Lecture 18: Introduction to Clustering

We now turn our attention away from supervised learning & focus instead on unsupervised learning & in particular clustering.

Broad def^n of unsupervised learning:

- Any learning task that relies only on input data $X = \{\underline{x}_0, \ldots, \underline{x}_{N-1}\}$ & does not rely on corresponding outputs $\{y_0, \ldots, y_{N-1}\}$.

ex PCA

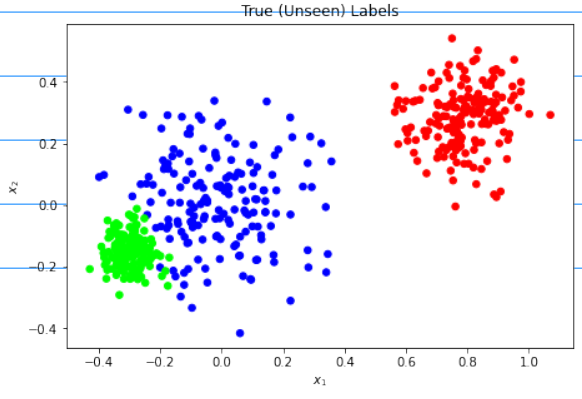
Clustering:

The unsupervised task of dividing the input data X into meaningful groups.

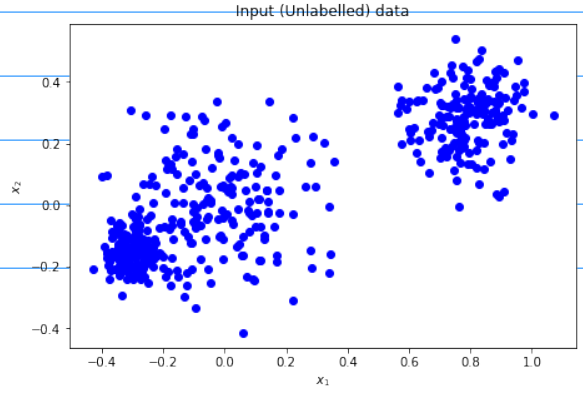
the task of discovering meaningful structure in the data.

Let us consider a toy example:

X, Y



X



In practical applications we only see X & not the labels Y so the goal is to group the X into meaningful clusters.

A simple & intuitive solution is to cluster X based on pairwise distances of the $\underline{x}_i$'s.

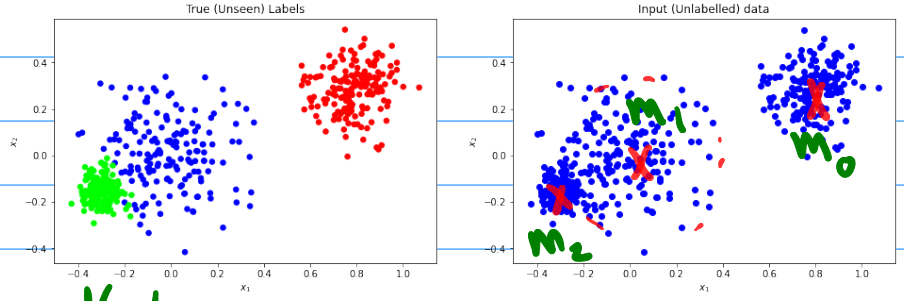
i.e. split X into K clusters

$$X = \bigcup_{k=0}^{K-1} C_k$$

s.t. $C_k \cap C_l = \emptyset$ & all $\underline{x}_i \in C_k$ are "close" in some sense.

18.1 K-Means

A particularly effective approach is called K-means clustering (simply K-means).



- Choose $K \geq 0$
- Choose/find Centers $\underline{m}_k, k = 0, \cdots, K-1$
- Assign $\underline{x}_i$ to cluster C_k with center $\underline{m}_k$, if

$$k = \underset{l=0,\cdots,K-1}{\operatorname{argmin}} \operatorname{dist}(\underline{x}_i, \underline{m}_l)$$

This simple recipe is often implemented in an iterative manner.

$$X = \{\underline{x}_0, \ldots, \underline{x}_{N-1}\}, \ \underline{x}_i \in \mathbb{R}^d, \ X = \bigcup_{k=0}^{K-1} C_k$$

write N_k for number of points assigned to cluster C_k.

Define the sum of squares error for C_k as

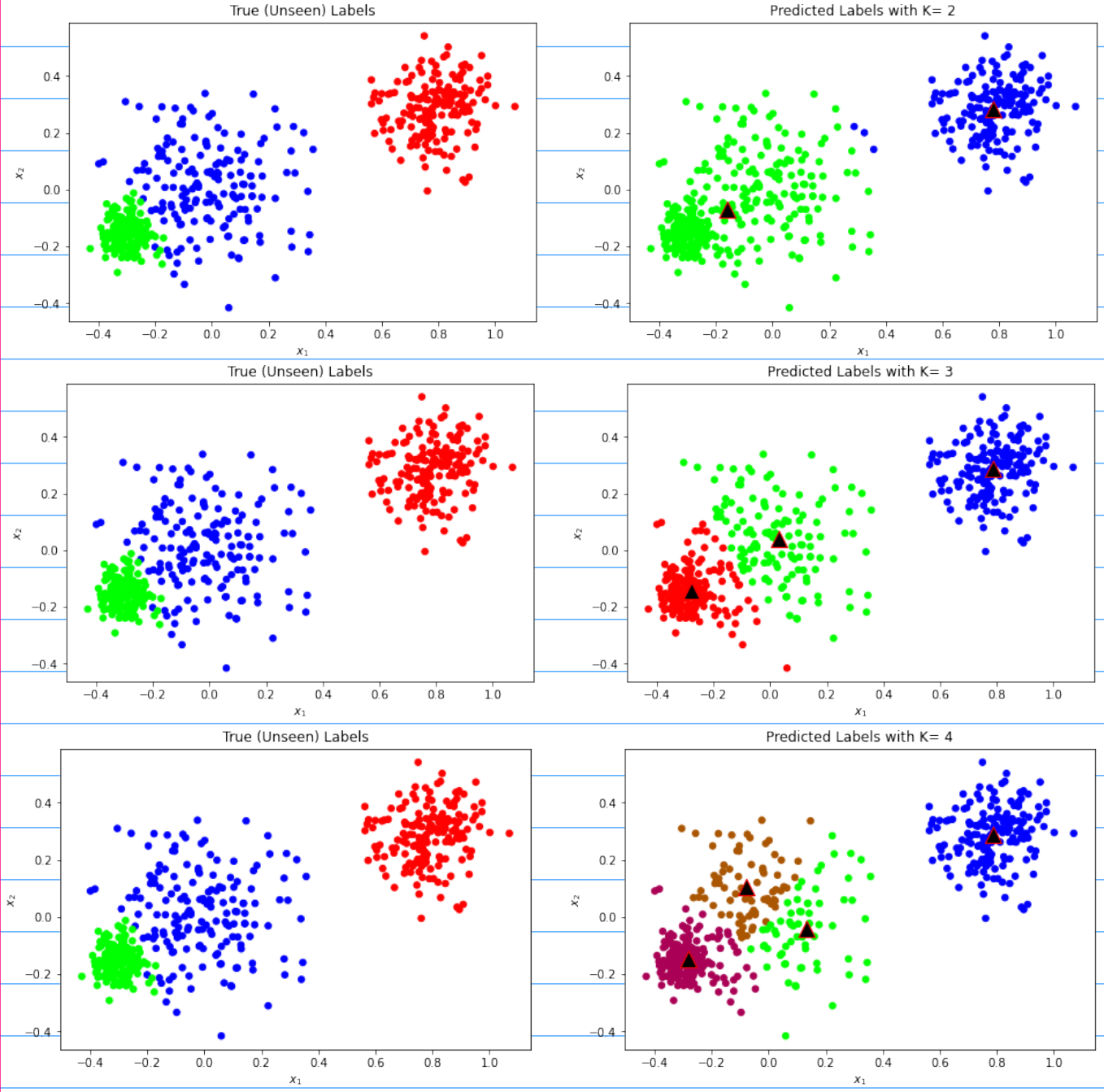
$$SSE(k) = \sum_{\{i \mid \underline{x}_i \in C_k\}} \|\underline{x}_i - \underline{m}_k\|^2$$

along with the cost function

$$KMM(\{C_k\}_{k=0}^{K-1}, \{\underline{m}_k\}_{k=0}^{K-1}) = \sum_{k=0}^{K-1} N_k \times SSE(k)$$

Then K-means clustering of X is done as.

1. Pick $\underline{m}_k$'s, ex randomly.
2. Compute C_k by solving
$$\underset{C_k}{\text{minimize}} \ KMM(C_k, \underline{m}_k).$$
3. Update the $\underline{m}_k$ by solving (finds means of current clusters)
$$\underset{\underline{m}_k}{\text{minimize}} \ KMM(C_k, \underline{m}_k).$$
4. Go to step 2 & repeat until convergence.



18.2 Choosing K

Note that the number of clusters K is chosen by the user. Clearly there is an optimal choice.

- if K is small we miss important clusters.
- if K is too large clusters are not meaningful. (Extreme case: K=N).

Something like CV does not exist here.
Instead we use the elbow strategy.

This is very heuristic but it works well.

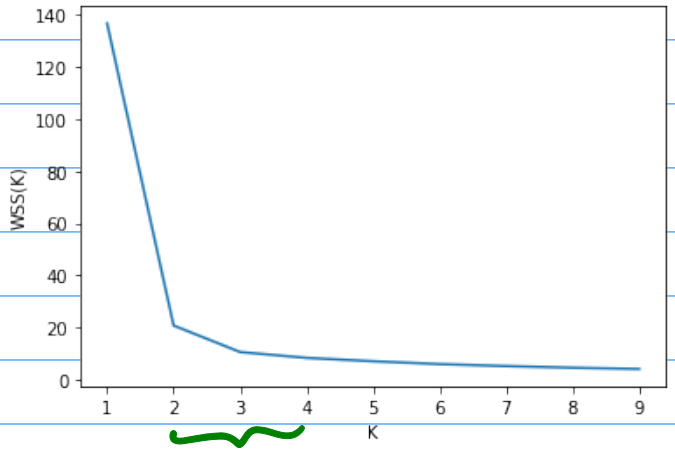
- Start with $k=1$ & increase k upto a given maximum number.
- Do K-means clustering for each K & compute

$$W(K) = \sum_{k=0}^{K-1} SSE(k)$$

← within cluster SSE.

- Plot W(K) as a function of K.

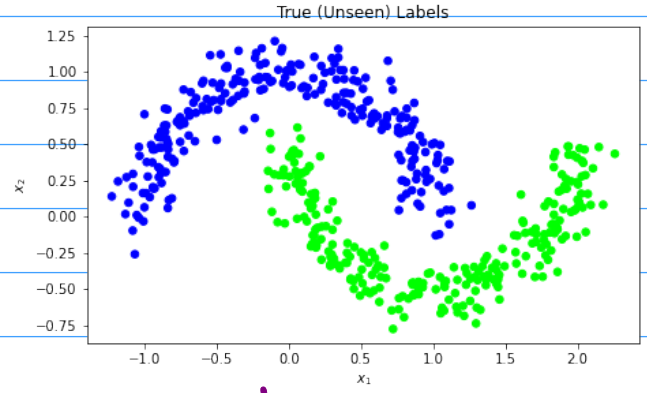
- Pick K in the "elbow" region.

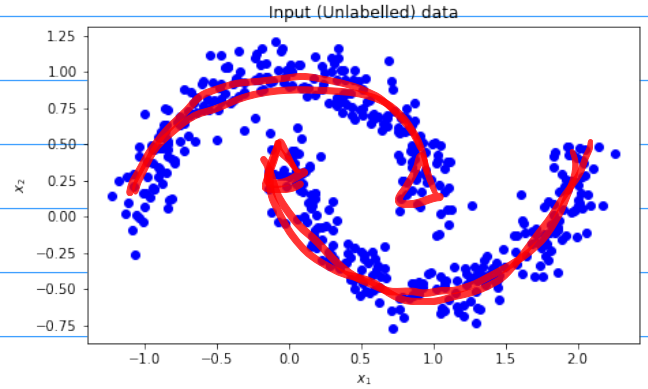


18.3 Problems with K-means

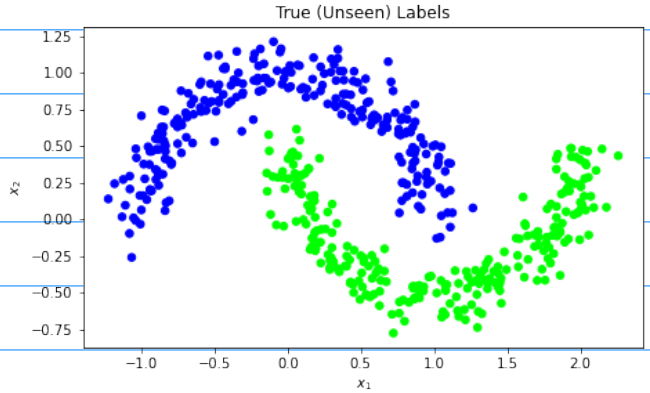
Even if we knew the correct value of k.
The k-means approach has major issue.
In particular that it is based on the notion of "closeness" in the Euclidean space.

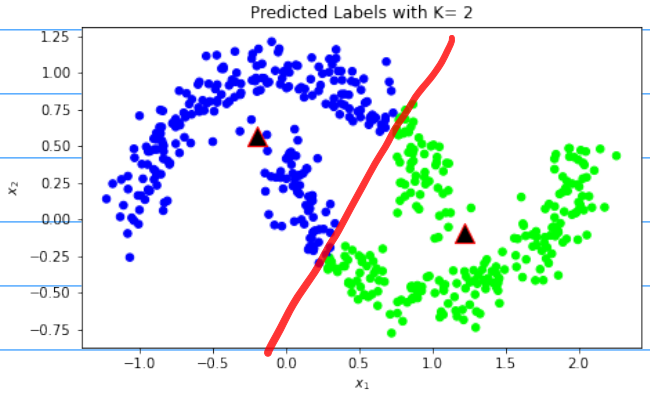
eg the two moons dataset:





clearly there are two clusters around two half circles. But the points within each half circle are not necessarily close!





K-Means fails! as it is blind to the geometry of the data! ie the half circles.