Nearest Neighbor Classifiers and the Curse of Dimensionality

Machine Learning Course - CS-433
Oct 28, 2021
Nicolas Flammarion
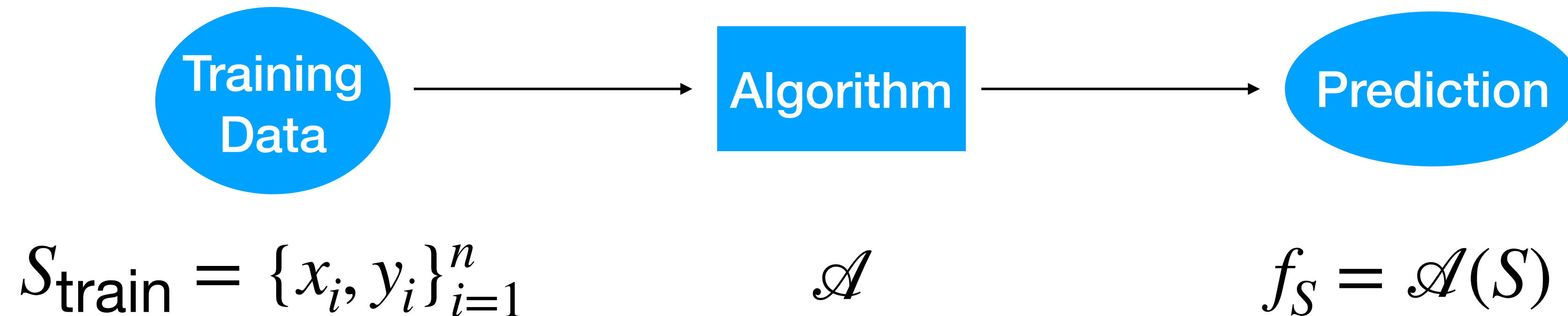
EPFL

Supervised machine learning

We observe some data $S_{\text{train}} = \{x_i, y_i\}_{i=1}^{n} \in \mathcal{X} \times \mathcal{Y}$

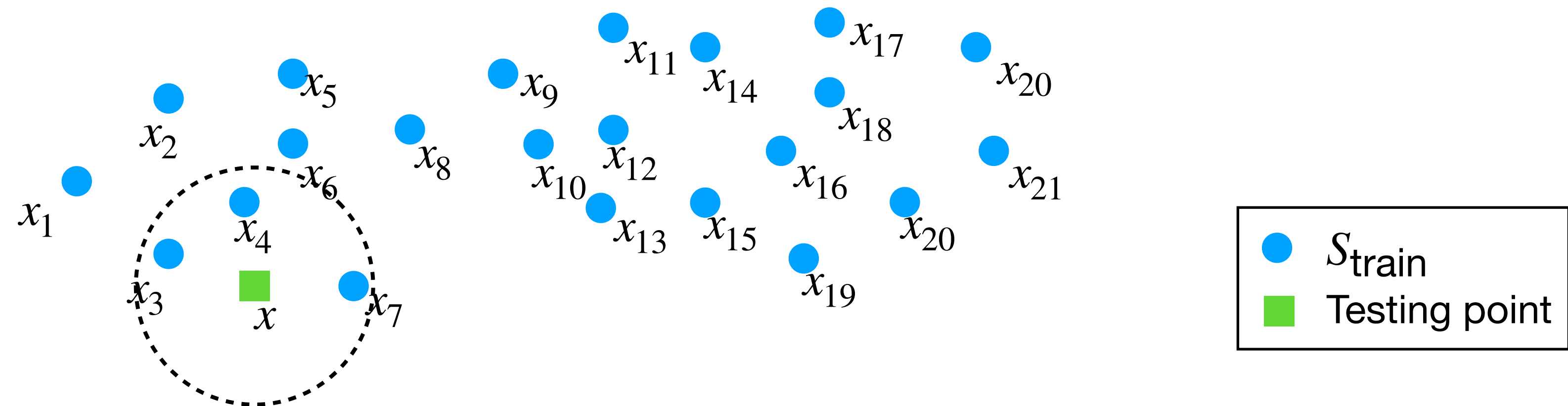<u>Goal</u>: given a new x, we want to predict its label y

<u>How:</u>

$$S_{\text{train}} = \{x_i, y_i\}_{i=1}^{n} \qquad \mathcal{A} \qquad f_S = \mathcal{A}(S)$$

Nearest neighbor function

$$\text{nbh}_{S_{train},k} : \mathcal{X} \to \mathcal{X}^k$$

$$x \mapsto \{k \text{ elements of } S_{\text{train}} \text{ the closest to } x\}$$

$$\text{nbh}_{S_{train},3}(x) = \{x_3, x_4, x_7\}$$

Nearest neighbor function

$$\text{nbh}_{S_{train},k} : \mathcal{X} \to \mathcal{X}^k$$

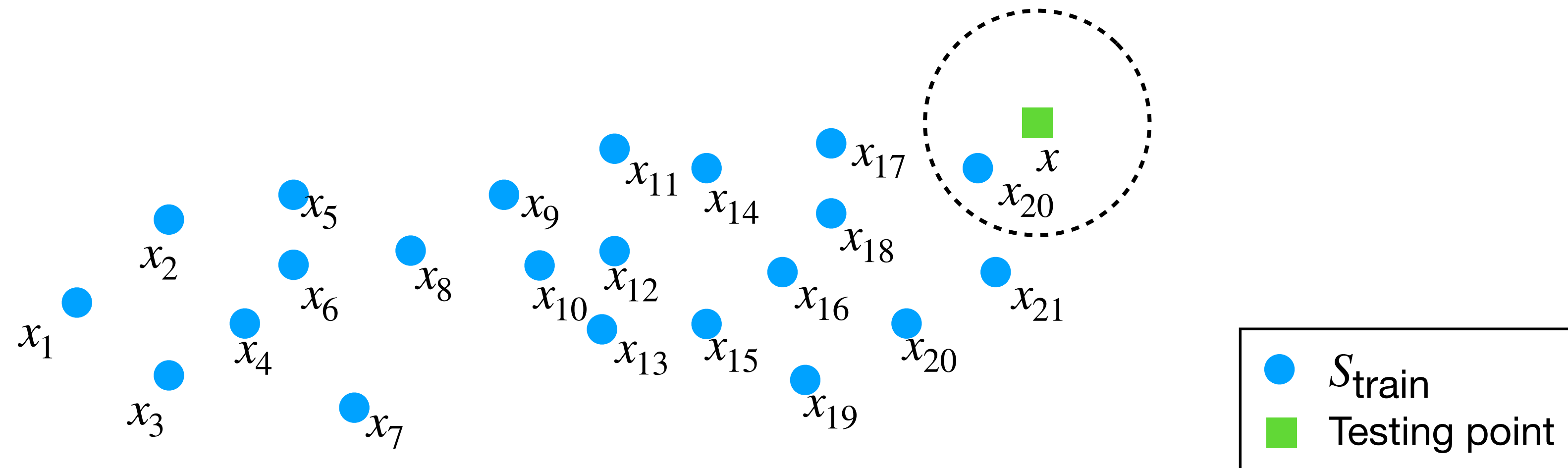$$x \mapsto \{k \text{ elements of } S_{\text{train}} \text{ the closest to } x\}$$

x_1
x_2
x_3
x_4
x_5
x_6
x_7
x_8
x_9
x_{10}
x_{11}
x_{12}
x_{13}
x_{14}
x_{15}
x_{16}
x_{17}
x_{18}
x_{19}
x_{20}
x_{20}
x_{21}
x

Legend:
- S_{train}
- Testing point

$$\text{nbh}_{S_{train},1}(x) = \{x_{20}\}$$

Nearest neighbor function

$$\text{nbh}_{S_{train},k} : \mathcal{X} \to \mathcal{X}^k$$

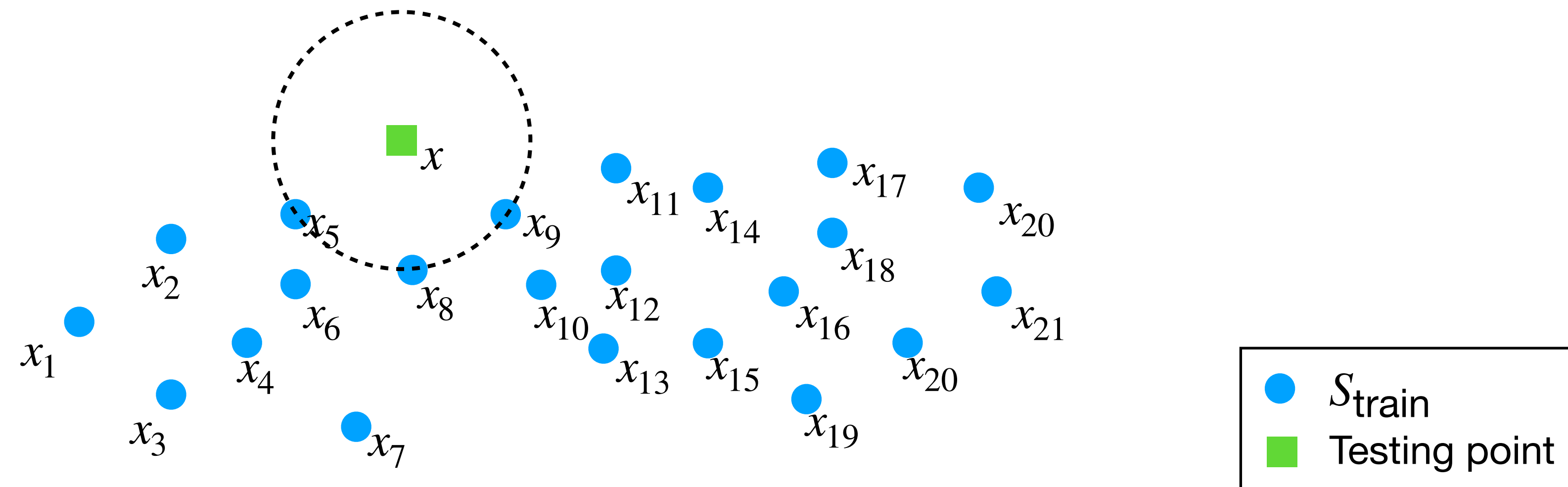$$x \mapsto \{k \text{ elements of } S_{\text{train}} \text{ the closest to } x\}$$

$$\text{nbh}_{S_{train},2}(x) = \{x_5, x_8\}$$

Not uniquely defined!
It will depend on the implementation
Often ties are broken randomly

Nearest neighbor function

$$\text{nbh}_{S_{train},k} : \mathcal{X} \to \mathcal{X}^k$$

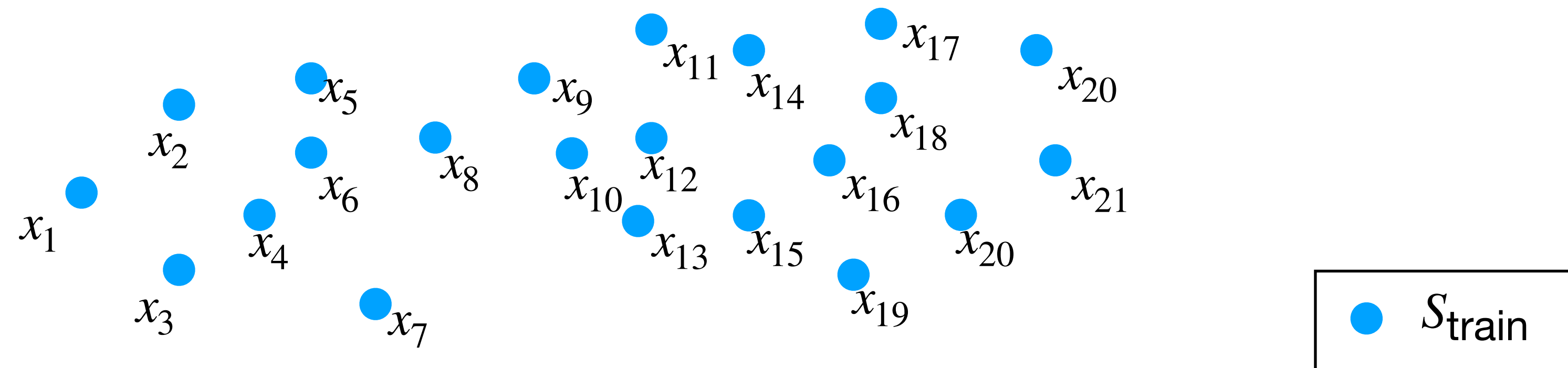$$x \mapsto \{k \text{ elements of } S_{\text{train}} \text{ the closest to } x\}$$
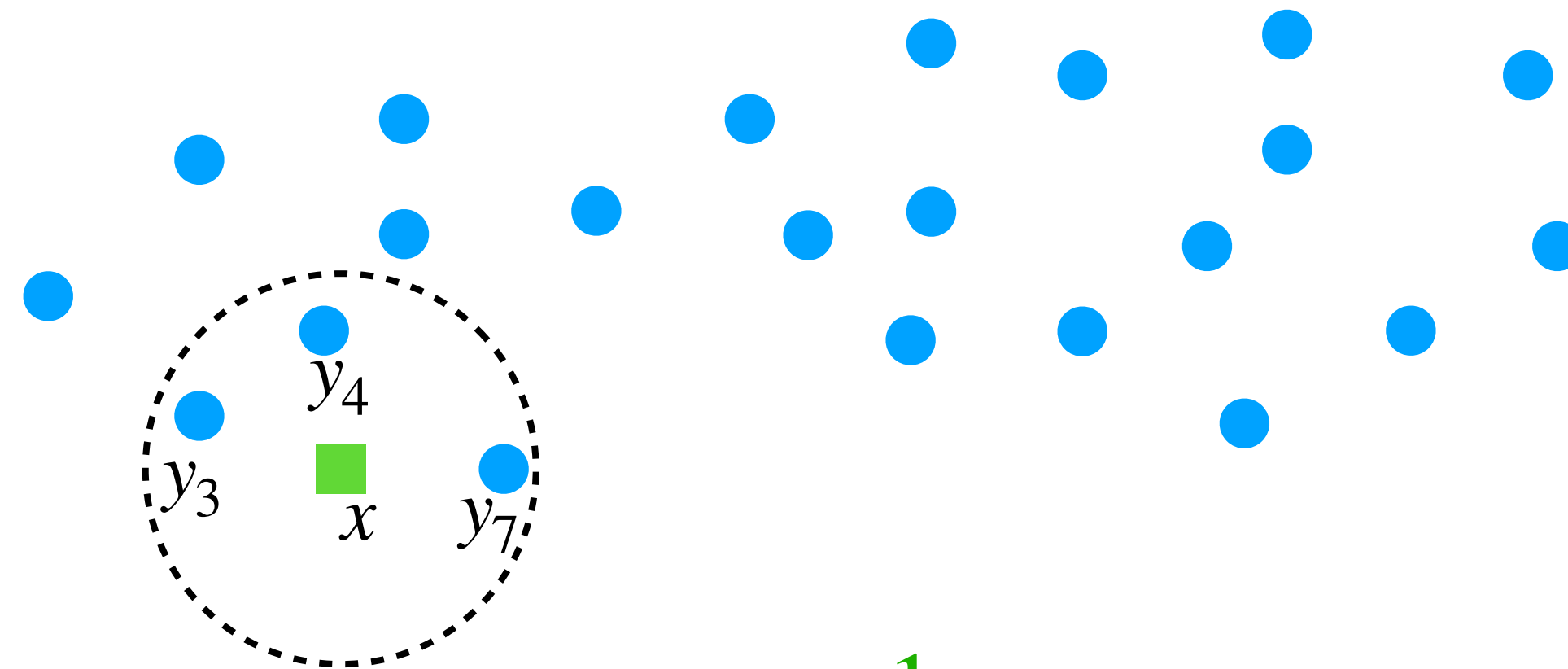

Rmq:

- Different metrics can be used

- Computational complexity when n is large (but efficient data structure may exist)

k-NN can be used for regression ($y \in \mathbb{R}$)
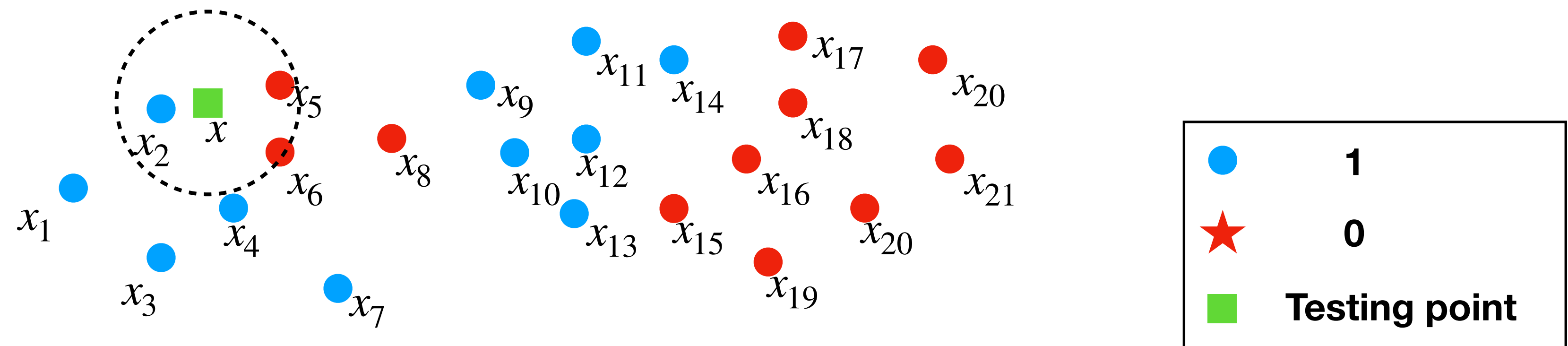
$$f_{S_{train},k}(x) = \frac{1}{k} \sum_{i:x_i \in nbh_{S_{train},k}(x)} y_i$$

$$f_{S_{train},3}(x) = \frac{1}{3}\left(y_3 + y_4 + y_7\right)$$

Legend:
- S_{train}
- Testing point

k-NN can be used for classification ($y \in \{0,1\}$)

$$f_{S_{train},k}(x) = \text{majority}\{y_i : x_i \in \text{nbh}_{S_{train},k}(x)\}$$
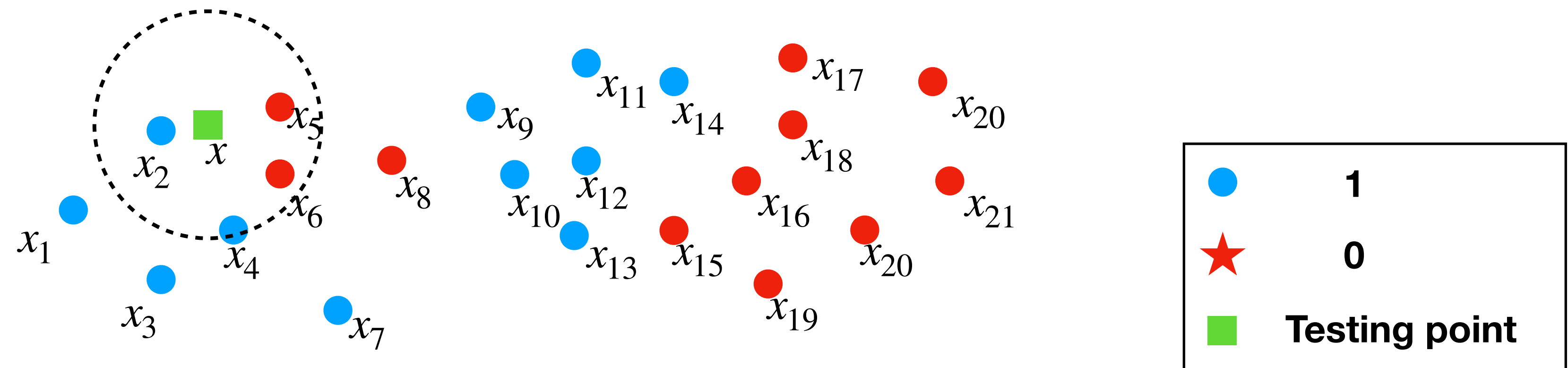

$$f_{S_{train},3}(x) = 0$$
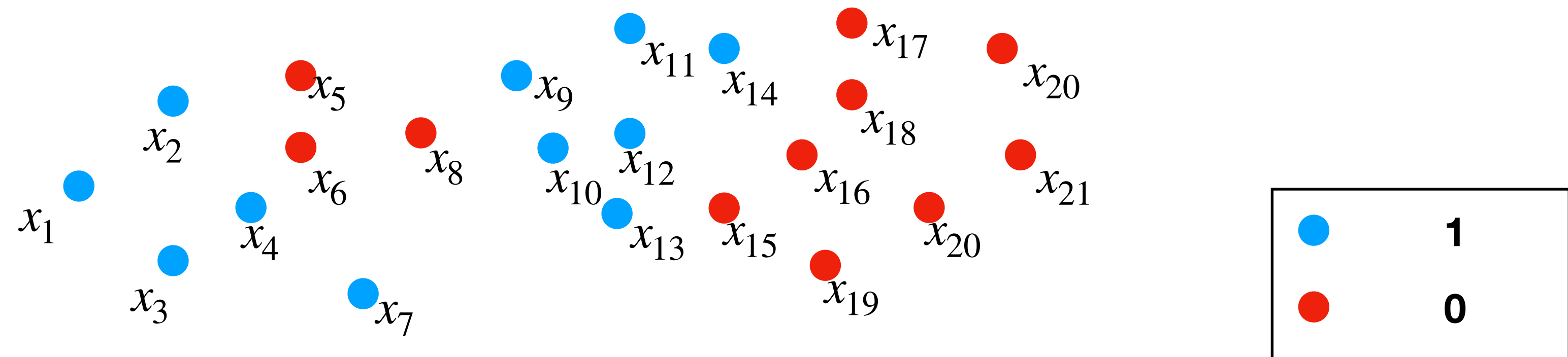
k-NN can be used for classification ($y \in \{0,1\}$)

$$f_{S_{train},k}(x) = \text{majority}\left\{y_i : x_i \in \text{nbh}_{S_{train},k}(x)\right\}$$

$$f_{S_{train},4}(x) = ? \quad \textbf{Tie!}$$

k-NN can be used for classification ($y \in \{0,1\}$)

$$f_{S_{train},k}(x) = \text{majority}\left\{y_i : x_i \in \text{nbh}_{S_{train},k}(x)\right\}$$
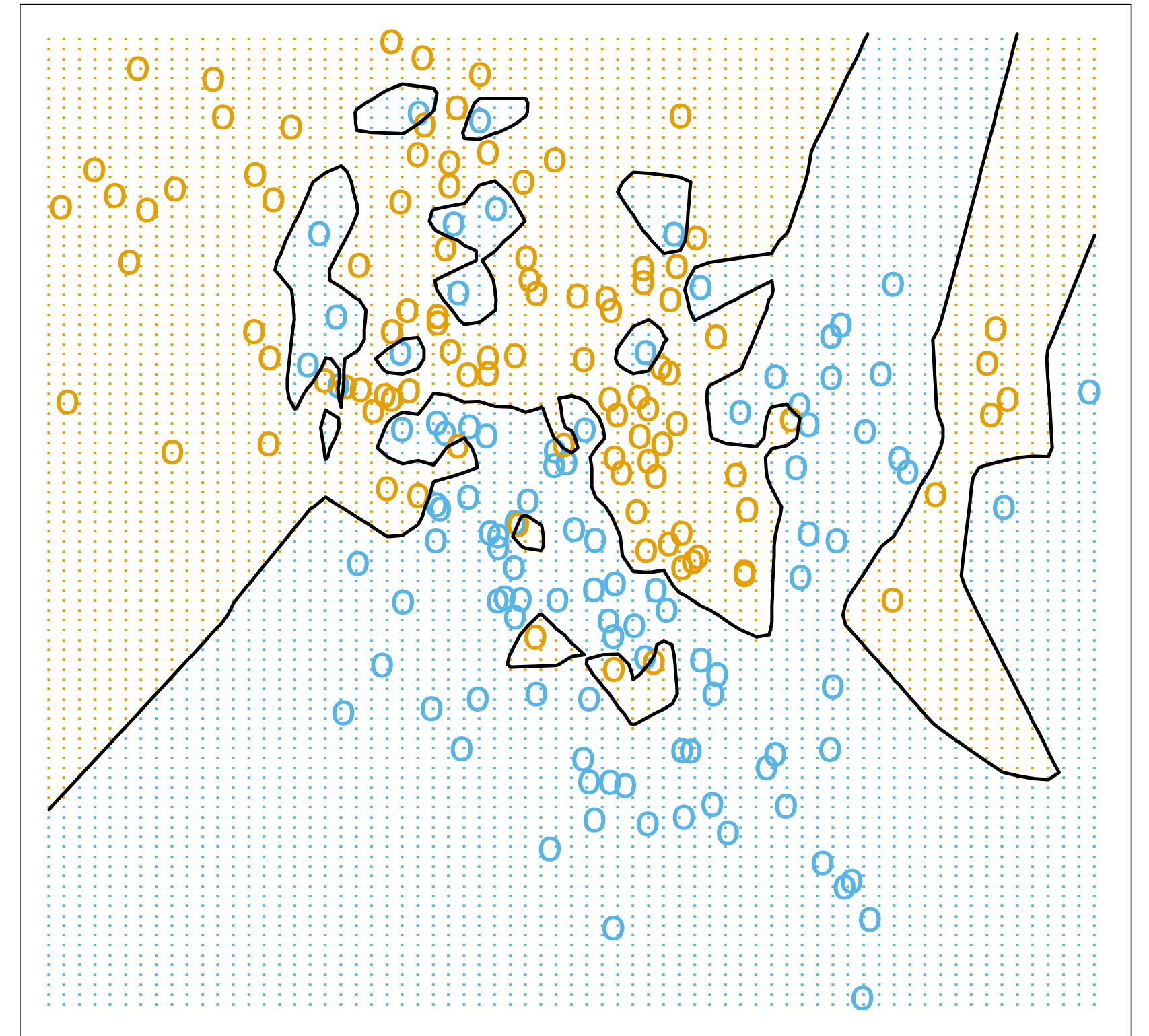

Rmq:

- k is often chosen odd to avoid ties

- Generalization: smoothing kernels; weighted linear combination of elements

Why does it make sense?

- Meaningful when there is spatial correlation

- Implicitly learns very complex decision boundaries in low dimension

Bias Variance for k-NN

Small k:

   Small bias - complex decision boundary

   Large variance - overfitting

Large k: ($k = n$ constant prediction)
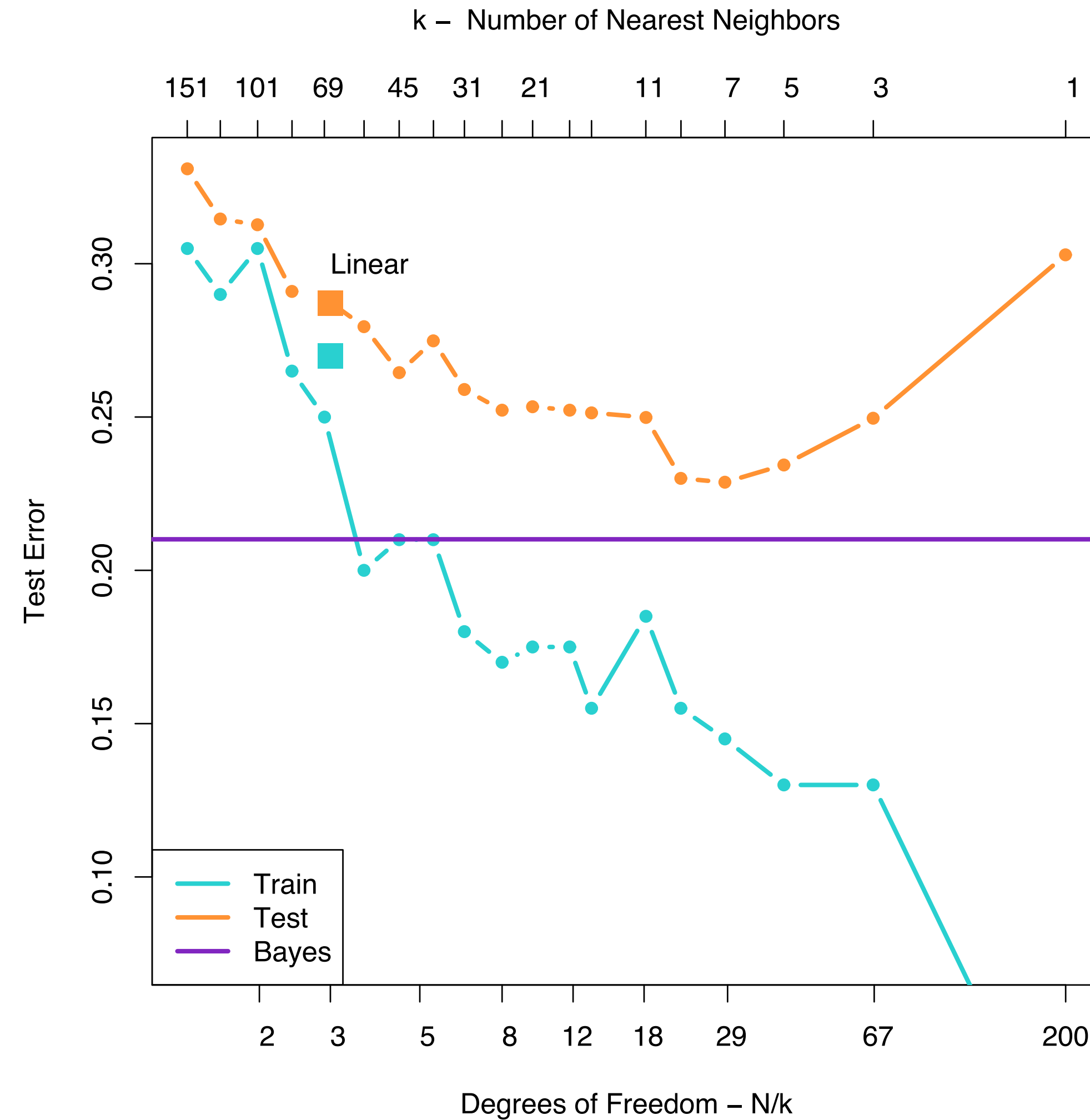
   Large bias

   Small variance
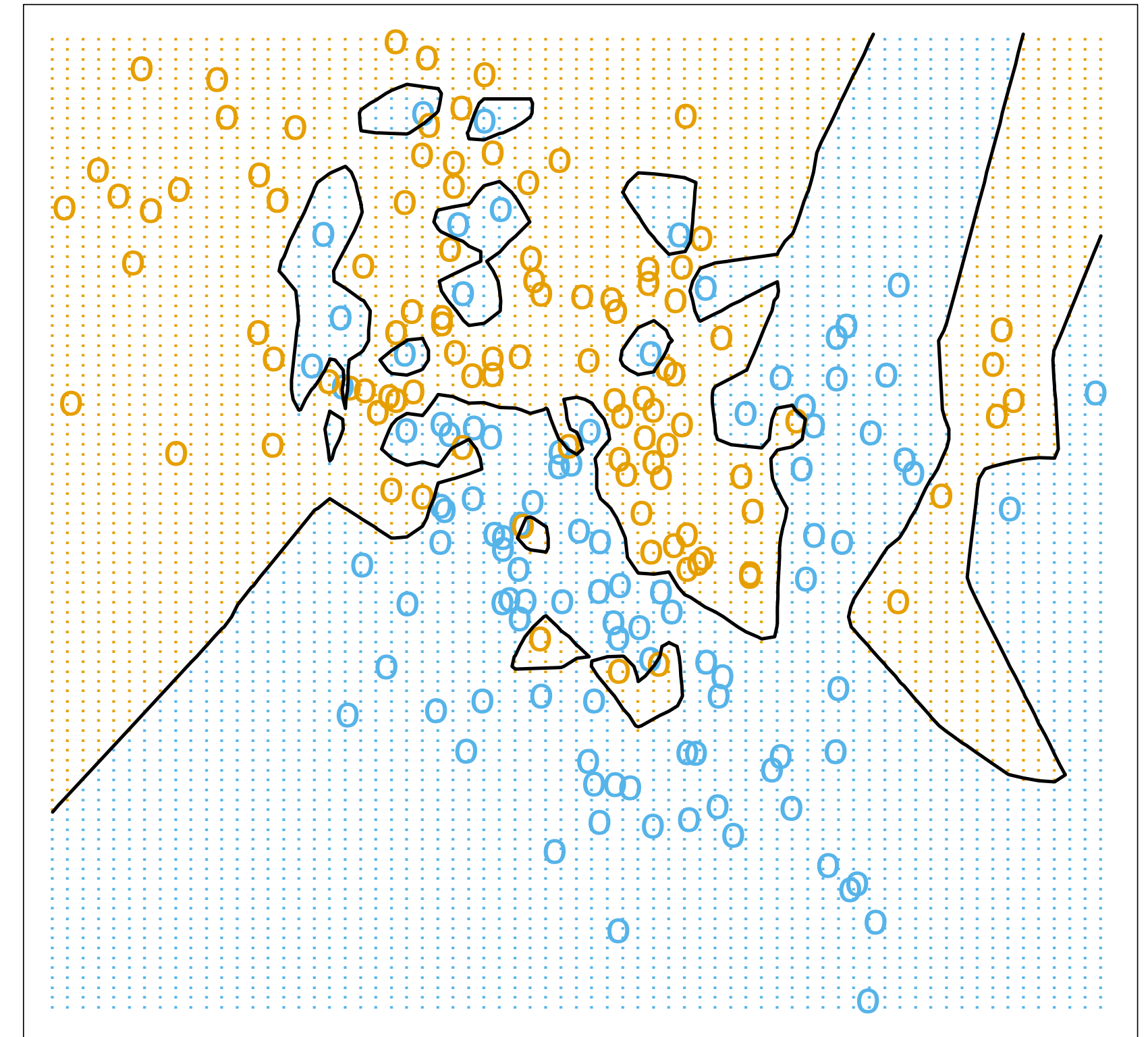
1-nearest neighbor classification

U-shape curve for k-NN bias-variance tradeoff

Complexity increases when k decreases

Find a k which balances the bias and the variance

Good k:

Small bias - complex enough decision boundary

Small variance - no overfitting
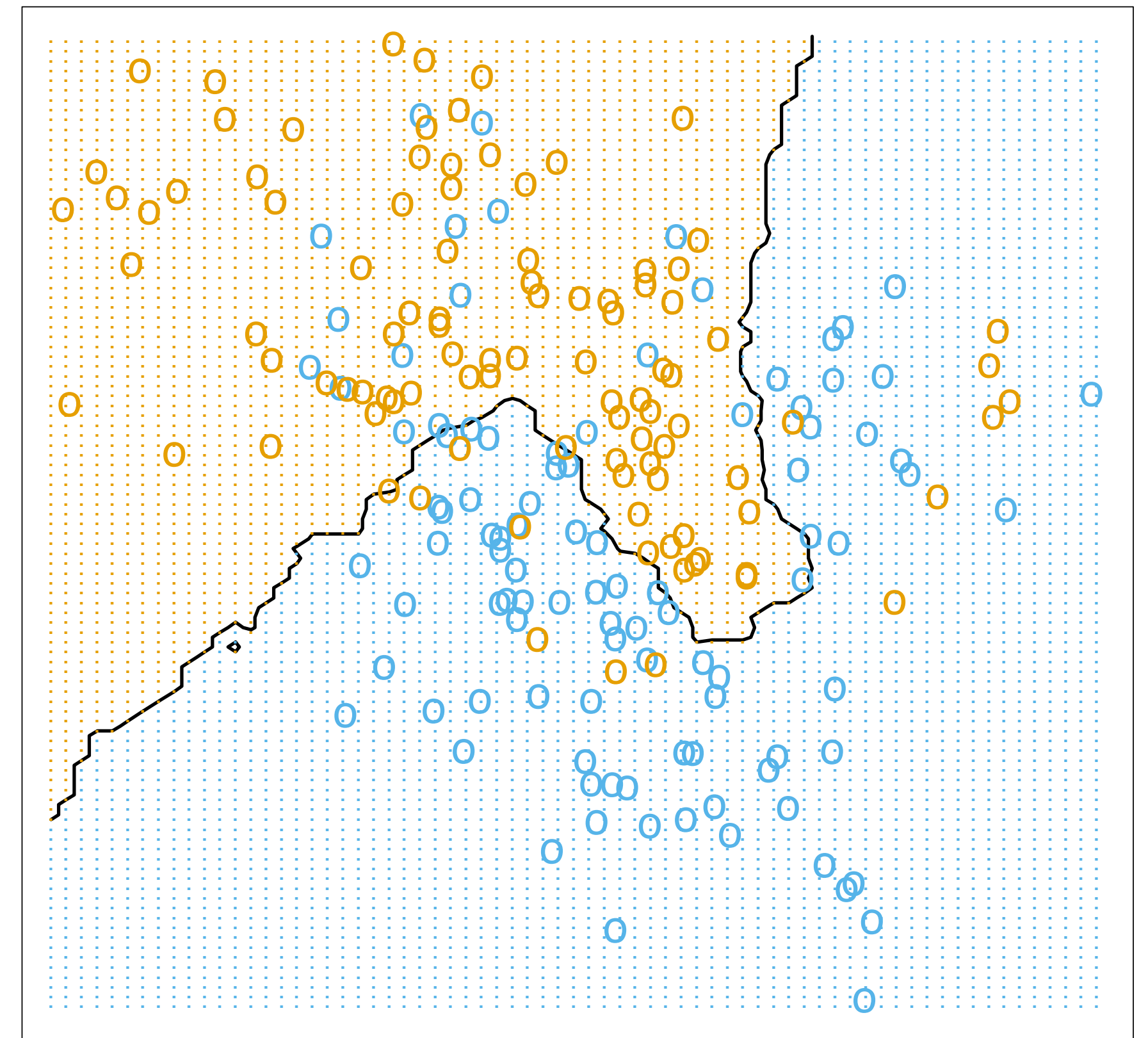
15-nearest neighbor classification

Curse of dimensionality

Claim 1: Fixed-size training sets cover a dwindling fraction of the input space.

Assume the data $x \sim \mathcal{U}([0,1]^d)$

Consider a blue box around the center x_0 of size r

$$\mathbb{P}(x \in \blacksquare \,) = r^d := \alpha$$

x_0

r

$\mathcal{X} = [0,1]^d$

Curse of dimensionality

Claim 1: Fixed-size training sets cover a dwindling fraction of the input space.

Assume the data $x \sim \mathcal{U}([0,1]^d)$

Consider a blue box around the center x_0 of size r

$$\mathbb{P}(x \in \square) = r^d := \alpha$$

If $d = 10$, to have:

$\alpha = 0.01$ we need $r = 0.63$

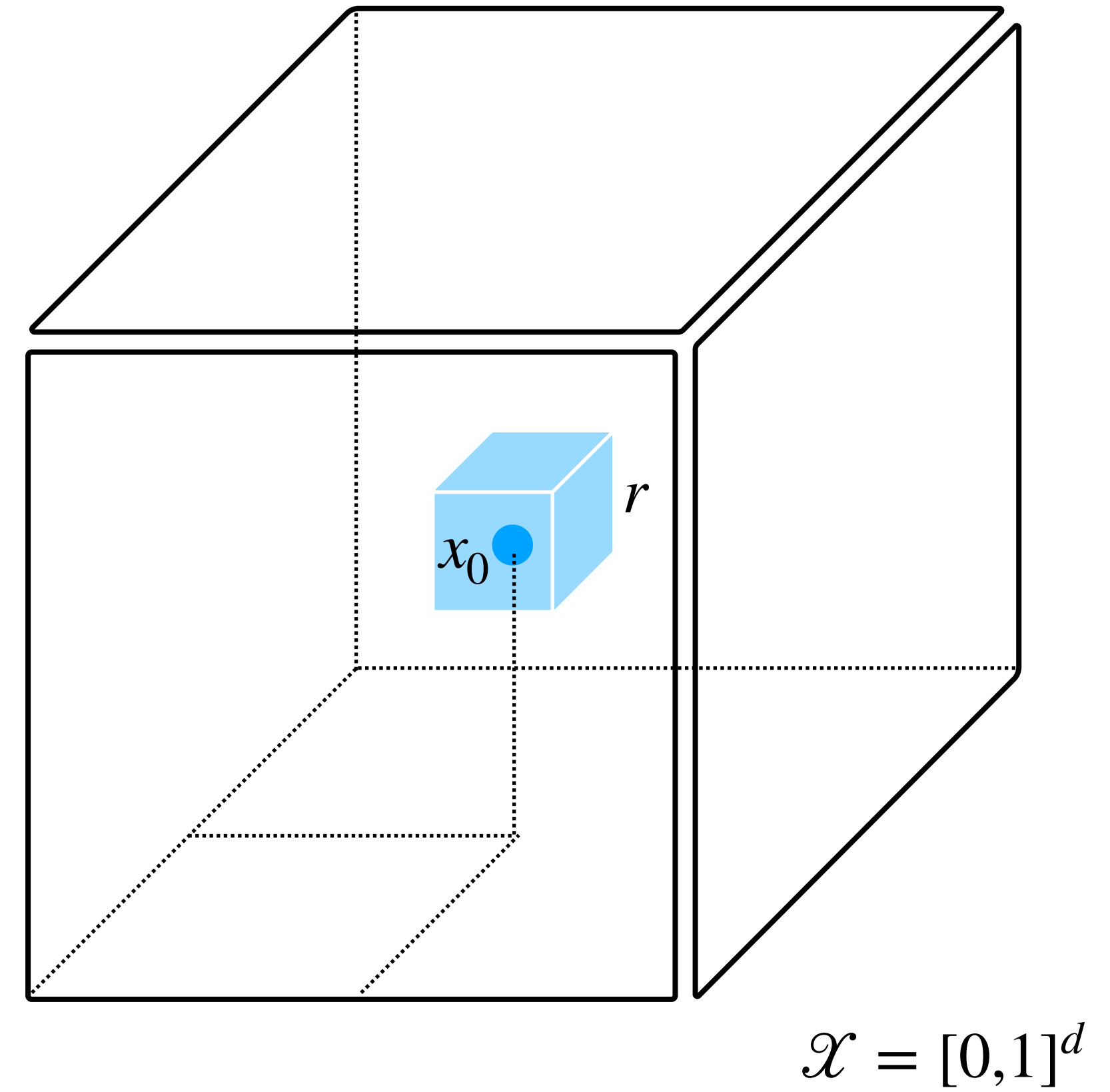$r = 0.63$

x_0

$\mathcal{X} = [0,1]^d$

Curse of dimensionality

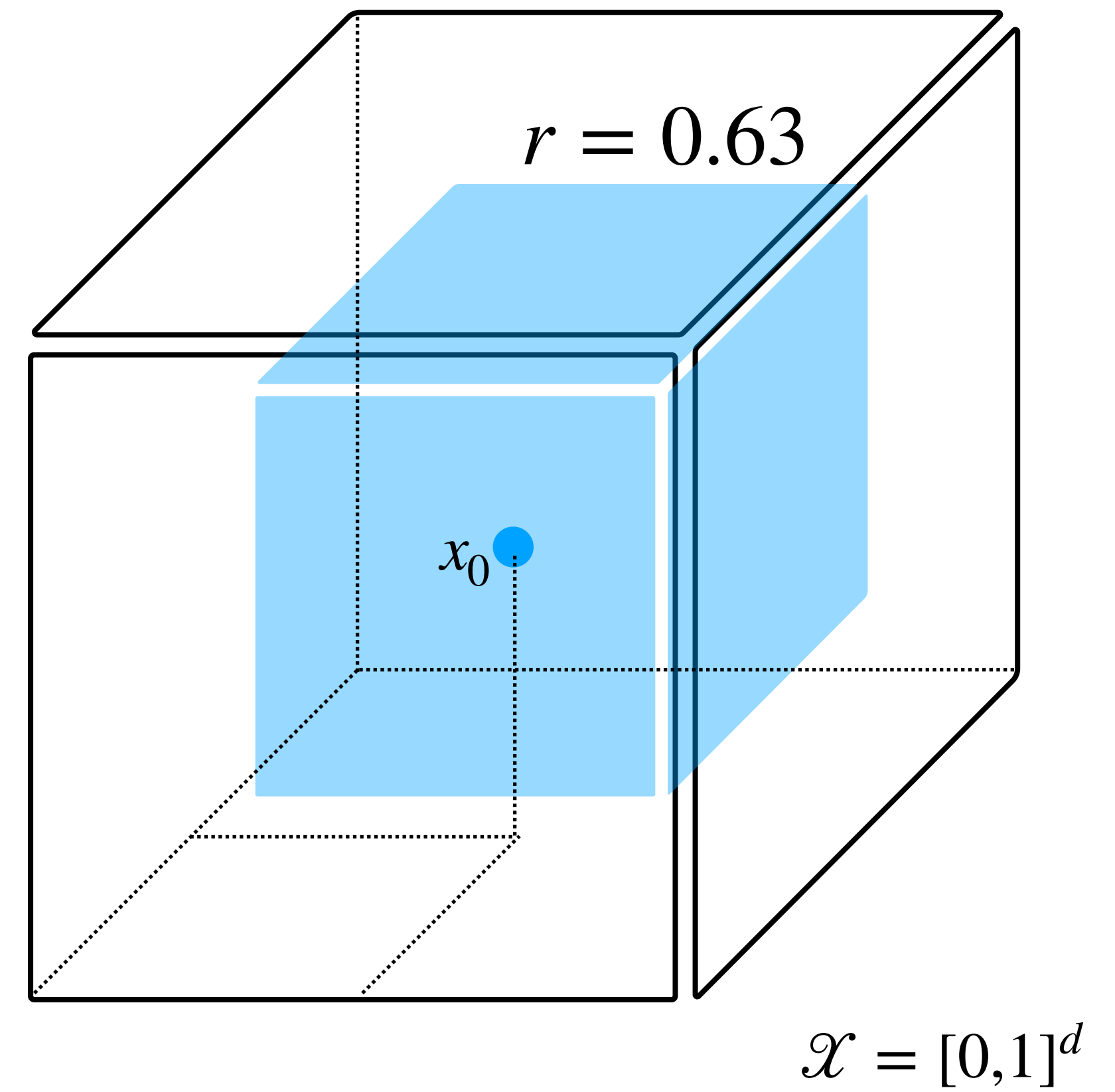Claim 1: Fixed-size training sets cover a dwindling fraction of the input space.

Assume the data $x \sim \mathcal{U}([0,1]^d)$

Consider a blue box around the center x_0 of size r

$$\mathbb{P}(x \in \square \;) = r^d := \alpha$$

If $d = 10$, to have:

$\quad \alpha = 0.01$ we need $r = 0.63$

$\quad \alpha = 0.1$ we need $r = 0.8$

We need to explore almost the whole box

$r = 0.8$

x_0

$\mathcal{X} = [0,1]^d$

Curse of dimensionality

Claim 2: In high-dimension, data-points are far from each other.

Consider n i.i.d. points uniform in the $[0,1]^d$

$$\mathbb{P}(\exists x_i \in \,\blacksquare\,) \geq 1/2 \implies r \geq \left(1 - \frac{1}{2^{1/n}}\right)^{1/d}$$

Proof: $\mathbb{P}(x \notin \,\blacksquare\,) = 1 - r^d$

$\mathbb{P}(x_i \notin \,\blacksquare\,, \forall i \leq n) = (1 - r^d)^n$

$\mathbb{P}(\exists x_i \in \,\blacksquare\,) = 1 - (1 - r^d)^n$

$\mathscr{X} = [0,1]^d$

Curse of dimensionality

Claim 2: In high-dimension, data-points are far from each other.
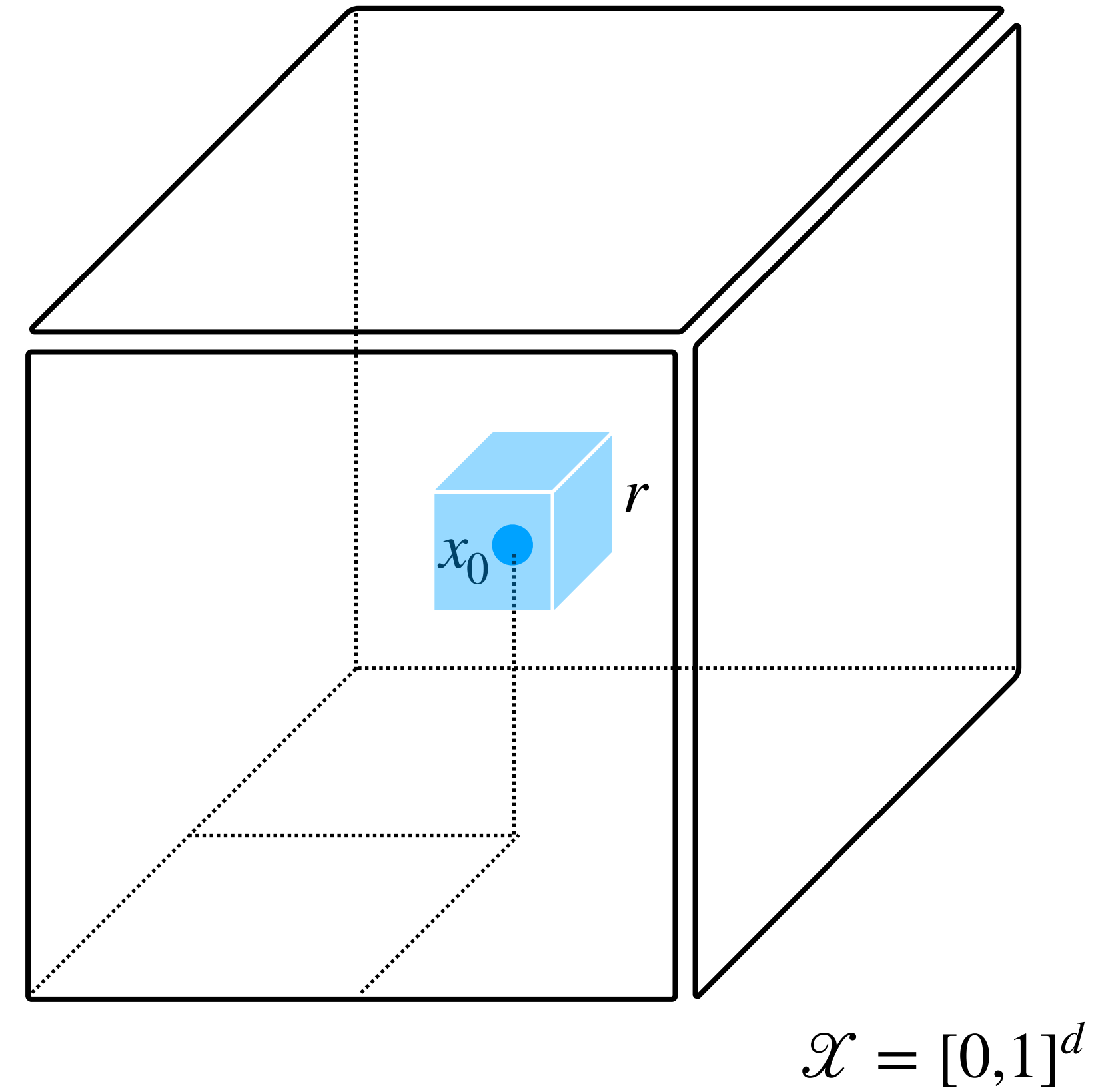
Consider n i.i.d. points uniform in the $[0,1]^d$
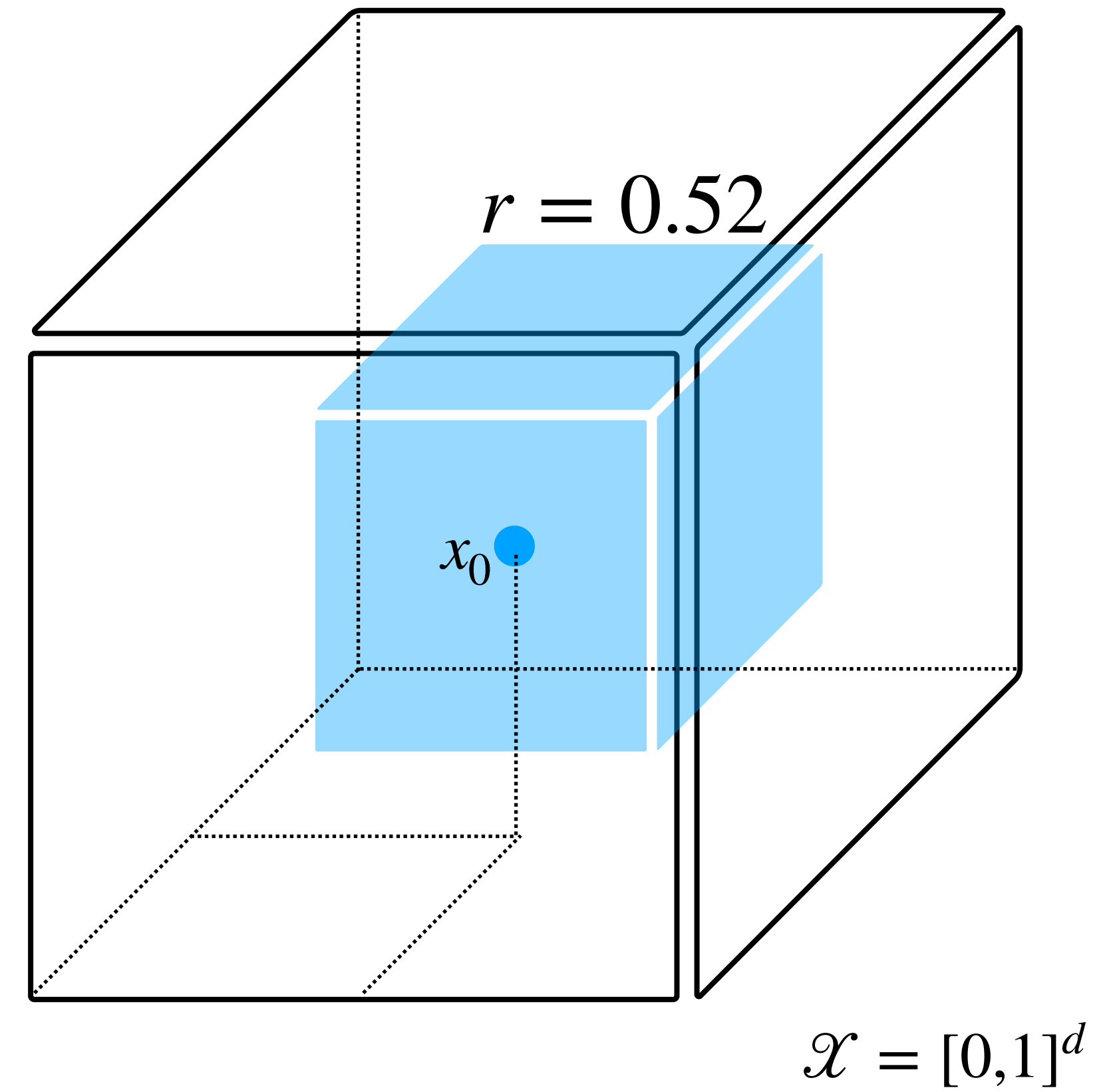
$$\mathbb{P}(\exists x_i \in \text{◻}) \geq 1/2 \implies r \geq \left(1 - \frac{1}{2^{1/n}}\right)^{1/d}$$

Proof: $\mathbb{P}(x \notin \text{◻}) = 1 - r^d$

$\mathbb{P}(x_i \notin \text{◻}, \forall i \leq n) = (1 - r^d)^n$

$\mathbb{P}(\exists x_i \in \text{◻}) = 1 - (1 - r^d)^n$

For $d = 10$, $n = 500$, we have $r \geq 0.52$

$r = 0.52$

x_0

$\mathcal{X} = [0,1]^d$

Generalization bound for 1-NN

Setup: $(X, Y) \sim \mathcal{D}$ over $\mathcal{X} \times \mathcal{Y} = [0,1]^d \times \{0,1\}$

Goal: Bound the classification error:

$$L(g) = \mathbb{P}_{(X,Y)\sim\mathcal{D}}(Y \neq g(X))$$

Baseline:

- Bayes classifier - minimizes L over all classifiers

$$g_*(x) = 1_{\eta(x)\geq 1/2} \text{ where } \eta(x) = \mathbb{P}(Y = 1 \,|\, X = x)$$

- Bayes risk - smallest probability of misclassification

$$L(g_*) = \mathbb{P}(g_*(X) \neq Y) = \mathbb{E}_{X\sim\mathcal{D}_X}[\min\{\eta(X),1-\eta(X)\}]$$

Generalization bound for 1-NN

Setup: $(X, Y) \sim \mathcal{D}$ ov

Goal: Bound the classif

Baseline:

Proof 1:

$$\eta(x) \geq 1/2 \iff \mathbb{P}(Y = 1 \,|\, X = x) \geq 1/2$$

$$\iff \mathbb{P}(Y = 1 \,|\, X = x) \geq \mathbb{P}(Y = 0 \,|\, X = x)$$

$$\iff 1 \in \arg \max_{y \in \{0,1\}} P(Y = y \,|\, X = x)$$

- Bayes classifier - minimizes L over all classifiers

$$g_*(x) = 1_{\eta(x) \geq 1/2} \text{ where } \eta(x) = \mathbb{P}(Y = 1 \,|\, X = x)$$

- Bayes risk - smallest probability of misclassification

$$L(g_*) = \mathbb{P}(g_*(X) \neq Y) = \mathbb{E}_{X \sim \mathcal{D}_X}[\min\{\eta(X), 1 - \eta(X)\}]$$

Generalization bound for 1-NN

Proof 2:

$$L(g_*) = \mathbb{E}_{(X,Y) \sim \mathscr{D}}[1_{g_*(X) \neq Y}]$$

$$= \mathbb{E}_{X \sim \mathscr{D}_X}[\mathbb{E}_{Y \sim \mathscr{D}_{Y|X}}[1_{g_*(X) \neq Y} | X]]$$

$$= \mathbb{E}_{X \sim \mathscr{D}_X}[\mathbb{E}_{Y \sim \mathscr{D}_{Y|X}}[1_{g_*(X) \neq Y} | X]1_{\eta(X) \geq 1/2} + E_{Y \sim \mathscr{D}_{Y|X}}[1_{g_*(X) \neq Y} | X]1_{\eta(X) > 1/2}]$$

$$= \mathbb{E}_{X \sim \mathscr{D}_X}[\mathbb{E}_{Y \sim \mathscr{D}_{Y|X}}[1_{1 \neq Y} | X]1_{\eta(X) \geq 1/2} + E_{Y \sim \mathscr{D}_{Y|X}}[1_{0 \neq Y} | X]1_{\eta(X) > 1/2}]$$

$$= \mathbb{E}_{X \sim \mathscr{D}_X}[\mathbb{P}(Y = 0 | X)1_{\eta(X) \geq 1/2} + \mathbb{P}(Y = 1 | X)1_{\eta(X) > 1/2}]$$

$$= \mathbb{E}_{X \sim \mathscr{D}_X}[\min\{\eta(X), 1 - \eta(X)\}]$$

- Bayes risk - smallest probability of misclassification

$$L(g_*) = \mathbb{P}(g_*(X) \neq Y) = \mathbb{E}_{X \sim \mathscr{D}_X}[\min\{\eta(X), 1 - \eta(X)\}]$$

Generalization bound for 1-NN

Assumption: $\exists c \geq 0, \, \forall x, x' \in \mathcal{X}$:

$$|\eta(x) - \eta(x')| \leq c\|x - x'\|_2$$

➡ Nearby points are likely to have the same label

Claim:

$$\mathbb{E}_{S_{train}}[L(g_{S_{train}})] \leq 2L(g_*) + c\mathbb{E}_{S_{train}, X \sim \mathcal{D}_X}[\|X - \text{nbh}_{S_{train}, 1}(x)\|]$$

$$\leq 2L(g_*) + 4c\sqrt{d}N^{-\frac{1}{d+1}}$$

geometric term:
average distance between
a random point and x

Interpretation:

Fixed d and $N \to \infty : \mathbb{E}_{S_{train}}[L(g_{S_{train}})] \leq 2L(g_*)$

Fixed N and $d \to \infty$: error increases exponentially fast

Interpolation method can generalize well: against common belief

Proof

We want to bound

$$\mathbb{E}_{S_{train}}[L(g_{S_{train}})] = \mathbb{E}_{S_{train}}[\mathbb{P}_{(X,Y)\sim\mathscr{D}}[g_{S_{train}}(X) \neq Y]]$$

We first sample n unlabeled examples $S_{train,X} = (X_1, \cdots X_n) \sim \mathscr{D}_X$, an

unlabeled example $X \sim \mathscr{D}$ and define $X' = \text{nbh}_{S_{train},1}(X)$

Finally we sample $Y \sim \eta(X)$ and $Y' \sim \eta(X')$

We have:

$$\mathbb{E}_{S_{train}}[L(g_{S_{train}})] = \mathbb{E}_{S_{X,train},X\sim\mathscr{D}_X,Y\sim\eta(X),Y'\sim\eta(X')}[1_{Y \neq g_{S_{train}(X)}}]$$

$$= \mathbb{E}_{S_{X,train},X\sim\mathscr{D}_X,Y\sim\eta(X),Y'\sim\eta(X')}[1_{Y \neq Y'}]$$

$$= \mathbb{E}_{S_{train,X}X\sim\mathscr{D}_X}[\mathbb{P}_{Y\sim\eta(X),Y'\sim\eta(X')}(Y \neq Y')]$$

Proof

Consider two points $x, x' \in [0,1]^d$.

Sample their labels $Y \sim \eta(x)$ and $Y' \sim \eta(x')$

Claim:

$$\mathbb{P}(Y' \neq Y) \leq 2 \min\{\eta(x), 1 - \eta(x)\} + c\|x - x'\|$$

• Simple case: $x = x'$

$$\mathbb{P}(Y' \neq Y) = \mathbb{E}[1_{Y' \neq Y} 1_{Y'=1} + 1_{Y' \neq Y} 1_{Y'=0}]$$
$$= \mathbb{P}(Y' = 1)\mathbb{P}(Y = 0) + \mathbb{P}(Y' = 1)\mathbb{P}(Y = 0)$$
$$= 2\eta(x)(1 - \eta(x))$$
$$\leq 2 \min\{\eta(x), 1 - \eta(x)\}$$

Case 1:	
Y=0	$(1 - \eta(x))$
Y'=1	$\eta(x)$
Case 2:	
Y=1	$\eta(x)$
Y'=0	$(1 - \eta(x))$

Proof

- General case:

$$\mathbb{P}(Y \neq Y') = \eta(x)(1 - \eta(x')) + \eta(x')(1 - \eta(x))$$

$$= \eta(x)(1 - \eta(x)) + \eta(x)(\eta(x) - \eta(x'))$$

$$+ \eta(x)(1 - \eta(x)) + (\eta(x') - \eta(x))(1 - \eta(x))$$

$$= 2\eta(x)(1 - \eta(x)) + (2\eta(x) - 1)(\eta(x) - \eta(x'))$$

$$\leq 2\eta(x)(1 - \eta(x)) + |(2\eta(x) - 1)| \, |\eta(x) - \eta(x)|$$

$$\leq 2\eta(x)(1 - \eta(x)) + |\eta(x) - \eta(x)|$$

$$\leq 2\eta(x)(1 - \eta(x)) + c\|x - x'\|$$

$$\leq 2\min\{\eta(x), 1 - \eta(x)\} + c\|x - x'\|$$

Proof

$$\mathbb{E}_{S_{train}}[L(g_{S_{train}})] = \mathbb{E}_{S_{X,train},X\sim\mathscr{D}_X,Y\sim\eta(X),Y'\sim\eta(X')}[1_{Y\neq g_{S_{train}(X)}}]$$

$$= \mathbb{E}_{S_{X,train},X\sim\mathscr{D}_X,Y\sim\eta(X),Y'\sim\eta(X')}[1_{Y\neq Y'}]$$

$$= \mathbb{E}_{S_{train,X}X\sim\mathscr{D}_X}[\mathbb{P}_{Y\sim\eta(X),Y'\sim\eta(X')}(Y\neq Y')]$$

$$\leq 2\mathbb{E}_{S_{train,X}X\sim\mathscr{D}_X}[2\min\{\eta(X),1-\eta(X)\}+c\|X-X'\|]$$

$$\leq 2L(g_*)+c\mathbb{E}_{S_{train}X\sim\mathscr{D}_X}[\|X-\mathsf{nbh}_{S_{train},1}(X)\|]$$
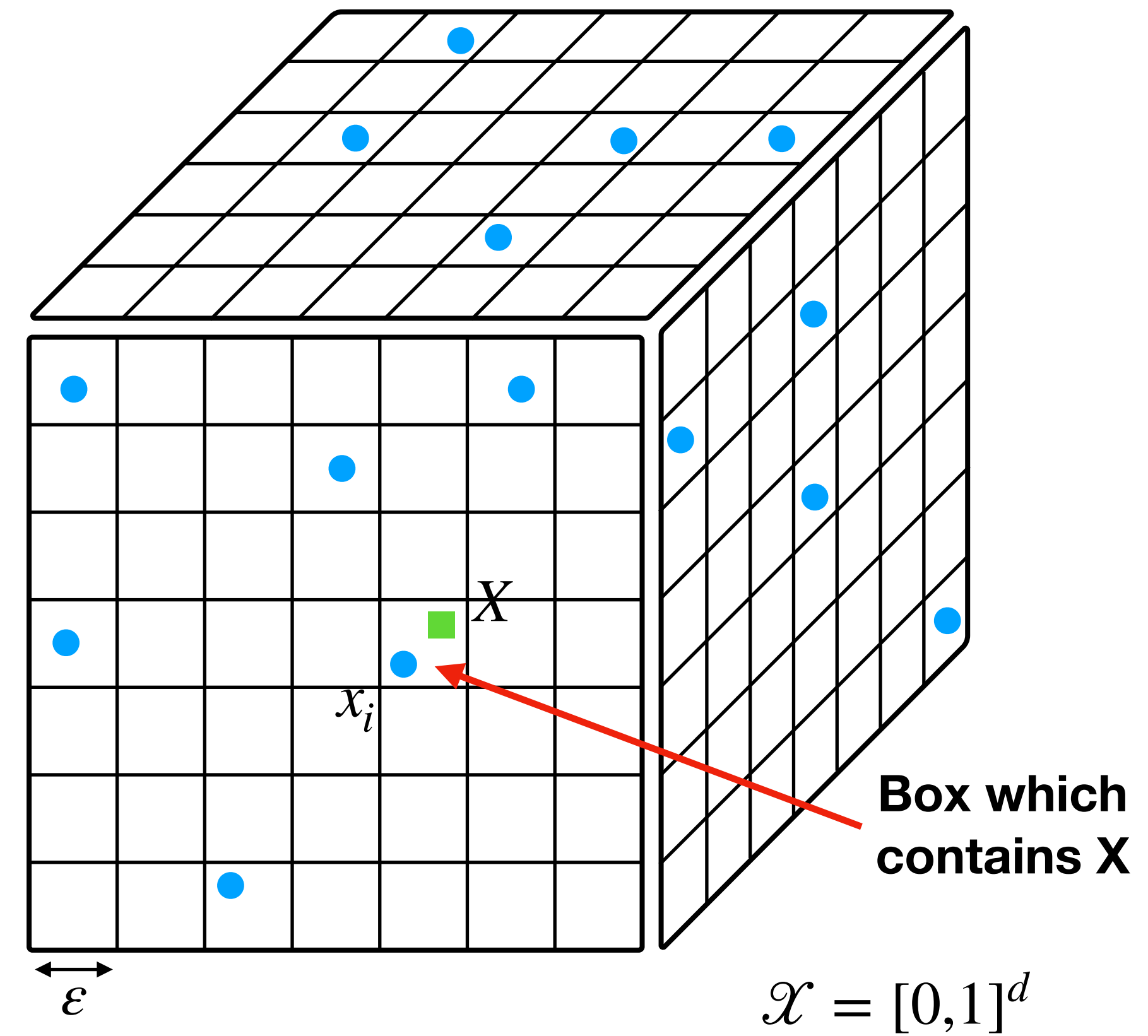
Bound on the geometric term

Consider a fresh sample $X \sim \mathcal{D}$ and denote by

$p_k = \mathbb{P}(X \in \text{Box}_k)$

Consider the box which contains X. Two options:

- The box contains an element of S_{train}. X has a neighbor in S_{train} at distance at most $\sqrt{d}\varepsilon$

It happens with probability $1 - (1 - p_k)^n$

X

x_i

ε

Box which contains X

$\mathcal{X} = [0,1]^d$

Bound on the geometric term

Consider a fresh sample $X \sim \mathscr{D}$ and denote by
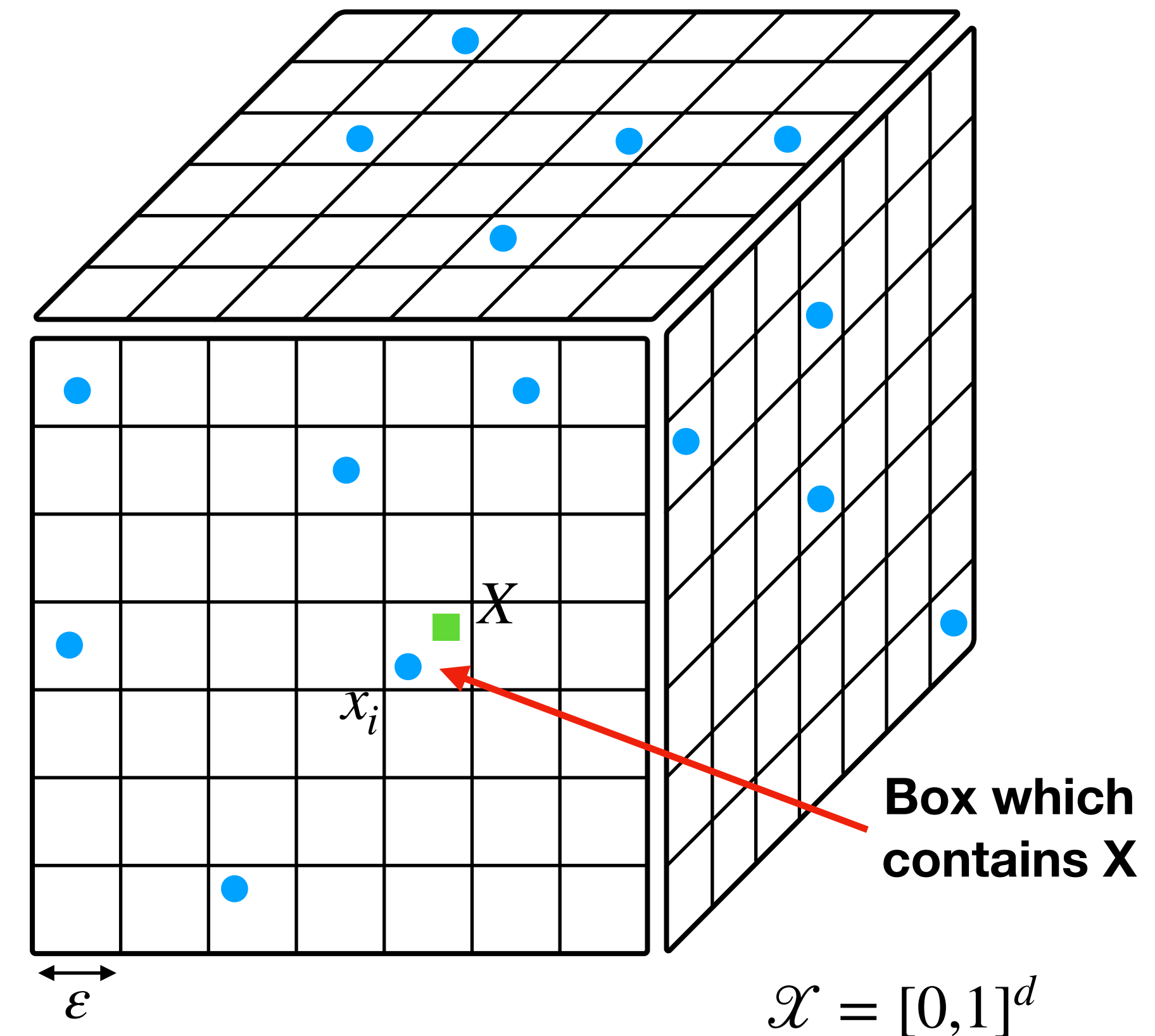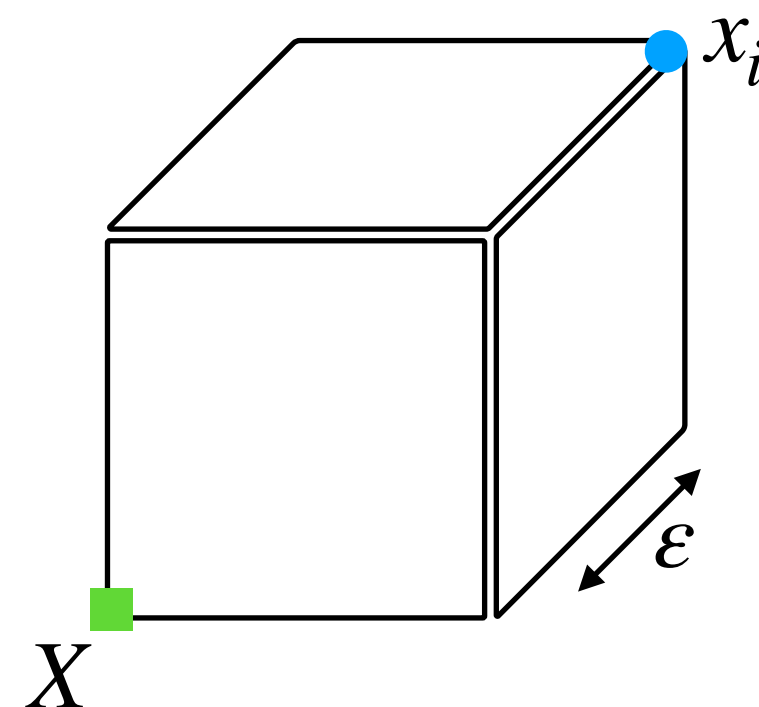
$p_k = \mathbb{P}(X \in \text{Box}_k)$

Consider the box which contains X. Two options:

- The box contains an element of S_{train}. X has a neighbor in S_{train} at distance at most $\sqrt{d}\varepsilon$

It happens with probability $1 - (1 - p_k)^n$

Proof: Consider the worst case:

$$\|X - x_i\| = \sqrt{\sum_{i=1}^{d} \varepsilon^2} = \sqrt{d}\varepsilon$$

x_i

X

X

x_i

Box which contains X

$\mathscr{X} = [0,1]^d$

Bound on the geometric term

Consider a fresh sample $X \sim \mathscr{D}$ and denote by
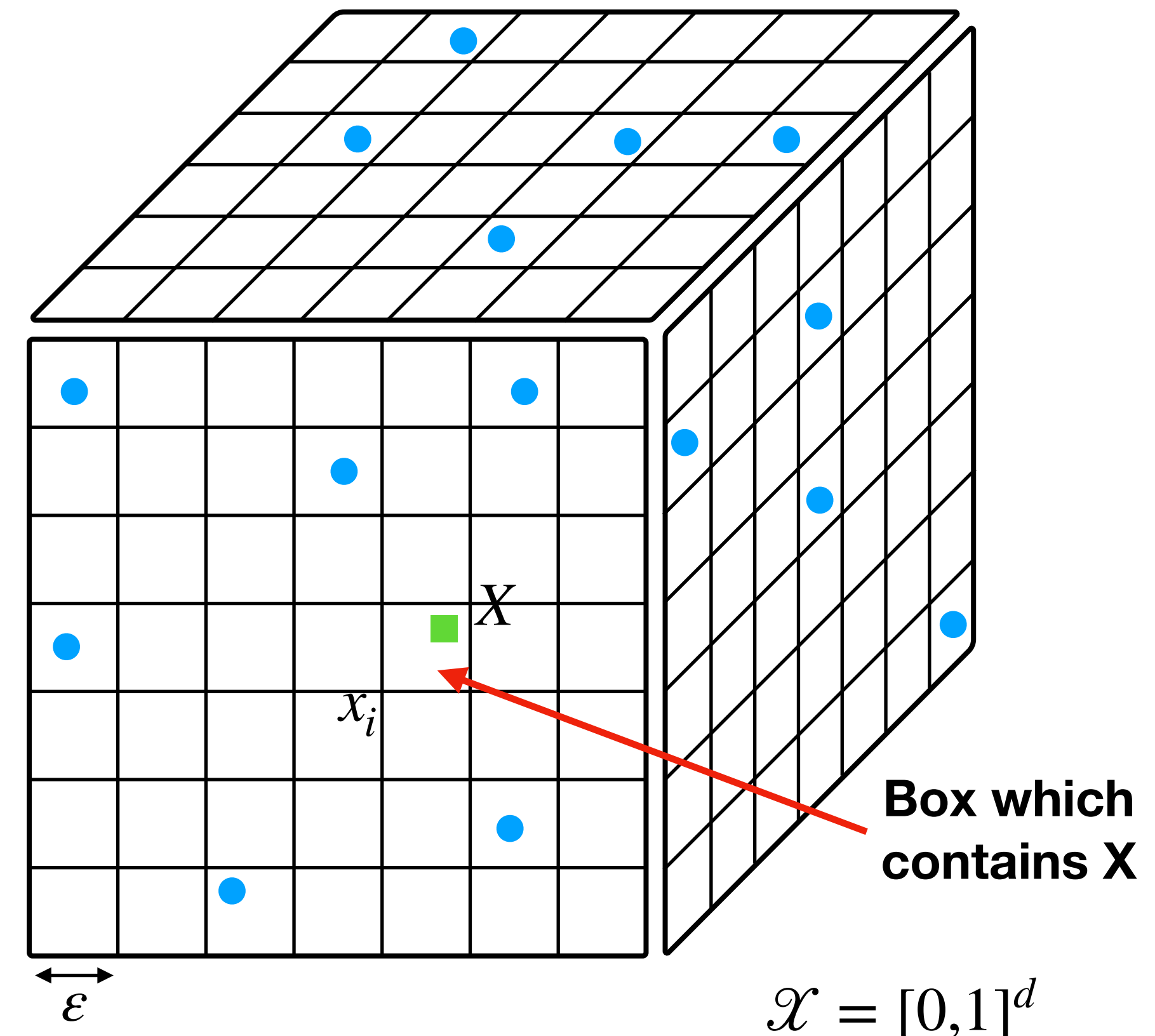
$p_k = \mathbb{P}(X \in \text{Box}_k)$

Consider the box which contains X. Two options:

- The box contains an element of S_{train}. X has a neighbor in S_{train} at distance at most $\sqrt{d}\varepsilon$

  It happens with probability $1 - (1 - p_k)^n$

- There is no element of S_{train}. The nearest neighbor of X can be at worst at a distance $\sqrt{d}$

  It happens with probability $(1 - p_k)^n$

X

x_i

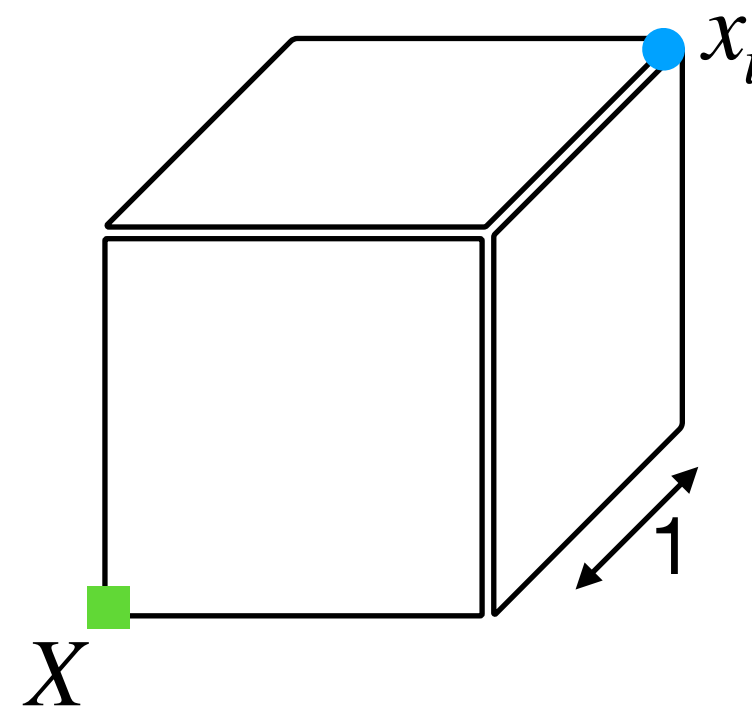**Box which contains X**

$\mathscr{X} = [0,1]^d$

Bound on the geometric term

Consider a fresh sample $X \sim \mathcal{D}$ and denote by

$p_k = \mathbb{P}(X \in \text{Box}_k)$
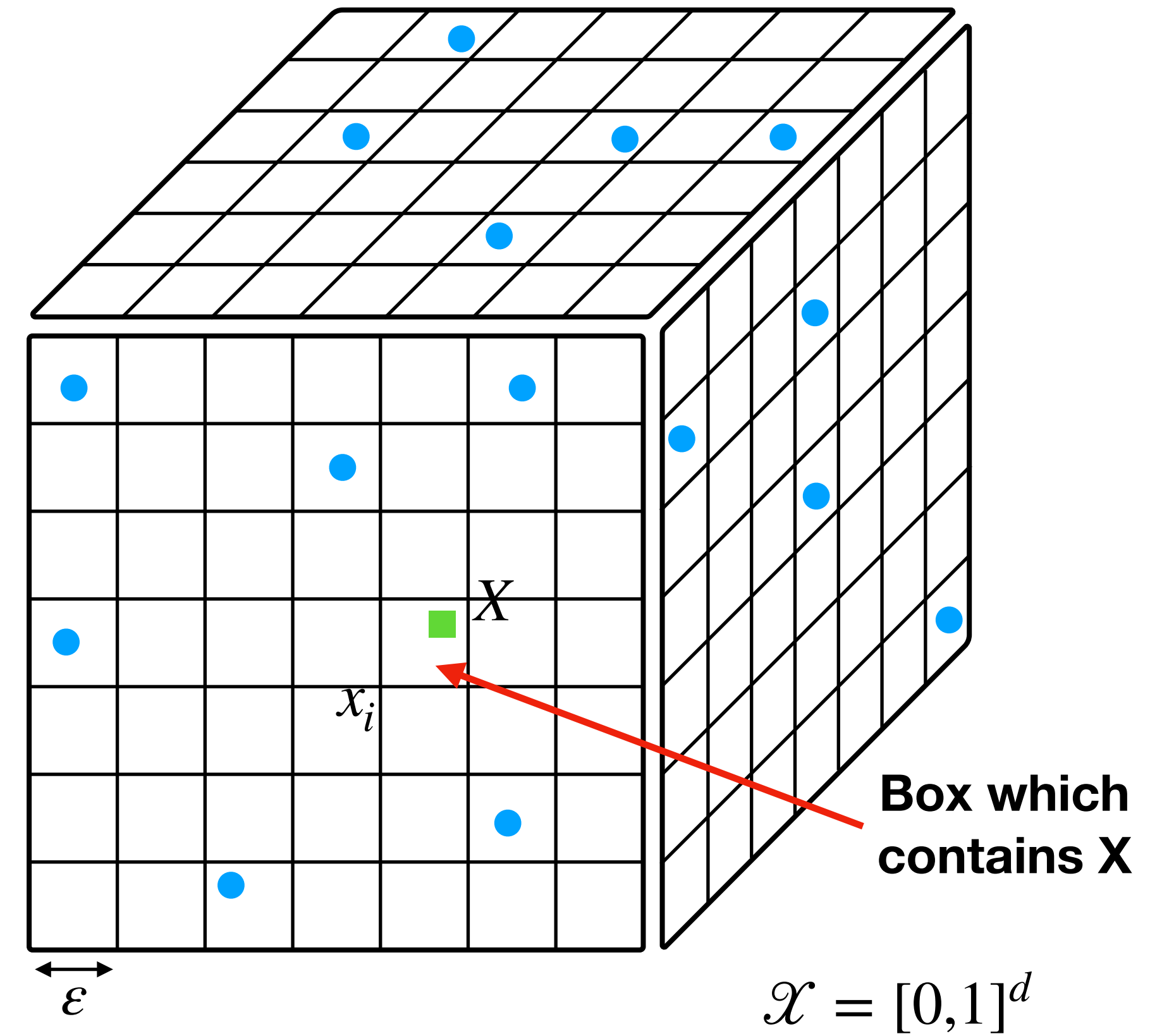
C

Proof: Consider the worst case:

$$\|X - x_i\| = \sqrt{\sum_{i=1}^{d} 1} = \sqrt{d}$$

x_i

1

X

of

X can be at worst at a distance $\sqrt{d}$

It happens with probability $(1 - p_k)^n$

ε

X

x_i

Box which contains X

$\mathcal{X} = [0,1]^d$

Bound on the geometric term

$$\mathbb{E}[X - \mathsf{nbh}(X)] = \sum_k p_k[(1-p_k)^n\sqrt{d} + (1-(1-p_k)^n)\sqrt{d}\varepsilon]$$

<u>Claim:</u> we get the bound by maximizing over p_k and ε

<u>Intuition:</u>

- If p_k is large: it is likely that I pick that box but it is also likely that I find a training point in that box

- If p_k is small, then we are fine since by definition this does not happen very often.

X

x_i

Box which contains X

ε

$\mathscr{X} = [0,1]^d$