Widrow-Hoff (or LMS) Learning Algorithm

The gradient of the MSE cost function is:

$$\nabla J_s(\boldsymbol{a}) = \boldsymbol{Y}^t(\boldsymbol{Ya} - \boldsymbol{b})$$

Hence, the batch update rule is:

$$\boldsymbol{a} \leftarrow \boldsymbol{a} - \eta \boldsymbol{Y}^t(\boldsymbol{Ya} - \boldsymbol{b})$$

The sequential update rule is:

$$\boldsymbol{a} \leftarrow \boldsymbol{a} - \eta(\boldsymbol{a}^t \boldsymbol{y}_k - b_k)\boldsymbol{y}_k$$

Or:

$$\boldsymbol{a} \leftarrow \boldsymbol{a} + \eta(b_k - \boldsymbol{a}^t \boldsymbol{y}_k)\boldsymbol{y}_k$$

Sequential Widrow-Hoff Learning Algorithm

- Set values of hyper-parameters (η, θ, and **b**)
- Initialise **a** to arbitrary solution
- For each sample, $\mathbf{y}_k$, in the dataset in turn
  - update solution: $\boldsymbol{a} \leftarrow \boldsymbol{a} + \eta(b_k - \boldsymbol{a}^t \boldsymbol{y}_k)\boldsymbol{y}_k$

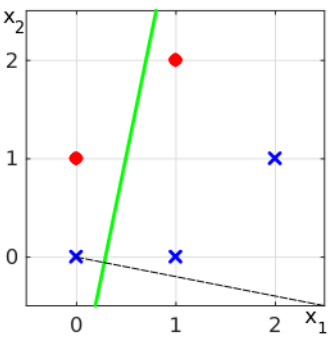
Iterate until $\left|\sum_k (b_k - \boldsymbol{a}^t \boldsymbol{y}_k)\boldsymbol{y}_k\right| < \theta$

Sequential Widrow-Hoff Learning Algorithm

- Initialise **a** to arbitrary solution and select learning rate and **b**
- Until convergence ($\left|\sum_k (b_k - \boldsymbol{a}^t \boldsymbol{y}_k)\boldsymbol{y}_k\right| < \theta$)
- For each sample, $\mathbf{y}_k$, in the dataset in turn
  - $\boldsymbol{a} \leftarrow \boldsymbol{a} + \eta(b_k - \boldsymbol{a}^t \boldsymbol{y}_k)\boldsymbol{y}_k$

Example:

$\mathbf{x}^T$	class	b
[0,0]	1	2
[1,0]	1	2
[2,1]	1	2
[0,1]	-1	2
[1,2]	-1	2

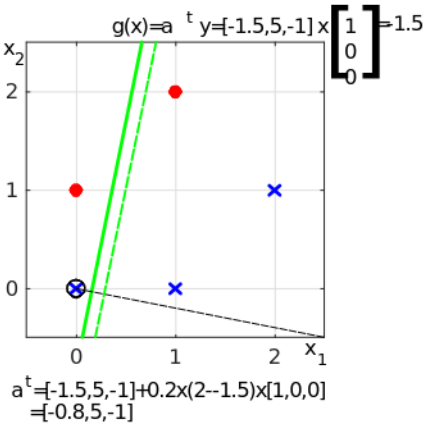


a^t initialised to [-1.5,5,-1] $\eta = 0.2$

Sequential Widrow-Hoff Learning Algorithm

- Initialise **a** to arbitrary solution and select learning rate and **b**
- Until convergence ($\left|\sum_k (b_k - \boldsymbol{a}^t \boldsymbol{y}_k)\boldsymbol{y}_k\right| < \theta$)
- For each sample, $\mathbf{y}_k$, in the dataset in turn
  - $\boldsymbol{a} \leftarrow \boldsymbol{a} + \eta(b_k - \boldsymbol{a}^t \boldsymbol{y}_k)\boldsymbol{y}_k$

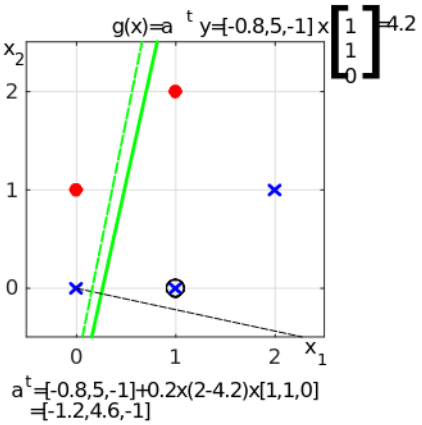
Example:



Sequential Widrow-Hoff Learning Algorithm

- Initialise **a** to arbitrary solution and select learning rate and **b**
- Until convergence ($\left|\sum_k (b_k - \boldsymbol{a}^t \boldsymbol{y}_k)\boldsymbol{y}_k\right| < \theta$)
- For each sample, $\mathbf{y}_k$, in the dataset in turn
  - $\boldsymbol{a} \leftarrow \boldsymbol{a} + \eta(b_k - \boldsymbol{a}^t \boldsymbol{y}_k)\boldsymbol{y}_k$

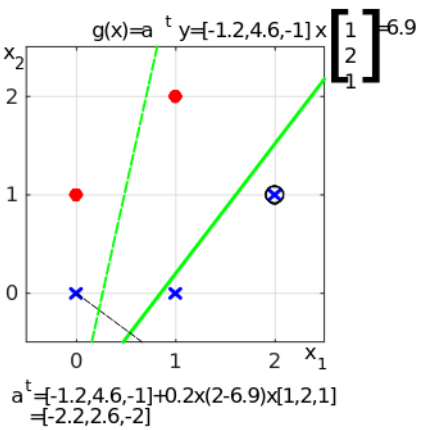
Example:



Sequential Widrow-Hoff Learning Algorithm

- Initialise **a** to arbitrary solution and select learning rate and **b**
- Until convergence ($\left|\sum_k (b_k - \boldsymbol{a}^t \boldsymbol{y}_k)\boldsymbol{y}_k\right| < \theta$)
- For each sample, $\mathbf{y}_k$, in the dataset in turn
  - $\boldsymbol{a} \leftarrow \boldsymbol{a} + \eta(b_k - \boldsymbol{a}^t \boldsymbol{y}_k)\boldsymbol{y}_k$

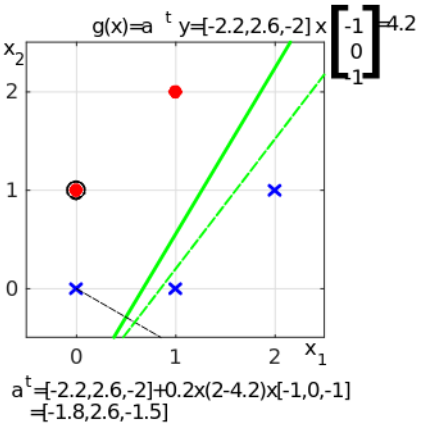
Example:



Sequential Widrow-Hoff Learning Algorithm

- Initialise **a** to arbitrary solution and select learning rate and **b**
- Until convergence ($\left|\sum_k (b_k - \boldsymbol{a}^t \boldsymbol{y}_k)\boldsymbol{y}_k\right| < \theta$)
- For each sample, $\mathbf{y}_k$, in the dataset in turn
  - $\boldsymbol{a} \leftarrow \boldsymbol{a} + \eta(b_k - \boldsymbol{a}^t \boldsymbol{y}_k)\boldsymbol{y}_k$

Example:

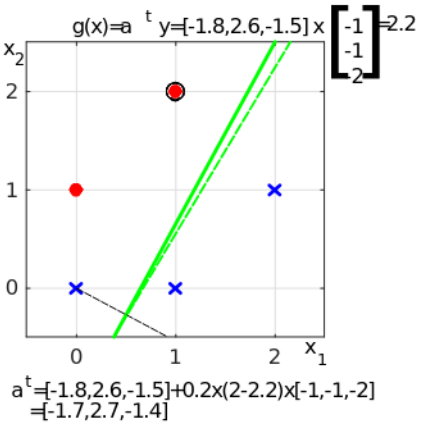


a^t=[-2.2,2.6,-2]+0.2x(2-4.2)x[-1,0,-1]
=[-1.8,2.6,-1.5]

Sequential Widrow-Hoff Learning Algorithm

- Initialise **a** to arbitrary solution and select learning rate and **b**
- Until convergence ($\left|\sum_k (b_k - \boldsymbol{a}^t \boldsymbol{y}_k)\boldsymbol{y}_k\right| < \theta$)
- For each sample, $\mathbf{y}_k$, in the dataset in turn
  - $\boldsymbol{a} \leftarrow \boldsymbol{a} + \eta(b_k - \boldsymbol{a}^t \boldsymbol{y}_k)\boldsymbol{y}_k$

Example:

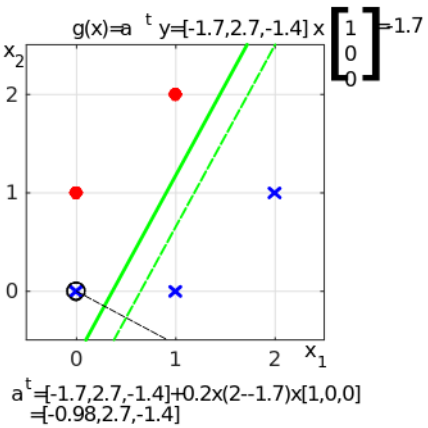


a^t=[-1.8,2.6,-1.5]+0.2x(2-2.2)x[-1,-1,-2]
=[-1.7,2.7,-1.4]

Sequential Widrow-Hoff Learning Algorithm

- Initialise **a** to arbitrary solution and select learning rate and **b**
- Until convergence ($\left|\sum_k (b_k - \boldsymbol{a}^t \boldsymbol{y}_k)\boldsymbol{y}_k\right| < \theta$)
- For each sample, $\mathbf{y}_k$, in the dataset in turn
  - $\boldsymbol{a} \leftarrow \boldsymbol{a} + \eta(b_k - \boldsymbol{a}^t \boldsymbol{y}_k)\boldsymbol{y}_k$

Example:

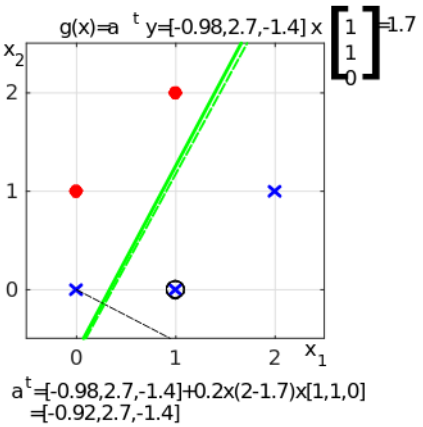


a^t=[-1.7,2.7,-1.4]+0.2x(2--1.7)x[1,0,0]
=[-0.98,2.7,-1.4]

Sequential Widrow-Hoff Learning Algorithm

- Initialise **a** to arbitrary solution and select learning rate and **b**
- Until convergence ($\left|\sum_k (b_k - \boldsymbol{a}^t \boldsymbol{y}_k)\,\boldsymbol{y}_k\right| < \theta$)
- For each sample, $\mathbf{y}_k$, in the dataset in turn
  - $\boldsymbol{a} \leftarrow \boldsymbol{a} + \eta(b_k - \boldsymbol{a}^t \boldsymbol{y}_k)\,\boldsymbol{y}_k$

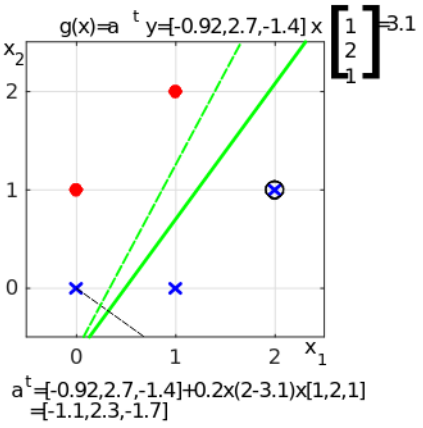
Example:



Sequential Widrow-Hoff Learning Algorithm

- Initialise **a** to arbitrary solution and select learning rate and **b**
- Until convergence ($\left|\sum_k (b_k - \boldsymbol{a}^t \boldsymbol{y}_k)\,\boldsymbol{y}_k\right| < \theta$)
- For each sample, $\mathbf{y}_k$, in the dataset in turn
  - $\boldsymbol{a} \leftarrow \boldsymbol{a} + \eta(b_k - \boldsymbol{a}^t \boldsymbol{y}_k)\,\boldsymbol{y}_k$

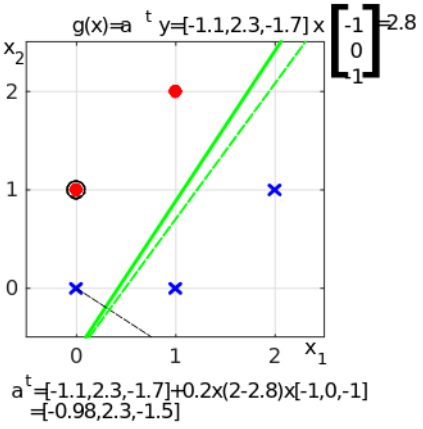
Example:



Sequential Widrow-Hoff Learning Algorithm

- Initialise **a** to arbitrary solution and select learning rate and **b**
- Until convergence ($\left|\sum_k (b_k - \boldsymbol{a}^t \boldsymbol{y}_k)\,\boldsymbol{y}_k\right| < \theta$)
- For each sample, $\mathbf{y}_k$, in the dataset in turn
  - $\boldsymbol{a} \leftarrow \boldsymbol{a} + \eta(b_k - \boldsymbol{a}^t \boldsymbol{y}_k)\,\boldsymbol{y}_k$

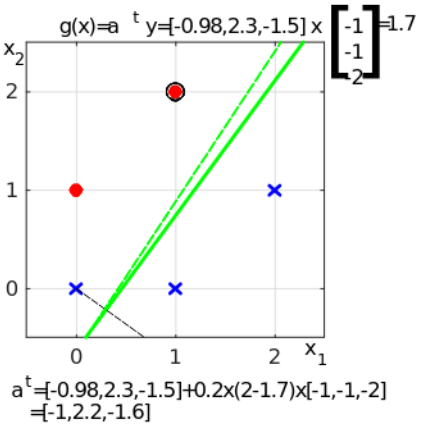
Example:



Sequential Widrow-Hoff Learning Algorithm

- Initialise **a** to arbitrary solution and select learning rate and **b**
- Until convergence ($\left|\sum_k (b_k - \boldsymbol{a}^t \boldsymbol{y}_k)\boldsymbol{y}_k\right| < \theta$)
- For each sample, $\mathbf{y}_k$, in the dataset in turn
  - $\boldsymbol{a} \leftarrow \boldsymbol{a} + \eta(b_k - \boldsymbol{a}^t \boldsymbol{y}_k)\boldsymbol{y}_k$

Example:

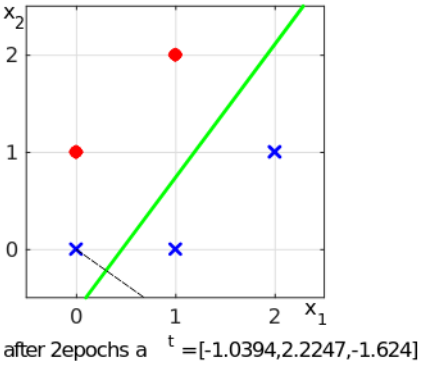


Sequential Widrow-Hoff Learning Algorithm

- Initialise **a** to arbitrary solution and select learning rate and **b**
- ~~Until convergence ($\left|\sum_k (b_k - \boldsymbol{a}^t \boldsymbol{y}_k)\boldsymbol{y}_k\right| < \theta$)~~ Stopping after 2 epochs
- For each sample, $\mathbf{y}_k$, in the dataset in turn
  - $\boldsymbol{a} \leftarrow \boldsymbol{a} + \eta(b_k - \boldsymbol{a}^t \boldsymbol{y}_k)\boldsymbol{y}_k$

Example:

$\mathbf{x}^T$	class	g
[0,0]	1	-1.04
[1,0]	1	1.19
[2,1]	1	1.79
[0,1]	-1	-2.66
[1,2]	-1	-2.06



Notes on Sequential and Batch Learning

If training data consists of n samples:

One update of the parameters is based on:

- n samples with batch learning
- 1 sample with sequential learning

An "epoch" is one pass through all the training data

One epoch corresponds to:

- 1 parameter update with batch learning
- n parameter updates with sequential learning

Mini-batch learning is intermediate between sequential and batch learning