CS M146 - Week 4

Xinzhu Bei

xzbei@cs.ucla.edu

February 3, 2018

Overview

- Exercises in class
- HW2 Q4
- Numpy tutorial
- HW2 Q5

Exercise in class

- Let $\sigma(z) = \frac{1}{1+\exp(-z)}$, show $\sigma(-z) = 1 - \sigma(z)$
- What is the gradient (respect to $\boldsymbol{\theta}$) of

$$-\sum_{i=1}^{m} y_i \log \sigma(\boldsymbol{\theta}^T x) + (1 - y_i) \log(1 - \sigma(\boldsymbol{\theta}^T x)) \tag{1}$$

  - we have $\sigma'(z) = \sigma(z)(1 - \sigma(z))$ (proved in class)
  - $$\frac{\partial h_\theta(\mathbf{x})}{\partial \theta_k} = \frac{\partial \sigma(\boldsymbol{\theta}^T \mathbf{x})}{\partial \theta_k} = \frac{\partial \sigma(\boldsymbol{\theta}^T \mathbf{x})}{\partial(\boldsymbol{\theta}^T \mathbf{x})} \frac{\partial(\boldsymbol{\theta}^T x)}{\partial \theta_k}$$

Exercise in Class

LMS regression can be solved analytically. Given a dataset $D = (x_1, y_1), (x_2, y_2), \cdots, (x_m, y_m)$, define matrix X and vector Y as follows:

$$X = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_m]_{d\times m}, Y = \begin{bmatrix} y_1 \\ y_2 \\ \cdots \\ y_m \end{bmatrix}_{m\times 1} \tag{2}$$

Show that the optimization problem

$$J(\mathbf{w}) = \frac{1}{2}\sum_{i=1}^{m}(y_i - \mathbf{w}^T\mathbf{x}_i)^2$$

is equivalent to

$$\min_{\mathbf{w}}(X^T\mathbf{w} - \mathbf{y})^T(X^T\mathbf{w} - \mathbf{y})$$

This can be solved analytically. Show that the solution w* is

$$\mathbf{w}^* = (XX^T)^{-1}X\mathbf{y}$$

Exercise in Class

$$
\begin{aligned}
f(\mathbf{w}) &= (X^T\mathbf{w} - \mathbf{y})^T(X^T\mathbf{w} - \mathbf{y}) \\
&= [(X^T\mathbf{w})^T - \mathbf{y}^T](X^T\mathbf{w} - \mathbf{y}) \\
&= [\mathbf{w}^T X - \mathbf{y}^T](X^T\mathbf{w} - \mathbf{y}) \\
&= \mathbf{w}^T X X^T \mathbf{w} - \mathbf{y}^T X^T \mathbf{w} - \mathbf{w}^T X \mathbf{y} + \mathbf{y}^T Y \\
&\overset{1}{=} \mathbf{w}^T X X^T \mathbf{w} - 2\mathbf{y}^T X^T \mathbf{w} + \mathbf{y}^T \mathbf{y}
\end{aligned} \tag{3}
$$

1 holds because $([\mathbf{y}^T X^T \mathbf{w}]_{1\times 1})^T = [\mathbf{y}^T X^T \mathbf{w}]_{1\times 1} = \mathbf{w}^T X \mathbf{y}$.

$$
\begin{gathered}
\nabla_{\mathbf{w}} f(\mathbf{w}) = 2XX^T\mathbf{w} - 2X\mathbf{y} = 0 \\
\mathbf{w} = (XX^T)^{-1} X\mathbf{y}
\end{gathered} \tag{4}
$$

This is slightly different from the formula in homework. You can match the formula in homework by taking $X' = X^T$.

HW2 Q4

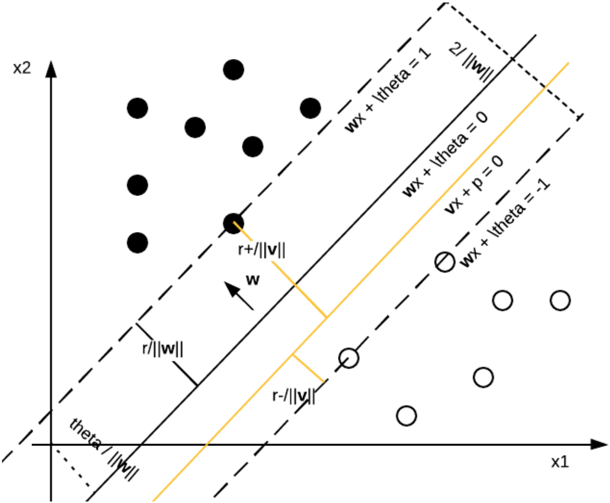
- Linearly Separable $\Rightarrow \exists$ a linear function with parameter $(\mathbf{w}, \theta)$,

$$\forall(\mathbf{x}_i, y_i) \in D, y_i = \begin{cases} 1 \text{ if } \mathbf{w}^T\mathbf{x}_i + \theta \geq 0 \\ -1 \text{ if } \mathbf{w}^T\mathbf{x}_i + \theta < 0 \end{cases} \tag{5}$$

- The following linear program is used to "find" the linear separator

$$\begin{aligned} \min_{(\mathbf{w},\theta),\delta} \quad & \delta \\ \text{s.t.} \quad & y_i(\mathbf{w}^T\mathbf{x}_i + \theta) \geq 1 - \delta, \\ & \delta \geq 0, \forall(\mathbf{x}_i, y_i) \in D \end{aligned} \tag{6}$$

HW2 Q4



HW2 Q4

Linearly Separable
$\Rightarrow \exists$ a linear function with parameter $(\mathbf{v}, \rho)$,

$$\forall(\mathbf{x}_i, y_i) \in D, y_i = \begin{cases} 1 \text{ if } \mathbf{v}^T\mathbf{x}_i + \rho \geq 0 \\ -1 \text{ if } \mathbf{v}^T\mathbf{x}_i + \rho < 0 \end{cases} \tag{7}$$

$\Rightarrow \exists$ a linear function with parameter $(\mathbf{v}, \rho)$,

$$\begin{gathered} \min_{(x,y)\in D, y=1} (\mathbf{v}^T x + p) \geq 0 > \max_{(x,y)\in D, y=-1} (\mathbf{v}^T x + p) \\ r^+ \geq 0 > r^- \\ r^+/||\mathbf{v}|| \geq 0 > r^-/||\mathbf{v}|| \end{gathered} \tag{8}$$

HW2 Q4

$$\begin{aligned} \min_{(\mathbf{w},\theta),\delta} \quad & \delta \\ \text{s.t.} \quad & y_i(\mathbf{w}^T\mathbf{x}_i + \theta) \geq 1 - \delta, \\ & \delta \geq 0, \forall(\mathbf{x}_i, y_i) \in D \end{aligned} \tag{9}$$

Observations

- The distance of a point x to a hyperplane $\mathbf{w}^T x + \theta$ is

$$r_x = (\mathbf{w}^T x + \theta)/||\mathbf{w}||$$

- $y_i(\mathbf{w}^T\mathbf{x}_i + \theta) = |r_x|/||\mathbf{w}||$
- If $\delta^* = 0$, then there must exist a hyperplane $(\mathbf{w}^T, \theta)$, such that $y_i(\mathbf{w}^T\mathbf{x}_i + \theta) \geq 1 - 0 = 1$.
- If there exist a hyperplane $(\mathbf{w}^T, \theta)$, such that $y_i(\mathbf{w}^T\mathbf{x}_i + \theta) \geq 1$, then $\delta^* = 0$.
- Even the linear program reach the minimum value $\delta^* = 0$, the optimal hyperplane $(\mathbf{w}^*, \theta^*)$ is not unique.

HW2 Q4.a - Entire Logic

1) Linearly Separable $\Rightarrow \exists$ a linear function with parameter $(\mathbf{v}, \rho)$ that satisfies condition (1).
2) Using $(\mathbf{v}, \rho)$ to show that there is $(\mathbf{w}, \theta)$ that satisfies

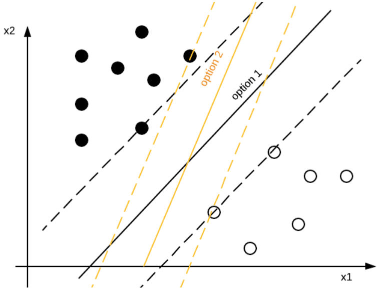
$$y_i(\mathbf{w}^T x_i + \theta) \geq 1$$

3) Trivial to show that

$$y_i(\mathbf{w}^T x_i + \theta) \geq 1 \Leftrightarrow (\mathbf{w}, \theta, \delta^* = 0) \text{ optimized condition 2}$$

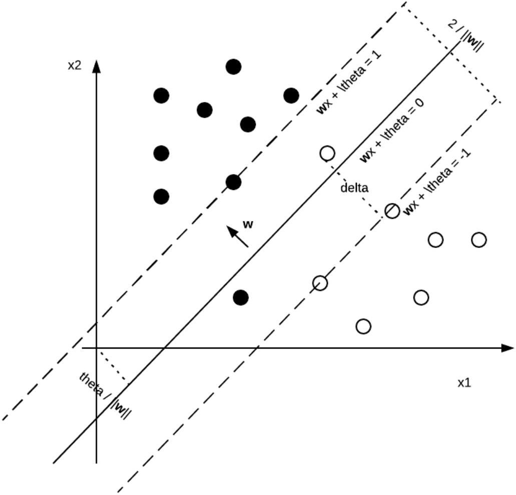
HW2 Q4 - Some Additional Interpretation

Statement: Even the linear program reach the minimum value $\delta^* = 0$, the optimal hyperplane $(\mathbf{w}^*, \theta^*)$ is not unique. (You can prove question a by simply finding one of them.)
Why? Because in many cases, there are more than one linear functions that can "perfectly" separate the data. And for each linear function $(\mathbf{v}_k, \rho_k)$, we can find a corresponding $(\mathbf{w}_k, \theta_k)$ that satisfies condition (2).



HW2 Q4 - linearly inseparable case

$$
\begin{aligned}
\min_{(\mathbf{w},\theta),\delta} \quad & \delta \\
\text{s.t.} \quad & y_i(\mathbf{w}^T \mathbf{x}_i + \theta) \geq 1 - \delta, \\
& \delta \geq 0, \\
& \forall (\mathbf{x}_i, y_i) \in D
\end{aligned} \tag{10}
$$



- ```
  import numpy
  import numpy as np
  from numpy import *
  ```
- NumPy's arrays are more compact than Python lists.
  ```
  b = np.array([6, 7, 8])
  b_ = [6,7,8]
  ```
- Attributes: shape, size, type, etc.
  ```
  >>> np.array([1, 2, 3]).shape
  (3,)
  >>> np.array([[1, 2, 3]]).shape
  (1, 3)
  >>> np.array([[1], [2], [3]]).shape
  (3, 1)
  ```
  ```

- ```
  >>> np.zeros( (3,4) )
  array([[ 0.,  0.,  0.,  0.],
         [ 0.,  0.,  0.,  0.],
         [ 0.,  0.,  0.,  0.]])
  ```
- ```
  >>> a = np.array([[3,1,5],
          [1,0,8],[2,1,4]])
  >>> for i in a:
  ...     print i
  [3 1 5]
  [1 0 8]
  [2 1 4]
  ```
- ```
  >>> ones_ = np.ones((3,1))
  >>> np.c_[ones_,a]
  array([[ 1.,  3.,  1.,  5.],
         [ 1.,  1.,  0.,  8.],
         [ 1.,  2.,  1.,  4.]])
  >>> np.column_stack((ones_,a))
  ```

```
>>> A = np.array([[1,1],[0,1]])
>>> B = np.array([[2,0],[3,4]])
>>> A+B
array([[3, 1],
       [3, 5]])
>>> A-B
array([[-1,  1],
       [-3, -3]])
>>> B**2
array([[ 4,  0],
       [ 9, 16]])
>>> A*B # elementwise product
array([[2, 0],
       [0, 4]])
>>> np.dot(A,B) # matrix product
array([[5, 4],
       [3, 4]])
>>> 2*A
array([[2, 2],
       [0, 2]])
```
```

HW2 Q5

Given N training instances, it is always possible to obtain a perfect fit (a fit in which all the data points are exactly predicted) by setting the degree of the regression to $N-1$.
A polynomial of degree n is of the form $p_{n-1}(x) = a_{n-1}x^{n-1} + \; + a_1 x + a_0$. To study the existence and uniqueness of such a polynomial consider the system of linear equations:

$$\begin{cases} a_{n-1}x_1^{n-1} + \; + a_1 x_1 + a_0 = y_1 \\ \qquad\qquad\qquad\qquad\qquad \cdots \\ a_{n-1}x_n^{n-1} + \; + a_1 x_n + a_0 = y_n \end{cases} \tag{11}$$

We write the system as

$$\begin{pmatrix} x_1^{n-1} & x_1^{n-2} & \cdots & x_1 & 1 \\ & & \cdots & & \\ x_n^{n-1} & x_n^{n-2} & \cdots & x_n & 1 \end{pmatrix} \begin{pmatrix} a_{n-1} \\ \cdots \\ a_0 \end{pmatrix} = \begin{pmatrix} y_1 \\ \cdots \\ y_n \end{pmatrix} \tag{12}$$

There are n unknowns with n equations. This results in an unique answer.

The End