Homework #2

Wednesday, 16 October 2024 19:28

1.

a/

```python
def perceptron_update(weights, bias, x, y):
    prediction = sum(w * x_i for w, x_i in zip(weights, x)) + bias

    if y * prediction <= 0:
        weights = [w + y * x_i for w, x_i in zip(weights, x)]
        bias += y

    return weights, bias

weights = [0, -1]
bias = 0.5
samples = [
    ([0, 0], -1),
    ([0, 1], 1),
    ([1, 0], 1),
    ([1, 1], 1)
]

converged = False
epoch = 0
while not converged:
    converged = True
    epoch += 1
    for x, y in samples:
        old_weights, old_bias = weights[:], bias
        weights, bias = perceptron_update(weights, bias, x, y)
        if weights != old_weights or bias != old_bias:
            converged = False
    print(f"Epoch {epoch}: weights = {weights}, bias = {bias}")

print(f"Final weights = {weights}, final bias = {bias}")
```

```
Epoch 1: weights = [0, 0], bias = 0.5
Epoch 2: weights = [0, 1], bias = 0.5
Epoch 3: weights = [1, 1], bias = 0.5
Epoch 4: weights = [1, 1], bias = -0.5
Epoch 5: weights = [1, 1], bias = -0.5
Final weights = [1, 1], final bias = -0.5
PS D:\Code\HWTasks\Machine Learning\Homework2>
```

b/ Data set in 1b is XOR problem.

(0,0) and (1,1) label -1

(0,1) and (1,0) label 1

$\Rightarrow$ No single line can seperate these 4 points correctly

So it cannot be solved by perceptron

2/

a/ $L(w) = \sum_{i=1}^{n} |w^T x_i - y_i|^2 = \|Xw - y\|^2$

$= (Xw - y)^T (Xw - y)$

$$= w^T X^T X w - 2y^T X w + y^T y$$

$$\Rightarrow \frac{\partial L(w)}{\partial w} = 2X^T X w - 2X^T y$$

b/

We have: $2X^T X w^* - 2X^T y = 0$

$$\Leftrightarrow \quad X^T X w^* = X^T y$$

Since X is full column rank $\Rightarrow X^T X$ is invertable

$$\Rightarrow \quad w^* = (X^T X)^{-1} X^T y$$

c/

We have: $\frac{\partial L_r(w)}{\partial w} = 2(Xw - y)\left(\frac{Xw - y}{w}\right)' + 2\lambda w$

$$= 2X^T X w - 2X^T y + 2\lambda w \quad = 0$$

$$\Leftrightarrow (X^T X + \lambda I) w^* = X^T y$$

$$\Leftrightarrow \quad w^* = (X^T X + \lambda I)^{-1} X^T y$$

d/

We have: $L_r = \|Xw - y\|^2 + \lambda \|w\|_2^2$

Let $\begin{cases} X' \text{ size } m \times m \text{ so that: } X' = \sqrt{\lambda} I \\ y' \text{ output size } m \times 1 \text{ with all element as } 0 \end{cases}$

$$\Rightarrow \quad X_{new} = \begin{bmatrix} X \\ X' \end{bmatrix}, \quad y_{new} = \begin{bmatrix} y \\ y' \end{bmatrix}$$

$$\Rightarrow \quad L_r = \|X_{new} w - y_{new}\|^2$$

So adding m artificial samples with features $\sqrt{\lambda}$ along the diagonal and labels as 0 replicates the effect of L2 regularization in ordinary least squares regression by penalizing large weight values.

regularization in ordinary least squares regression by penalizing large weight values.