

# Apprentissage Automatique I MATH80629

**Summary**  
Supervised Learning  
— Week #3

# Apprentissage Automatique I

## MATH80629

Apprentissage supervisé — **Sommaire**  
— Semaine #3

# Modèles pour l'apprentissage supervisé

- Modèles linéaires (surtout)
- Classification
  1. Modèles non probabilistes
    - Plus proche voisin (K-NN), Machine à vecteurs de support (SVMs)
  2. Modèles Probabilistes
    - Naive Bayes

# Apprentissage supervisé

Ensemble  
Entraînement

	Nb.bed.	Area	Neigh.	.	.		Sell-ability
$x_0$	1	0	0	0	0	$y_0$	1
$x_1$	1	100	1	.2	.5	$y_1$	2
$x_2$	3	200	0	.1	.2	$y_2$	0
$x_3$	1	150	1	.4	.1	$y_3$	2
$x_4$	2	210	2	.5	1.1	$y_4$	1

$X$   $Y$

Tâche

$$f : \mathbb{R}^n \rightarrow \{0, 1, 2\}$$

Ensemble  
Test

	Nb.bed.	Area	Neigh.	.	.		Sell-ability
$x_0$	1	0	0	0	0	$y_0$	?
$x_1$	2	50	1	.3	.8	$y_1$	?
$x_2$	1	100	1	.5	1.4	$y_2$	?
$x_3$	4	170	0	.7	.4	$y_3$	?
$x_4$	1	120	3	.9	.5	$y_4$	?

$X^{\text{new}}$   $Y^{\text{new}}$

# Apprentissage supervisé

Ensemble  
Entraînement

	Nb.bed.	Area	Neigh.	.	.		Sell-ability
$x_0$	1	0	0	0	0	$y_0$	1
$x_1$	1	100	1	.2	.5	$y_1$	2
$x_2$	3	200	0	.1	.2	$y_2$	0
$x_3$	1	150	1	.4	.1	$y_3$	2
$x_4$	2	210	2	.5	1.1	$y_4$	1

$X$   $Y$

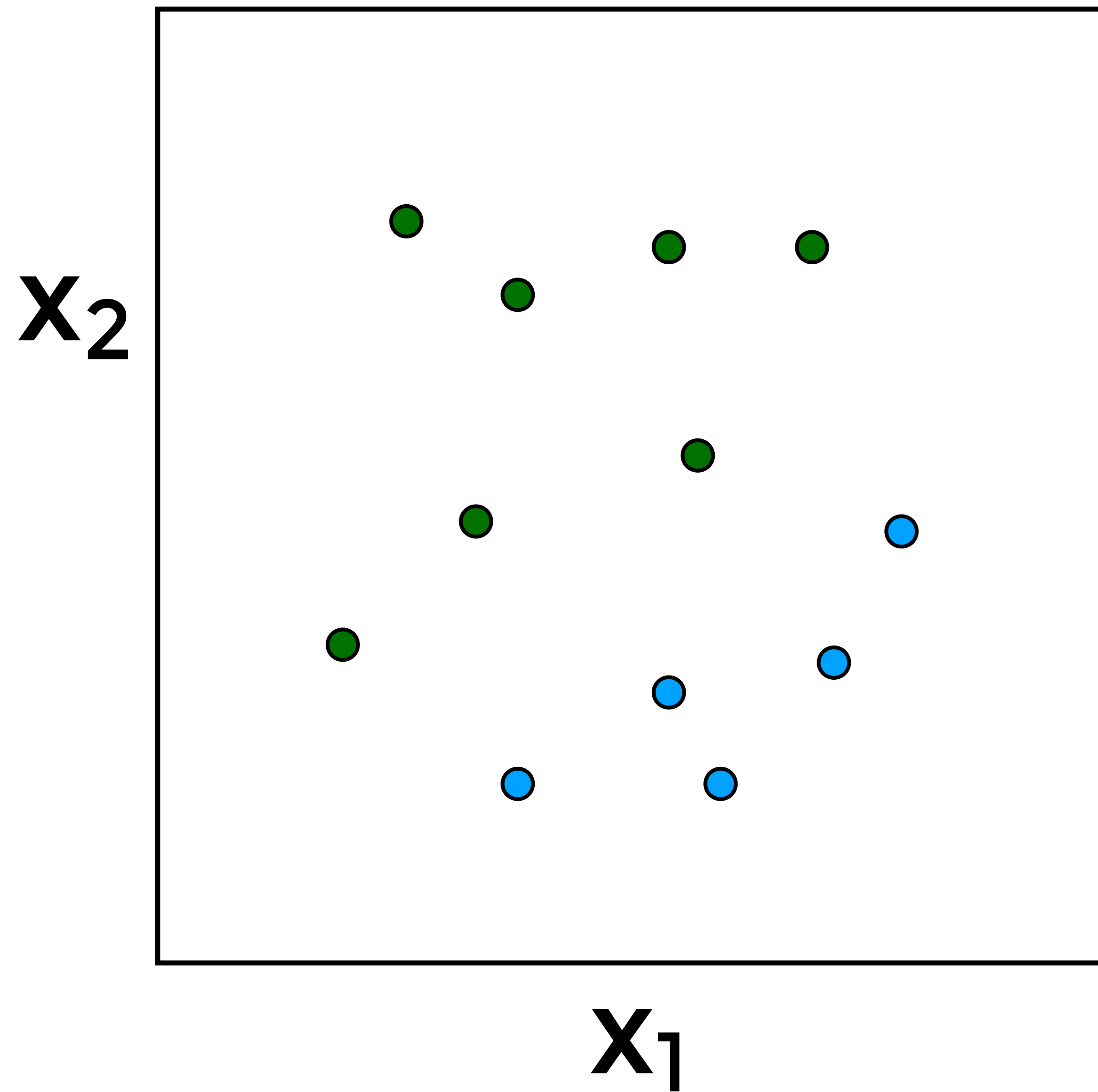
Tâche

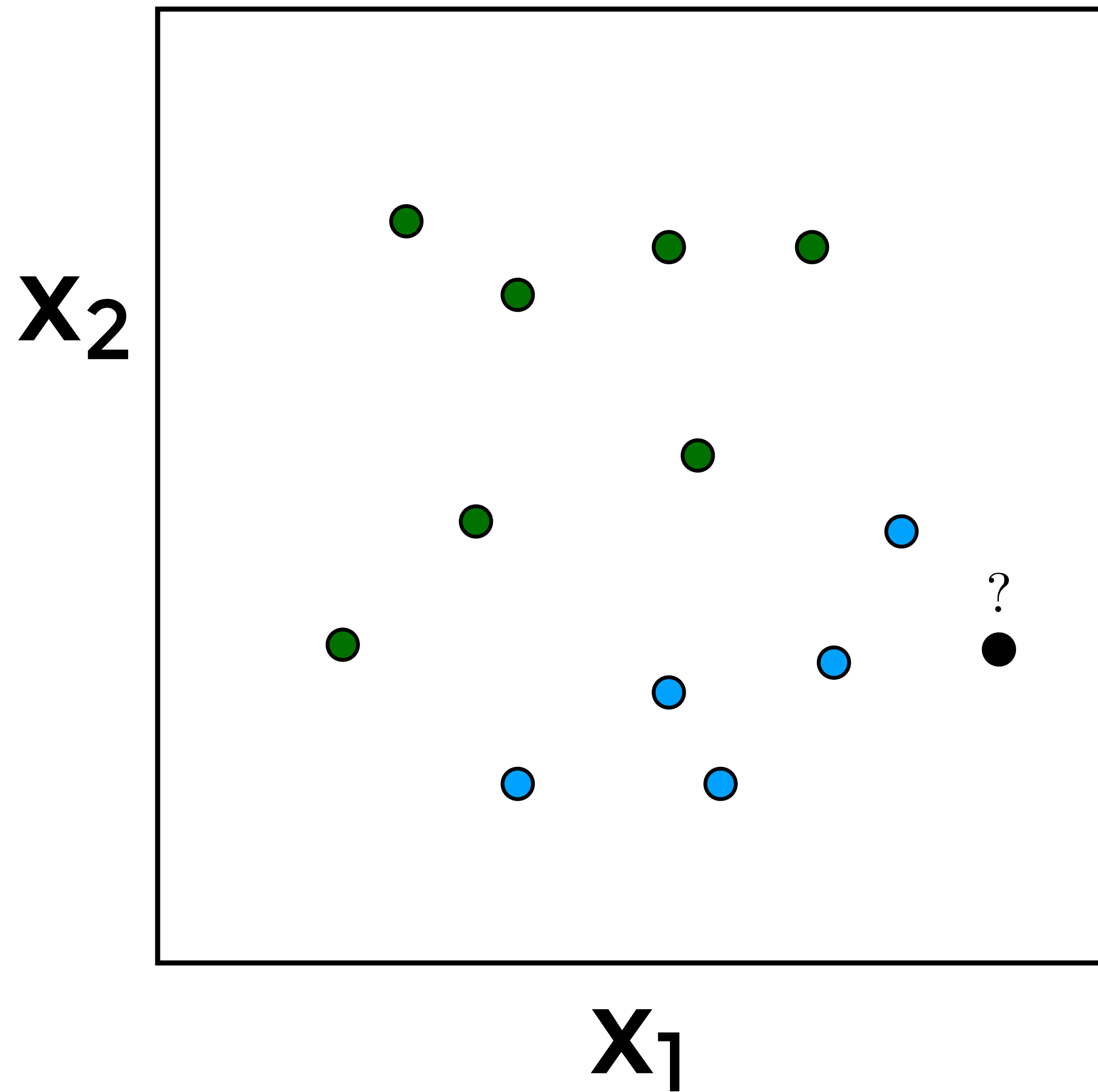
Modèle  $f: \mathbb{R}^n \rightarrow \{0, 1, 2\}$

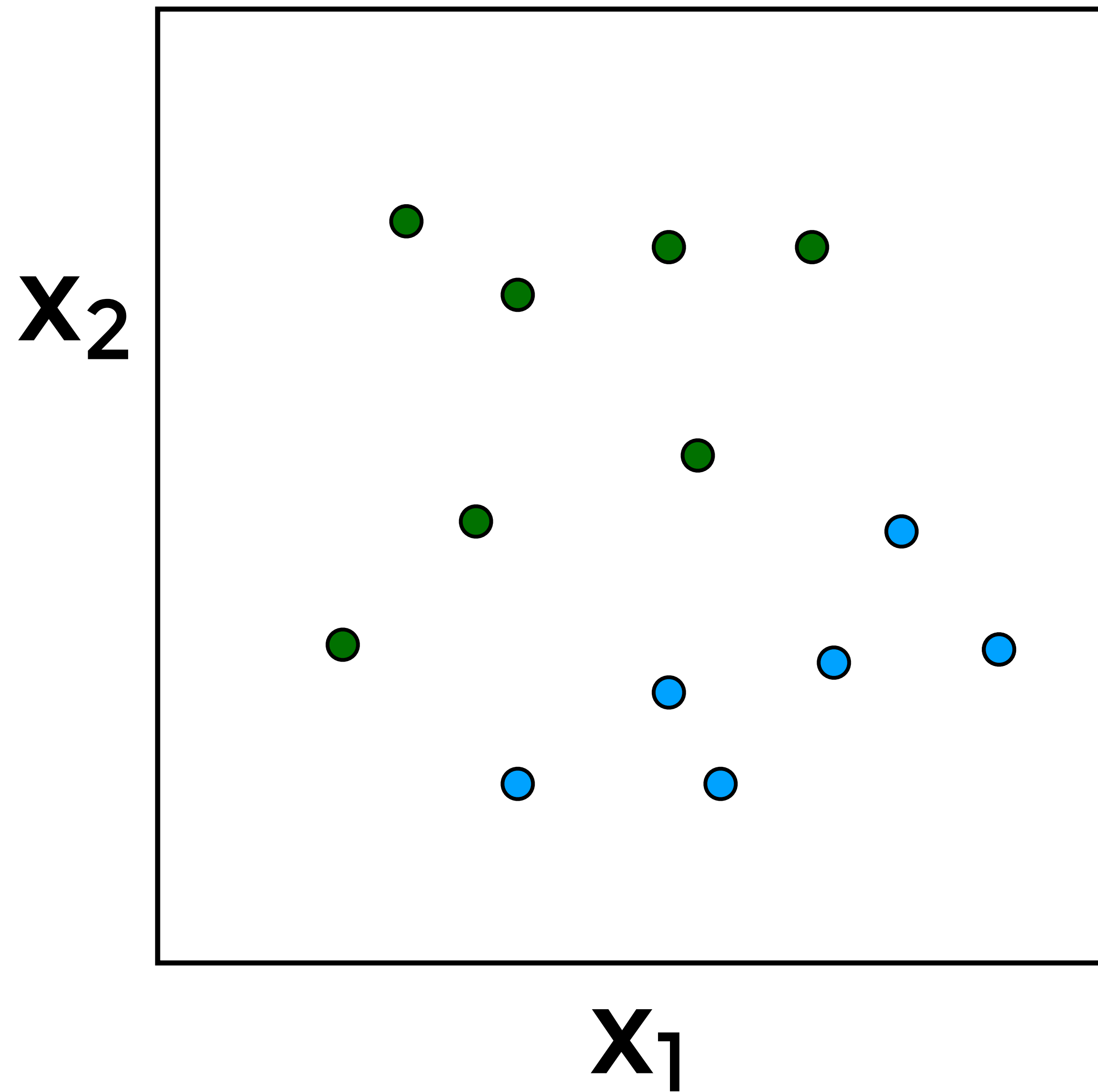
Ensemble  
Test

	Nb.bed.	Area	Neigh.	.	.		Sell-ability
$x_0$	1	0	0	0	0	$y_0$	?
$x_1$	2	50	1	.3	.8	$y_1$	?
$x_2$	1	100	1	.5	1.4	$y_2$	?
$x_3$	4	170	0	.7	.4	$y_3$	?
$x_4$	1	120	3	.9	.5	$y_4$	?

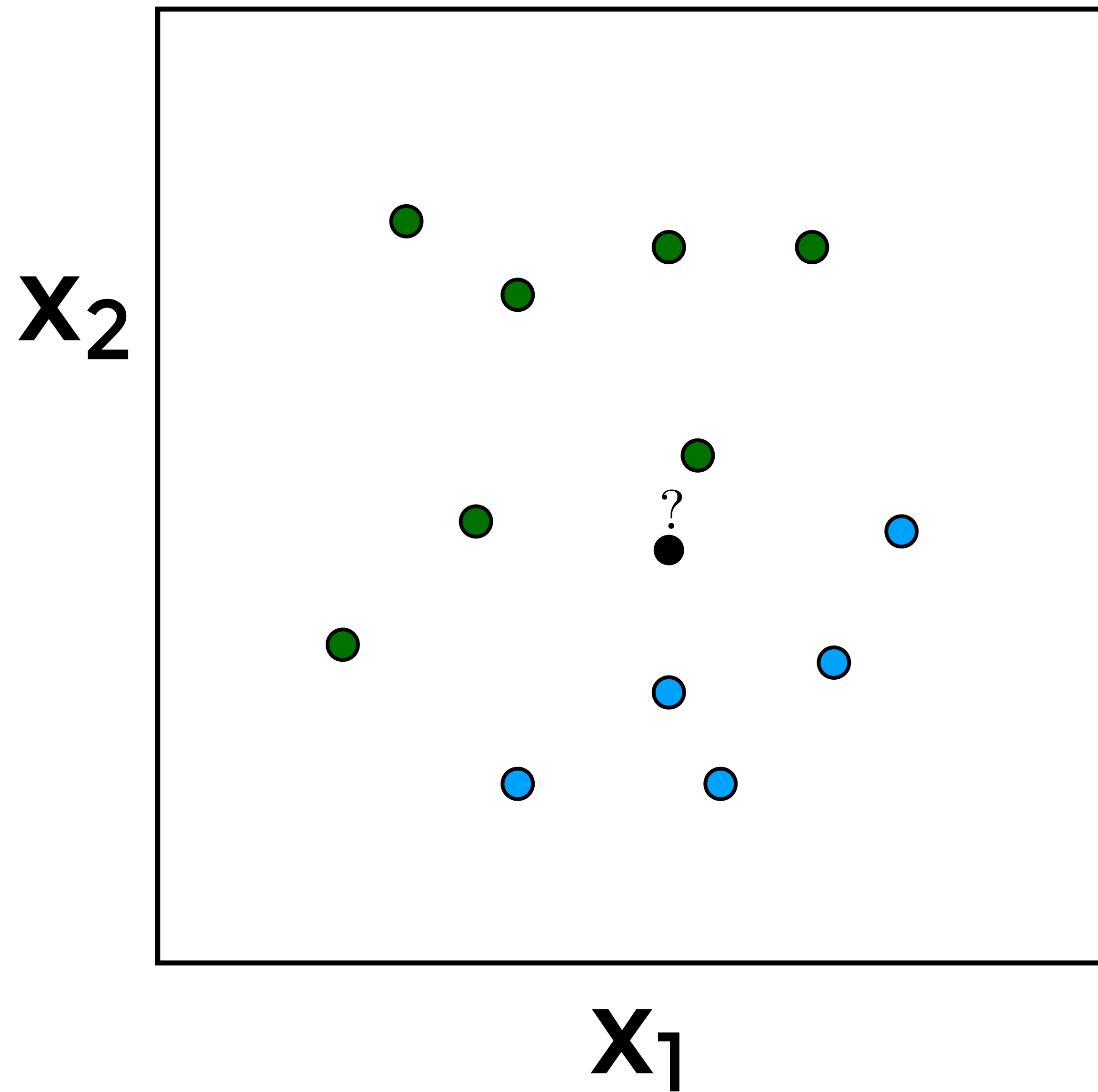
$X^{\text{new}}$   $Y^{\text{new}}$

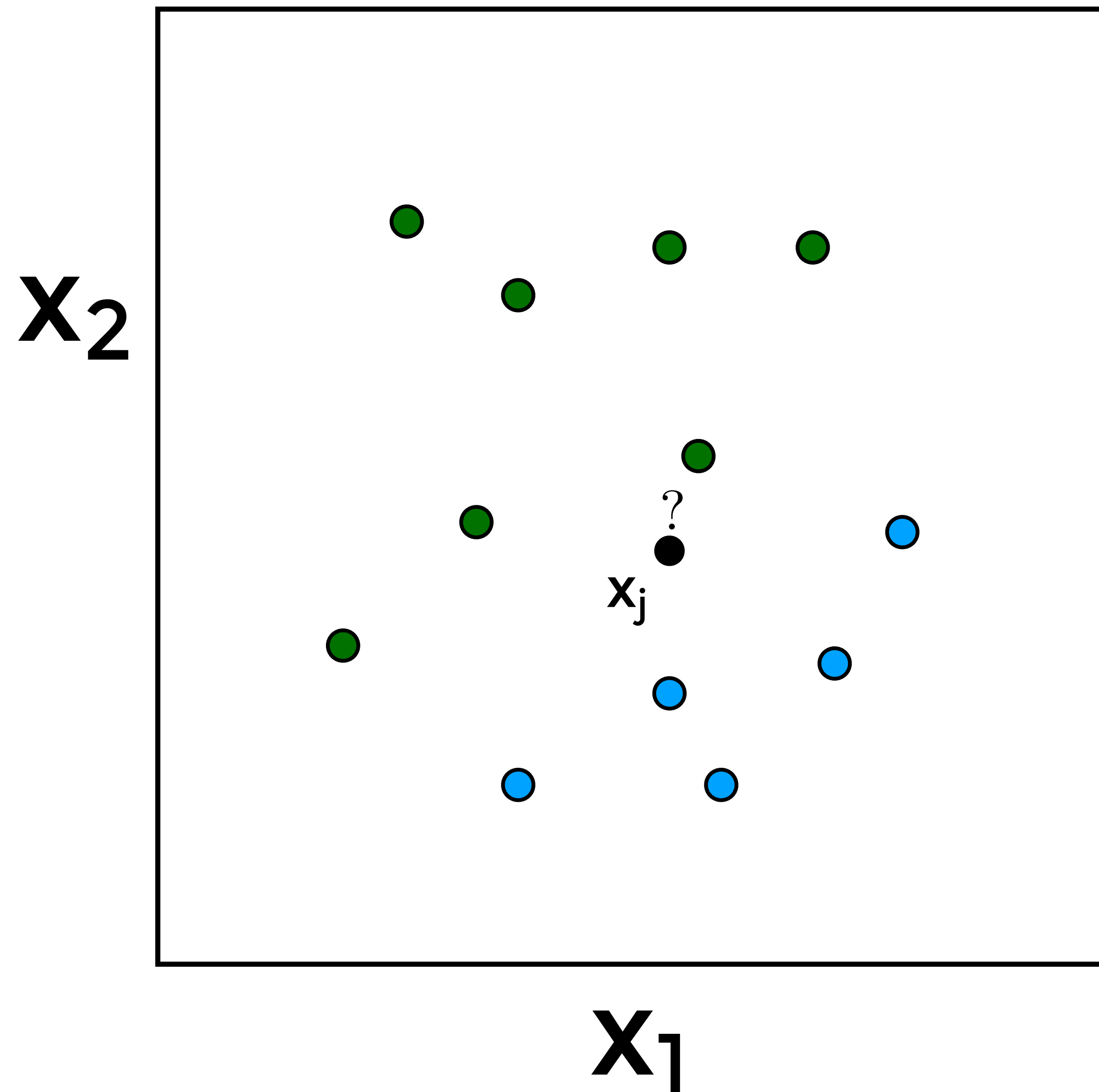








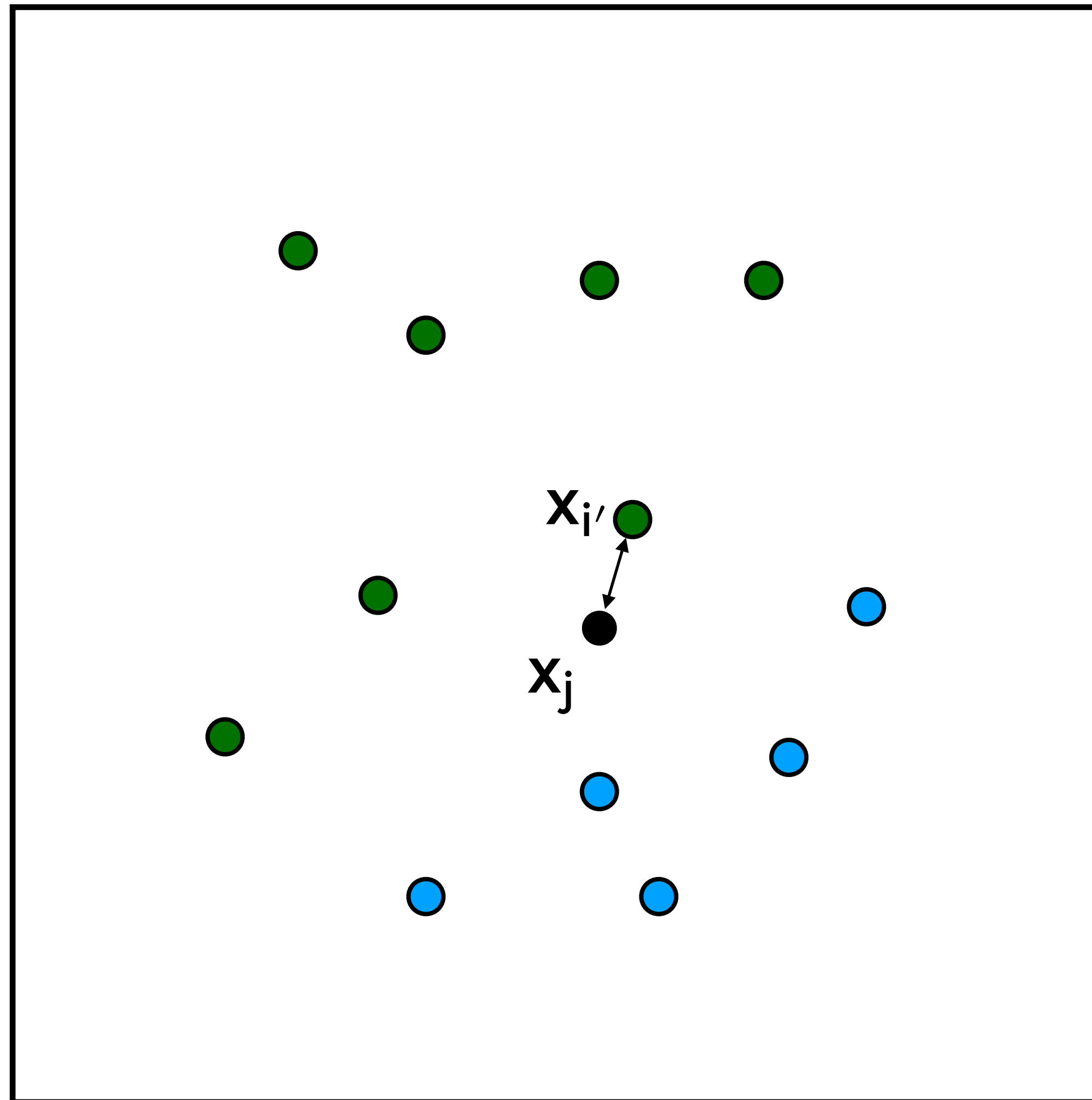




$$i' = \arg \min_i \text{dist}(\mathbf{x}_i, \mathbf{x}_j)$$

$$y_j = y_{i'}$$

$X_2$



$X_1$

- **1-NN**

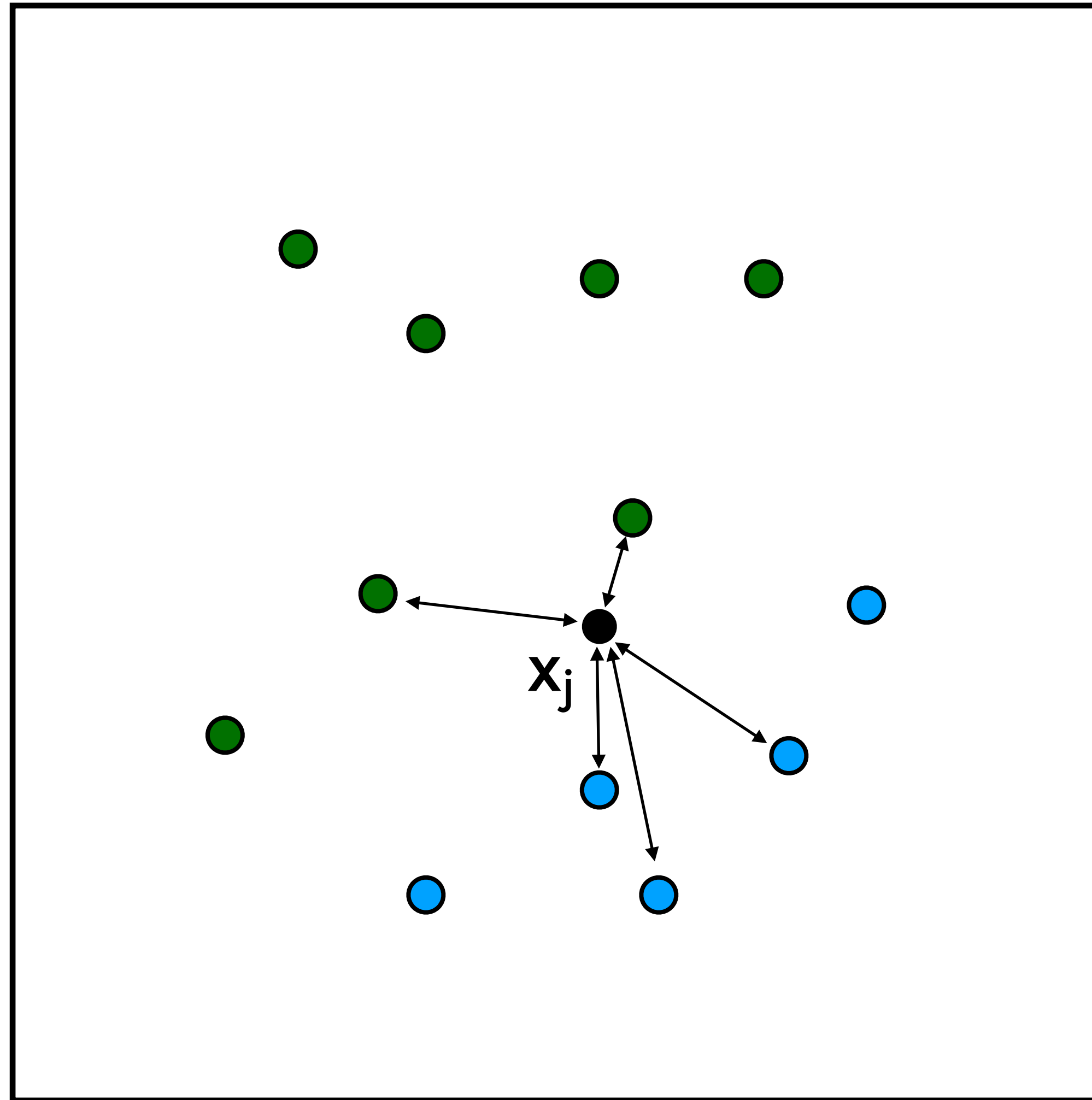
Donnée classifiée en fonction de son voisin le plus proche

$k = 5$  (assumption)

$i = \arg \text{sort}_i \text{dist}(\mathbf{x}_i, \mathbf{x}_j)$

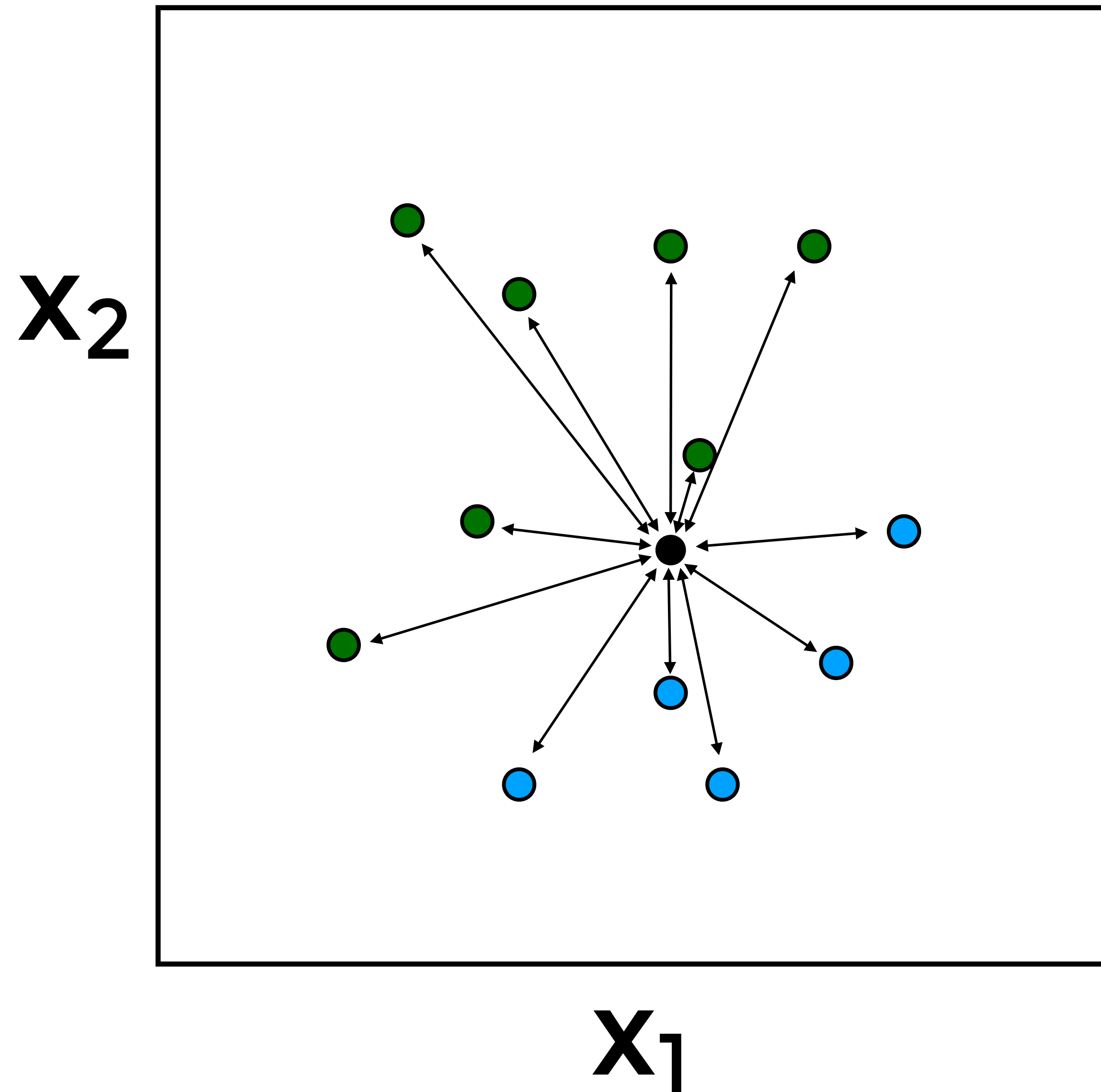
$y_j = \text{majority}(i_{:5})$

$X_2$



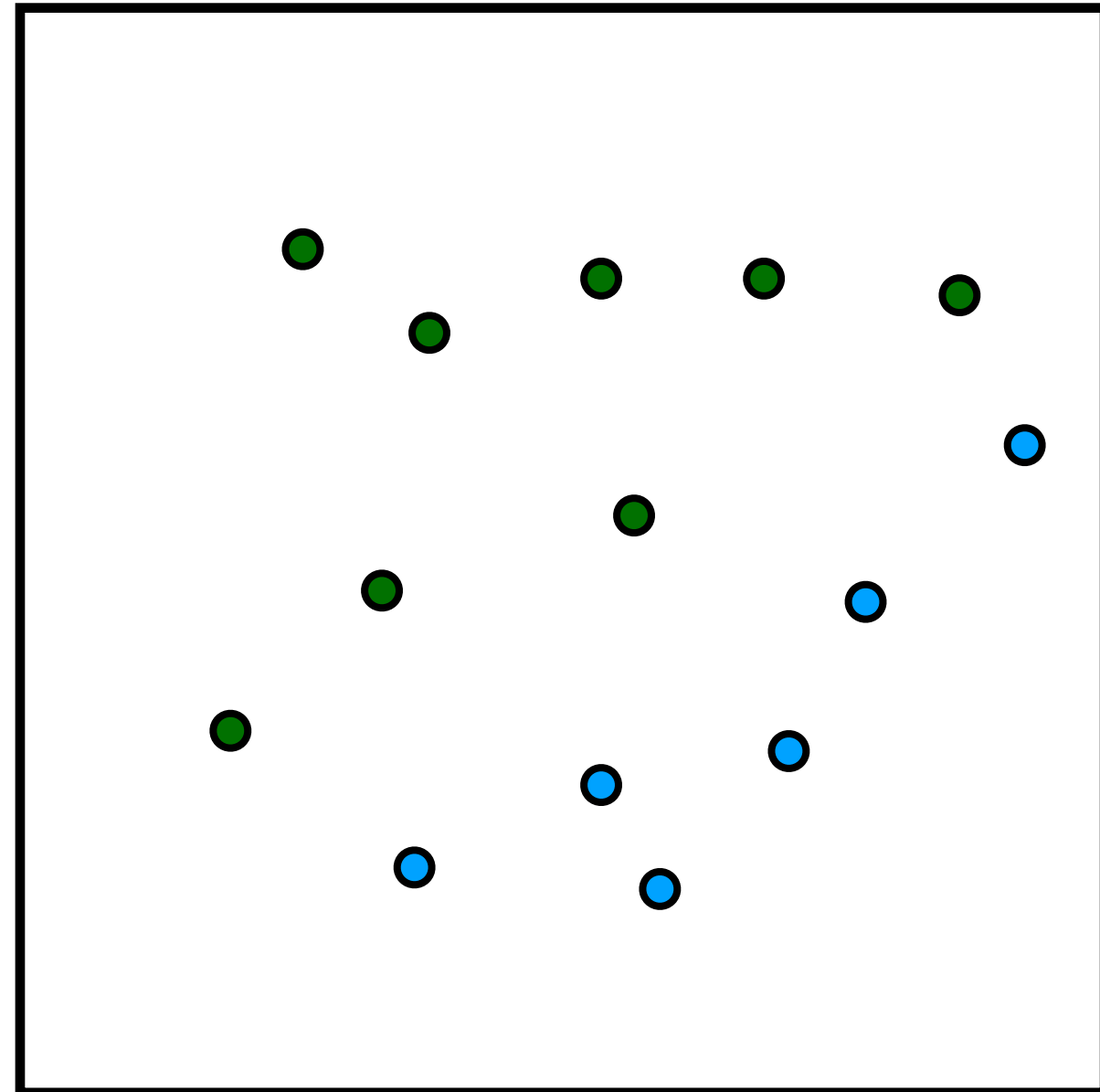
$X_1$

- **K-NN**  
Donnée classifiée en fonction de ses K voisins les plus proches

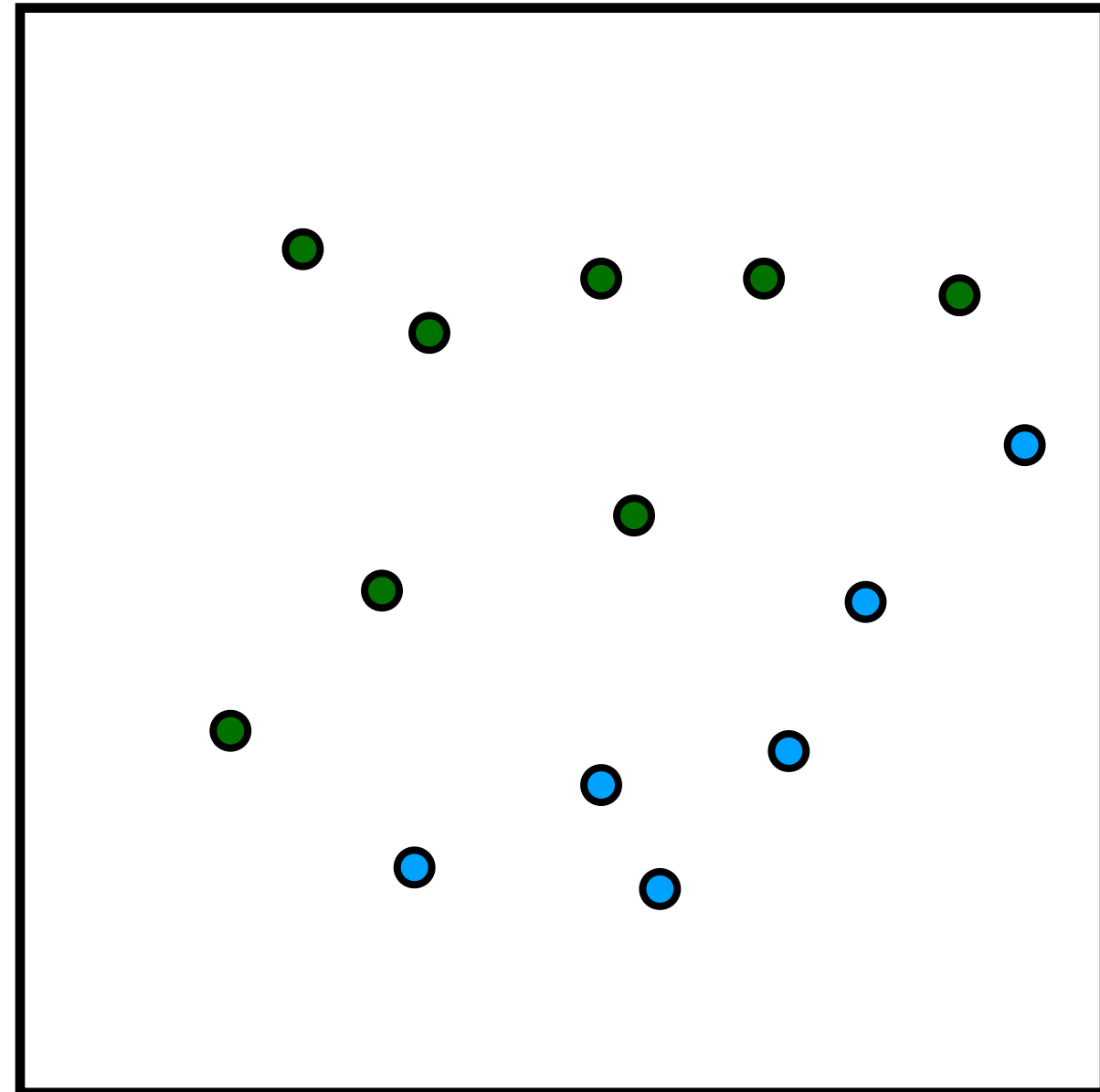


- **weighted-NN**  
Donnée classifiée en fonction de tous ses voisins. La contribution de chacun est pondérée par sa distance.

# Classification linéaire

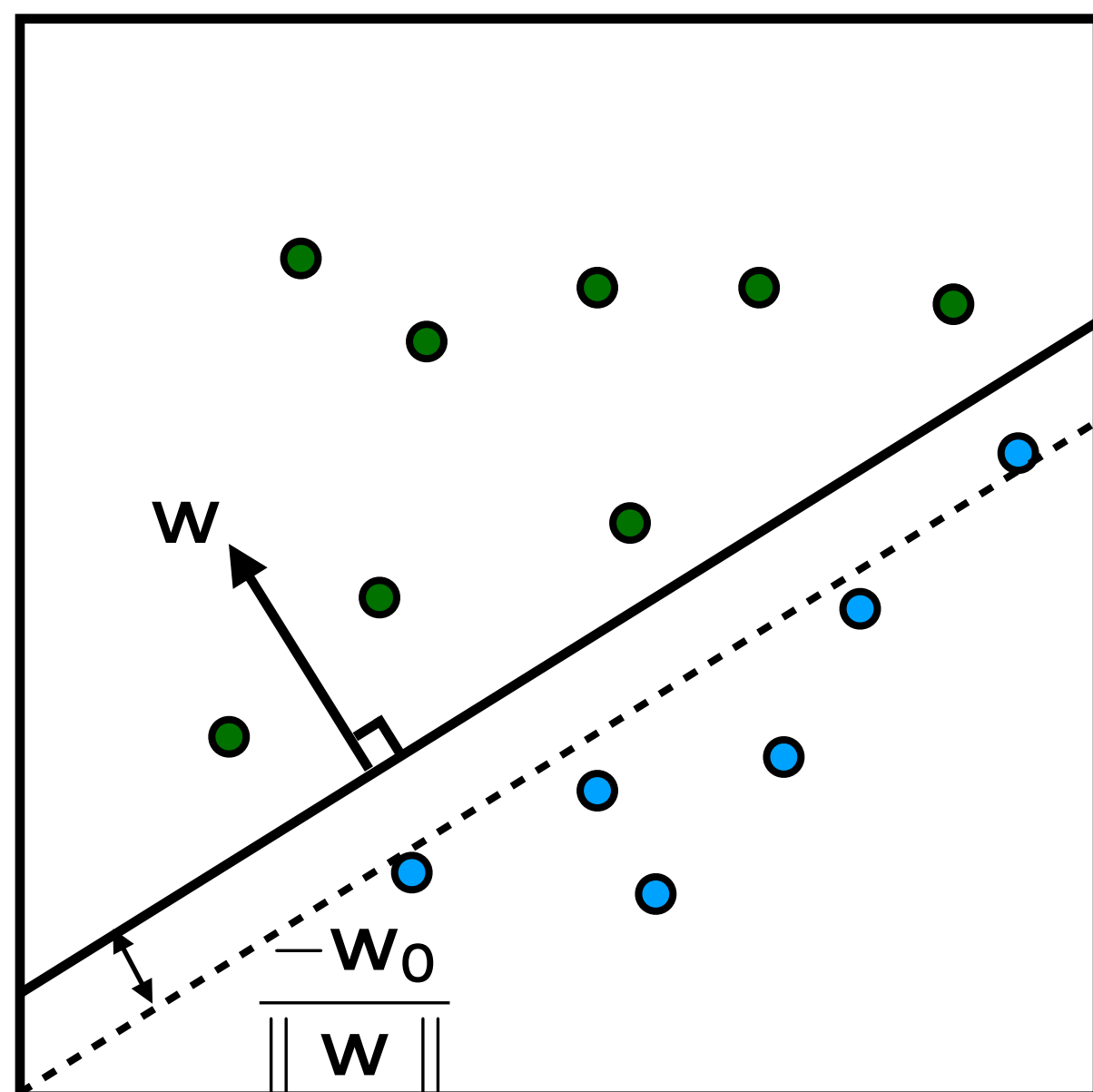


# Classification linéaire



$$y(\mathbf{x}) = \mathbf{w}^\top \mathbf{x} + w_0$$

# Classification linéaire



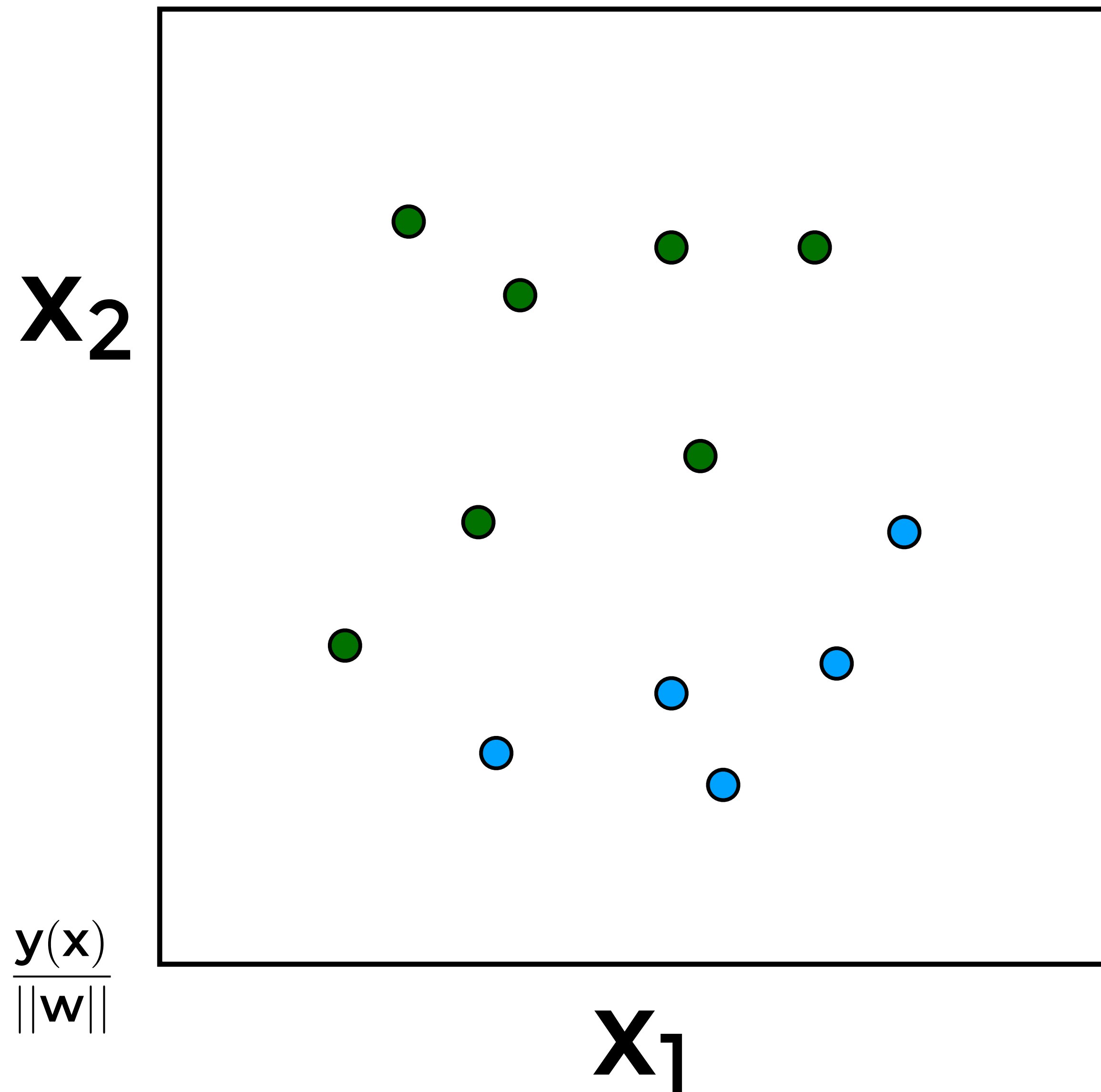
$$y(\mathbf{x}) = \mathbf{w}^\top \mathbf{x} + w_0$$

$$(\mathbf{w}^\top \mathbf{x} + w_0) > 0 \implies \bullet$$

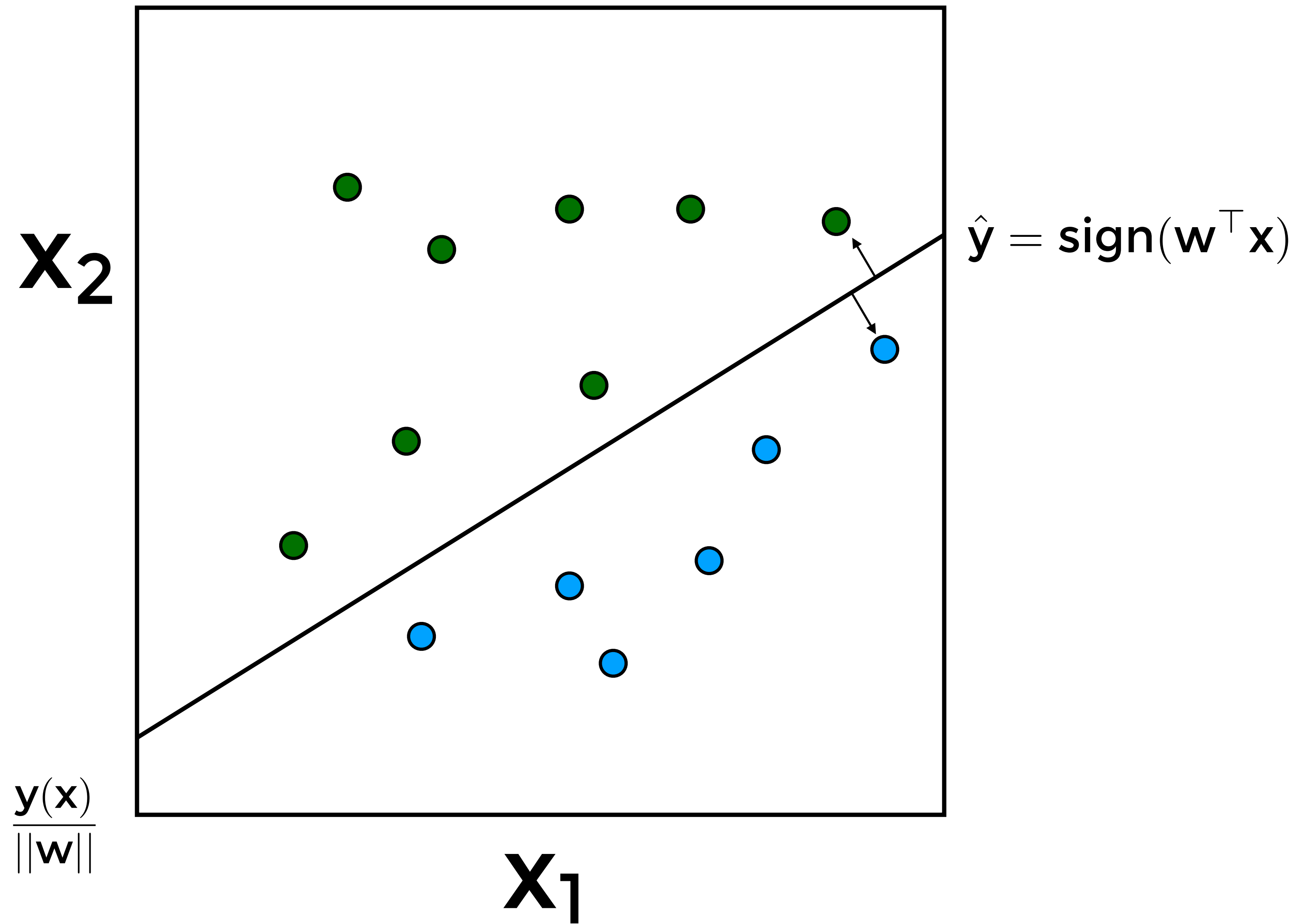
$$(\mathbf{w}^\top \mathbf{x} + w_0) < 0 \implies \bullet$$

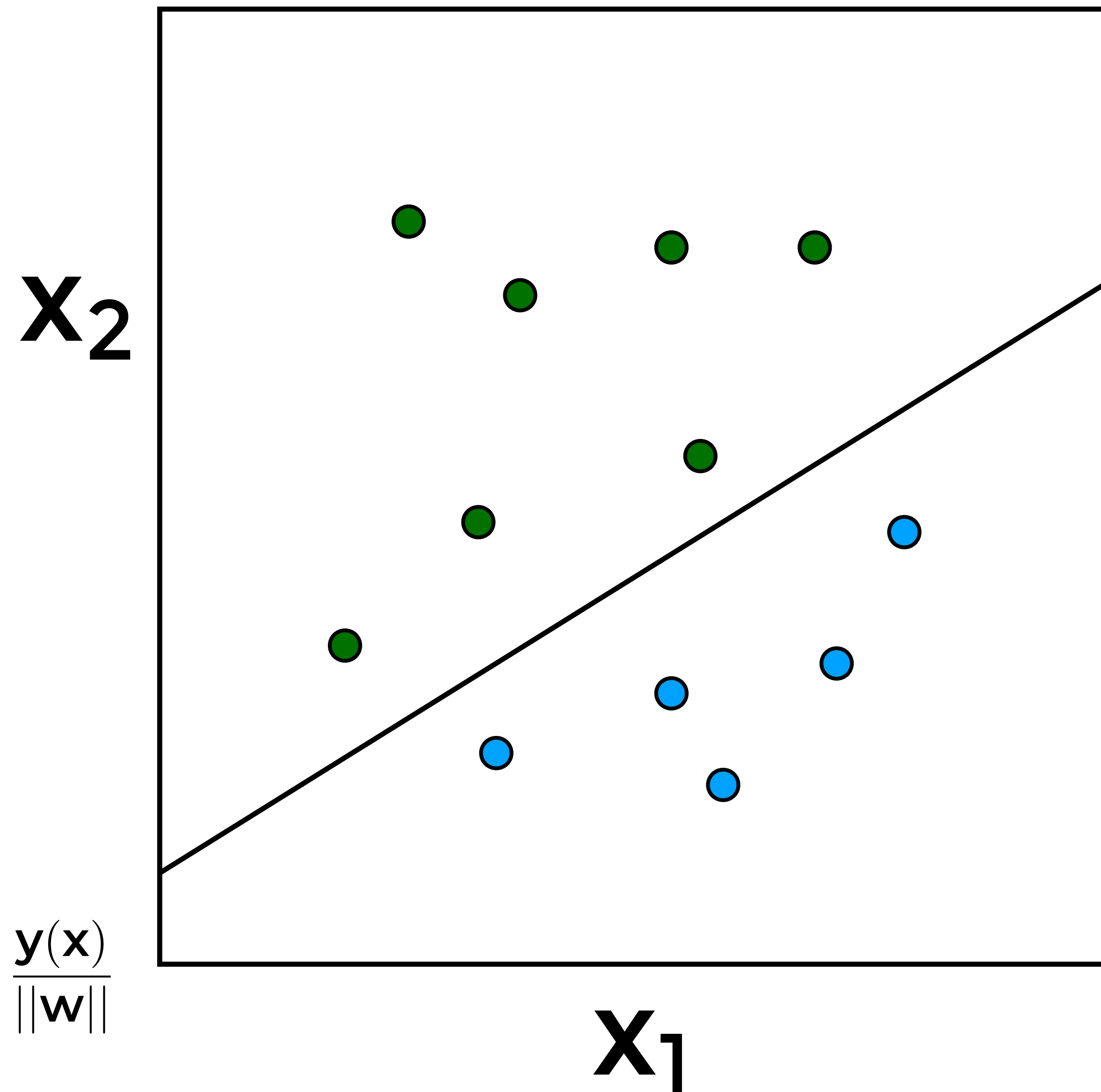
Décision

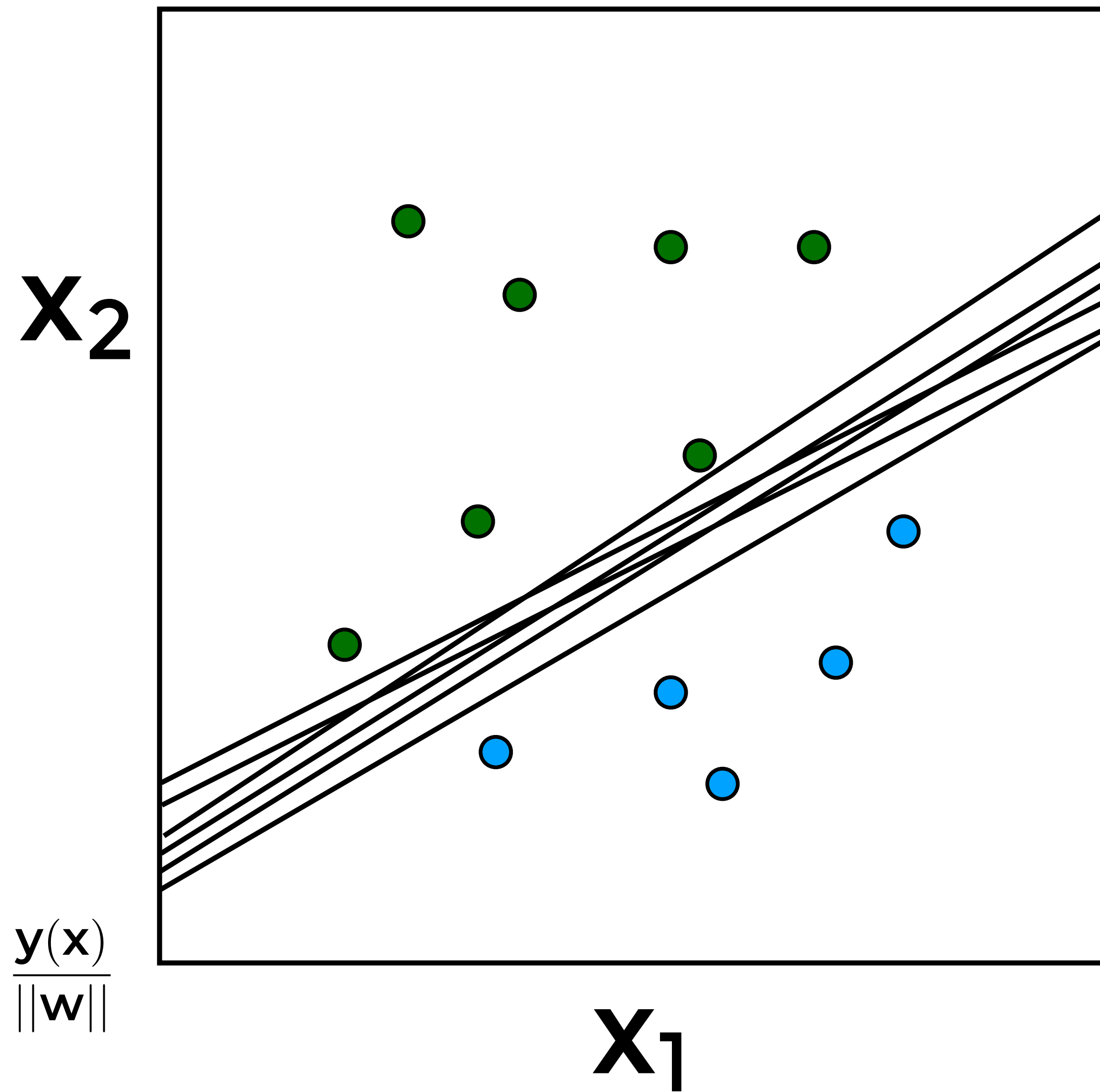


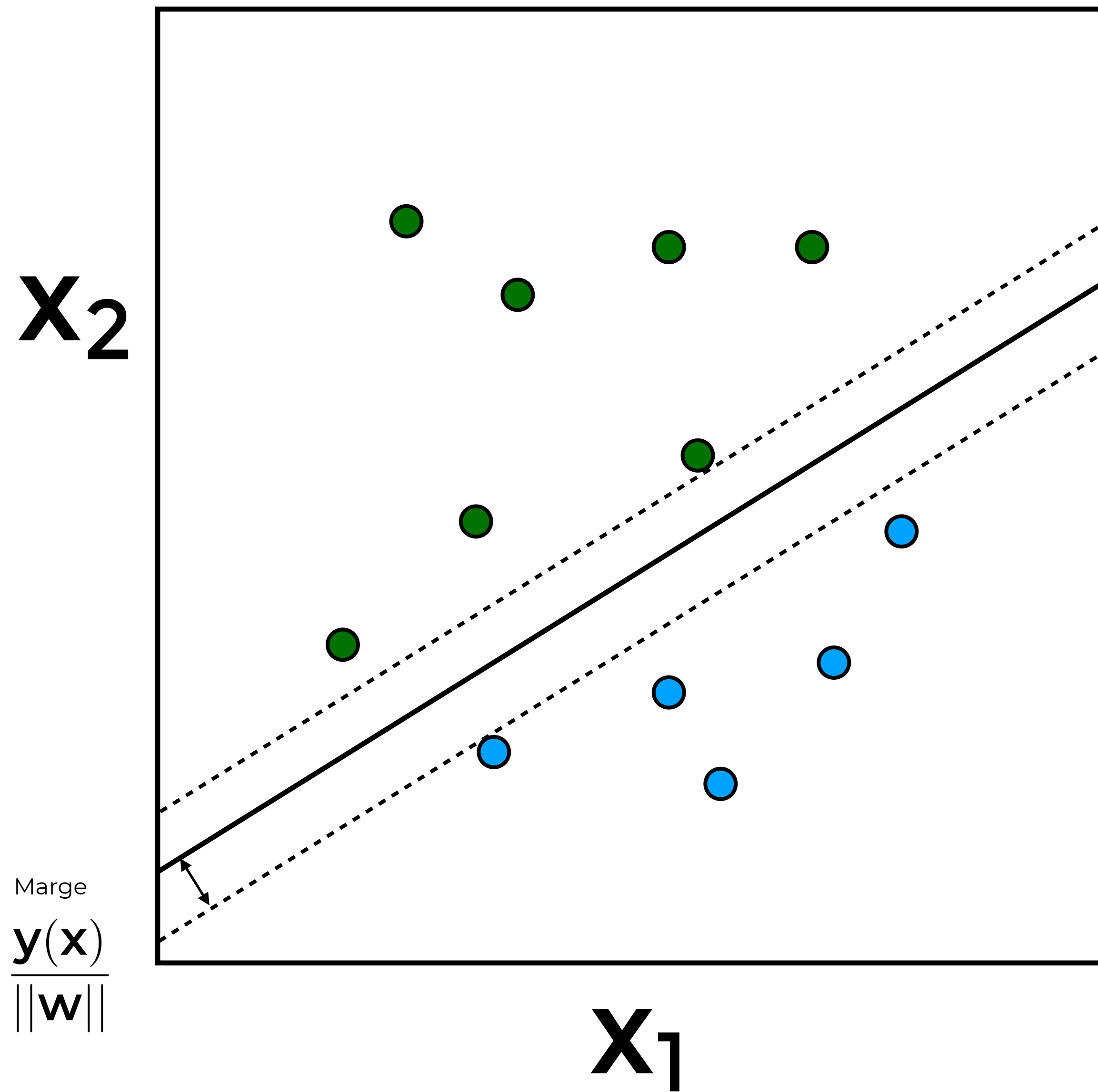


$$\frac{y(\mathbf{x})}{\|\mathbf{w}\|}$$

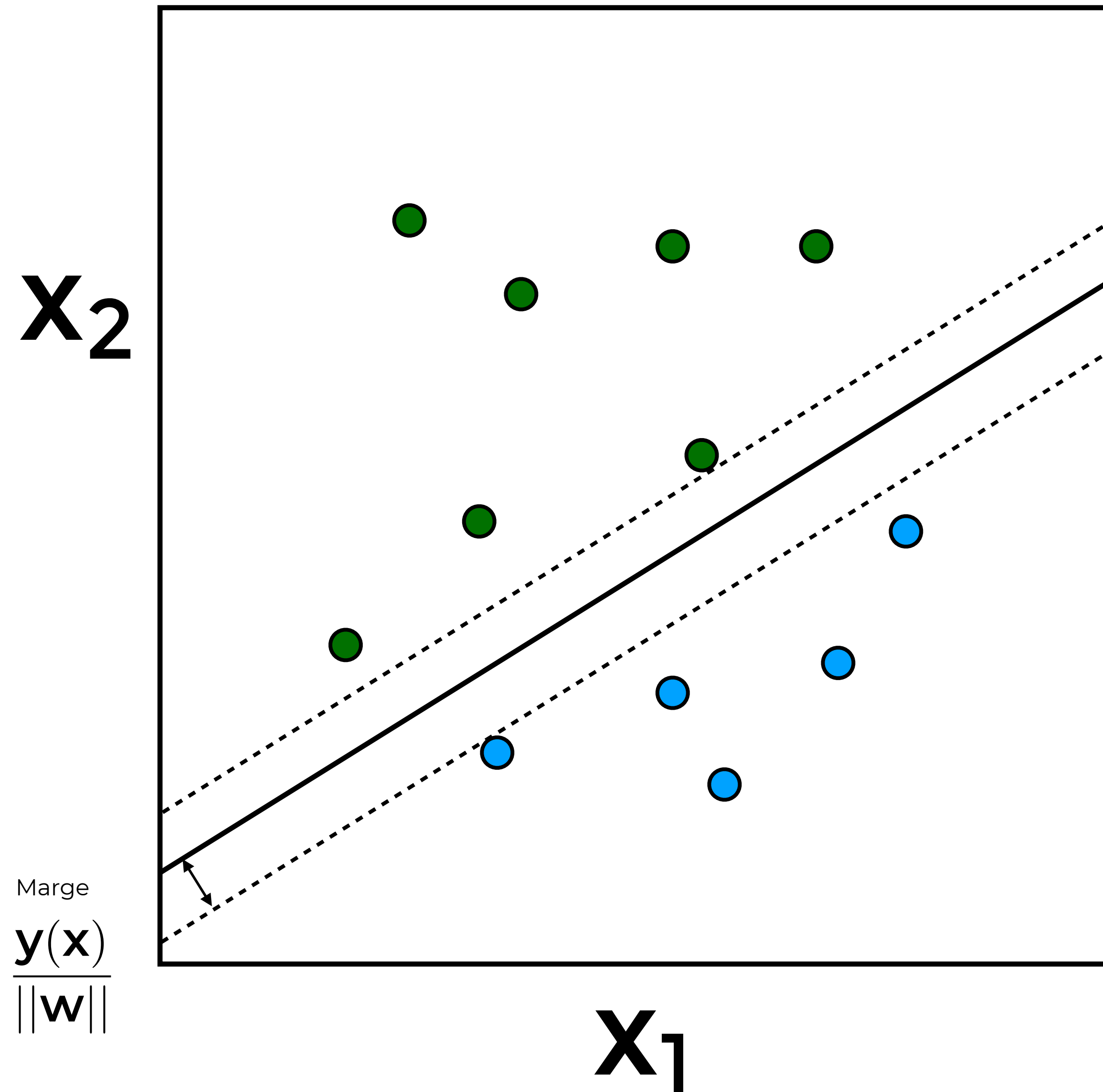








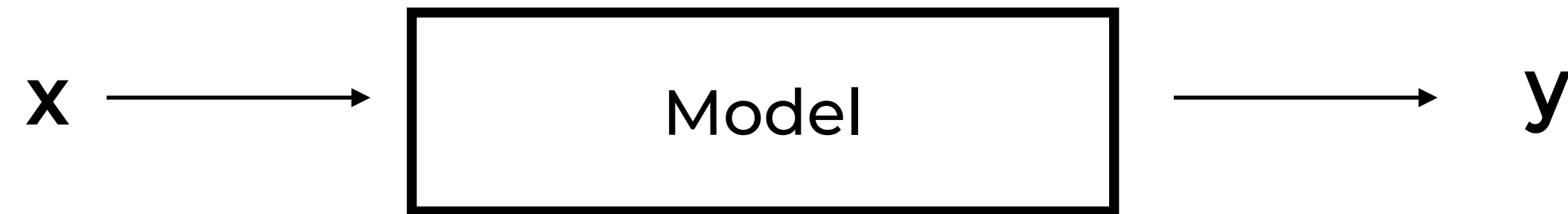
# SVM: L'objectif est de maximiser la marge



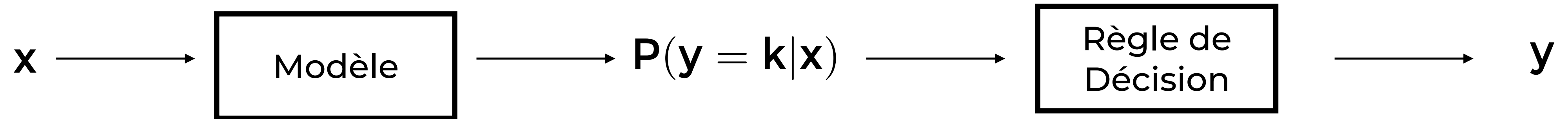
# **Modèles probabilistes pour la classification**

# Les modèles probabilistes séparent la décision de l'inférence

Modèle  
non probabiliste



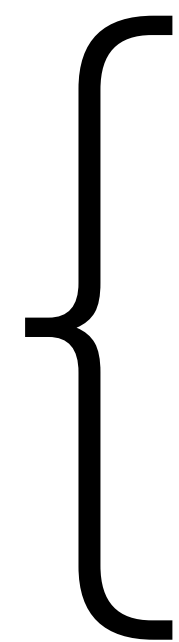
Modèle  
Probabiliste





# Modèle Probabiliste

Souvent  
Ensemble



1. On pose un modèle:  $P( X, Y )$ 
  - Ce modèle explique comment les données sont générées
2. Paramétrise les distributions:  $P( X, Y | \text{Parameters} )$
3. L'objectif (en général MLE)
4. On apprend les paramètres du modèle:
  - P. ex., Naive Bayes: on apprend les probabilités des classes conditionnelles  $P( X | Y )$  et la prière  $P( Y )$
5. On utilise le modèle (p. ex., pour faire des prédictions)