COMP90051
Statistical Machine Learning

Workshop Week 5

Xudong Han

https://github.com/HanXudong/COMP90051_2020_S1

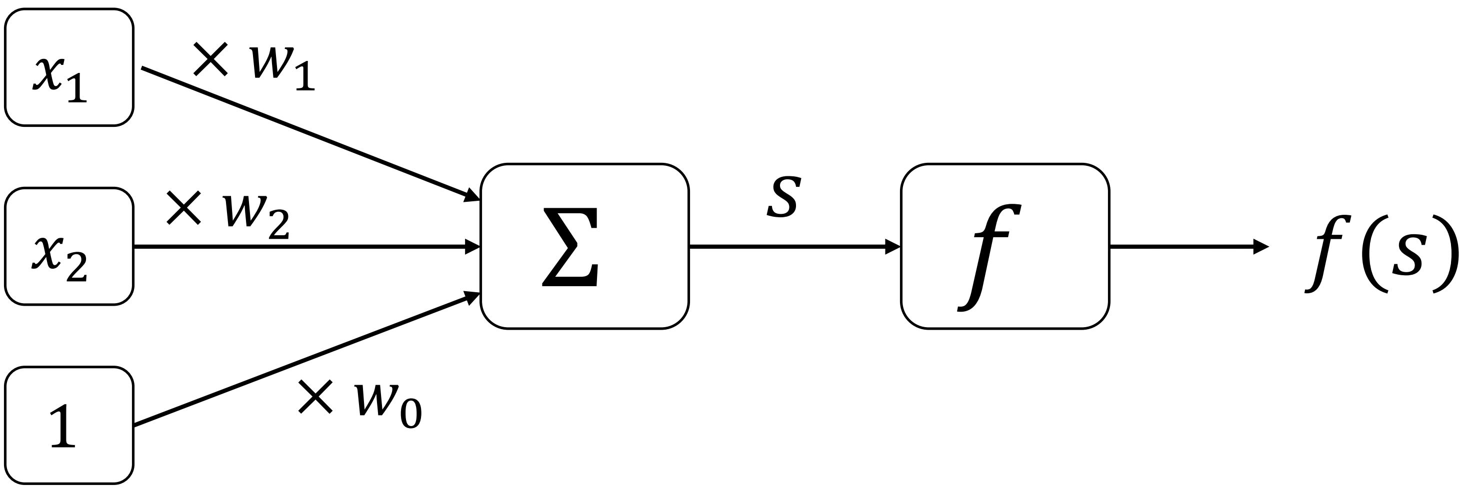
Review

- Logistic Regression =
  Linear Regression + Logistic function

- $\theta = p(y = 1|\vec{x}) = \sigma(\vec{w}^T\vec{X} \ + b)$

- Loss function: Cross-Entropy

Definition of the perceptron



- $f(s) = \begin{cases} 1 & \text{if } s \geq 0, \\ \text{-}1 & \text{otherwise} \end{cases}$

Perceptron training algorithm

$\textsc{Perceptron}(\mathbf{w}_0)$

$\mathbf{w}_1 \leftarrow \mathbf{w}_0 \qquad \triangleright$ typically $\mathbf{w}_0 = \mathbf{0}$

for $t \leftarrow 1$ **to** T **do**

    $\textsc{Receive}(\mathbf{x}_t)$

    $\widehat{y}_t \leftarrow \operatorname{sgn}(\mathbf{w}_t \cdot \mathbf{x}_t)$

    $\textsc{Receive}(y_t)$

    if $(\widehat{y}_t \neq y_t)$ **then**

        $\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t + y_t \mathbf{x}_t \qquad \triangleright$ more generally $\eta y_t \mathbf{x}_t, \eta > 0.$

    else $\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t$

return $\mathbf{w}_{T+1}$

Dataset

<x1,x2>	y
<1,1>	1
<1,2>	1
<0,0>	-1
<-1,0>	-1

w = <0,0,0>

Perceptron training algorithm

- The above training procedure is equivalent to performing sequential gradient descent on the following objective function:

$$F(\mathbf{w}) = \frac{1}{T}\sum_{t=1}^{T} \max\left(0, -y_t(\mathbf{w} \cdot \mathbf{x}_t)\right)$$

$$\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t - \eta \nabla_{\mathbf{w}} F(\mathbf{w})$$

Evaluation

- proportion of misclassified instances (error rate)

Convergence

- $\vec{w}^* \, \vec{w}_{k+1} \geq (k+1)\gamma$
- $\left||\vec{w}_k|\right|^2 \leq kR^2$
- $\vec{w}^* \vec{w}_k \leq |\vec{w}^*||\vec{w}_k|$
- $k \leq \frac{R^2}{\gamma^2}$