Adaboost

Isaac

Overview

- **Adative boosting base on three ideas :**
  1. Aggregation multiple weak learners
  2. For each weak learner have different weight
  3. Resample
- Advantage
  1. Reduce Bias
  2. Models for **Low Variance** & **High Bias**

Algorithm

Algorithm: Adaboost ($\mathcal{S}, D_1, T, Weak$)

Input: $\mathcal{S} = \{x_i, y_i\}_{i=1}^m$, D_1, T, $Weak(\cdot,\cdot)$
Output: $H(\cdot)$

for $t = 1$ **to** T **do**
Obtain a weak hypothesis using D_t.
$h_t \leftarrow Weak(\mathcal{S}, D_t)$.
Select $\alpha_t \in \mathbb{R}$. Usually:

$$\alpha_t = \frac{1}{2} \ln \left(\frac{1 - R_{emp}(h_t, \mathcal{S}, D_t)}{R_{emp}(h_t, \mathcal{S}, D_t)} \right). \quad (4)$$

Update:

$$D_{t+1}(i) = \frac{D_t(i) exp(-\alpha_t y_i h_t(x_i))}{Z_t}. \quad (5)$$

where Z_t is a normalization factor so that D_{t+1} will be a distribution.
end

Return final hypothesis:

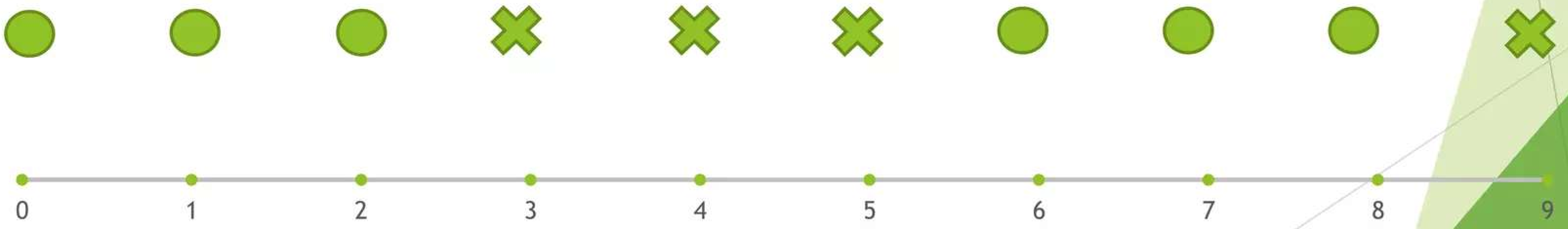
$$H(x) = sign\left(\sum_{t=1}^{T} \frac{\alpha_t}{\sum_t \alpha_t} h_t(x) \right). \quad (6)$$

Fig. 1: Algoritmo Adaboost.

Example of AdaBoosting

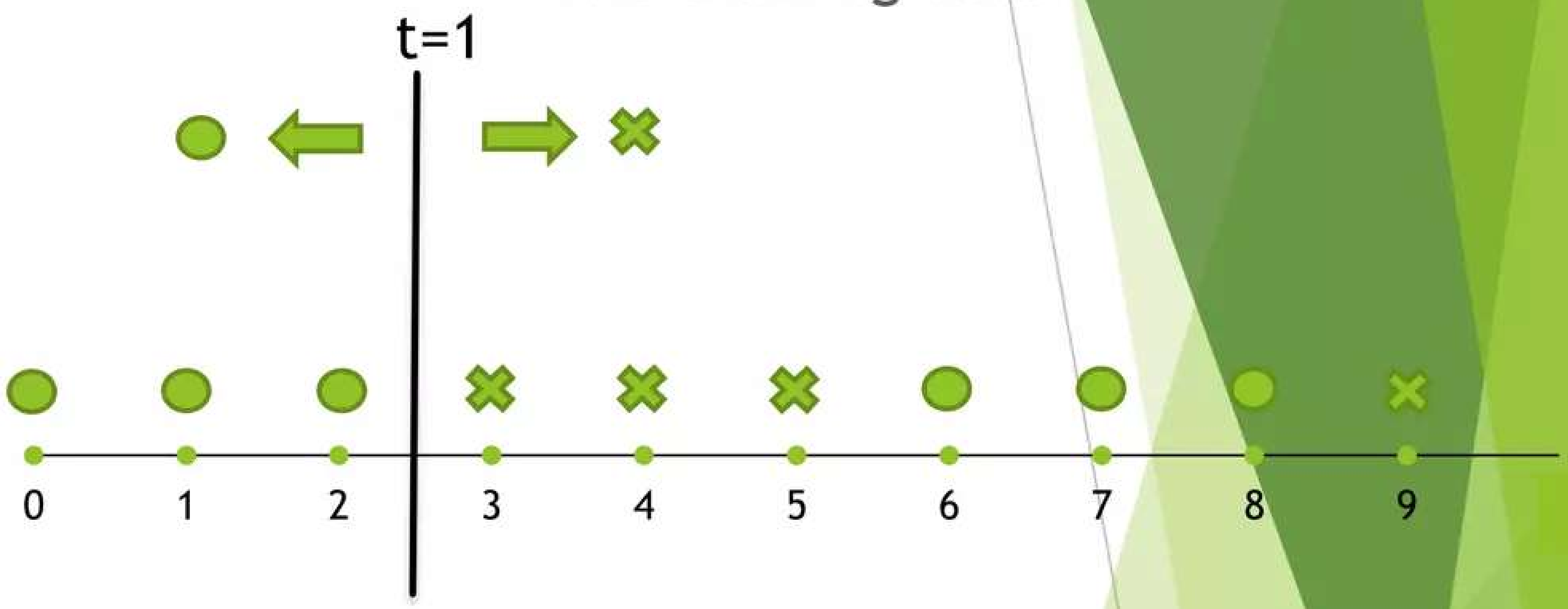
The training data

For example , there is one dimension 10 points, some of show others show
We want to use AdaBoosting to classify those 10 points.



Running the algorithm(1)

The training data



We start with the following probabilities:

(t means iter)

p_0	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8	p_9
0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1

$t = 1$. The best threshold is between 2 and 3.

$$h_1(x) = I(x < 2.5)$$
$$\epsilon_1 = 0.3$$
$$\alpha_1 = 0.423649$$

Updating the probabilities:

index:	0	1	2	3	4	5	6	7	8	9
correct:	y	y	y	y	y	y	n	n	n	y
old p_i	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
pre-normalized p_i	.06547	.06547	.06547	.06547	.06547	.06547	.15	.15	.15	.06547
$Z_1 = 0.916515$										
new p_i	.07143	.07143	.07143	.07143	.07143	.07143	.16667	.16667	.16667	.07143

$f_1(x) = 0.423649 \boxed{I(x < 2.5)}$ 3 mistakes

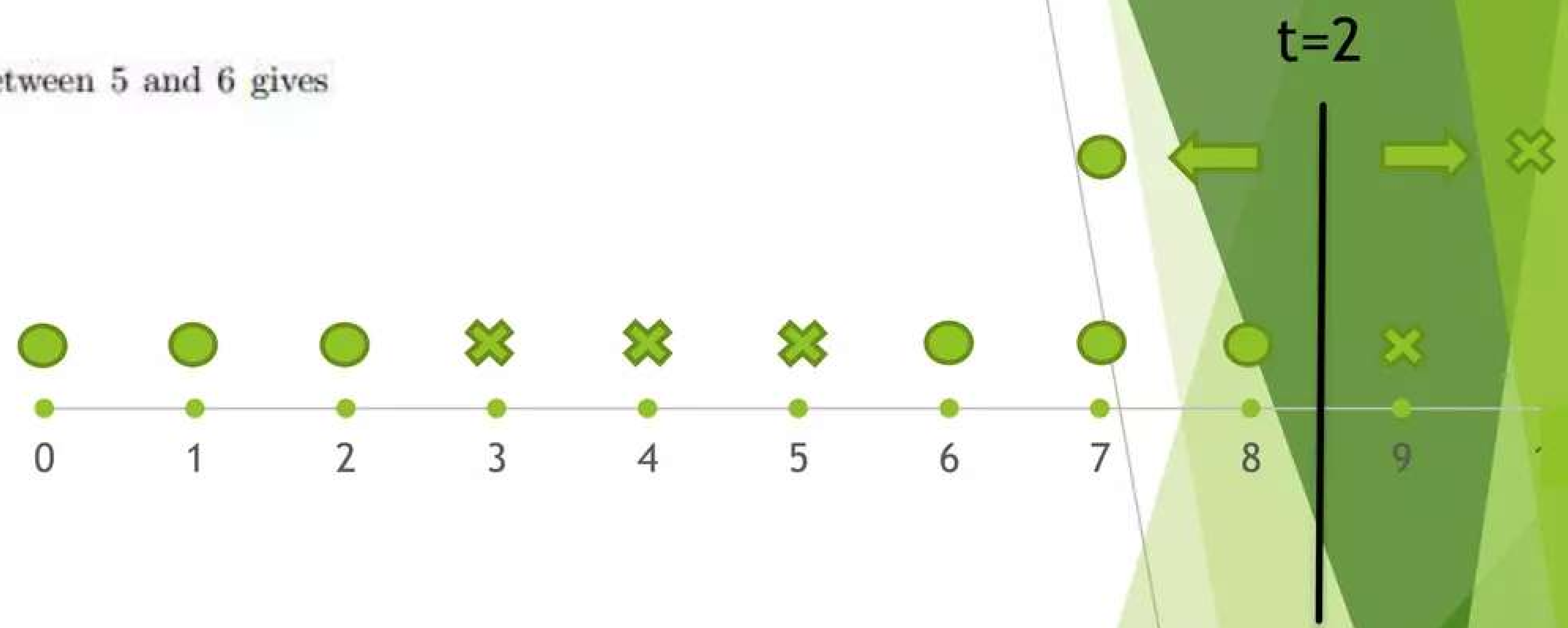
If x < 2.5 then I = 1 else I = -1
And if f(x) > 0 will classify to ●
else f(x) < 0 will classify to ✖

Running the algorithm(2)

$t = 2$. Now a threshold between 2 and 3 gives error of 0.5, the threshold between 5 and 6 gives 0.28, the threshold between 8 and 9 gives 0.214, best.

$$h_2(x) = I(x < 8.5)$$
$$\epsilon_2 = 0.214$$
$$\alpha_2 = 0.6496$$

The training data



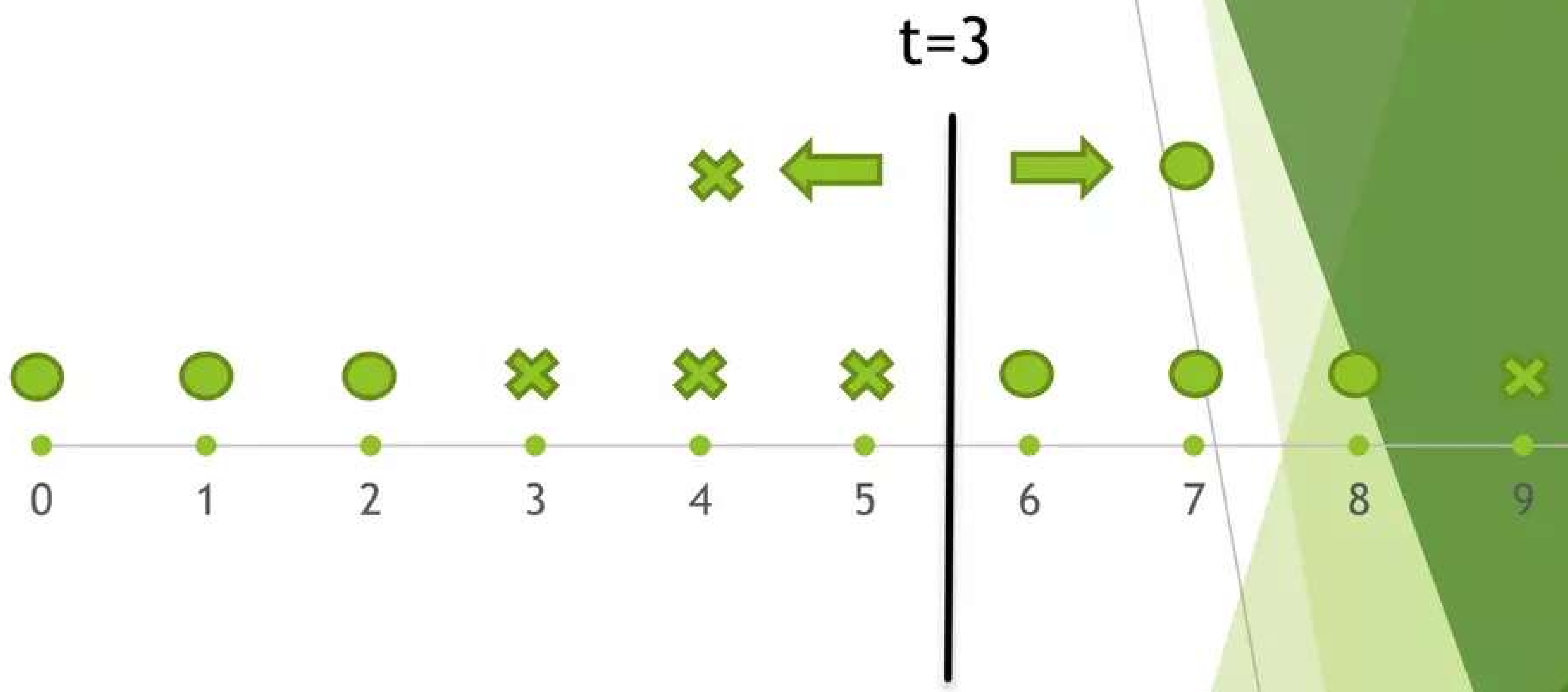
Updating the probabilities:

index:	0	1	2	3	4	5	6	7	8	9
correct:	y	y	y	n	n	n	y	y	y	y
pre-normalized p_i	.037	.037	.037	.137	.137	.137	.087	.087	.087	.037
$Z_2 = 0.82$										
new p_i	.045	.045	.045	.167	.167	.167	.106	.106	.106	.045

$f_2(x) = 0.423649\ I(x < 2.5) + 0.6496\ I(x < 8.5)$,   3 mistakes

Running the algorithm(3)

The training data



$t = 3$. The best threshold is between 5 and 6.

$$h_3(x) = I(x > 5.5)$$
$$\epsilon_3 = 0.1818$$
$$\alpha_3 = 0.7520$$

Updating the probabilities:

index:	0	1	2	3	4	5	6	7	8	9
correct:	n	n	n	y	y	y	y	y	y	n
pre-normalized p_i	.0964	.0964	.0964	.078	.078	.078	.05	.05	.05	.0964
$Z_3 = 0.77139$										
new p_i	.125	.125	.125	.102	.102	.102	.064	.064	.064	.125

$f_3(x) = 0.423649\ I(x < 2.5) + 0.6496\ I(x < 8.5) + 0.752\ I(x > 5.5)$, 0 mistakes

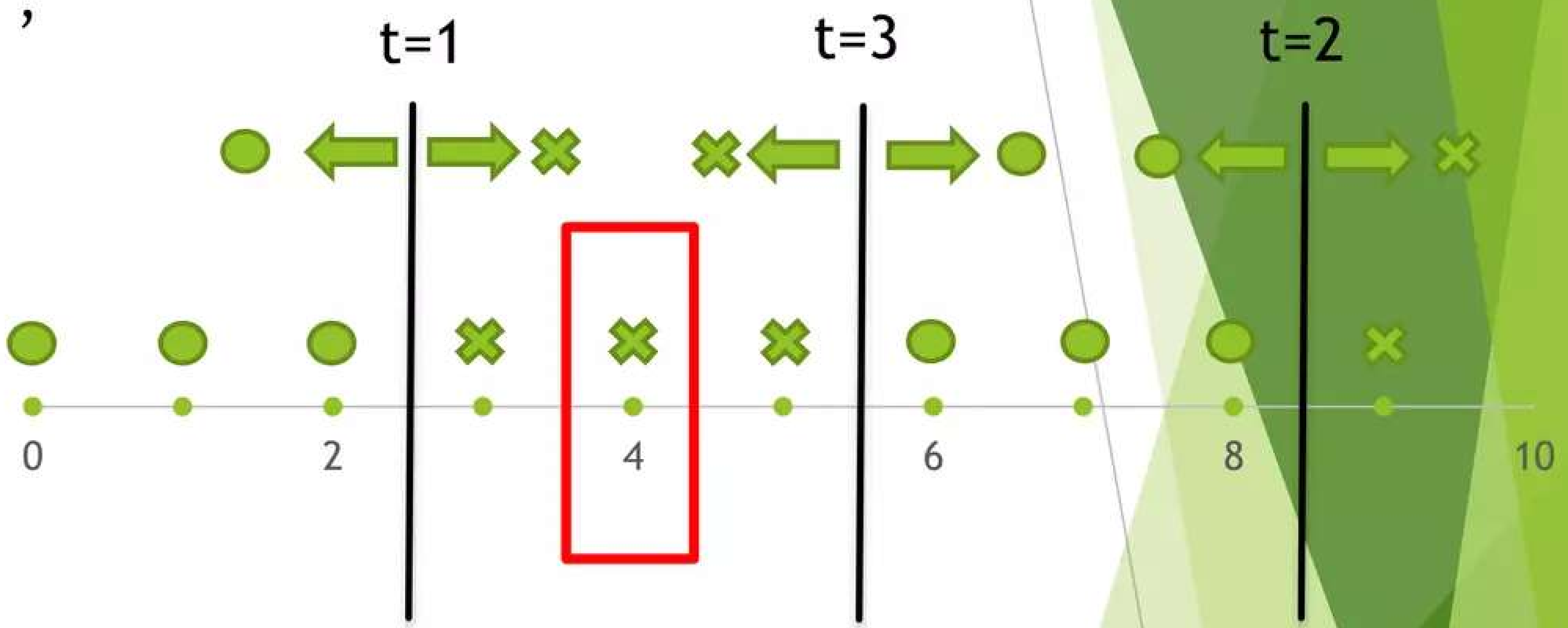
Running the algorithm(4)

For example : we want to decide point 4 is or ,
we need to follow

$f_3(x) = 0.423649 * (-1) + 0.6496 * (-1) + 0.752 * 1$
$= -0.321249$

So, point 4 will classify to

The training data



Updating the probabilities:

index:	0	1	2	3	4	5	6	7	8	9
correct:	n	n	n	y	y	y	y	y	y	n
pre-normalized p_i	.0964	.0964	.0964	.078	.078	.078	.05	.05	.05	.0964
$Z_3 = 0.77139$										
new p_i	.125	.125	.125	.102	.102	.102	.064	.064	.064	.125

$f_3(x) = 0.423649\ I(x < 2.5) + 0.6496\ I(x < 8.5) + 0.752\ I(x > 5.5)$, 0 mistakes

Benifit

1. Adaboost is a high precise classifier
2. We can use many models to build subclassifier
3. When using simple classifier the calculated result is easy to understand
4. No need to do feature selection
5. No need to worry about overfitting

Disadvantages

1. Very sensitive to outlier(means outlier may do big influence)