SVM入门理解

Reference :

- **StatQuest with Josh Starmer:Support Vector Machines**
- **支持向量机通俗导论（理解SVM的三层境界）https://blog.csdn.net/v_july_v/article/details/7624837**

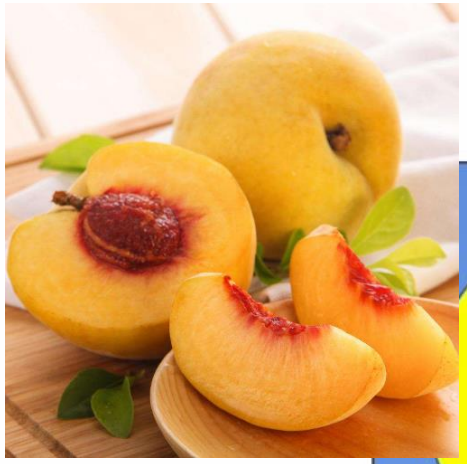
内容：

- 最大间隔分类器
  - 函数距离
  - 几何距离
  - 最大间隔分类
- 支持向量分类器
  - 软间隔-松弛变量
- 支持向量机
  - 对偶问题
  - 核函数
  - 多项式核

0.引子：水果分类

- A fruit is either
  - small or large and
  - yellow or purple.
- A small yellow fruit is an unripe plum.It is not good to eat.黄色小水果是未成熟的李子，不好吃。
- A small purple fruit is a ripe plum.It is good to eat.当紫色小水果是成熟的李子，很好吃。
- A large yellow fruit is a ripe peach. It is good to eat.黄色大水果是成熟的桃子，很好吃。
- A large purple fruit is a rotten peach. It is not good to eat. 紫色大水果是腐烂的桃子，不好吃。

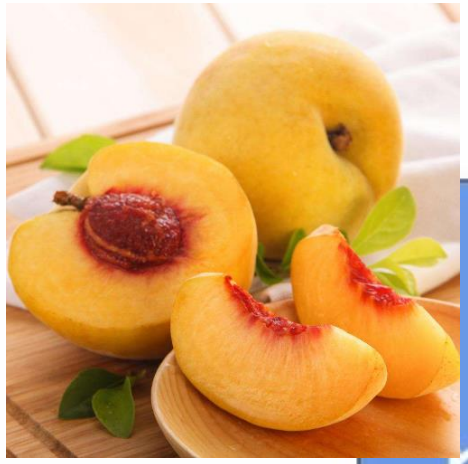
large



yellow

purple

small



large

ripe peach

rotten peach

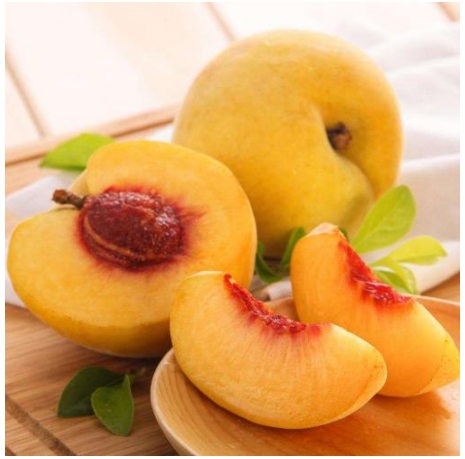
yellow

purple

unripe plum

ripe plum

small



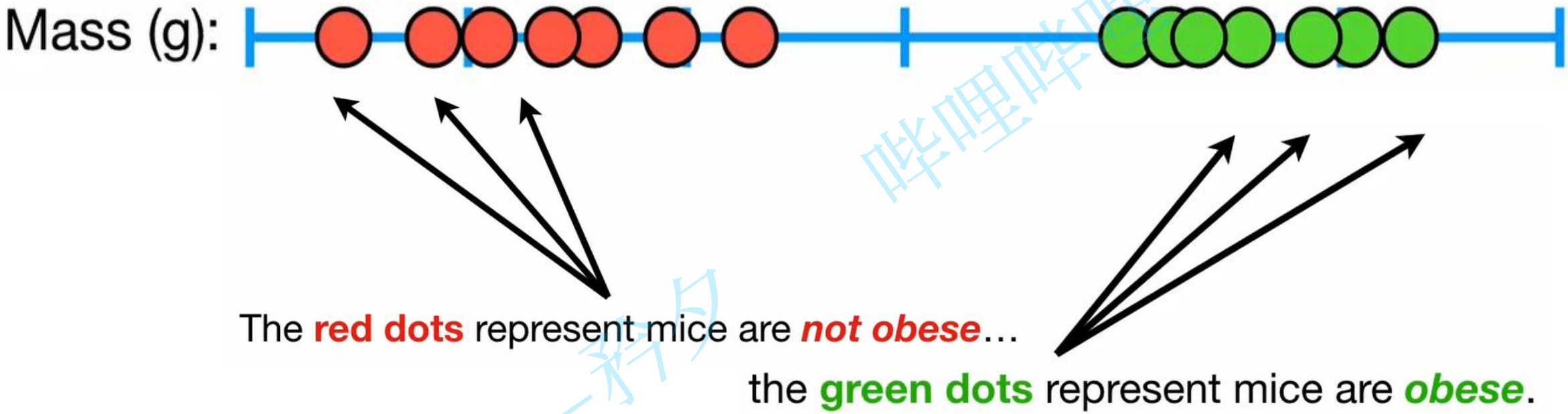
large

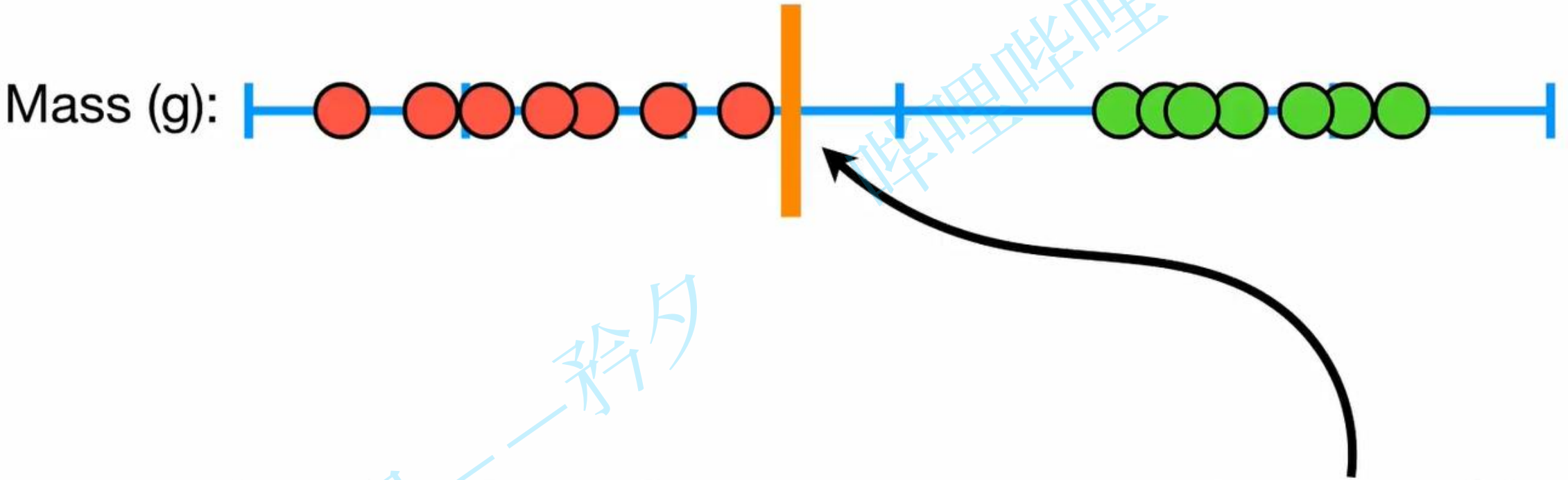
yellow

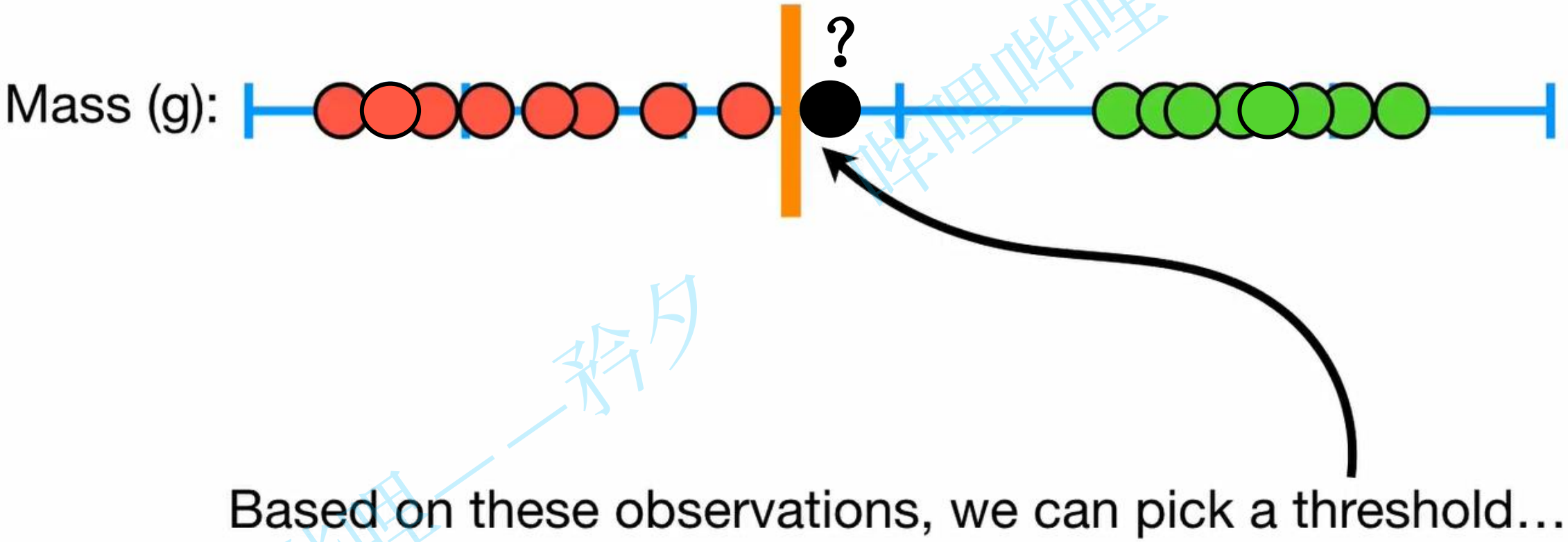
purple

small

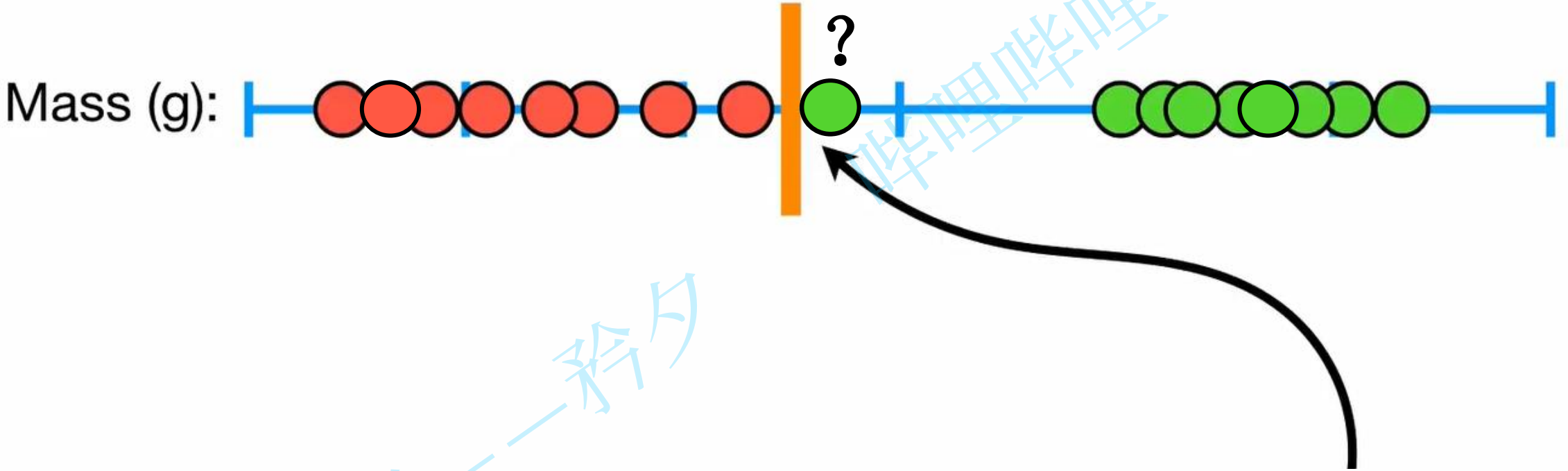
1.最大间隔分类器

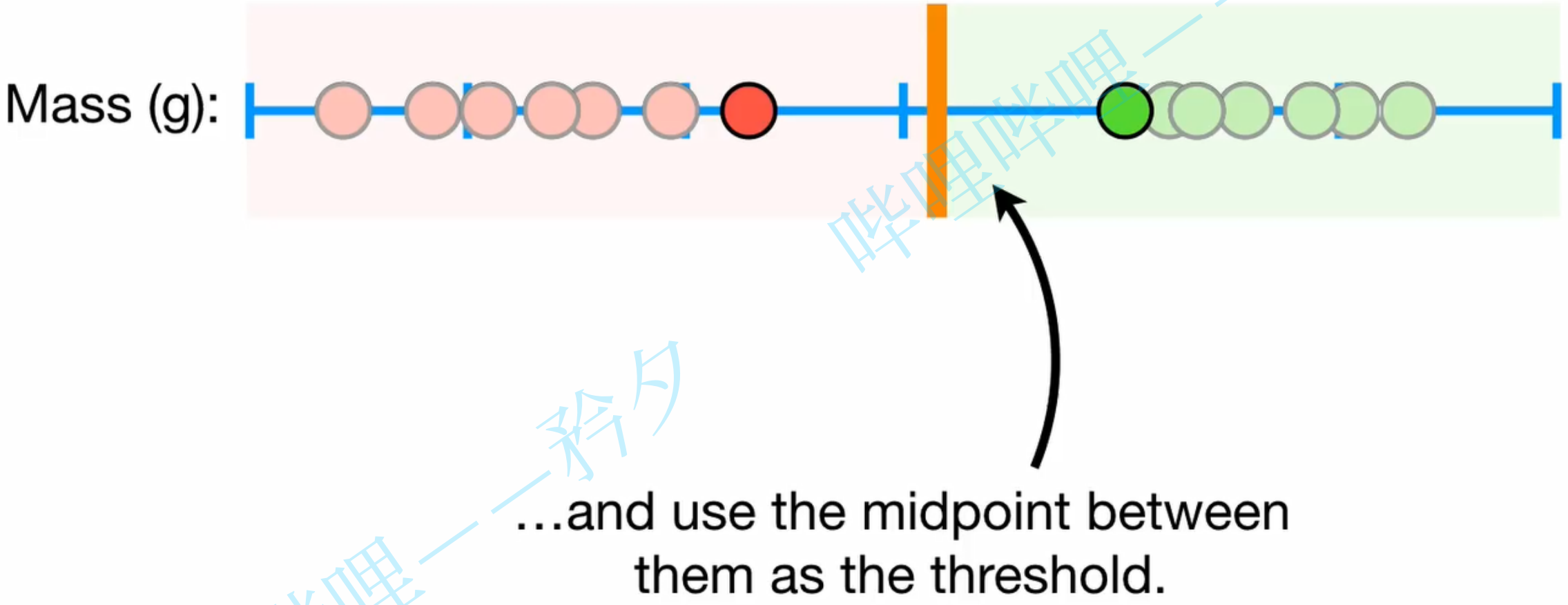

Mass (g):
The red dots represent mice are not obese…
the green dots represent mice are obese.

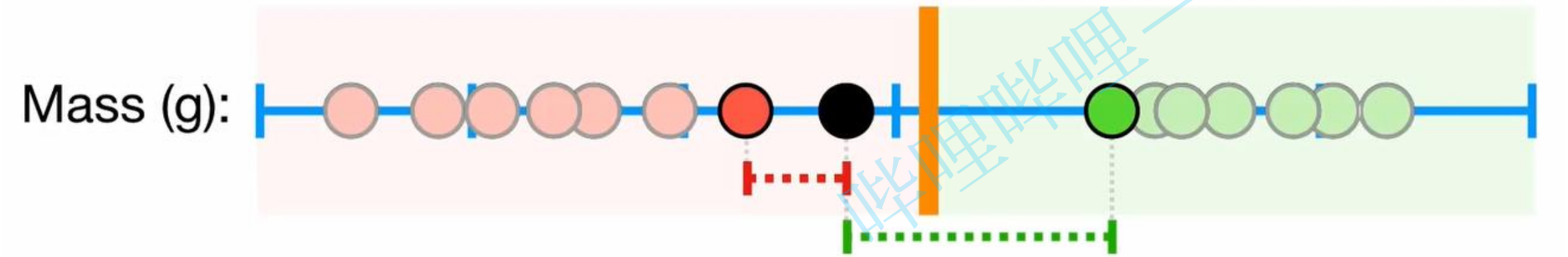

Mass (g):
Based on these observations, we can pick a threshold…


Mass (g):
?
Based on these observations, we can pick a threshold…

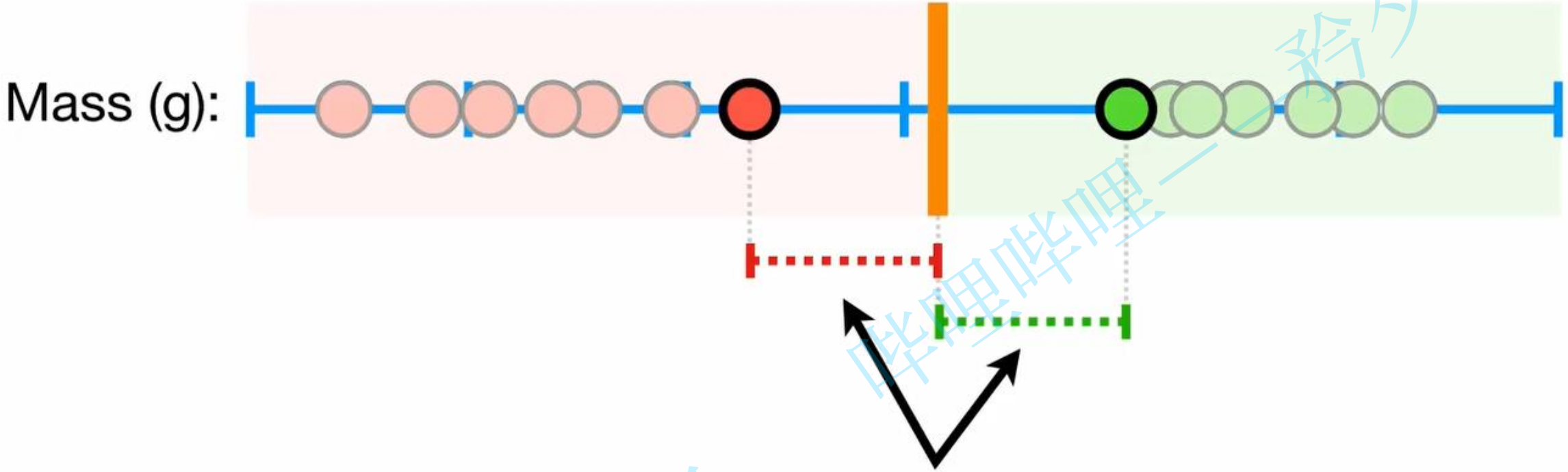
但是它离左边更近，这样分合理吗？如何优化？




Mass (g):
…and use the midpoint between them as the threshold.

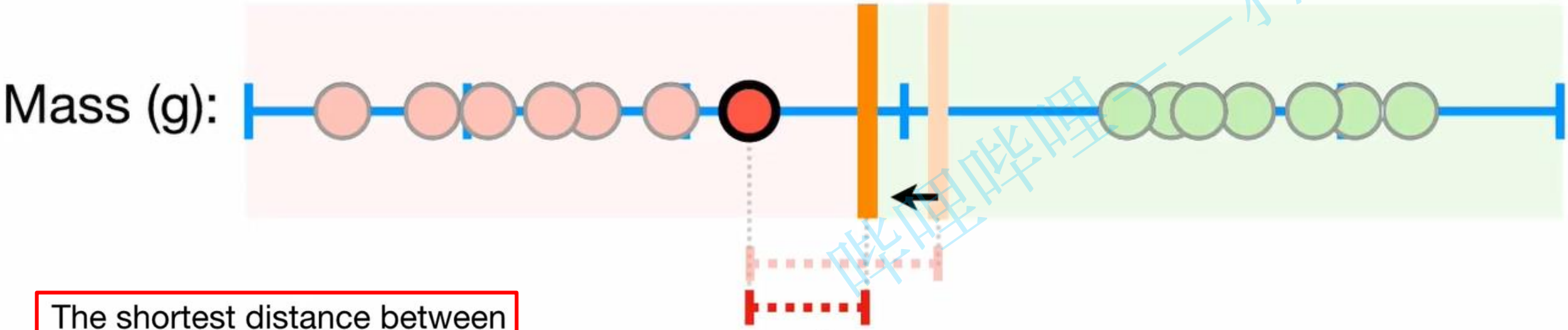


待分类的黑球落在阈值左边，也离红色更近，分为红球类，
表示不肥胖的样本



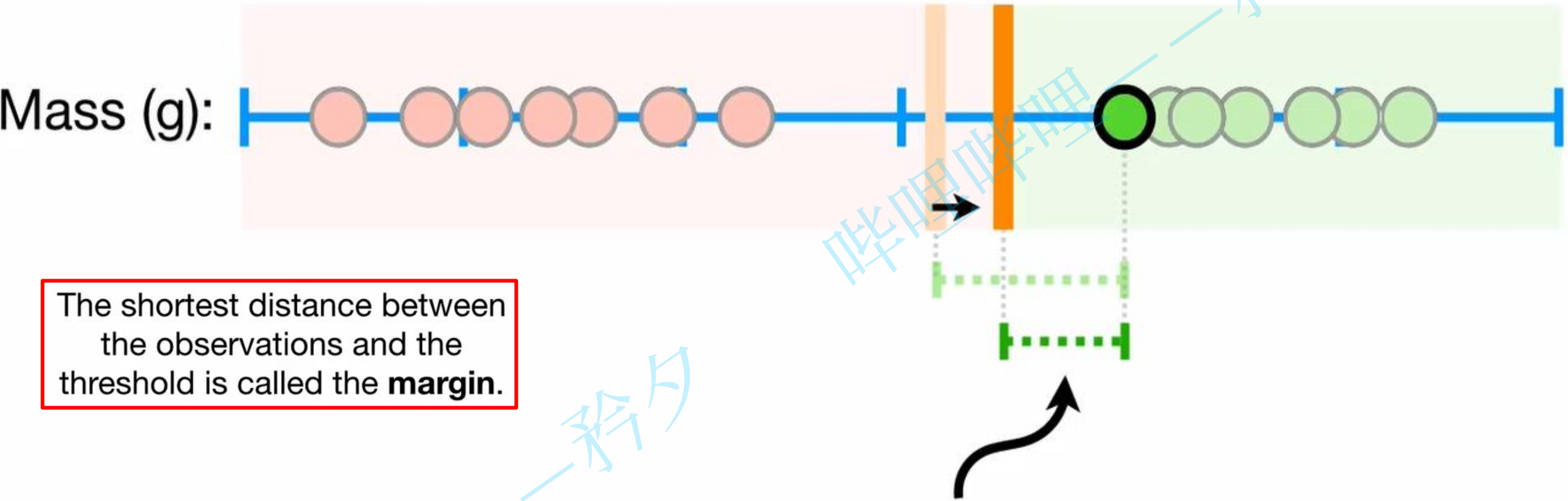
The shortest distance between the observations and the threshold is called the **margin**.

When the threshold is halfway between the two observations, the **margin** is as large as it can be.



The shortest distance between the observations and the threshold is called the **margin**.

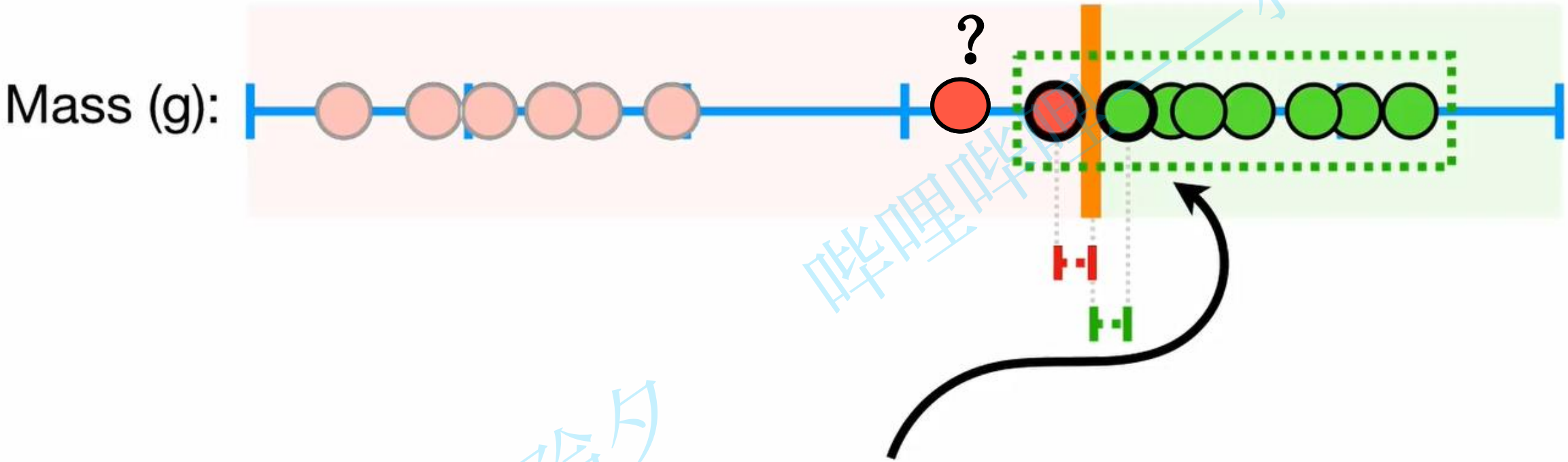
…then the distance between the threshold and the observation that is ***not obese*** would be smaller…



The shortest distance between the observations and the threshold is called the **margin**.

…then the distance between the ***obese*** observation and the threshold would get smaller…

最大间隔分类器



Hard Margin
硬间隔

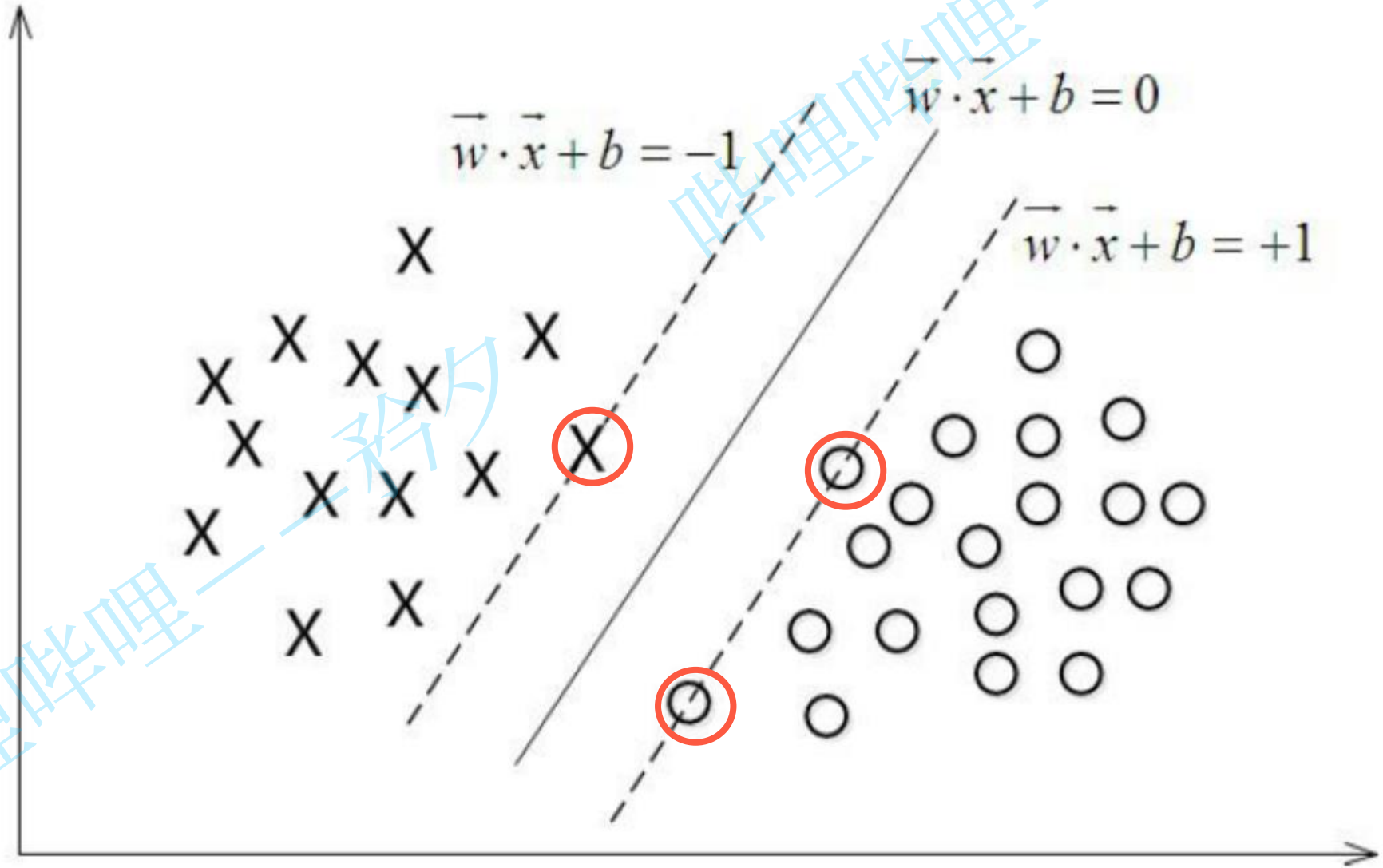
In this case, the **Maximum Margin Classifier** would be super close to the ***obese*** observations…

1.最大间隔分类器-数学补充

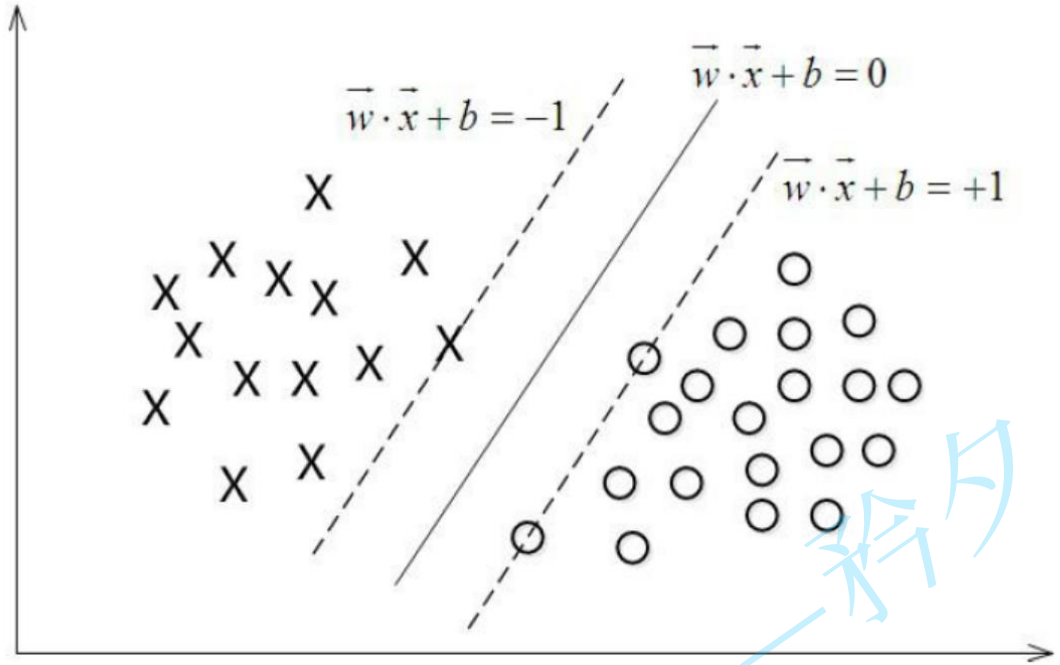
- 支持向量
- 函数间隔
- 几何间隔
- 最大间隔分类器

支持向量

$$f(x) = w^T x + b$$



函数间隔



$$f(x) = w^T x + b$$

点到分类线的距离：$|w^T x + b|$

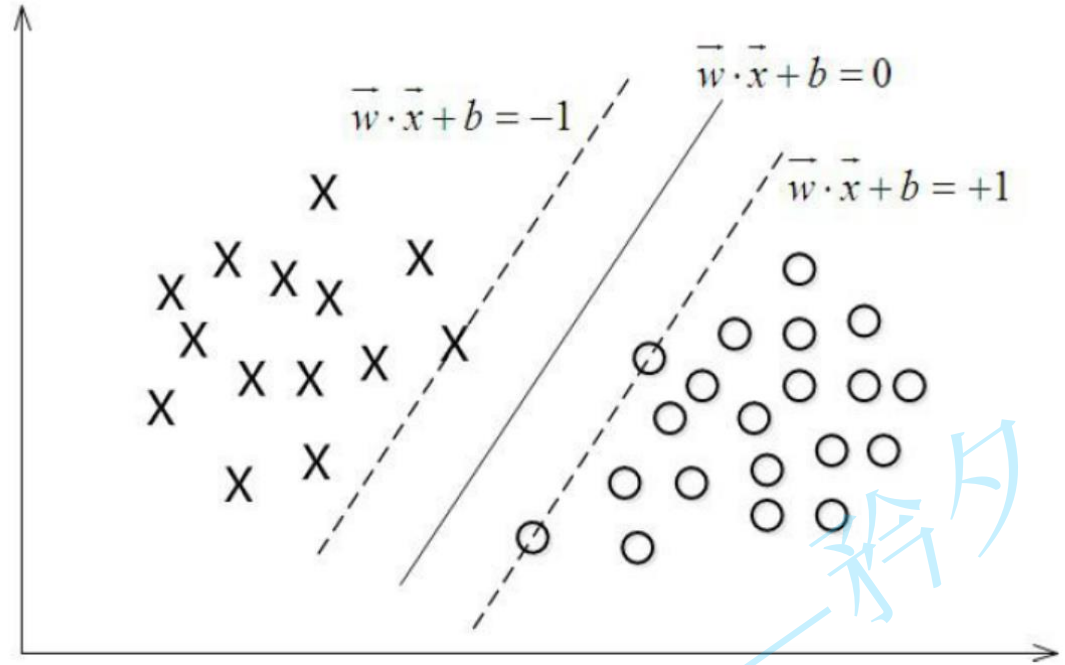
函数间隔表示为：

$$\hat{\gamma} = y(w^T x + b) = y f(x) = |f(x)|$$

训练样本的函数间隔：

$$\hat{\gamma} = \min \hat{\gamma}_i, \quad (i = 1, \ldots, n)$$

几何间隔



$$f(x) = w^T x + b$$

点到分类线的距离：$|w^T x + b|$

函数间隔表示为：

$$\hat{\gamma} = y(w^T x + b) = y f(x) = |f(x)|$$

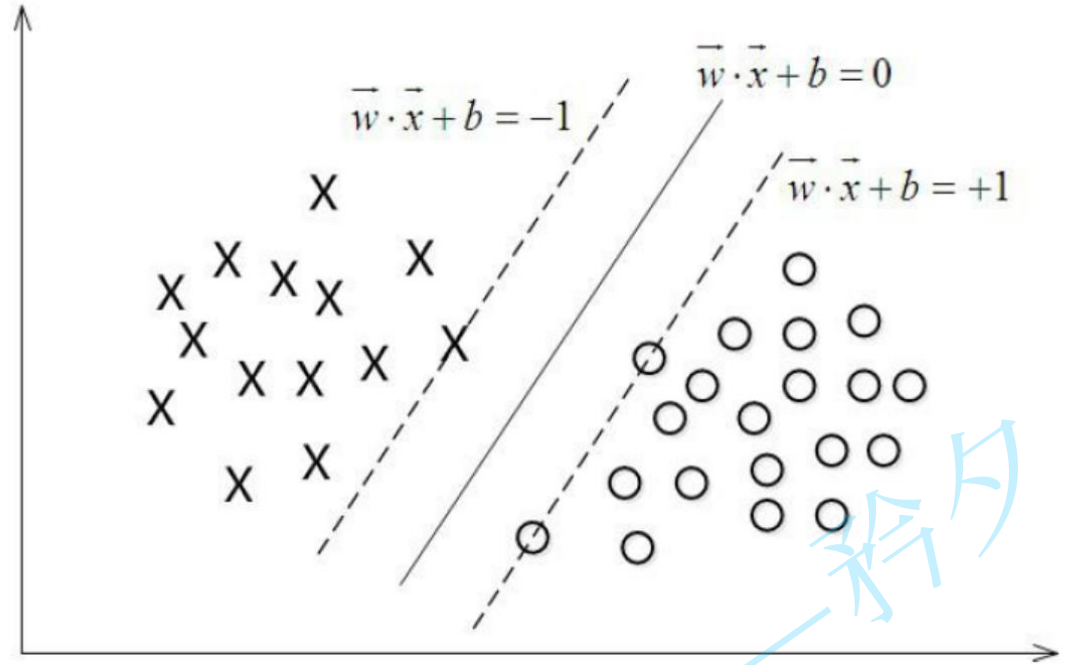
训练样本的函数间隔：

$$\hat{\gamma} = \min \hat{\gamma}_i, \quad (i = 1, ..., n)$$

几何间隔表示为：$\tilde{\gamma} = \dfrac{\hat{\gamma}}{\|w\|}$

最大间隔分类器

$$f(x) = w^T x + b$$



点到分类线的距离：$|w^T x + b|$

训练样本的函数间隔：

$$\hat{\gamma} = \min \hat{\gamma}_i, \quad (i = 1, ..., n)$$

几何间隔表示为：$\tilde{\gamma} = \dfrac{\hat{\gamma}}{\|w\|}$

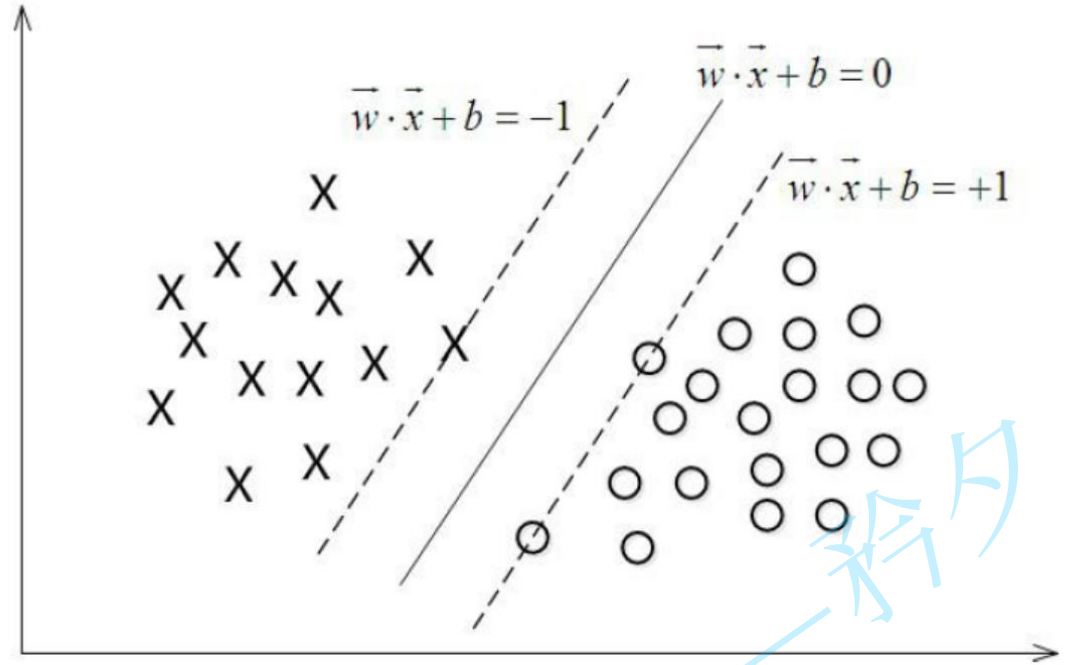
最大间隔分类器的目标函数：

$$\max \tilde{\gamma}$$

$$y_i(w^T x_i + b) = \hat{\gamma}_i \geq \hat{\gamma}, \quad i = 1, ..., n$$

最大间隔分类器

$$f(x) = w^T x + b$$



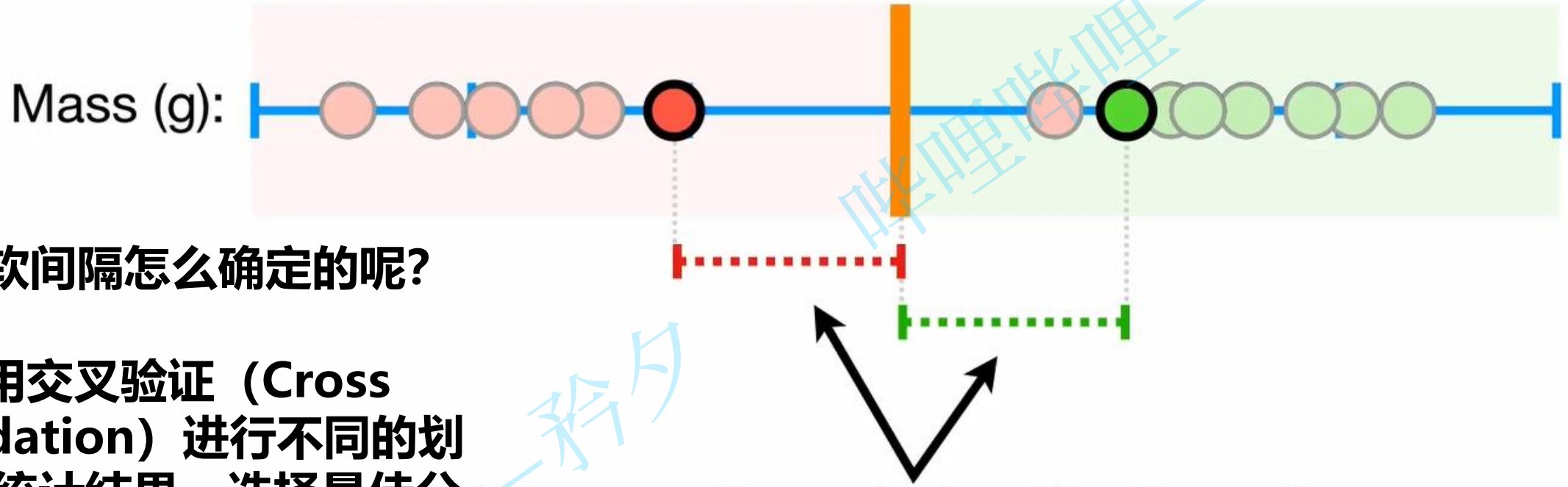
最大间隔分类器的目标函数：

$$\max \tilde{\gamma}$$
$$y_i(w^T x_i + b) = \hat{\gamma}_i \geq \hat{\gamma}, \quad i = 1, \ldots, n$$

令函数间隔=1，目标函数变为：

$$\max \frac{1}{\|w\|}, \quad s.t. \quad y_i(w^T x_i + b) \geq 1, i = 1, \ldots, n$$

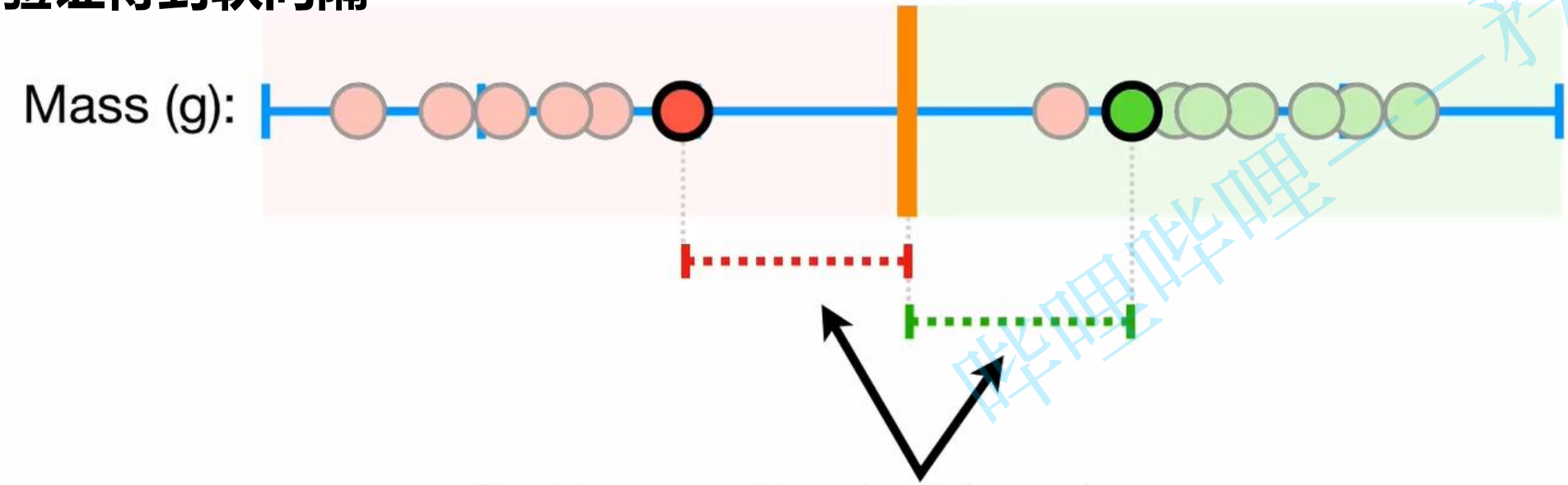
2.支持向量分类器



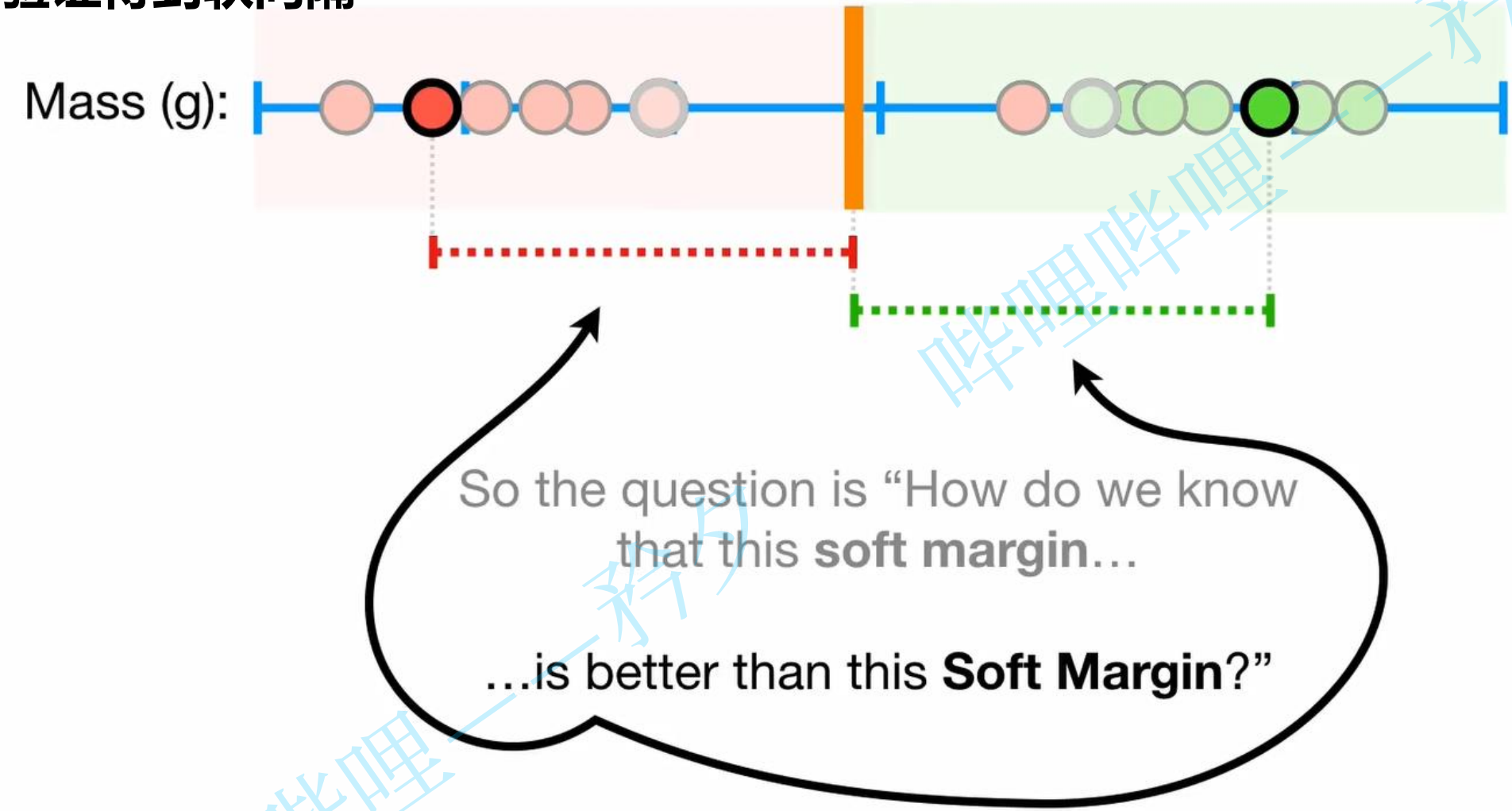
Q：软间隔怎么确定的呢?

A：用交叉验证（Cross Validation）进行不同的划分，统计结果，选择最佳分类器

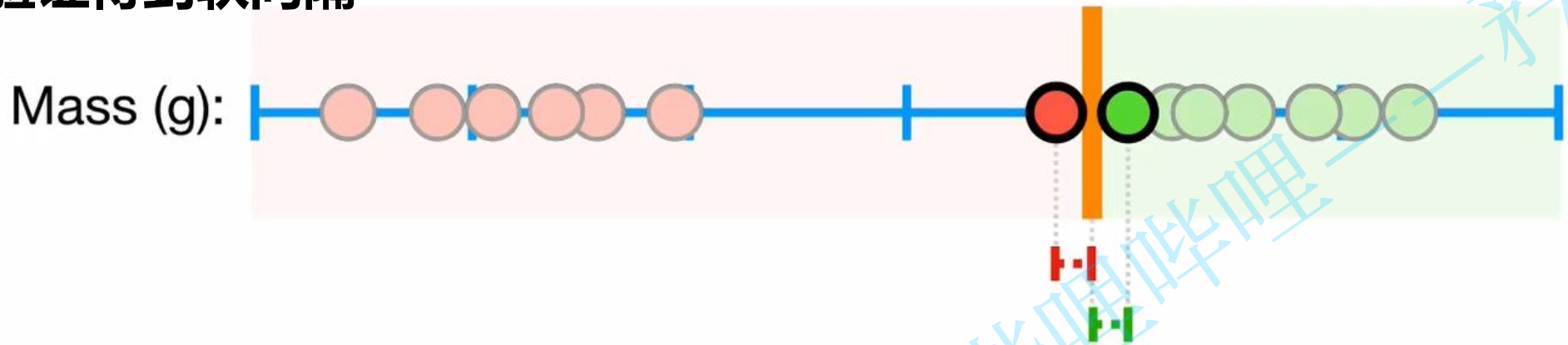
交叉验证得到软间隔



交叉验证得到软间隔

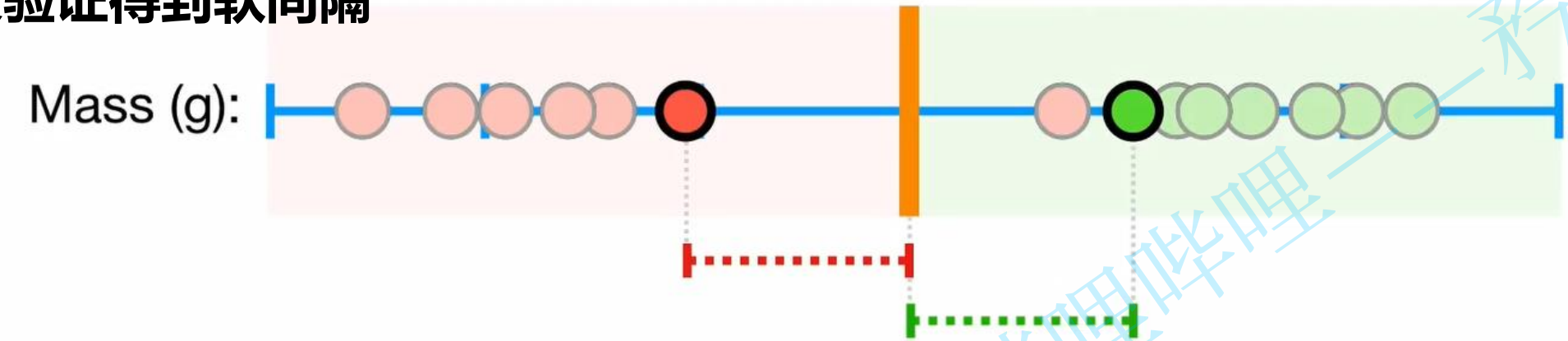


交叉验证得到软间隔



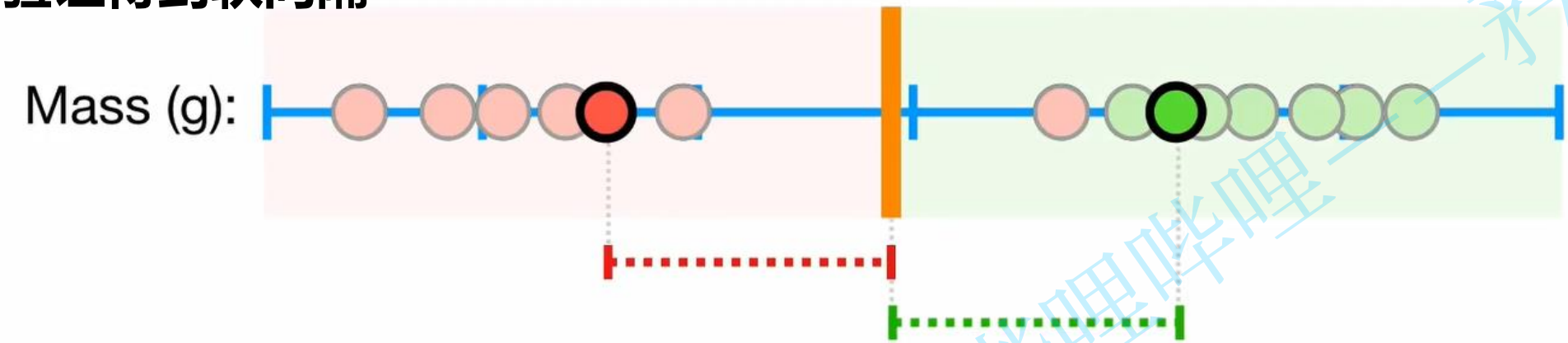
The answer is simple: We use **Cross Validation** to determine how many misclassifications and observations to allow inside of the **Soft Margin** to get the best classification.

交叉验证得到软间隔



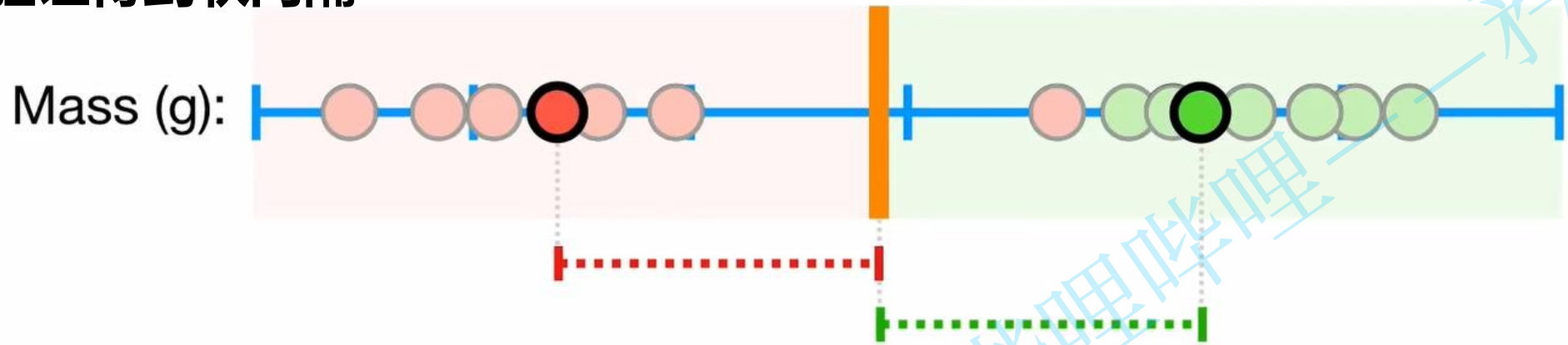
The answer is simple: We use **Cross Validation** to determine how many misclassifications and observations to allow inside of the **Soft Margin** to get the best classification.

交叉验证得到软间隔



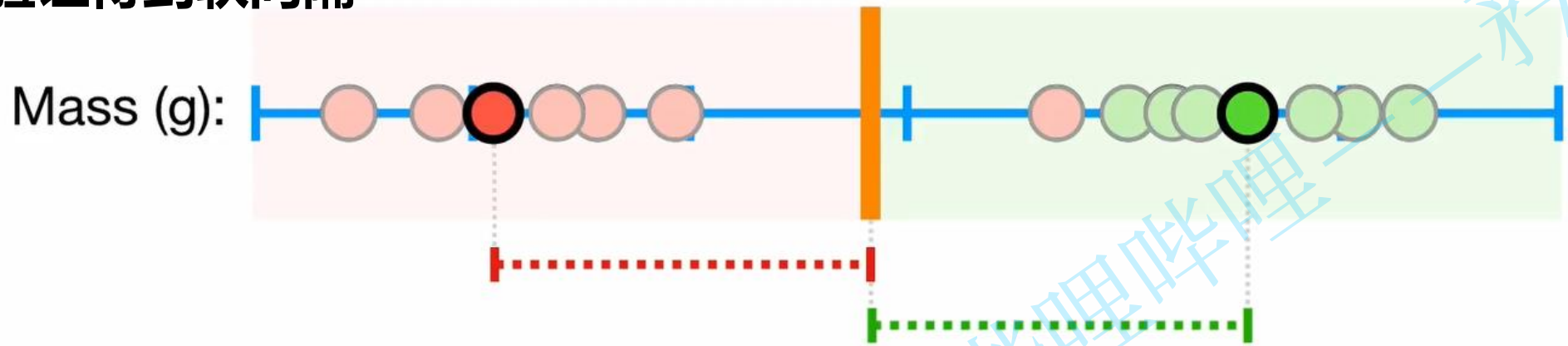
The answer is simple: We use **Cross Validation** to determine how many misclassifications and observations to allow inside of the **Soft Margin** to get the best classification.

交叉验证得到软间隔



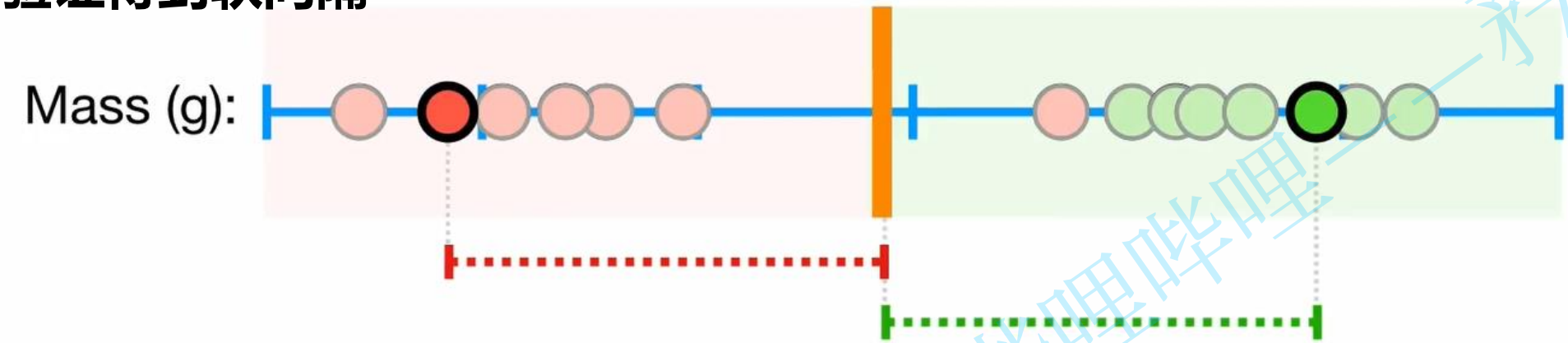
The answer is simple: We use **Cross Validation** to determine how many misclassifications and observations to allow inside of the **Soft Margin** to get the best classification.

交叉验证得到软间隔



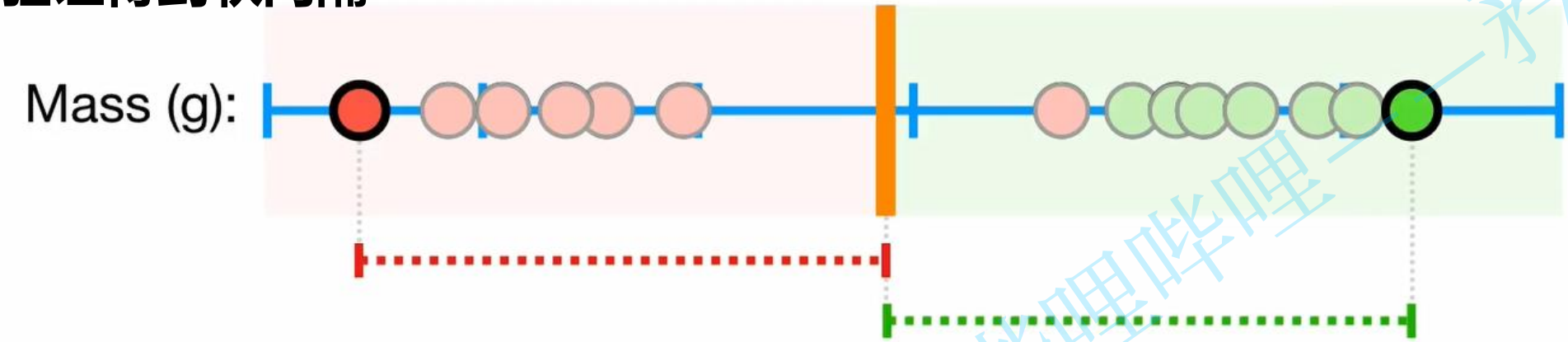
The answer is simple: We use **Cross Validation** to determine how many misclassifications and observations to allow inside of the **Soft Margin** to get the best classification.

交叉验证得到软间隔



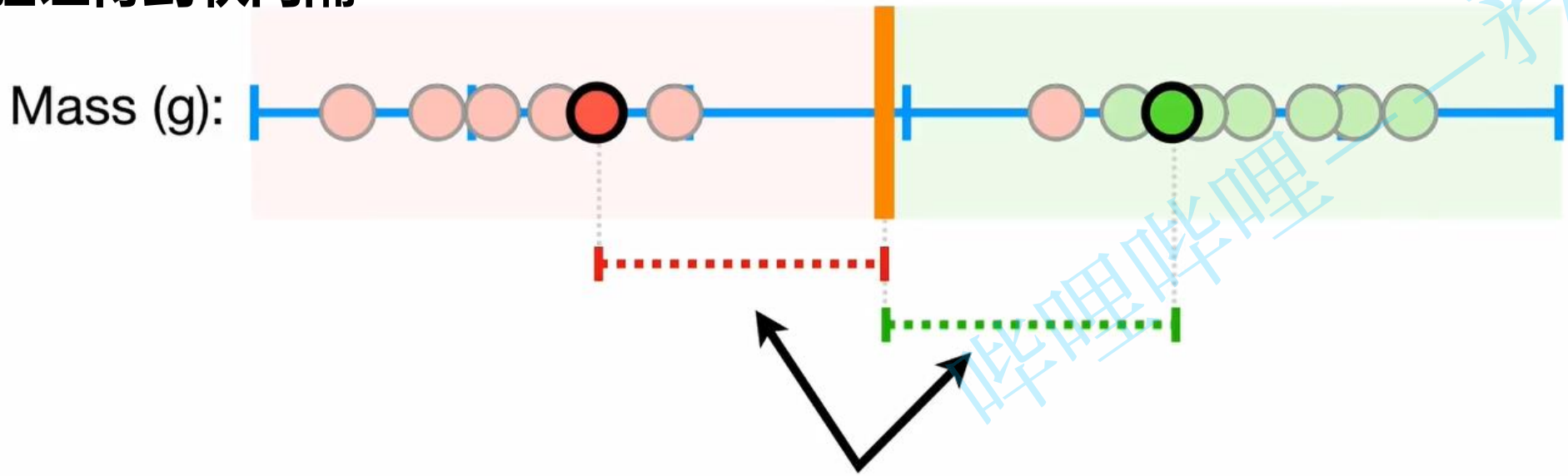
The answer is simple: We use **Cross Validation** to determine how many misclassifications and observations to allow inside of the **Soft Margin** to get the best classification.

交叉验证得到软间隔



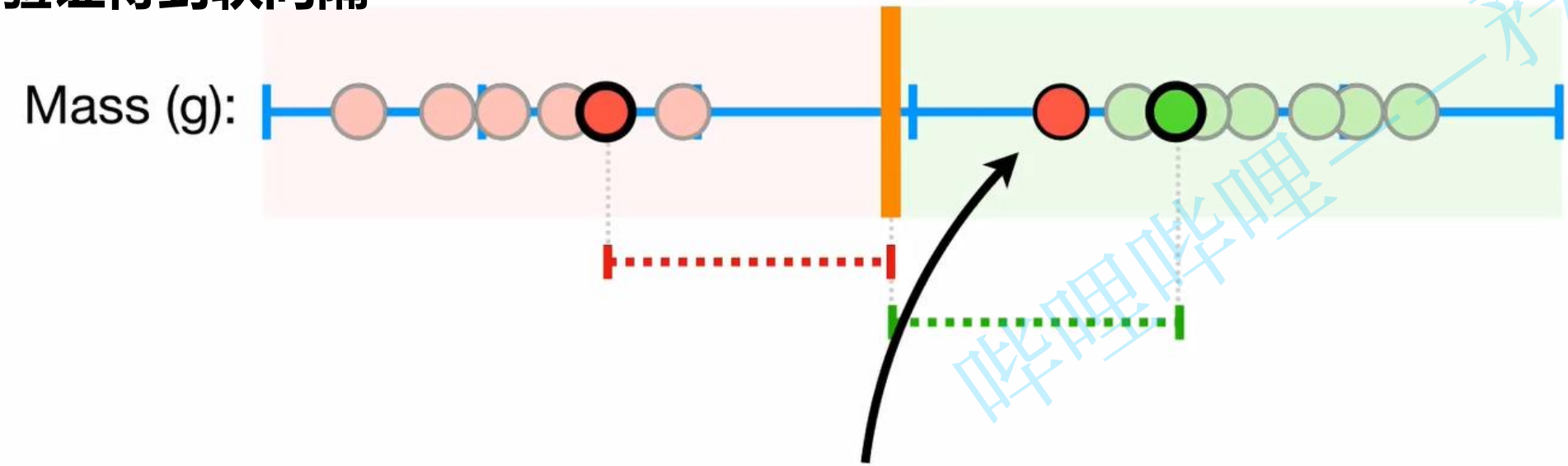
The answer is simple: We use **Cross Validation** to determine how many misclassifications and observations to allow inside of the **Soft Margin** to get the best classification.

交叉验证得到软间隔

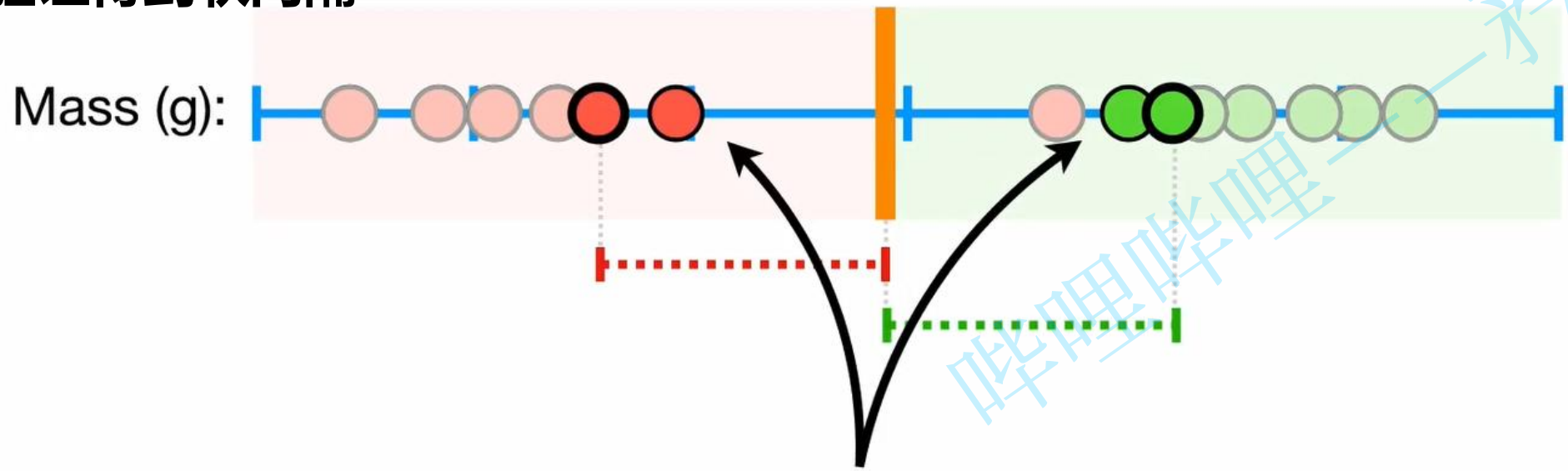


For example, if **Cross Validation** determined that this was the best **Soft Margin**…

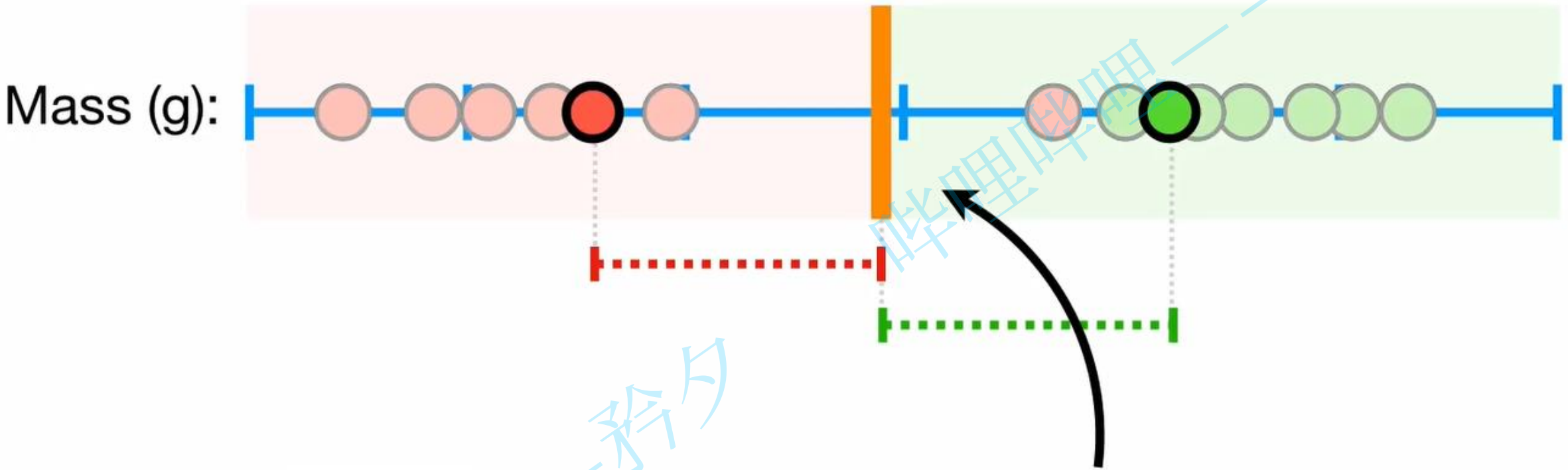
交叉验证得到软间隔



交叉验证得到软间隔

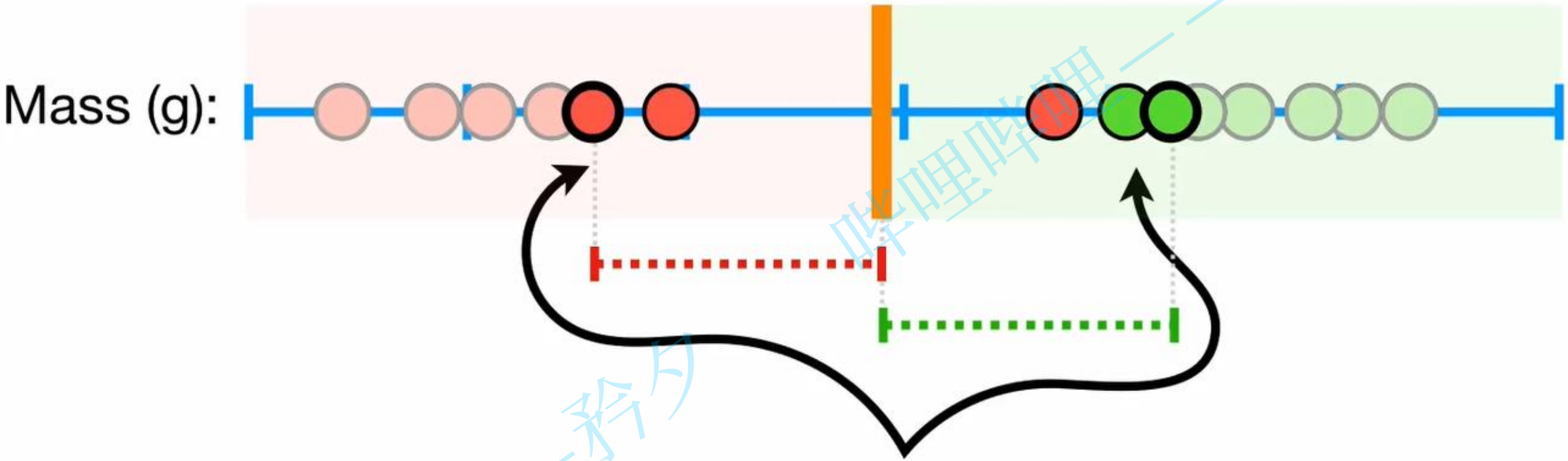


…and two observations, that are correctly classified, to be within the **Soft Margin**.

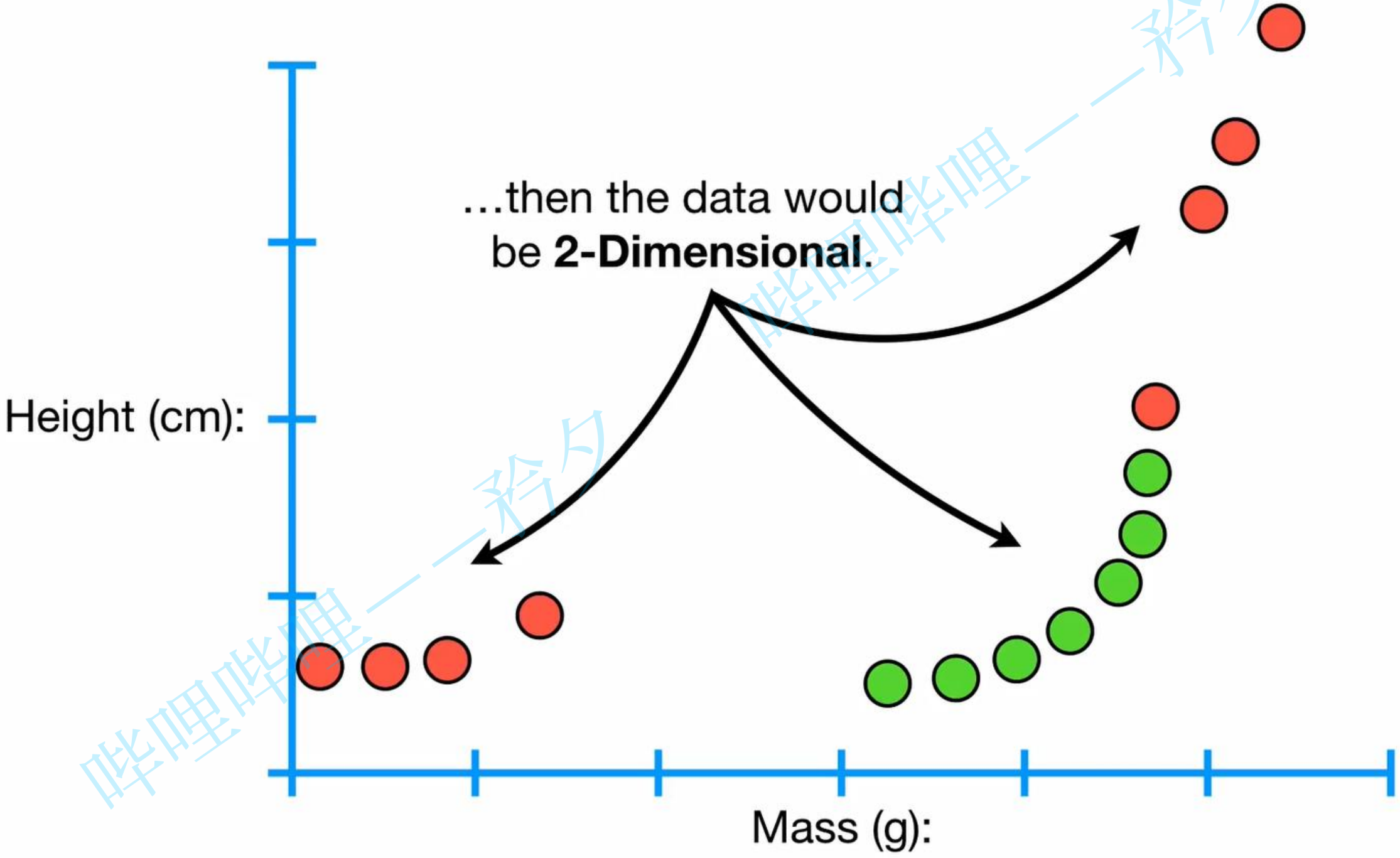


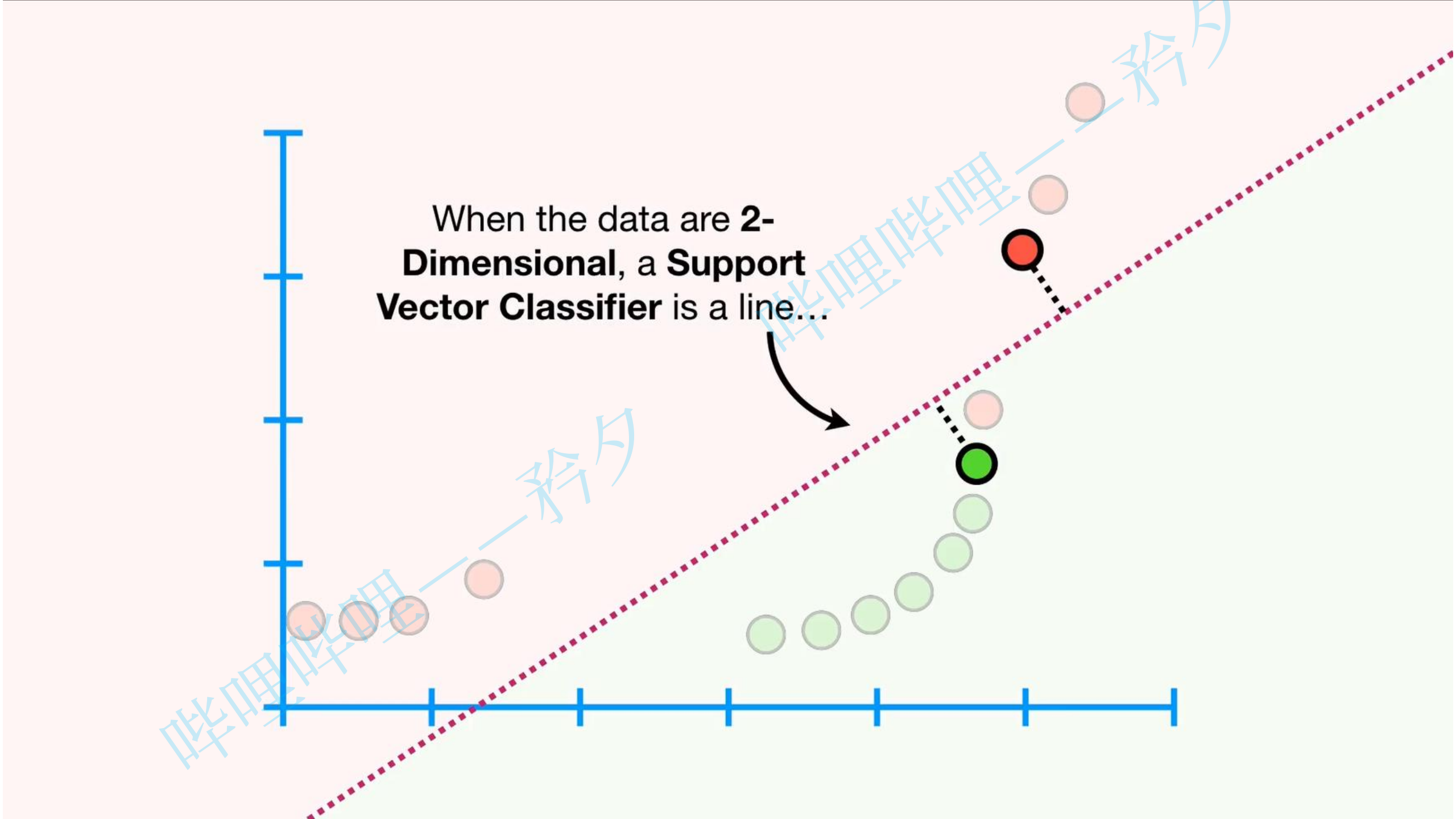
we are using a **Soft Margin Classifier** aka a **Support Vector Classifier** to classify observations.

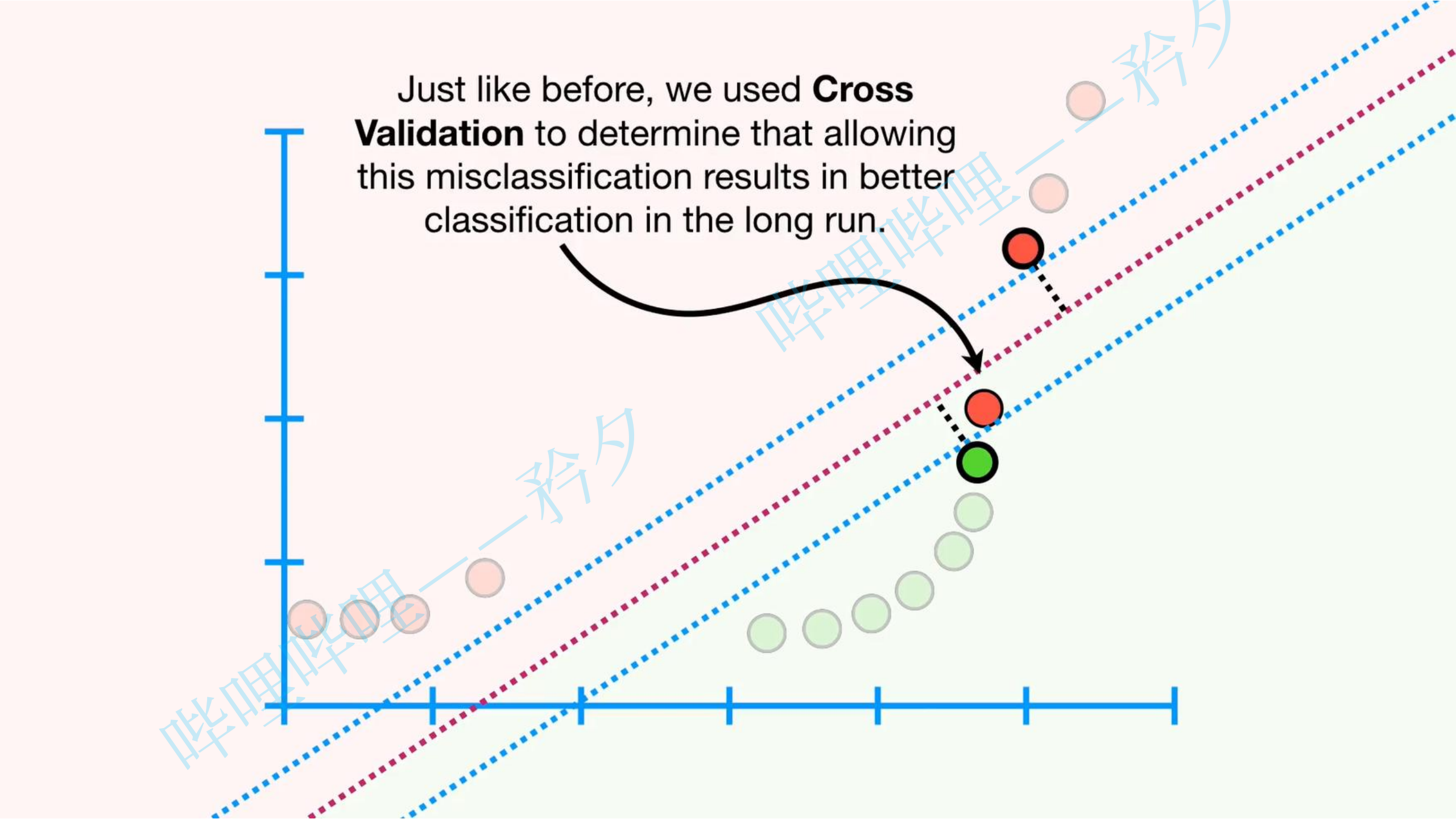
支持向量分类器

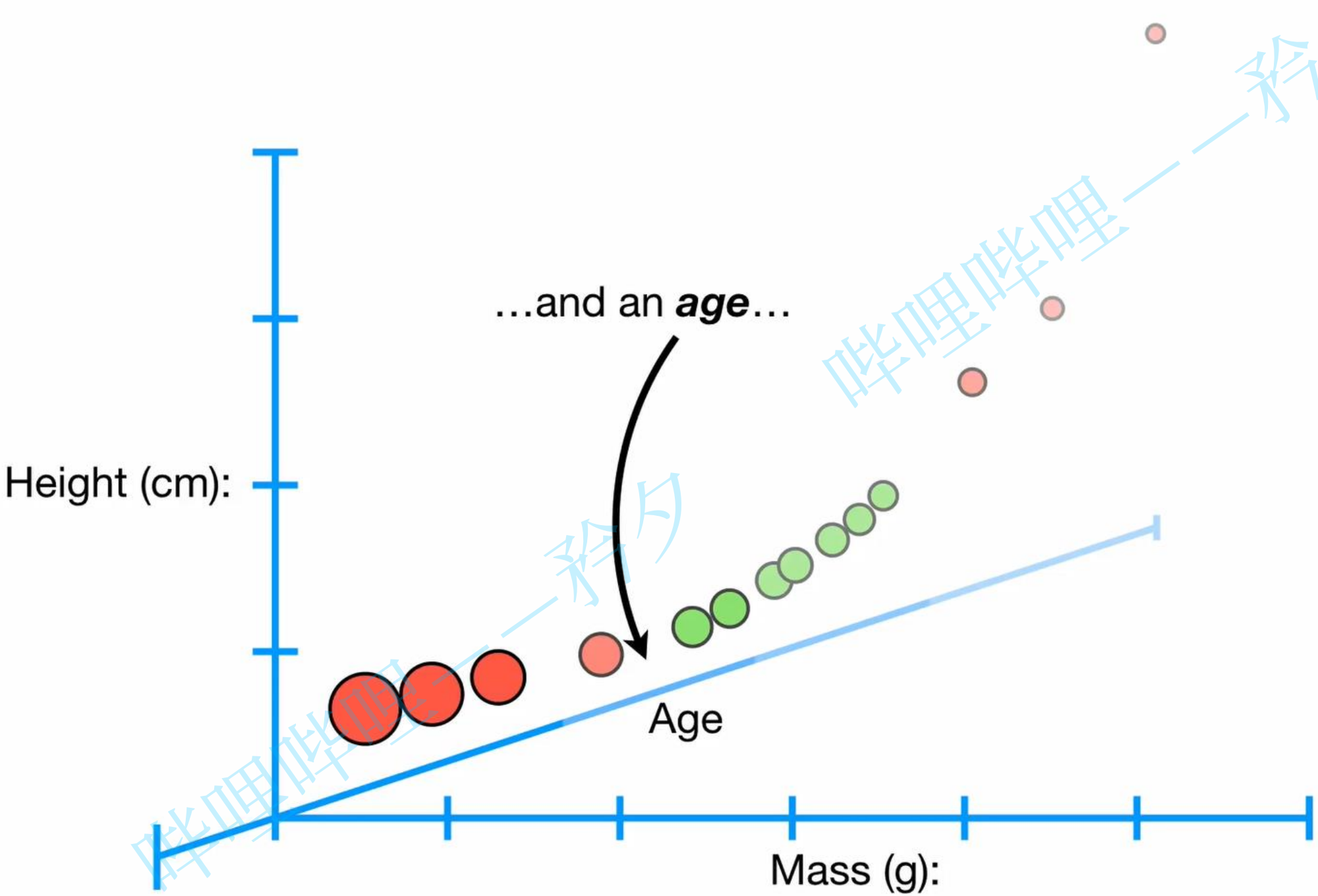


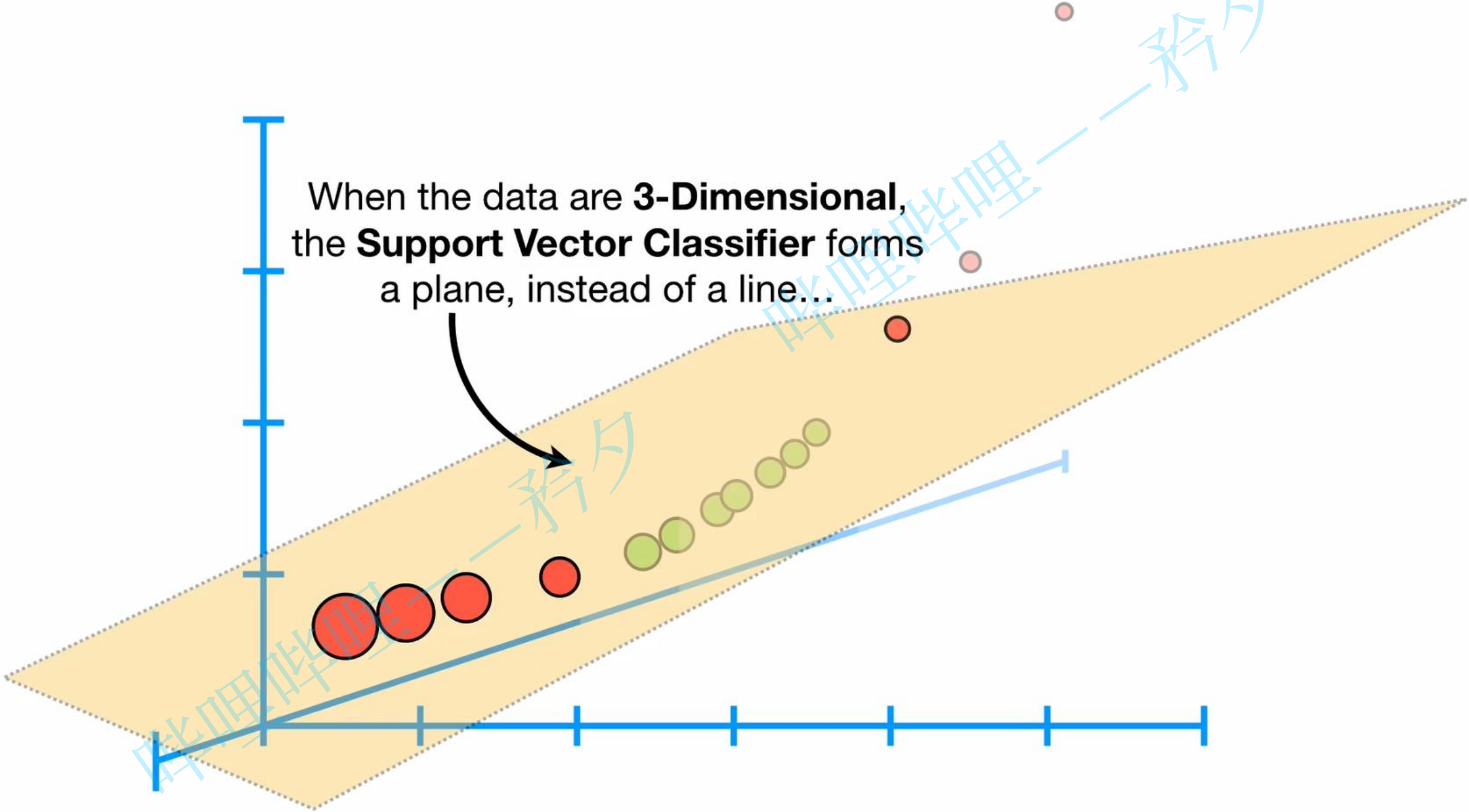
The name **Support Vector Classifier** comes from the fact that the observations on the edge *and within* the **Soft Margin** are called **Support Vectors**.


…then the data would be **2-Dimensional**.
Height (cm):
Mass (g):


When the data are **2-Dimensional**, a **Support Vector Classifier** is a line...


Just like before, we used **Cross Validation** to determine that allowing this misclassification results in better classification in the long run.


…and an *age*…
Height (cm):
Age
Mass (g):


When the data are **3-Dimensional**,
the **Support Vector Classifier** forms
a plane, instead of a line...

NOTE: If we measured ***mass***, ***height***, ***age*** and ***blood pressure***, then the data would be in **4 Dimensions**…

暂时画不出来！

回顾：

- 一维数据：支持向量分类器是一维空间的一个0维点
- 二维数据：支持向量分类器是二维空间的一条一维直线
- 三维数据：支持向量分类器是三维空间的一个二维平面
- 四维数据：支持向量分类器是四维空间的一个三维超平面

……

把上面有维度的分类器统称为**超平面**，超平面是数学定义的n维欧氏空间中的(n-1)维子空间（严谨论述请查阅相关数学资料）

2.支持向量分类器-数学补充

- 松弛变量

松弛变量-解决outliers

原来的约束条件：$y_i(w^T x_i + b) \geq 1, \quad i = 1, \ldots, n$

加了松弛变量的约束条件：

$$y_i(w^T x_i + b) \geq 1 - \xi_i, i = 1, \ldots, n$$

不能允许松弛变量任意大，因此在目标函数中加入正则项，
使得松弛变量尽量小：

$$\min \quad \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} \xi_i$$

松弛变量-解决outliers

新的目标优化函数：

$$\min \quad \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i$$

$$s.t. \quad y_i(w^T x_i + b) \geq 1 - \xi_i, i = 1, ..., n$$

$$\xi_i \geq 0, i = 1, ..., n$$

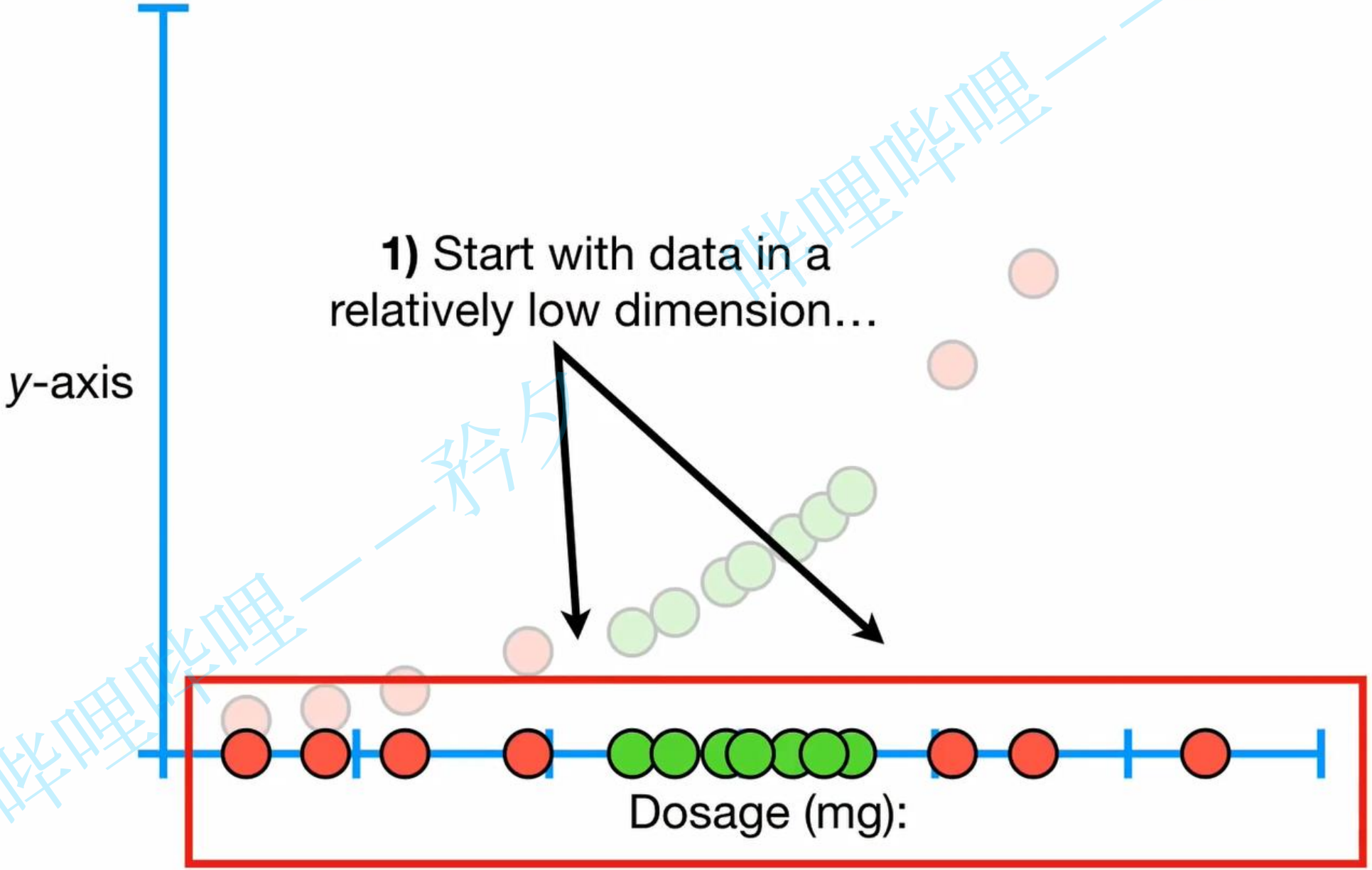
3.支持向量机

In this new example, with tons of overlap, we are now looking at **Drug Dosages**…

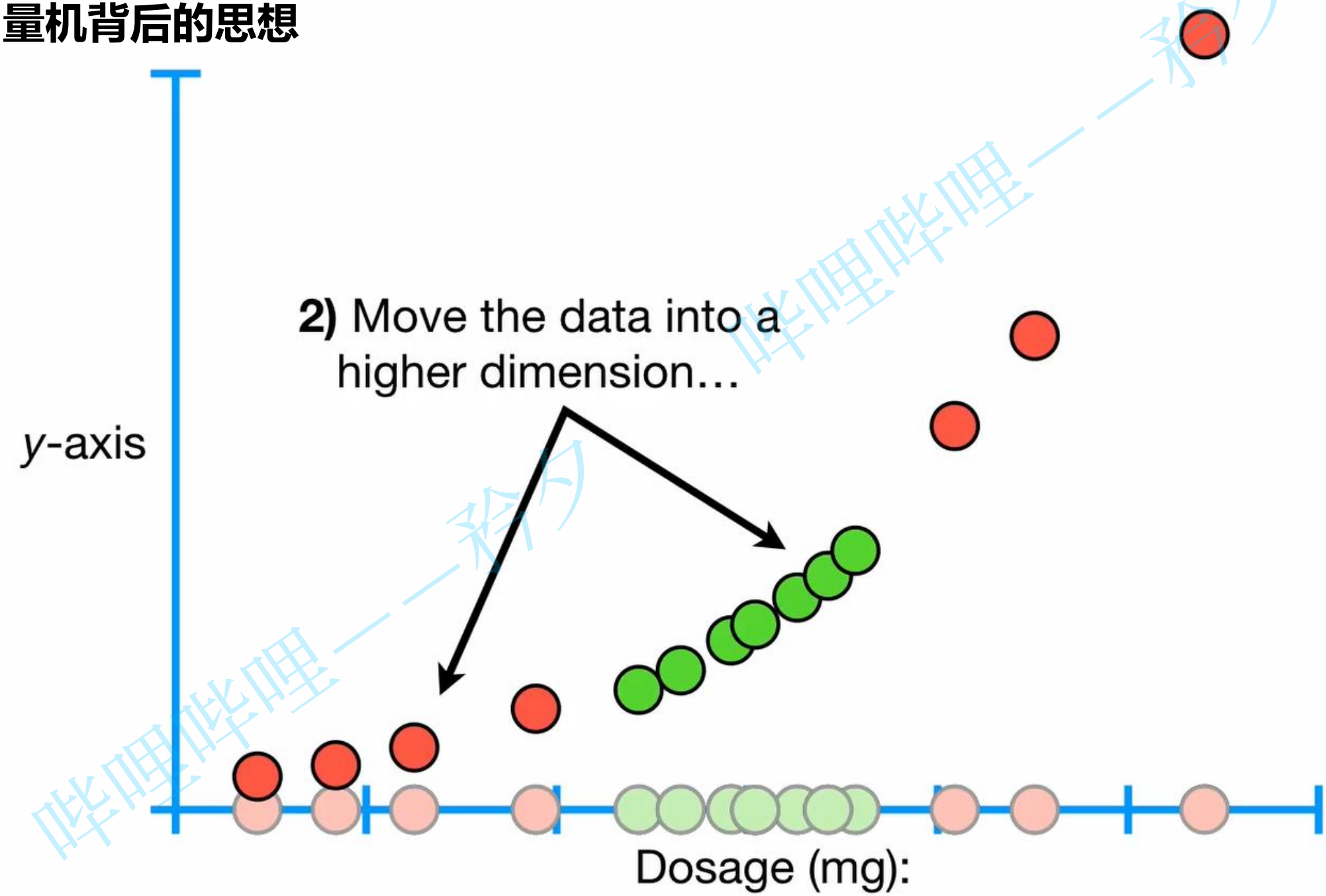
线性不可分，咋办?

Support Vector Machines!!!

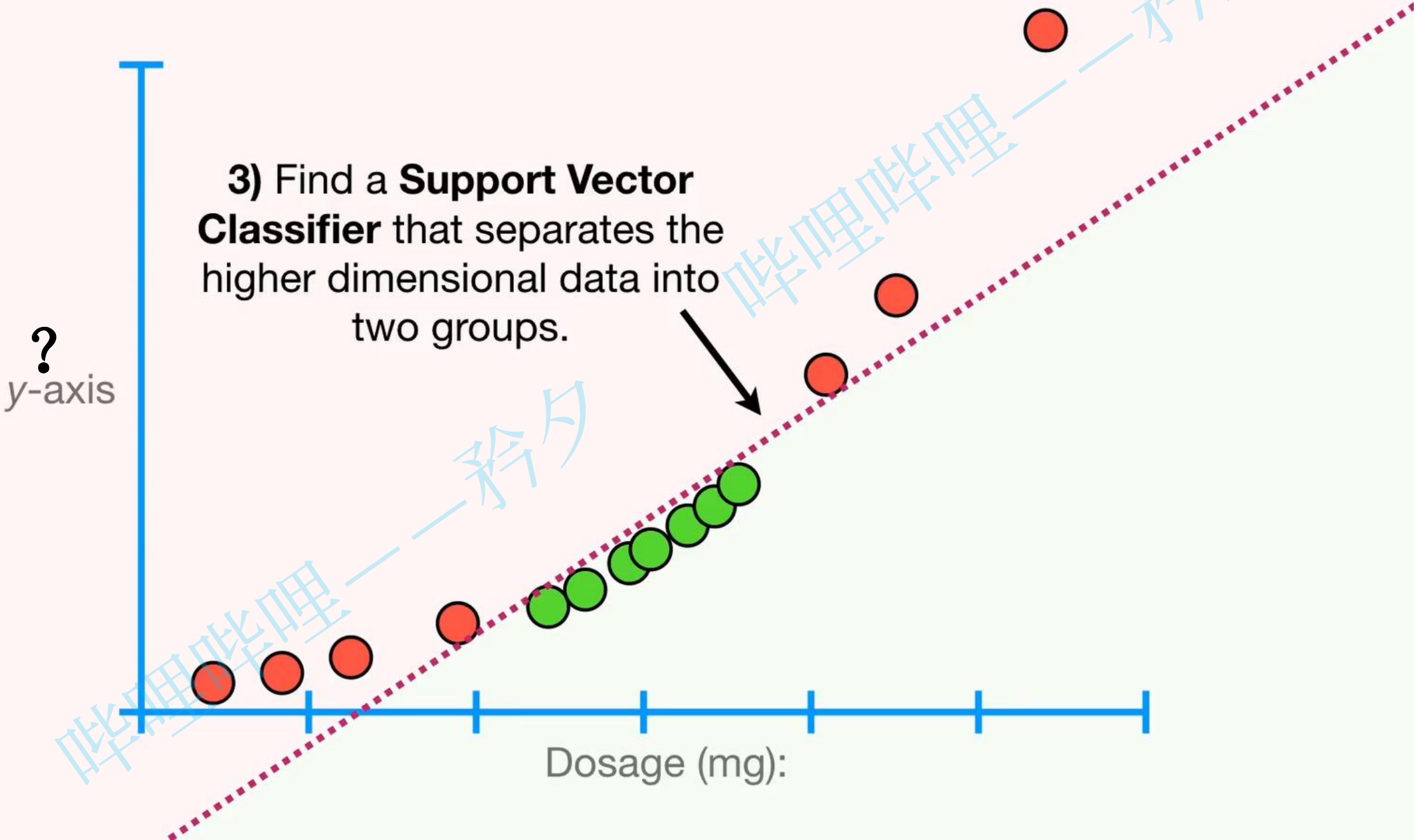
支持向量机背后的思想



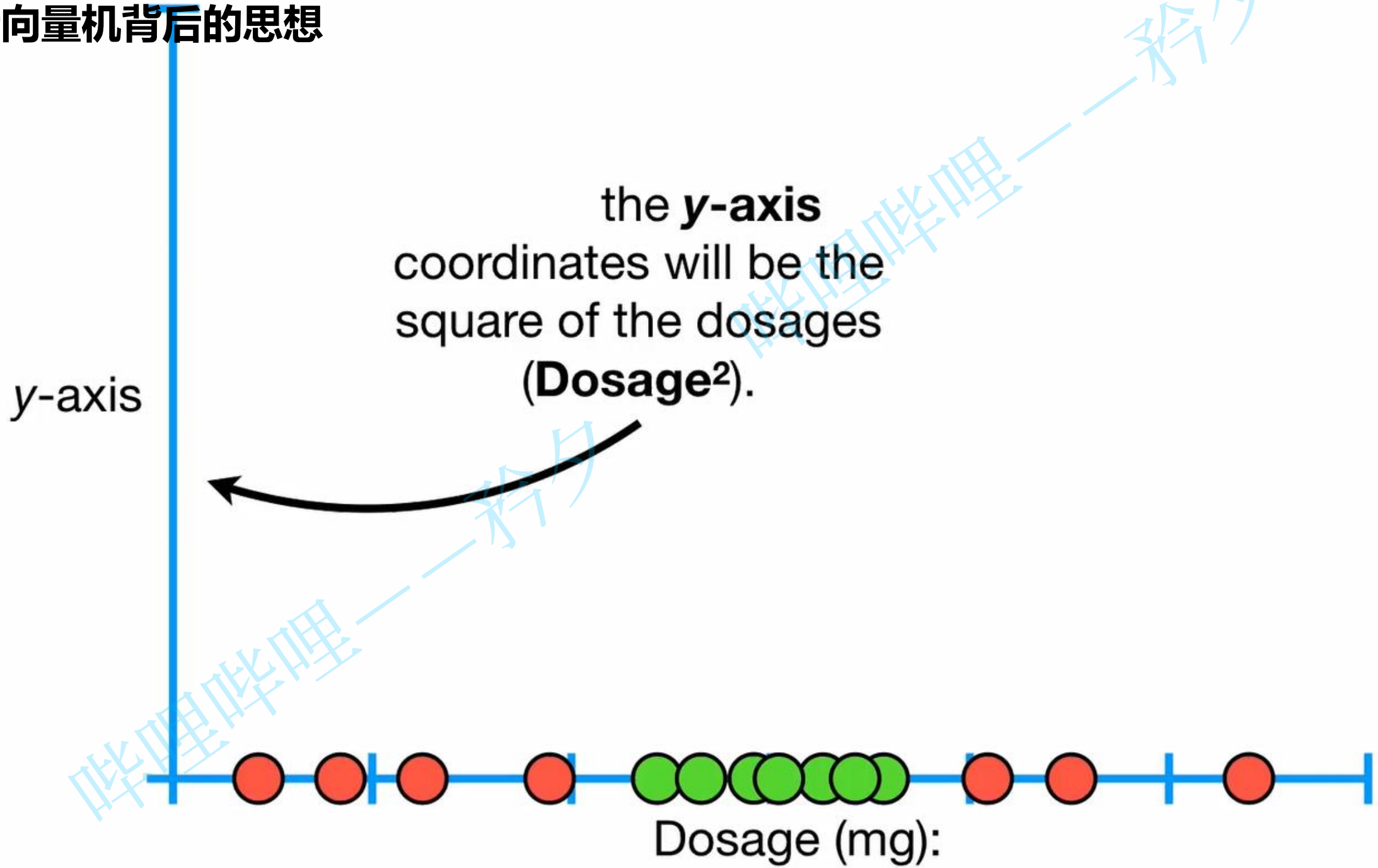
支持向量机背后的思想



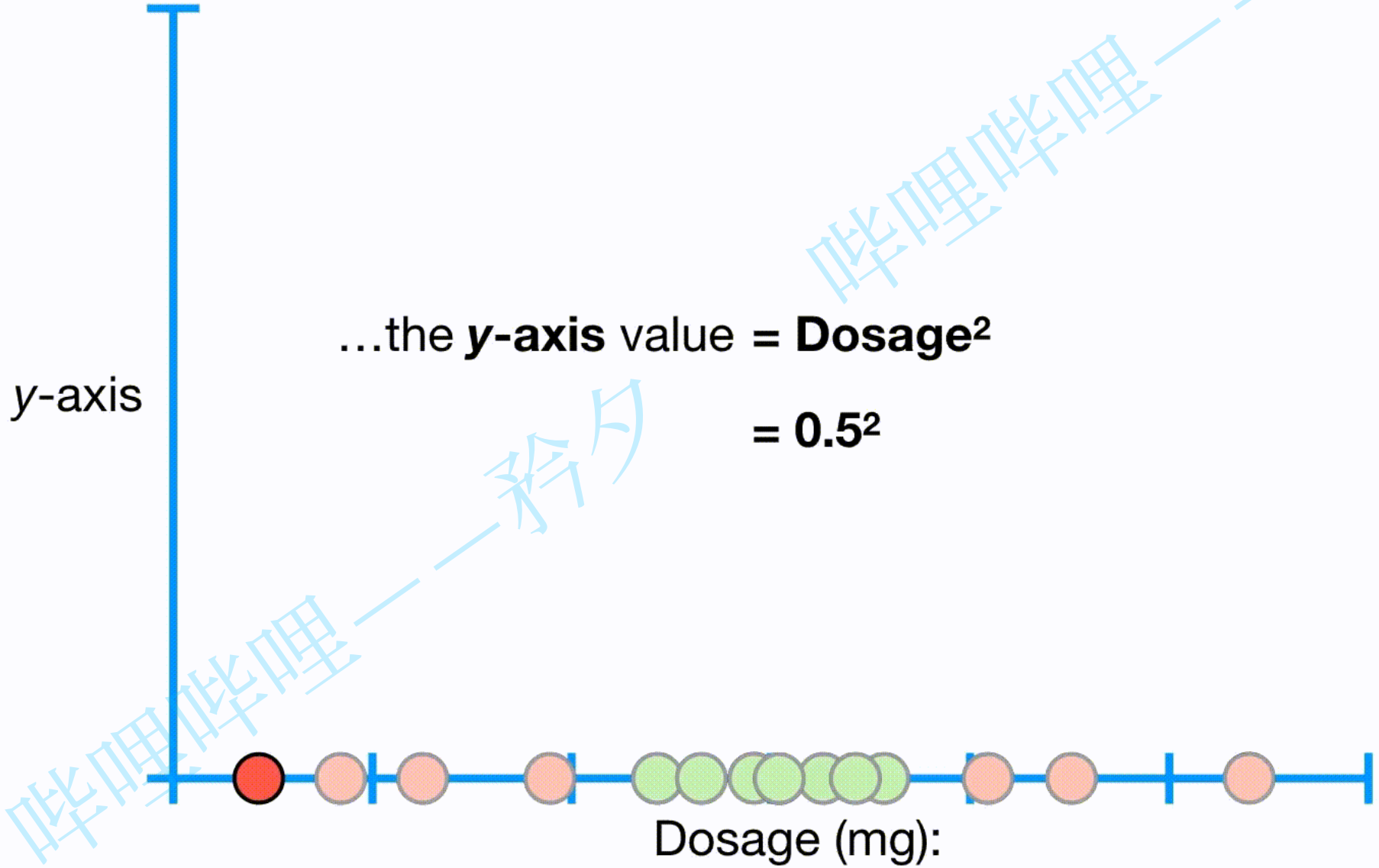
支持向量机背后的思想



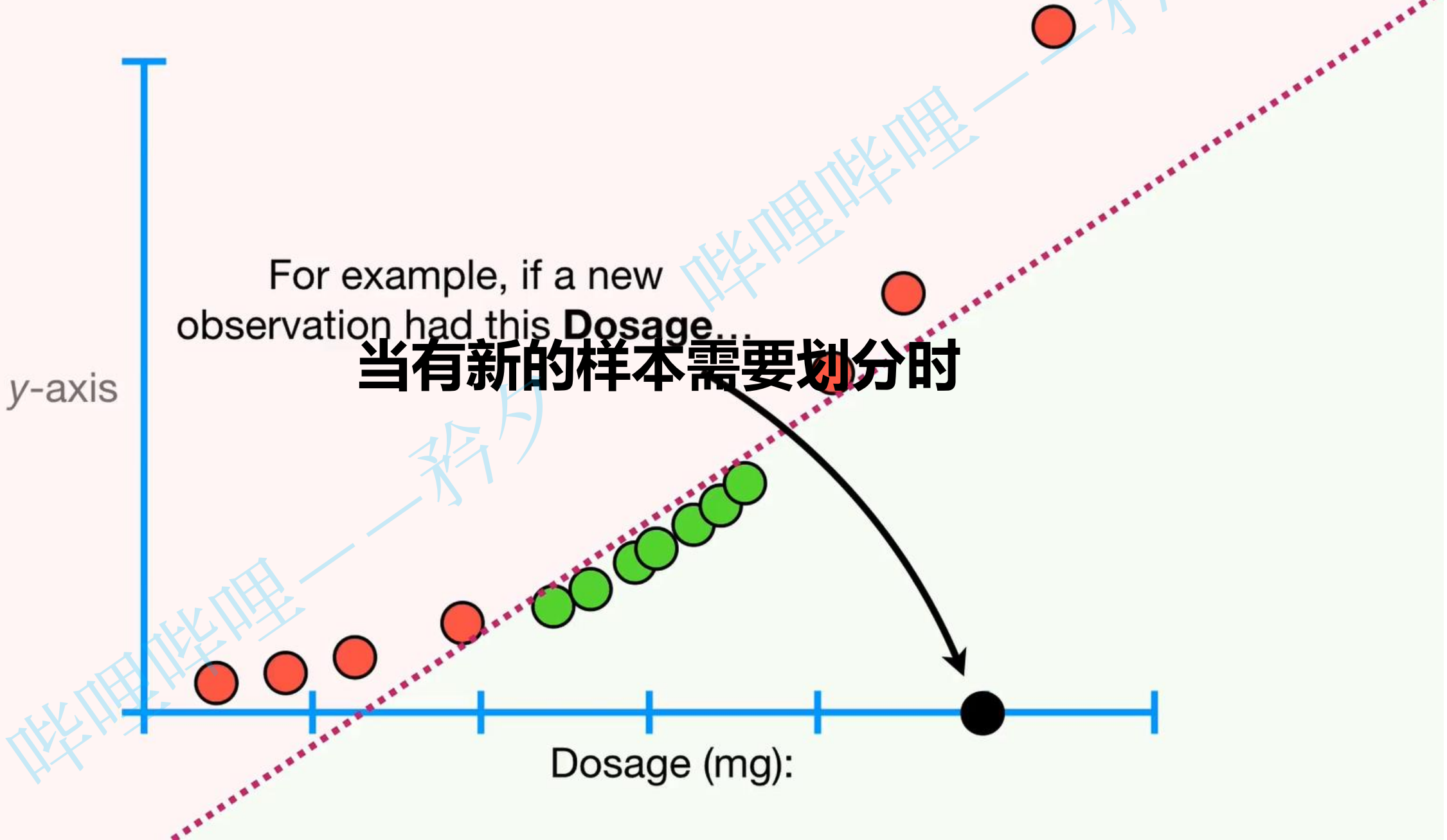
支持向量机背后的思想



支持向量机背后的思想

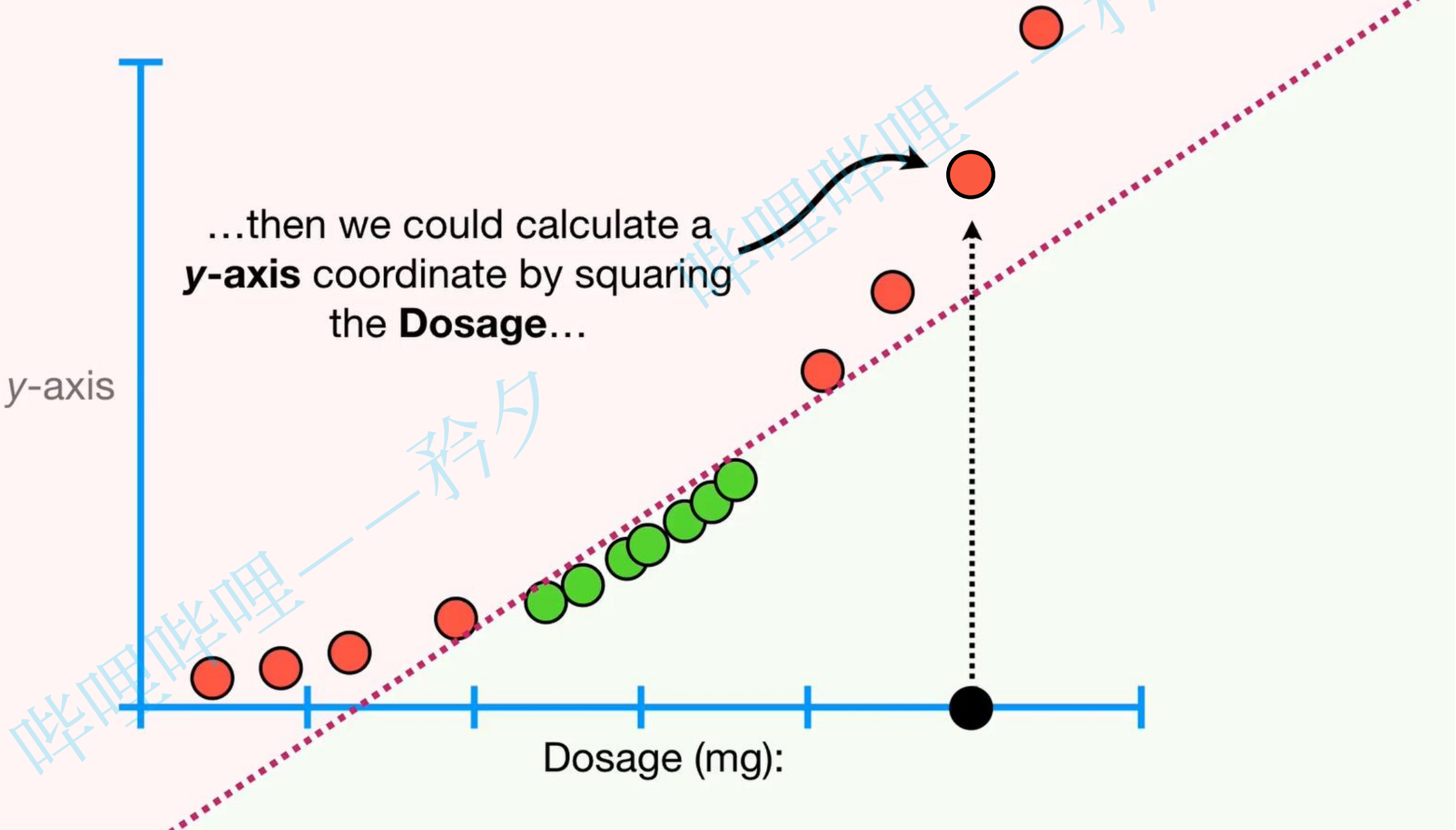


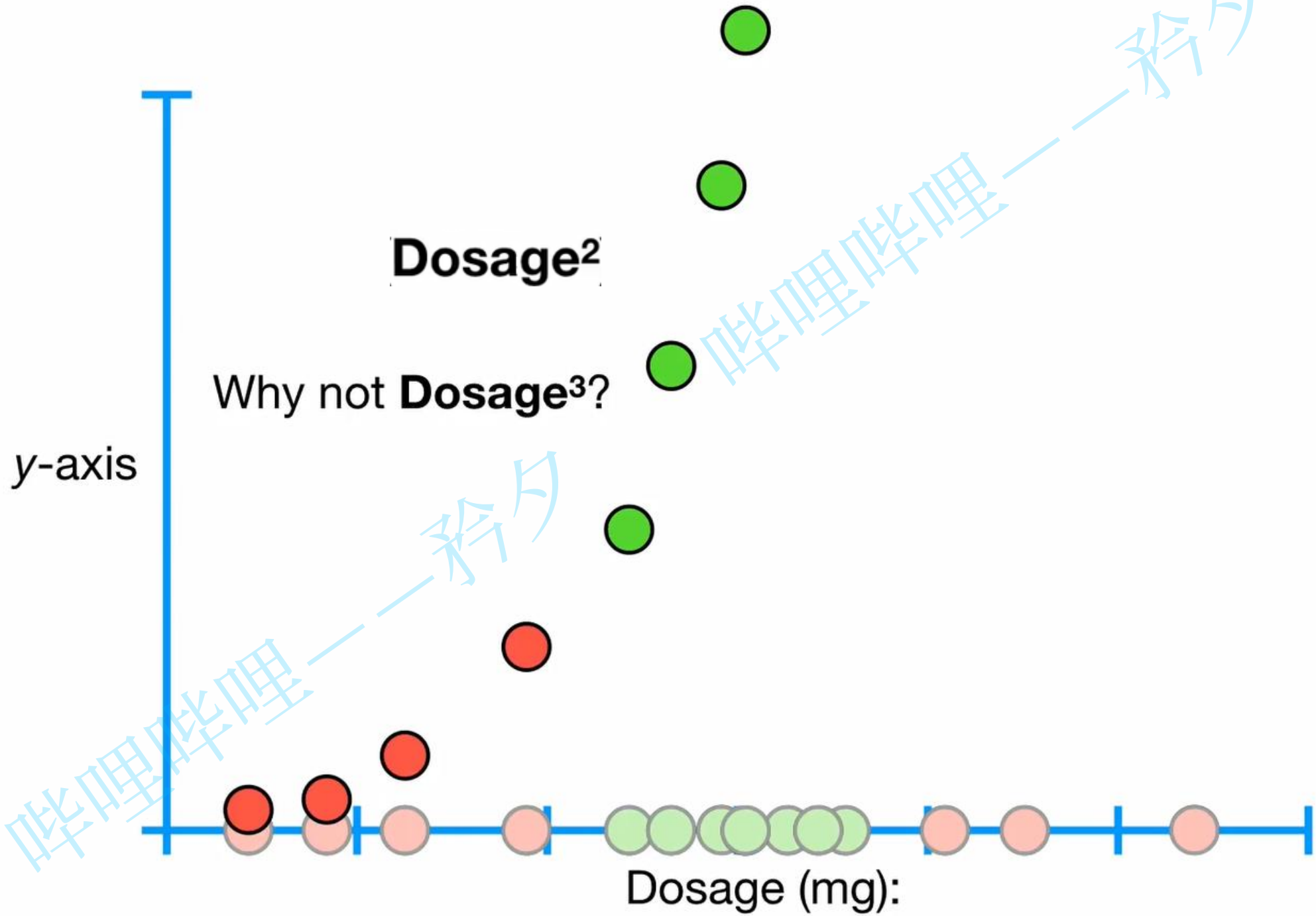
支持向量机背后的思想



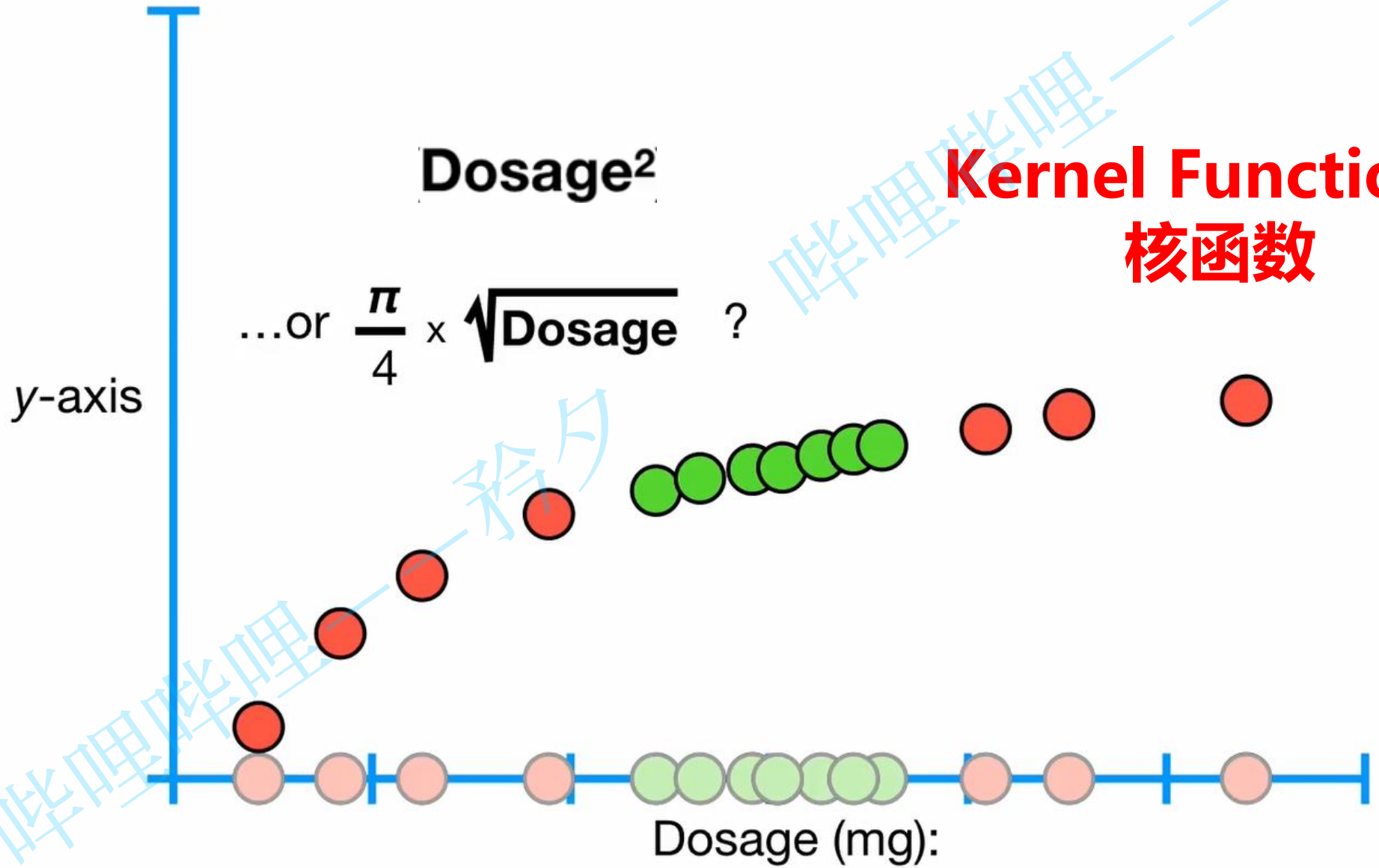
当有新的样本需要划分时

支持向量机背后的思想




Dosage²
Why not Dosage³?
y-axis
Dosage (mg):

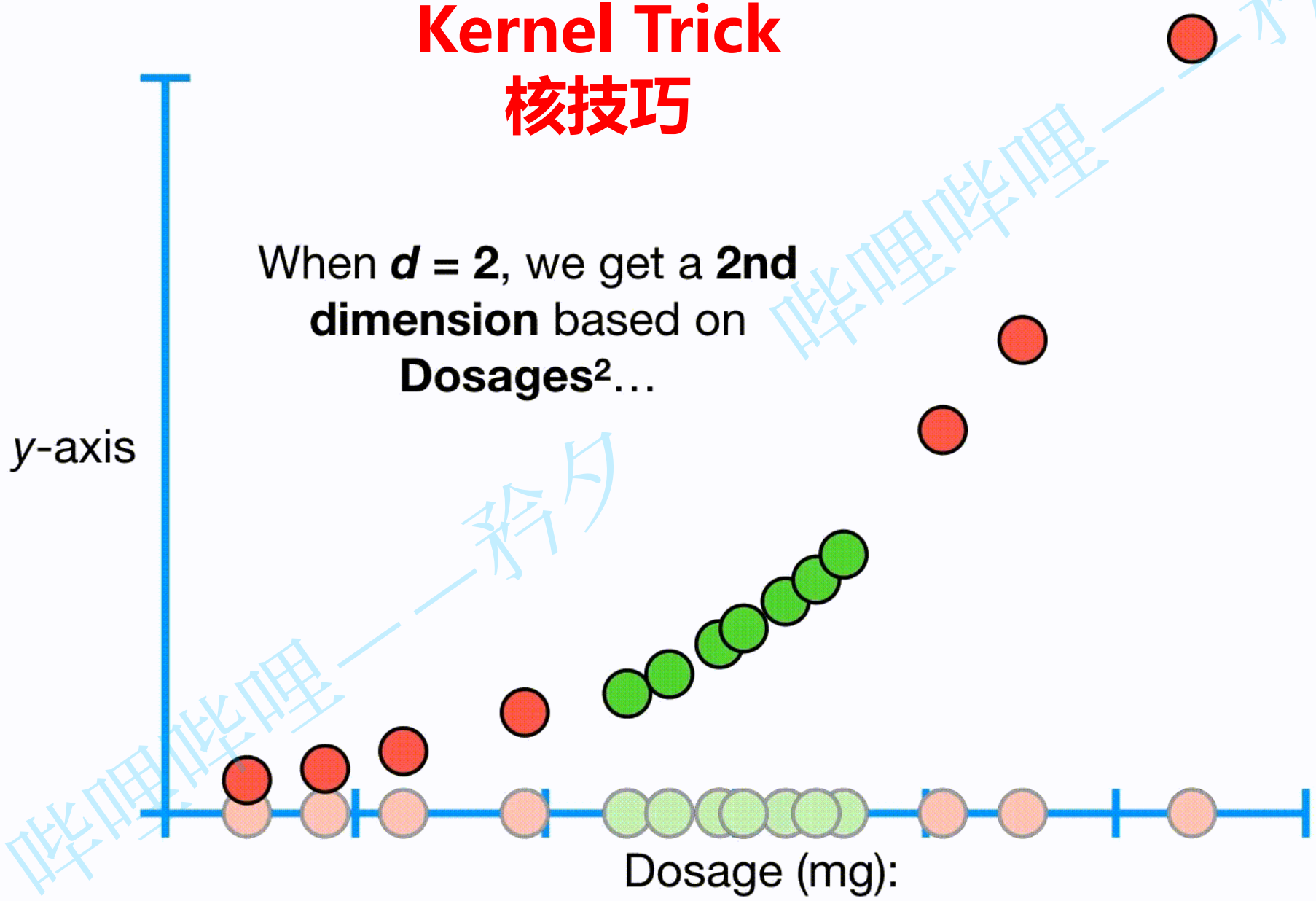
Kernel Functions
核函数



For this example, I used the **Polynomial Kernel**, which has a parameter, ***d***, which stands for the **degree** of the polynomial.

When ***d* = 1**, the **Polynomial Kernel** computes the relationships between each pair of observations in **1-Dimension**…

Kernel Trick
核技巧



3.支持向量机-数学补充

- 对偶
- 核函数

对偶

原目标函数：

$$\min \frac{1}{2}\|w\|^2, \quad s.t. \quad y_i(w^T x_i + b) \geq 1, i = 1, ..., n$$

拉格朗日函数：

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{n} \alpha_i (y_i(w^T x_i + b) - 1)$$

目标函数变为：

$$\min_{w,b} \max_{\alpha_i \geq 0} \mathcal{L}(w, b, \alpha) = p^*$$

对偶

目标函数变为：

$$\min_{w,b} \max_{\alpha_i \geq 0} \mathcal{L}(w, b, \alpha) = p^*$$

对偶：

$$\max_{\alpha_i \geq 0} \min_{w,b} \mathcal{L}(w, b, \alpha) = d^*$$

当满足以下条件求解对偶问题的解才等于原始问题的解：

$$p^* = d^*$$

当满足以下条件时上等式才成立（这里满足）：

KKT条件

求解对偶三步骤

$$\max_{\alpha_i \geq 0} \min_{w,b} \mathcal{L}(w, b, \alpha) = d^*$$

(1) 首先固定 α，要让 $\mathcal{L}$ 关于 w 和 b 最小化，我们分别对 w，b 求偏导数，即令 $\partial\mathcal{L}/\partial w$ 和 $\partial\mathcal{L}/\partial b$ 等于零：

$$\frac{\partial L}{\partial w} = 0 \Rightarrow w = \sum_{i=1}^{n} \alpha_i y_i x_i$$

$$\frac{\partial L}{\partial b} = 0 \Rightarrow \sum_{i=1}^{n} \alpha_i y_i = 0$$

求解对偶三步骤

$$\max_{\alpha_i \geq 0} \min_{w,b} \mathcal{L}(w, b, \alpha) = d^*$$

(1) 首先固定 α，要让 $\mathcal{L}$ 关于 w 和 b 最小化，我们分别对 w，b 求偏导数，即令 $\partial\mathcal{L}/\partial w$ 和 $\partial\mathcal{L}/\partial b$ 等于零：

$$\frac{\partial L}{\partial w} = 0 \Rightarrow w = \sum_{i=1}^{n} \alpha_i y_i x_i$$

$$\frac{\partial L}{\partial b} = 0 \Rightarrow \sum_{i=1}^{n} \alpha_i y_i = 0$$

将以上结果代入之前的 $\mathcal{L}$，得到：

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{n} \alpha_i (y_i(w^T x_i + b) - 1)$$

求解对偶三步骤

$$\max_{\alpha_i \geq 0} \min_{w,b} \mathcal{L}(w, b, \alpha) = d^*$$

(1) 首先固定 α，要让 $\mathcal{L}$ 关于 w 和 b 最小化，我们分别对 w，b 求偏导数，即令 $\partial\mathcal{L}/\partial w$ 和 $\partial\mathcal{L}/\partial b$ 等于零：

$$\frac{\partial L}{\partial w} = 0 \Rightarrow w = \sum_{i=1}^{n} \alpha_i y_i x_i$$

$$\frac{\partial L}{\partial b} = 0 \Rightarrow \sum_{i=1}^{n} \alpha_i y_i = 0$$

进而得到：

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j x_i^T x_j - \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j x_i^T x_j - b \sum_{i=1}^{n} \alpha_i y_i + \sum_{i=1}^{n} \alpha_i$$

$$= \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j x_i^T x_j$$

求解对偶三步骤

(2) 求对 α 的极大，即是关于对偶问题的最优化问题。经过上面第一个步骤的求 w 和 b，得到的拉格朗日函数式子已经没有了变量 w，b，只有 α。从上面的式子得到：

$$\max_{\alpha} \quad \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j x_i^T x_j$$

$$s.t. \quad \alpha_i \geq 0, i = 1, ..., n$$

$$\sum_{i=1}^{n} \alpha_i y_i = 0$$

求解对偶三步骤

这样，求出了 α_i，从而根据：

$$w^* = \sum_{i=1}^{n} \alpha_i y_i x_i$$

$$b^* = -\frac{\max_{i:y_i=-1} w^{*T} x_i + \min_{i:y_i=1} w^{*T} x_i}{2}$$

即可求出 w, b，最终得出分离超平面和分类决策函数。

第(2)步还未求出αi，需要靠**SMO**求出αi:

(3) 在求得 $\mathcal{L}(w, b, a)$ 关于 w 和 b 最小化，以及对 α 的极大之后，最后一步便是利用 SMO 算法求解对偶问题中的拉格朗日乘子 α。

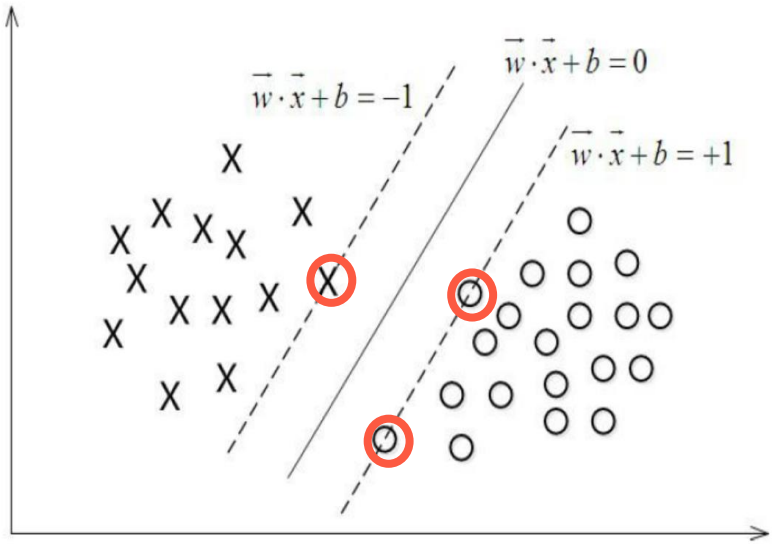
核函数-解决非线性分类

前面的推导得到了超平面的参数：$w^* = \sum_{i=1}^{n} \alpha_i y_i x_i$

超平面函数：

$$f(x) = (\sum_{i=1}^{n} \alpha_i y_i x_i)^T x + b$$

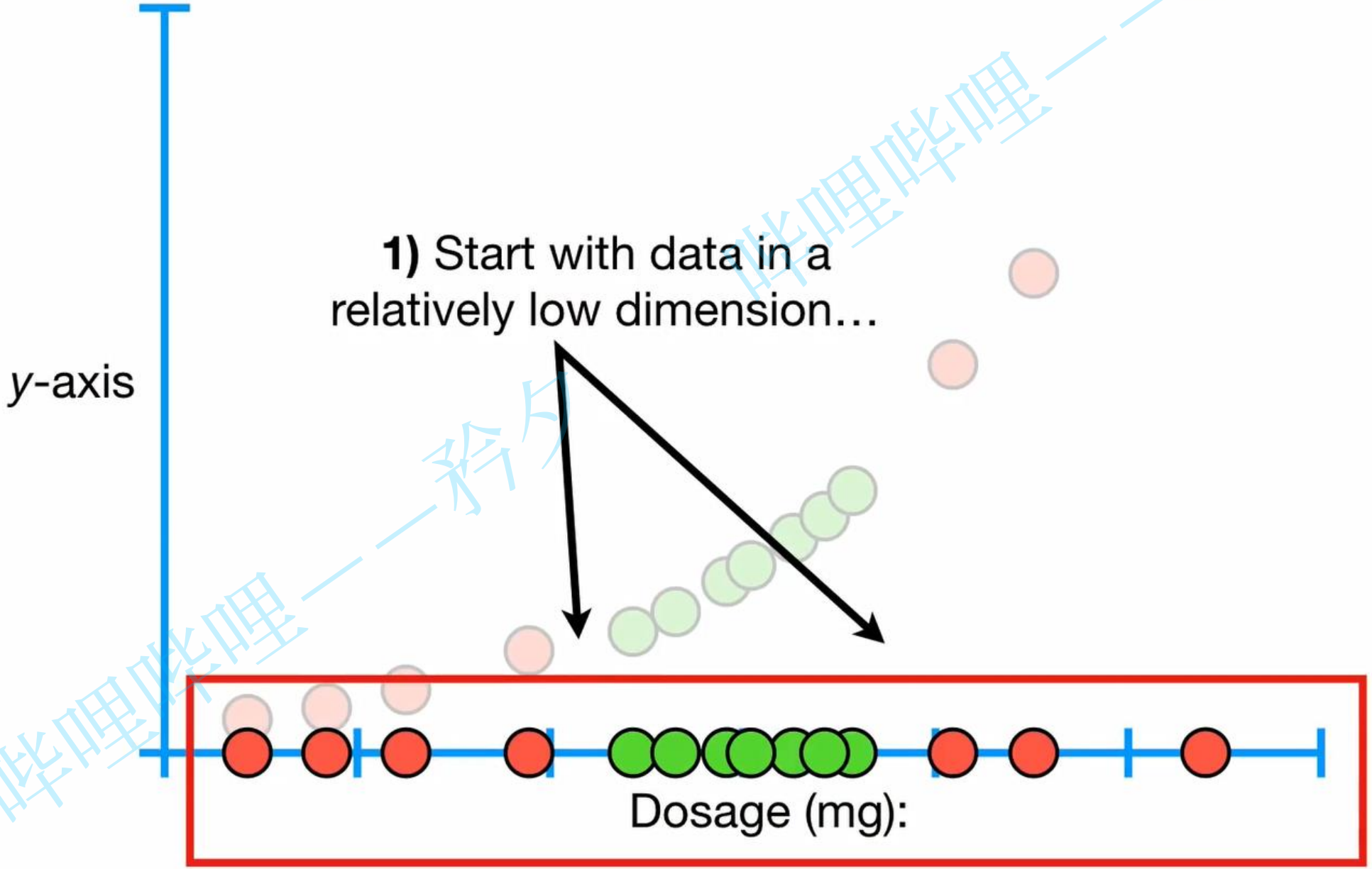
$$= \sum_{i=1}^{n} \alpha_i y_i \langle x_i, x \rangle + b$$



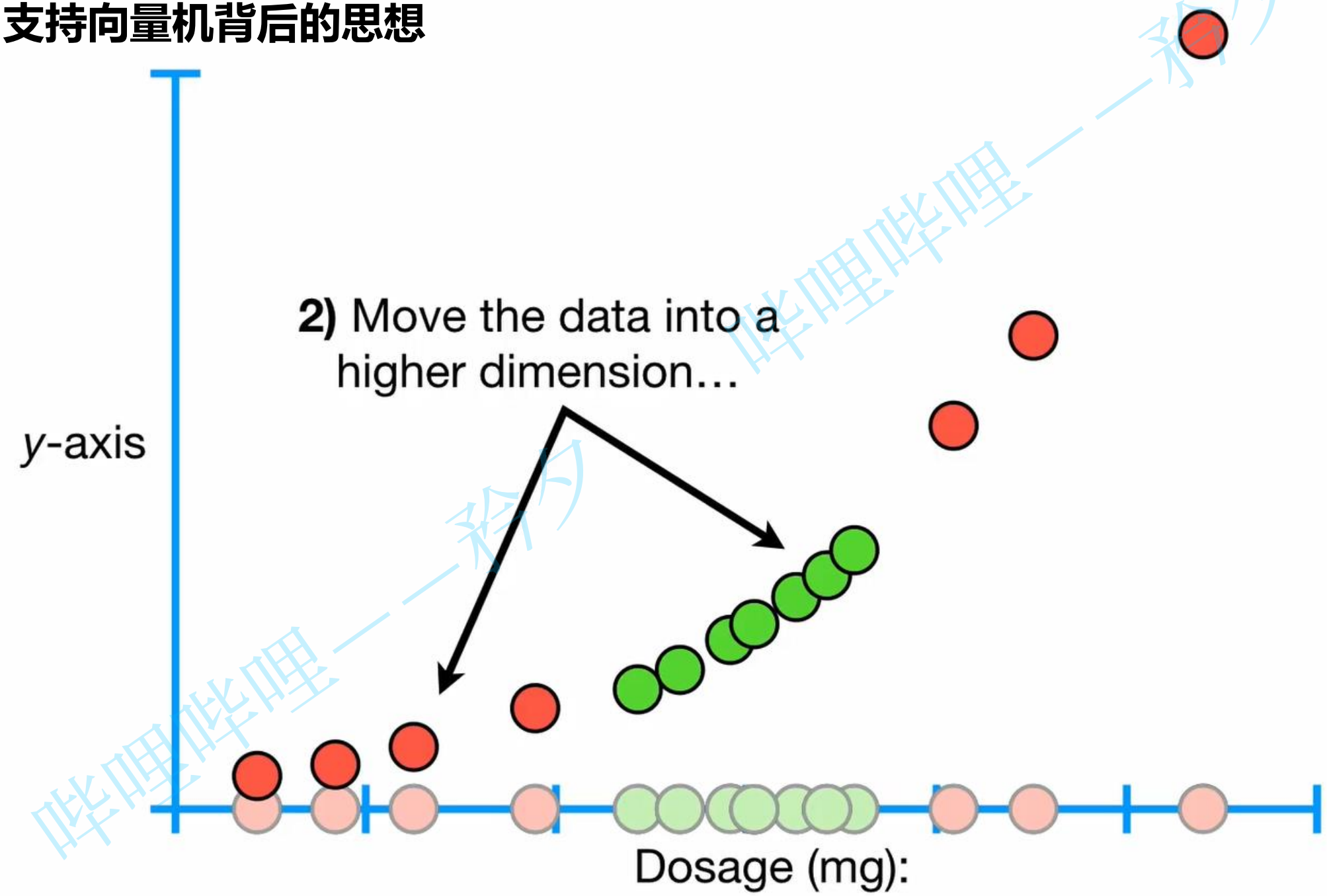
非支持向量所对应的系数都是等于零：

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{n} \alpha_i (y_i(w^T x_i + b) - 1)$$

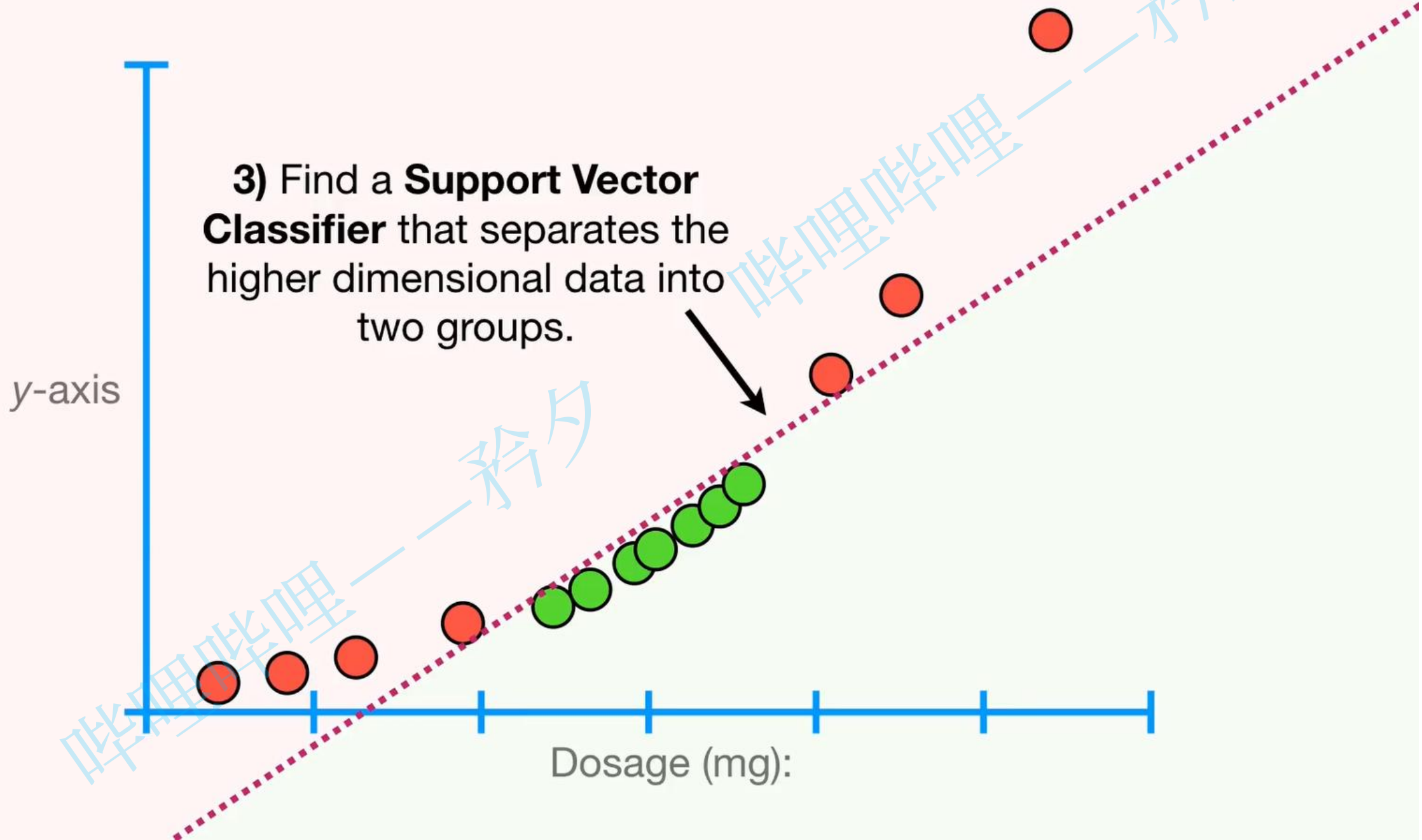
回顾：支持向量机背后的思想



回顾：支持向量机背后的思想



回顾：支持向量机背后的思想

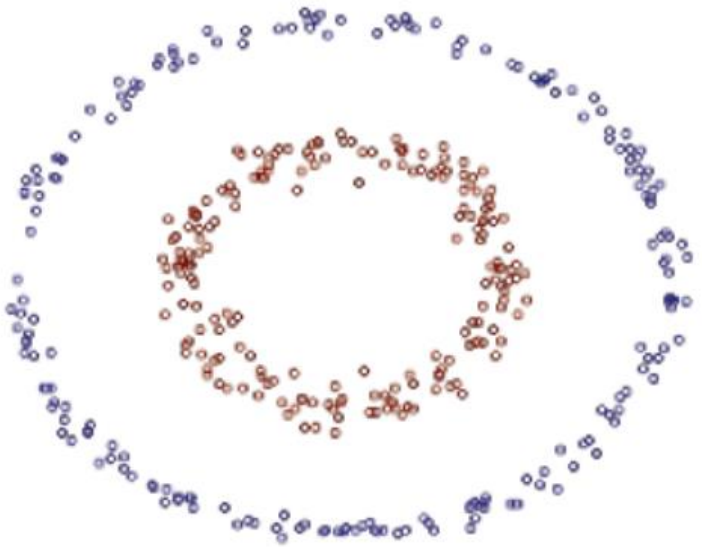


核函数-解决非线性分类

核函数未出现前的解法：

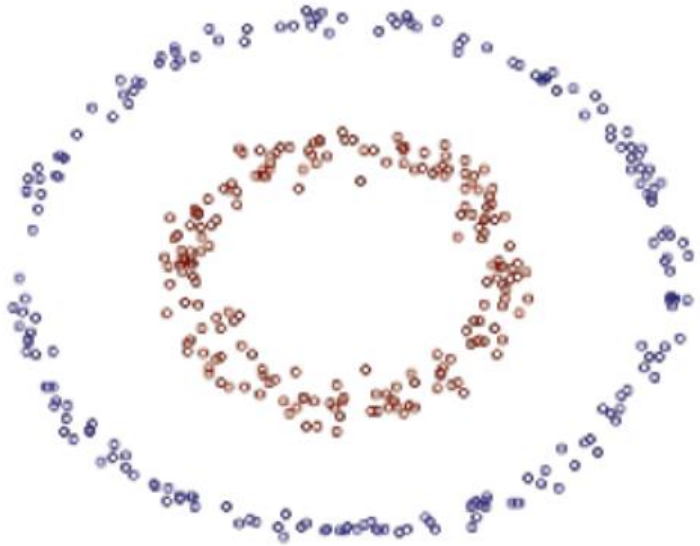
1.使用一个非线性映射Φi将低维数据变换到一个高维特征空间

2.在高维特征空间使用线性学习器分类



线性不可分

核函数-解决非线性分类



线性不可分

用X1 和X2 来表示这个二维平面的两个坐标

一条二次曲线的方程可以写作：

$$a_1X_1 + a_2X_1^2 + a_3X_2 + a_4X_2^2 + a_5X_1X_2 + a_6 = 0$$

五维空间：

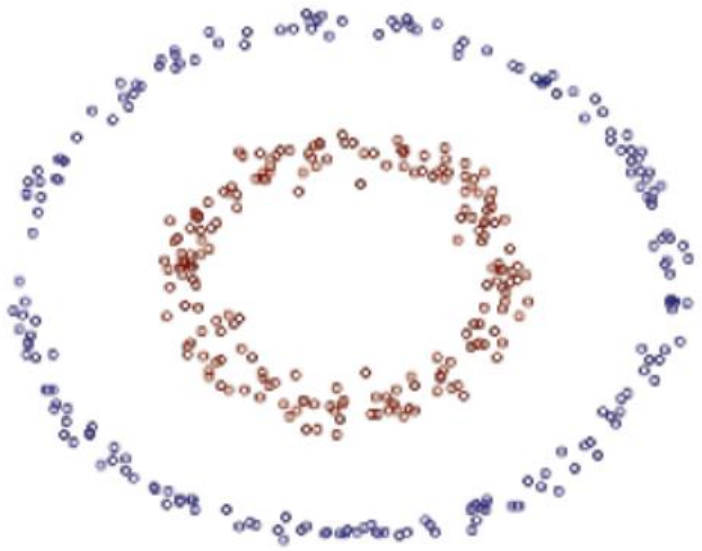
$$Z_1 = X_1, Z_2 = X_1^2, Z_3 = X_2, Z_4 = X_2^2, Z_5 = X_1X_2$$

二次曲线的方程在五维空间可以写作：

$$\sum_{i=1}^{5} a_iZ_i + a_6 = 0$$

超平面，线性可分！

核函数-解决非线性分类



线性不可分

$$\sum_{i=1}^{5} a_i Z_i + a_6 = 0$$

超平面，线性可分！

Φ：二维->五维
三维->十九维

维数灾难！

核函数-解决非线性分类

分类函数：

$$(\langle x_1, x_2\rangle + 1)^2 = 2\eta_1\xi_1 + \eta_1^2\xi_1^2 + 2\eta_2\xi_2 + \eta_2^2\xi_2^2 + 2\eta_1\eta_2\xi_1\xi_2 + 1$$

$$f(x) = (\sum_{i=1}^{n} \alpha_i y_i x_i)^T x + b$$

$$= \sum_{i=1}^{n} \alpha_i y_i \langle x_i, x\rangle + b$$

低维

核函数

隐式映射成：

$$f(x) = \sum_{i=1}^{n} \alpha_i y_i \langle \phi(x_i), \phi(x)\rangle + b$$

高维

设两个向量

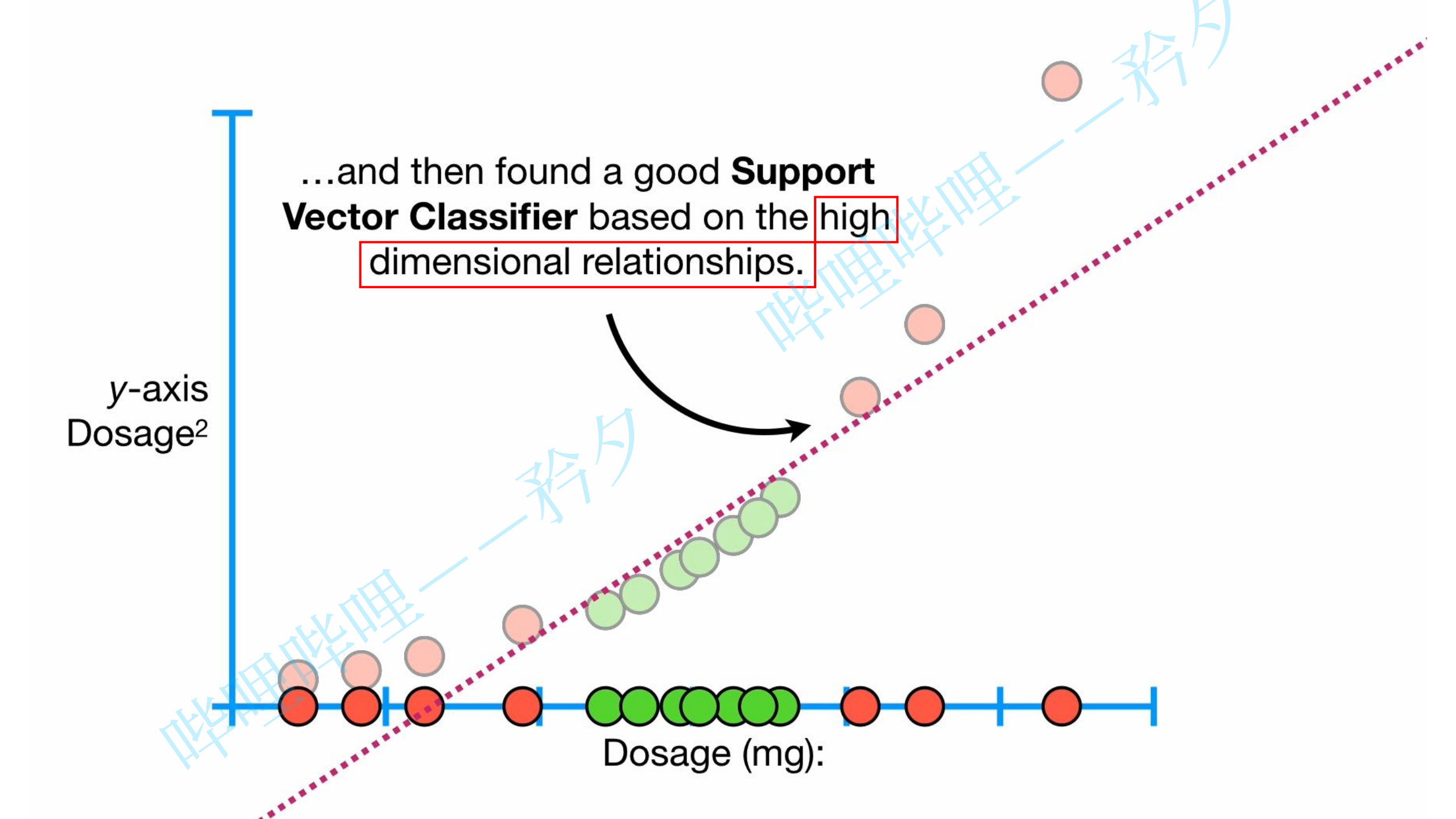
$x_1 = (\eta_1, \eta_2)^T$

$x_2 = (\xi_1, \xi_2)^T$

$$\langle \phi(x_1), \phi(x_2)\rangle = \eta_1\xi_1 + \eta_1^2\xi_1^2 + \eta_2\xi_2 + \eta_2^2\xi_2^2 + \eta_1\eta_2\xi_1\xi_2$$

$\phi(\cdot)$ 即是到前面说的五维空间的映射，因此映射过后的内积为：

The Polynomial Kernel

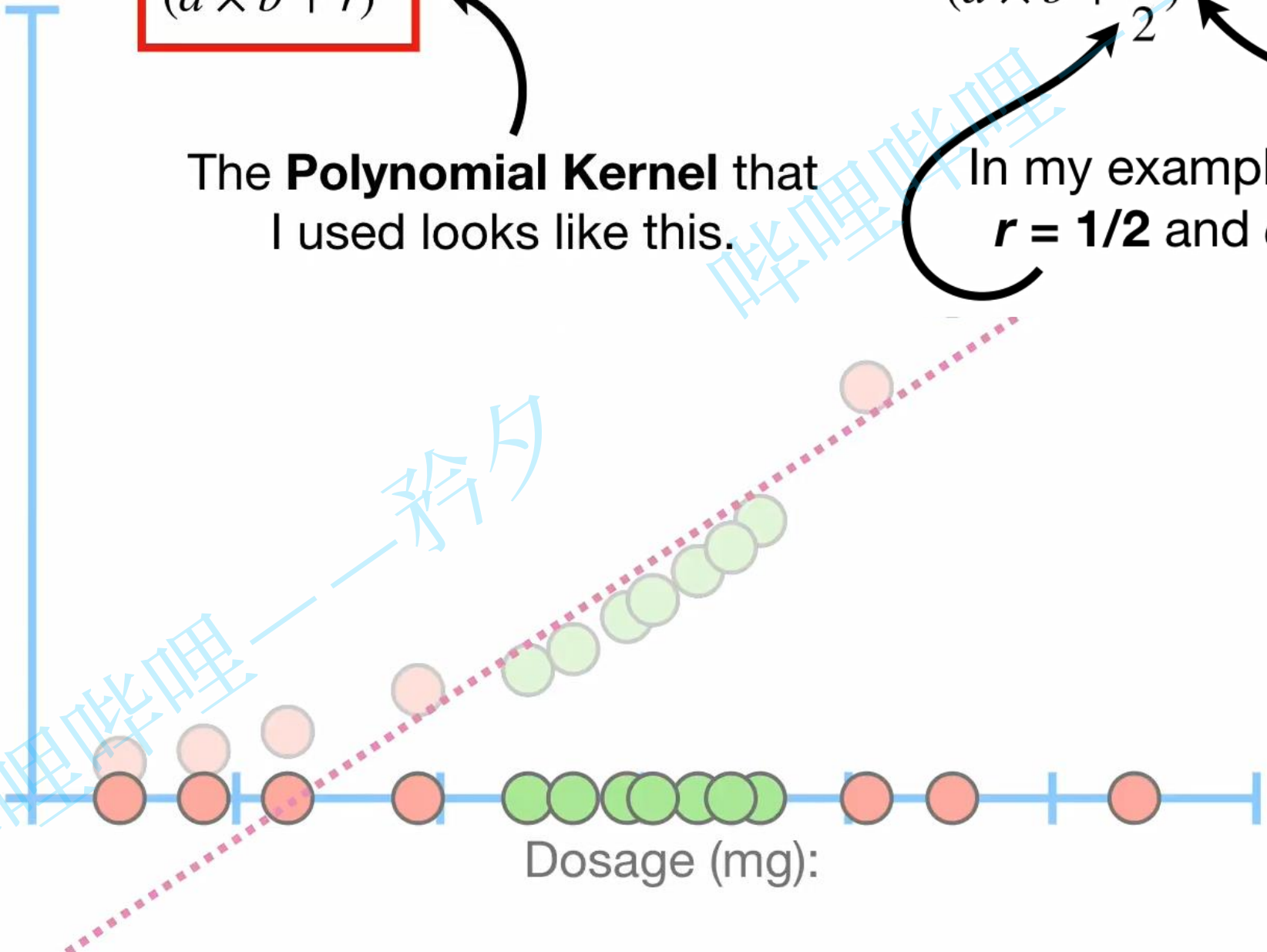

…and then found a good **Support Vector Classifier** based on the high dimensional relationships.
y-axis
Dosage²
Dosage (mg):

$$(a \times b + r)^d$$

The **Polynomial Kernel** that I used looks like this.

$$(a \times b + \frac{1}{2})^2$$

In my example, I set ***r* = 1/2** and ***d* = 2**.



$$(a \times b + \frac{1}{2})^2 = (a \times b + \frac{1}{2})(a \times b + \frac{1}{2})$$

$$= ab + a^2b^2 + \frac{1}{4}$$

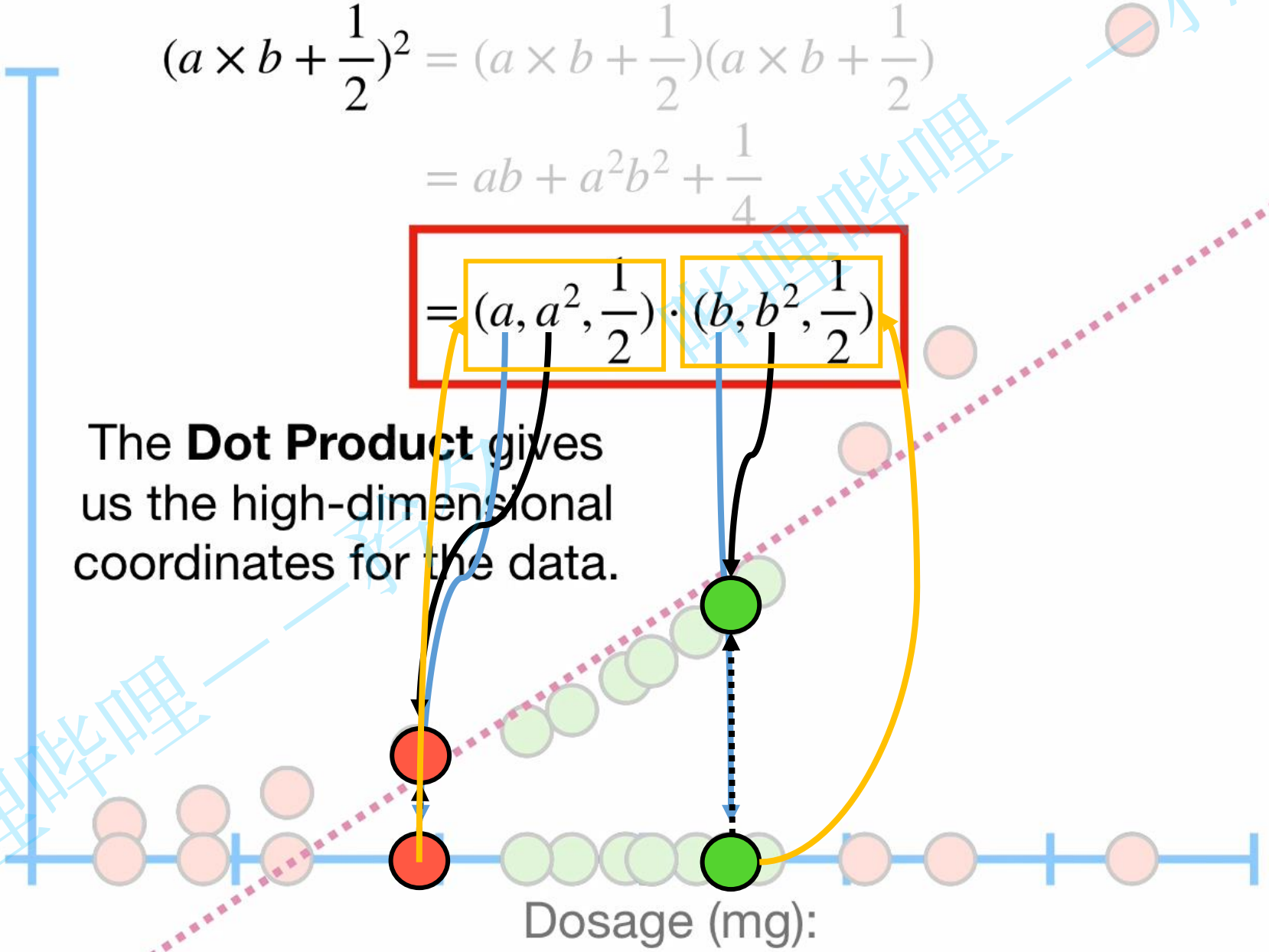
$$= (a, a^2, \frac{1}{2}) \cdot (b, b^2, \frac{1}{2})$$

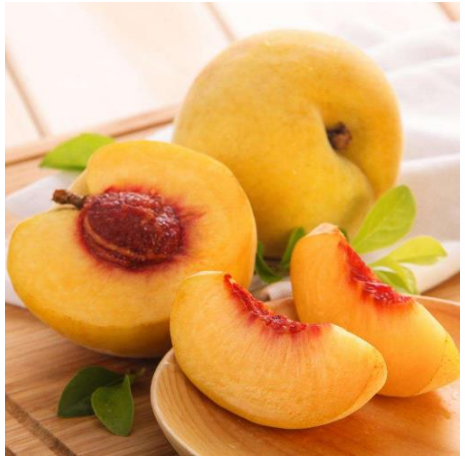
$$(a \times b + \frac{1}{2})^2 = (a \times b + \frac{1}{2})(a \times b + \frac{1}{2})$$

$$= ab + a^2b^2 + \frac{1}{4}$$

$$= (a, a^2, \frac{1}{2}) \cdot (b, b^2, \frac{1}{2})$$

The **Dot Product** gives us the high-dimensional coordinates for the data.



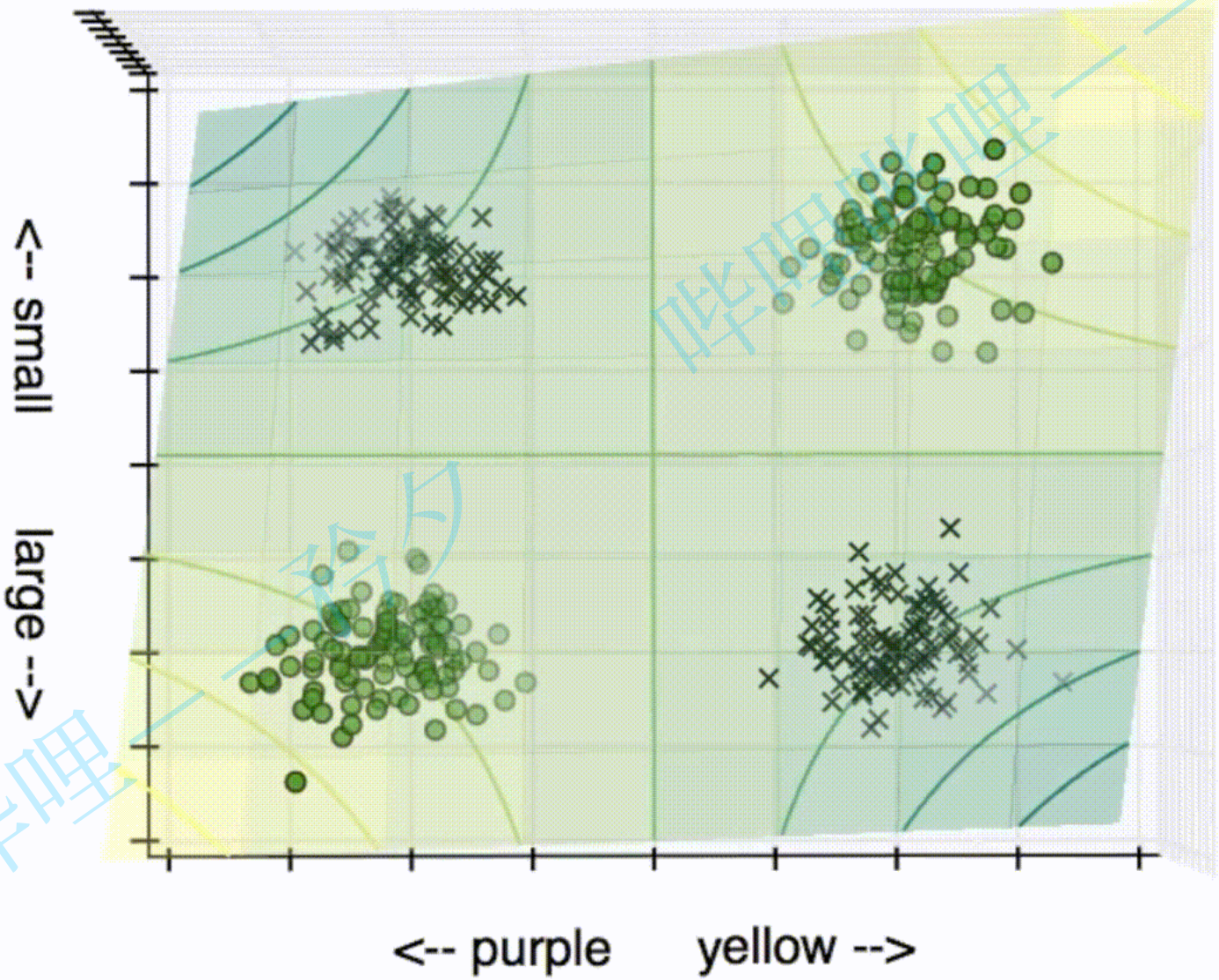


large

yellow

purple

small


<-- small
large -->
<-- purple
yellow -->

OVER