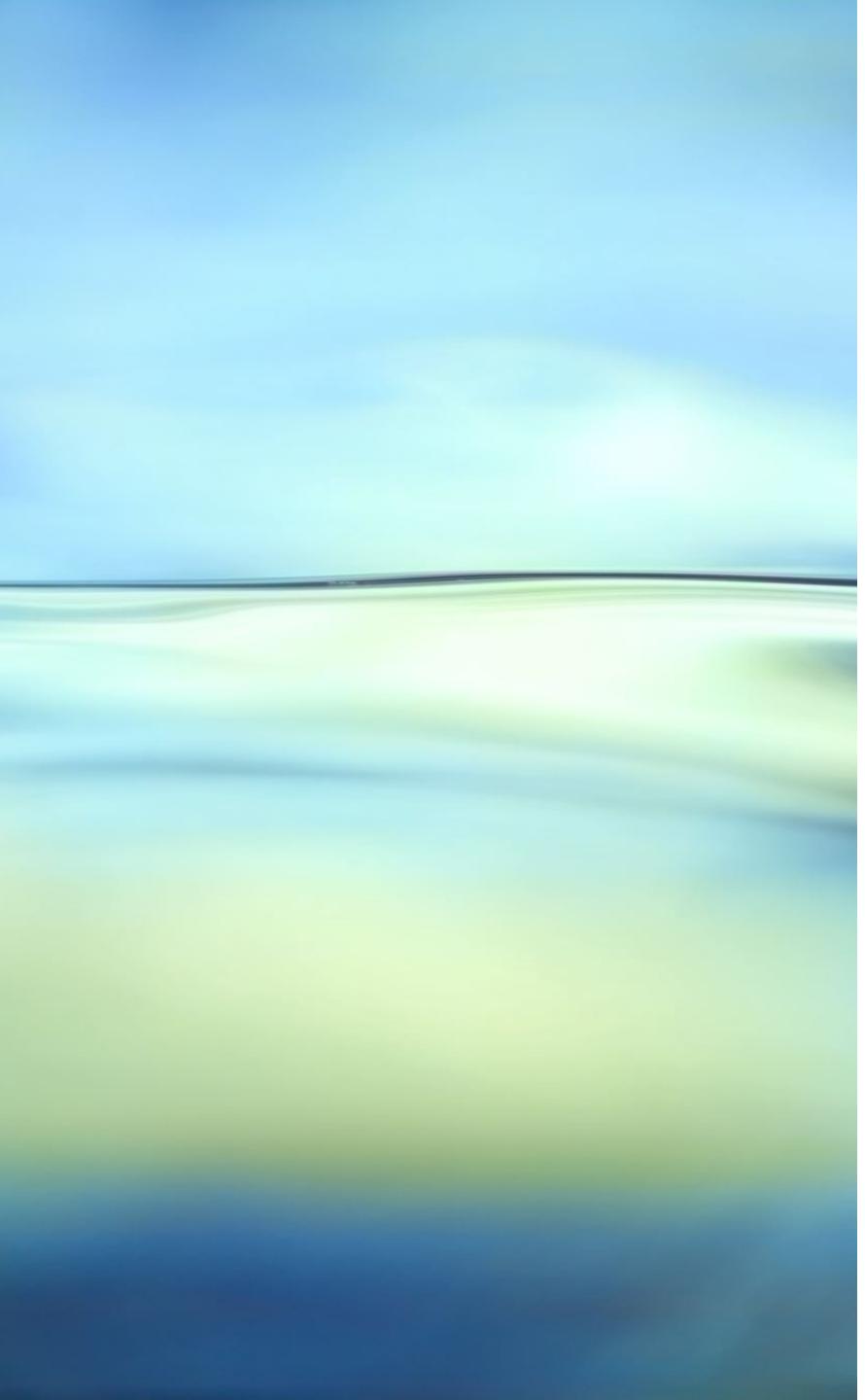


A close-up photograph of several red flowers, likely poppies, with numerous small, clear water droplets clinging to their petals. The background is blurred, showing more of the same flowers in soft focus.

BOOSTING



Boosting

What is boosting?

Benefit of boosting

How boosting works

Clarify things about weak learner

Adaboost

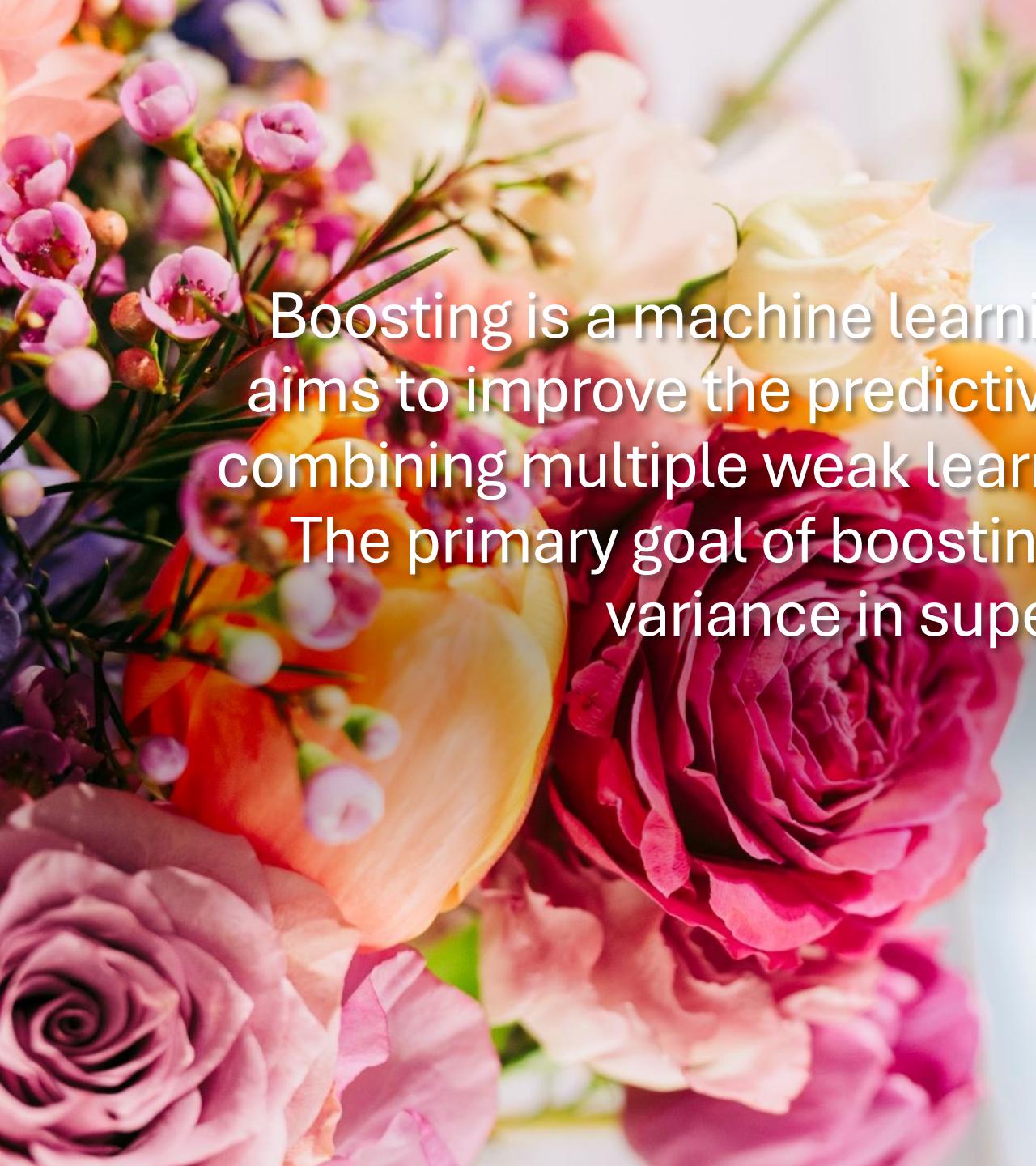
Gradient boosting

LightGBM, XGBoost, CatBoost

Further reading

A close-up photograph of a colorful bouquet of flowers. The bouquet includes several large, full roses in shades of pink, red, and yellow. There are also some smaller, delicate flowers like wax flowers and baby's breath. The colors are bright and varied, creating a lush, floral composition.

What is boosting?

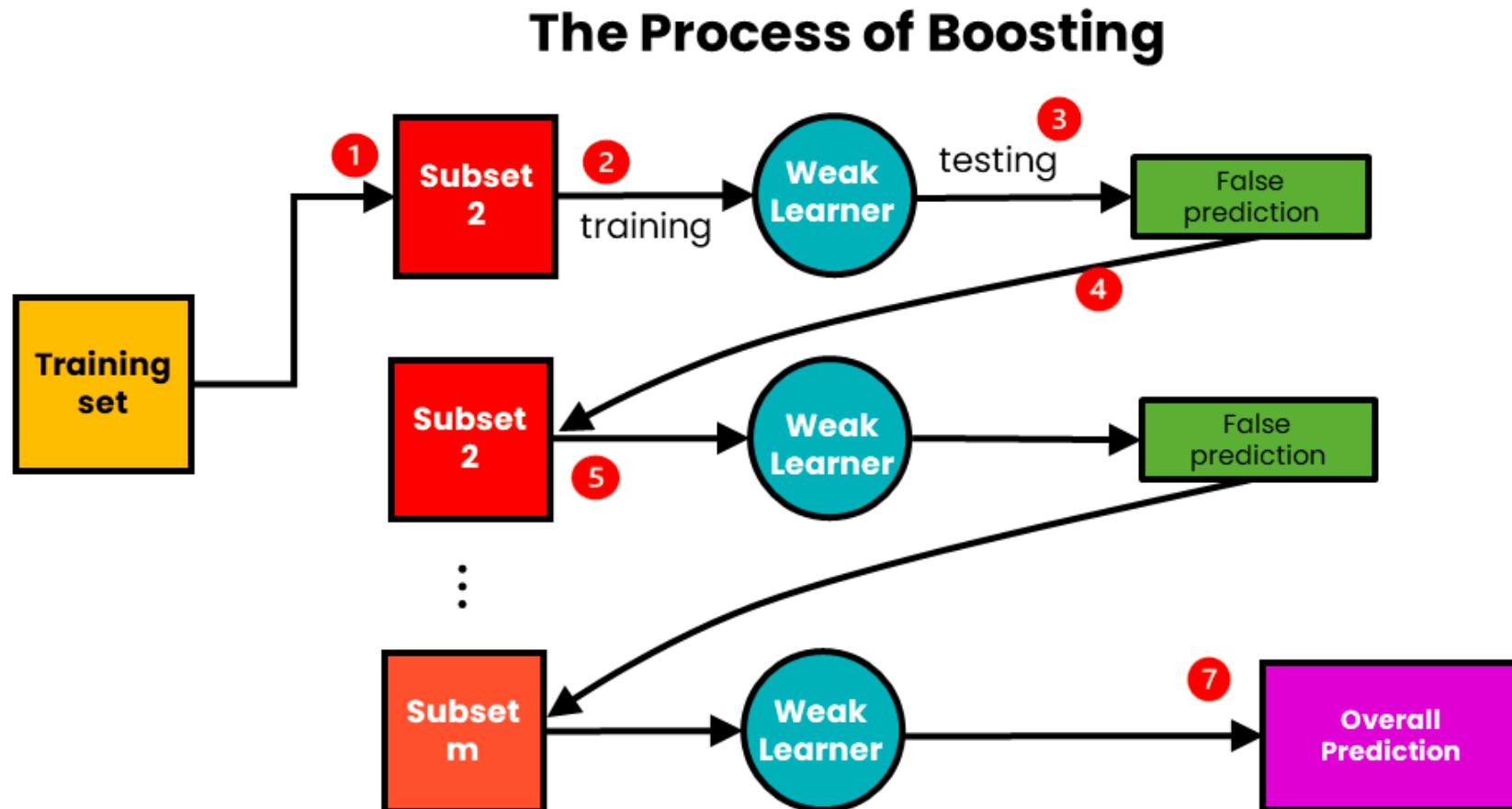
A close-up photograph of a colorful bouquet of flowers. The bouquet includes several large, full roses in shades of pink, red, and orange. Interspersed among the roses are clusters of smaller, delicate pink flowers, likely carnations or wax flowers. The flowers are arranged in a dense, circular pattern, filling the left side of the frame.

Boosting is a machine learning ensemble technique that aims to improve the predictive performance of a model by combining multiple weak learners to create a strong learner. The primary goal of boosting is to reduce both bias and variance in supervised learning.

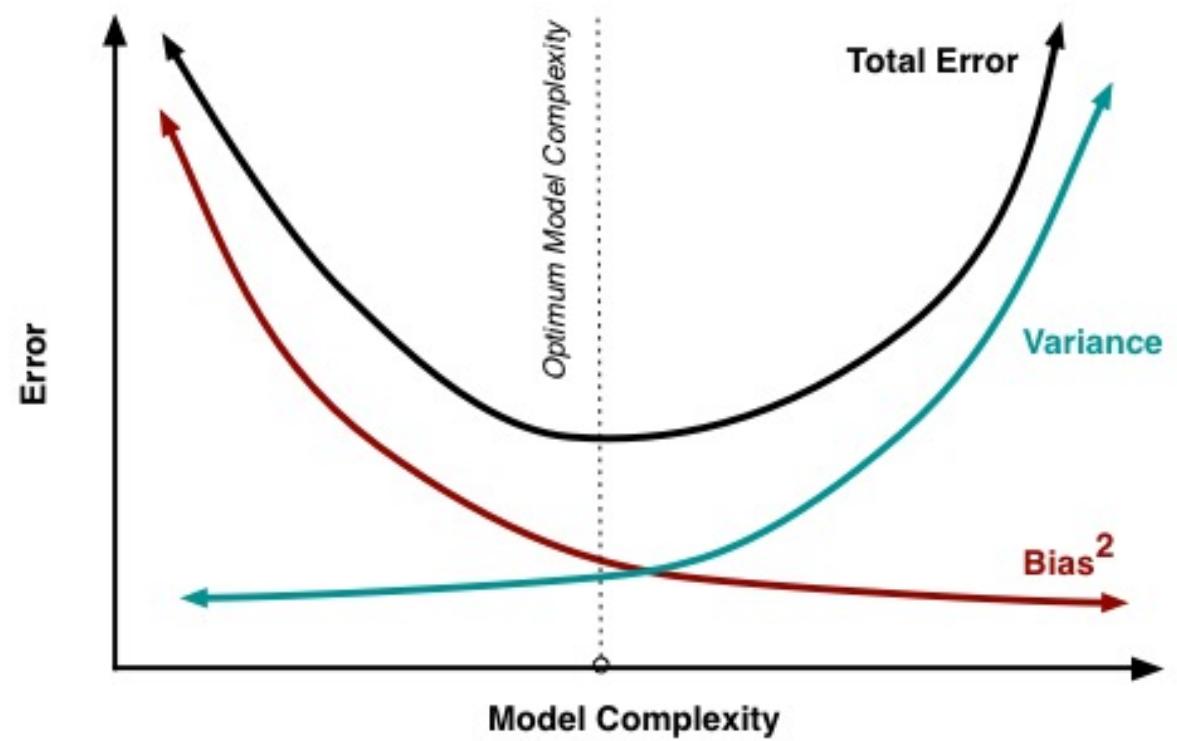
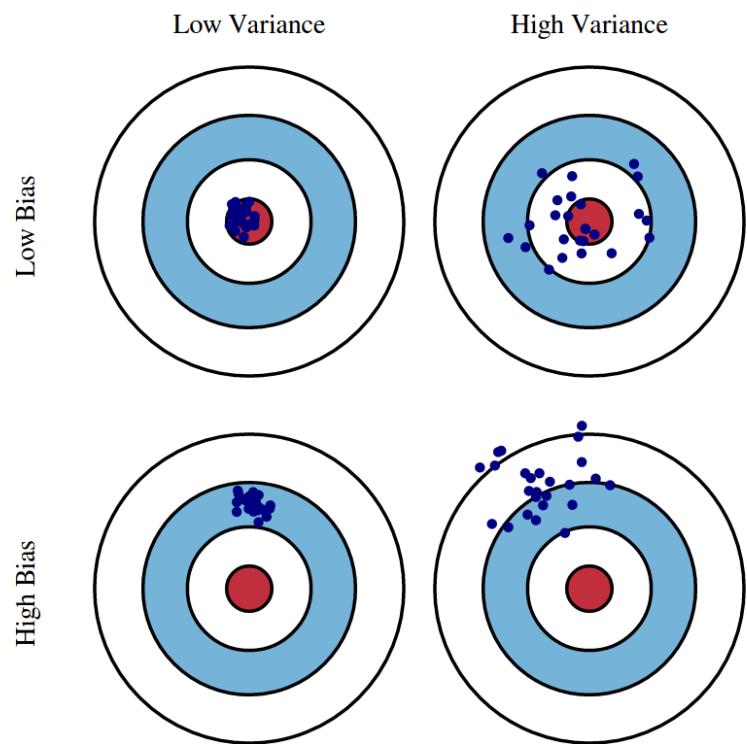
Benefit of boosting (with SL)

- Improve predictive performance
- Reduction of bias (and variance)
- Robust to overfitting
- **Sensitive to outliers**

How boosting works



How boosting works





Clarify things about weak
learner

Clarify things about weak learner



AdaBoost

AdaBoost

- What is AdaBoost?
- Calculation step
- Calculation concept
- Calculation example
- Discussion about AdaBoost
- Code

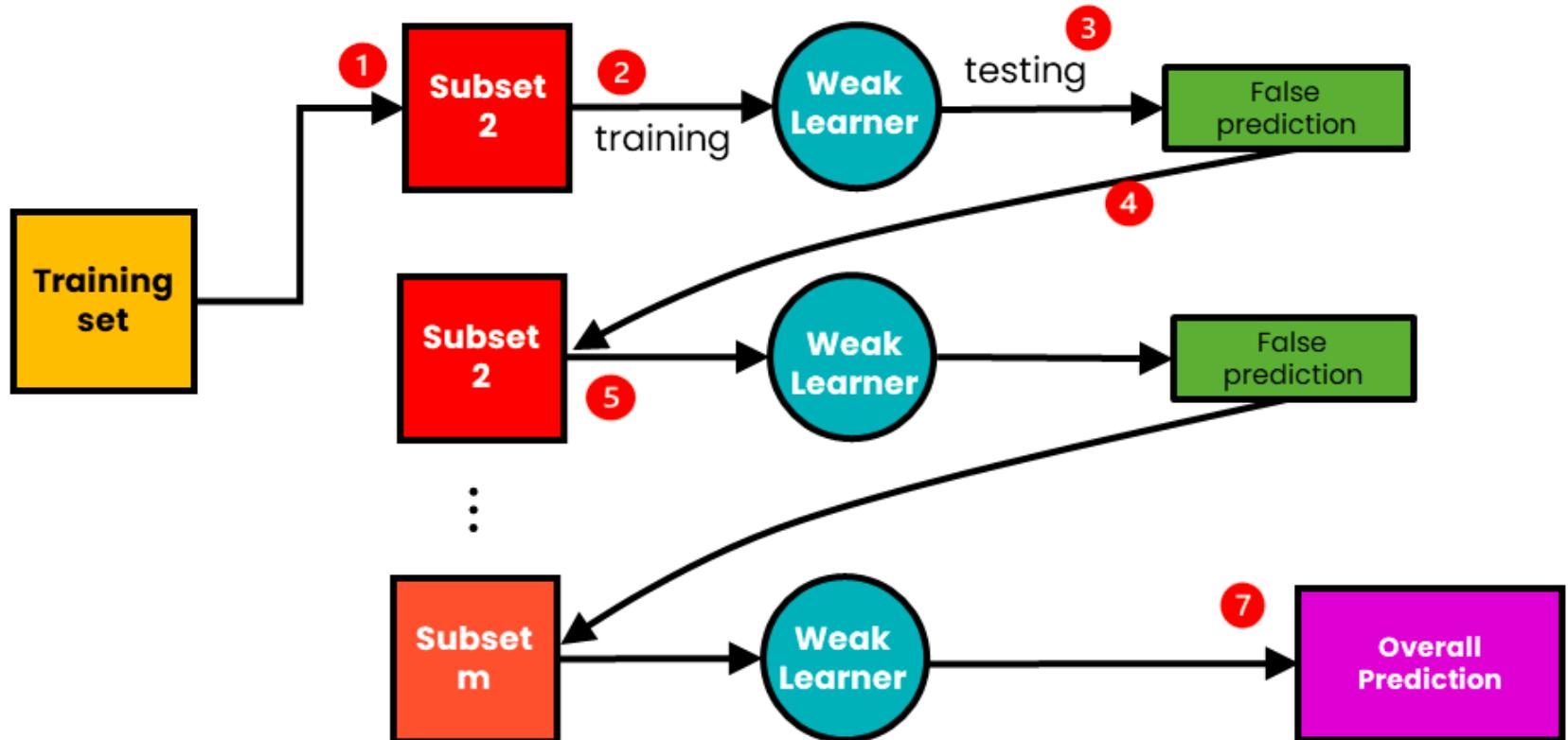
What is AdaBoost?

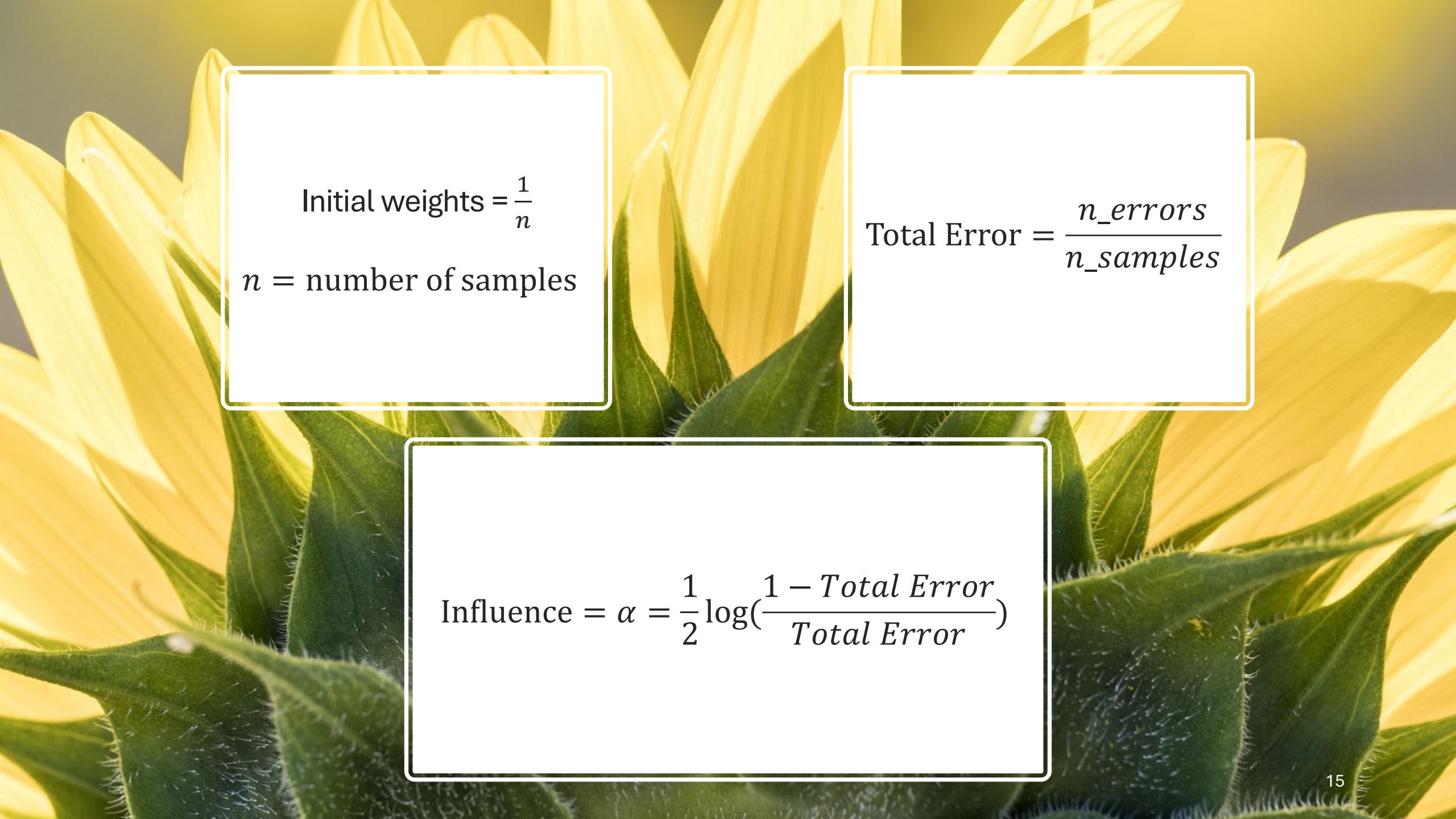
AdaBoost, short for Adaptive Boosting, is a popular ensemble learning technique used primarily for classification tasks. It works by combining multiple weak classifiers to create a strong classifier.

Calculation step

1. Initialize weights
2. Train weak learner
3. Calculate Total Error
4. Calculate influence
5. Update sample weights
6. Normalize sample weights
7. Repeat step 2-6 until meet stopping criteria

The Process of Boosting

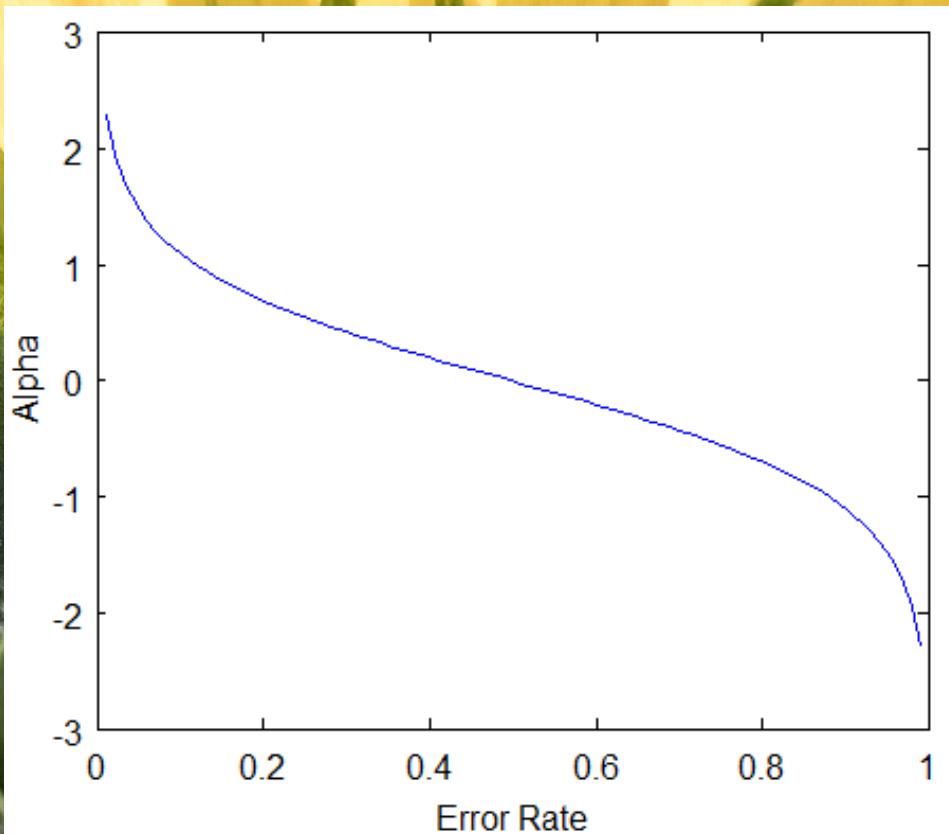


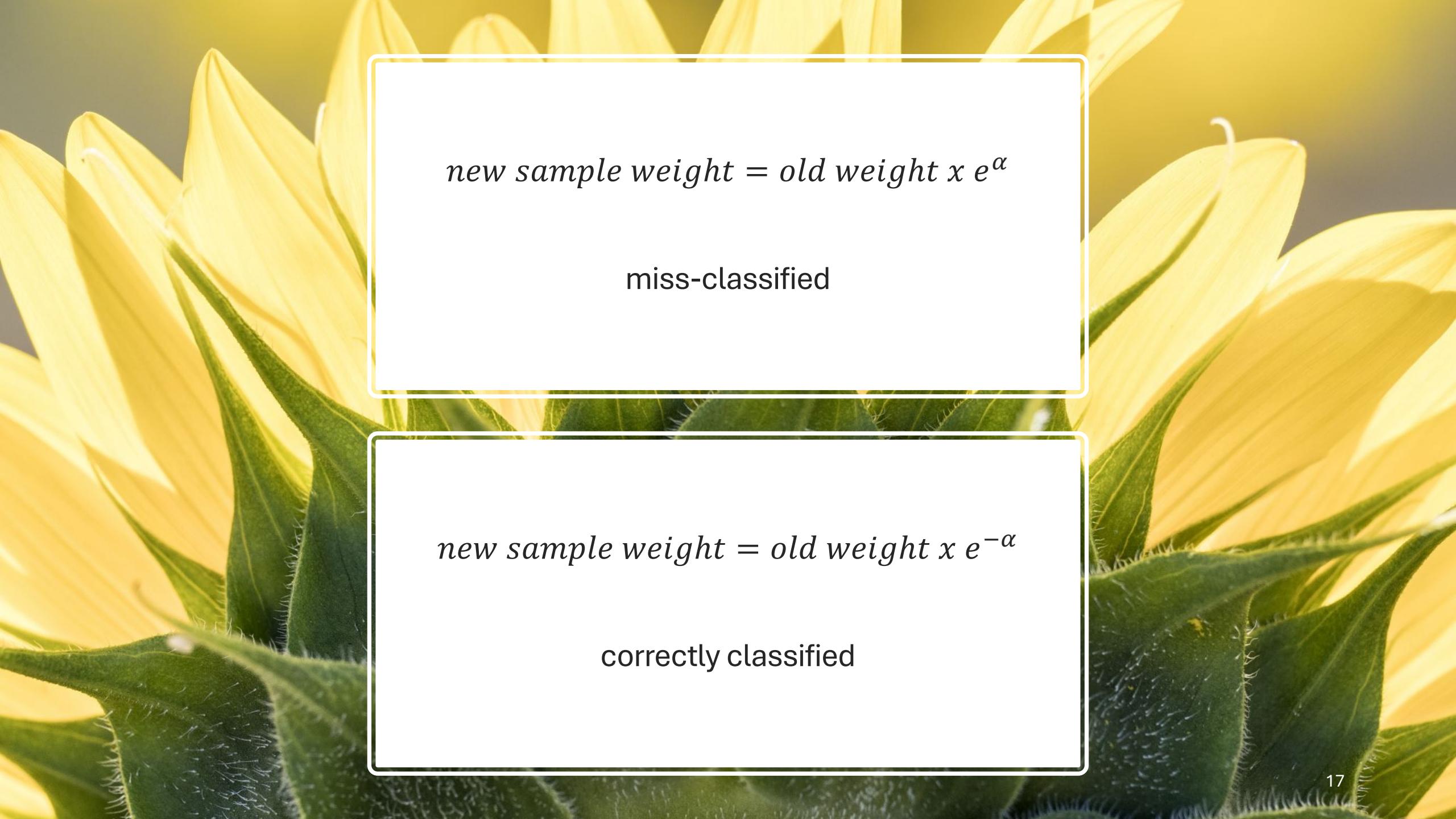
A close-up photograph of sunflowers, with their bright yellow petals and green leaves filling the frame. Three white rectangular boxes with black borders are overlaid on the image, containing mathematical formulas.
$$\text{Initial weights} = \frac{1}{n}$$

n = number of samples

$$\text{Total Error} = \frac{n_errors}{n_samples}$$
$$\text{Influence} = \alpha = \frac{1}{2} \log\left(\frac{1 - \text{Total Error}}{\text{Total Error}}\right)$$

$$\text{Influence} = \alpha = \frac{1}{2} \log\left(\frac{1 - \text{Total Error}}{\text{Total Error}}\right)$$



A close-up photograph of sunflower petals, showing their characteristic yellow color and green centers.
$$\text{new sample weight} = \text{old weight} \times e^\alpha$$

miss-classified

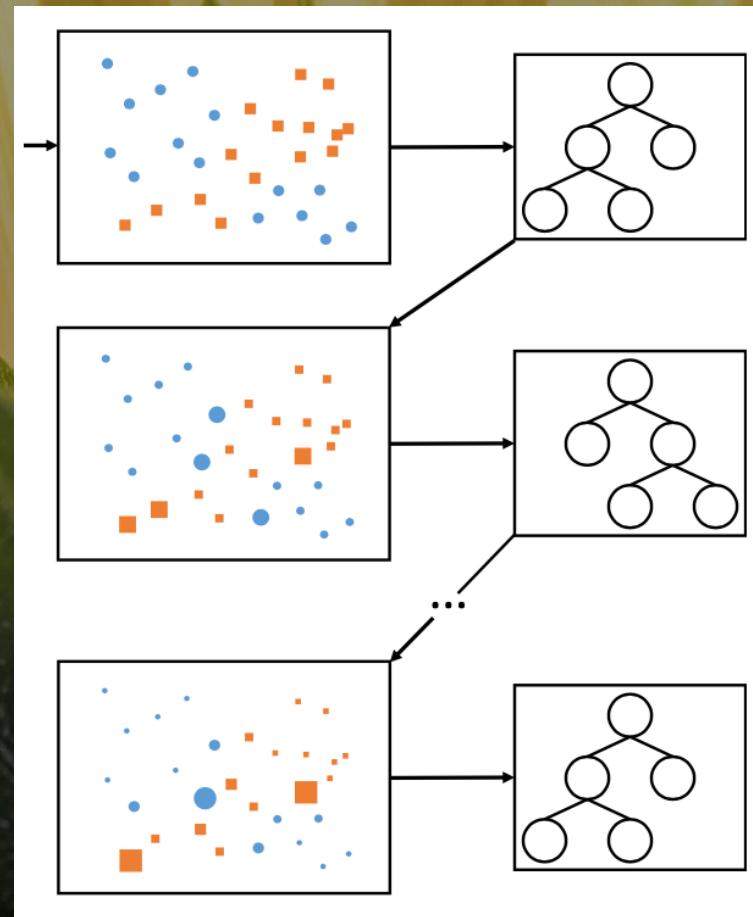
$$\text{new sample weight} = \text{old weight} \times e^{-\alpha}$$

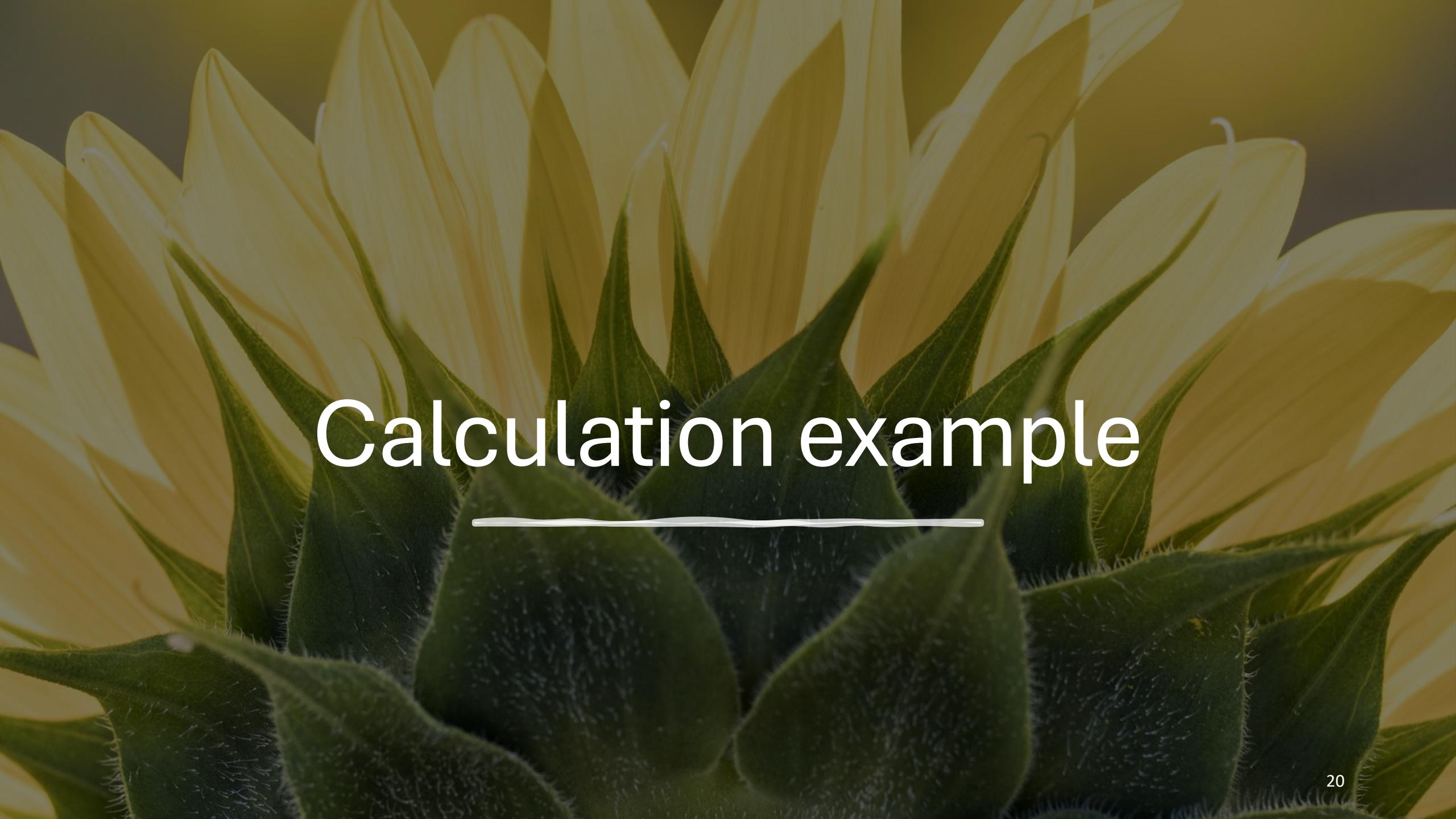
correctly classified

$$H(x) = \alpha_1 h_1(x) + \alpha_2 h_2(x) + \cdots + \alpha_m h_m(x)$$

final prediction = sign(H(x))

Calculation concept





Calculation example

Initialize weights

x_1	x_2	x_3	x_4	y	weight
$x_{1,1}$	$x_{1,2}$	$x_{1,3}$	$x_{1,4}$	<i>True</i>	1/6
$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	$x_{2,4}$	<i>True</i>	1/6
$x_{3,1}$	$x_{3,2}$	$x_{3,3}$	$x_{3,4}$	<i>False</i>	1/6
$x_{4,1}$	$x_{4,2}$	$x_{4,3}$	$x_{4,4}$	<i>True</i>	1/6
$x_{5,1}$	$x_{5,2}$	$x_{5,3}$	$x_{5,4}$	<i>False</i>	1/6
$x_{6,1}$	$x_{6,2}$	$x_{6,3}$	$x_{6,4}$	<i>False</i>	1/6

Iteration1 – train weak learner

x_1	x_2	x_3	x_4	y	$h_1(x) \Rightarrow \hat{y}$	weight
$x_{1,1}$	$x_{1,2}$	$x_{1,3}$	$x_{1,4}$	True	True	1/6
$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	$x_{2,4}$	True	True	1/6
$x_{3,1}$	$x_{3,2}$	$x_{3,3}$	$x_{3,4}$	False	True	1/6
$x_{4,1}$	$x_{4,2}$	$x_{4,3}$	$x_{4,4}$	True	False	1/6
$x_{5,1}$	$x_{5,2}$	$x_{5,3}$	$x_{5,4}$	False	False	1/6
$x_{6,1}$	$x_{6,2}$	$x_{6,3}$	$x_{6,4}$	False	False	1/6

Iteration1 – Calculate Total Error

x_1	x_2	x_3	x_4	y	$h_1(x) \Rightarrow \hat{y}$	weight
$x_{1,1}$	$x_{1,2}$	$x_{1,3}$	$x_{1,4}$	True	True	1/6
$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	$x_{2,4}$	True	True	1/6
$x_{3,1}$	$x_{3,2}$	$x_{3,3}$	$x_{3,4}$	False	True	1/6
$x_{4,1}$	$x_{4,2}$	$x_{4,3}$	$x_{4,4}$	True	False	1/6
$x_{5,1}$	$x_{5,2}$	$x_{5,3}$	$x_{5,4}$	False	False	1/6
$x_{6,1}$	$x_{6,2}$	$x_{6,3}$	$x_{6,4}$	False	False	1/6

$$Total\ Error = \frac{n_errors}{n_samples} = \frac{2}{6} = \frac{1}{3}$$

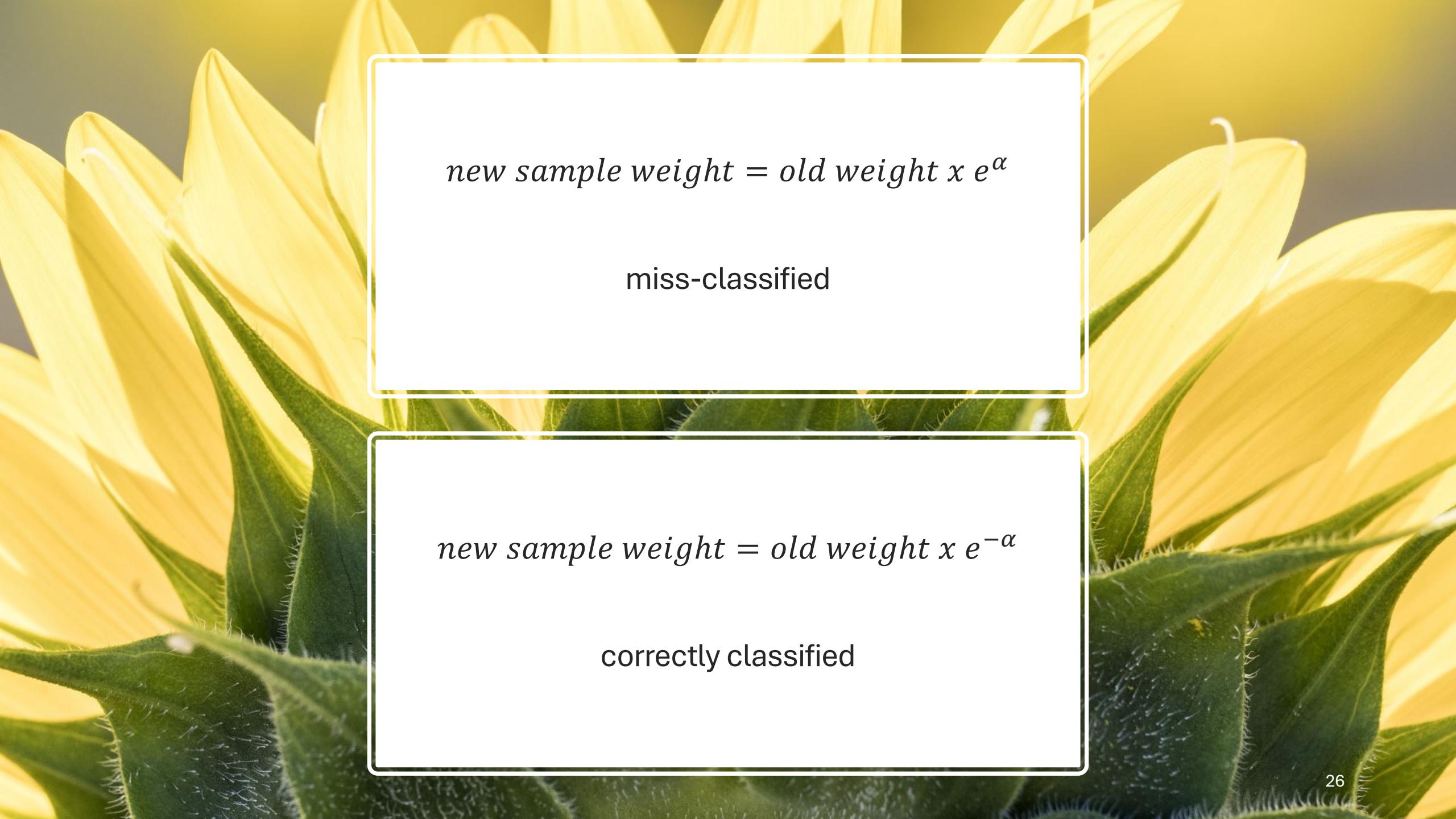
Iteration1 – Calculate influence

x_1	x_2	x_3	x_4	y	$h_1(x) \Rightarrow \hat{y}$	weight
$x_{1,1}$	$x_{1,2}$	$x_{1,3}$	$x_{1,4}$	True	True	1/6
$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	$x_{2,4}$	True	True	1/6
$x_{3,1}$	$x_{3,2}$	$x_{3,3}$	$x_{3,4}$	False	True	1/6
$x_{4,1}$	$x_{4,2}$	$x_{4,3}$	$x_{4,4}$	True	False	1/6
$x_{5,1}$	$x_{5,2}$	$x_{5,3}$	$x_{5,4}$	False	False	1/6
$x_{6,1}$	$x_{6,2}$	$x_{6,3}$	$x_{6,4}$	False	False	1/6

$$Influence = \alpha_1 = \frac{1}{2} \log \left(\frac{1 - Total\ Error}{Total\ Error} \right) = \frac{1}{2} \log \frac{\left(1 - \frac{1}{3}\right)}{\frac{1}{3}} = 0.35$$

Iteration1 – update sample weights

x_1	x_2	x_3	x_4	y	$h_1(x) \Rightarrow \hat{y}$	weight
$x_{1,1}$	$x_{1,2}$	$x_{1,3}$	$x_{1,4}$	<i>True</i>	<i>True</i>	$\frac{1}{6}e^{-0.35}$
$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	$x_{2,4}$	<i>True</i>	<i>True</i>	$\frac{1}{6}e^{-0.35}$
$x_{3,1}$	$x_{3,2}$	$x_{3,3}$	$x_{3,4}$	<i>False</i>	<i>True</i>	$\frac{1}{6}e^{0.35}$
$x_{4,1}$	$x_{4,2}$	$x_{4,3}$	$x_{4,4}$	<i>True</i>	<i>False</i>	$\frac{1}{6}e^{0.35}$
$x_{5,1}$	$x_{5,2}$	$x_{5,3}$	$x_{5,4}$	<i>False</i>	<i>False</i>	$\frac{1}{6}e^{-0.35}$
$x_{6,1}$	$x_{6,2}$	$x_{6,3}$	$x_{6,4}$	<i>False</i>	<i>False</i>	$\frac{1}{6}e^{-0.35}$



$$\text{new sample weight} = \text{old weight} \times e^\alpha$$

miss-classified

$$\text{new sample weight} = \text{old weight} \times e^{-\alpha}$$

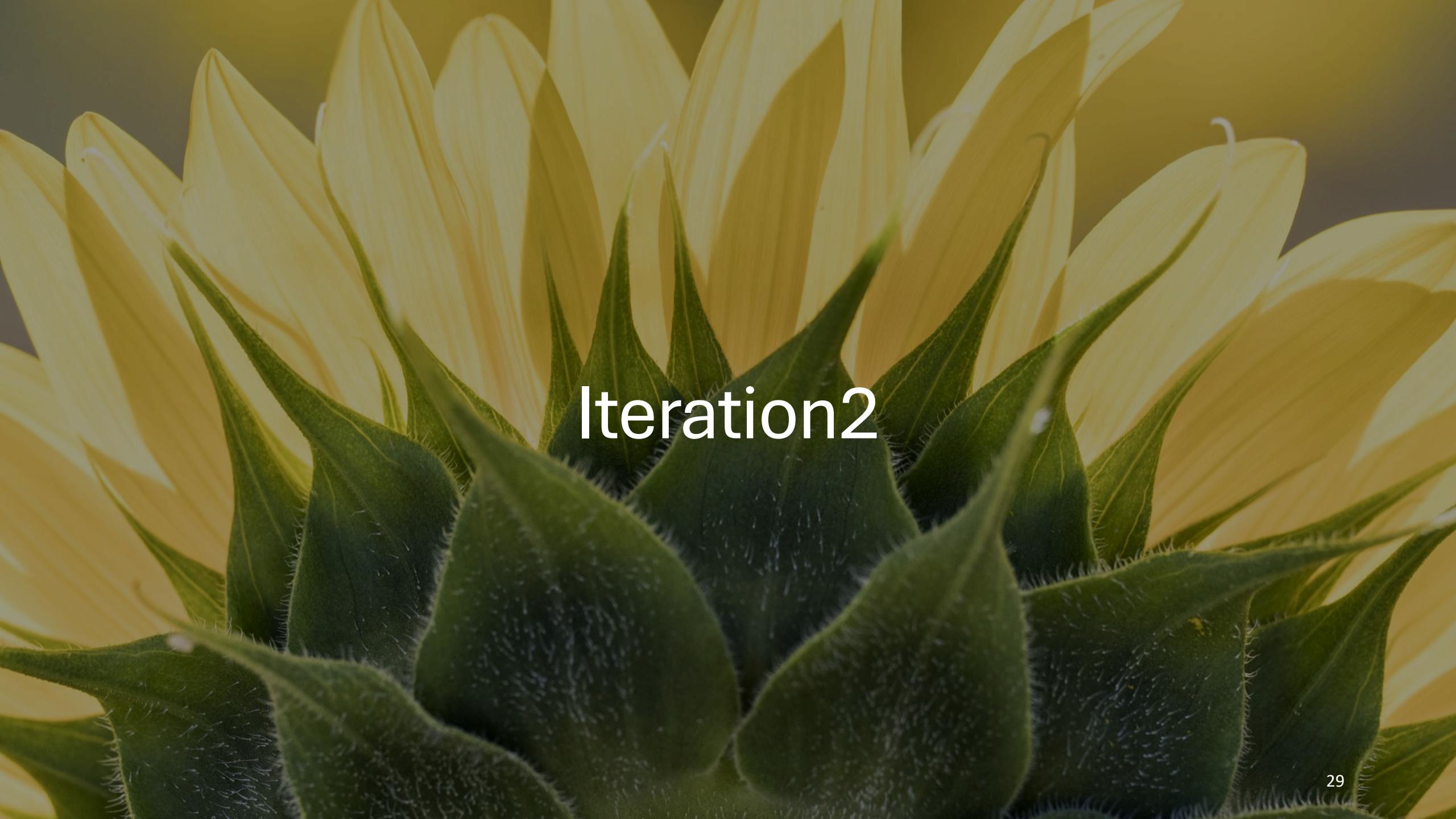
correctly classified

Iteration1 – update sample weights

x_1	x_2	x_3	x_4	y	$h_1(x) \Rightarrow \hat{y}$	weight
$x_{1,1}$	$x_{1,2}$	$x_{1,3}$	$x_{1,4}$	True	True	0.70
$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	$x_{2,4}$	True	True	0.70
$x_{3,1}$	$x_{3,2}$	$x_{3,3}$	$x_{3,4}$	False	True	1.42
$x_{4,1}$	$x_{4,2}$	$x_{4,3}$	$x_{4,4}$	True	False	1.42
$x_{5,1}$	$x_{5,2}$	$x_{5,3}$	$x_{5,4}$	False	False	0.70
$x_{6,1}$	$x_{6,2}$	$x_{6,3}$	$x_{6,4}$	False	False	0.70

Iteration1 – Normalize sample weights

x_1	x_2	x_3	x_4	y	$h_1(x) \Rightarrow \hat{y}$	weight
$x_{1,1}$	$x_{1,2}$	$x_{1,3}$	$x_{1,4}$	True	True	0.12
$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	$x_{2,4}$	True	True	0.12
$x_{3,1}$	$x_{3,2}$	$x_{3,3}$	$x_{3,4}$	False	True	0.25
$x_{4,1}$	$x_{4,2}$	$x_{4,3}$	$x_{4,4}$	True	False	0.25
$x_{5,1}$	$x_{5,2}$	$x_{5,3}$	$x_{5,4}$	False	False	0.12
$x_{6,1}$	$x_{6,2}$	$x_{6,3}$	$x_{6,4}$	False	False	0.12

A close-up photograph of a sunflower head. The flower has numerous bright yellow, petal-like structures radiating from a dark center. In the foreground, several large, serrated green leaves are visible, some with fine hairs along their edges.

Iteration2

Iteration2 – train weak learner

x_1	x_2	x_3	x_4	y	$h_2(x) \Rightarrow \hat{y}$	weight
$x_{1,1}$	$x_{1,2}$	$x_{1,3}$	$x_{1,4}$	True	False	0.12
$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	$x_{2,4}$	True	True	0.12
$x_{3,1}$	$x_{3,2}$	$x_{3,3}$	$x_{3,4}$	False	False	0.25
$x_{4,1}$	$x_{4,2}$	$x_{4,3}$	$x_{4,4}$	True	True	0.25
$x_{5,1}$	$x_{5,2}$	$x_{5,3}$	$x_{5,4}$	False	False	0.12
$x_{6,1}$	$x_{6,2}$	$x_{6,3}$	$x_{6,4}$	False	False	0.12

Iteration2 – Calculate Total Error

x_1	x_2	x_3	x_4	y	$h_2(x) \Rightarrow \hat{y}$	weight
$x_{1,1}$	$x_{1,2}$	$x_{1,3}$	$x_{1,4}$	True	False	0.12
$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	$x_{2,4}$	True	True	0.12
$x_{3,1}$	$x_{3,2}$	$x_{3,3}$	$x_{3,4}$	False	False	0.25
$x_{4,1}$	$x_{4,2}$	$x_{4,3}$	$x_{4,4}$	True	True	0.25
$x_{5,1}$	$x_{5,2}$	$x_{5,3}$	$x_{5,4}$	False	False	0.12
$x_{6,1}$	$x_{6,2}$	$x_{6,3}$	$x_{6,4}$	False	False	0.12

$$Total\ Error = \frac{n_errors}{n_samples} = \frac{1}{6}$$

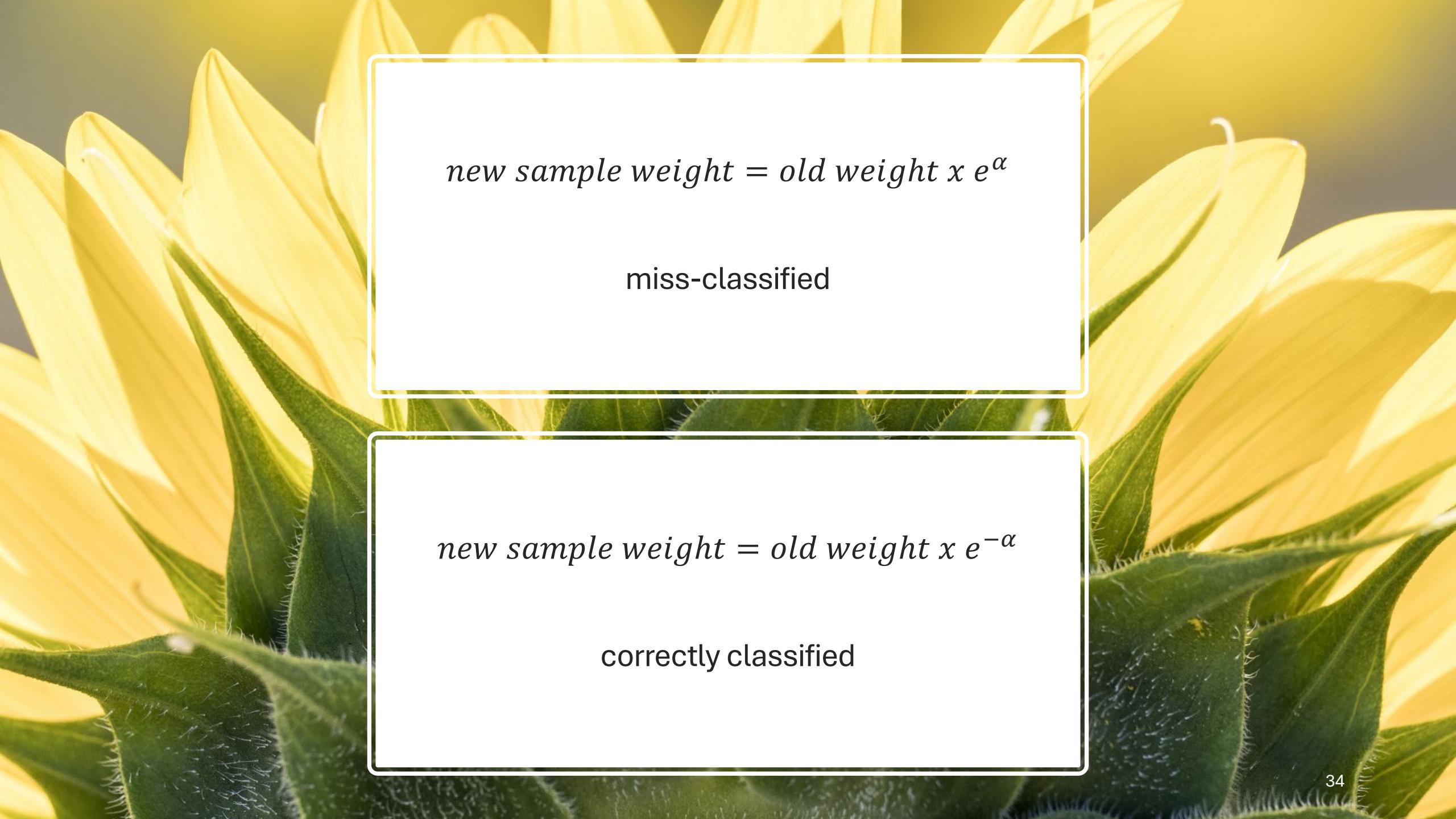
Iteration2 – Calculate influence

x_1	x_2	x_3	x_4	y	$h_2(x) \Rightarrow \hat{y}$	weight
$x_{1,1}$	$x_{1,2}$	$x_{1,3}$	$x_{1,4}$	True	False	0.12
$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	$x_{2,4}$	True	True	0.12
$x_{3,1}$	$x_{3,2}$	$x_{3,3}$	$x_{3,4}$	False	False	0.25
$x_{4,1}$	$x_{4,2}$	$x_{4,3}$	$x_{4,4}$	True	True	0.25
$x_{5,1}$	$x_{5,2}$	$x_{5,3}$	$x_{5,4}$	False	False	0.12
$x_{6,1}$	$x_{6,2}$	$x_{6,3}$	$x_{6,4}$	False	False	0.12

$$Influence = \alpha_2 = \frac{1}{2} \log \left(\frac{1 - Total\ Error}{Total\ Error} \right) = \frac{1}{2} \log \frac{\left(1 - \frac{1}{6}\right)}{\frac{1}{6}} = 0.80$$

Iteration2 – update sample weights

x_1	x_2	x_3	x_4	y	$h_2(x) \Rightarrow \hat{y}$	weight
$x_{1,1}$	$x_{1,2}$	$x_{1,3}$	$x_{1,4}$	True	False	$0.12e^{0.80}$
$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	$x_{2,4}$	True	True	$0.12e^{-0.80}$
$x_{3,1}$	$x_{3,2}$	$x_{3,3}$	$x_{3,4}$	False	False	$0.25e^{-0.80}$
$x_{4,1}$	$x_{4,2}$	$x_{4,3}$	$x_{4,4}$	True	True	$0.25e^{-0.80}$
$x_{5,1}$	$x_{5,2}$	$x_{5,3}$	$x_{5,4}$	False	False	$0.12e^{-0.80}$
$x_{6,1}$	$x_{6,2}$	$x_{6,3}$	$x_{6,4}$	False	False	$0.12e^{-0.80}$



$$\text{new sample weight} = \text{old weight} \times e^\alpha$$

miss-classified

$$\text{new sample weight} = \text{old weight} \times e^{-\alpha}$$

correctly classified

Iteration2 – update sample weights

x_1	x_2	x_3	x_4	y	$h_2(x) \Rightarrow \hat{y}$	weight
$x_{1,1}$	$x_{1,2}$	$x_{1,3}$	$x_{1,4}$	True	False	0.27
$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	$x_{2,4}$	True	True	0.05
$x_{3,1}$	$x_{3,2}$	$x_{3,3}$	$x_{3,4}$	False	False	0.17
$x_{4,1}$	$x_{4,2}$	$x_{4,3}$	$x_{4,4}$	True	True	0.17
$x_{5,1}$	$x_{5,2}$	$x_{5,3}$	$x_{5,4}$	False	False	0.05
$x_{6,1}$	$x_{6,2}$	$x_{6,3}$	$x_{6,4}$	False	False	0.05

Iteration2 – Normalize sample weights

x_1	x_2	x_3	x_4	y	$h_2(x) \Rightarrow \hat{y}$	weight
$x_{1,1}$	$x_{1,2}$	$x_{1,3}$	$x_{1,4}$	<i>True</i>	<i>False</i>	0.42
$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	$x_{2,4}$	<i>True</i>	<i>True</i>	0.08
$x_{3,1}$	$x_{3,2}$	$x_{3,3}$	$x_{3,4}$	<i>False</i>	<i>False</i>	0.25
$x_{4,1}$	$x_{4,2}$	$x_{4,3}$	$x_{4,4}$	<i>True</i>	<i>True</i>	0.25
$x_{5,1}$	$x_{5,2}$	$x_{5,3}$	$x_{5,4}$	<i>False</i>	<i>False</i>	0.08
$x_{6,1}$	$x_{6,2}$	$x_{6,3}$	$x_{6,4}$	<i>False</i>	<i>False</i>	0.08

$$H(x) = 0.35h_1(x) + 0.80h_2(x)$$

final prediction = $\text{sign}(H(x))$

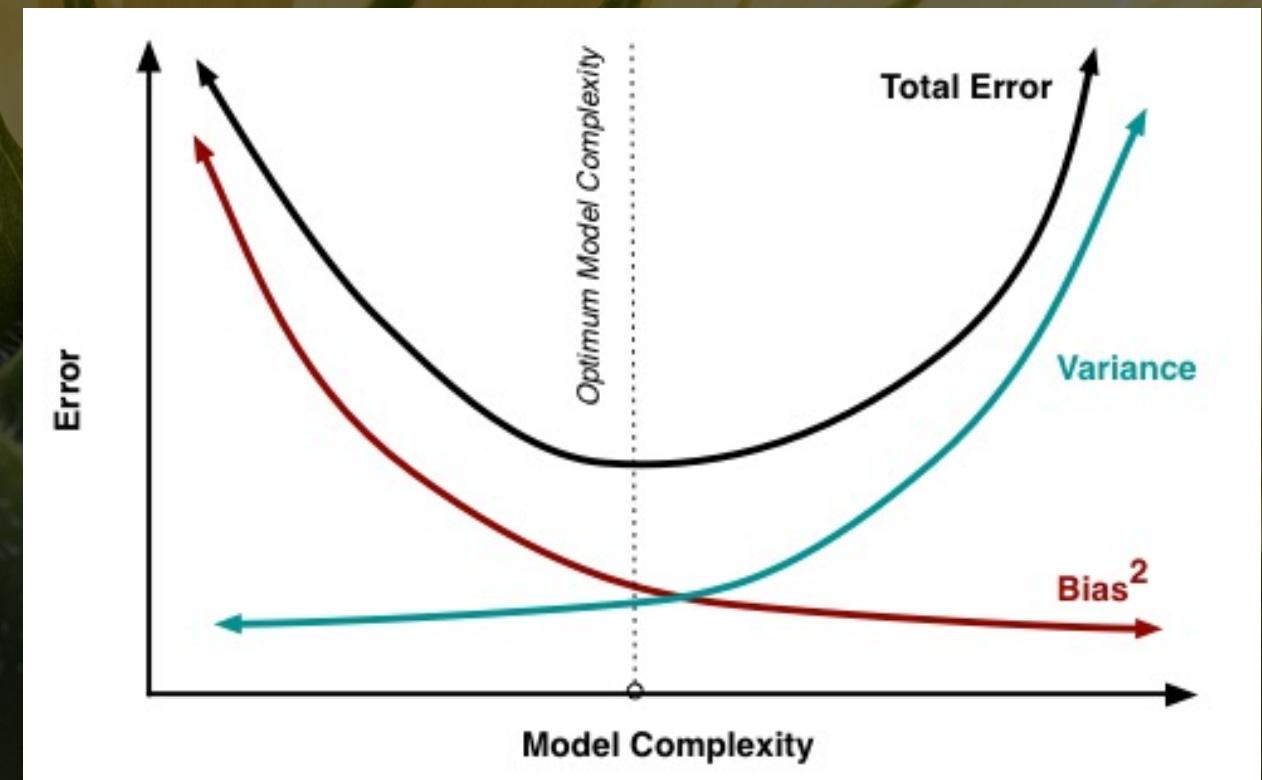
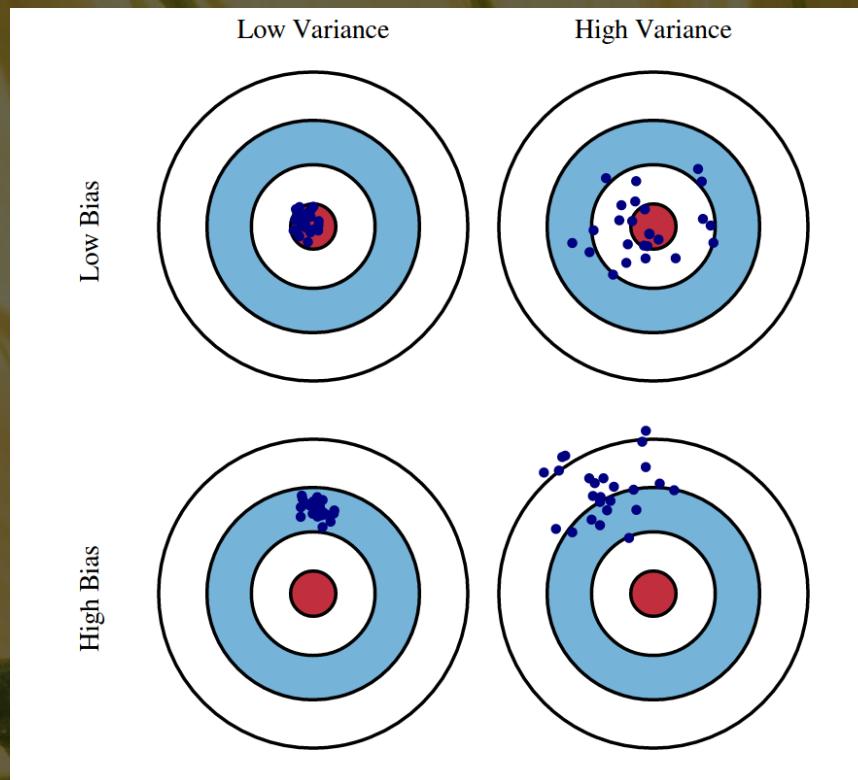


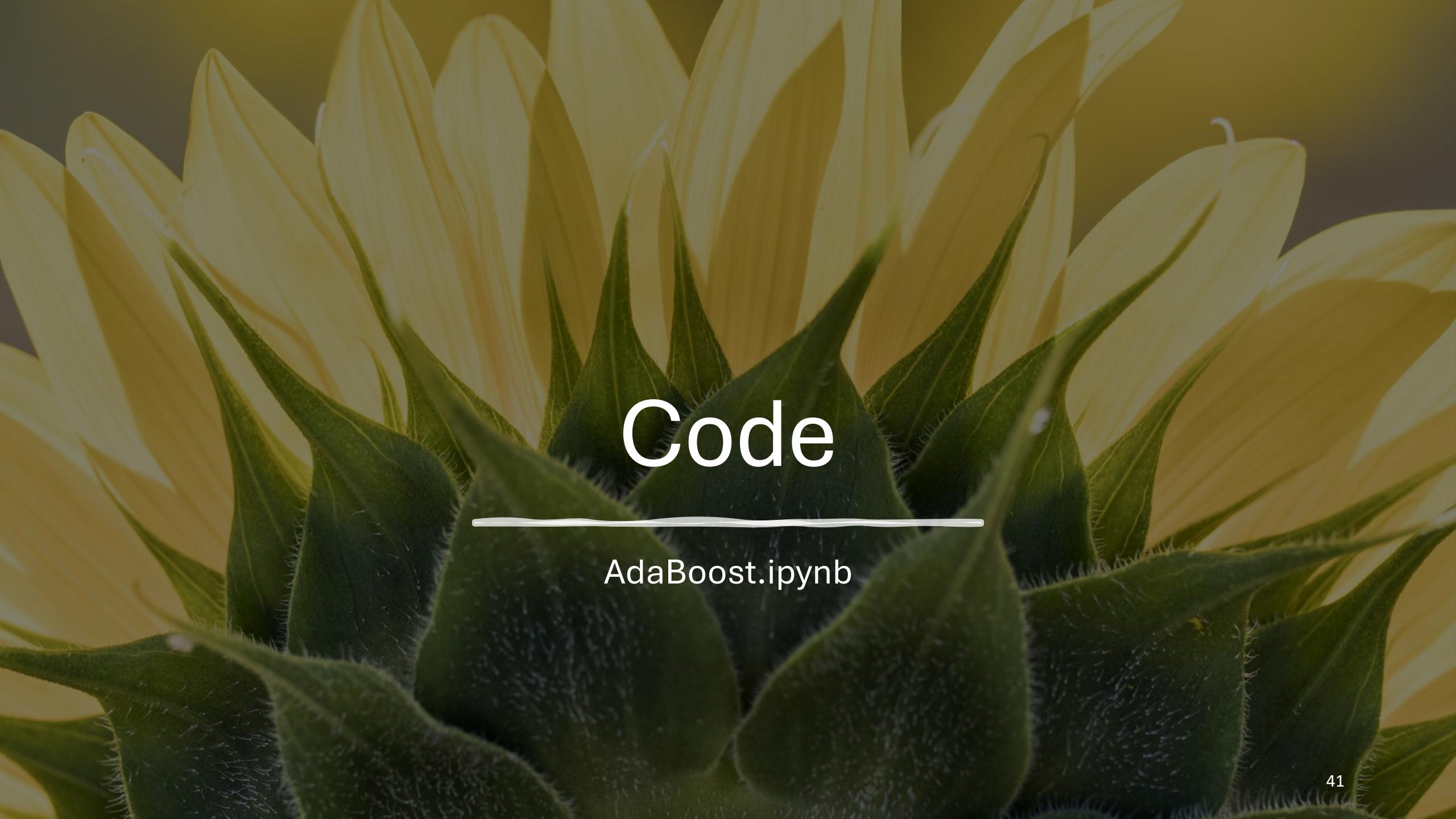
Discussion about AdaBoost

Discussion about AdaBoost

- Mechanism behind AdaBoost
- Why weak learner?
- Extension from AdaBoost
- Suitable n_iterations for AdaBoost

Discussion about AdaBoost



The background of the slide is a close-up photograph of a sunflower head. The image shows the intricate details of the yellow petals and the dark green, serrated leaves at the base. The lighting is soft, highlighting the texture of the flower.

Code

AdaBoost.ipynb

Gradient boosting

Gradient Boosting

- What is gradient boosting?
- Calculation step
- Calculation concept
- Calculation example
- Discussion about gradient boosting
- Code

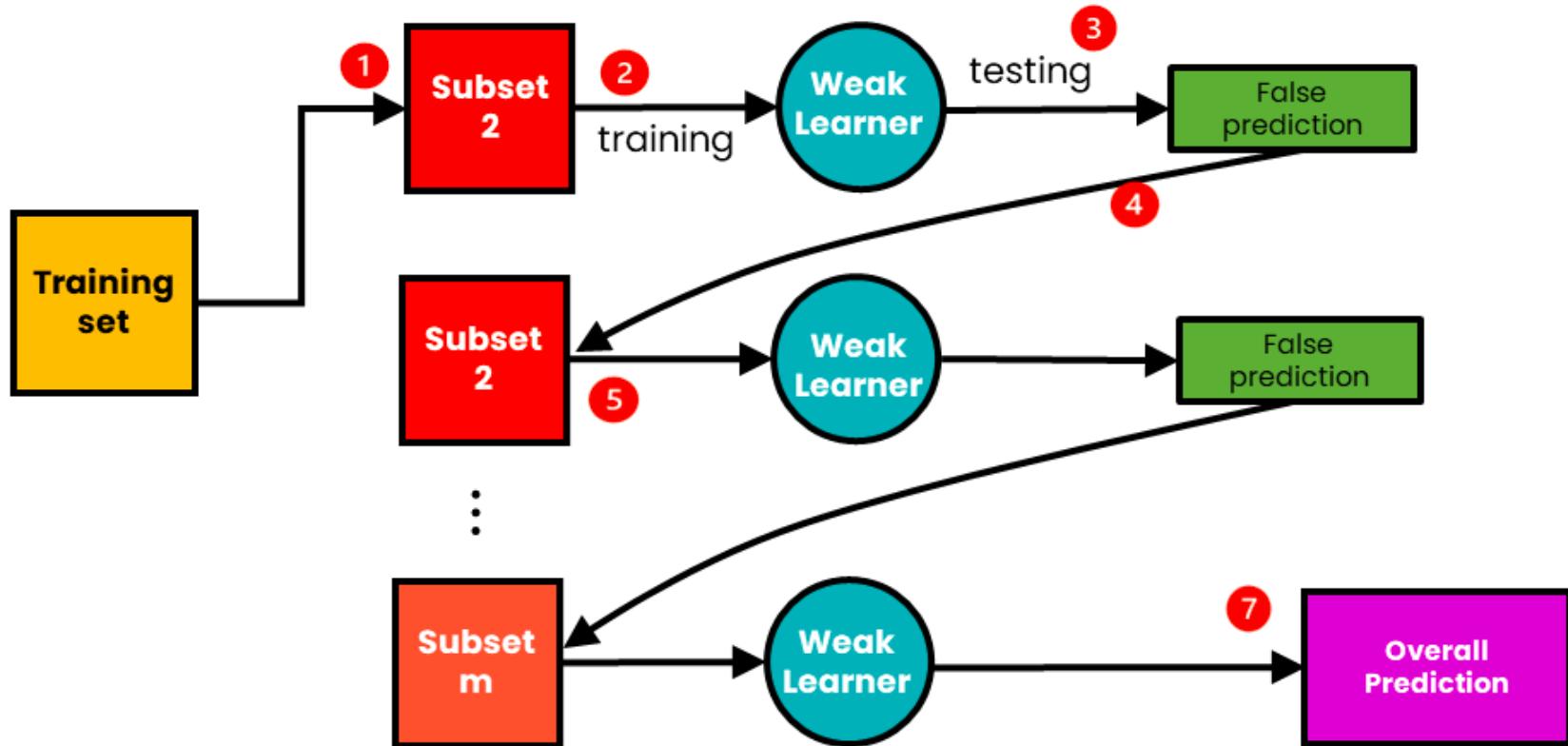
What is Gradient boosting?

Gradient Boosting for regression is a machine learning technique that constructs a predictive model by sequentially adding weak learners to minimize the errors of previous models.

Calculation step

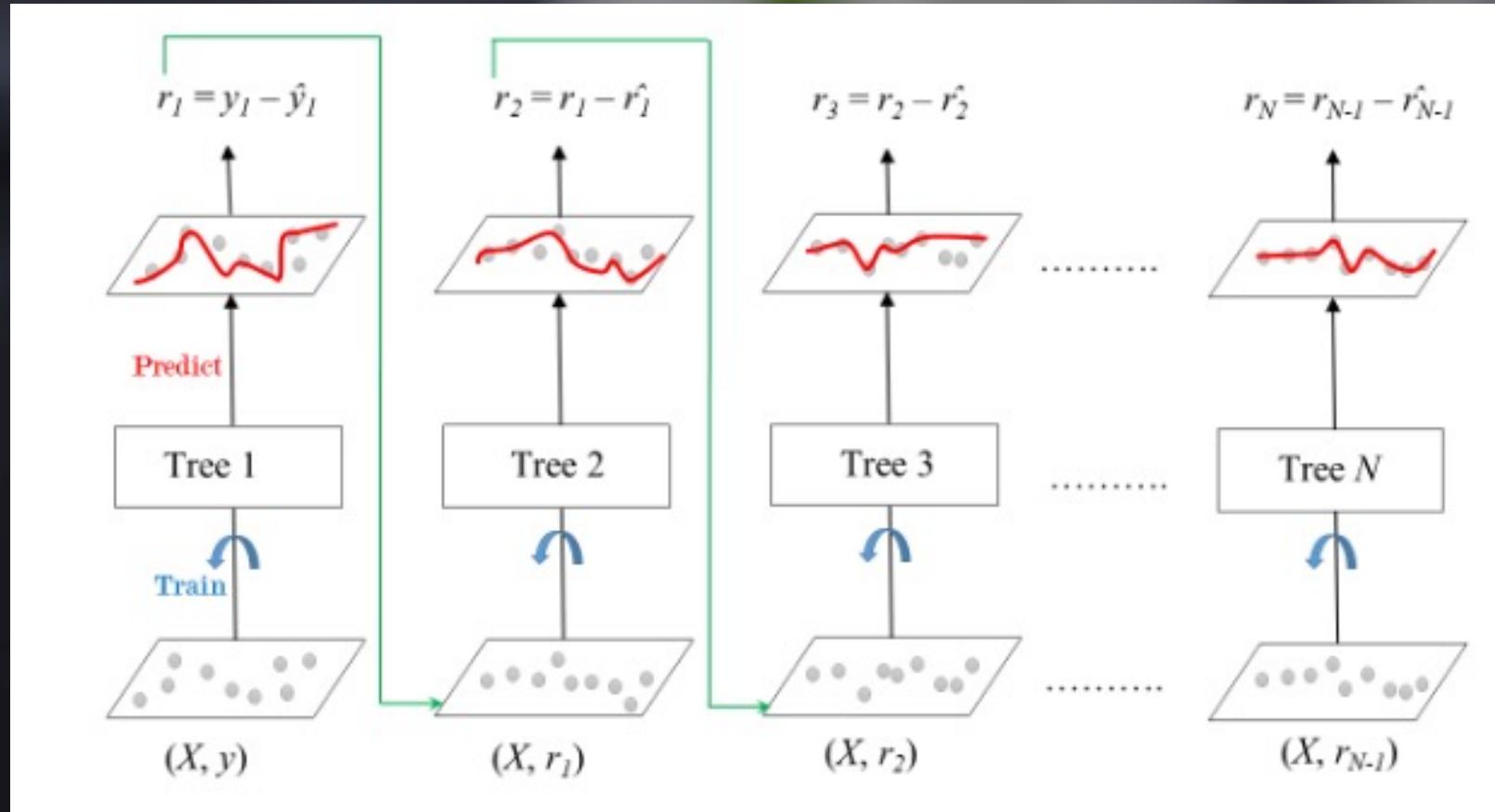
1. Initialize the model
2. Calculate pseudo residuals
3. Train weak learner
4. Update the model
5. Repeat step 2-4 until meet stopping criteria

The Process of Boosting



$$\hat{y} = F_0(x) + \sum_{m=1}^M \eta h_m(x)$$

Calculation concept



Calculation example

Initialize the model

x_1	x_2	x_3	y	$\hat{y} = F_0(x)$
$x_{1,1}$	$x_{1,2}$	$x_{1,3}$	35,000	30,000
$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	10,000	30,000
$x_{3,1}$	$x_{3,2}$	$x_{3,3}$	15,000	30,000
$x_{4,1}$	$x_{4,2}$	$x_{4,3}$	20,000	30,000
$x_{5,1}$	$x_{5,2}$	$x_{5,3}$	40,000	30,000
$x_{6,1}$	$x_{6,2}$	$x_{6,3}$	60,000	30,000

Iteration1 – calculate pseudo residuals

x_1	x_2	x_3	y	$F_0(x)$	r_1
$x_{1,1}$	$x_{1,2}$	$x_{1,3}$	35,000	30,000	5,000
$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	10,000	30,000	-20,000
$x_{3,1}$	$x_{3,2}$	$x_{3,3}$	15,000	30,000	-15,000
$x_{4,1}$	$x_{4,2}$	$x_{4,3}$	20,000	30,000	-10,000
$x_{5,1}$	$x_{5,2}$	$x_{5,3}$	40,000	30,000	10,000
$x_{6,1}$	$x_{6,2}$	$x_{6,3}$	60,000	30,000	30,000

Iteration1 – train weak learner

x_1	x_2	x_3	r_1	$h_1(x)$
$x_{1,1}$	$x_{1,2}$	$x_{1,3}$	5,000	4,800
$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	-20,000	-21,000
$x_{3,1}$	$x_{3,2}$	$x_{3,3}$	-15,000	-16,000
$x_{4,1}$	$x_{4,2}$	$x_{4,3}$	-10,000	-10,000
$x_{5,1}$	$x_{5,2}$	$x_{5,3}$	10,000	11,000
$x_{6,1}$	$x_{6,2}$	$x_{6,3}$	30,000	29,000

Iteration1 – update the model

กำหนดให้ $\eta = 0.1$

$$\text{จาก } \hat{y} = F_0(x) + \sum_{m=1}^M \eta h_m(x)$$

ณ iteration1 จะได้ว่า

$$\hat{y} = F_0(x) + 0.1h_m(x)$$

Iteration1 – update the model

x_1	x_2	x_3	y	$\hat{y} = F_0(x) + 0.1h_1(x)$
$x_{1,1}$	$x_{1,2}$	$x_{1,3}$	35,000	30,480
$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	10,000	27,900
$x_{3,1}$	$x_{3,2}$	$x_{3,3}$	15,000	28,400
$x_{4,1}$	$x_{4,2}$	$x_{4,3}$	20,000	29,000
$x_{5,1}$	$x_{5,2}$	$x_{5,3}$	40,000	31,000
$x_{6,1}$	$x_{6,2}$	$x_{6,3}$	60,000	32,900

A close-up photograph of a single, vibrant red rose. The rose is positioned in the lower right quadrant of the frame, resting on a dark, textured surface that appears to be a piece of wood or stone. The background is blurred, showing hints of green foliage and other flowers, creating a soft, bokeh effect.

iteration2

Iteration2 - calculate pseudo residuals

x_1	x_2	x_3	y	$\hat{y} = F_0(x) + 0.1h_1(x)$	r_2
$x_{1,1}$	$x_{1,2}$	$x_{1,3}$	35,000	30,480	4,520
$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	10,000	27,900	-17,900
$x_{3,1}$	$x_{3,2}$	$x_{3,3}$	15,000	28,400	-13,400
$x_{4,1}$	$x_{4,2}$	$x_{4,3}$	20,000	29,000	-9,000
$x_{5,1}$	$x_{5,2}$	$x_{5,3}$	40,000	31,000	9,000
$x_{6,1}$	$x_{6,2}$	$x_{6,3}$	60,000	32,900	27,100

Iteration2 – train weak learner

x_1	x_2	x_3	r_2	$h_1(x)$
$x_{1,1}$	$x_{1,2}$	$x_{1,3}$	4,520	4,500
$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	-17,900	-18,000
$x_{3,1}$	$x_{3,2}$	$x_{3,3}$	-13,400	-13,000
$x_{4,1}$	$x_{4,2}$	$x_{4,3}$	-9,000	-9,000
$x_{5,1}$	$x_{5,2}$	$x_{5,3}$	9,000	10,000
$x_{6,1}$	$x_{6,2}$	$x_{6,3}$	27,100	27,000

Iteration2 – update the model

กำหนดให้ $\eta = 0.1$

$$\text{จาก } \hat{y} = F_0(x) + \sum_{m=1}^M \eta h_m(x)$$

ณ iteration1 จะได้ว่า

$$\hat{y} = F_0(x) + 0.1h_1(x) + 0.1h_2(x)$$

Iteration2 – update the model

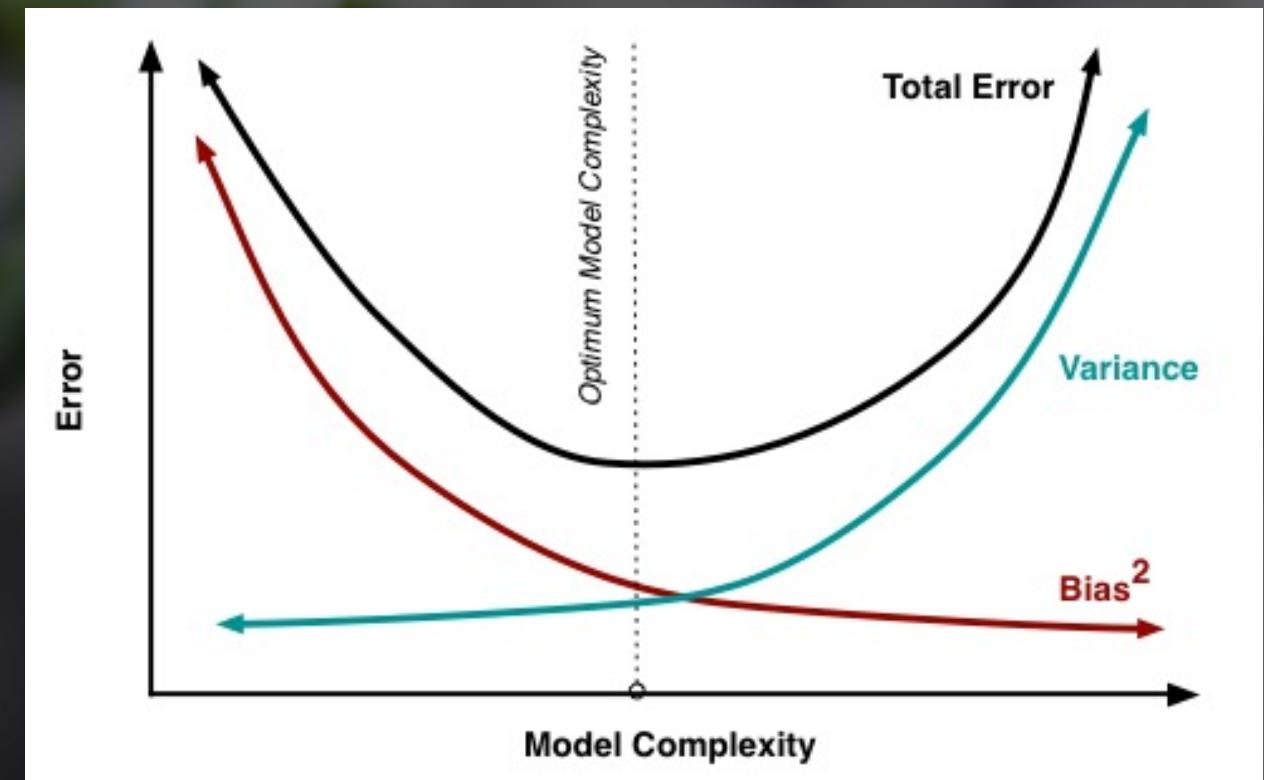
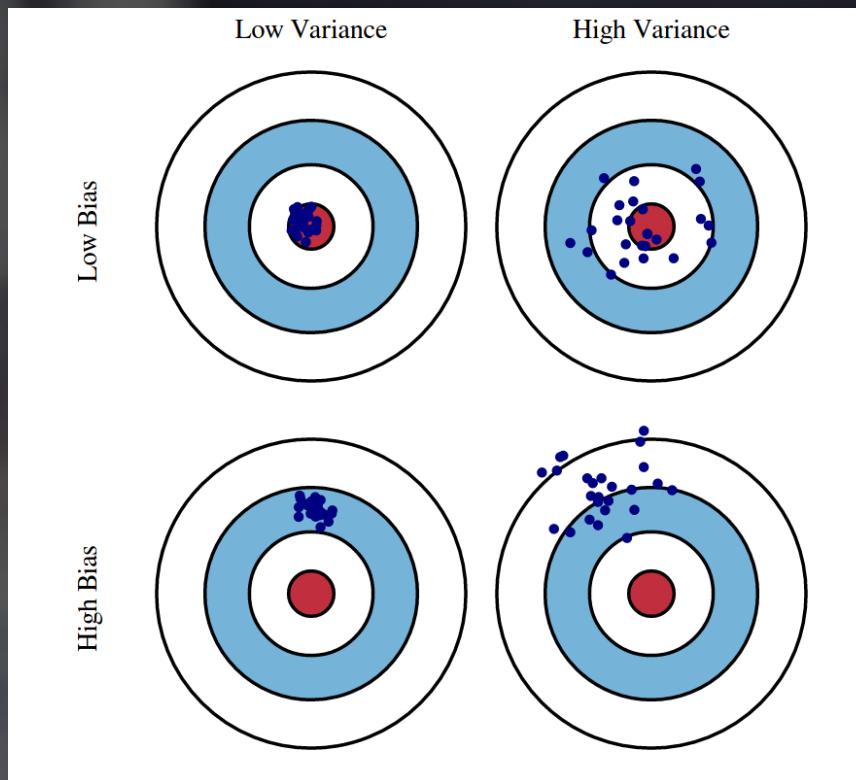
x_1	x_2	x_3	y	$\hat{y} = F_0(x) + 0.1h_1(x) + 0.1h_2(x)$
$x_{1,1}$	$x_{1,2}$	$x_{1,3}$	35,000	30,930
$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	10,000	26,100
$x_{3,1}$	$x_{3,2}$	$x_{3,3}$	15,000	27,100
$x_{4,1}$	$x_{4,2}$	$x_{4,3}$	20,000	28,100
$x_{5,1}$	$x_{5,2}$	$x_{5,3}$	40,000	32,000
$x_{6,1}$	$x_{6,2}$	$x_{6,3}$	60,000	35,600

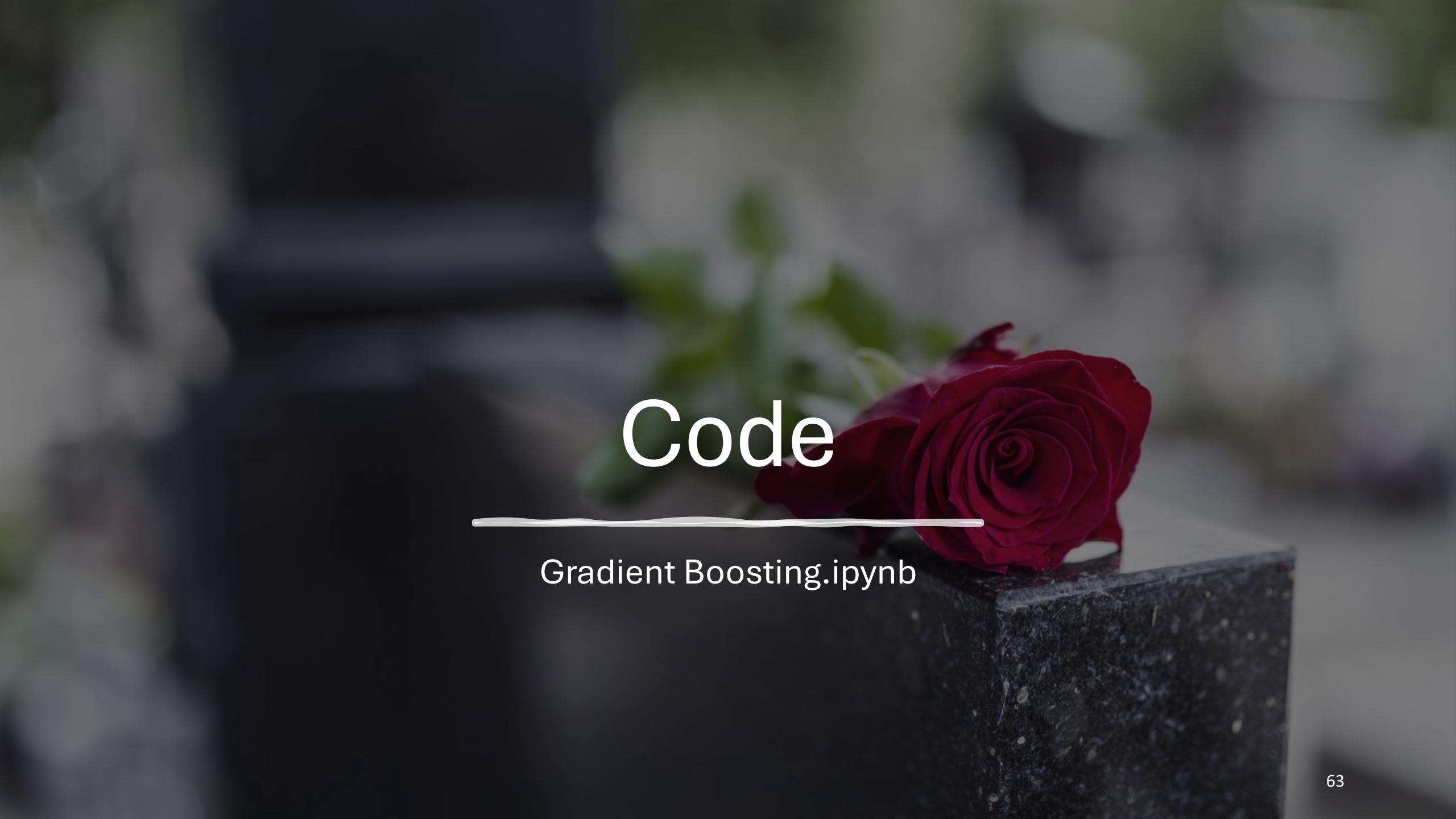
Discussion about gradient boosting

Discussion about gradient boosting

- Why don't we fit pseudo residuals with strong learner?
- What is the intuition behind gradient boosting?
- Why weak learner?
- Suitable n_iterations for gradient boosting

Discussion about gradient boosting



A close-up photograph of a single red rose with its stem and some green leaves. The rose is positioned on the right side of the frame, resting on a dark, textured surface that appears to be a book or a piece of wood. The background is blurred, showing more of the same red roses and green foliage.

Code

Gradient Boosting.ipynb



LightGBM, XGBoost, CatBoost

LightGBM, XGBoost, CatBoost

Gradient boosting

- + regularization
- + approximate greedy algorithm
- + weighted quantile sketch
- + sparsity-aware split finding
- + parallel learning
- + cache-aware access
- + blocks for out-of-core computation
- + etc

Further reading

- AdaBoost for regression
- Gradient boost for classification

QUESTION & ANSWER



Reference

- <https://www.analyticsvidhya.com/blog/2023/01/ensemble-learning-methods-bagging-boosting-and-stacking/>
- <https://www.analyticsvidhya.com/blog/2021/09/adaboost-algorithm-a-complete-guide-for-beginners/>
- <https://www.geeksforgeeks.org/implementing-the-adaboost-algorithm-from-scratch/>
- <https://www.geeksforgeeks.org/ml-gradient-boosting/>
- <https://www.youtube.com/watch?v=OtD8wVaFm6E>

