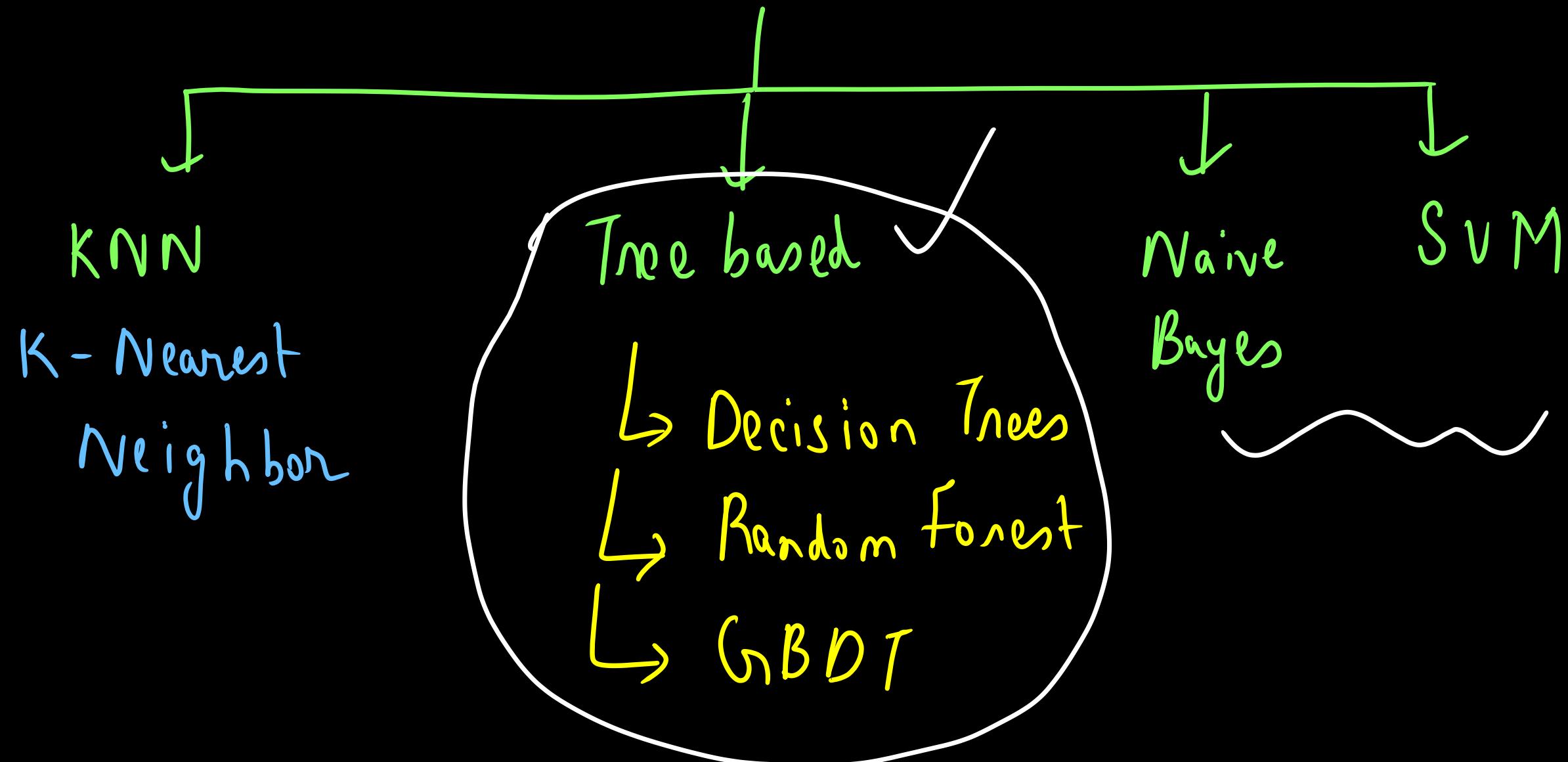


→ K-NN

ML - Supervised



K-NN (K- Nearest Neighbour).

Blinkit / Instamart

↳ Classify the stones as high traffic,
moderate traffic, low traffic.

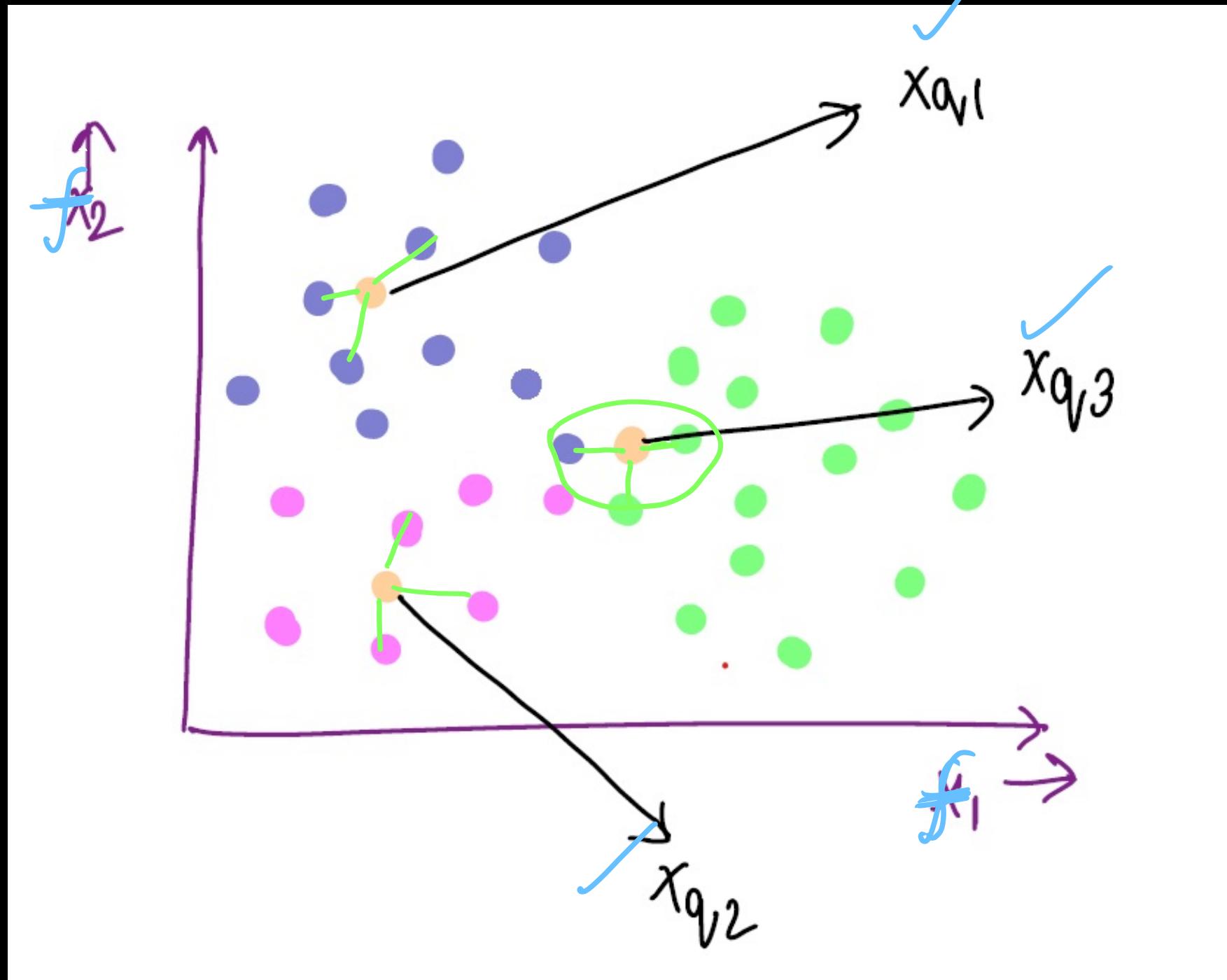
K - NN

↳ Non - linear

↳ Multi-class

↳ Classification / Regression

K=3 ✓



→ Pink
→ Green
→ Blue

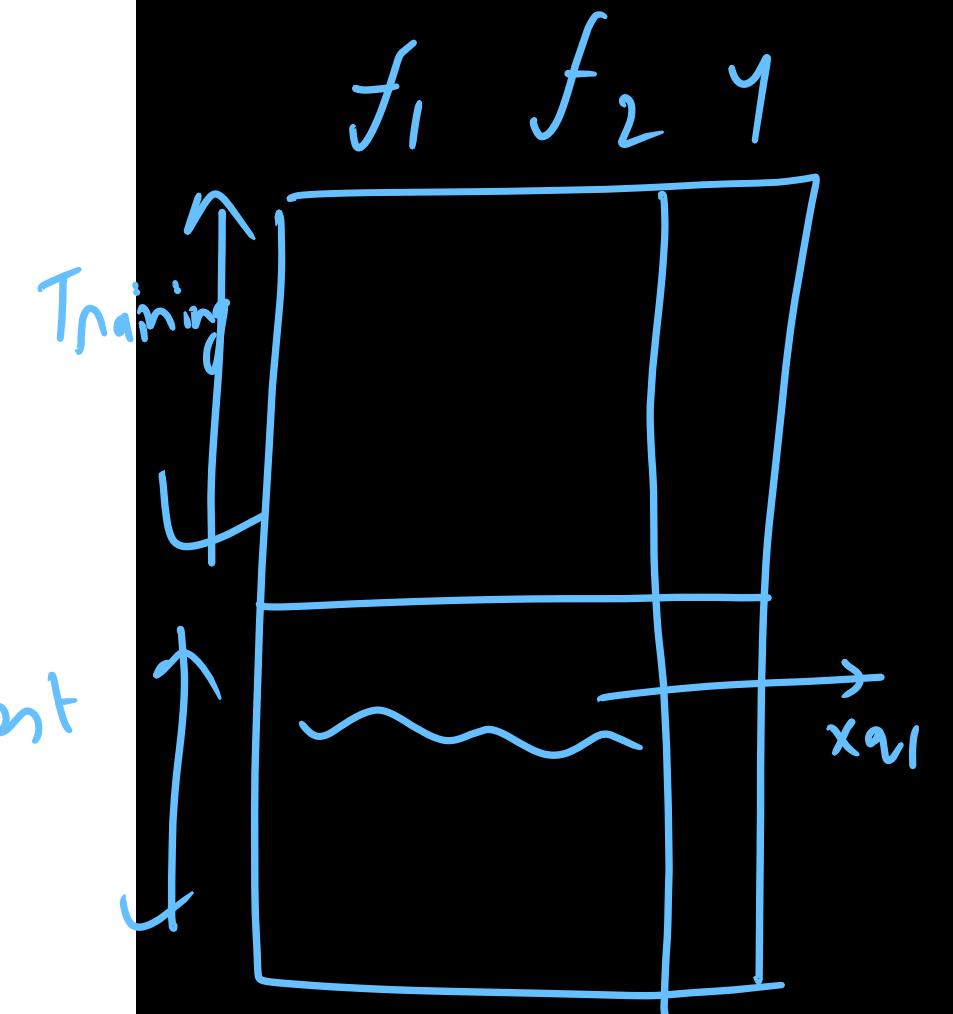
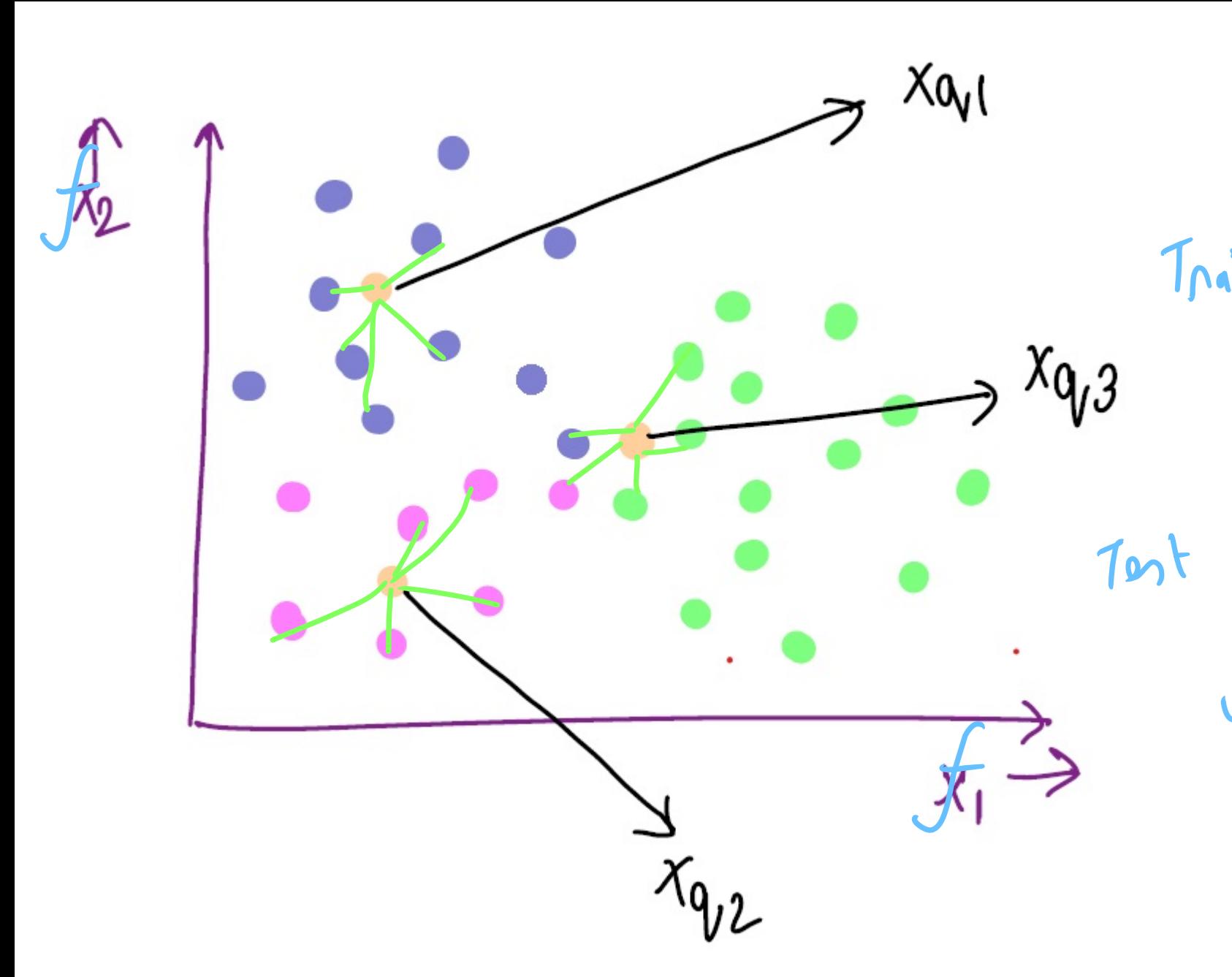
$x_{q1} \rightarrow$ blue

$x_{q2} \rightarrow$ pink

$x_{q3} \rightarrow$?? G

2 green, 1 blue

$K=5$



Algorithm

1. Compute the dist. of test point x_q ✓
from every training point
2. Take the top - K closest distance ✓
3. For classification → find the majority
with K class

```
def _predict(self, x):  
    # Compute distances between x and all examples in the training set  
    distances = [euclidean_distance(x, x_train) for x_train in self.X_train]  
    # Sort by distance and return indices of the first k neighbors  
    k_idx = np.argsort(distances)[: self.k]  
    # Extract the labels of the k nearest neighbor training samples  
    k_neighbor_labels = [self.y_train[i] for i in k_idx]
```

Logistic Regression

- GD
- ω^*, b^*
- Parametric algorithm

KNN

- No GD
- Store all training data
- KNN does everything at test time.
- Non parametric

k - NN

↳ K Nearest Neighbours



Hyperparameter

Training data point

	f_1	f_2	y
$x^{(1)}$	3	6	1
$x^{(2)}$	6	4	1
$x^{(3)}$	8	2	3
$x^{(4)}$	7	5	3
$x^{(5)}$	1	4	2
$x^{(6)}$	2	2	2

Test data point

	f_1	f_2	y
x_{q1}	2	5	

1. Compute distance of each test point x_q from every train datapoint

$\checkmark x_q = \begin{bmatrix} f_1 & f_2 \\ 2 & 5 \end{bmatrix}$ ← Test data point

	f_1	f_2	y	distance
$x^{(1)}$	3	6	1	1.41
$x^{(2)}$	6	4	1	4.123
$x^{(3)}$	8	2	3	6.708
$x^{(4)}$	7	5	3	5
$x^{(5)}$	1	4	2	1.41
$x^{(6)}$	2	2	2	3

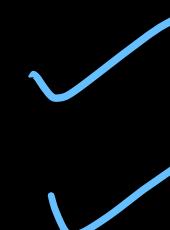
$$\begin{aligned} & \sqrt{(2-3)^2 + (5-6)^2} \\ &= \sqrt{1^2 + 1^2} = \sqrt{2} = 1.41 \\ & \sqrt{(2-6)^2 + (5-4)^2} \\ &= \sqrt{4^2 + 1^2} = \sqrt{17} \end{aligned}$$

→ Pick $k = 3$ data points having min. distance

Train

	f_1	f_2	y	distance
$x^{(1)}$	3	6	1	1.41
$x^{(2)}$	6	4	1	4.123
$x^{(3)}$	8	2	3	6.708
$x^{(4)}$	7	5	3	5
$x^{(5)}$	1	4	2	1.41
$x^{(6)}$	2	2	2	3

Test $x_{q1} = [2, 5]$



~~(1, 2, 2)~~

2

→ find majority class with K data points

	f_1	f_2	y	
$x^{(1)}$	3	6	1	<u>distance</u>
$x^{(5)}$	1	4	2	1.41
$x^{(6)}$	2	2	2	3

$K = 3$

$K=4$ (even) \rightarrow Tie \rightarrow Not choose even K

	f_1	f_2	y	<u>distance</u>
$x^{(1)}$	3	6	1	1.41
$x^{(2)}$	6	4	1	4.123
$x^{(3)}$	8	2	3	6.708
$x^{(4)}$	7	5	3	5
$x^{(5)}$	1	4	2	1.41
$x^{(6)}$	2	2	2	3

1, 1, 2, 2

K=5 (odd) \rightarrow tie $\rightarrow \underbrace{1 \text{ or } 2}$

1, 1, 2, 2, 3

	f_1	f_2	y	<u>distance</u>
$x^{(1)}$	3	6	1	1.41
$x^{(2)}$	6	4	1	4.123
$x^{(3)}$	8	2	3	6.708
$x^{(4)}$	7	5	3	5
$x^{(5)}$	1	4	2	1.41
$x^{(6)}$	2	2	2	3

K=3

1, 2, 3



Assumption

→ Data points that are close in
feature space are likely
to belong to same class.

