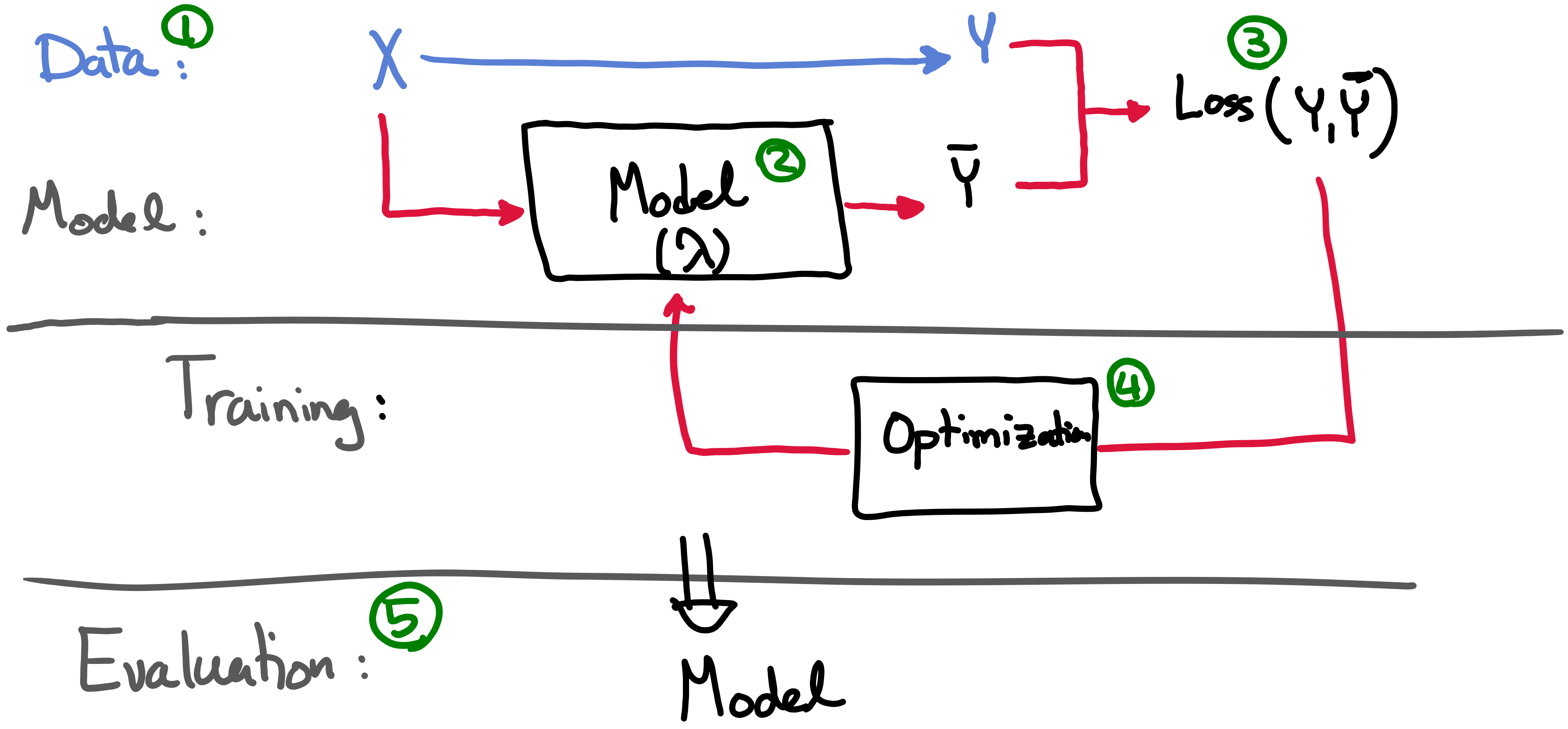
Machine Learning in Physics: Models

Sadegh Raeisi

Supervised: Ingredients



Outline

- A Simple model
- Linear vs non-linear
- Inherently non-linear models

Notation

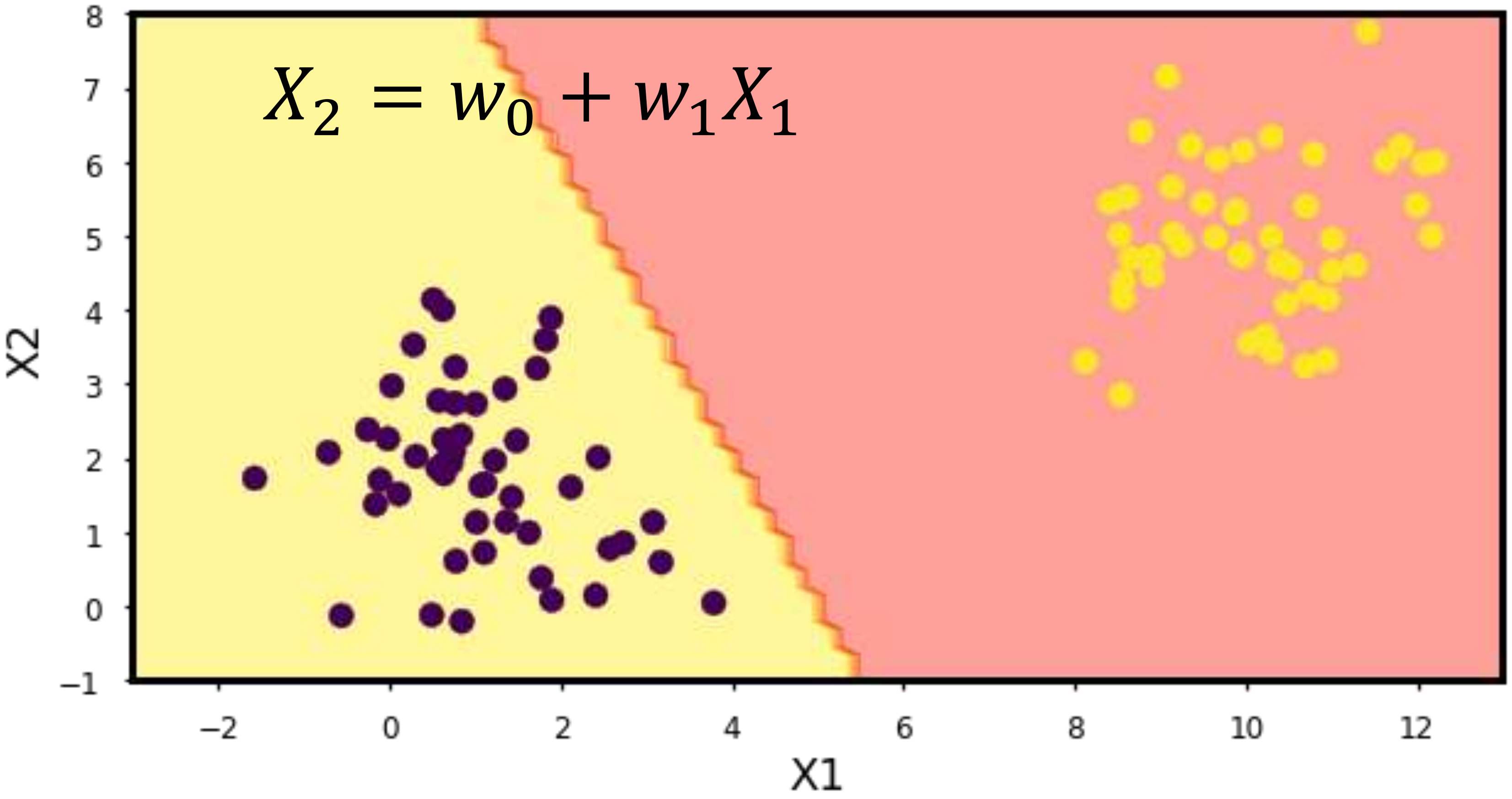
Features of the sample

Label of the sample

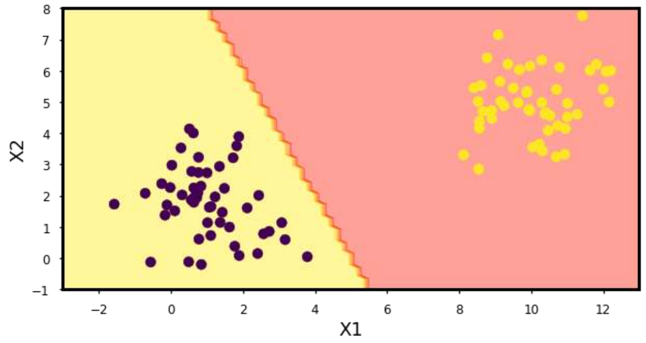
$$f_w\left(\vec{X}^i\right) = \bar{Y}$$

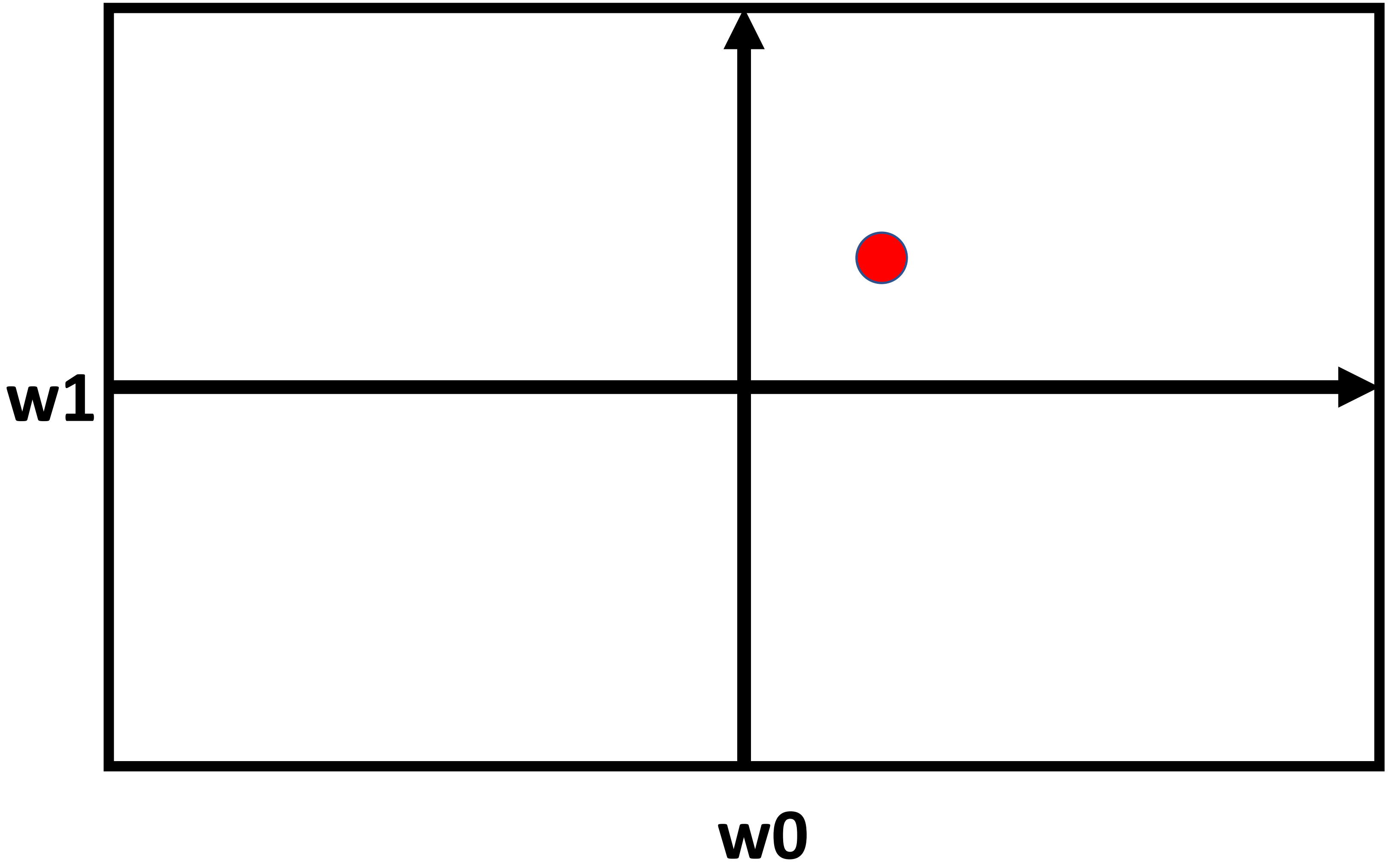
Parameters of the model

Feature space and decision boundary



Model space





A Simple Model

Simplest model possible

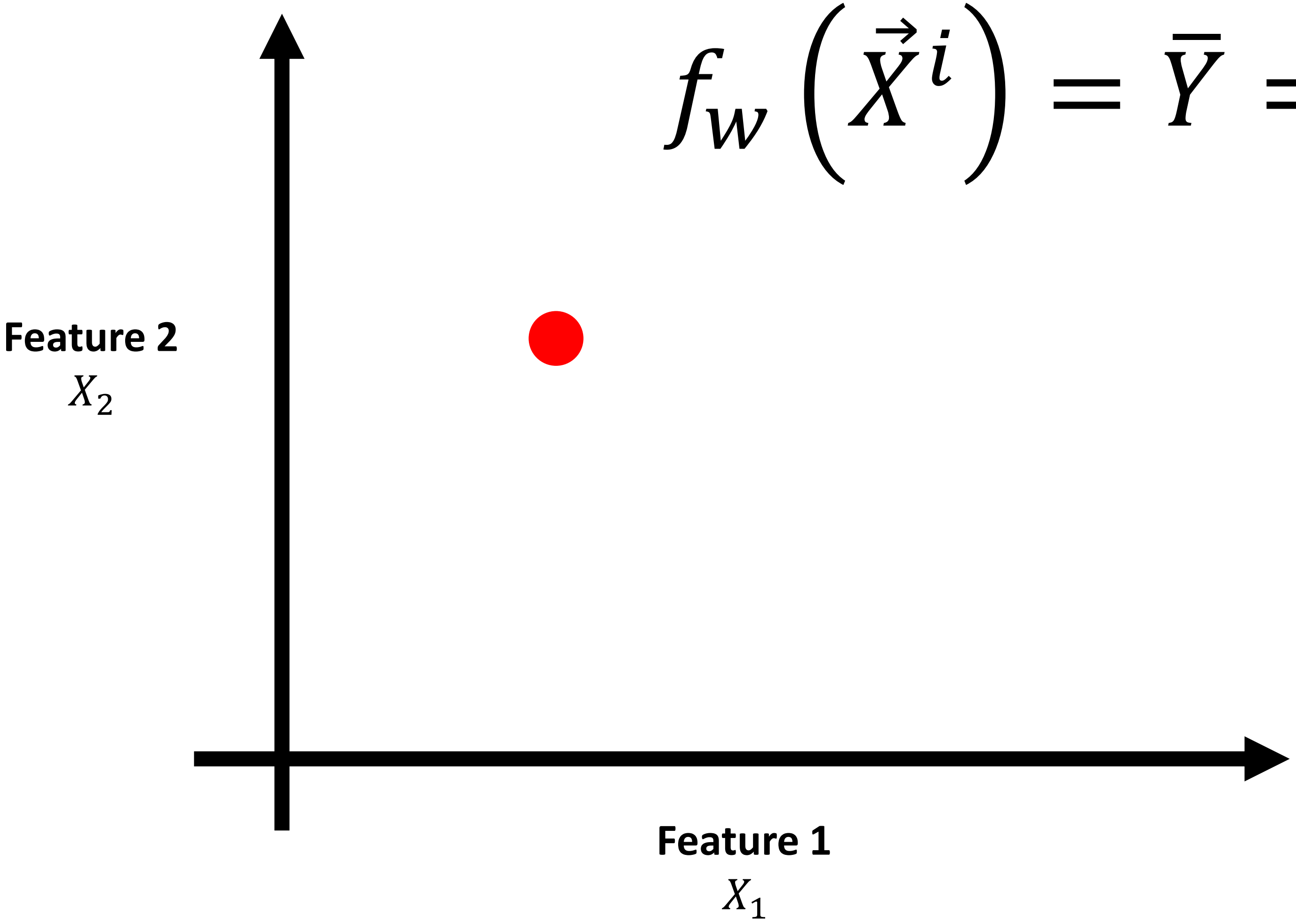
$$f_w\left(\vec{X}^i\right) = \text{const.}$$

Simplest model possible: linear model

$$f_w\left(\vec{X}^i\right) = \sum_j w_j\, X_j^{(i)}$$

One sample

$$f_w\left(\vec{X}^i\right) = \bar{Y} = w_2X_2 + w_1X_1 + w_0$$



Vectorization

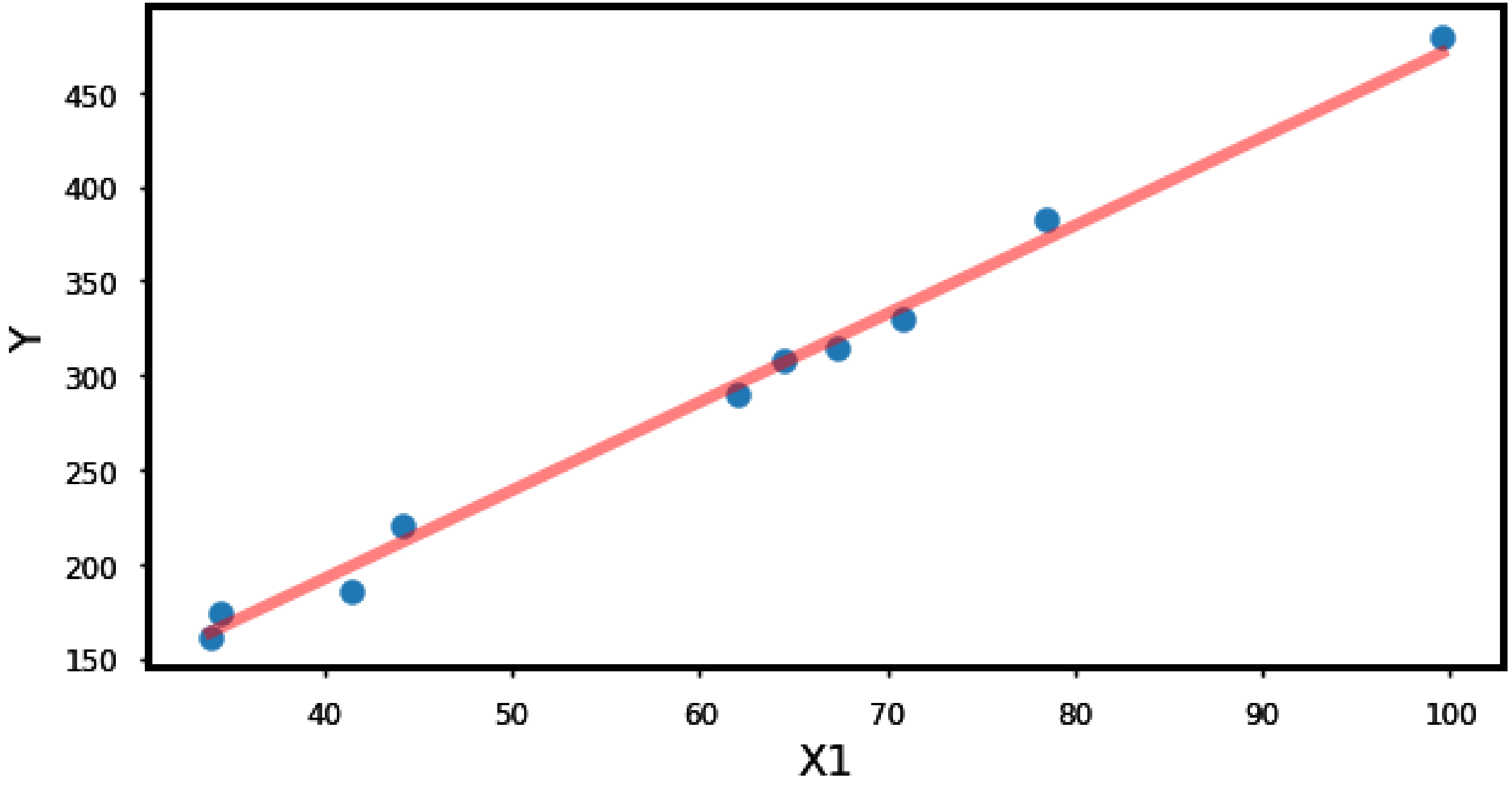
$$f_w\left(\vec{X}^i\right) = \sum_j w_j \, X_j^{(i)}$$

$$= \overrightarrow{w}.\vec{X}^i$$

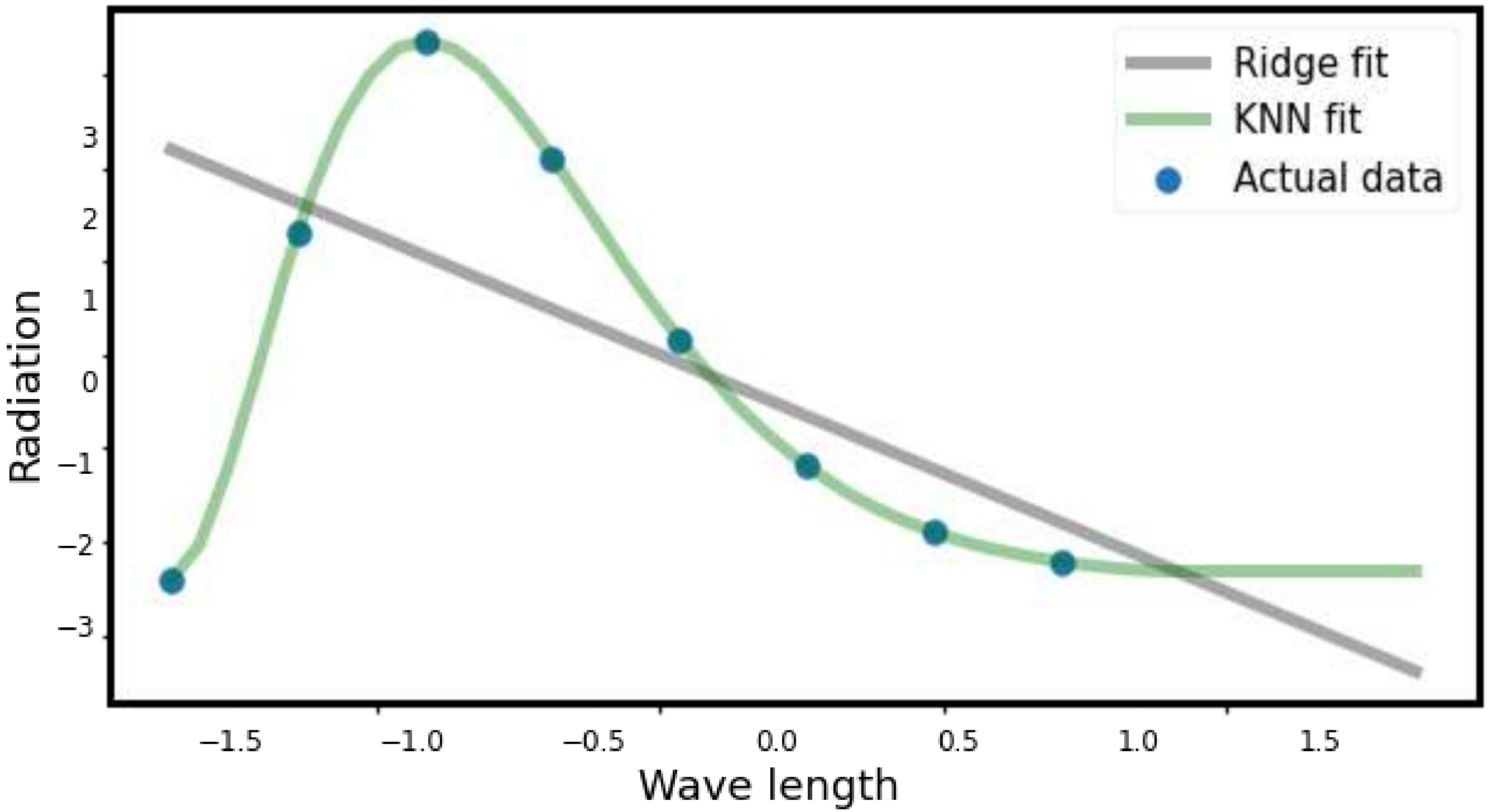
$$\overrightarrow{w} = (w_0, w_1, \cdots w_{nf})$$

$$\vec{X} = (1, X_1, \cdots X_{nf})$$

Regression

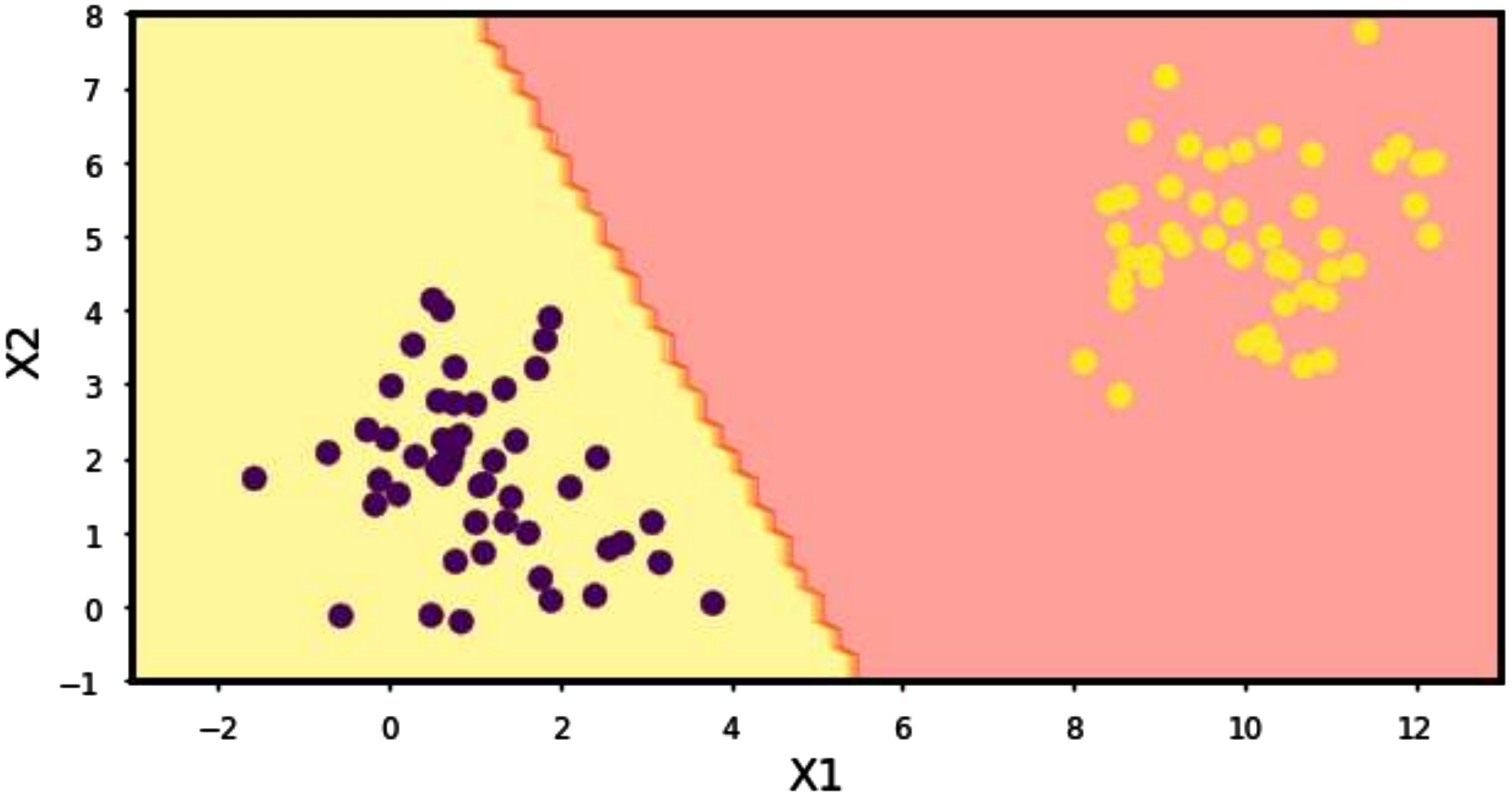


Regression



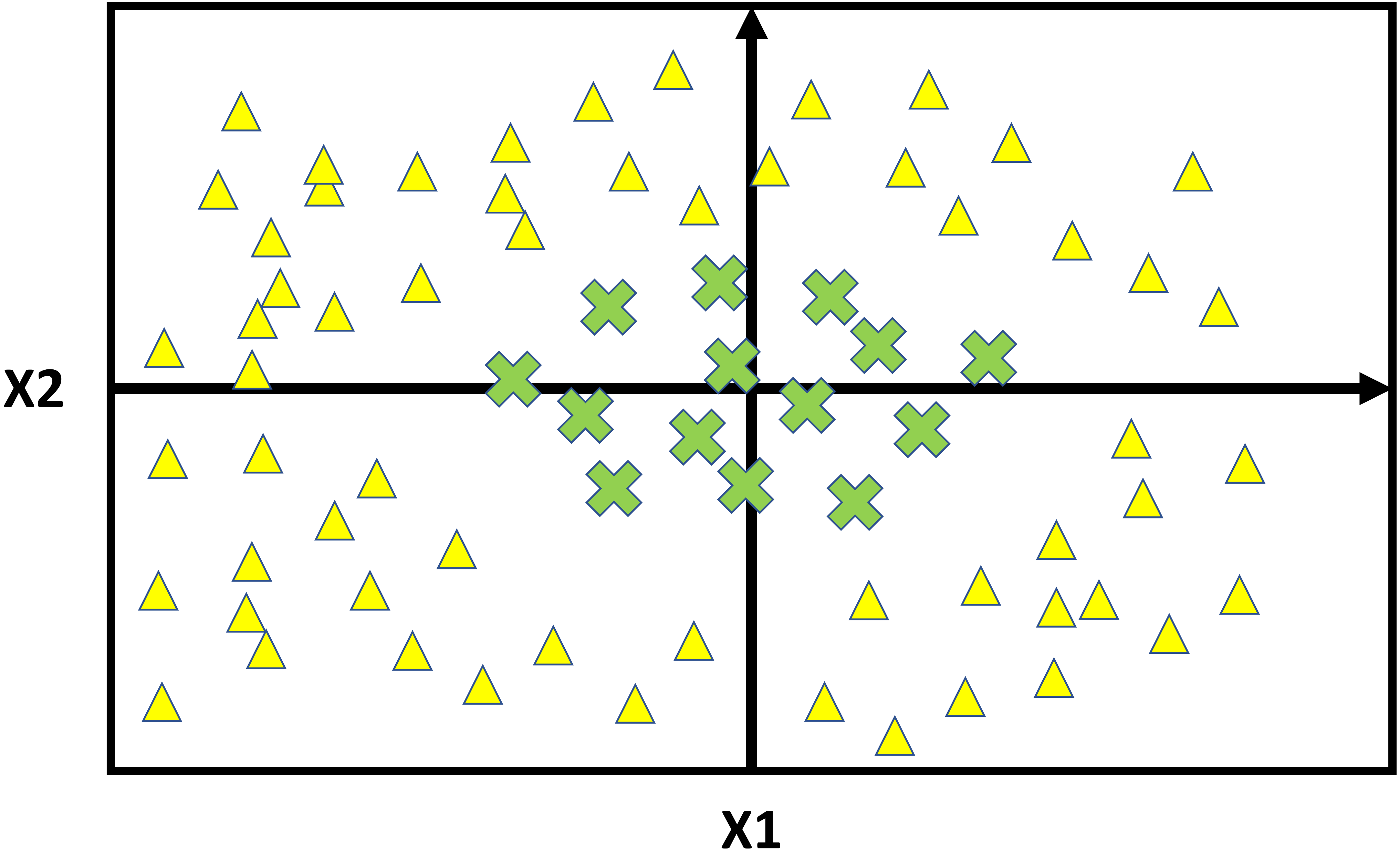
Ridge fit
KNN fit
Actual data
Radiation
3
2
1
0
−1
−2
−3
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
Wave length

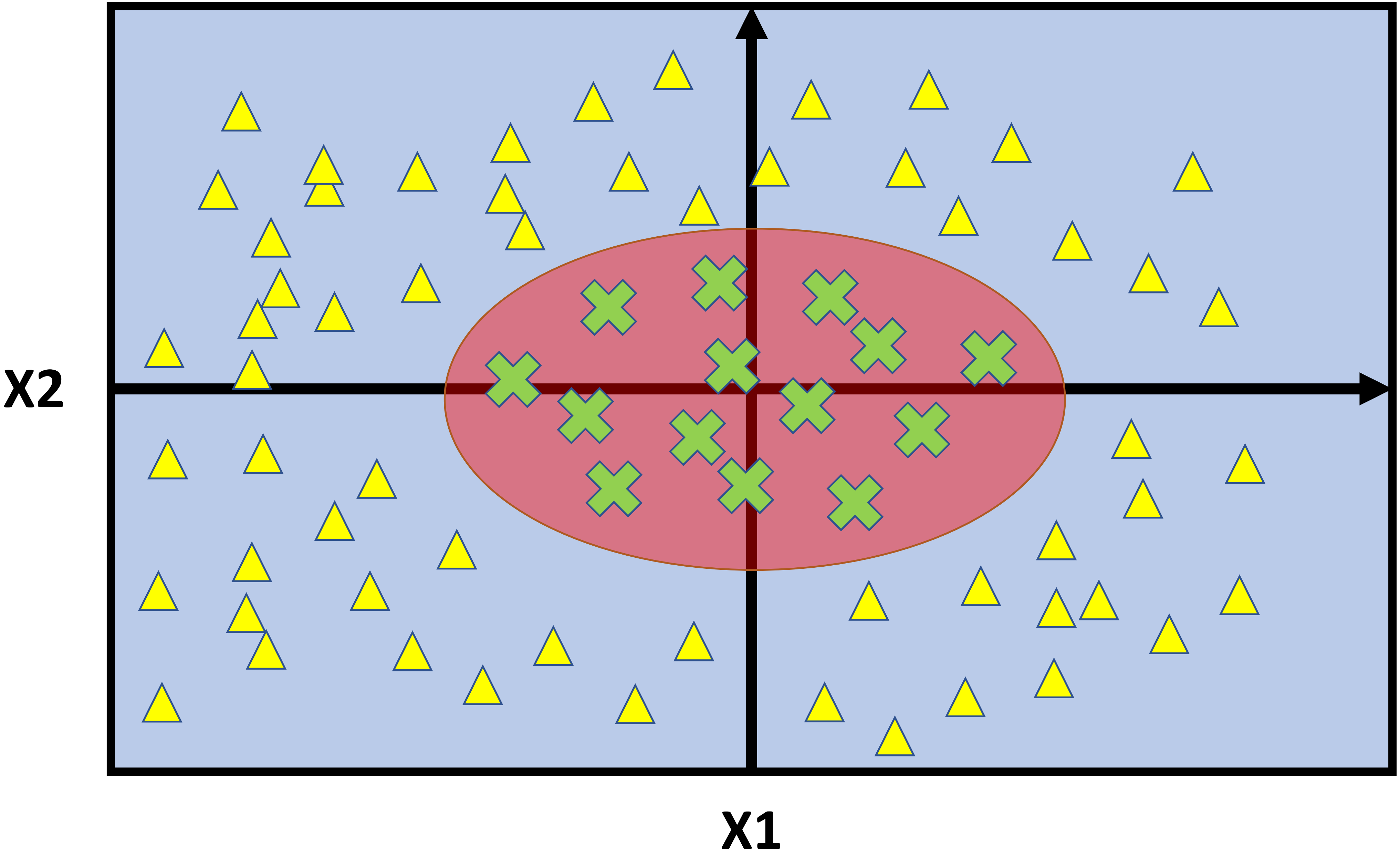
Classification



What is a good fit?

Linear vs. Non-linear


X2
X1


X2
X1

Polynomial models

How can we make a quadratic model?

Example:

$$\vec{X} = (X_1, X_2)$$

$$f_w\left(\vec{X}^i\right) = w_0 + w_1 X_1 + w_2 X_2$$

$$+\boldsymbol{w_3 X_1^2} + \boldsymbol{w_4 X_1 X_2} + \boldsymbol{w_5 X_2^2}$$

How?

1. Feature Transformation

$$\vec{X} \Rightarrow \Phi(X)$$

Example: Quadratic model

$$\vec{X} = (1, X_1, X_2) \quad \Rightarrow \Phi(X) = (1, X_1, X_2, X_1^2, X_2^2, X_1 X_2)$$

How?

2. Kernel

Often we are interested in a scalar product $\langle \nu^{(1)}, \nu^{(2)} \rangle$

$$
\begin{tikzcd}
\boldsymbol{\nu}^{(1)}, \boldsymbol{\nu}^{(2)} \arrow[r] \arrow[dr] & \boldsymbol{\Phi}(\boldsymbol{\nu}^{(1)}), \boldsymbol{\Phi}(\boldsymbol{\nu}^{(2)}) \arrow[r] & \langle \boldsymbol{\Phi}(\boldsymbol{\nu}^{(1)}), \boldsymbol{\Phi}(\boldsymbol{\nu}^{(2)}) \rangle \\
& \boldsymbol{K}(\boldsymbol{\nu}^{(1)}, \boldsymbol{\nu}^{(2)}) = \langle \boldsymbol{\Phi}(\boldsymbol{\nu}^{(1)}), \boldsymbol{\Phi}(\boldsymbol{\nu}^{(2)}) \rangle &
\end{tikzcd}
$$

Example:

$$\Phi\left(\begin{bmatrix} X_1 \\ X_2 \end{bmatrix}\right) = \begin{bmatrix} X_1^2 \\ X_2^2 \\ \sqrt{2}X_1X_2 \end{bmatrix}$$

$$\left\langle \boldsymbol{\Phi}\left(\begin{bmatrix} X_1 \\ X_2 \end{bmatrix}\right), \boldsymbol{\Phi}\left(\begin{bmatrix} Z_1 \\ Z_2 \end{bmatrix}\right) \right\rangle$$

Example:

$$\Phi\left(\begin{bmatrix} X_1 \\ X_2 \end{bmatrix}\right) = \begin{bmatrix} X_1^2 \\ X_2^2 \\ \sqrt{2}X_1X_2 \end{bmatrix}$$

$$\left\langle \boldsymbol{\Phi}\left(\begin{bmatrix} X_1 \\ X_2 \end{bmatrix}\right), \boldsymbol{\Phi}\left(\begin{bmatrix} Z_1 \\ Z_2 \end{bmatrix}\right) \right\rangle = \left\langle \begin{bmatrix} X_1^2 \\ X_2^2 \\ \sqrt{2}X_1X_2 \end{bmatrix}, \begin{bmatrix} Z_1^2 \\ Z_2^2 \\ \sqrt{2}Z_1Z_2 \end{bmatrix} \right\rangle$$

Example:

$$\Phi\left(\begin{bmatrix} X_1 \\ X_2 \end{bmatrix}\right) = \begin{bmatrix} X_1^2 \\ X_2^2 \\ \sqrt{2}X_1X_2 \end{bmatrix}$$

$$\left\langle \boldsymbol{\Phi}\left(\begin{bmatrix} X_1 \\ X_2 \end{bmatrix}\right), \boldsymbol{\Phi}\left(\begin{bmatrix} Z_1 \\ Z_2 \end{bmatrix}\right) \right\rangle = \left\langle \begin{bmatrix} X_1^2 \\ X_2^2 \\ \sqrt{2}X_1X_2 \end{bmatrix}, \begin{bmatrix} Z_1^2 \\ Z_2^2 \\ \sqrt{2}Z_1Z_2 \end{bmatrix} \right\rangle$$

$$= \boldsymbol{X_1^2Z_1^2 + X_2^2Z_2^2 + 2X_1X_2Z_1Z_2}$$

Example:

$$\mathrm{K}(X, Z) = \langle X, Z \rangle^2$$

$$\langle X, Z \rangle^2 = \left\langle \begin{bmatrix} X_1 \\ X_2 \end{bmatrix}, \begin{bmatrix} Z_1 \\ Z_2 \end{bmatrix} \right\rangle^{\mathbf{2}}$$

Example:

$$\mathrm{K}(X, Z) = \langle X, Z \rangle^2$$

$$\langle X, Z \rangle^2 = \left\langle \begin{bmatrix} X_1 \\ X_2 \end{bmatrix}, \begin{bmatrix} Z_1 \\ Z_2 \end{bmatrix} \right\rangle^{\mathbf{2}} = (\boldsymbol{X_1 Z_1} + \boldsymbol{X_2 Z_2})^{\mathbf{2}}$$

Example:

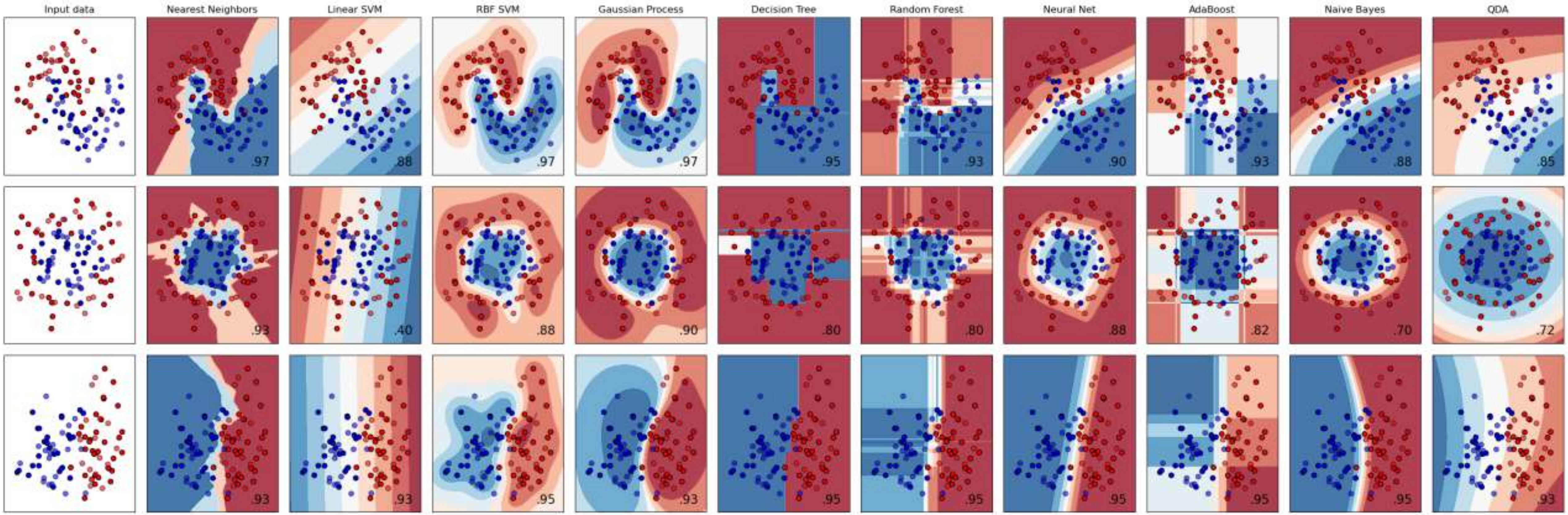
$$\mathrm{K}(X,Z) = \langle X,Z\rangle^2$$

$$\langle X,Z\rangle^2 = \left\langle \begin{bmatrix} X_1 \\ X_2 \end{bmatrix}, \begin{bmatrix} Z_1 \\ Z_2 \end{bmatrix} \right\rangle^{\mathbf{2}} = (\boldsymbol{X_1 Z_1} + \boldsymbol{X_2 Z_2})^{\mathbf{2}}$$

$$= \boldsymbol{X_1^2 Z_1^2} + \boldsymbol{X_2^2 Z_2^2} + \mathbf{2}\boldsymbol{X_1 X_2 Z_1 Z_2} = \left\langle \boldsymbol{\Phi}\left(\begin{bmatrix} X_1 \\ X_2 \end{bmatrix}\right), \boldsymbol{\Phi}\left(\begin{bmatrix} Z_1 \\ Z_2 \end{bmatrix}\right) \right\rangle$$

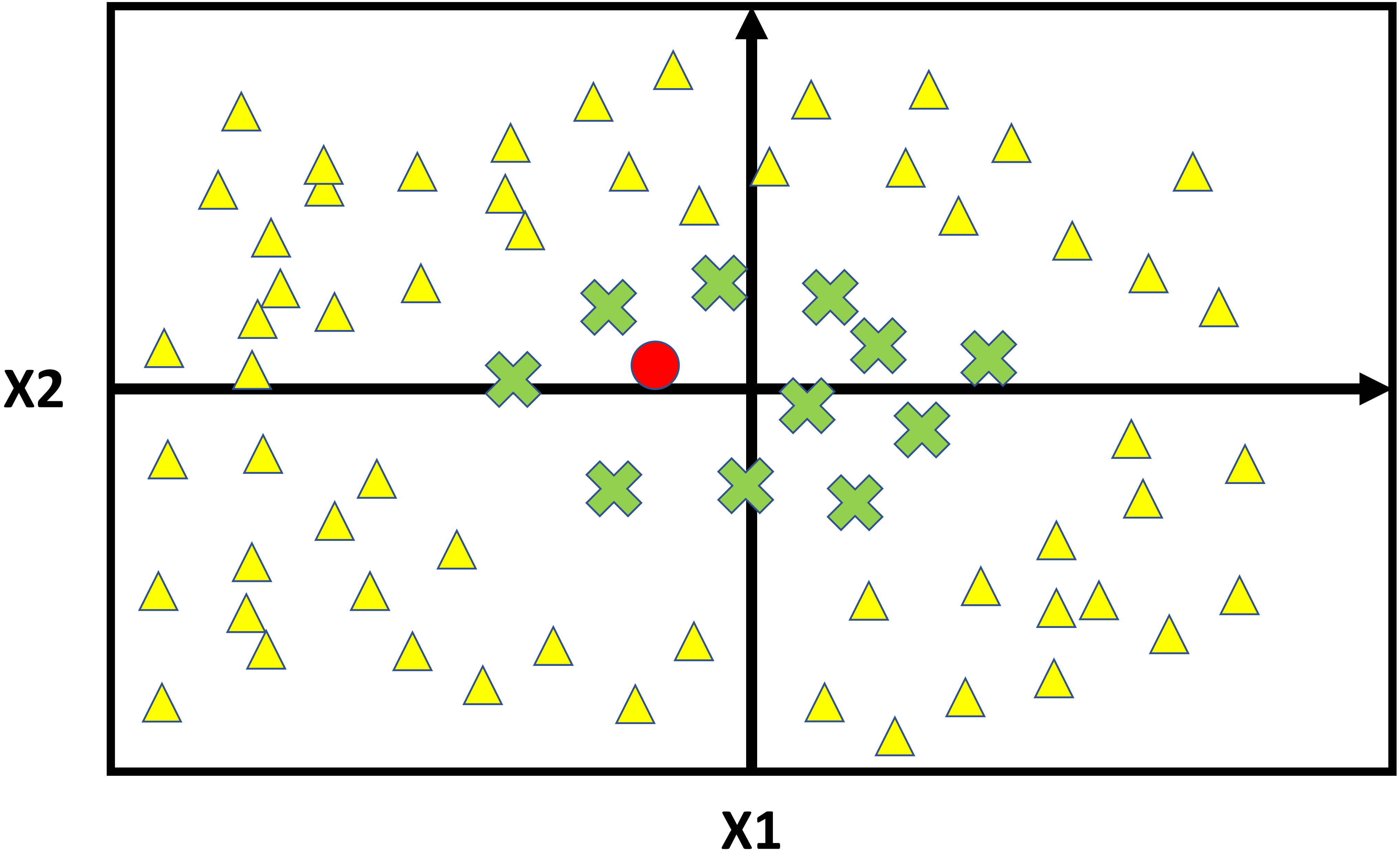
Inherently non-linear models

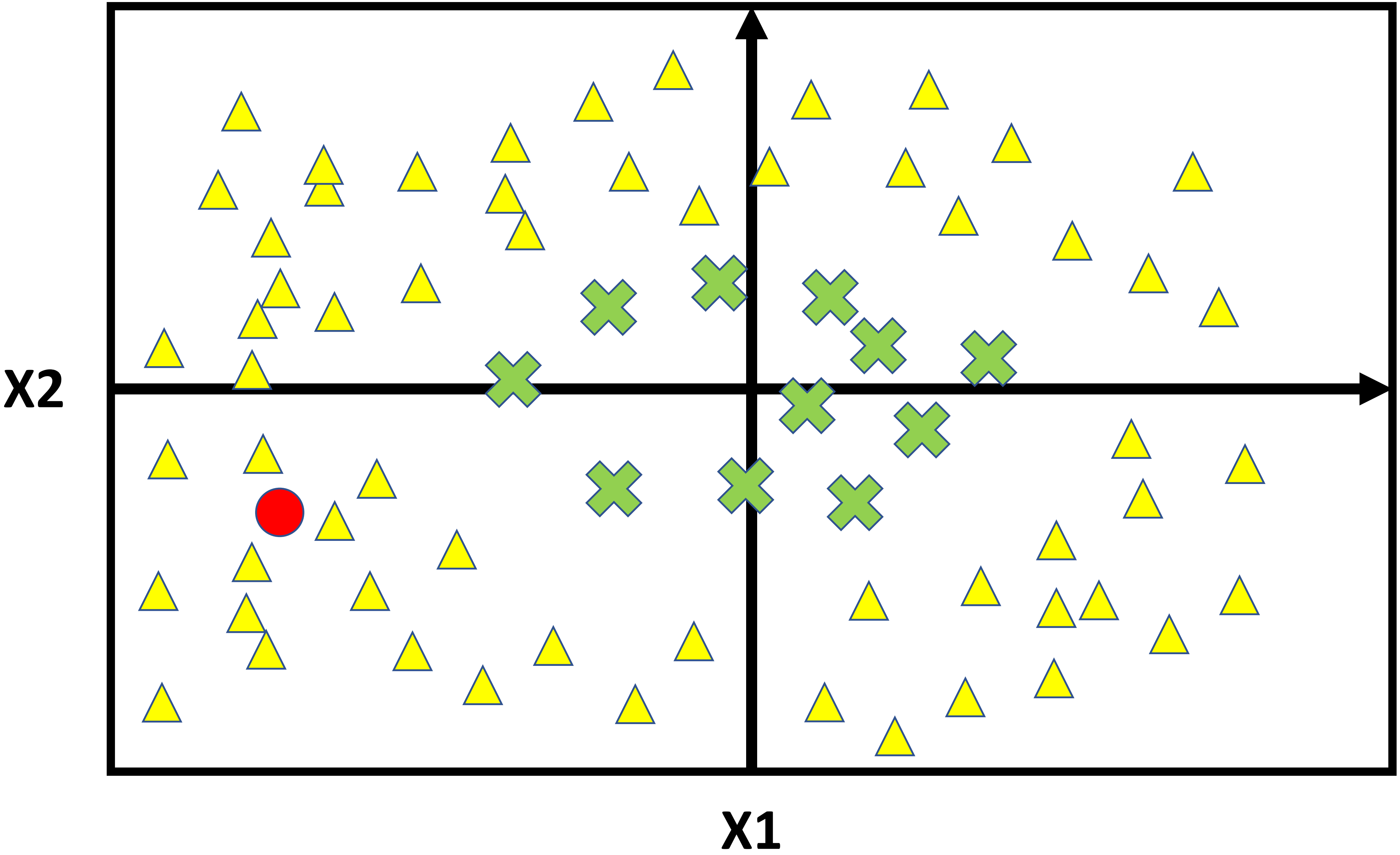
There are many different techniques …

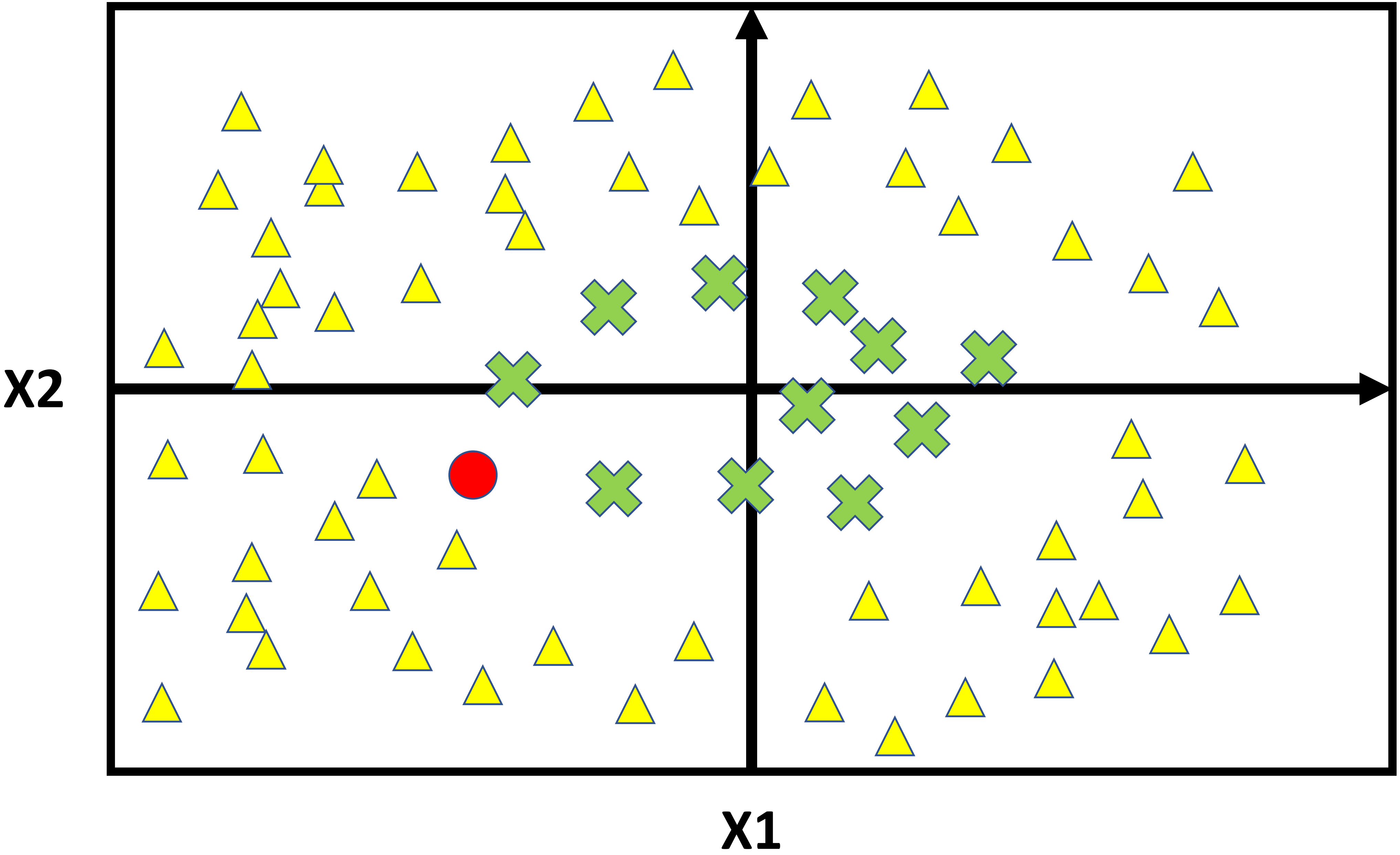


[Classifier comparison — scikit-learn 1.0 documentation]()

K Nearest neighbours KNN


X2
X1


X2
X1


X2
X1

Variables of the KNN model

- How many neighbours?
- Policy?
  - Majority
  - Weighted distance
- Metric

Learning vs memorizing

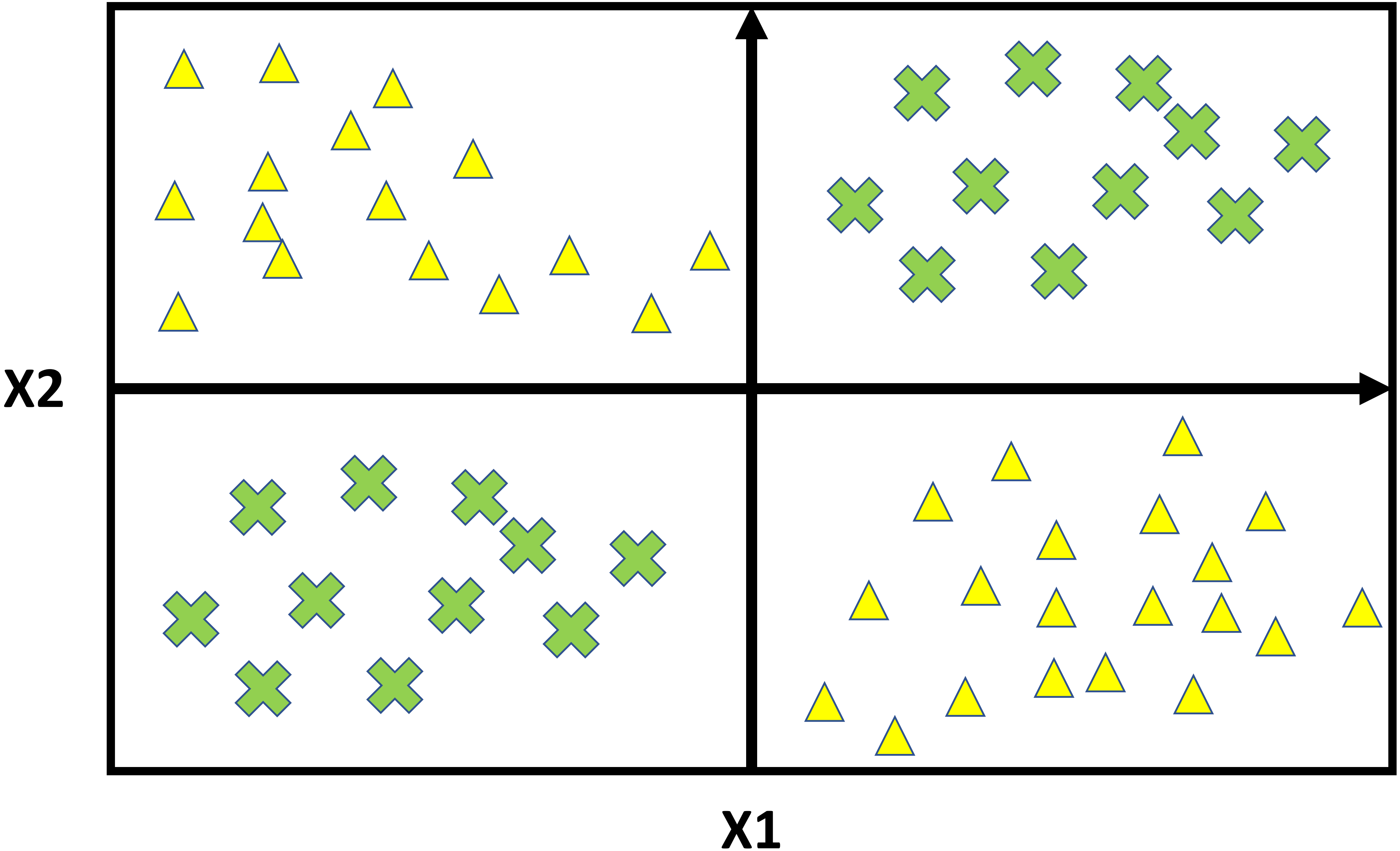
- Training is roughly equivalent to storing all the data points
- Prediction:
  - Cross-checking the input with the stored data points.

What happens if

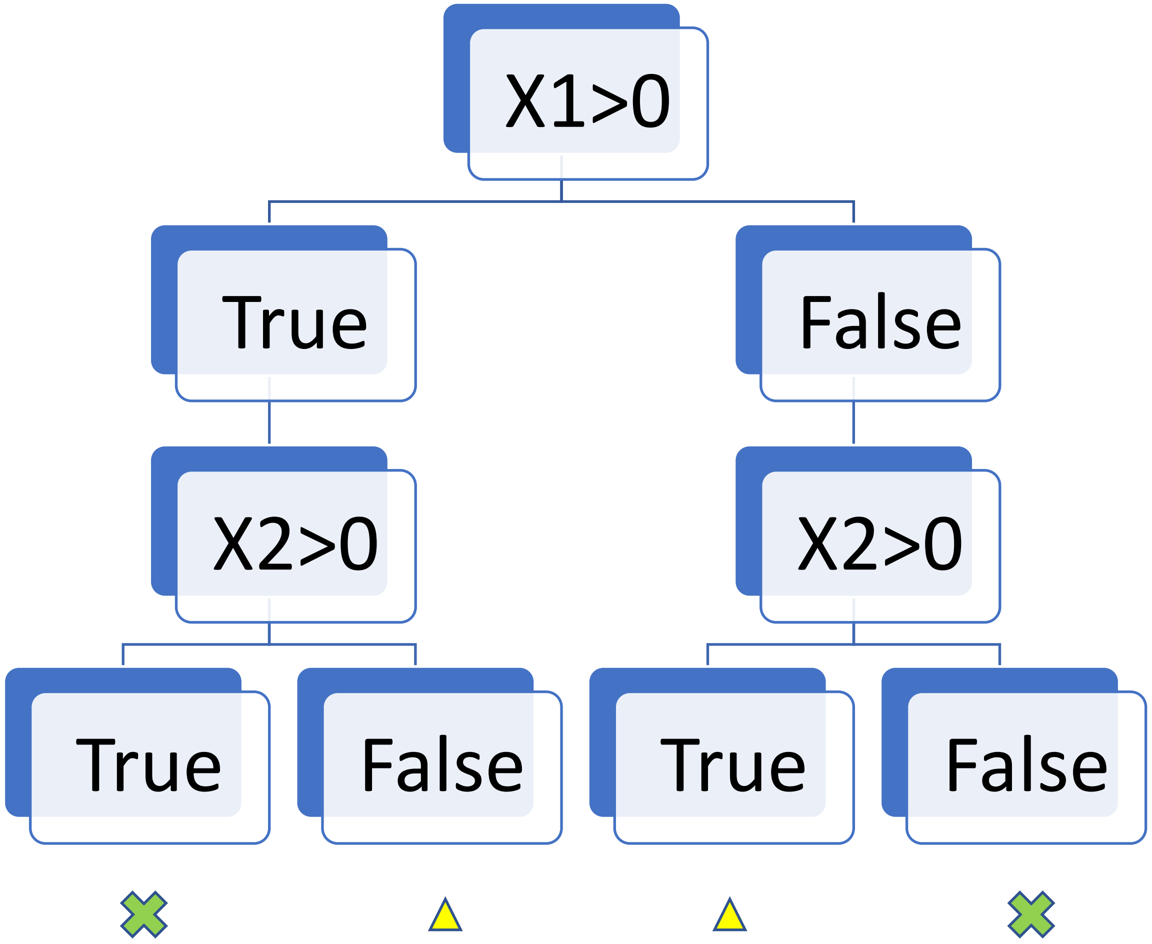
- $k \to n_s$?

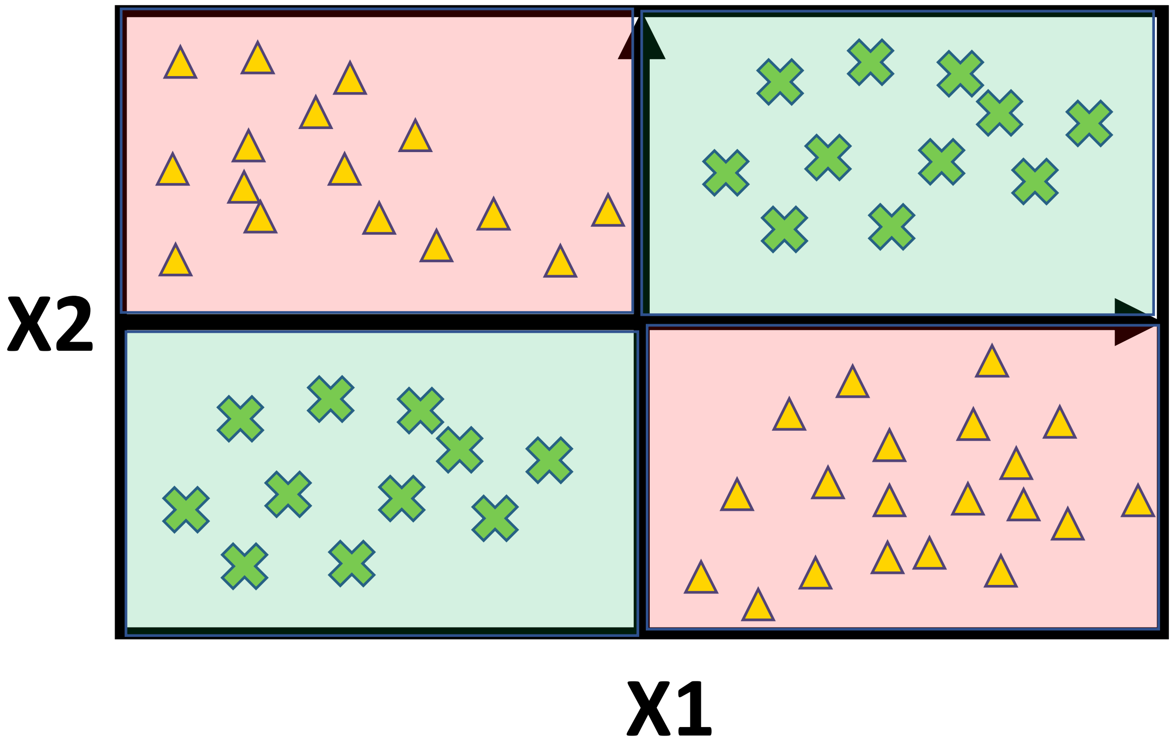
- $k \to 1$?

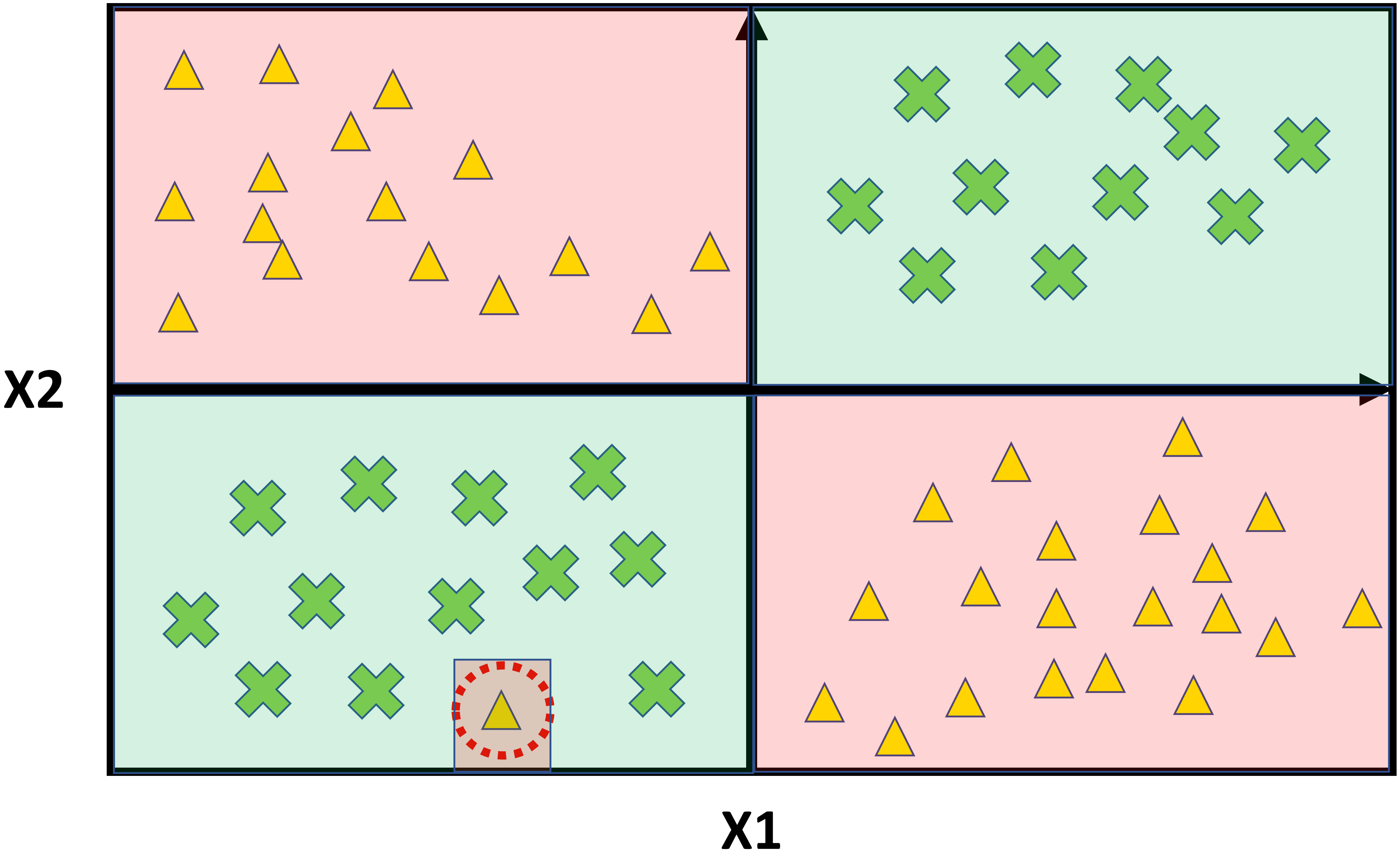
Decision Trees


X2
X1

Decision tree






X2
X1

What is the main variable?

Depth of the tree

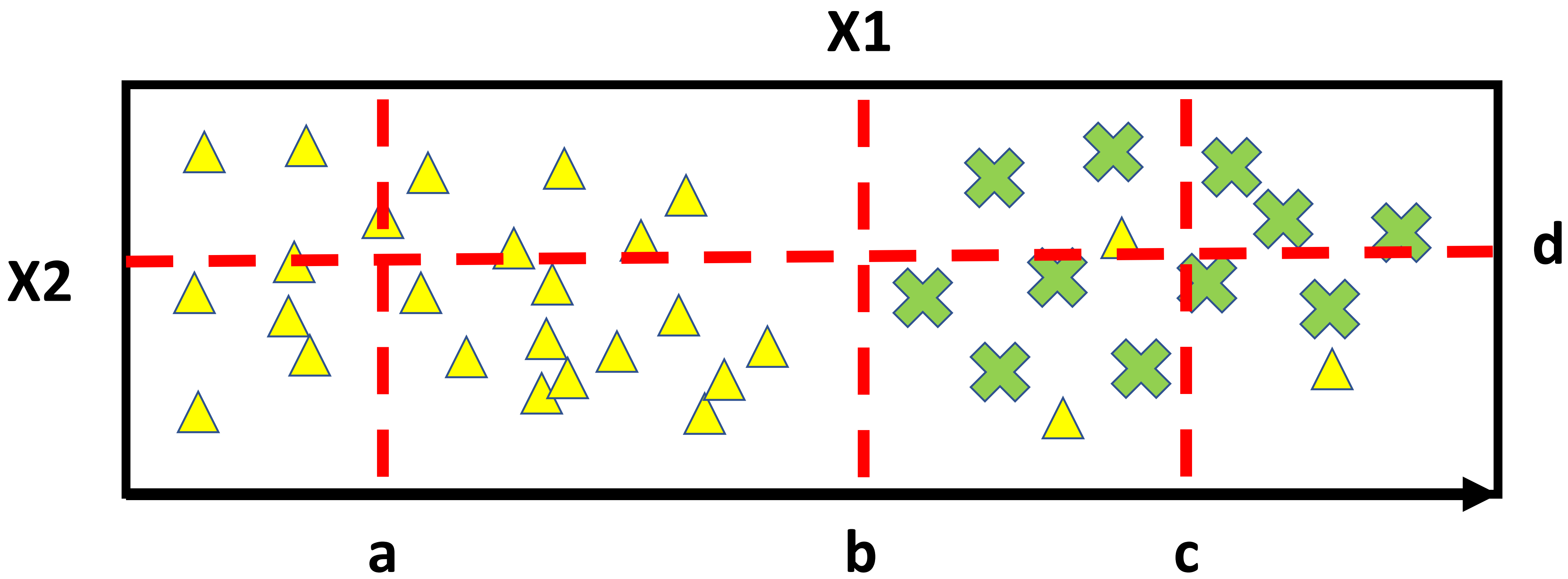
What happens if $depth \to \infty$?

What are we optimizing?
What is the objective?

- What is the most informative questions to ask?

  - Information gain

  - Variance reduction



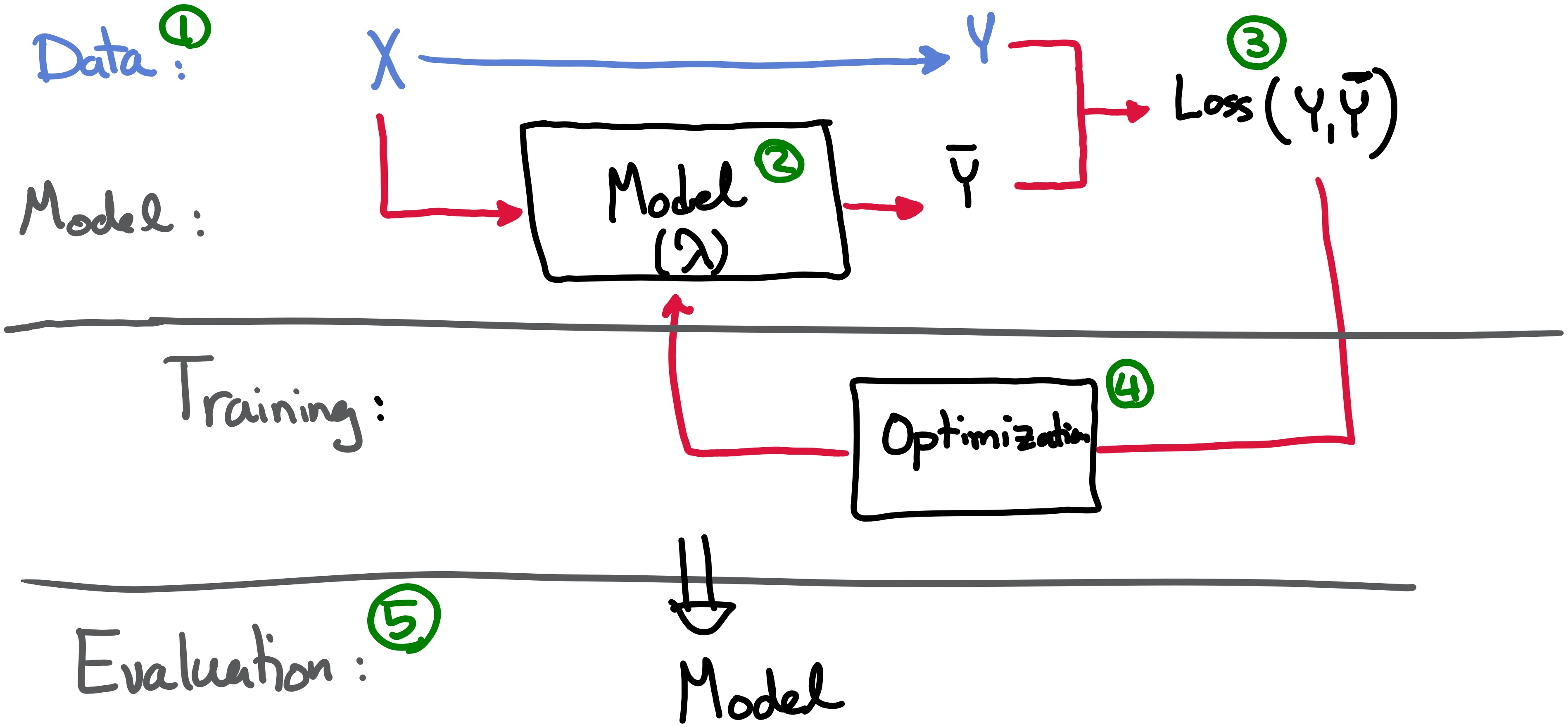
X2 > d

X1 > a

X1 > b

X1 > c

So far ...


Data:
1
X
Y
Model:
Model
(λ)
2
Ȳ
3
Loss(Y,Ȳ)
Training:
Optimization
4
Evaluation:
5
Model