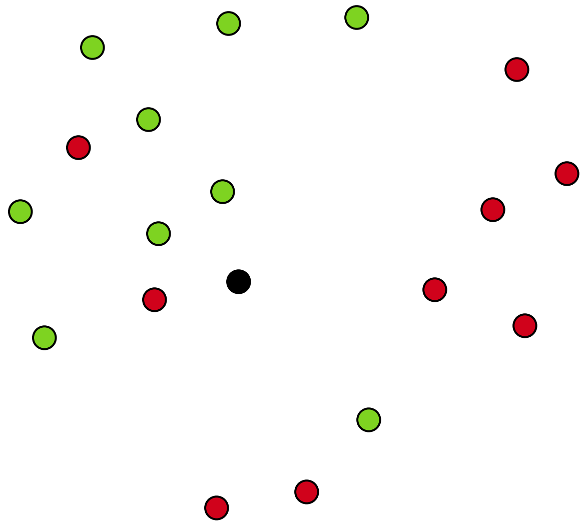
KNN

Machine Learning Techniques

Karthik Thiagarajan

KNN

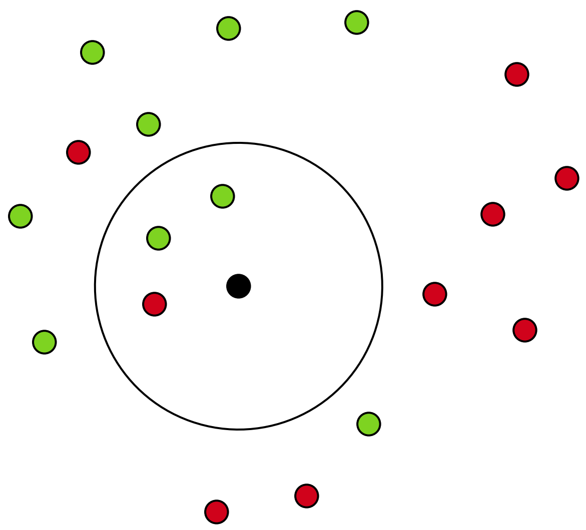
$$D = \{(\mathbf{x}_1, y_1),\ \cdots, (\mathbf{x}_n, y_n)\}$$



train (1)
train (0)
test

KNN

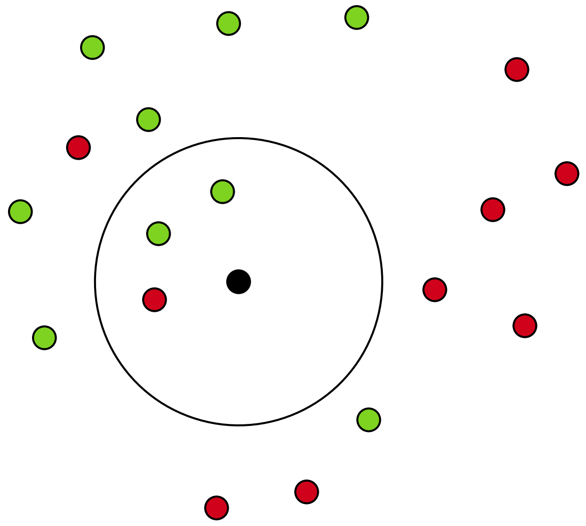
$$D = \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_n, y_n)\}$$



train (1)
train (0)
test

KNN

$$D = \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_n, y_n)\}$$

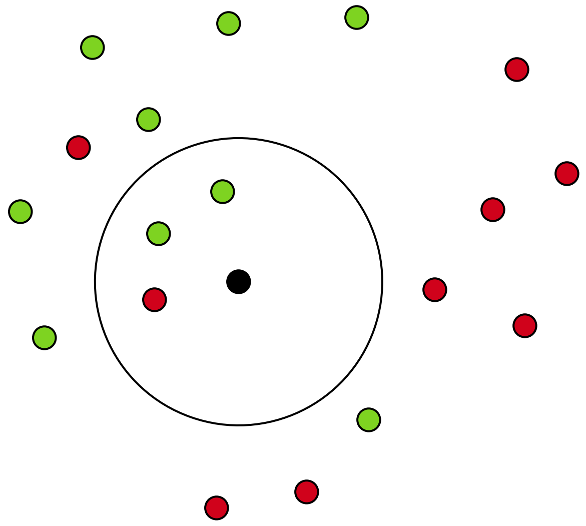


KNN$(D, d, k, \mathbf{x})$

train (1)
train (0)
test

KNN

$$D = \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_n, y_n)\}$$



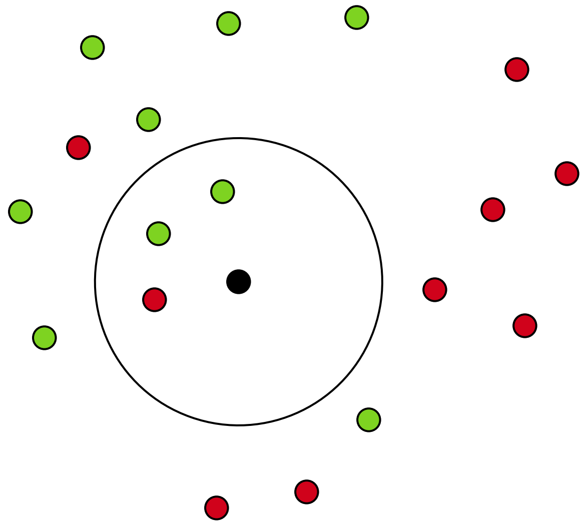
KNN$(D, d, k, \mathbf{x})$

- $L, N \leftarrow$ empty lists

train (1)
train (0)
test

KNN

$D = \{(\mathbf{x}_1, y_1),\ \cdots, (\mathbf{x}_n, y_n)\}$



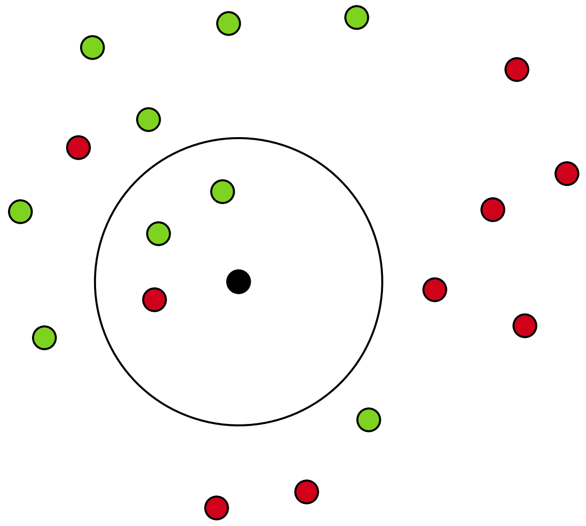
KNN$(D, d, k, \mathbf{x})$

- $L, N \leftarrow$ empty lists
- for $\mathbf{x}_i \in D$:
  - $L \leftarrow (d(\mathbf{x}_i, \mathbf{x}), y_i)$

train (1)
train (0)
test

KNN

$D = \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_n, y_n)\}$



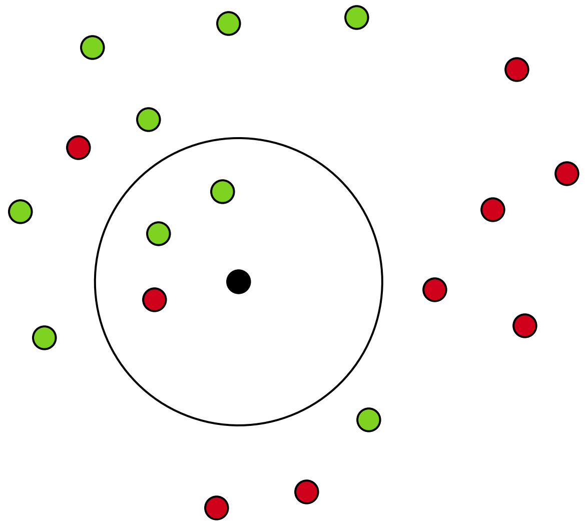
KNN$(D, d, k, \mathbf{x})$

- $L, N \leftarrow$ empty lists
- for $\mathbf{x}_i \in D$:
  - $L \leftarrow (d(\mathbf{x}_i, \mathbf{x}), y_i)$
- sort L in ascending order of distance

train (1)
train (0)
test

KNN

$D = \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_n, y_n)\}$



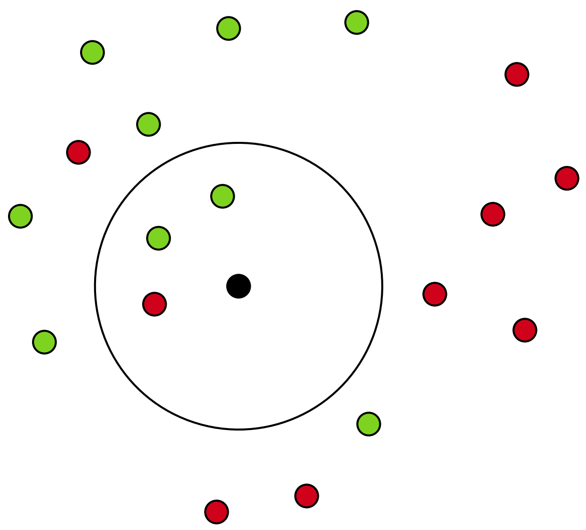
KNN$(D, d, k, \mathbf{x})$

- $L, N \leftarrow$ empty lists
- for $\mathbf{x}_i \in D$:
  - $L \leftarrow (d(\mathbf{x}_i, \mathbf{x}), y_i)$
- sort L in ascending order of distance
- $N \leftarrow$ labels of first k elements in L

train (1)
train (0)
test

KNN

$$D = \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_n, y_n)\}$$



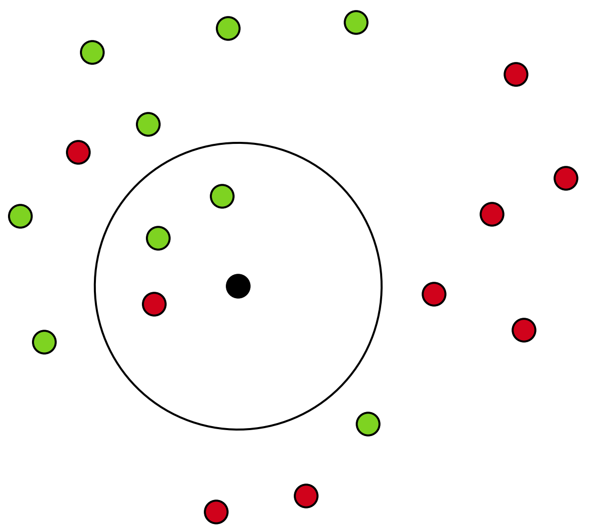
KNN$(D, d, k, \mathbf{x})$

- $L, N \leftarrow$ empty lists
- for $\mathbf{x}_i \in D$:
  - $L \leftarrow (d(\mathbf{x}_i, \mathbf{x}), y_i)$
- sort L in ascending order of distance
- $N \leftarrow$ labels of first k elements in L
- $p_{Y|X} \leftarrow$ PMF(N)

train (1)
train (0)
test

KNN

$D = \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_n, y_n)\}$



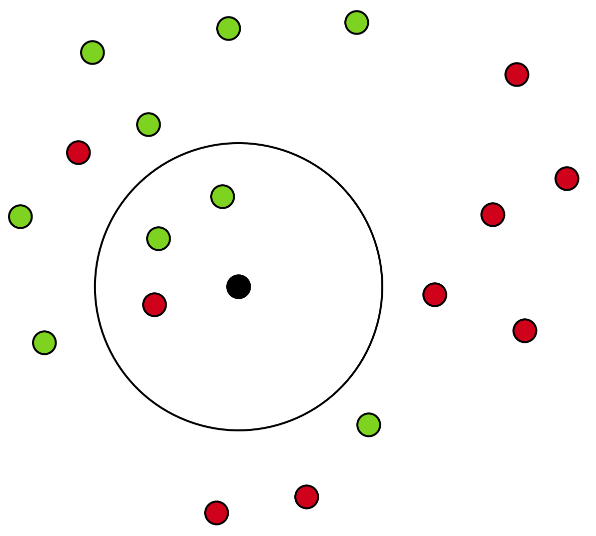
KNN$(D, d, k, \mathbf{x})$

- $L, N \leftarrow$ empty lists
- for $\mathbf{x}_i \in D$:
  - $L \leftarrow (d(\mathbf{x}_i, \mathbf{x}), y_i)$
- sort L in ascending order of distance
- $N \leftarrow$ labels of first k elements in L
- $p_{Y|X} \leftarrow$ PMF(N)
- return mode$(p_{Y|X})$

train (1)
train (0)
test

KNN

$D = \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_n, y_n)\}$



KNN$(D, d, k, \mathbf{x})$

- $L, N \leftarrow$ empty lists
- for $\mathbf{x}_i \in D$:
  - $L \leftarrow (d(\mathbf{x}_i, \mathbf{x}), y_i)$
- sort L in ascending order of distance
- $N \leftarrow$ labels of first k elements in L
- $p_{Y|X} \leftarrow$ PMF(N)
- return mode$(p_{Y|X})$

$Y \in \{0, 1\}$

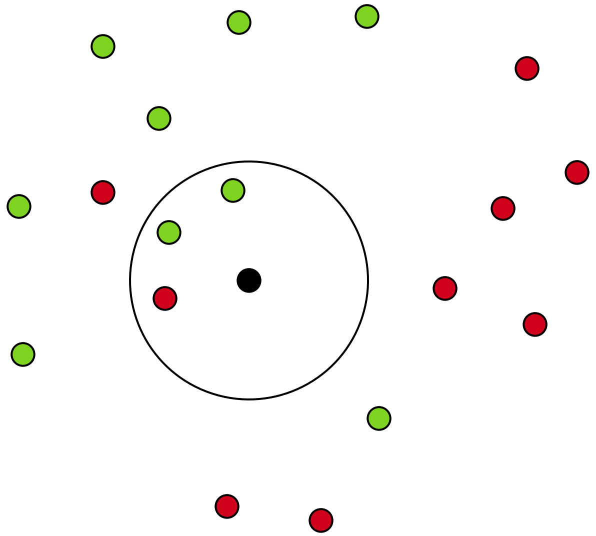
$$p_{Y|X}(y \mid \mathbf{x}) = \begin{cases} 1/3, & y = 0 \\ 2/3, & y = 1 \end{cases}$$

mode$(p_{Y|X}) = 1$

train (1)
train (0)
test

KNN

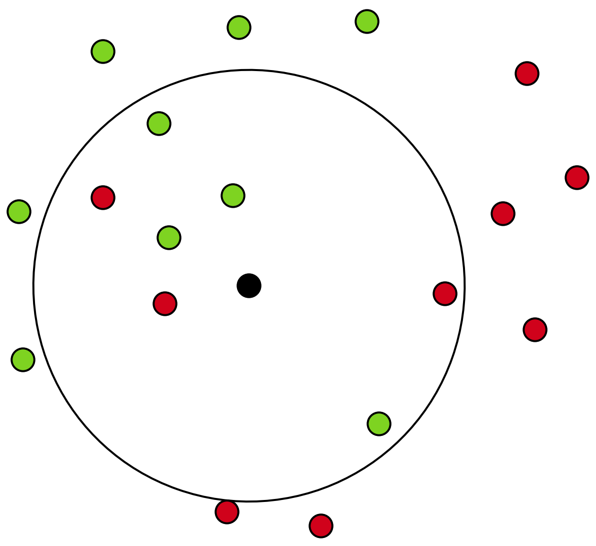
$$D = \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_n, y_n)\}$$



train (1)
train (0)
test

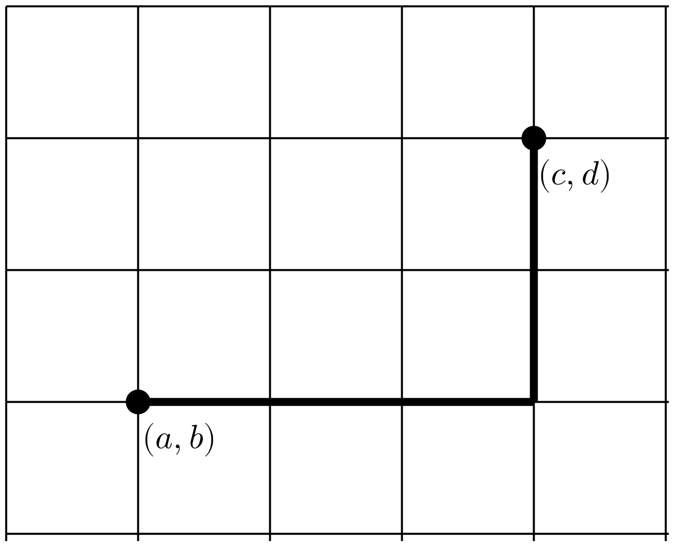
KNN

$$D = \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_n, y_n)\}$$

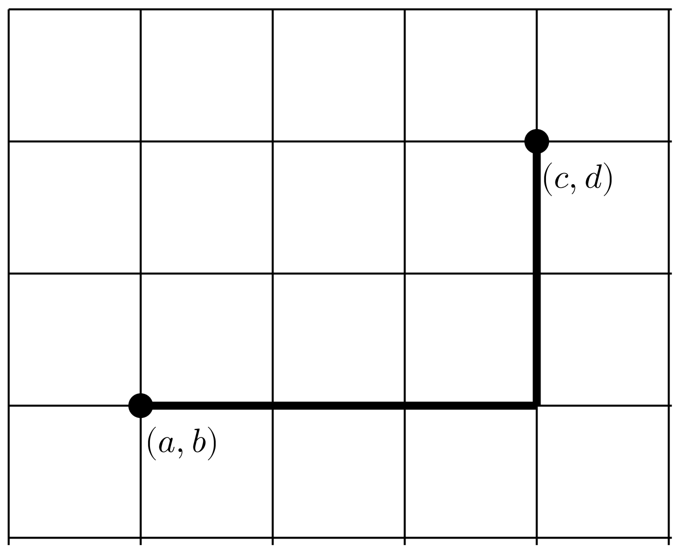


train (1)
train (0)
test

Manhattan Distance (Taxicab distance)

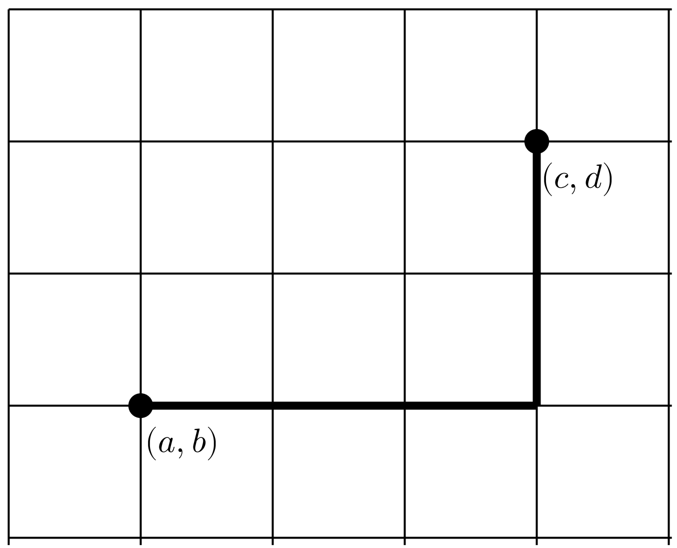


Manhattan Distance (Taxicab distance)

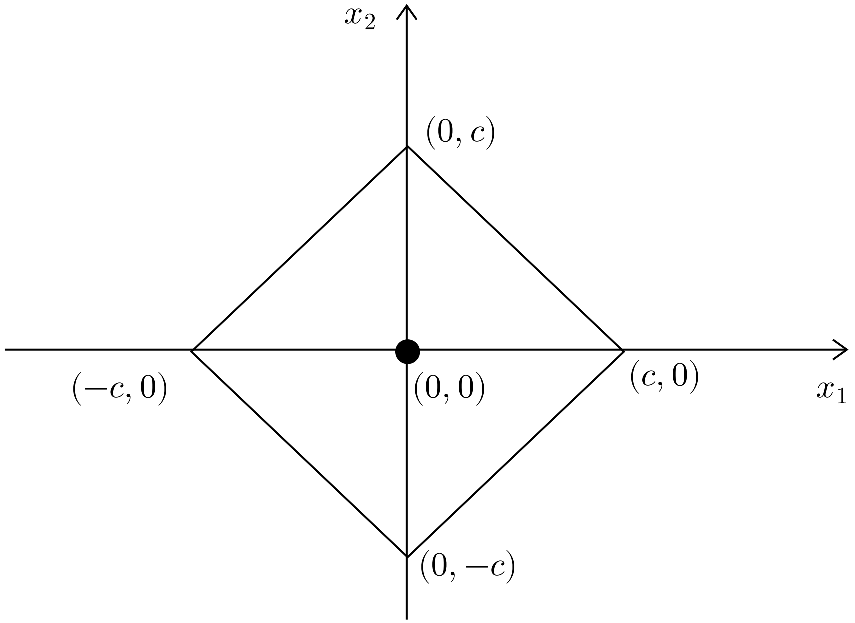


$$d((a,b),(c,d)) = |a - c| + |b - d|$$

Manhattan Distance (Taxicab distance)

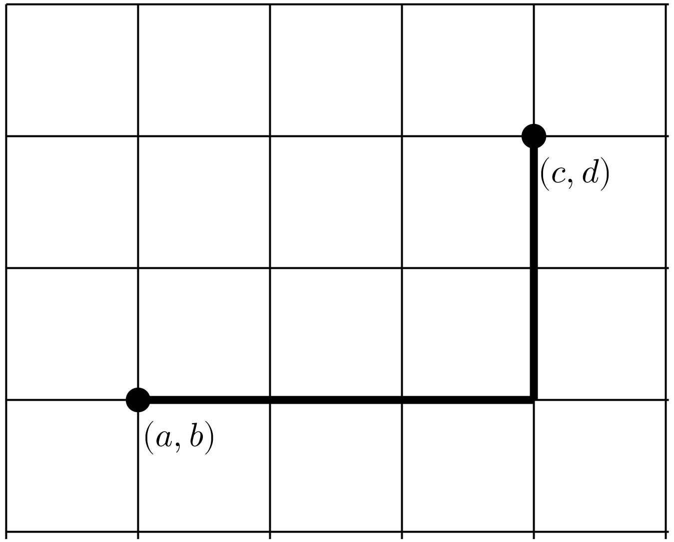


$$d((a,b),(c,d)) = |a-c| + |b-d|$$

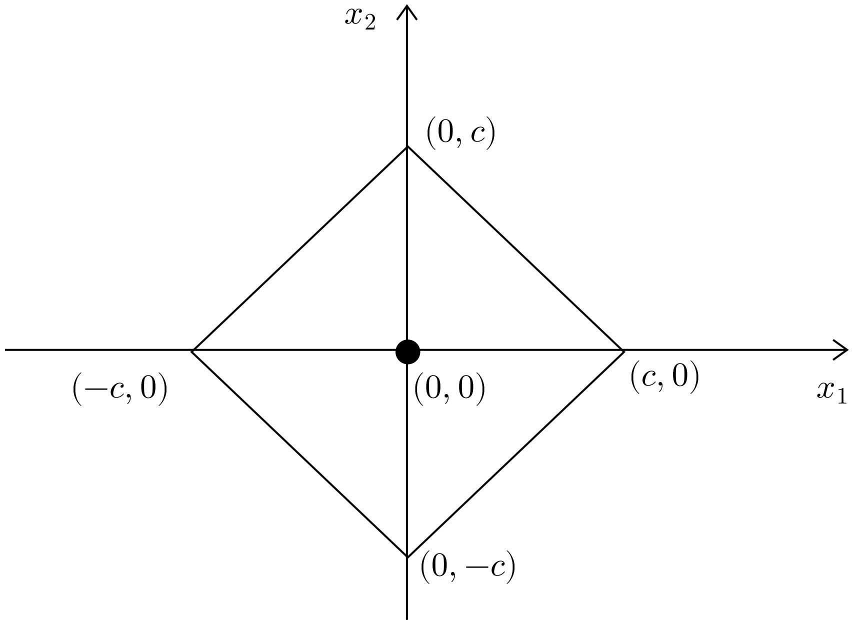


$$|x_1| + |x_2| = c$$

Manhattan Distance (Taxicab distance)



$$d((a, b), (c, d)) = |a - c| + |b - d|$$

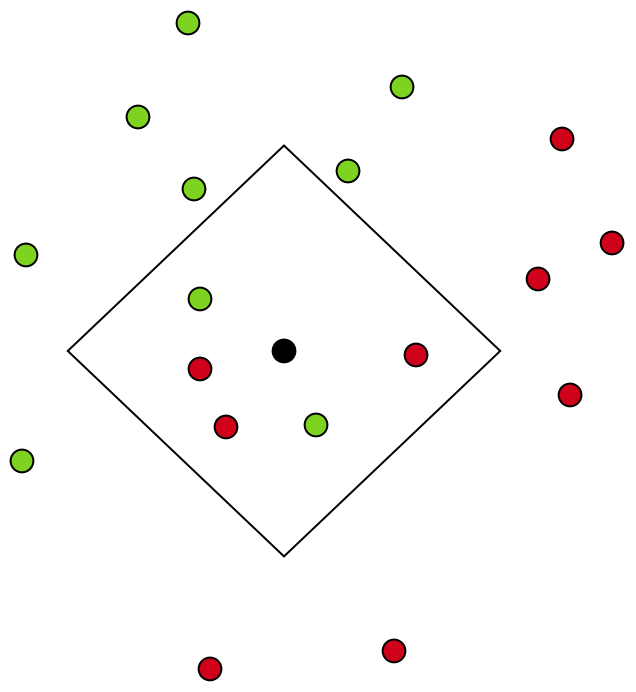


$$|x_1| + |x_2| = c$$

- $x_1 + x_2 = c$
- $x_1 - x_2 = c$
- $-x_1 + x_2 = c$
- $-x_1 - x_2 = c$

KNN

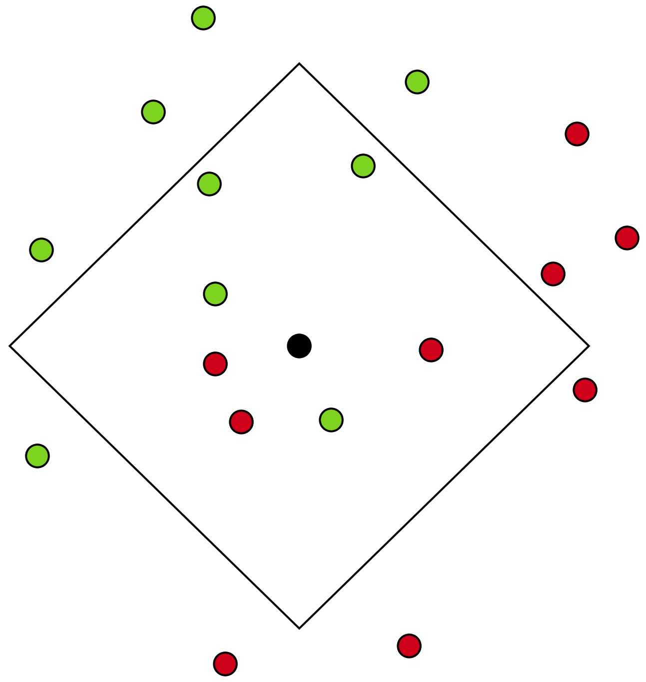
$$D = \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_n, y_n)\}$$



train (1)
train (0)
test

KNN

$$D = \{(\mathbf{x}_1, y_1),\ \cdots, (\mathbf{x}_n, y_n)\}$$



train (1)
train (0)
test

p-norms (Minkowski Distance)

$$||\mathbf{x}||_p = \left(|x_1|^p + \cdots + |x_n|^p\right)^{1/p}$$

p-norms (Minkowski Distance)

$$||\mathbf{x}||_p = \left(|x_1|^p + \cdots + |x_n|^p\right)^{1/p}$$

L_1 norm: $$||\mathbf{x}||_1 = |x_1| + \cdots + |x_n|$$

p-norms (Minkowski Distance)

$$||\mathbf{x}||_p = \left(|x_1|^p + \cdots + |x_n|^p\right)^{1/p}$$

L_1 norm: $$||\mathbf{x}||_1 = |x_1| + \cdots + |x_n|$$

L_2 norm: $$||\mathbf{x}||_2 = \sqrt{{x_1}^2 + \cdots + {x_n}^2}$$

p-norms (Minkowski Distance)

$$||\mathbf{x}||_p = \left(|x_1|^p + \cdots + |x_n|^p\right)^{1/p}$$

L_1 norm: $$||\mathbf{x}||_1 = |x_1| + \cdots + |x_n|$$

L_2 norm: $$||\mathbf{x}||_2 = \sqrt{{x_1}^2 + \cdots + {x_n}^2}$$

L_∞ norm: $$||\mathbf{x}||_\infty = \max\{|x_1|, \cdots, |x_n|\}$$

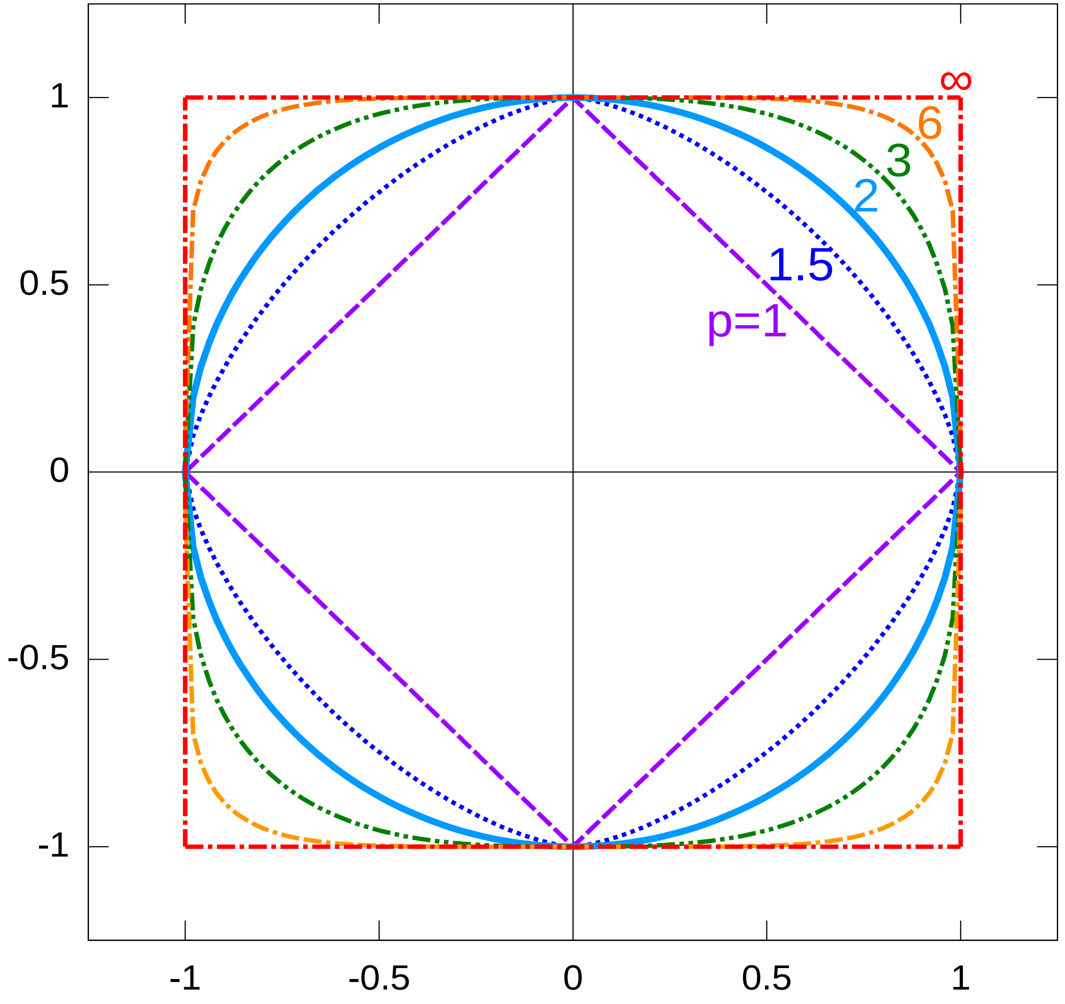
p-norms (Minkowski Distance)

$$||\mathbf{x}||_p = \left(|x_1|^p + \cdots + |x_n|^p\right)^{1/p}$$

L_1 norm: $$||\mathbf{x}||_1 = |x_1| + \cdots + |x_n|$$

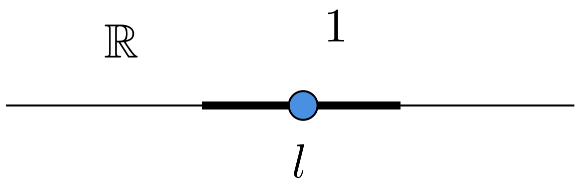
L_2 norm: $$||\mathbf{x}||_2 = \sqrt{{x_1}^2 + \cdots + {x_n}^2}$$

L_∞ norm: $$||\mathbf{x}||_\infty = \max\{|x_1|, \cdots, |x_n|\}$$

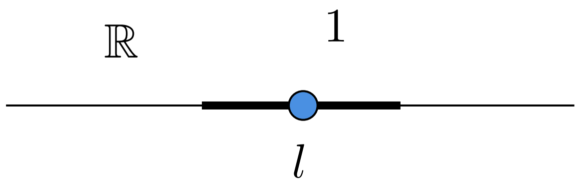


Source

Curse of Dimensionality

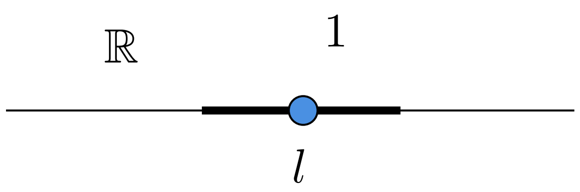


Curse of Dimensionality

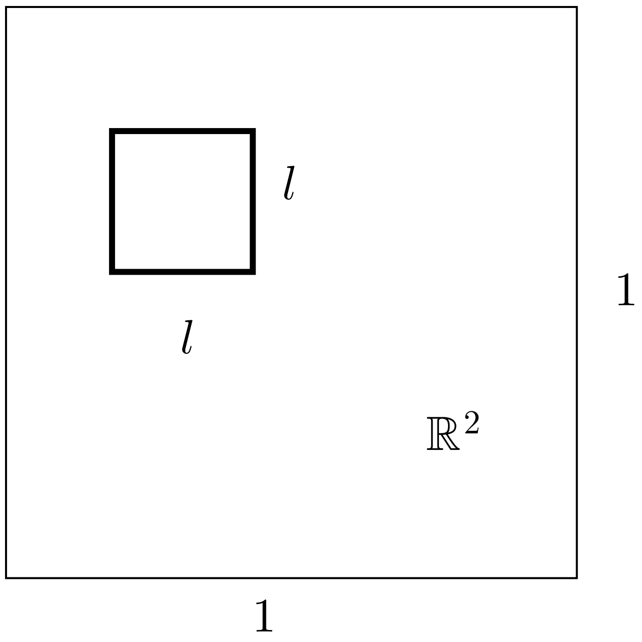


$$ln = k \implies l = \frac{k}{n}$$

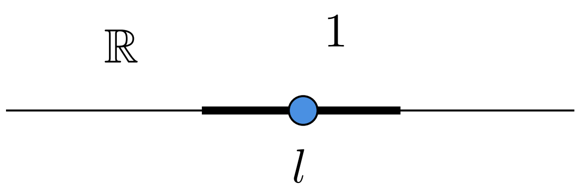
Curse of Dimensionality



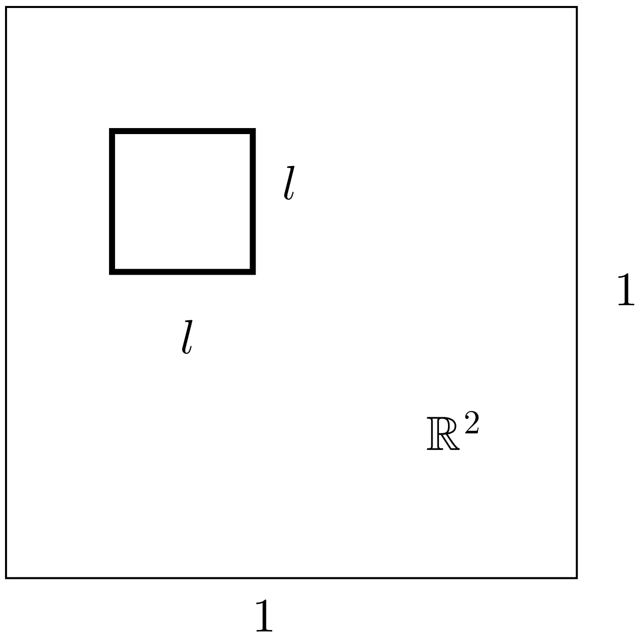
$$ln = k \implies l = \frac{k}{n}$$



Curse of Dimensionality

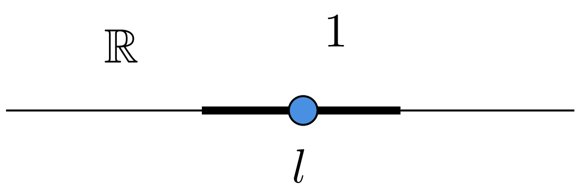


$$ln = k \implies l = \frac{k}{n}$$

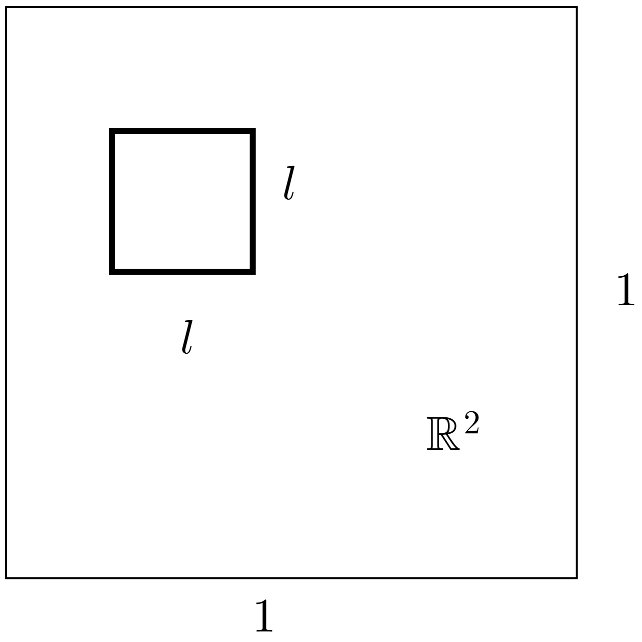


$$l^2 n = k \implies l = \sqrt{\frac{k}{n}}$$

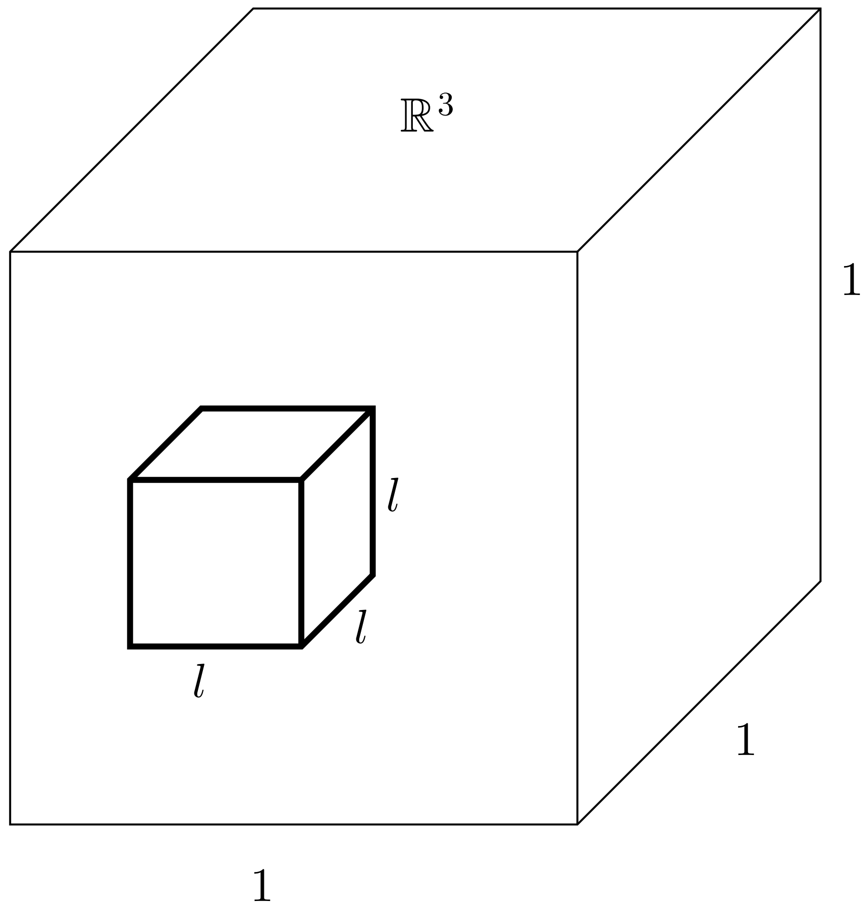
Curse of Dimensionality



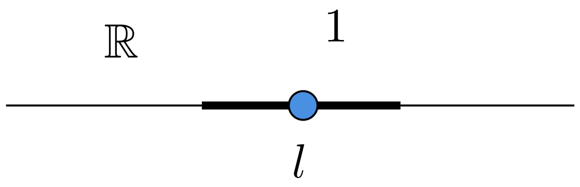
$$ln = k \implies l = \frac{k}{n}$$



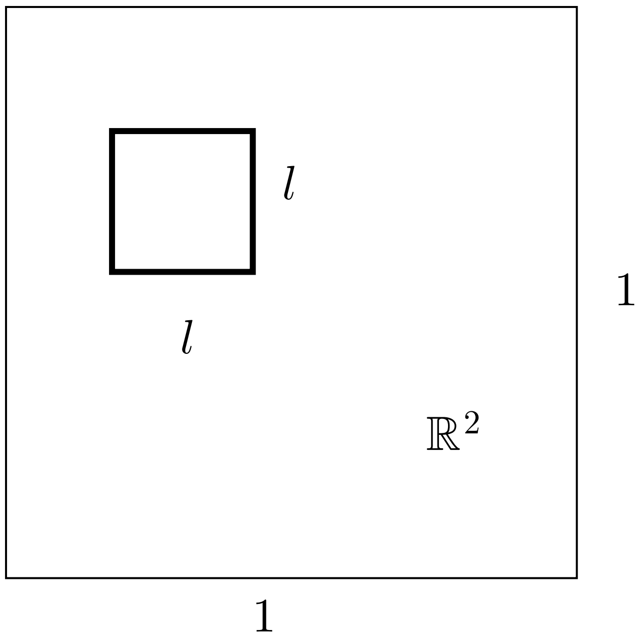
$$l^2 n = k \implies l = \sqrt{\frac{k}{n}}$$



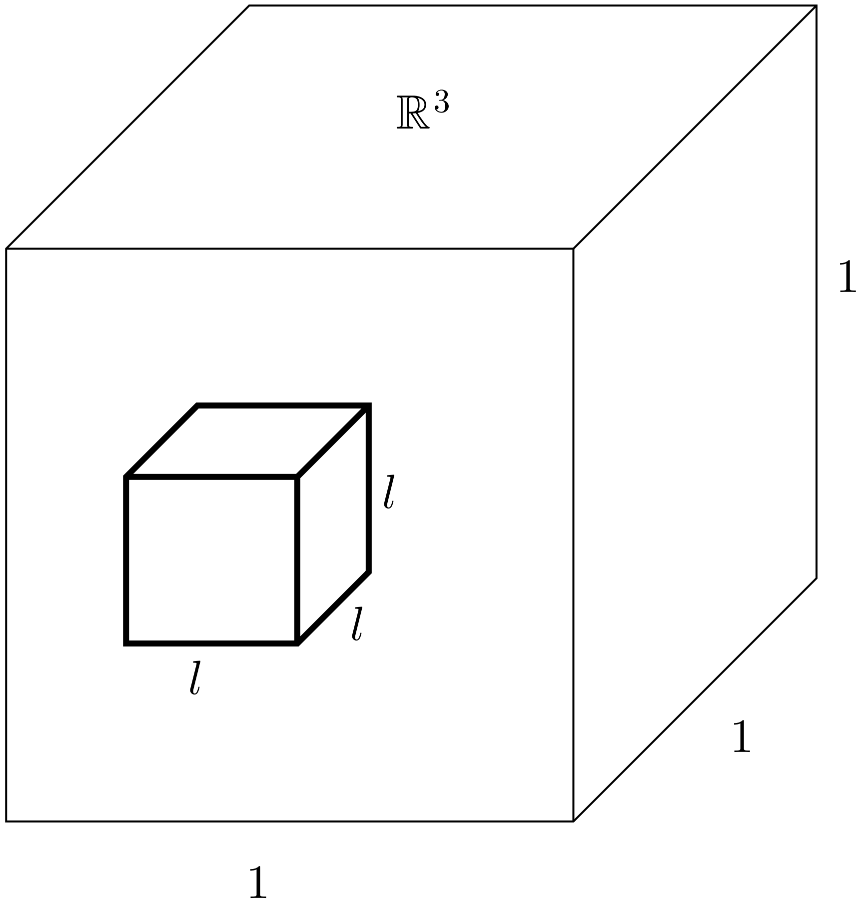
Curse of Dimensionality



$$ln = k \implies l = \frac{k}{n}$$

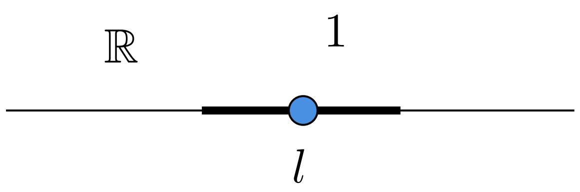


$$l^2 n = k \implies l = \sqrt{\frac{k}{n}}$$

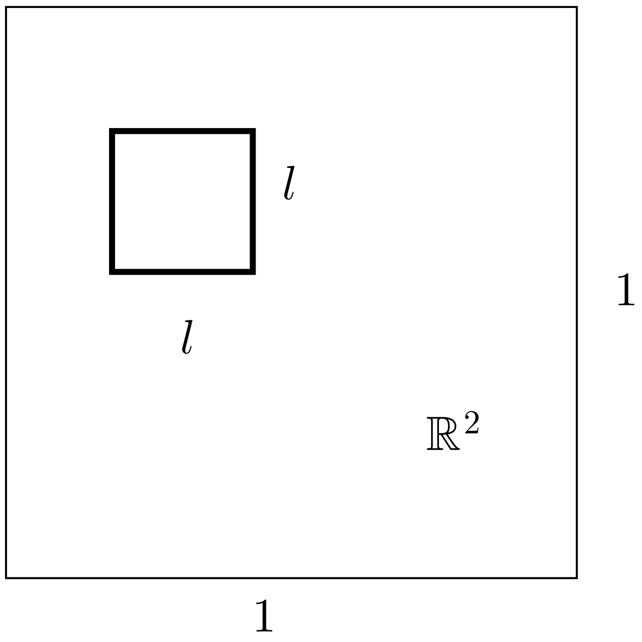


$$l^3 n = k \implies l = \sqrt[3]{\frac{k}{n}}$$

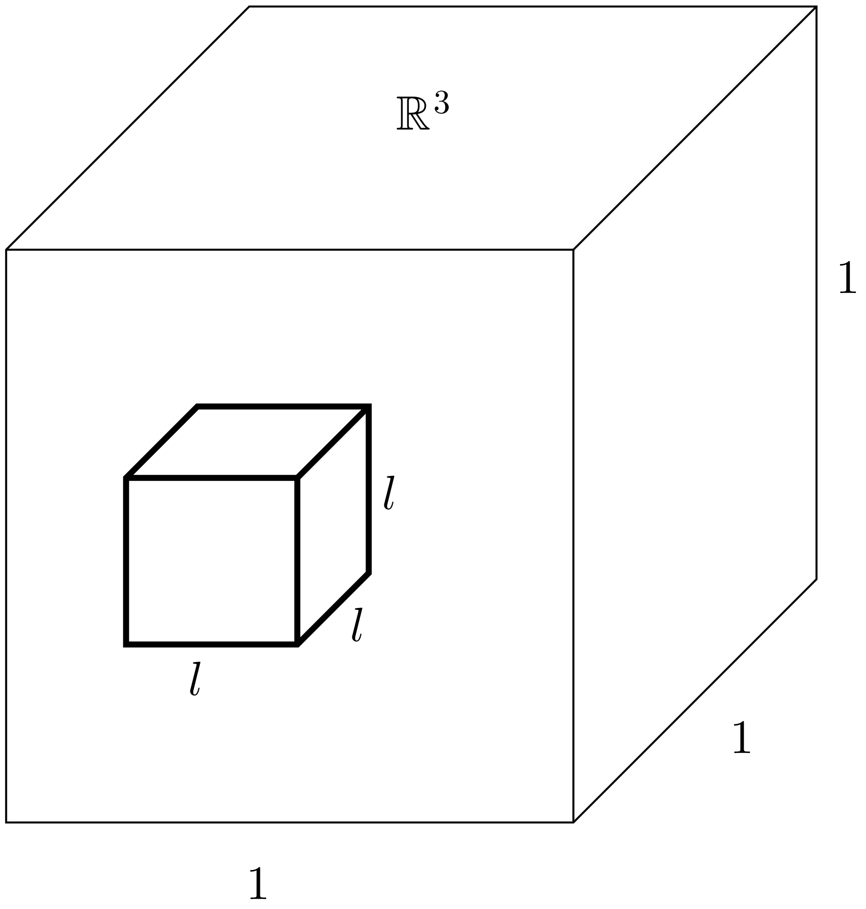
Curse of Dimensionality



$$ln = k \Longrightarrow l = \frac{k}{n}$$



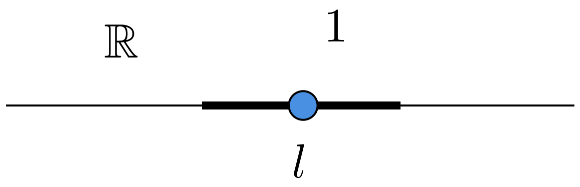
$$l^2 n = k \Longrightarrow l = \sqrt{\frac{k}{n}}$$



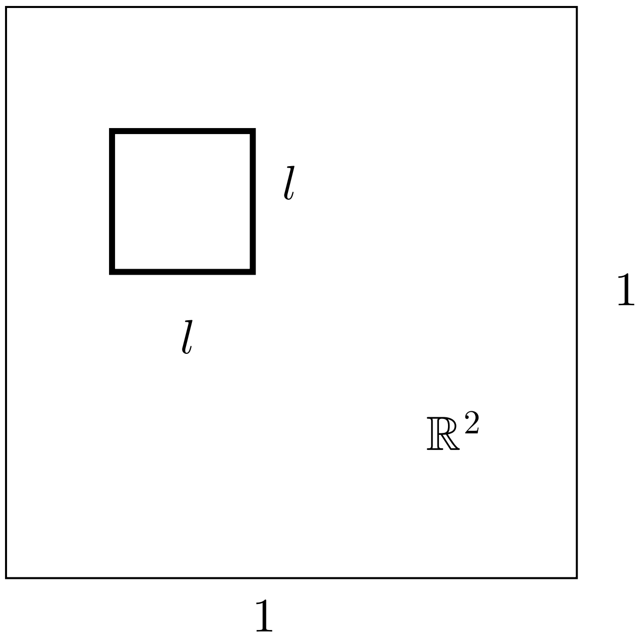
$\mathbb{R}^d$

$$l^3 n = k \Longrightarrow l = \sqrt[3]{\frac{k}{n}}$$

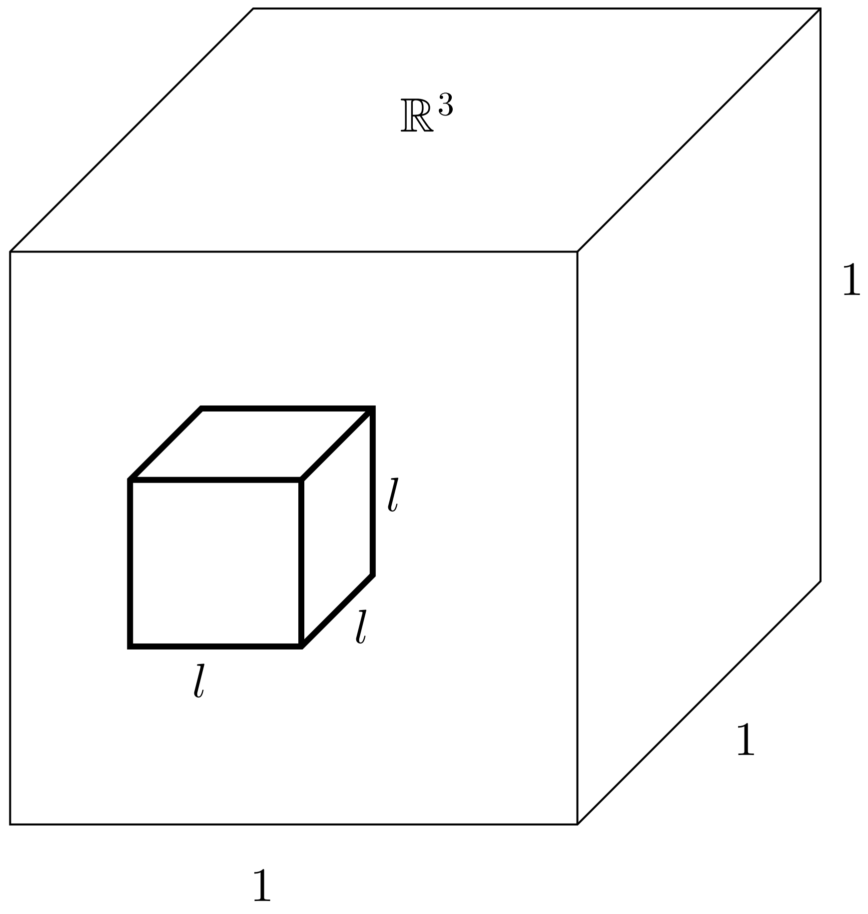
Curse of Dimensionality



$$ln = k \Longrightarrow l = \frac{k}{n}$$



$$l^2 n = k \Longrightarrow l = \sqrt{\frac{k}{n}}$$



$$l^3 n = k \Longrightarrow l = \sqrt[3]{\frac{k}{n}}$$

$\mathbb{R}^d$

$$l^d n = k \Longrightarrow l = \sqrt[d]{\frac{k}{n}}$$

Curse of Dimensionality

$$l = \sqrt[d]{\frac{k}{n}}$$

$$n = 1000$$
$$k = 10$$

Curse of Dimensionality

$$l = \sqrt[d]{\frac{k}{n}}$$

$$n = 1000$$
$$k = 10$$

$$l = \sqrt[d]{\frac{10}{1000}}$$

$$= (0.01)^{1/d}$$

Curse of Dimensionality

$$l = \sqrt[d]{\frac{k}{n}}$$

$$n = 1000$$
$$k = 10$$

$$l = \sqrt[d]{\frac{10}{1000}}$$

$$= (0.01)^{1/d}$$

d	l
1	
2	
3	
5	
10	
100	
1000	

Curse of Dimensionality

$$l = \sqrt[d]{\frac{k}{n}}$$

$n = 1000$
$k = 10$

$$l = \sqrt[d]{\frac{10}{1000}}$$

$$= (0.01)^{1/d}$$

d	l
1	0.01
2	0.10
3	0.22
5	0.40
10	0.63
100	0.95
1000	0.99

Curse of Dimensionality

$$l = \sqrt[d]{\frac{k}{n}}$$

$$n = 1000$$
$$k = 10$$

$$l = \sqrt[d]{\frac{10}{1000}}$$

$$= (0.01)^{1/d}$$

d	l
1	0.01
2	0.10
3	0.22
5	0.40
10	0.63
100	0.95
1000	0.99

$$n = \frac{k}{l^d}$$

$$k = 10$$
$$l = 0.1$$

$$n = \frac{10}{0.1^d}$$
$$= 10^{d+1}$$

Curse of Dimensionality

$$l = \sqrt[d]{\frac{k}{n}}$$

$n = 1000$
$k = 10$

$$l = \sqrt[d]{\frac{10}{1000}}$$

$$= (0.01)^{1/d}$$

d	l
1	0.01
2	0.10
3	0.22
5	0.40
10	0.63
100	0.95
1000	0.99

$$n = \frac{k}{l^d}$$

$k = 10$
$l = 0.1$

$$n = \frac{10}{0.1^d}$$

$$= 10^{d+1}$$

d	n
1	
2	
3	
5	
10	
100	

Curse of Dimensionality

$$l = \sqrt[d]{\frac{k}{n}}$$

$n = 1000$
$k = 10$

$$l = \sqrt[d]{\frac{10}{1000}}$$

$$= (0.01)^{1/d}$$

d	l
1	0.01
2	0.10
3	0.22
5	0.40
10	0.63
100	0.95
1000	0.99

$$n = \frac{k}{l^d}$$

$k = 10$
$l = 0.1$

$$n = \frac{10}{0.1^d}$$

$$= 10^{d+1}$$

d	n
1	10^2
2	10^3
3	10^4
5	10^5
10	10^{11}
100	10^{101}

Curse of Dimensionality

$$l = \sqrt[d]{\frac{k}{n}}$$

$n = 1000$
$k = 10$

$$l = \sqrt[d]{\frac{10}{1000}}$$

$$= (0.01)^{1/d}$$

d	l
1	0.01
2	0.10
3	0.22
5	0.40
10	0.63
100	0.95
1000	0.99

$$n = \frac{k}{l^d}$$

$k = 10$
$l = 0.1$

$$n = \frac{10}{0.1^d}$$

$$= 10^{d+1}$$

d	n
1	10^2
2	10^3
3	10^4
5	10^5
10	10^{11}
100	10^{101}

Assumptions

- Points that are "close" to me are "similar" to me
- I am not lonely

Curse of Dimensionality

$$l = \sqrt[d]{\frac{k}{n}}$$

$$n = 1000$$
$$k = 10$$

$$l = \sqrt[d]{\frac{10}{1000}}$$

$$= (0.01)^{1/d}$$

d	l
1	0.01
2	0.10
3	0.22
5	0.40
10	0.63
100	0.95
1000	0.99

$$n = \frac{k}{l^d}$$

$$k = 10$$
$$l = 0.1$$

$$n = \frac{10}{0.1^d}$$
$$= 10^{d+1}$$

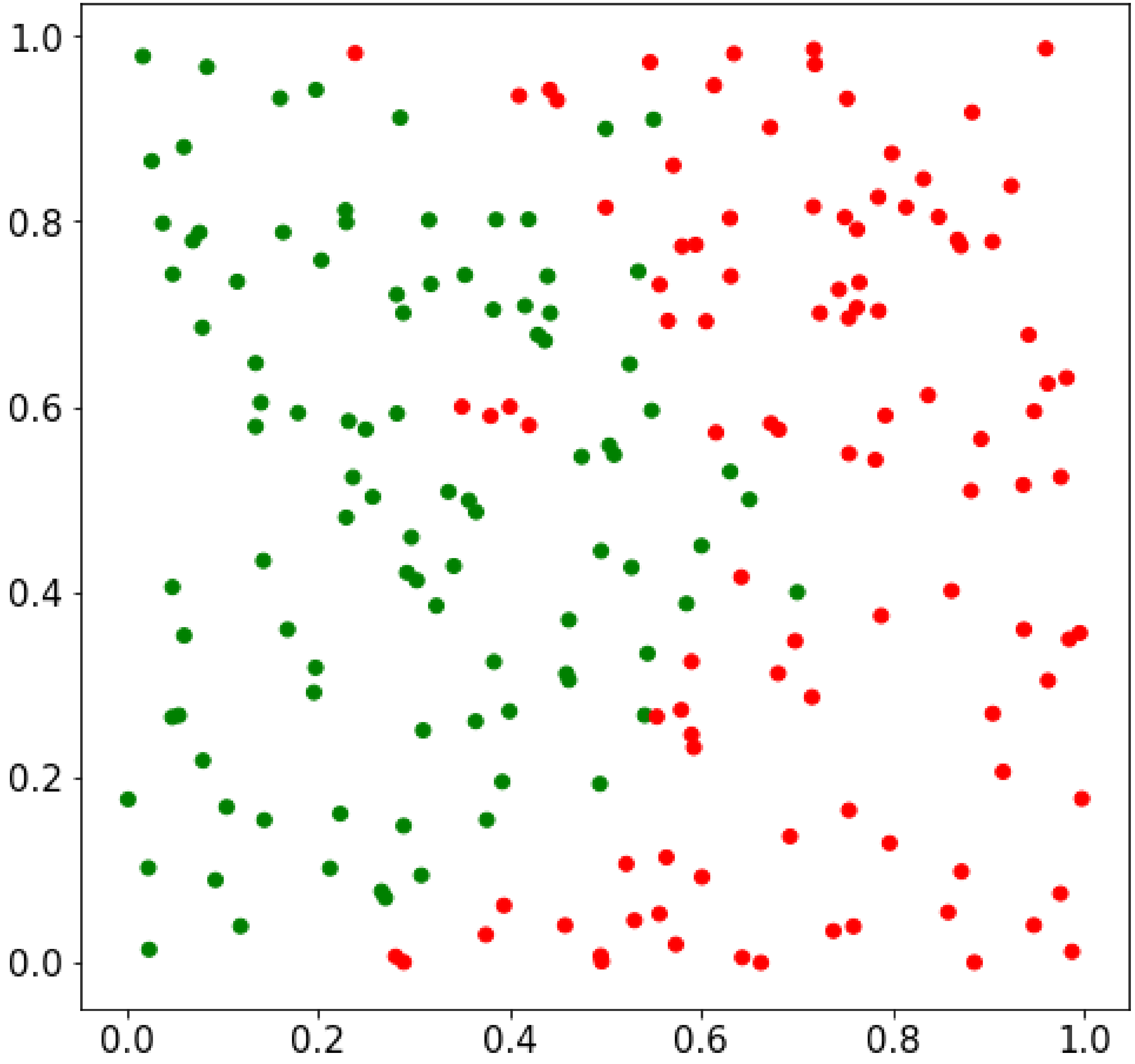
d	n
1	10^2
2	10^3
3	10^4
5	10^5
10	10^{11}
100	10^{101}

Assumptions

- Points that are "close" to me are "similar" to me
- I am not lonely

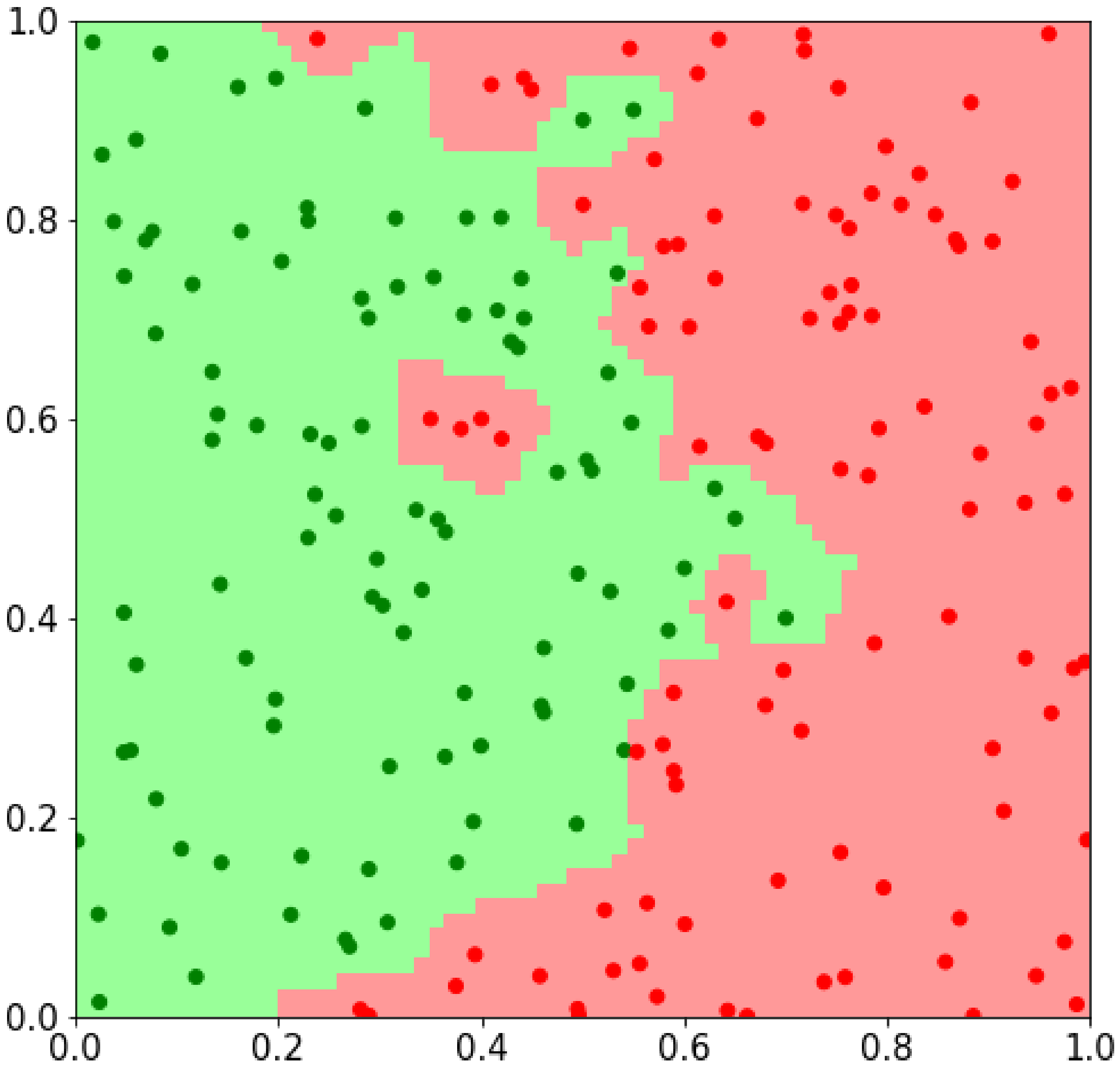
Sparsity

Decision Boundary



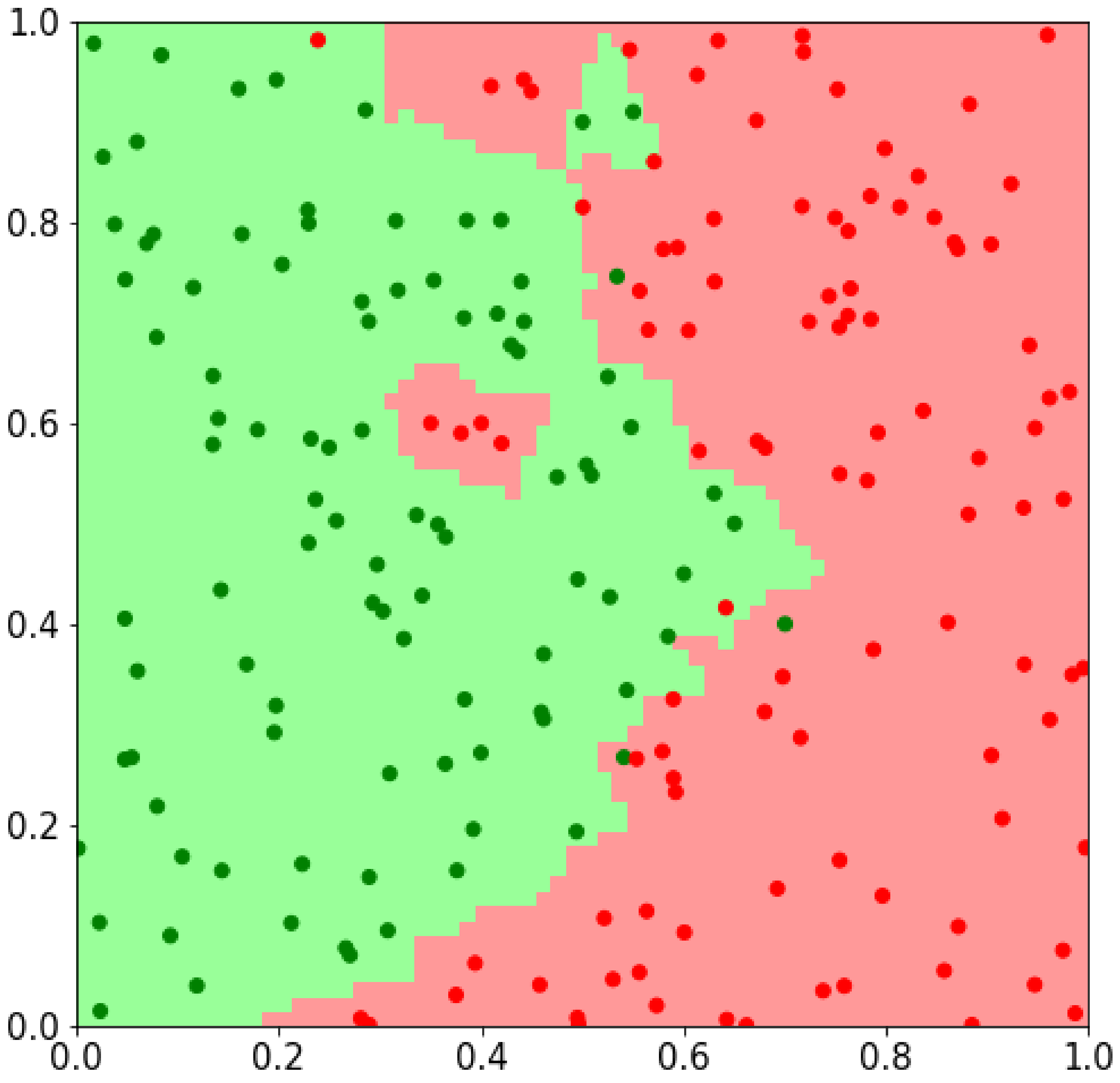
Decision Boundary

$k = 1$



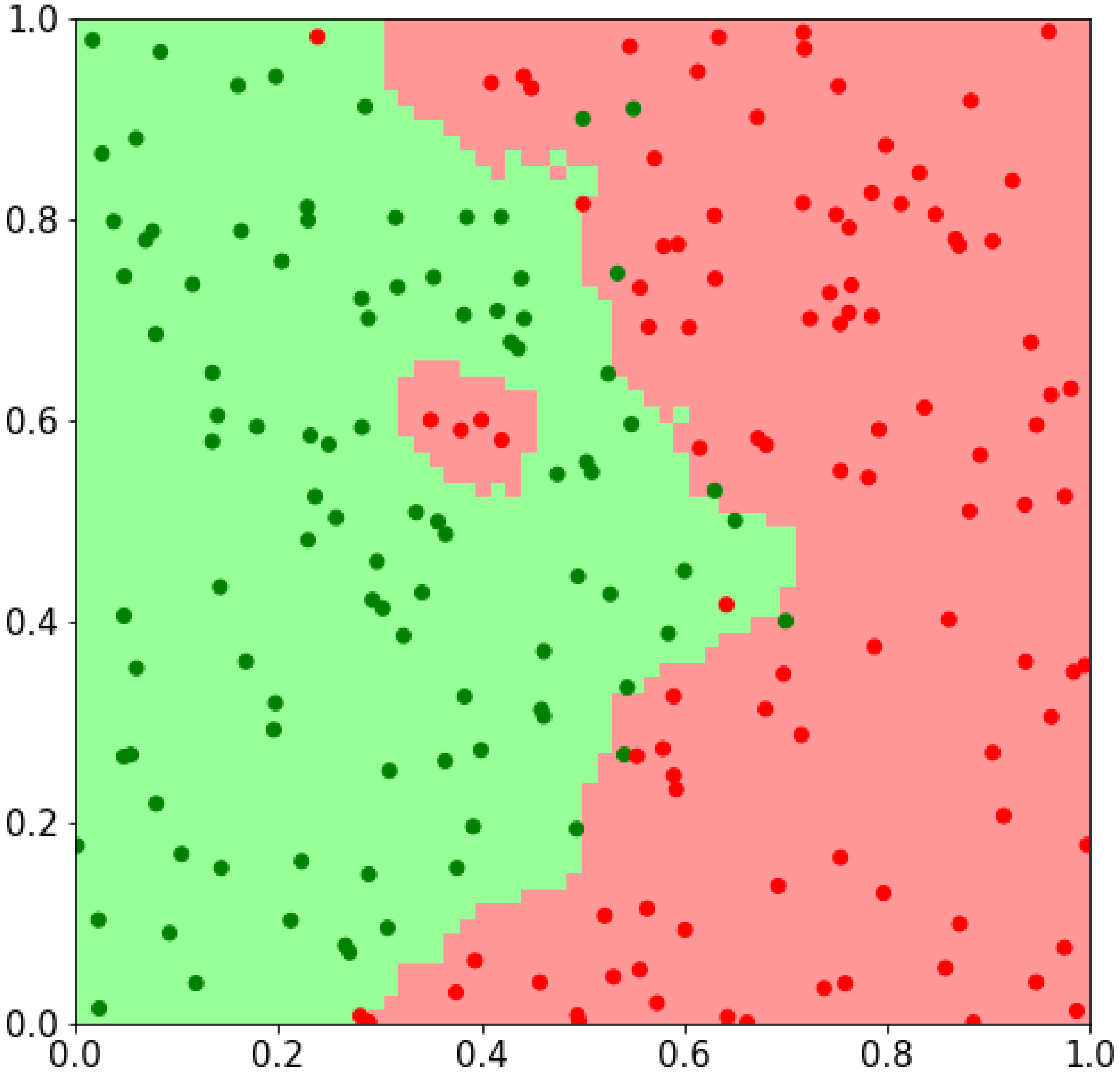
Decision Boundary

$k = 3$



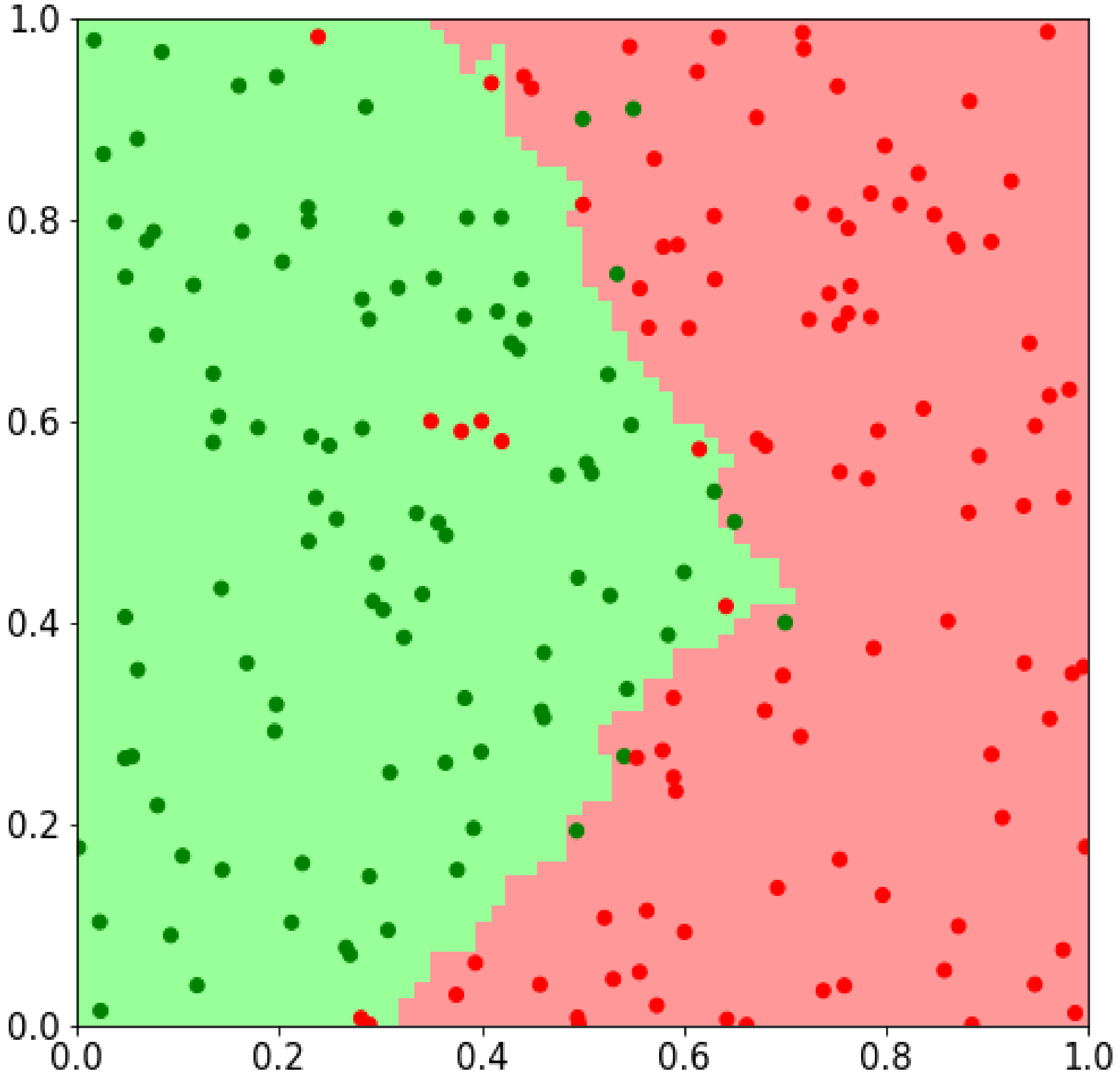
Decision Boundary

$k = 5$



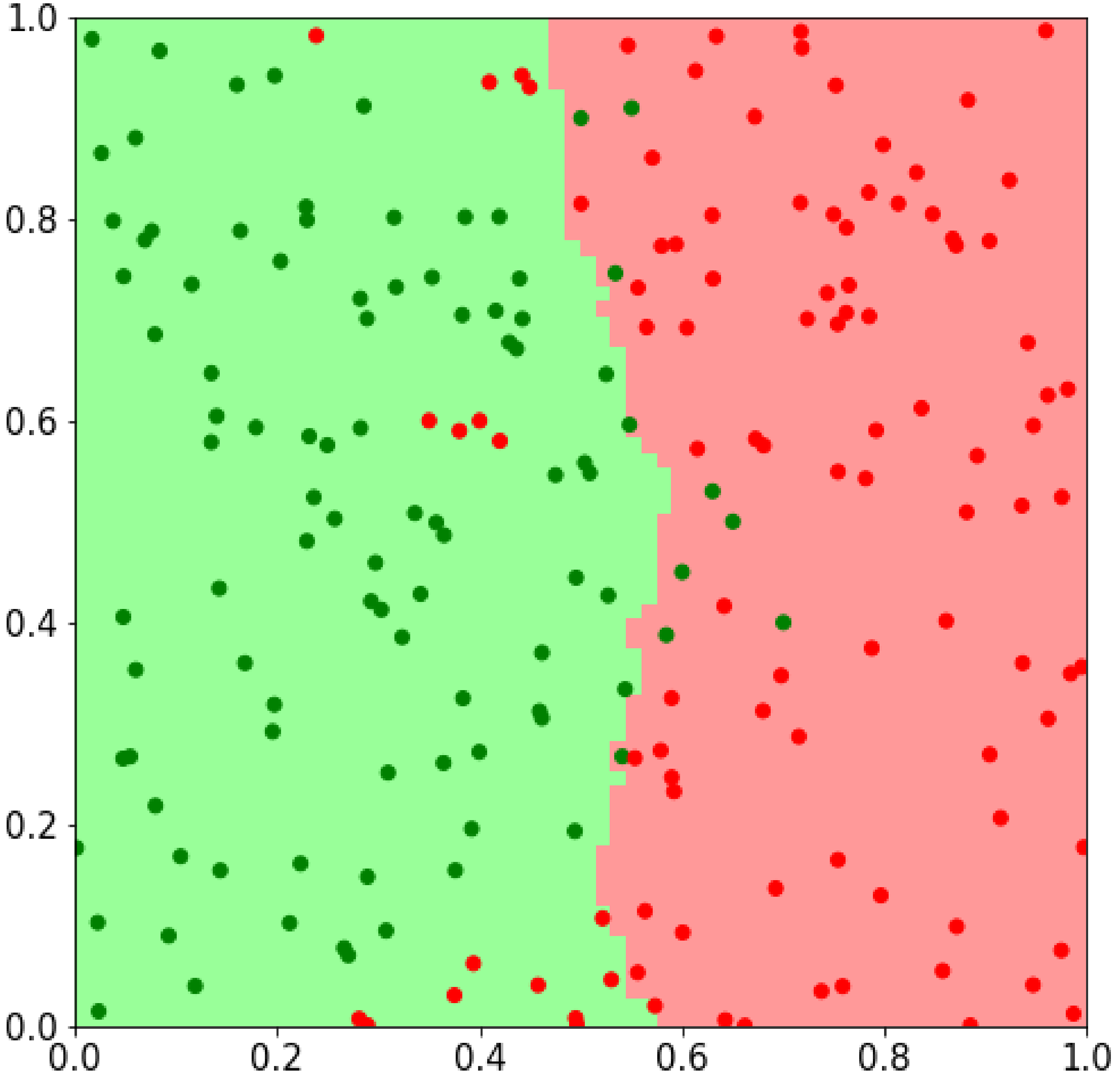
Decision Boundary

$k = 9$



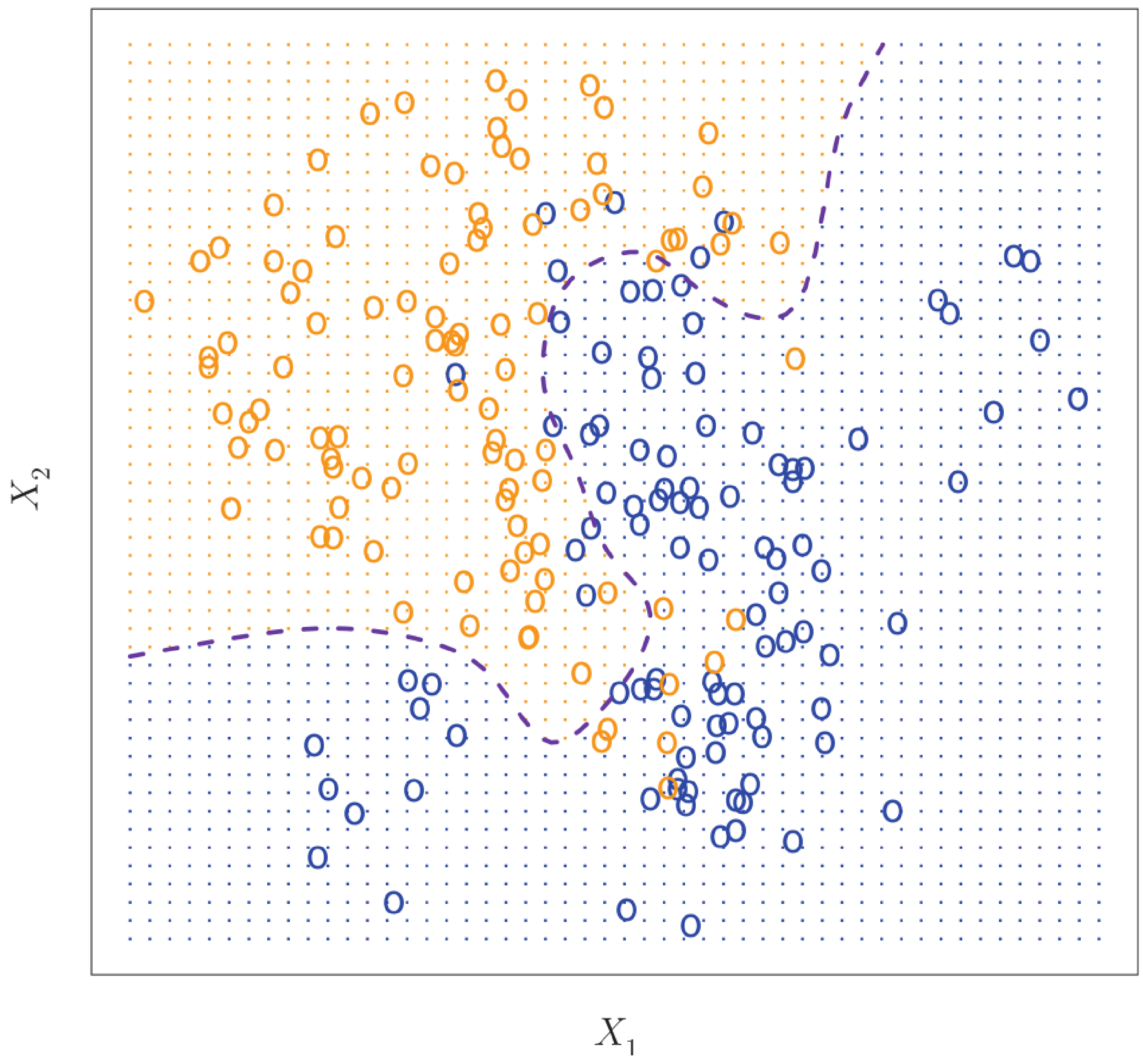
Decision Boundary

$k = 100$



Model Complexity

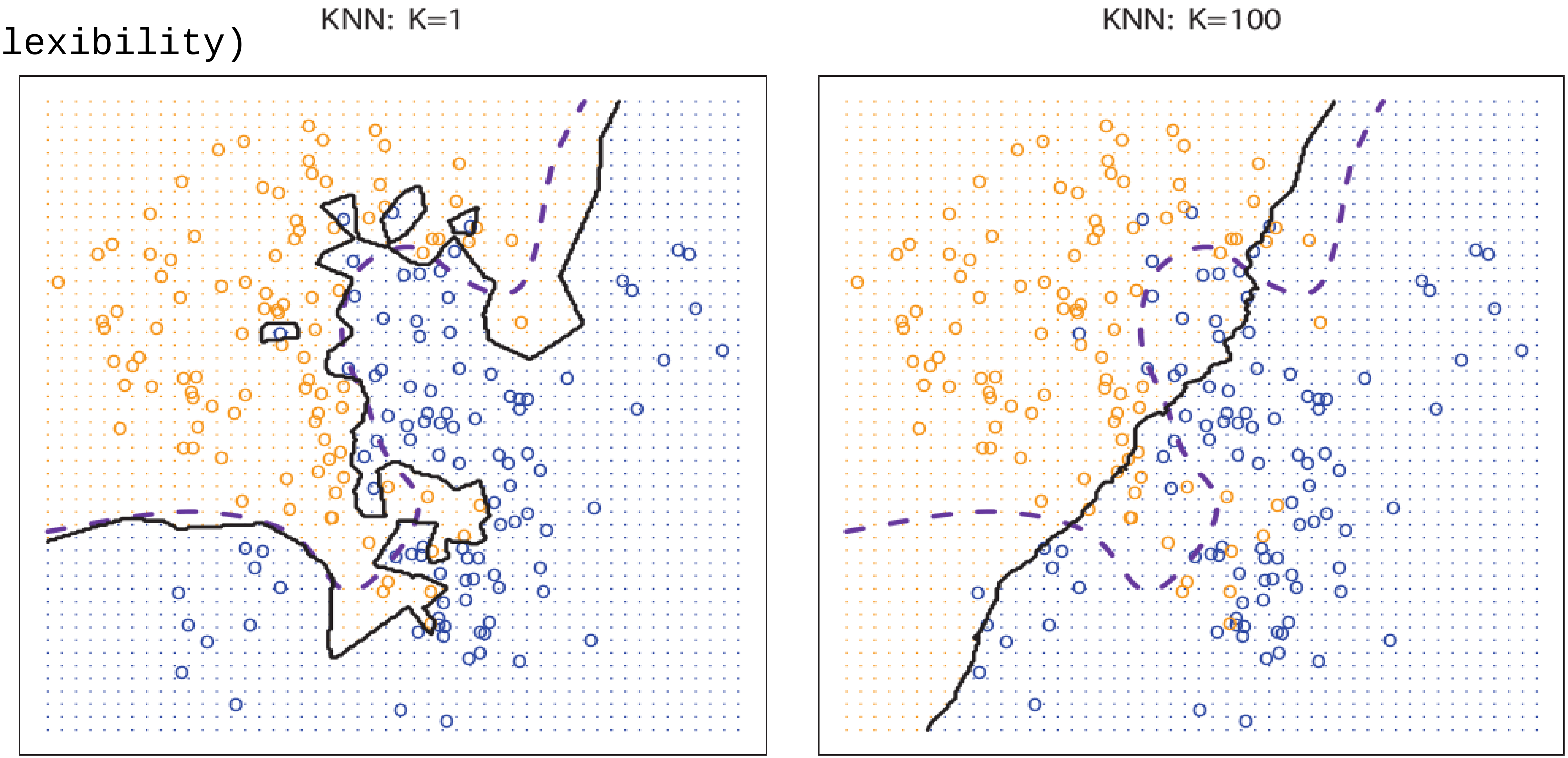
(flexibility)



Source: ISLR (38)

Model Complexity

(flexibility)



Source: ISLR (41)

Model Complexity

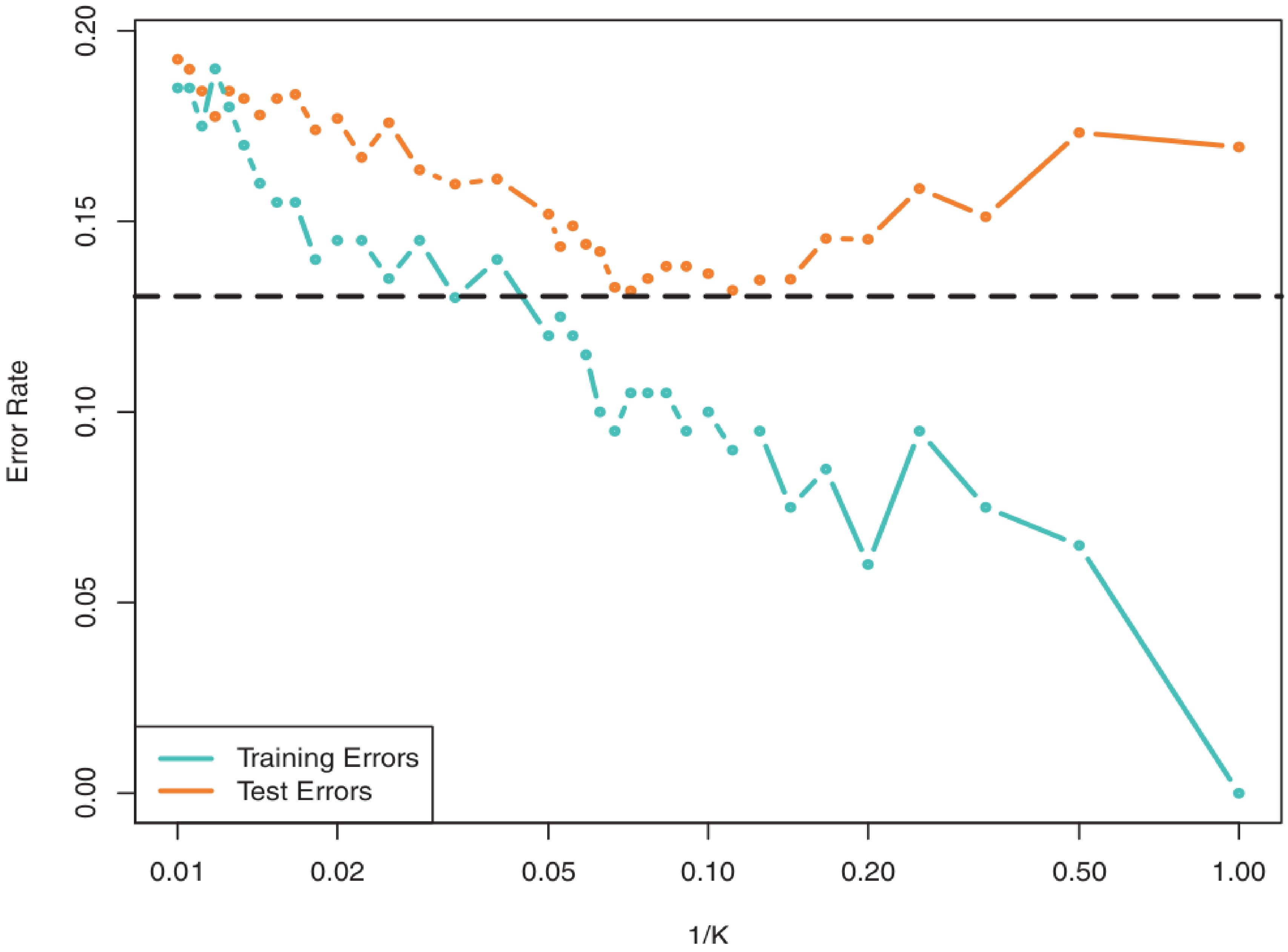
(flexibility)

$$\text{Loss}(h) = \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}[h(x_i) \neq y_i]$$

Training
- 200 points

Test
- 5000 points

Source: ISLR (42)



Validation

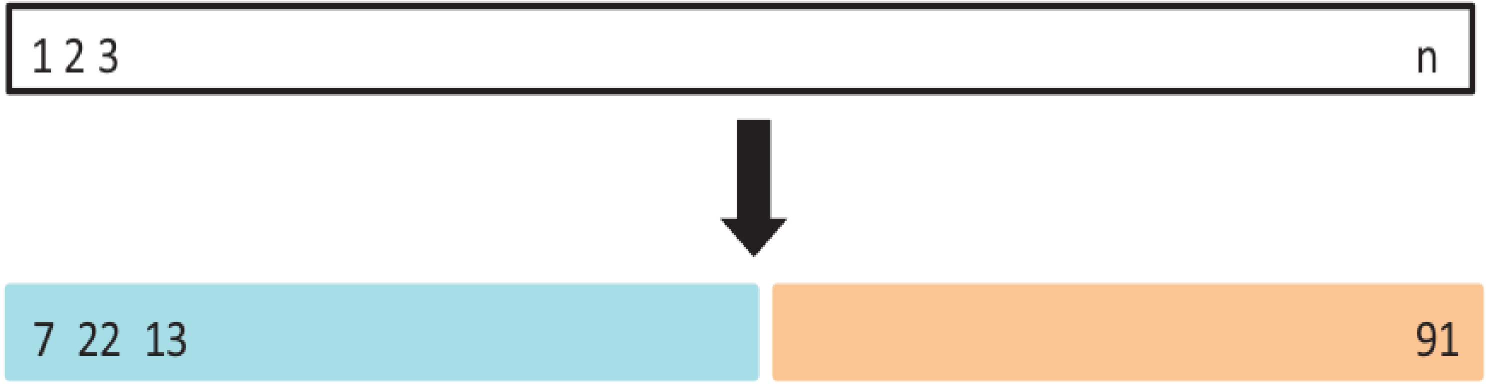
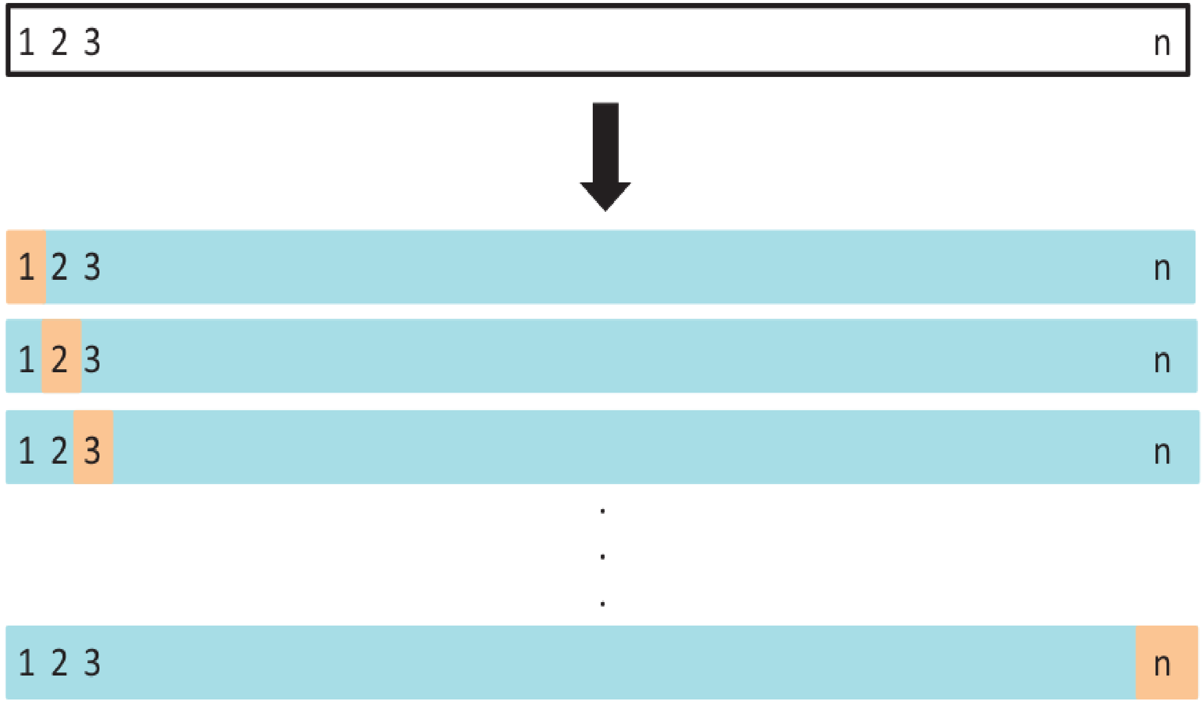


FIGURE 5.1. *A schematic display of the validation set approach. A set of n observations are randomly split into a training set (shown in blue, containing observations 7, 22, and 13, among others) and a validation set (shown in beige, and containing observation 91, among others). The statistical learning method is fit on the training set, and its performance is evaluated on the validation set.*

Source: ISLR (177)

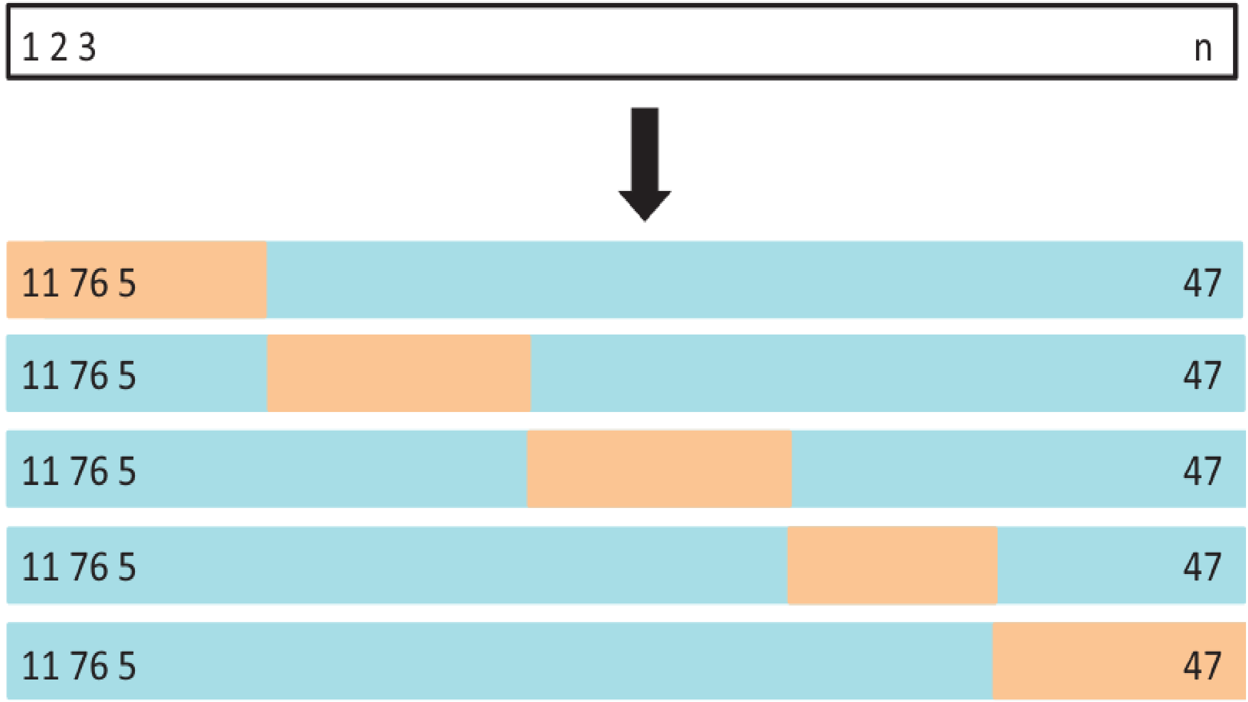
Cross Validation

Leave One Out CV (LOOCV)



$$CV_n = \frac{1}{n}\sum_{i=1}^{n} \text{Loss}(h_i)$$

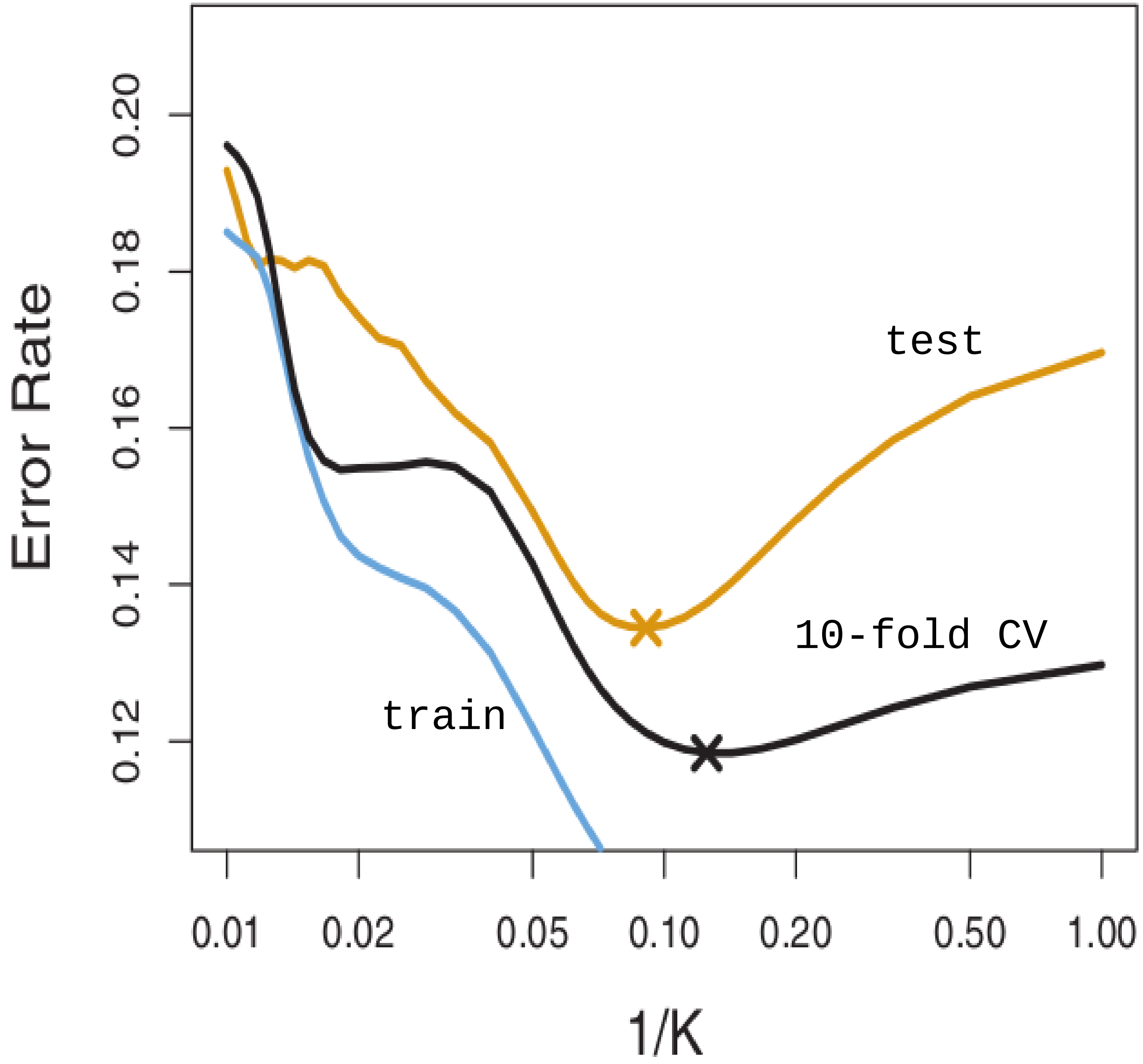
k-fold CV



$$CV_k = \frac{1}{k}\sum_{i=1}^{k} \text{Loss}(h_i)$$

Source: ISLR (179 & 181)

Cross Validation



Source: ISLR (186)

Advantages

- Very easy to implement
- Interpretable
- Easy to explain to non-experts

Disadvantages

- Computationally expensive
  - n_1 train, n_2 test
  - $n_1 n_2$ distances
  - Sort n_1 values n_2 times
- No model is learnt
  - Storage for training data
- Curse of dimensionality
  - Distances behave weirdly in higher dimensions