Machine Learning Homework 1

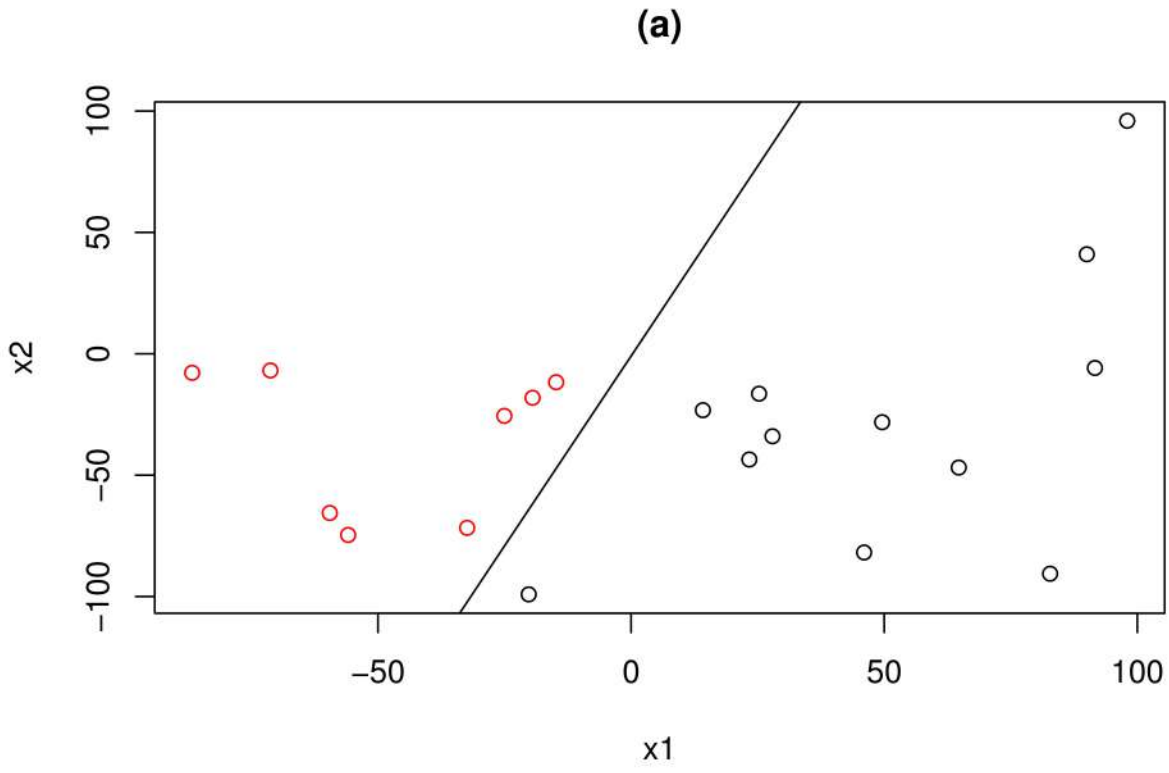
Karan Sarkar

September 9, 2019

Problem 1.4

(a)

```
h <- function(x, w) {
  sign(cbind(1, x) %*% t(w))
}

w0 <- runif(1, -100, 100)
w1 <- runif(1, -100, 100)
w2 <- runif(1, -100, 100)

f <- function(x) {
  weights = matrix(c(w0, w1, w2), ncol = 3)
  h(x, weights)
}

x <- matrix(runif(40, -100, 100), ncol = 2)
y <- f(x)

plot(x, col = as.factor(y), xlab = "x1", ylab = "x2", main = "(a)")
abline(a= -w0/w2, b= -w1/w2)
```

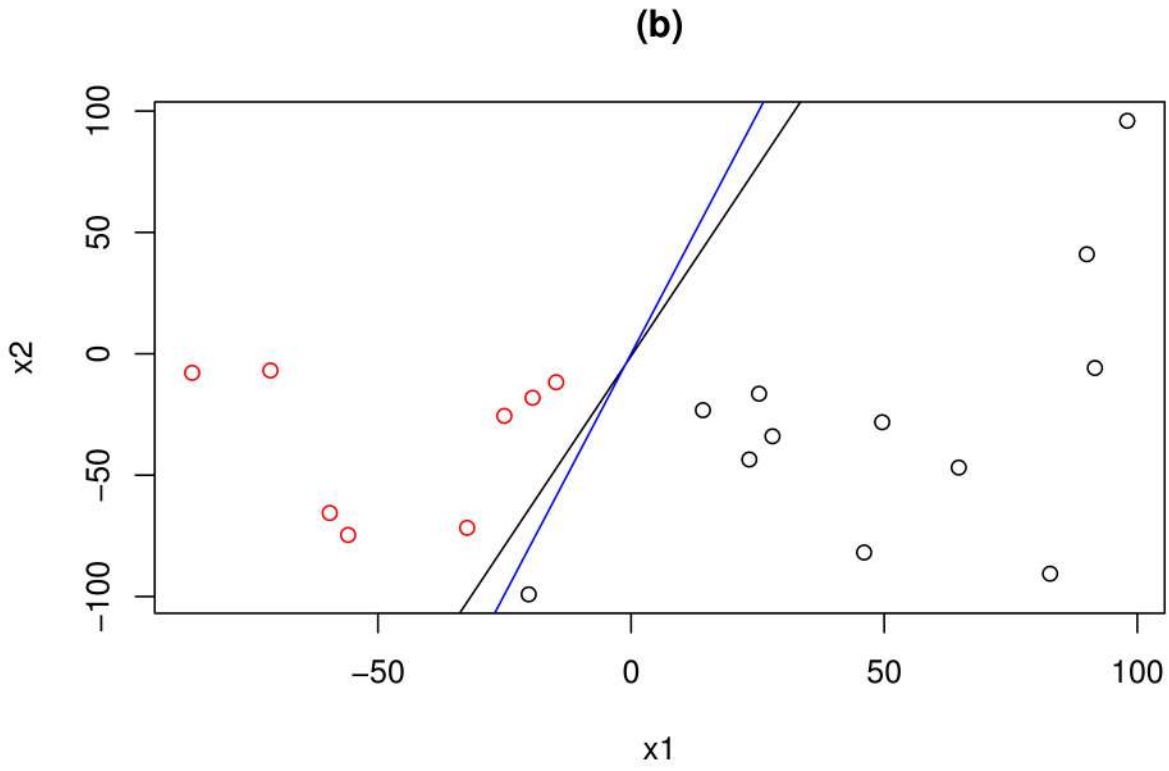


(b)

```
i = 0
w <- matrix(c(0,0,0), ncol = 3)
while (TRUE) {
  y_hat <- h(x, w)
  if (sum(y_hat != y) == 0) {
    break
  }
  mistakes = matrix(cbind(x, y, y_hat)[y_hat != y, ], ncol = 4)
  x_t = matrix(mistakes[1, 1:2], ncol = 2)
  w <- w + cbind(1, x_t) * mistakes[1, 3]
  i <- i + 1
}

print(i)
```

```
## [1] 10
```

```
plot(x, col = as.factor(y), xlab = "x1", ylab = "x2", main = "(b)")
abline(a= -w0/w2, b= -w1/w2)
abline(a= -w[1]/w[3], b= -w[2]/w[3], col = "blue")
```



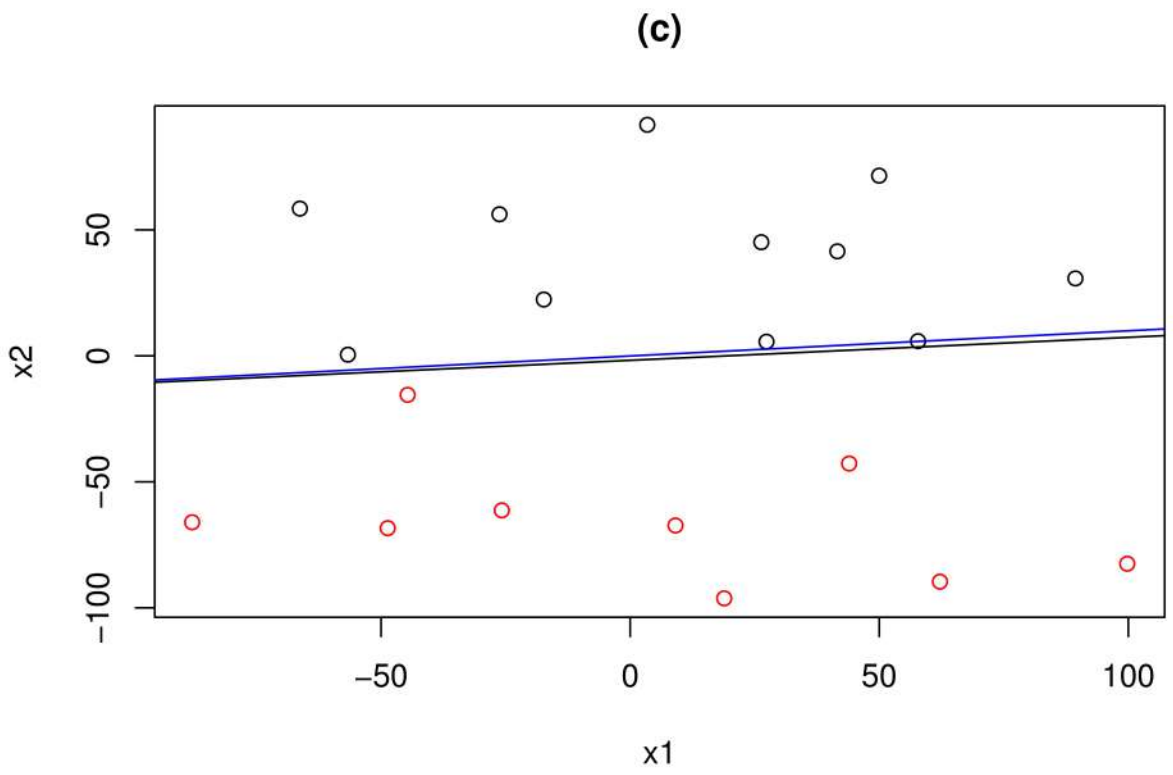
We can see here that the blue final hypothesis is very close to the black true function.

```
h <- function(x, w) {
  sign(cbind(1, x) %*% t(w))
}

w0 <- runif(1, -100, 100)
w1 <- runif(1, -100, 100)
w2 <- runif(1, -100, 100)

f <- function(x) {
  weights = matrix(c(w0, w1, w2), ncol = 3)
  h(x, weights)
}

x <- matrix(runif(40, -100, 100), ncol = 2)
y <- f(x)

i = 0
w <- matrix(c(0,0,0), ncol = 3)
while (TRUE) {
  y_hat <- h(x, w)
  if (sum(y_hat != y) == 0) {
    break
  }
  mistakes = matrix(cbind(x, y, y_hat)[y_hat != y, ], ncol = 4)
  x_t = matrix(mistakes[1, 1:2], ncol = 2)
```

```
  w <- w + cbind(1, x_t) * mistakes[1, 3]
  i <- i + 1
}

print(i)
```

```
## [1] 10
```

```
plot(x, col = as.factor(y), xlab = "x1", ylab = "x2", main = "(c)")
abline(a= -w0/w2, b= -w1/w2)
abline(a= -w[1]/w[3], b= -w[2]/w[3], col = "blue")
```



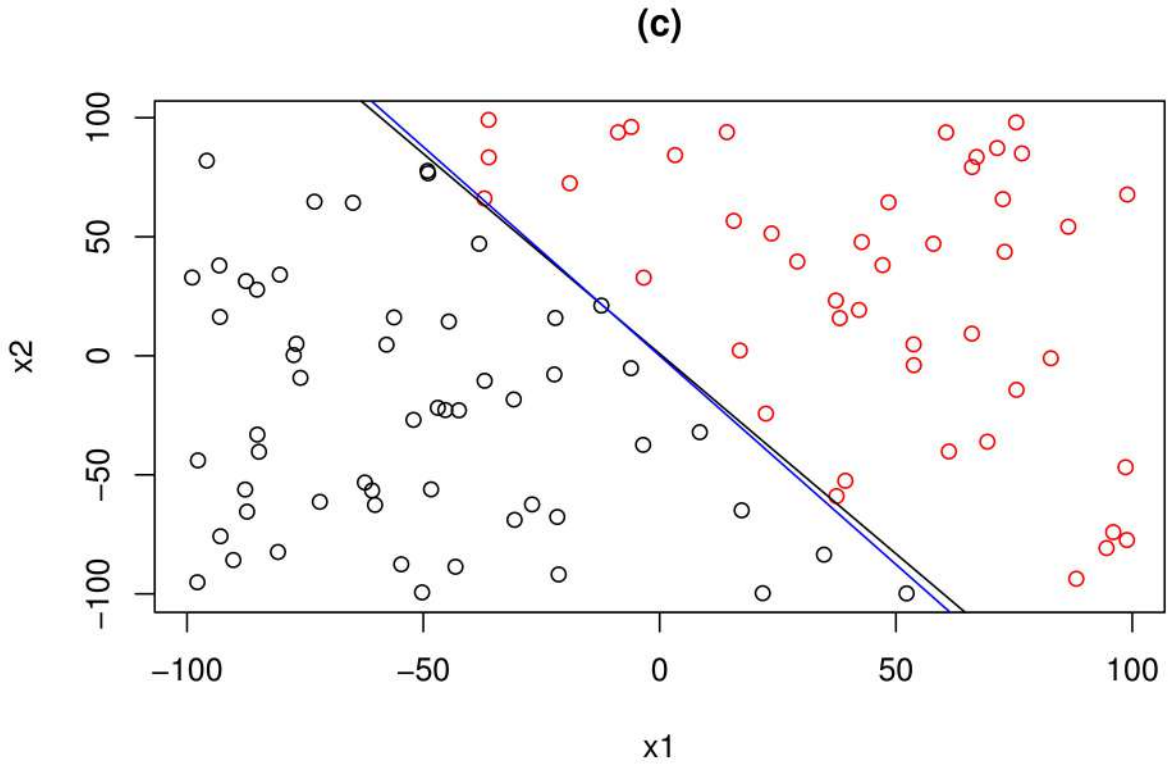
Again we see that the final hypothesis converges close to the true function. The number of iterations at 14 is slightly greater than 5 however.

```
h <- function(x, w) {
  sign(cbind(1, x) %*% t(w))
}

w0 <- runif(1, -100, 100)
w1 <- runif(1, -100, 100)
w2 <- runif(1, -100, 100)

f <- function(x) {
  weights = matrix(c(w0, w1, w2), ncol = 3)
  h(x, weights)
}
```

```
x <- matrix(runif(200, -100, 100), ncol = 2)
y <- f(x)

i = 0
w <- matrix(c(0,0,0), ncol = 3)
while (TRUE) {
  y_hat <- h(x, w)
  if (sum(y_hat != y) == 0) {
    break
  }
  mistakes = matrix(cbind(x, y, y_hat)[y_hat != y, ], ncol = 4)
  x_t = matrix(mistakes[1, 1:2], ncol = 2)
  w <- w + cbind(1, x_t) * mistakes[1, 3]
  i <- i + 1
}

print(i)
```

```
## [1] 4
```

```
plot(x, col = as.factor(y), xlab = "x1", ylab = "x2", main = "(c)")
abline(a= -w0/w2, b= -w1/w2)
abline(a= -w[1]/w[3], b= -w[2]/w[3], col = "blue")
```

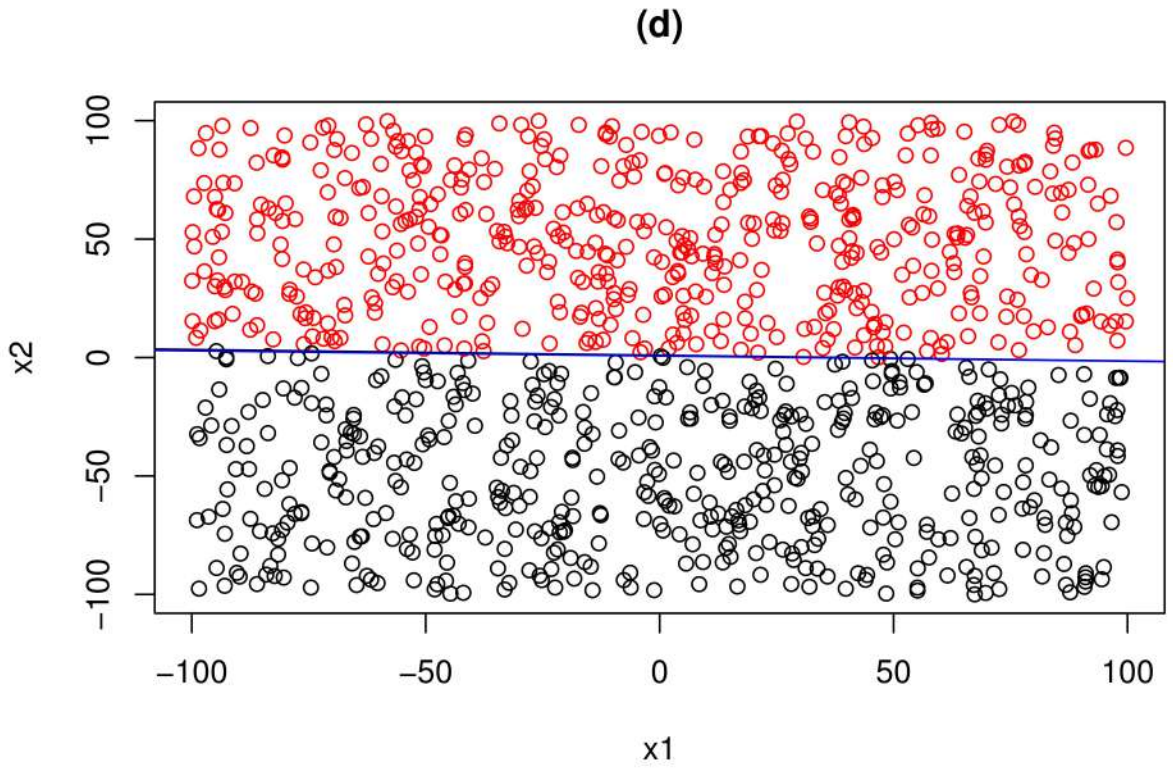


We see that the convergence is much slower for 100 data points, taking up 7463 iterations. The final hypothesis still matches closely to the true function though.

```r
h <- function(x, w) {
  sign(cbind(1, x) %*% t(w))
}

w0 <- runif(1, -100, 100)
w1 <- runif(1, -100, 100)
w2 <- runif(1, -100, 100)

f <- function(x) {
  weights = matrix(c(w0, w1, w2), ncol = 3)
  h(x, weights)
}

x <- matrix(runif(2000, -100, 100), ncol = 2)
y <- f(x)

i = 0
w <- matrix(c(0,0,0), ncol = 3)
while (TRUE) {
  y_hat <- h(x, w)
  if (sum(y_hat != y) == 0) {
    break
  }
  mistakes = matrix(cbind(x, y, y_hat)[y_hat != y, ], ncol = 4)
  x_t = matrix(mistakes[1, 1:2], ncol = 2)
  w <- w + cbind(1, x_t) * mistakes[1, 3]
  i <- i + 1
}

print(i)
```

```
## [1] 4866
```

```r
plot(x, col = as.factor(y), xlab = "x1", ylab = "x2", main = "(d)")
abline(a= -w0/w2, b= -w1/w2)
abline(a= -w[1]/w[3], b= -w[2]/w[3], col = "blue")
```



We see that the convergence is much slower for 1000 data points, taking up 2372 iterations. The final hypothesis still matches closely to the true function though.