Deep Learning

Vijaya Saradhi

IIT Guwahati

Fri, 18^{th} Sept 2020

Perceptron

Preliminininaries

- Let the following hold for these classes:

$$\mathbf{w}^T\mathbf{x} > 0 \quad \forall\, \mathbf{x} \in \mathcal{C}_1$$
$$\mathbf{w}^T\mathbf{x} \leq 0 \quad \forall\, \mathbf{x} \in \mathcal{C}_2$$

- The case that if $\mathbf{w}^T\mathbf{x} = 0$ then $\mathbf{x} \in \mathcal{C}_2$

Perceptron

Update Rule 02

- If the following is violated then there is no change in $\mathbf{w}$

$$\begin{array}{lll} \mathbf{w}(n+1) = \mathbf{w}(n) - \eta(n)\mathbf{x}(n) & \text{if } \mathbf{w}^T(n)\mathbf{x}(n) > 0 & \mathbf{x}(n) \in \mathcal{C}_2 \\ \mathbf{w}(n+1) = \mathbf{w}(n) + \eta(n)\mathbf{x}(n) & \text{if } \mathbf{w}^T(n)\mathbf{x}(n) \leq 0 & \mathbf{x}(n) \in \mathcal{C}_1 \end{array}$$

- The case that if $\mathbf{w}^T\mathbf{x} = 0$ then $\mathbf{x} \in \mathcal{C}_2$

Perceptron

Initialization

- Let $\mathbf{w} = \mathbf{0}$
- Let $\eta(n) = 1$

Assumption

- Suppose $\mathbf{w}^T(n)\mathbf{x}(n) < 0$ for $n = 1, 2, \cdots$
- $\mathbf{x}(n) \in \mathcal{C}_1$ for $n = 1, 2, \cdots$
- Classes $\mathcal{C}_1$ and $\mathcal{C}_2$ are linearly separable
- Update $\mathbf{w}(n+1)$

$$\begin{array}{lll} \mathbf{w}(n+1) = \mathbf{w}(n) + \eta(n)\mathbf{x}(n) & \text{if } \mathbf{w}^T(n)\mathbf{x}(n) \leq 0 & \mathbf{x}(n) \in \mathcal{C}_1 \\ \mathbf{w}(n+1) = \mathbf{0} + \mathbf{x}(n) & \text{if } \mathbf{w}^T(n)\mathbf{x}(n) \leq 0 & \mathbf{x}(n) \in \mathcal{C}_1 \end{array}$$

Applying Perceptron Update Rule

Iterate

- $\mathbf{x}(1) \in \mathcal{C}_1$
- $\mathbf{w}^T(1)\mathbf{x}(1) = 0$
- Update Rule is: $\mathbf{w}(2) = \mathbf{w}(1) + \mathbf{x}(1) = \mathbf{x}(1)$

Applying Perceptron Update Rule

Iterate

- $\mathbf{x}(2) \in \mathcal{C}_1$
- $\mathbf{w}^T(2)\mathbf{x}(2) \leq 0$
- Update Rule is: $\mathbf{w}(3) = \mathbf{w}(2) + \mathbf{x}(2)$
- That is $\mathbf{w}(3) = \mathbf{x}(1) + \mathbf{x}(2)$

Applying Perceptron Update Rule

Iterate

- $\mathbf{x}(3) \in \mathcal{C}_1$
- $\mathbf{w}^T(3)\mathbf{x}(3) \leq 0$
- Update Rule is: $\mathbf{w}(4) = \mathbf{w}(3) + \mathbf{x}(3)$
- That is $\mathbf{w}(4) = \mathbf{x}(1) + \mathbf{x}(2) + \mathbf{x}(3)$

Applying Perceptron Update Rule

Iterate

- $\mathbf{x}(4) \in \mathcal{C}_1$
- $\mathbf{w}^T(4)\mathbf{x}(4) \leq 0$
- Update Rule is: $\mathbf{w}(5) = \mathbf{w}(4) + \mathbf{x}(4)$
- That is $\mathbf{w}(5) = \mathbf{x}(1) + \mathbf{x}(2) + \mathbf{x}(3) + \mathbf{x}(4)$

Applying Perceptron Update Rule

Iterate

- $\mathbf{x}(n) \in \mathcal{C}_1$
- $\mathbf{w}^T(n)\mathbf{x}(n) \leq 0$
- Update Rule is: $\mathbf{w}(n+1) = \mathbf{w}(n) + \mathbf{x}(n)$
- That is $\mathbf{w}(n+1) = \mathbf{x}(1) + \mathbf{x}(2) + \mathbf{x}(3) + \mathbf{x}(4) \cdots + \mathbf{x}(n)$

Linearly Separable Assumption

Make use of assumption

- As $\mathcal{C}_1$ and $\mathcal{C}_2$ are linearly separable
- There exists a $\mathbf{w}_o$
- For which $\mathbf{w}_o^T(n)\mathbf{x}(n) > 0$ for $\mathbf{x}(1), \mathbf{x}(2), \cdots \mathbf{x}(n) \in \mathcal{C}_1$

Linearly Separable Assumption

Compute the norm of $\mathbf{w}(n)$

- We want to understand what is the norm (length of vector) of $\mathbf{w}(n)$
- Why? Does $\mathbf{w}(n)$ keeps on added with $\mathbf{x}(n)$? Will the norm be unbounded?
- However, $\mathcal{C}_1$ and $\mathcal{C}_2$ are linearly separable

Linearly Separable Assumption

Minimum value of inner product

- Let the α be a quantity defined as:
  $\alpha = \min\left\{\mathbf{w}_o^T\mathbf{x}(1), \mathbf{w}_o^T\mathbf{x}(2), \mathbf{w}_o^T\mathbf{x}(3), \cdots \mathbf{w}_o^T\mathbf{x}(n)\right\}$
- That is

$$\alpha = \min_{\mathbf{x}(n)\in\mathcal{C}_1} \mathbf{w}_o^T(n)\mathbf{x}(n)$$

Linearly Separable Assumption

Update rule as per mis-classification

- $\mathbf{w}(n+1) = \mathbf{x}(1) + \mathbf{x}(2) + \mathbf{x}(3) + \mathbf{x}(4) \cdots + \mathbf{x}(n)$
- To compute norm of $\mathbf{w}(n)$
- Multiply both sides of this equation with $\mathbf{w}_o^T$
- $\mathbf{w}_o^T \mathbf{w}(n+1) = \mathbf{w}_o^T \mathbf{x}(1) + \mathbf{w}_o^T \mathbf{x}(2) + \mathbf{w}_o^T \mathbf{x}(3) + \mathbf{w}_o^T \mathbf{x}(4) \cdots + \mathbf{w}_o^T \mathbf{x}(n)$
- Replace every term $\mathbf{w}_o^T \mathbf{x}(n)$ with α
- $\mathbf{w}_o^T \mathbf{w}(n+1) \geq \alpha + \alpha + \alpha + \alpha + \cdots + \alpha$
- $\mathbf{w}_o^T \mathbf{w}(n+1) \geq n\alpha$

Linearly Separable Assumption

Key equation

$\mathbf{w}_o^T \mathbf{w}(n+1) \geq n\alpha$

Linearly Separable Assumption

Cauchy-Bunyakovsky-Schwarz inequality

- For all $\mathbf{u}$ and $\mathbf{v}$
- $|\mathbf{u}^T\mathbf{u}|.|\mathbf{v}^T\mathbf{v}\| \geq |\mathbf{u}^T\mathbf{v}|$

Apply Cauchy-Bunyakovsky-Schwarz inequality

- Given two vectors $\mathbf{w}_o$ and $\mathbf{w}(n+1)$
- $\|\mathbf{w}_o\|^2.\|\mathbf{w}(n+1)\|^2 \geq \left[\mathbf{w}_o^T\mathbf{w}(n+1)\right]^2$
- We have obtained $\mathbf{w}_o^T\mathbf{w}(n+1) \geq n\alpha$
- That is $\left[\mathbf{w}_o^T\mathbf{w}(n+1)\right]^2 \geq n^2\alpha^2$
- $\|\mathbf{w}_o\|^2.\|\mathbf{w}(n+1)\|^2 \geq \left[\mathbf{w}_o^T\mathbf{w}(n+1)\right]^2 \geq n^2\alpha^2$
- $\|\mathbf{w}_o\|^2.\|\mathbf{w}(n+1)\|^2 \geq n^2\alpha^2$

Linearly Separable Assumption

Norm of $\mathbf{w}(n+1)$

Norm of $\mathbf{w}(n+1)$ is: $\|\mathbf{w}(n+1)\|^2 \geq \frac{n^2\alpha^2}{\|\mathbf{w}_o\|^2}$

Alternate: Norm

Norm of $\mathbf{w}(n+1)$

$$\begin{aligned}
\mathbf{w}(k+1) &= \mathbf{w}(k) + \mathbf{x}(k) \text{ for } k = 1, 2, \cdots, n \\
\|\mathbf{w}(k+1)\|^2 &= (\mathbf{w}(k) + \mathbf{x}(k))^2 \\
\|\mathbf{w}(k+1)\|^2 &= \|\mathbf{w}(k)\|^2 + \|\mathbf{x}(k)\|^2 + 2\mathbf{w}^T(k)\mathbf{x}(k) \\
\|\mathbf{w}(k+1)\|^2 &\leq \|\mathbf{w}(k)\|^2 + \|\mathbf{x}(k)\|^2
\end{aligned}$$

Alternate: Norm

Norm of $\mathbf{w}(n+1)$

$$\|\mathbf{w}(k+1)\|^2 - \|\mathbf{w}(k)\|^2 \quad \leq \|\mathbf{x}(k)\|^2 \text{ for } k = 1, 2, \cdots, n$$

Alternate: Norm

Norm of $\mathbf{w}(n+1)$

$$
\begin{aligned}
\|\mathbf{w}(2)\|^2 - \|\mathbf{w}(1)\|^2 &\leq \|\mathbf{x}(1)\|^2 \\
&+ \\
\|\mathbf{w}(3)\|^2 - \|\mathbf{w}(2)\|^2 &\leq \|\mathbf{x}(2)\|^2 \\
&+ \\
\|\mathbf{w}(4)\|^2 - \|\mathbf{w}(3)\|^2 &\leq \|\mathbf{x}(3)\|^2 \\
&+ \\
\|\mathbf{w}(n)\|^2 - \|\mathbf{w}(n-1)\|^2 &\leq \|\mathbf{x}(n-1)\|^2 \\
&+ \\
&\vdots \\
&+ \\
\|\mathbf{w}(n+1)\|^2 - \|\mathbf{w}(n)\|^2 &\leq \|\mathbf{x}(n)\|^2
\end{aligned}
$$

Alternate: Norm

Norm of $\mathbf{w}(n+1)$

$$\begin{aligned}
\|\mathbf{w}(n+1)\|^2 &\leq \sum_{k=1}^{n} \|\mathbf{x}(k)\|^2 \\
\|\mathbf{w}(n+1)\|^2 &\leq n\beta \\
\text{where } \beta &= \max_{\mathbf{x}(k)\in\mathcal{C}_1} \|\mathbf{x}(k)\|^2
\end{aligned}$$

Alternate: Norm

Norm of $\mathbf{w}(n+1)$

$$
\begin{aligned}
\|\mathbf{w}(n+1)\|^2 &\geq \frac{n^2\alpha^2}{\|\mathbf{w}_o\|^2} \\
&\text{and} \\
\|\mathbf{w}(n+1)\|^2 &\leq n\beta \\
\frac{n_{max}^2\alpha^2}{\|\mathbf{w}_o\|^2} &= n_{max}\beta \\
n_{max} &= \frac{\beta\|\mathbf{w}_o\|^2}{\alpha^2}
\end{aligned}
$$

Norm of $\mathbf{w}(n+1)$

That is maximum number of iterations are bounded. Perceptron should converge.

Algorithm

Incremental

Initialization Set $\mathbf{w}(0) = \mathbf{0}$; Perform following computations for n = 1, 2, ...

Activation At time step n, provide the input vector $\mathbf{x}(n)$ and desired response $d(n)$

Response $sgn(\mathbf{w}^T(n)\mathbf{x}(n))$ Output is {-1, +1}

Adaptation $\mathbf{w}(n+1) = \mathbf{w}(n) + \eta\,[d(n) - y(n)]\,\mathbf{x}(n)$
Where

$$d(n) = \begin{cases} +1 & \text{if } \mathbf{x}(n) \in \mathcal{C}_1 \\ -1 & \text{if } \mathbf{x}(n) \in \mathcal{C}_2 \end{cases}$$

Iterate Increment n and go to activation step

Batch Algorithm

Objective (Cost) Function

- Compute: $\mathbf{w}^T(n)\mathbf{x}(n)$
- Treat the above quantity as the objective function
- With the modification $\mathbf{w}^T(n)\mathbf{x}(n)d(n)$
- For one $\mathbf{x}(n)$ the above objective function is used:
- For many $\mathbf{x}(n)$'s we have:

$$J(\mathbf{w}) = \sum_{\mathbf{x}(n)\in\mathcal{H}} \left(-(\mathbf{w}^T(n)\mathbf{x}(n)d(n))\right)$$

- The above objective function should be minimized

Batch Algorithm

Objective Function - Intuition

- We have to minimize or maximize a given objective function
- Percentron rule: $\mathbf{w}^T(n)\mathbf{x}(n) > 0$ $\mathbf{x}(n) \in \mathcal{C}_1$
- $\mathbf{w}^T(n)\mathbf{x}(n)$ quantity for $\mathcal{C}_1$ is positive, d(n) $= +1$. Decrease it by multiplying it -1
- Percentron rule: $\mathbf{w}^T(n)\mathbf{x}(n) \leq 0$ $\mathbf{x}(n) \in \mathcal{C}_1$
- $\mathbf{w}^T(n)\mathbf{x}(n)$ quantity for $\mathcal{C}_1$ is negative, d(n) $=$ -1. Decrease it by multiplying it -1
- That is for any $\mathbf{x}(n)$, the quantity $-(\mathbf{w}^T(n)\mathbf{x}(n)d(n))$ to be minimized

Batch Algorithm

Apply Gradient Descent Rule

- Compute direction: $\bigtriangledown J(\mathbf{w}) = \sum_{\mathbf{x}(n)\in\mathcal{H}} (-\mathbf{x}(n)d(n))$
- Update $\mathbf{w}(n+1) = \mathbf{w}(n) - \eta(n) \bigtriangledown J(\mathbf{w})$
- That is $\mathbf{w}(n+1) = \mathbf{w}(n) - \eta(n) \sum_{\mathbf{x}(n)\in\mathcal{H}} (-\mathbf{x}(n)d(n)))$