

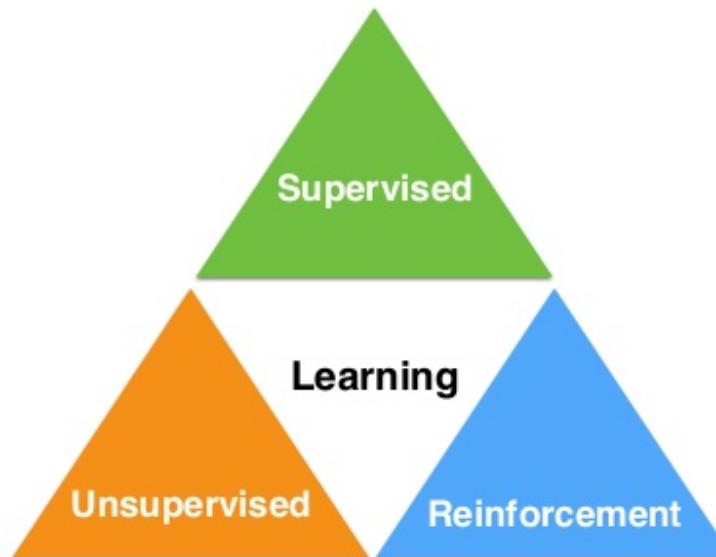
A photograph of a classic brick clock tower. The tower features a large, ornate clock face with Roman numerals and a small secondary dial below it. It is topped with a white, fluted column and a small sphere. The tower is made of red brick and has white decorative elements. The background is a clear blue sky.

Introduction to Machine Learning

Clustering and Python ML Libraries

Types of machine learning

- Labeled data
- Direct feedback
- Predict outcome/future



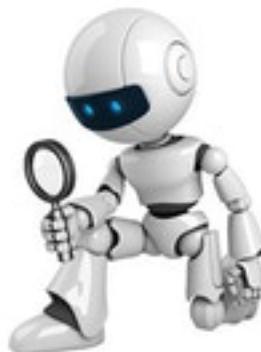
- No labels
- No feedback
- “Find hidden structure”

- Decision process
- Reward system
- Learn series of actions

Supervised Learning



Unsupervised Learning

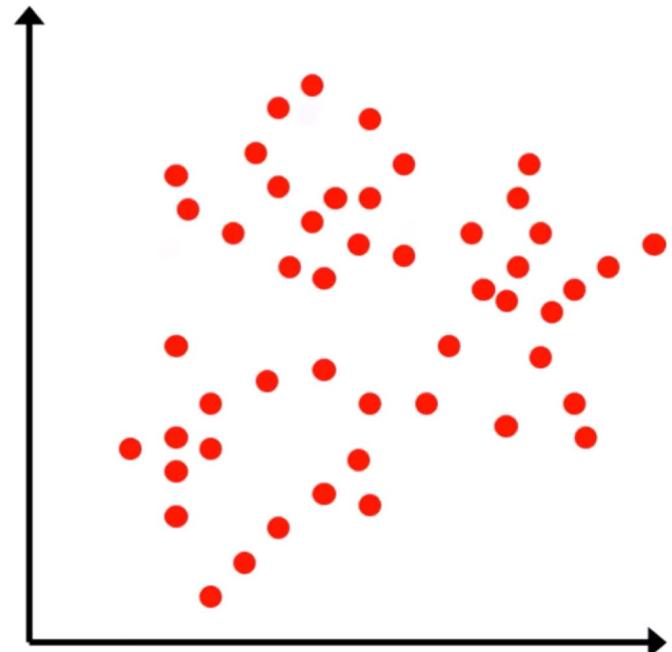


Reinforcement Learning



K-Mean clustering

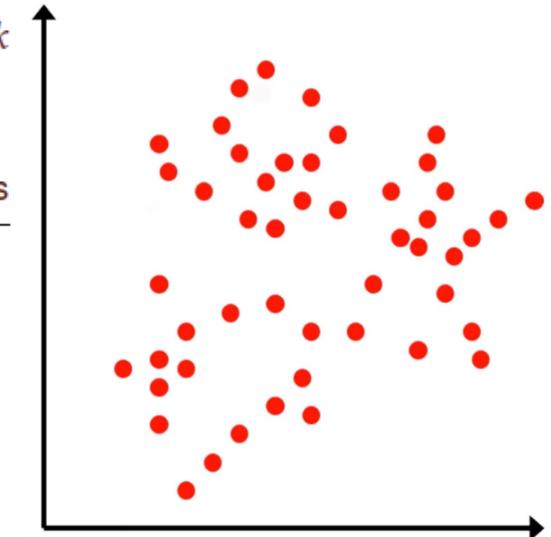
- Unsupervised learning
- Data is just examples x_1, \dots, x_N





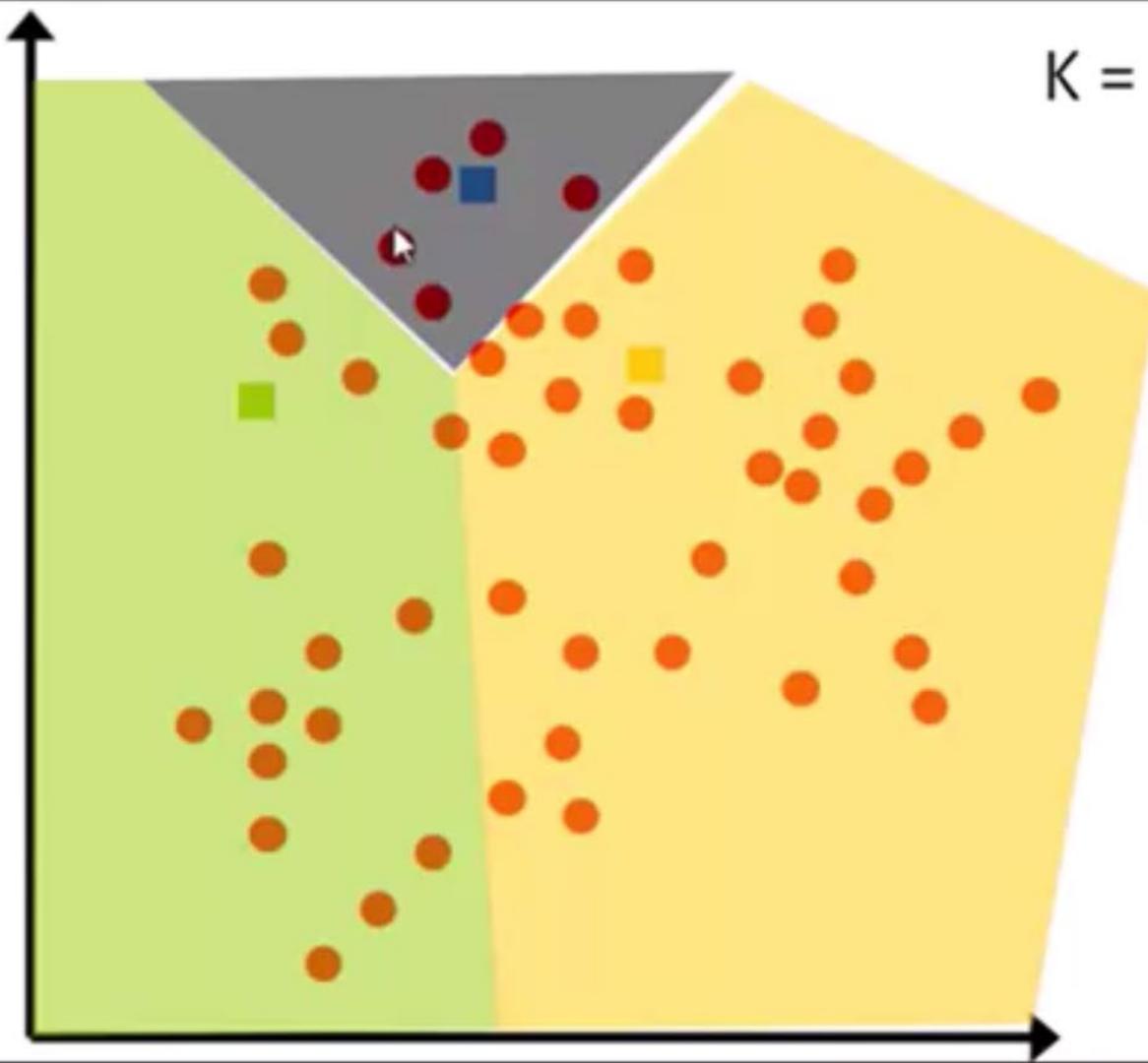
Algorithm 4 K-MEANS(D, K)

```
1: for  $k = 1$  to  $K$  do
2:    $\mu_k \leftarrow$  some random location      // randomly initialize center for  $k$ th cluster
3: end for
4: repeat
5:   for  $n = 1$  to  $N$  do
6:      $z_n \leftarrow \operatorname{argmin}_k ||\mu_k - x_n||$     // assign example  $n$  to closest center
7:   end for
8:   for  $k = 1$  to  $K$  do
9:      $X_k \leftarrow \{ x_n : z_n = k \}$           // points assigned to cluster  $k$ 
10:     $\mu_k \leftarrow \operatorname{MEAN}(X_k)$            // re-estimate center of cluster  $k$ 
11:  end for
12: until  $\mu$ s stop changing
13: return  $z$                                 // return cluster assignments
```

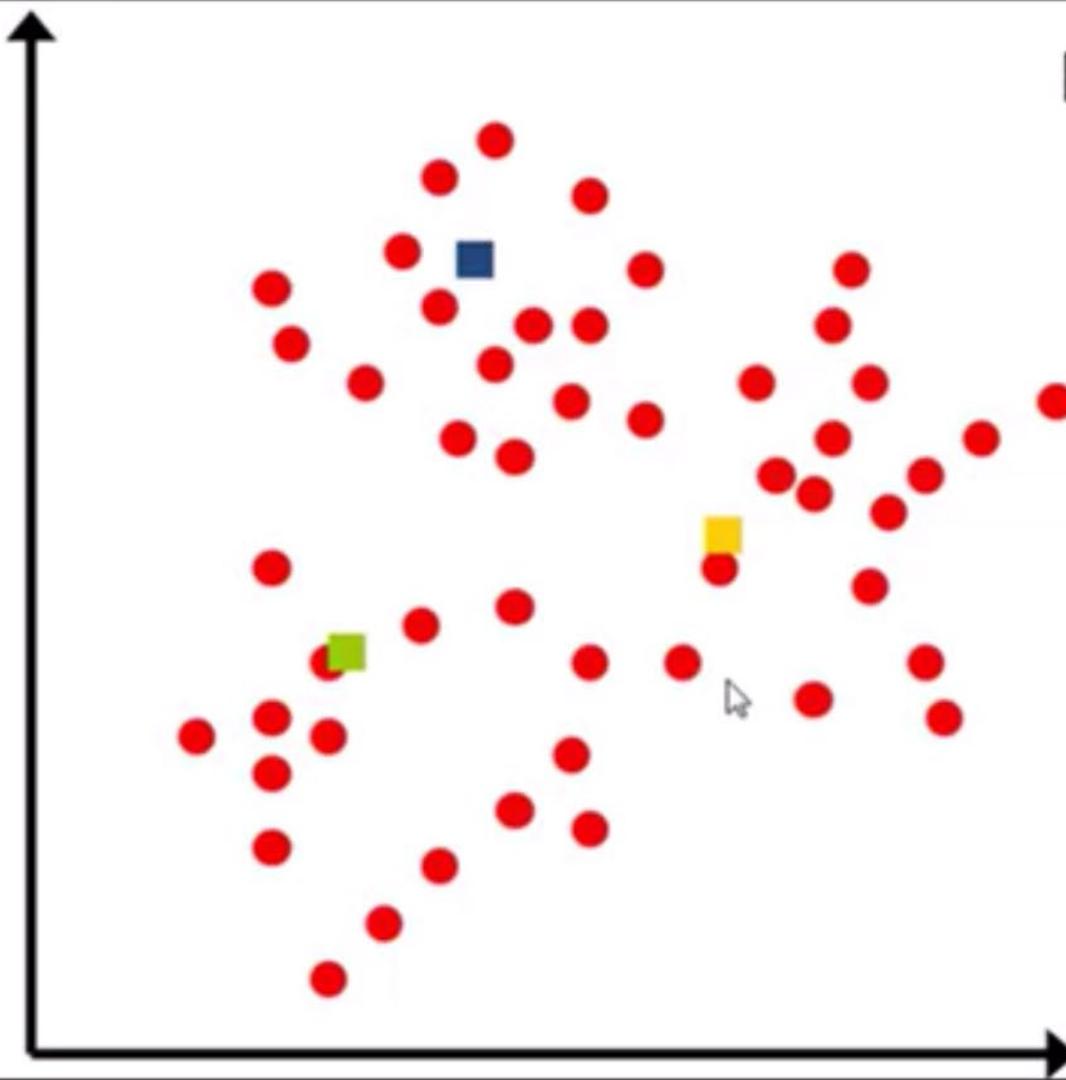




$K = 3$



$K = 3$





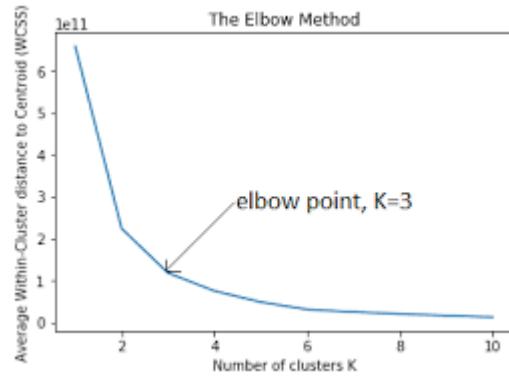


Will it converge (stop changing)?

- Will k-means converge (stop changing)?

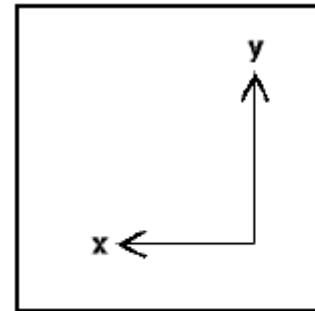
K-Means hyperparameters

- K
 - Choose using elbow method

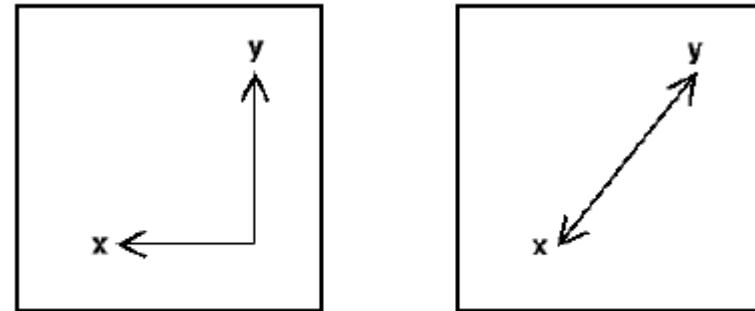


K-Means hyperparameters

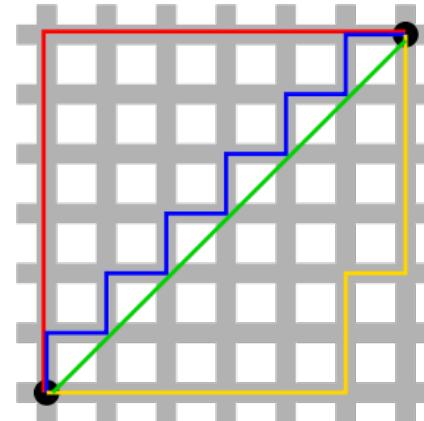
- Distance
 - Euclidean
 - Manhattan



Manhattan

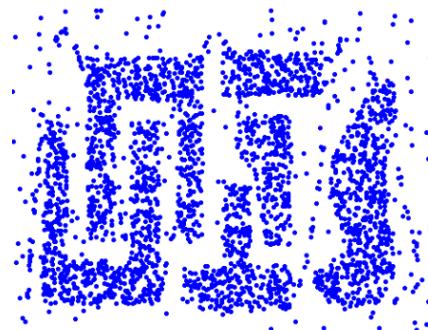


Euclidean

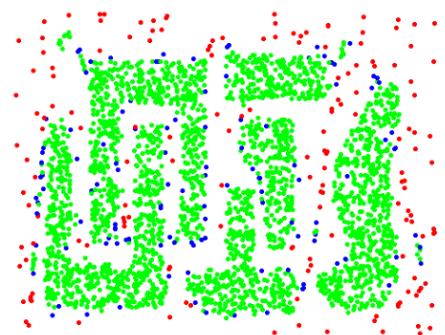


Alternative: Density-based clustering (DBSCAN)

- ε -neighborhood: objects within distance ε
- Three types of points
 - Core: ε -neighborhood contains $\geq \text{MinPts}$
 - Border: not core but inside neighborhood of a core point
 - Noise: any other point



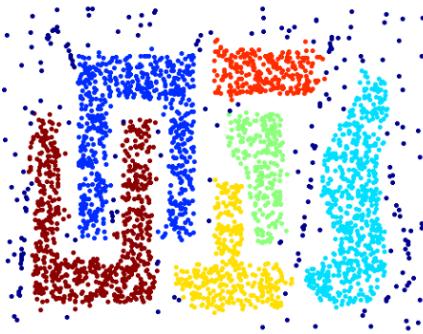
Original Points



Point types: core,
border and outliers

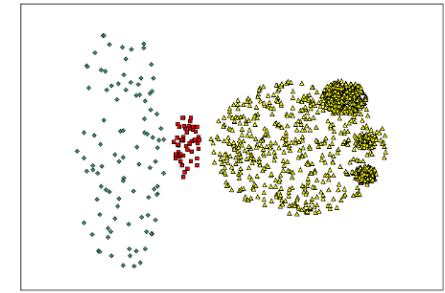
Alternative: Density-based clustering (DBSCAN)

- Cluster
 - Core and all points density-reachable from core
 - Density-reachable: accessible through chain of ε -neighborhoods

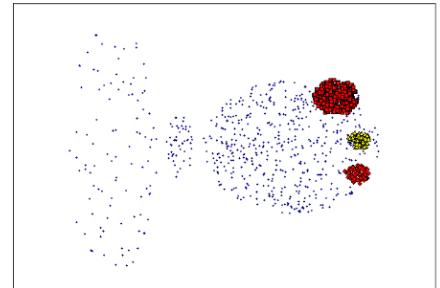


Original Points

Clusters



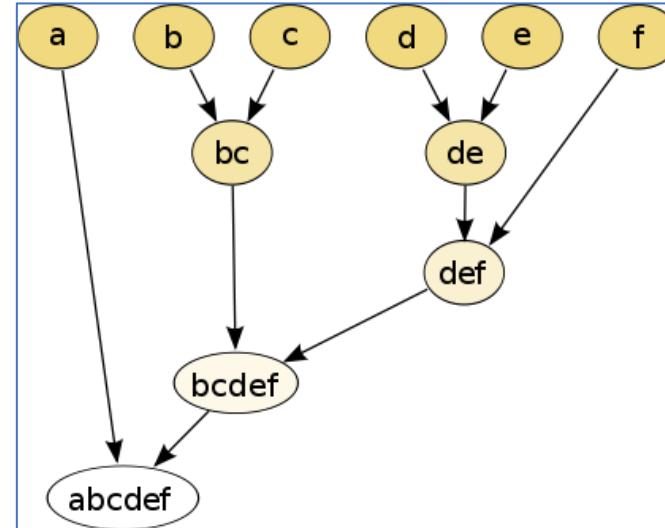
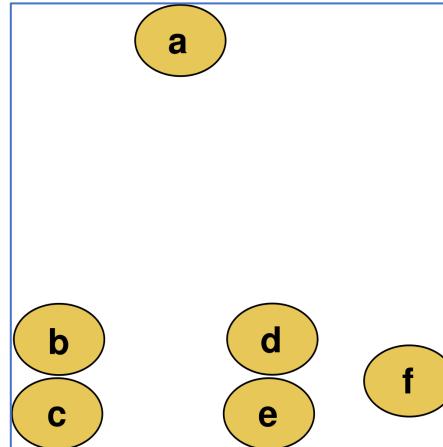
(MinPts=4, Eps=9.92).



(MinPts=4, Eps=9.75)

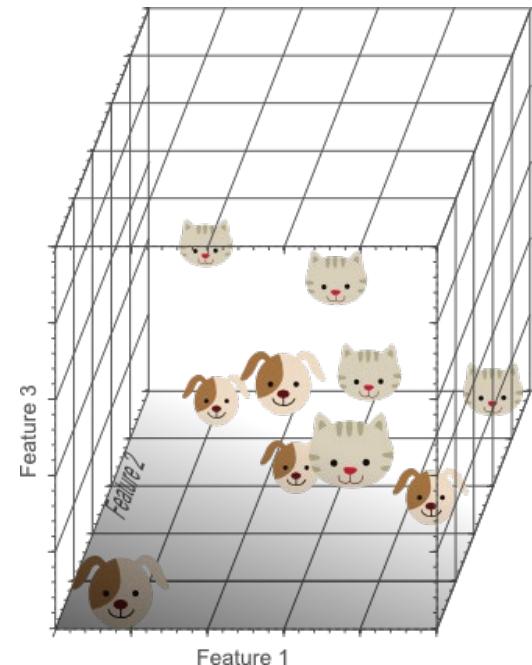
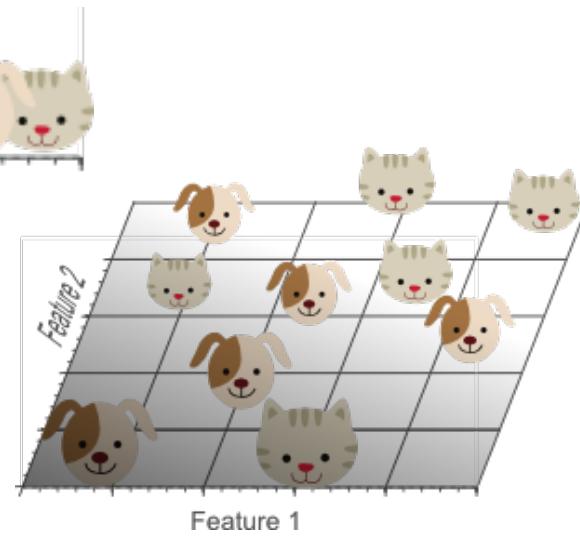
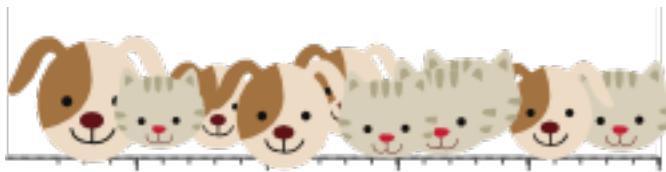
Alternative: Hierarchical clustering

- Build hierarchy (binary tree) of clusters using greedy method
- Top down (divisive): Start with one cluster, split
- Bottom up (agglomerative): Start with $\# \text{clusters} = \# \text{data points}$, merge
- Result is dendrogram



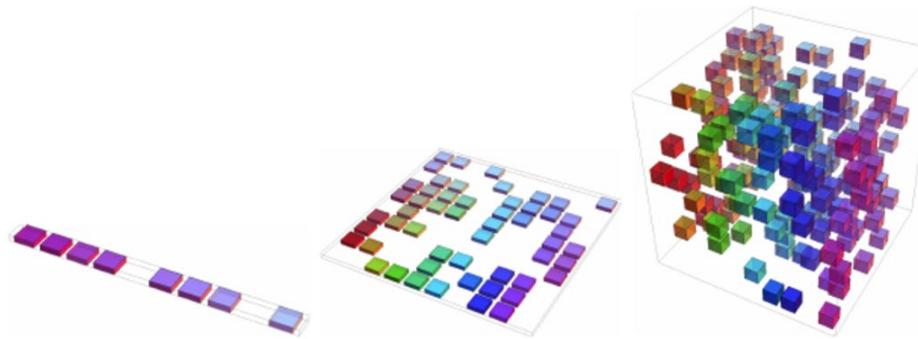
The Curse of Dimensionality

- Cost

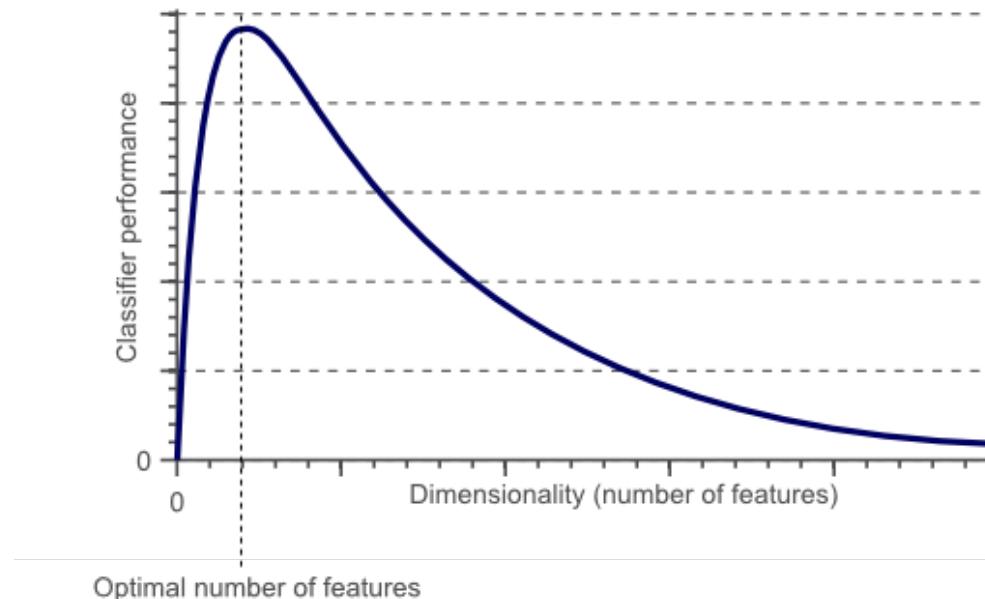


The Curse of Dimensionality

- ML performance

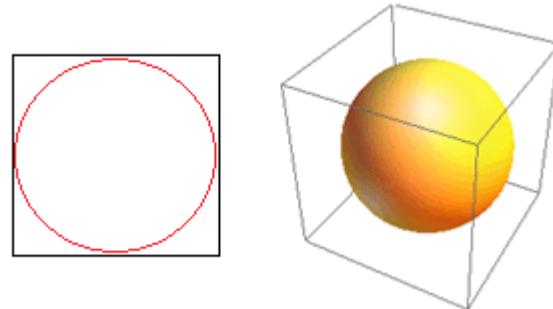


The Curse of Dimensionality



The Curse of Dimensionality

- Cost
- Classifier performance
- Mathematical phenomena



Sklearn (scikit-learn)