

B529: Homework 1

Nathan Byers

Tuesday, February 03, 2015

Question 1

Please use the perceptron algorithm to manually find a weight vector $\mathbf{w} = [w_0, w_1, w_2]^T$ for linear classification of 3 data points (\mathbf{x}_i, y_i) satisfying $\text{sign}(\mathbf{w}^T \mathbf{x}_i) = y_i$. The input data is

$$\mathbf{x}_1 = (1, 2, 2)^T \quad y_1 = +1$$

$$\mathbf{x}_2 = (1, 1, 2)^T \quad y_2 = +1$$

$$\mathbf{x}_3 = (1, 2, 0)^T \quad y_3 = -1$$

The first element in vector \mathbf{x} is always $x_0 = 1$ (See slides in lecture 1).

In the first step of the perceptron algorithm, the initial assignment of \mathbf{w} is $[0, 6, 6]^T$. Please provide the intermediate steps of the computation. (30 points)

Answer 1

- Step 1: $\mathbf{w} = [0, 6, 6]^T$
- Step 2:

$$\text{sign}(\mathbf{w}^T \mathbf{x}_1) = \text{sign} \left(\begin{bmatrix} 0 & 6 & 6 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 2 \\ 2 \end{bmatrix} \right) = \text{sign}(24) = 1$$

$$\text{sign}(\mathbf{w}^T \mathbf{x}_2) = \text{sign} \left(\begin{bmatrix} 0 & 6 & 6 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix} \right) = \text{sign}(18) = 1$$

$$\text{sign}(\mathbf{w}^T \mathbf{x}_3) = \text{sign} \left(\begin{bmatrix} 0 & 6 & 6 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix} \right) = \text{sign}(12) \neq -1$$

$$\mathbf{w} = \mathbf{w} + y_3 \mathbf{x}_3 = \begin{bmatrix} 0 \\ 6 \\ 6 \end{bmatrix} + -1 \cdot \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix} = \begin{bmatrix} -1 \\ 4 \\ 6 \end{bmatrix}$$

- Step 2: iteration

$$\text{sign}(\mathbf{w}^T \mathbf{x}_1) = \text{sign} \left(\begin{bmatrix} -1 & 4 & 6 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 2 \\ 2 \end{bmatrix} \right) = \text{sign}(19) = 1$$

$$\text{sign}(\mathbf{w}^T \mathbf{x}_2) = \text{sign} \left(\begin{bmatrix} -1 & 4 & 6 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix} \right) = \text{sign}(15) = 1$$

$$\text{sign}(\mathbf{w}^T \mathbf{x}_1) = \text{sign} \left(\begin{bmatrix} -1 & 4 & 6 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix} \right) = \text{sign}(7) \neq -1$$

$$\mathbf{w} = \mathbf{w} + y_3 \mathbf{x}_3 = \begin{bmatrix} -1 \\ 4 \\ 6 \end{bmatrix} + -1 \cdot \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix} = \begin{bmatrix} -2 \\ 2 \\ 6 \end{bmatrix}$$

- Step 2: iteration

$$\text{sign}(\mathbf{w}^T \mathbf{x}_1) = \text{sign} \left(\begin{bmatrix} -2 & 2 & 6 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 2 \\ 2 \end{bmatrix} \right) = \text{sign}(14) = 1$$

$$\text{sign}(\mathbf{w}^T \mathbf{x}_1) = \text{sign} \left(\begin{bmatrix} -2 & 2 & 6 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix} \right) = \text{sign}(12) = 1$$

$$\text{sign}(\mathbf{w}^T \mathbf{x}_1) = \text{sign} \left(\begin{bmatrix} -2 & 2 & 6 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix} \right) = \text{sign}(2) \neq -1$$

$$\mathbf{w} = \mathbf{w} + y_3 \mathbf{x}_3 = \begin{bmatrix} -2 \\ 2 \\ 6 \end{bmatrix} + -1 \cdot \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix} = \begin{bmatrix} -4 \\ 0 \\ 6 \end{bmatrix}$$

- Step 2: iteration

$$\text{sign}(\mathbf{w}^T \mathbf{x}_1) = \text{sign} \left(\begin{bmatrix} -4 & 0 & 6 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 2 \\ 2 \end{bmatrix} \right) = \text{sign}(8) = 1$$

$$\text{sign}(\mathbf{w}^T \mathbf{x}_1) = \text{sign} \left(\begin{bmatrix} -4 & 0 & 6 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix} \right) = \text{sign}(8) = 1$$

$$\text{sign}(\mathbf{w}^T \mathbf{x}_1) = \text{sign} \left(\begin{bmatrix} -4 & 0 & 6 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix} \right) = \text{sign}(-4) = -1$$

Final $\mathbf{w} = (-4, 0, 6)^T$

Question 2

Please use R to manually compute the weight vector $\mathbf{w} = [w_0, w_1, w_2]^T$ for linear regression of N input data points (x_i, y_i) that minimizes the sum of squared error $(\mathbf{w}^T \mathbf{x}_i - y_i)^2$. The input data is

$$\mathbf{x}_1 = (1, 2, 3)^T \quad y_1 = 2$$

$$\mathbf{x}_2 = (1, 1, 2)^T \quad y_2 = 1$$

$$\mathbf{x}_3 = (1, 1, 0)^T \quad y_3 = -1$$

$$\mathbf{x}_4 = (1, -1, -2)^T \quad y_4 = -2$$

The first element in vector \mathbf{x} is always $x_0 = 1$ (See slides in lecture 1)

Please provide the intermediate steps of the computation (20 points).

Answer 2

```
X <- matrix(c(1, 2, 3, 1, 1, 2, 1, 1, 0, 1, -1, -2), nrow = 4, byrow = TRUE)
X
```

```
##      [,1] [,2] [,3]
## [1,]    1    2    3
## [2,]    1    1    2
## [3,]    1    1    0
## [4,]    1   -1   -2
```

```
y <- matrix(c(2, 1, -1, -2), ncol = 1)
y
```

```
##      [,1]
## [1,]    2
## [2,]    1
## [3,]   -1
## [4,]   -2
```

```
Xt <- t(X)
Xt
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    1    1    1    1
## [2,]    2    1    1   -1
## [3,]    3    2    0   -2
```

```
w <- solve(Xt %*% X) %*% Xt %*% y
w
```

```
##      [,1]
## [1,] -0.45
## [2,] -0.45
## [3,]  1.05
```

Question 3

In Fisher's linear discriminant analysis, we search for a vector \mathbf{w} such that all data points are well separated after they are projected to the direction defined by \mathbf{w} . When the input data points are

$$\mathbf{x}_1 = (3, 4)^T \quad y_1 = 1$$

$$\mathbf{x}_2 = (1, 2)^T \quad y_2 = 1$$

$$\mathbf{x}_3 = (1, 0)^T \quad y_3 = -1$$

$$\mathbf{x}_4 = (1, -2)^T \quad y_4 = -1$$

(Note that the first element in vector \mathbf{x} is NOT $x_0=1$ in this problem), please use R to manually find a vector $\mathbf{w}=[w_1, w_2]$ with Fisher's LDA and provide the intermediate steps (20 points).

Answer 3

Here I create a matrix with each column containing an \mathbf{x} vector as well as the y scalar.

```
Xy <- matrix(c(3, 4, 1, 1, 2, 1, 1, 0, -1, 1, -2, -1), ncol = 4)
colnames(Xy) <- c("x1hat", "x2hat", "x3hat", "x4hat")
rownames(Xy) <- c("x1", "x2", "y")
Xy
```

```
##      x1hat x2hat x3hat x4hat
## x1      3      1      1      1
## x2      4      2      0     -2
## y       1      1     -1     -1
```

Now I calculate the \mathbf{m} vectors.

```
m1hat <- matrix(apply(Xy[, Xy[3, ] == 1], 1, mean)[-3], ncol = 1)
rownames(m1hat) <- c("m1", "m2")
m1hat
```

```
##      [,1]
## m1      2
## m2      3
```

```
m2hat <- matrix(apply(Xy[, Xy[3, ] == -1], 1, mean)[-3], ncol = 1)
rownames(m2hat) <- c("m1", "m2")
m2hat
```

```
##      [,1]
## m1      1
## m2     -1
```

Then I create a function that takes the difference between an \mathbf{x} vector and the appropriate \mathbf{m} vector, and calculates the outer product between that vector and its transpose.

```
outerProduct <- function(i, Xy.matrix, m1, m2){
  if(Xy.matrix[3, i] == 1){m <- m1} else {m <- m2}
  v <- Xy.matrix[1:2, i] - m
  v %*% t(v)
}
```

Now I loop through the columns of the \mathbf{Xy} matrix using the `outerProduct()` function, and I get a matrix whose rows can be summed to find the four values of the \mathbf{Sw} matrix.

```
products <- sapply(1:dim(Xy)[2], outerProduct, Xy.matrix = Xy,
                  m1 = m1hat, m2 = m2hat)
products
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    1    1    0    0
## [2,]    1    1    0    0
## [3,]    1    1    0    0
## [4,]    1    1    1    1
```

```
Sw <- matrix(rowSums(products), ncol = 2)
Sw
```

```
##      [,1] [,2]
## [1,]    2    2
## [2,]    2    4
```

Now I find the optimal solution, $\mathbf{w} \propto \mathbf{S}_w^{-1}(\mathbf{m}_1 - \mathbf{m}_2)$.

```
w <- solve(Sw) %*% (m1hat - m2hat)
w
```

```
##      [,1]
## [1,] -1.0
## [2,]  1.5
```

Question 4

In the problem of determining if a peptide-spectrum-match is a correct one or not, we have the following problem:

- *Input: scores of peptide spectrum matches: $\mathbf{x}_i = [x]$*
- *Output: if the spectrum is generated from the peptide: $\mathbf{y}_i = +1 / -1$*
- *Prior probity: $P(y = 1) = 0.1, P(y = -1) = 0.9$*
- *Likelihood:*

$$P(x|y = 1) = \frac{1}{2\sqrt{2\pi}} e^{-\frac{(x-10)^2}{8}}$$

$$P(x|y = -1) = \frac{1}{\sqrt{2\pi}} e^{-\frac{(x-5)^2}{2}}$$

- *Cost function:*

$e(f(\mathbf{x}), g(\mathbf{x}))$	+1	-1
+1	0	1000
-1	1	0

Please compute the classification function that minimizes the cost (error). Please provide the intermediate steps of the computation. (30 points)

Answer 4

$$\ln \lambda = \ln \frac{P(y = -1)}{P(y = 1)} \frac{c_{1,-1}}{c_{-1,1}} = \ln \frac{0.9}{0.1} \frac{1000}{1} = \ln(9.10498)$$

$$\ln l(x) = \ln \frac{P(\mathbf{x}|y = -1)}{P(\mathbf{x}|y = 1)} = \ln \frac{1}{2\sqrt{2\pi}} e^{-\frac{(x-10)^2}{8}} - \ln \frac{1}{\sqrt{2\pi}} e^{-\frac{(x-5)^2}{2}}$$