

# **Machine Learning for Large-Scale Data Analysis and Decision Making (MATH80629A) Winter 2023**

**Week #3 - Summary**



# Quiz 1

Login to your Gradescope account

# Announcement

- **Practical Lab session** on week #4 (January 27)
- **Team Registration**, due: **January 20, 2023** (Today)
- Next week: **Homework 1** will be released!  
It is due **February 17, 2023 (3 weeks)**

# Today

- **Second Quiz** on Gradescope!
- Summary of Supervised Learning Algorithms
- Q&A
- Hands-on session

# Models for supervised learning

- (Mostly) linear models
- Focus on **classification**

## 1. Non-Probabilistic Models

- Nearest Neighbor, Support Vector Machines (SVMs)

## 2. Probabilistic Models

- Naive Bayes

# Supervised learning

Train Data

	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
$X$						$Y$	

Task

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

Test Data

	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	$y_4$	?
$X^{\text{new}}$						$Y^{\text{new}}$	

# Supervised learning

Train Data

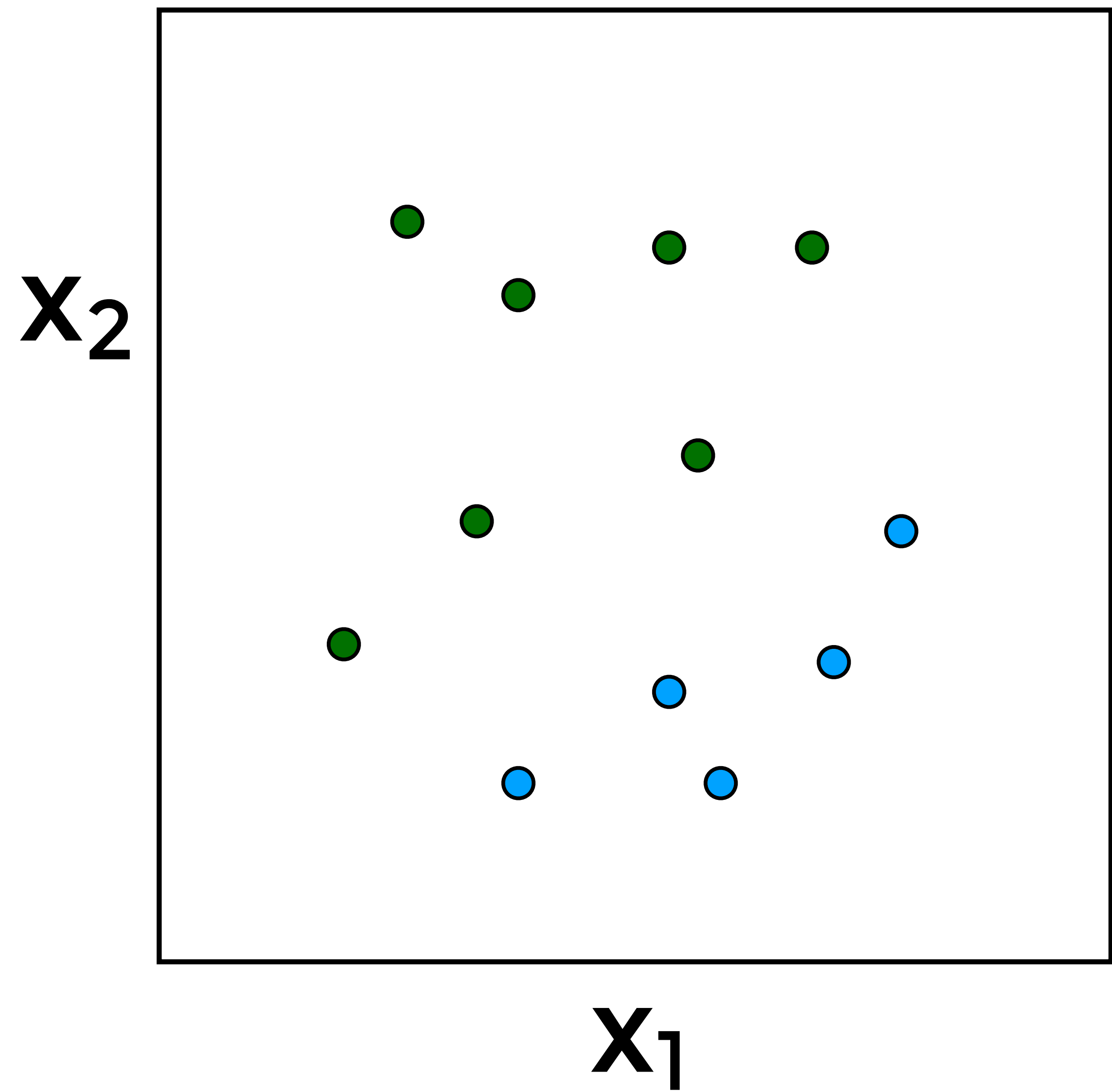
	Nb.bed.	Area	Neigh.	.	.		Sell-ability
$x_0$	1	0	0	0	0	$y_0$	1
$x_1$	1	100	1	.2	.5	$y_1$	2
$x_2$	3	200	0	.1	.2	$y_2$	0
$x_3$	1	150	1	.4	.1	$y_3$	2
$x_4$	2	210	2	.5	1.1	$y_4$	1
$X$						$Y$	

Task

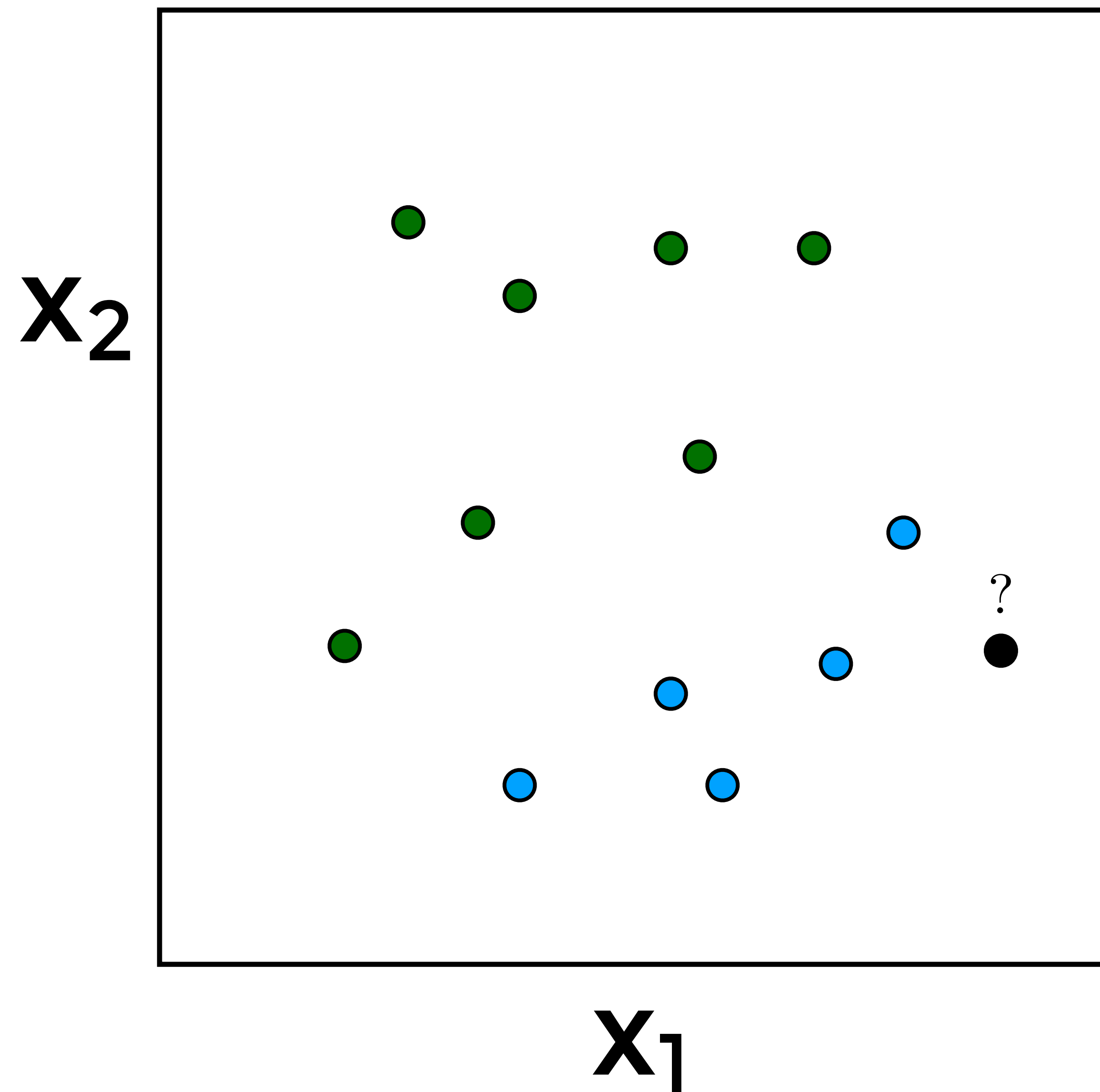
Models  $\mathbf{f} : \mathbb{R}^n \rightarrow \{0, 1, 2\}$

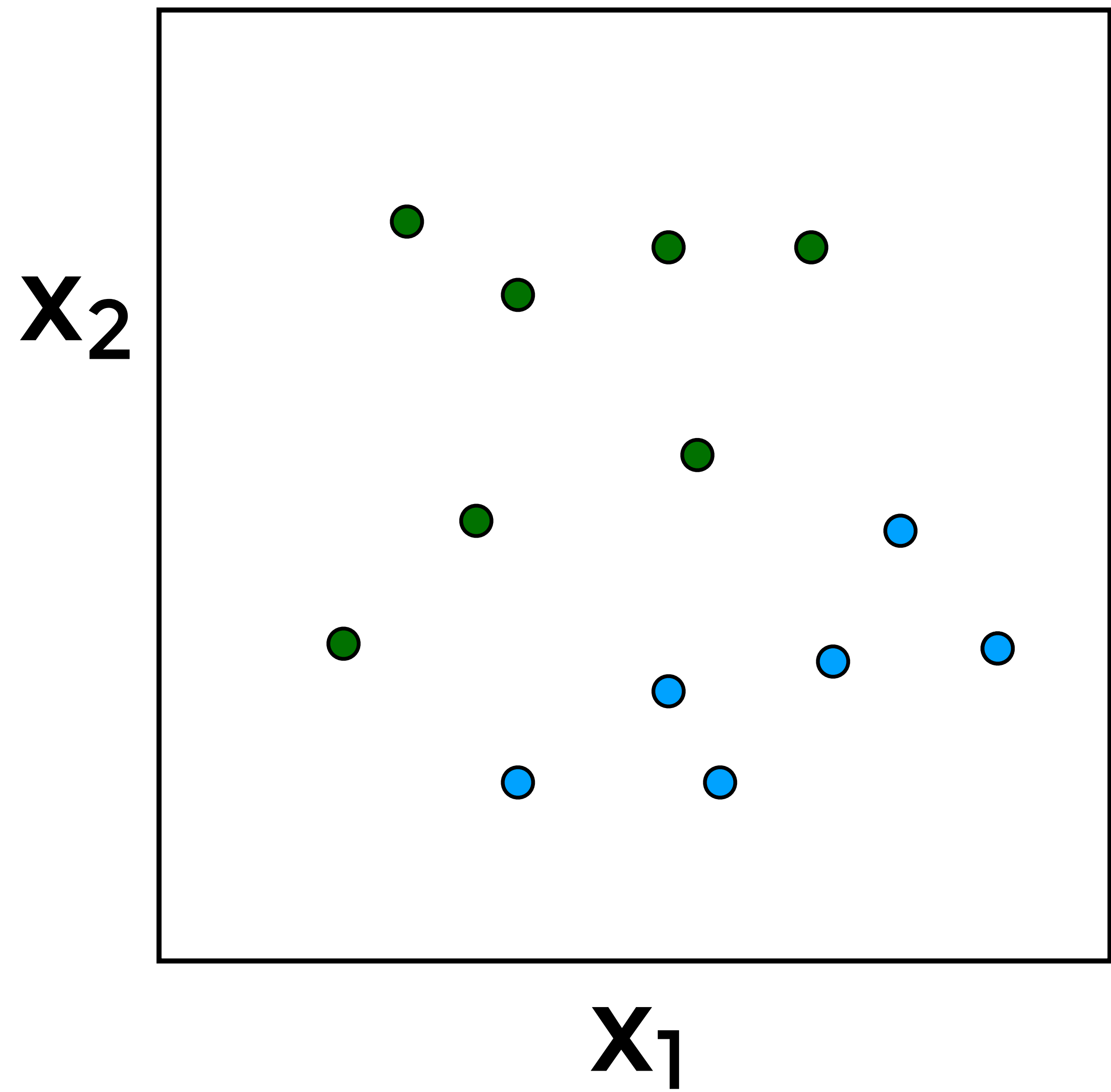
Test Data

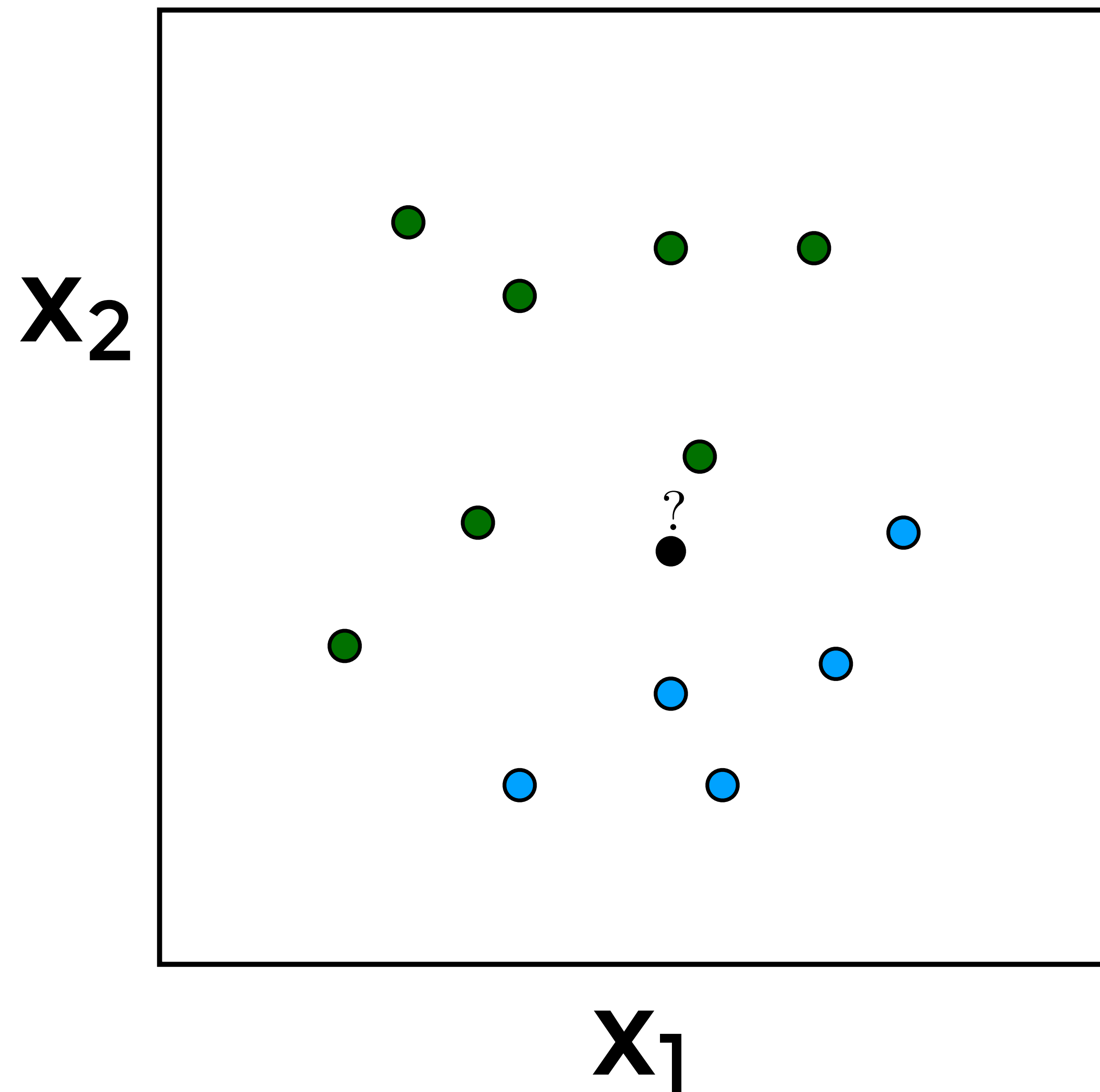
	Nb.bed.	Area	Neigh.	.	.		Sell-ability
$x_0$	1	0	0	0	0	$y_0$	?
$x_1$	2	50	1	.3	.8	$y_1$	?
$x_2$	1	100	1	.5	1.4	$y_2$	?
$x_3$	4	170	0	.7	.4	$y_3$	?
$x_4$	1	120	3	.9	.5	$y_4$	?
$X^{\text{new}}$						$Y^{\text{new}}$	

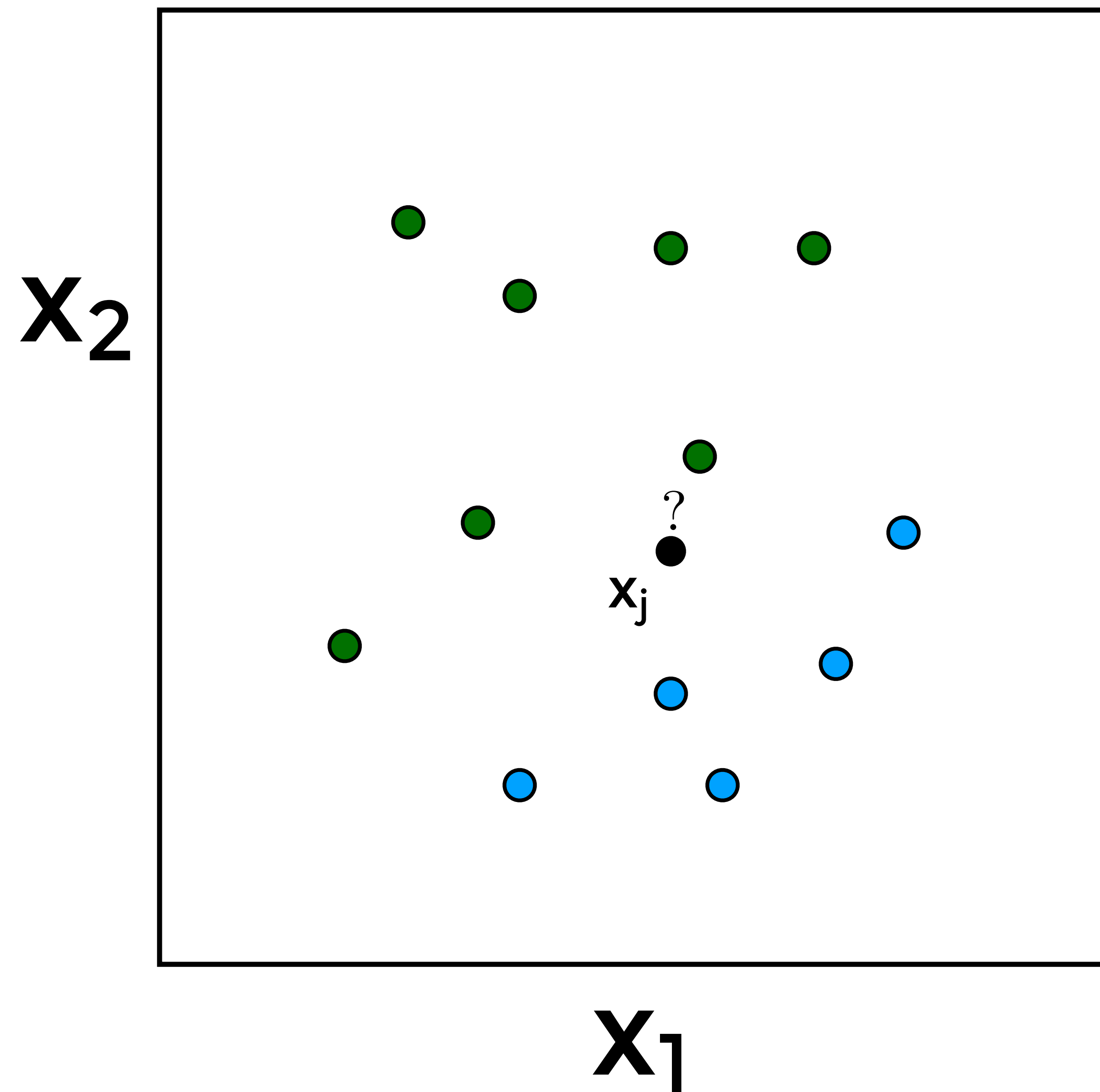








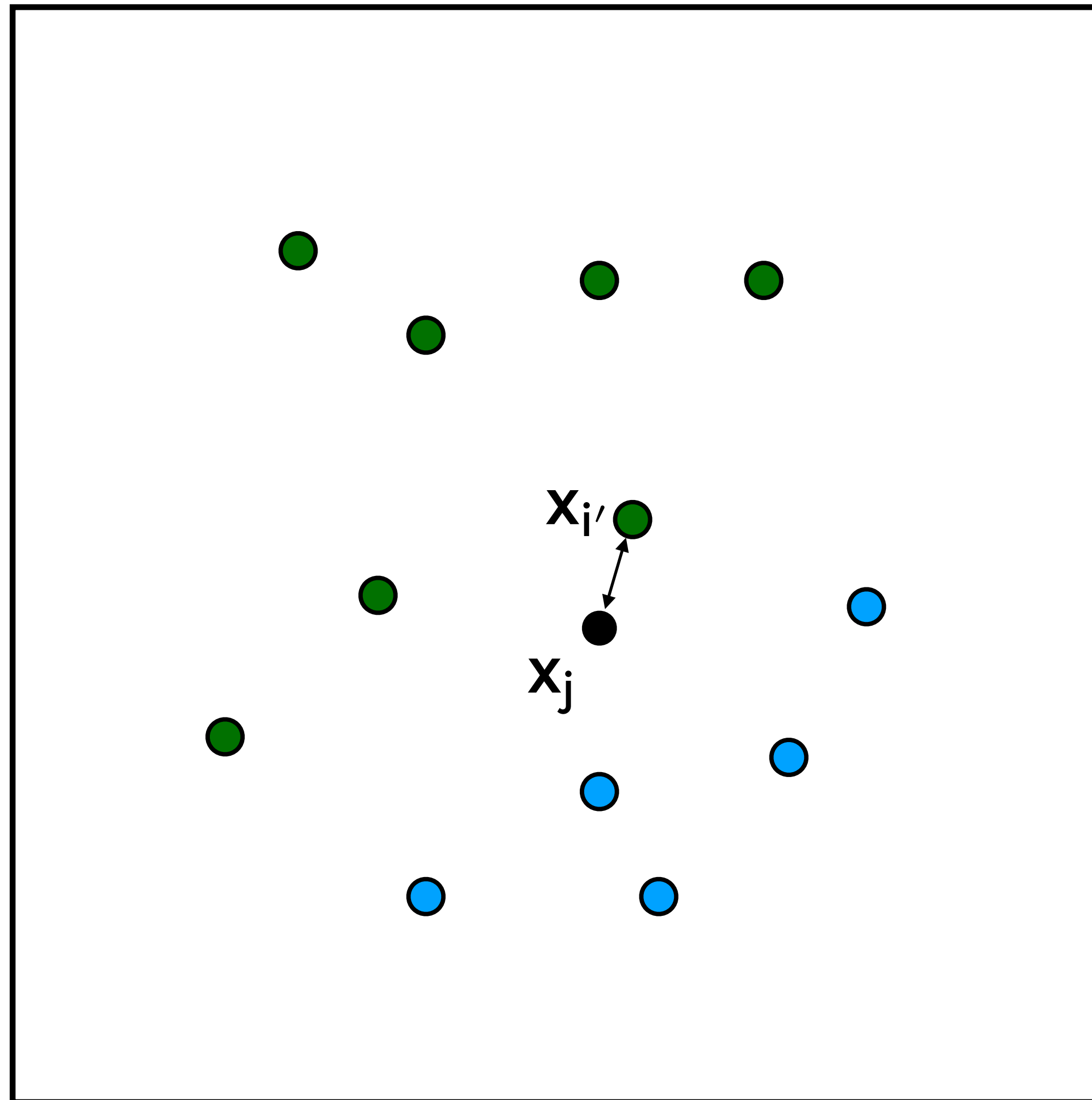




$$i' = \arg \min_i \text{dist}(\mathbf{x}_i, \mathbf{x}_j)$$

$$\mathbf{y}_j = \mathbf{y}_{i'}$$

$\mathbf{x}_2$



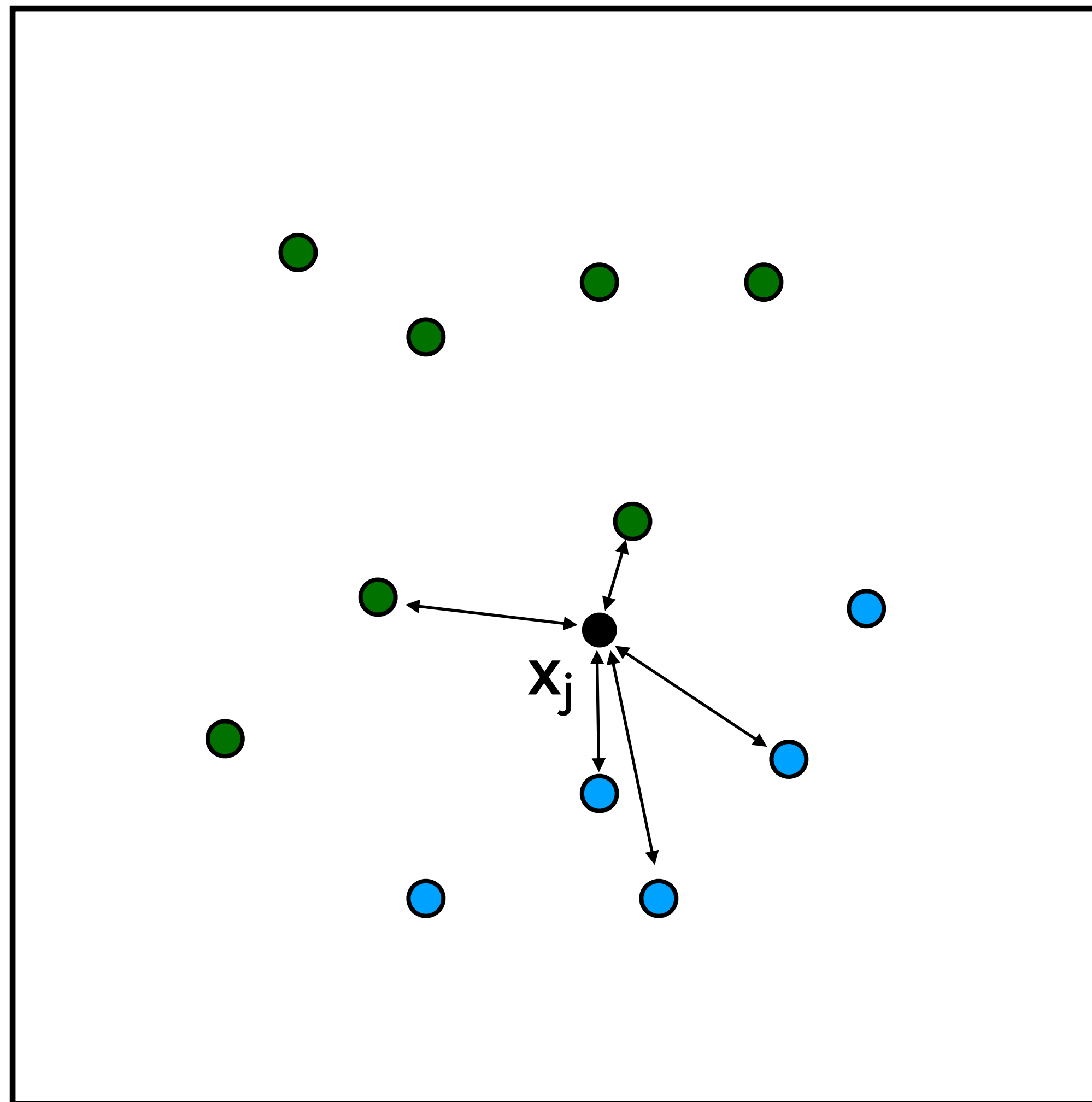
$x_1$

- **1-NN**

Instance classified  
according to its nearest  
neighbor

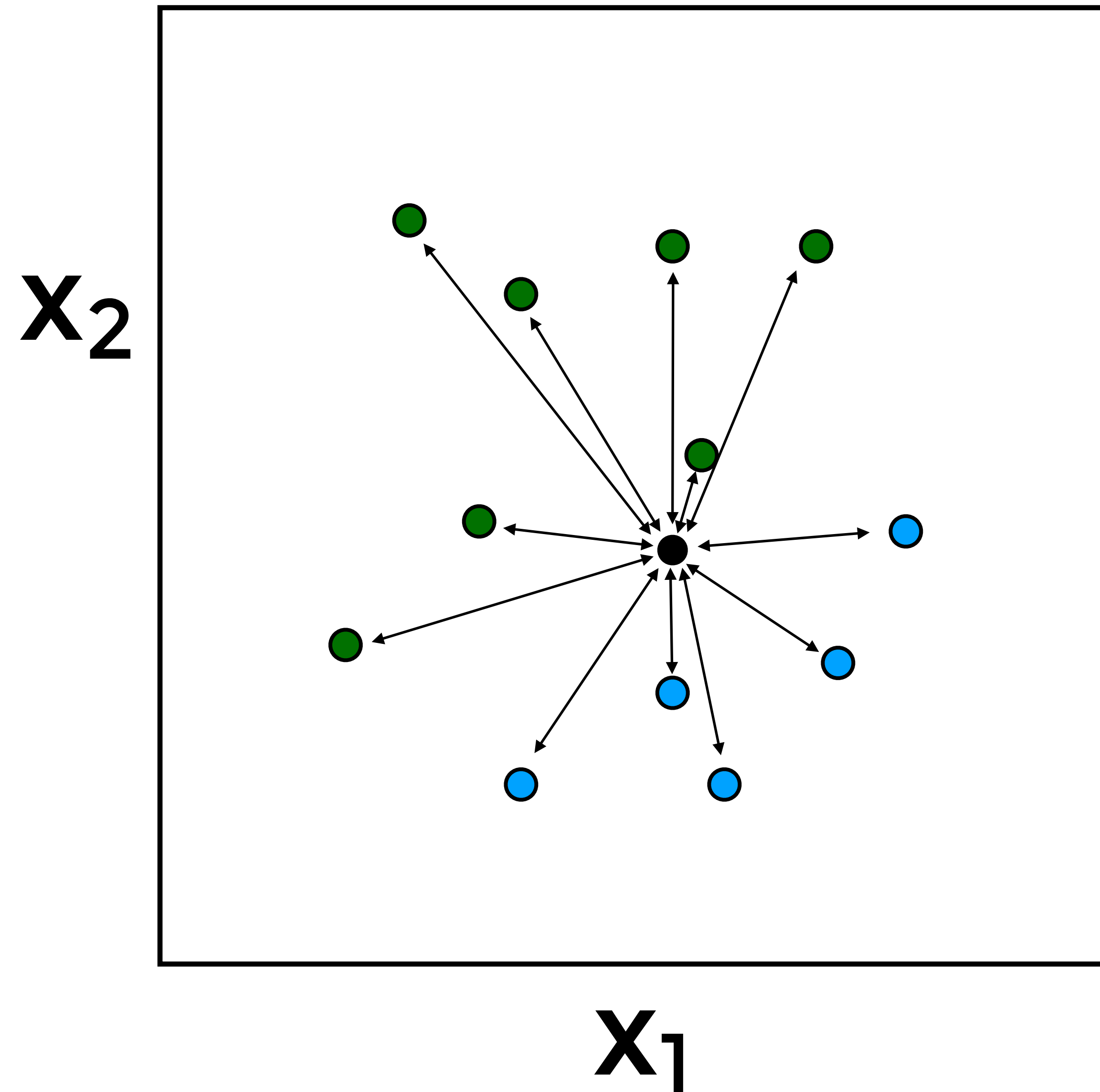
$k = 5$  (assumption)  
 $i = \arg \text{sort}_i \text{dist}(\mathbf{x}_i, \mathbf{x}_j)$   
 $y_j = \text{majority}(i_{1:5})$

$X_2$



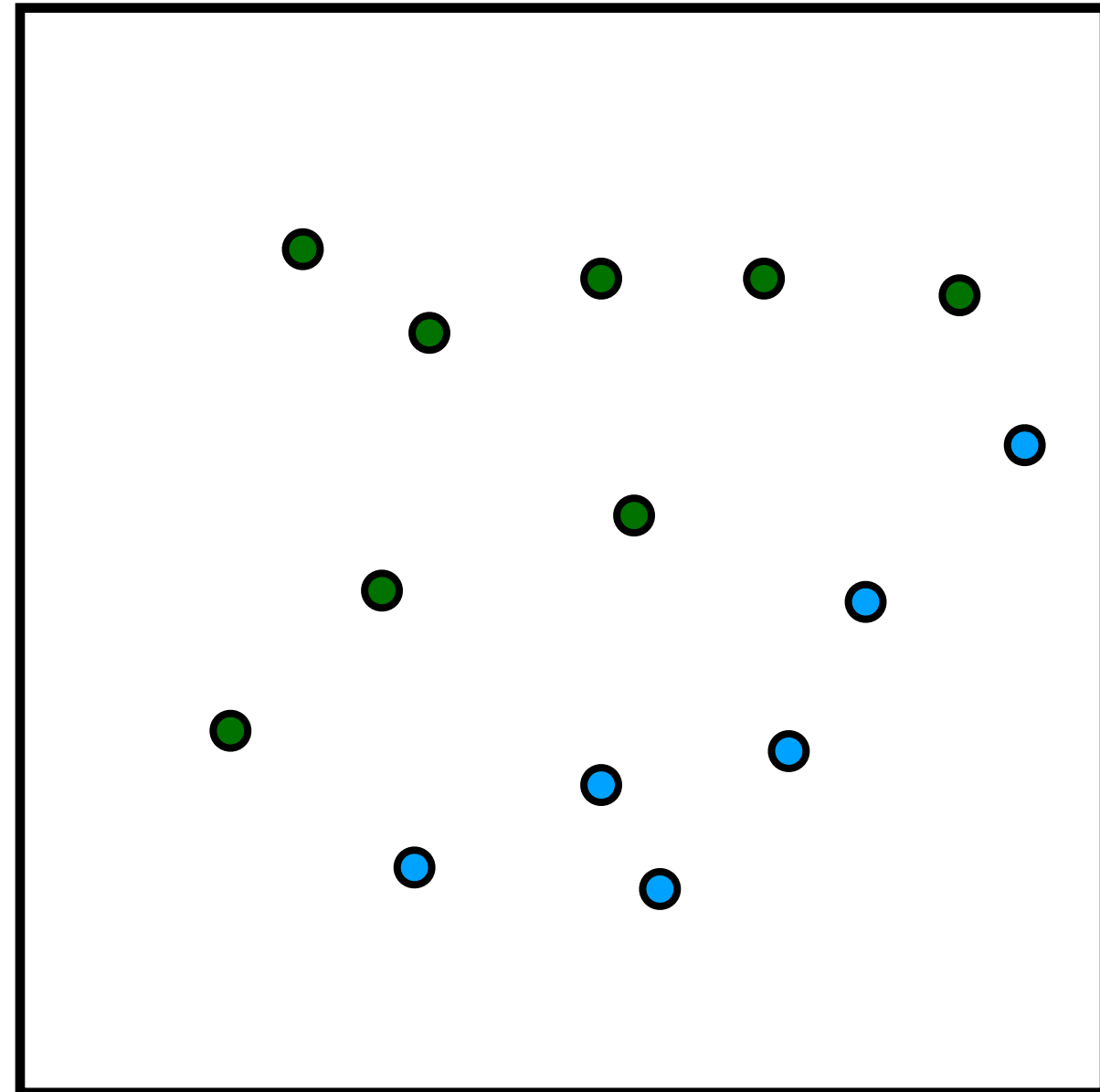
$X_1$

- **K-NN**  
 Instance classified  
 according to the majority of  
 its K nearest neighbors



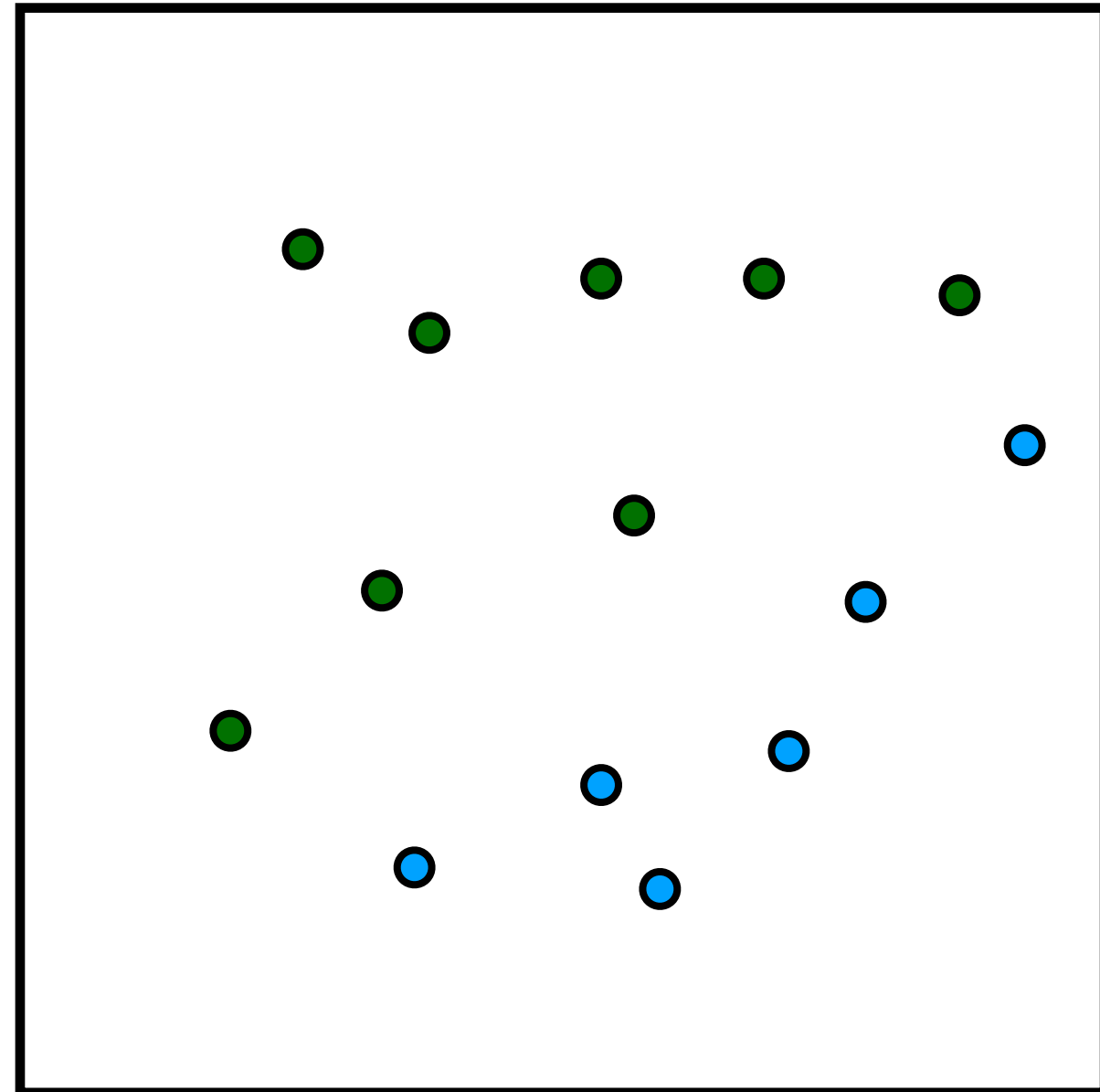
- **weighted-NN**  
Instance classified according to all neighbors. The contribution of each neighbor is weighted by its distance.

# Linear Classification



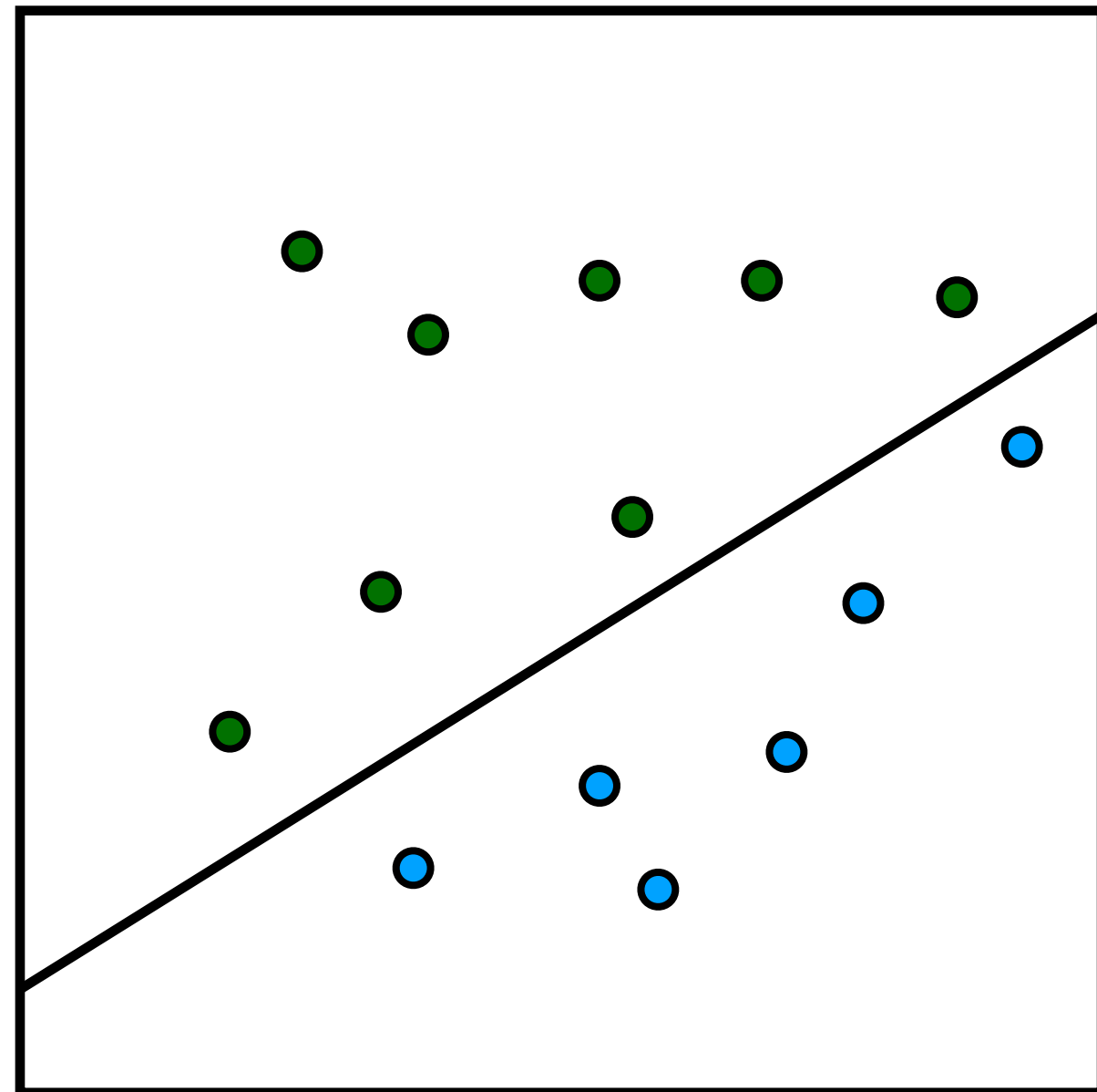


# Linear Classification



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

# Linear Classification



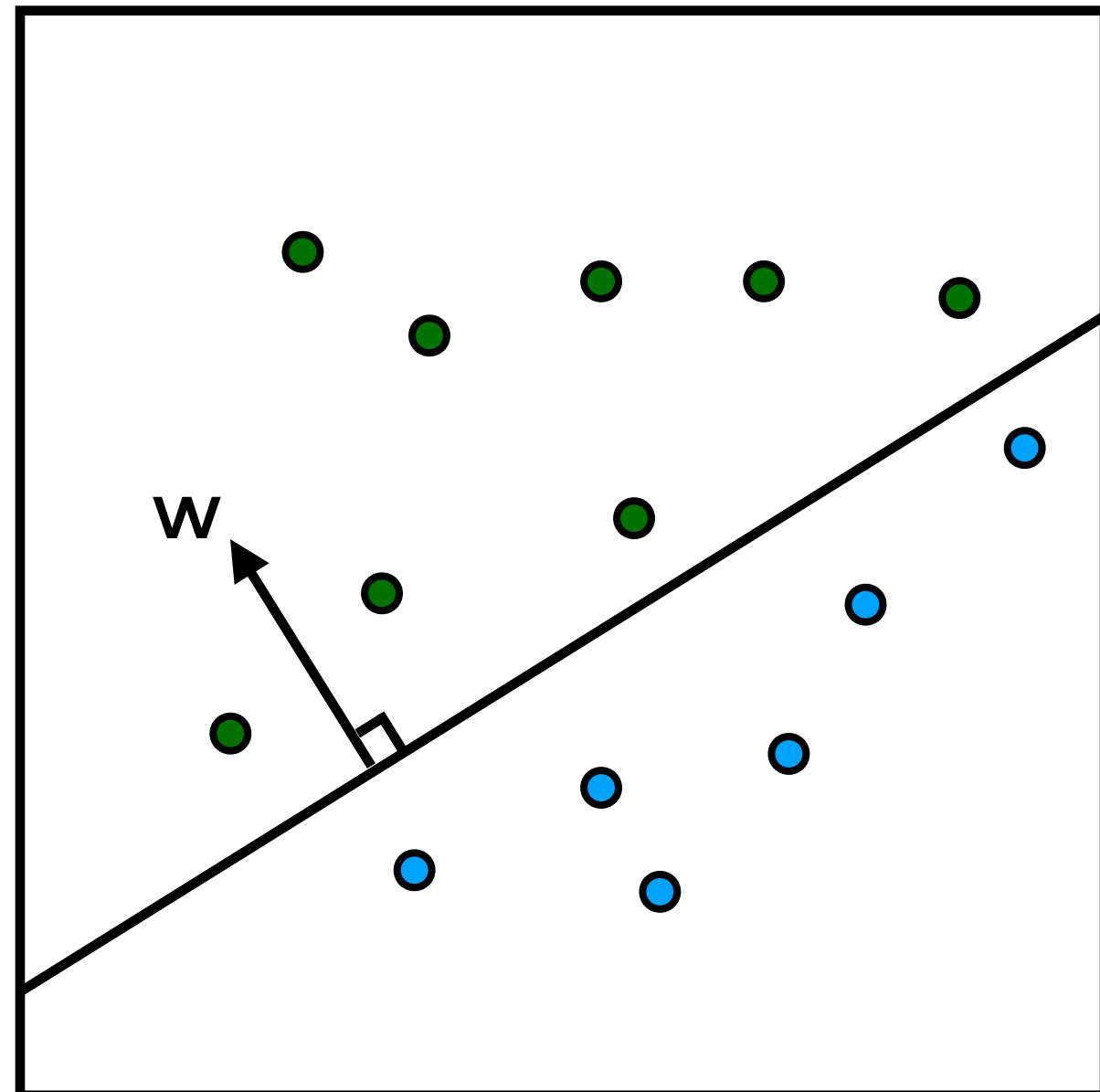
$$\mathbf{y}(\mathbf{x}) = \mathbf{w}^\top \mathbf{x} + w_0$$

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

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

Decision

# Linear Classification



decision boundary:  $y(\mathbf{x}) = 0$

take two points on the boundary:  $\mathbf{x}_a, \mathbf{x}_b$

then:  $\mathbf{w}^\top \mathbf{x}_a + w_0 = \mathbf{w}^\top \mathbf{x}_b + w_0$

$\implies \mathbf{w}^\top (\mathbf{x}_a - \mathbf{x}_b) = 0$

$\implies \mathbf{w}$  is perpendicular to the decision boundary

$\mathbf{w}$  represents the orientation of the decision boundary

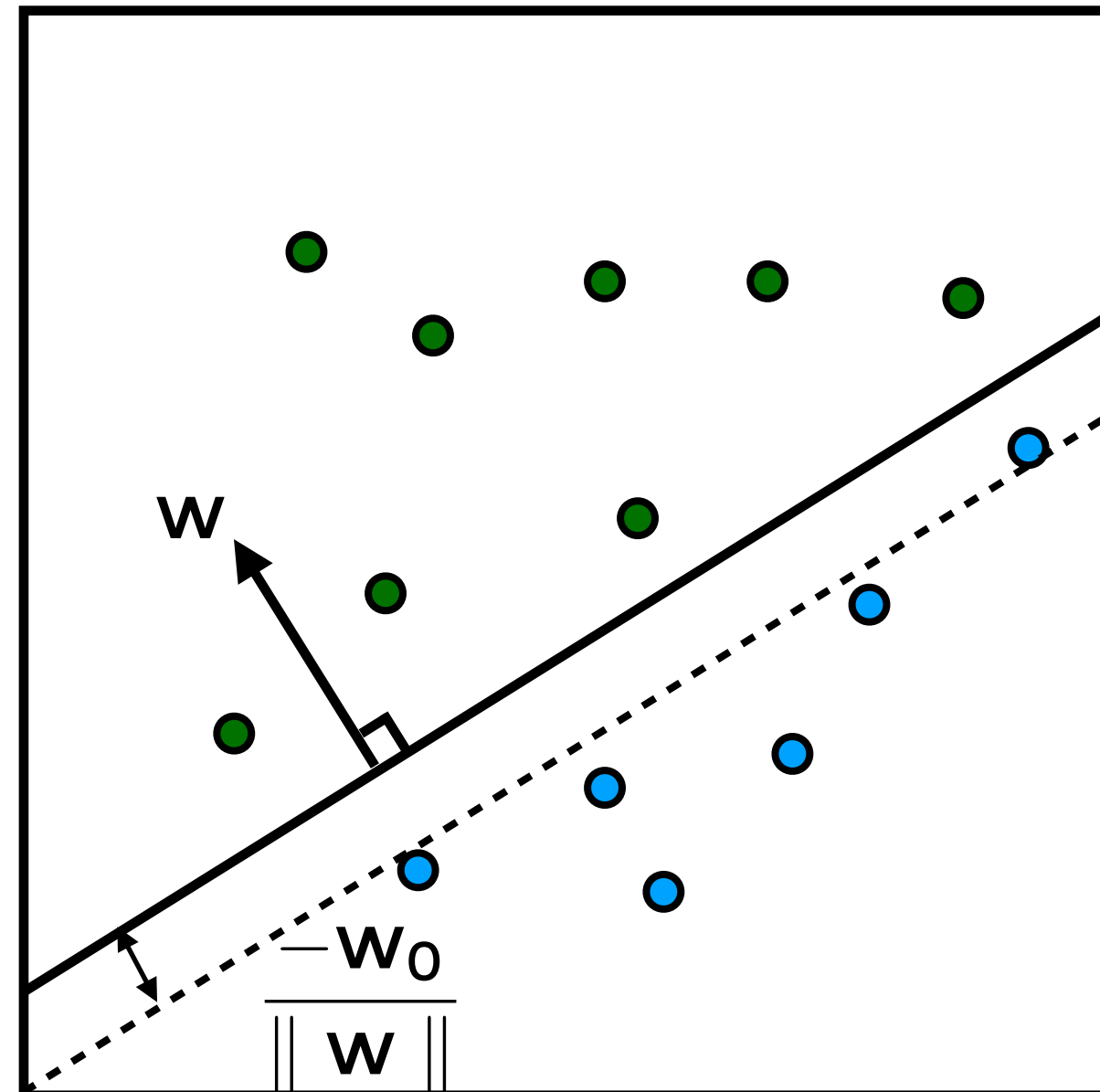
$$y(\mathbf{x}) = \mathbf{w}^\top \mathbf{x} + w_0$$

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

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

Decision

# Linear Classification



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

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

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

Decision

$w_0$  is a scalar

you can think of it like an intercept

take  $\mathbf{x}'$  as the closest point on the decision boundary to the origin

$$\mathbf{x}' = \beta \mathbf{w}$$

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

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

$$\implies 0 = \beta \|\mathbf{w}\|^2 + w_0$$

$$\implies \beta = \frac{-w_0}{\|\mathbf{w}\|^2}$$

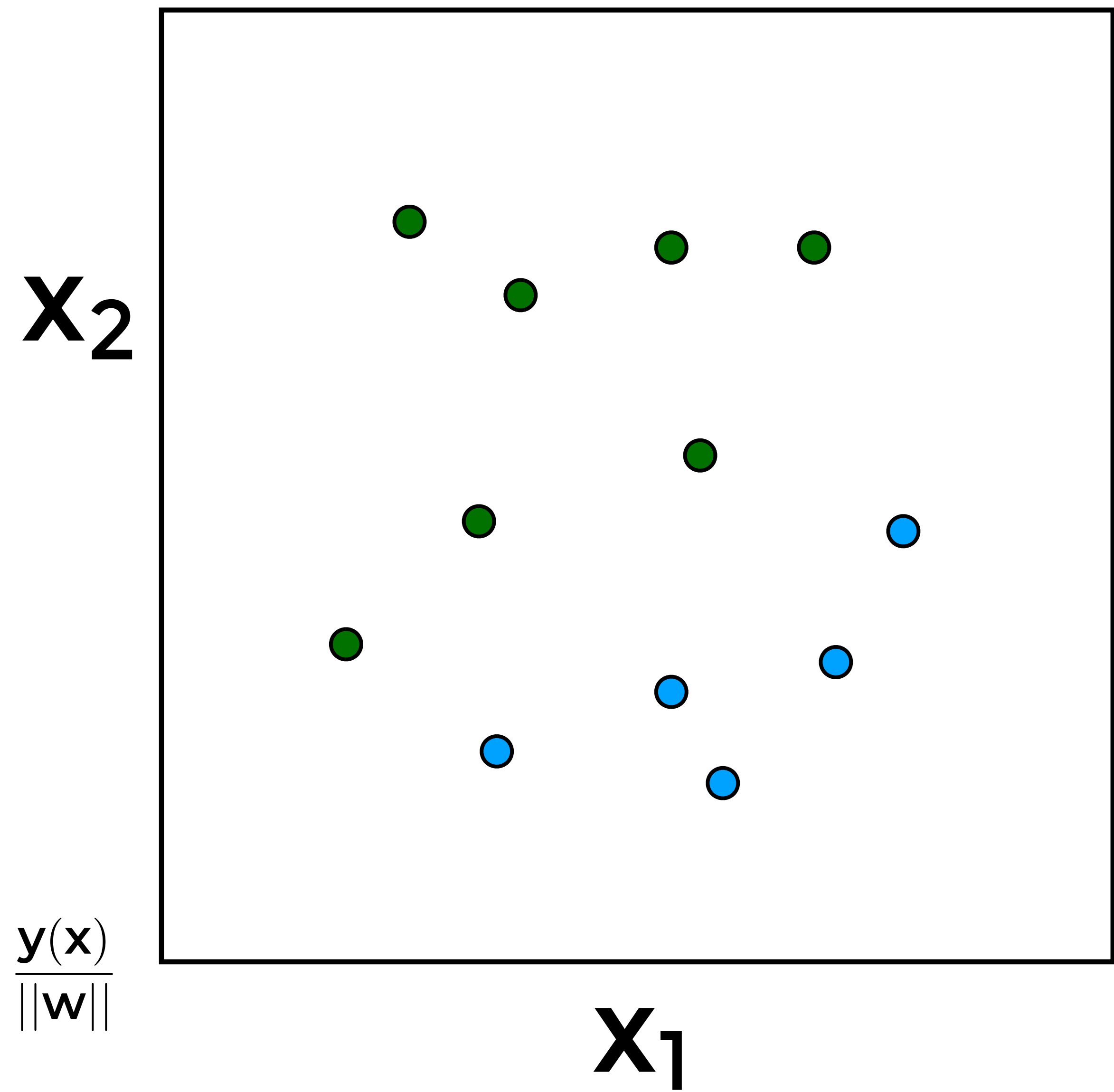
Then you know that the distance from the origin to  $\mathbf{x}'$  is:

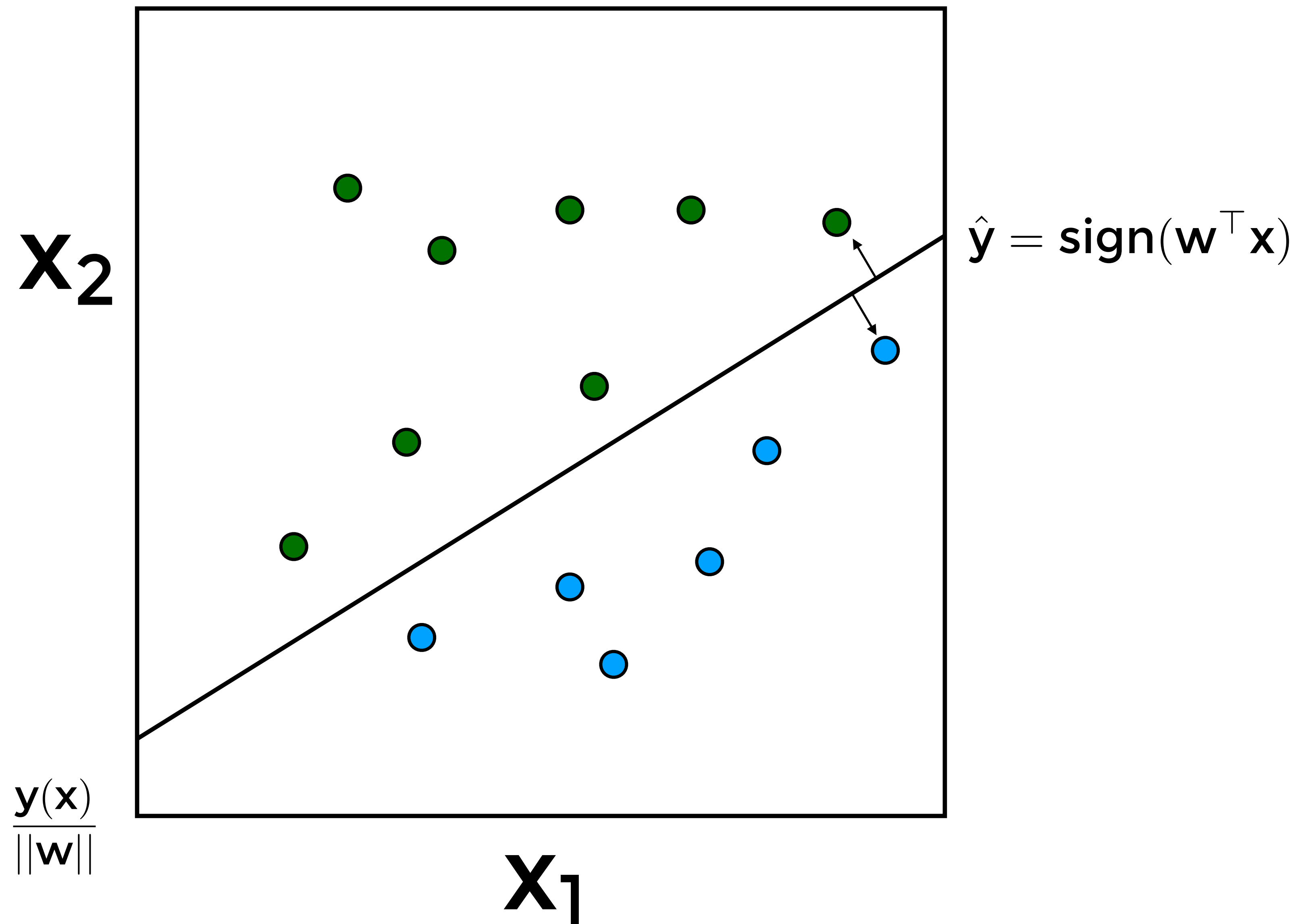
$$\|\mathbf{x}'\| = \|\beta \mathbf{w}\|$$

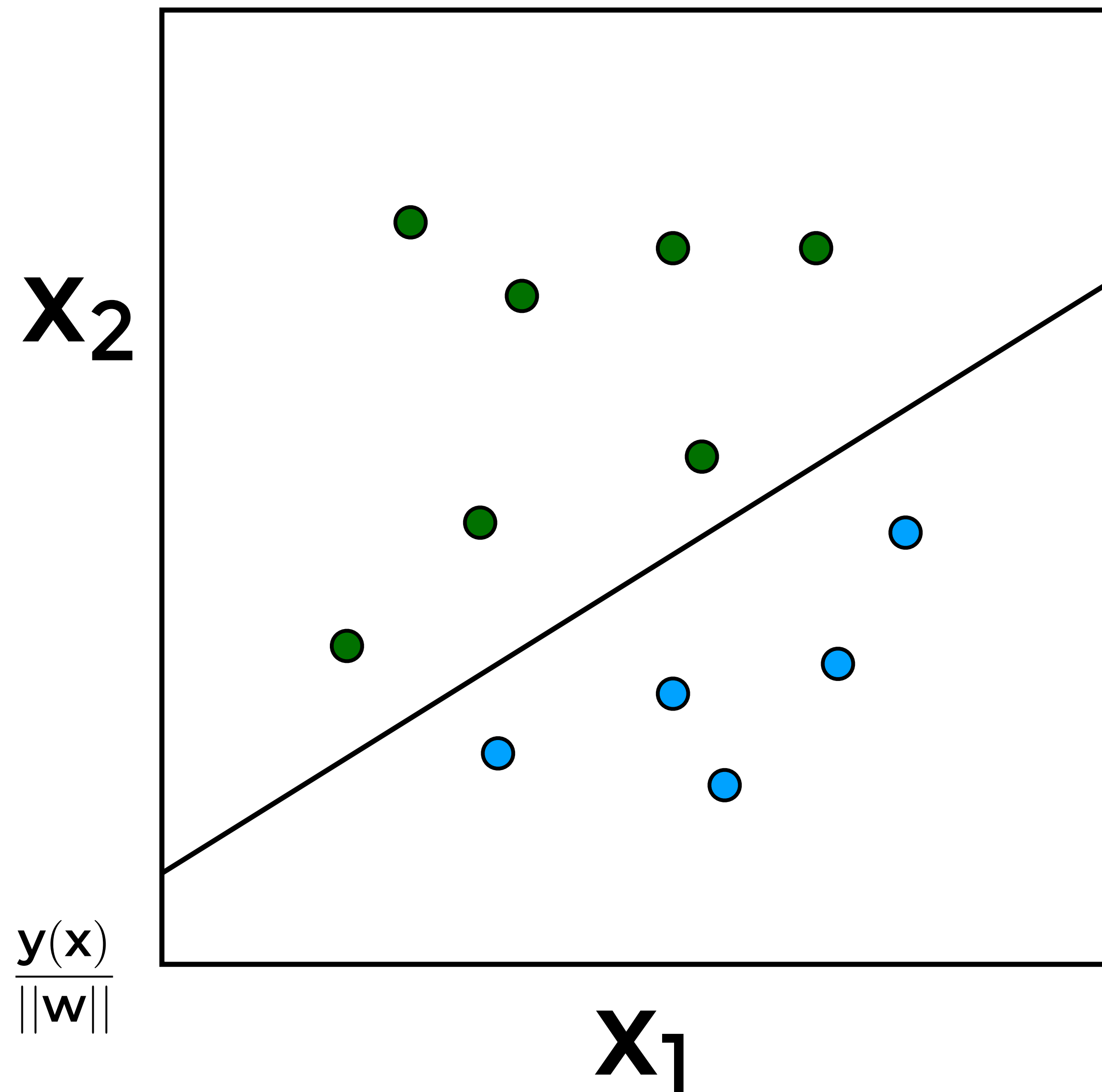
$$\implies \|\mathbf{x}'\| = \beta \|\mathbf{w}\|$$

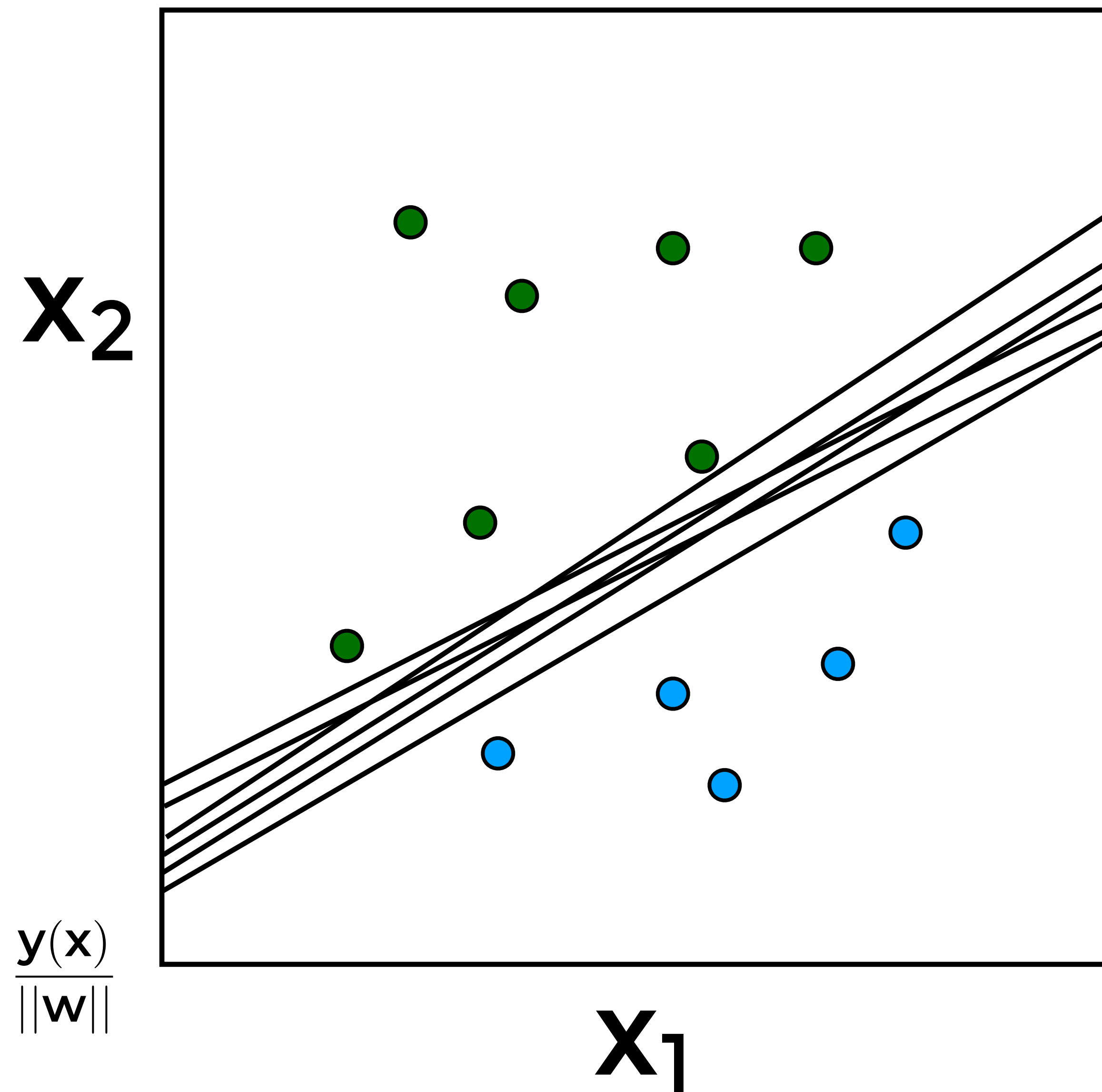
$$\implies \|\mathbf{x}'\| = \frac{-w_0}{\|\mathbf{w}\|^2} \|\mathbf{w}\|$$

$$\implies \|\mathbf{x}'\| = \frac{-w_0}{\|\mathbf{w}\|}$$

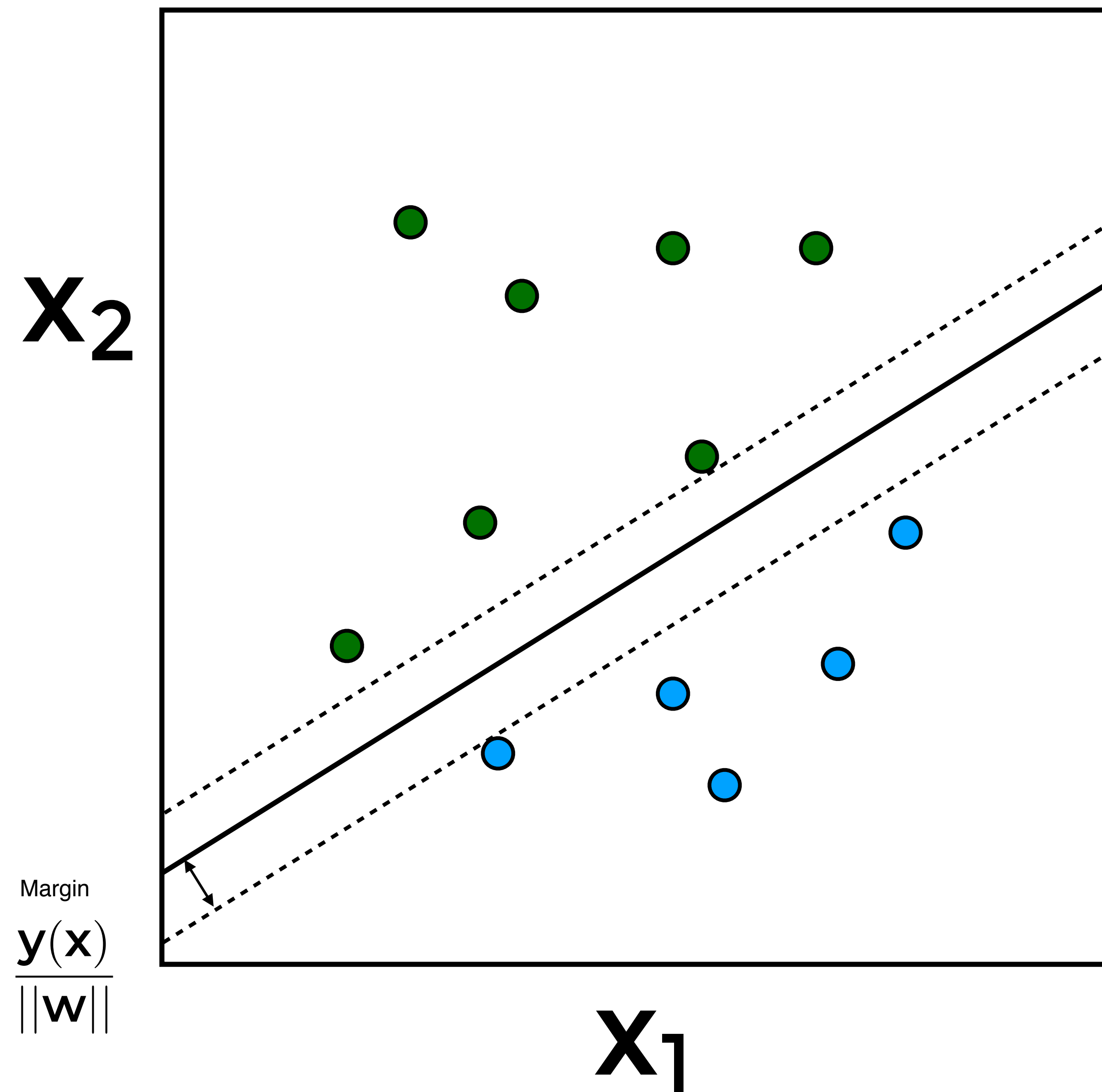




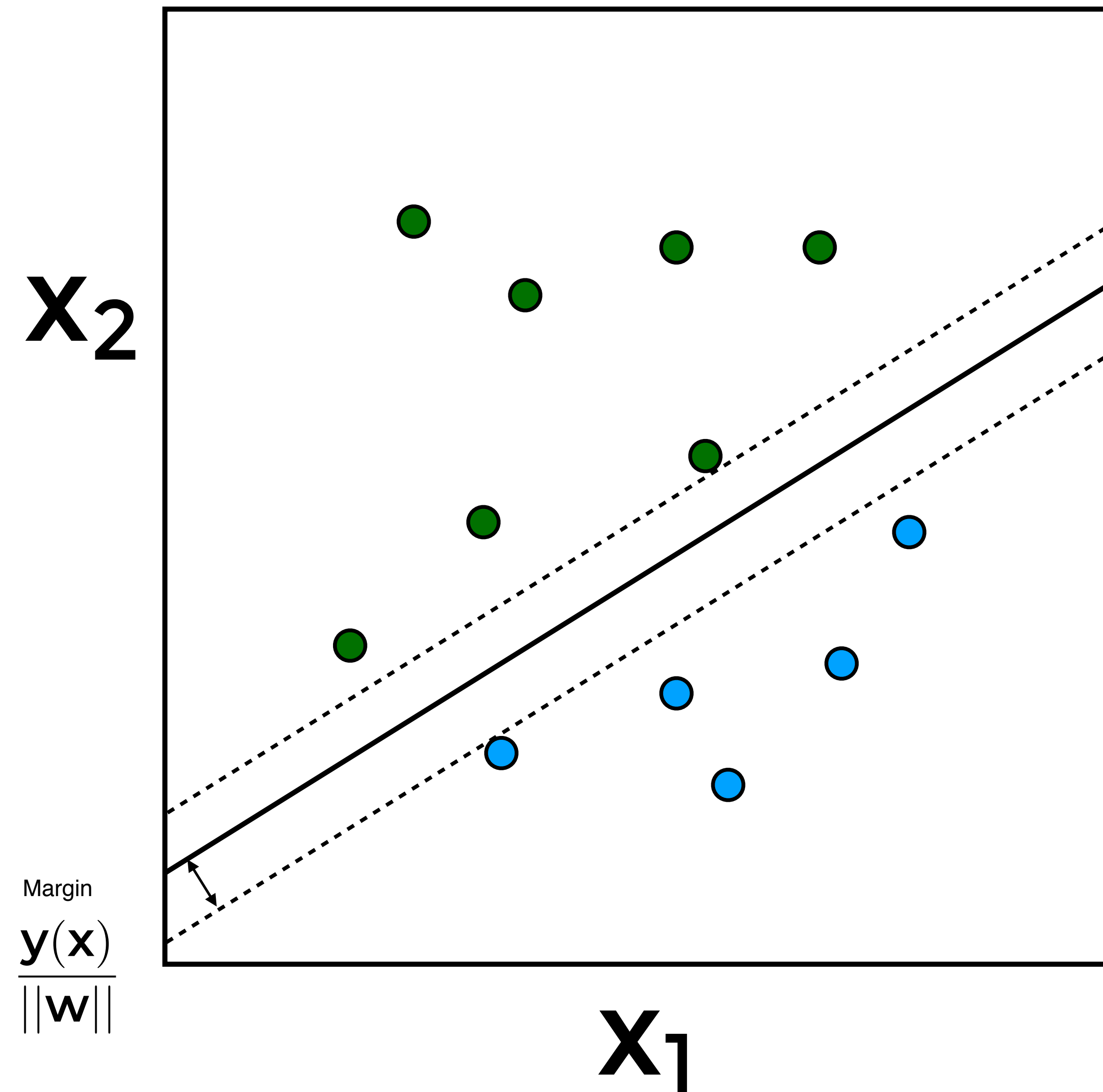








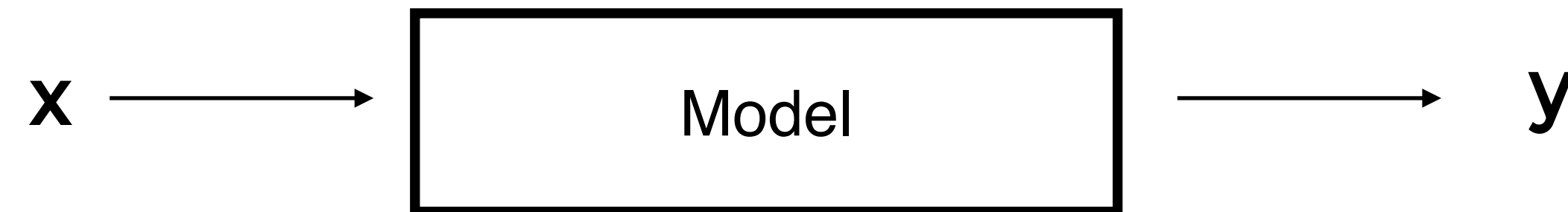
The objective is to find the separating boundary that maximizes the margin



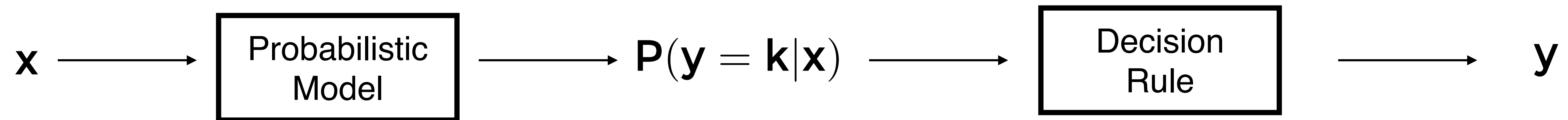
# Probabilistic Models for Classification

# Probabilistic Models separate Decision and Inference

Non-Probabilistic  
Modelling



Probabilistic  
Modelling



# Probabilistic models

1. Model the conditional directly:

$$P(y = k|x)$$

2. Model the joint (or the prior and the class conditionals):

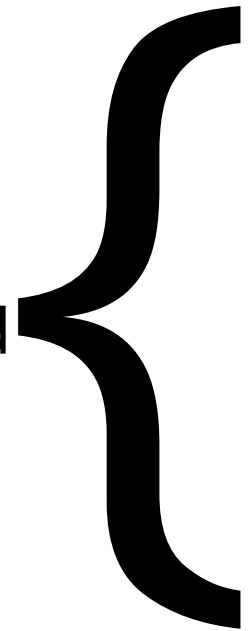
Bayes'  
Theorem

$$\underbrace{P(y = k|x)}_{\text{posterior}} \propto \underbrace{P(y = k, x)}_{\text{joint}}$$

$$= \underbrace{P(x | y = k)}_{\text{class conditional densities}} \underbrace{P(y = k)}_{\text{class prior}}$$

# Probabilistic Modelling

Often  
intertwined



**1. Posit a model:  $P(X, Y)$**

- How the data is generated

**2. Parametrize the distributions:  $P(X, Y | \text{Parameters})$**

**3. Set the objective (e.g., MLE)**

**4. Learn the parameters of the model:**

- E.g., Naive Bayes: learn the parameters of the class conditional  $P(X | Y)$  and of the prior  $P(Y)$

**5. Use the model (e.g., for predictions)**