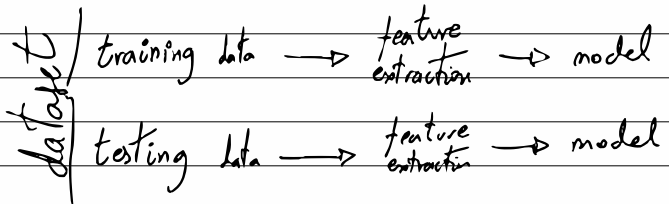# Classification

Computer must learn from data $\Rightarrow$ find underlying patterns.

- **Training:**
   inputs (every input arranged to one vector)
   feature extraction $\rightarrow$ representation of inputs

- **Testing**

   dataset {
   training data $\longrightarrow$ feature extraction $\rightarrow$ model

   testing data $\longrightarrow$ feature extraction $\rightarrow$ model
   }

Nearest Neighbour methods

1/ Compute the "euclidian distance" to every single
   point in the feature space

2/ Choose the "nearest point"'s class as its label.

2/ Majority vote: make a radius and choose
   the class which more points in that radius belong to.

   * How many points should be considered?

   $\rightarrow$ > k : too rigid. (get rid of noise and outliers). $\longrightarrow$ (generalized).
   $\rightarrow$ < k : overfitting. ($\downarrow$ computational cost) $\longrightarrow$ need for many samples ?

   ✓ KNN ✗
   {
   simple
   one parameter (k)   costly
   multi-class   memory required
   }