-1, 11))
D <- cbind(D, class = y)

p <- ggplot(D, aes(x = x1, y = x2, color = as.factor(class))) + geom_point() +
  guides(colour = guide_legend(title = "Class"))
p
```

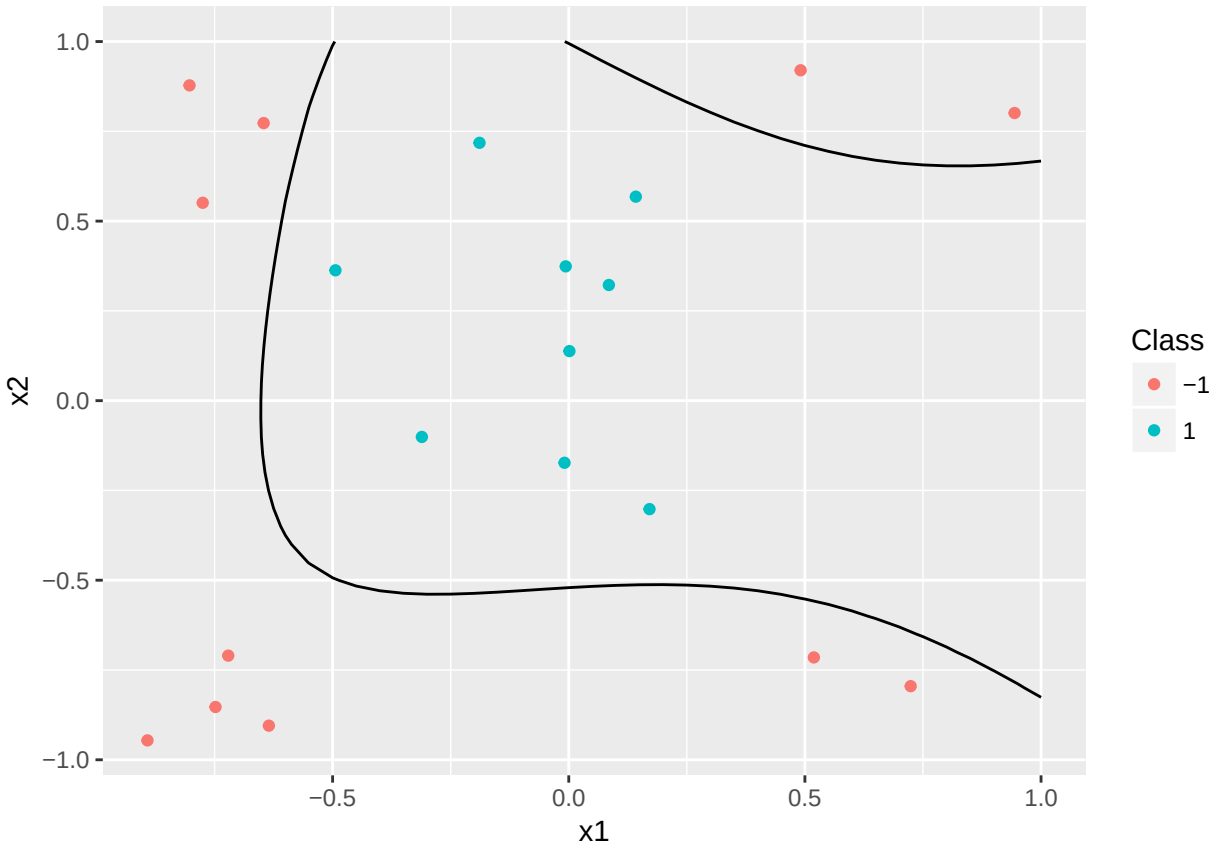


Below, we plot the classification regions for our final hypothesis in $\mathcal{X}$-space for Φ_2 and Φ_3 respectively.

```
D_Phi2 <- data.frame(x1 = D$x1, x2 = D$x2, x1_sq = D$x1^2, x1x2 = D$x1 * D$x2,
                     x2_sq = D$x2^2, class = D$class)
D_Phi3 <- data.frame(x1 = D$x1, x2 = D$x2, x1_sq = D$x1^2, x1x2 = D$x1 * D$x2,
                      x2_sq = D$x2^2, x1_cub = D$x1^3, x1_sqx2 = D$x1^2 * D$ x2,
                      x1x2_sq = D$x1 * D$x2^2, x2_cub = D$x2^3, class = D$class)

X_Phi2 <- as.matrix(cbind(1, D_Phi2[, 1:5]))
y <- D$class
X_Phi2_cross <- solve(t(X_Phi2) %*% X_Phi2) %*% t(X_Phi2)
w_lin_Phi2 <- as.vector(X_Phi2_cross %*% y)

X_Phi3 <- as.matrix(cbind(1, D_Phi3[, 1:9]))
y <- D$class
X_Phi3_cross <- solve(t(X_Phi3) %*% X_Phi3) %*% t(X_Phi3)
w_lin_Phi3 <- as.vector(X_Phi3_cross %*% y)

cc2 <- emdbook::curve3d(1 * w_lin_Phi2[1] + x * w_lin_Phi2[2] + y * w_lin_Phi2[3] +
                          x^2 * w_lin_Phi2[4] + x * y * w_lin_Phi2[5] + y^2 * w_lin_Phi2[6],
                        xlim = c(-1, 1), ylim = c(-1, 1), sys3d = "none")
dimnames(cc2$z) <- list(cc2$x, cc2$y)
mm2 <- reshape2::melt(cc2$z)
p + geom_contour(data = mm2, aes(x = Var1, y = Var2, z = value), breaks = 0,
                 colour = "black")
```




```
cc3 <- emdbook::curve3d(1 * w_lin_Phi3[1] + x * w_lin_Phi3[2] + y * w_lin_Phi3[3] +
                          x^2 * w_lin_Phi3[4] + x * y * w_lin_Phi3[5] +
                          y^2 * w_lin_Phi3[6] + x^3 * w_lin_Phi3[7] +
                          x^2 * y * w_lin_Phi3[8] + x * y^2 * w_lin_Phi3[9] +
                          y^3 * w_lin_Phi3[10], xlim = c(-1, 1), ylim = c(-1, 1),
                        sys3d = "none")
dimnames(cc3$z) <- list(cc3$x, cc3$y)
mm3 <- reshape2::melt(cc3$z)
p + geom_contour(data = mm3, aes(x = Var1, y = Var2, z = value), breaks = 0,
                 colour = "black")
```



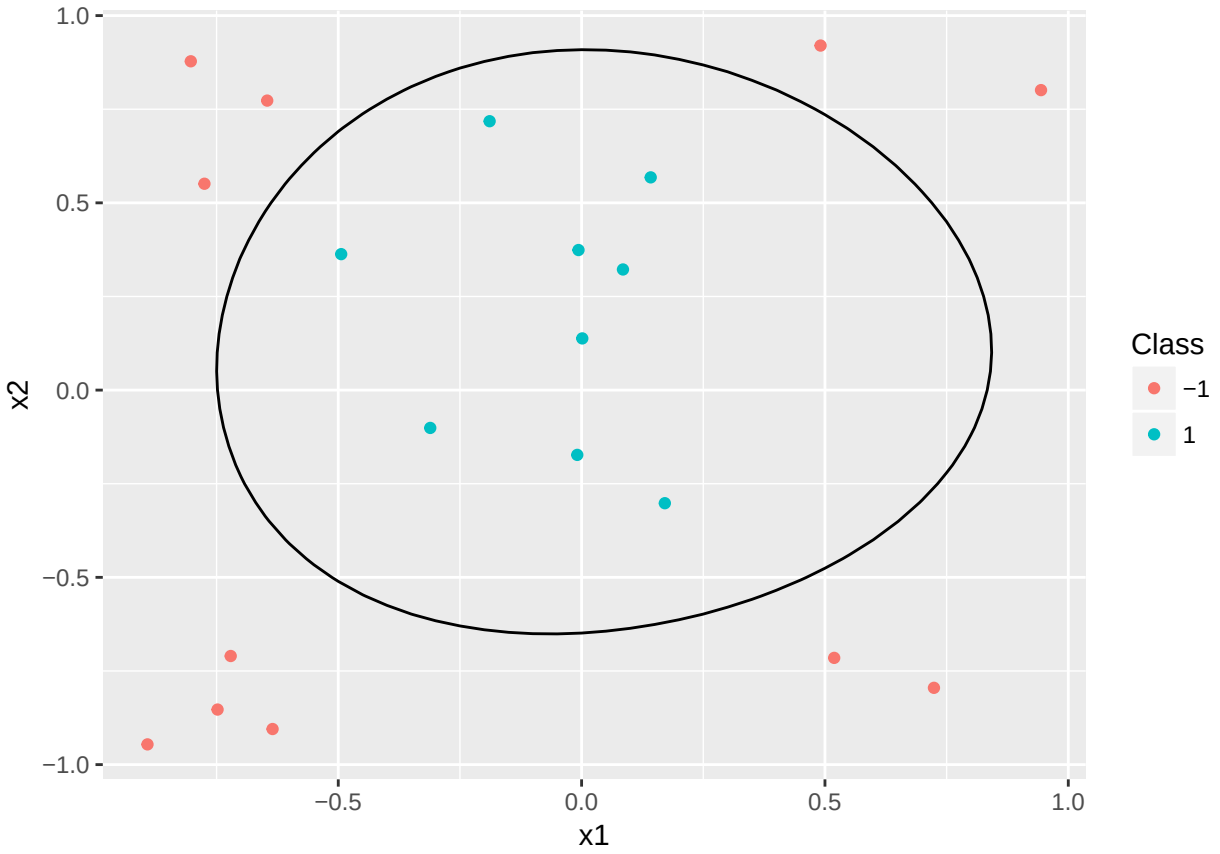
(b) It seems that the fit obtained with Φ_3 has overfitted the data since it cannot capture the overall (circular) data structure very well.

(c) Usually regularization helps to avoid overfitting, so we now use the pseudo-inverse algorithm

$$w_{reg} = (X^T X + \lambda I)^{-1} X^T y$$

with $\lambda = 1$ to address the overfitting identified in part (b).

```
lambda <- 1
X_Phi3_cross_reg <- solve(t(X_Phi3) %*% X_Phi3 + lambda * diag(10)) %*% t(X_Phi3)
w_lin_Phi3_reg <- as.vector(X_Phi3_cross_reg %*% y)

cc3_reg <- emdbook::curve3d(1 * w_lin_Phi3_reg[1] + x * w_lin_Phi3_reg[2] +
                              y * w_lin_Phi3_reg[3] + x^2 * w_lin_Phi3_reg[4] +
                              x * y * w_lin_Phi3_reg[5] + y^2 * w_lin_Phi3_reg[6] +
                              x^3 * w_lin_Phi3_reg[7] + x^2 * y * w_lin_Phi3_reg[8] +
                              x * y^2 * w_lin_Phi3_reg[9] + y^3 * w_lin_Phi3_reg[10],
                            xlim = c(-1, 1), ylim = c(-1, 1), sys3d = "none")
dimnames(cc3_reg$z) <- list(cc3_reg$x, cc3_reg$y)
mm3_reg <- reshape2::melt(cc3_reg$z)
p + geom_contour(data = mm3_reg, aes(x = Var1, y = Var2, z = value), breaks = 0,
                 colour = "black")
```



We may clearly see that the 3rd order polynomial transform Φ_3 regularized with $\lambda = 1$ results in a classifier almost identical to the 2nd order polynomial transform Φ_2.

Problem 8.14

(a) We begin by taking a look at the Kernel-Gram matrix K defined by $K_{ij} = \Phi(x_i)^T\Phi(x_j)$, we have that

$$K = \begin{pmatrix} - & \Phi(x_1)^T & - \\ & \vdots & \\ - & \Phi(x_N)^T & - \end{pmatrix} \begin{pmatrix} | & & | \\ \Phi(x_1) & \cdots & \Phi(x_N) \\ | & & | \end{pmatrix} = Z^T Z.$$

With that in mind, we may write that

$$\begin{aligned} E_{aug}(\tilde{w}) &= ||Z\tilde{w} - y||^2 + \lambda \tilde{w}^T \tilde{w} \\ &= ||ZZ^T\beta - y||^2 + \lambda \beta^T ZZ^T \beta \\ &= ||K\beta - y||^2 + \lambda \beta^T K \beta = E(\beta). \end{aligned}$$

Since $\tilde{w}^*$ minimizes $E_{aug}(\tilde{w})$, we obviously have that β^* minimizes $E(\beta)$.

(b) The matrix K is clearly symmetric since

$$K^T = (Z^T Z)^T = Z^T Z = K.$$

(c) To solve the minimization problem in part (a), we have to compute the gradient of $E(\beta)$; we get that

$$\begin{aligned}\nabla_\beta E(\beta) &= \nabla_\beta(\beta^T K^T K\beta - 2\beta^T K^T y + y^T y + \lambda\beta^T K\beta)\\ &= (K^T K + K^T K)\beta - 2K^T y + \lambda(K + K^T)\beta\\ &= 2K[(K + \lambda I)\beta - y].\end{aligned}$$

To get $\nabla_\beta E(\beta)$ to be 0, one should solve for β that satisfies

$$(K + \lambda I)\beta = y.$$

We may note that K is positive semidefinite since for any $x \neq 0$, we have

$$x^T Kx = x^T Z^T Zx = ||Zx||^2 \geq 0;$$

and since $\lambda > 0$, we can conclude that $K + \lambda I$ is positive definite, and consequently invertible. Thus the solution β^* to the minimization problem is

$$\beta^* = (K + \lambda I)^{-1} y.$$

The β^* above can be computed without visiting the $\mathcal{Z}$-space because in the case of specific kernels (like polynomial kernels or Gauss-RBF kernels), we can use the kernel trick.

(d) The final hypothesis is given by

$$\begin{aligned}g(x) &= \text{sign}(\tilde{w}^{*T}\Phi(x))\\ &= \text{sign}\left(\sum_{n=1}^{N} \beta_n^* z_n^T \Phi(x)\right)\\ &= \text{sign}\left(\sum_{n=1}^{N} \beta_n^* \underbrace{\Phi(x_n)^T\Phi(x)}_{=K(x_n,x)}\right)\\ &= \text{sign}\left(\sum_{n=1}^{N} \beta_n^* K(x_n, x)\right).\end{aligned}$$

Problem 8.15

(a) Since $\mathcal{H}_i \subset \mathcal{H}_{i+1}$, we know that $|\mathcal{H}_i| \leq |\mathcal{H}_{i+1}|$, and

$$E_{in}(g_i) = \min_{h\in\mathcal{H}_i} E_{in}(h) \geq \min_{h\in\mathcal{H}_{i+1}} E_{in}(h) = E_{in}(g_{i+1})$$

for any $i = 1, 2, \cdots$.

(b) Let $p_i = \mathbb{P}[g^* \in \mathcal{H}_i] = \mathbb{P}[g^* = g_i]$, so if p_i is small then $\Omega(\mathcal{H}_i)$ is large, which implies that the model is complex. It is obvious that

$$g^* \in \mathcal{H}_i \Rightarrow g^* \in \mathcal{H}_{i+1},$$

thus we get that

$$p_i = \mathbb{P}[g^* \in \mathcal{H}_i] \leq \mathbb{P}[g^* \in \mathcal{H}_{i+1}] = p_{i+1}$$

for any $i = 1, 2, \cdots$.

(c) We know from the generalization bound that

$$\begin{aligned}\mathbb{P}[|E_{in}(g_i) - E_{out}(g_i)| > \epsilon | g^* = g_i] &\leq \frac{\mathbb{P}[|E_{in}(g_i) - E_{out}(g_i)| > \epsilon \cap g^* = g_i]}{\mathbb{P}[g^* = g_i]} \\ &\leq \frac{\mathbb{P}[|E_{in}(g_i) - E_{out}(g_i)| > \epsilon]}{\mathbb{P}[g^* = g_i]} \\ &\leq \frac{4 m_{\mathcal{H}_i}(2N) e^{-\epsilon^2 N/8}}{p_i}.\end{aligned}$$

Problem 8.16

We know that a soft order constraint is typically formulated as

$$\sum_{q=0}^{Q} w_q^2 \leq C;$$

so if we consider $0 < C_1 \leq C_2$, we know that

$$\sum_{q=0}^{Q} w_q^2 \leq C_1 \leq C_2.$$

This means that $\mathcal{H}_{C_1} \subset \mathcal{H}_{C_2}$ which can be posed within the SRM framework. However, the $\{\mathcal{H}_\lambda\}_{\lambda>0}$ with regular items does not contain such a relationship, so the SRM framework cannot be used in this case.

Problem 8.17

(a) Yes, it seems possible that

$$E_{in}(\mathcal{H}_m) < E_{in}(\mathcal{H}_{m+1})$$

because the VC-dimension characterizes the complexity of the hypothesis set.

(b) We know that

$$\mathbb{P}[|E_{in}(g_m) - E_{out}(g_m)| > \epsilon | g^* = g_i] \leq \frac{4 m_{\mathcal{H}_m}(2N) e^{-\epsilon^2 N/8}}{p_m},$$

and also that $m_{\mathcal{H}_m}(2N) \leq (2N)^{d_{VC}(\mathcal{H}_m)} + 1$. Consequently, we may write that

$$\mathbb{P}[|E_{in}(g_m) - E_{out}(g_m)| > \epsilon | g^* = g_i] \leq \frac{4((2N)^{d_{VC}(\mathcal{H}_m)} + 1) e^{-\epsilon^2 N/8}}{p_m}.$$

Problem 8.18

(a) Yes, in my opinion since we only have m hypotheses in our hypothesis subset $\mathcal{H}_m$, we know, with probability at least $1 - \delta$, that

$$E_{out}(h_m) \leq E_{in}(h_m) + \sqrt{\frac{\ln(2m/\delta)}{2N}} \leq \nu + \sqrt{\frac{\ln(2m/\delta)}{2N}}.$$

(b) We can formulate this process within the SRM structure since, obviously, the $\mathcal{H}_m$'s form a structure ($\mathcal{H}_m \subset \mathcal{H}_{m+1}$). It is easy to see that if we output h_m, this means that

$$h_m = \text{argmin}_{h \in \mathcal{H}_m} E_{in}(h).$$

Moreover, we also have that

$$h^* = \text{argmin}_{h_m} E_{in}(h_m) + \Omega(\mathcal{H}_m)$$

where $\Omega(\mathcal{H}_m) = m$ this will give priority to the h_m corresponding to the smaller subscript m, which characterizes exactly the SRM framework.

(c) Since we are within the SRM framework, we may write that

$$\mathbb{P}[|E_{in}(h_m) - E_{out}(h_m)| > \epsilon | h^* = h_i] \leq \frac{4m_{\mathcal{H}_m}(2N)e^{-\epsilon^2 N/8}}{p_m}.$$