Clustering

Clustering

- Clustering uses **unlabeled data** and looks for similarities between groups (clusters) in order to attempt to segment the data into separate clusters.
- Keep in mind that we don't actually know the true correct label for this data!

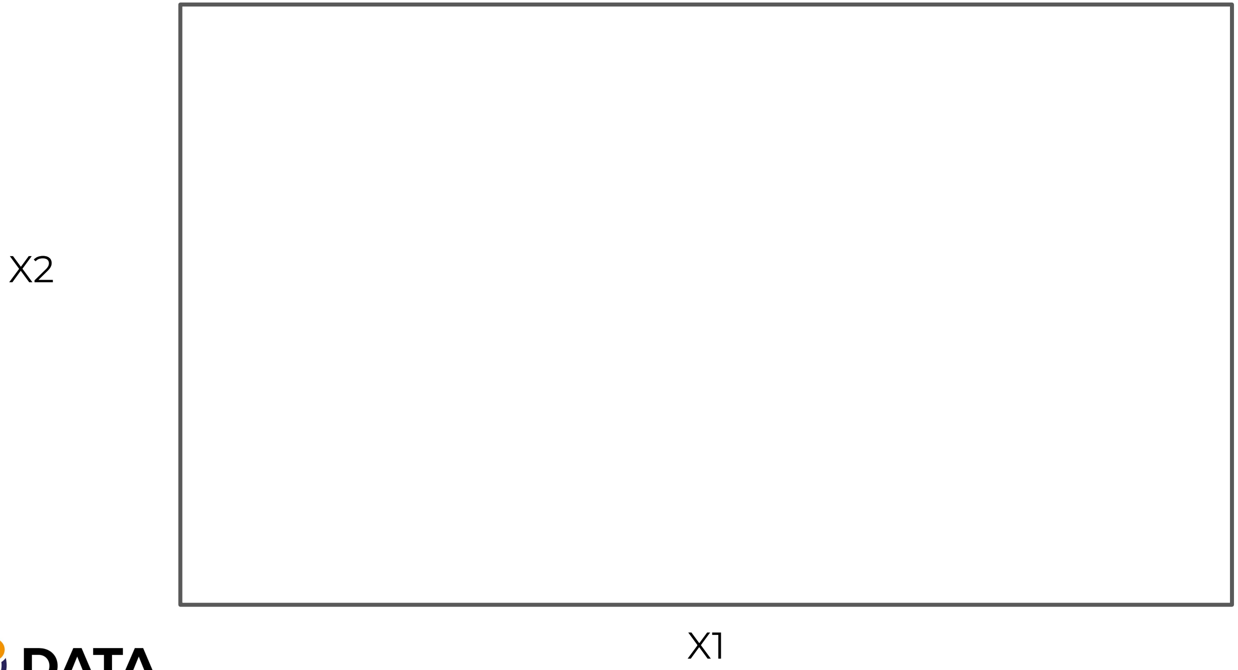
Clustering

- How could we cluster this data together?

X1	X2
2	4
6	3
...	...
1	2

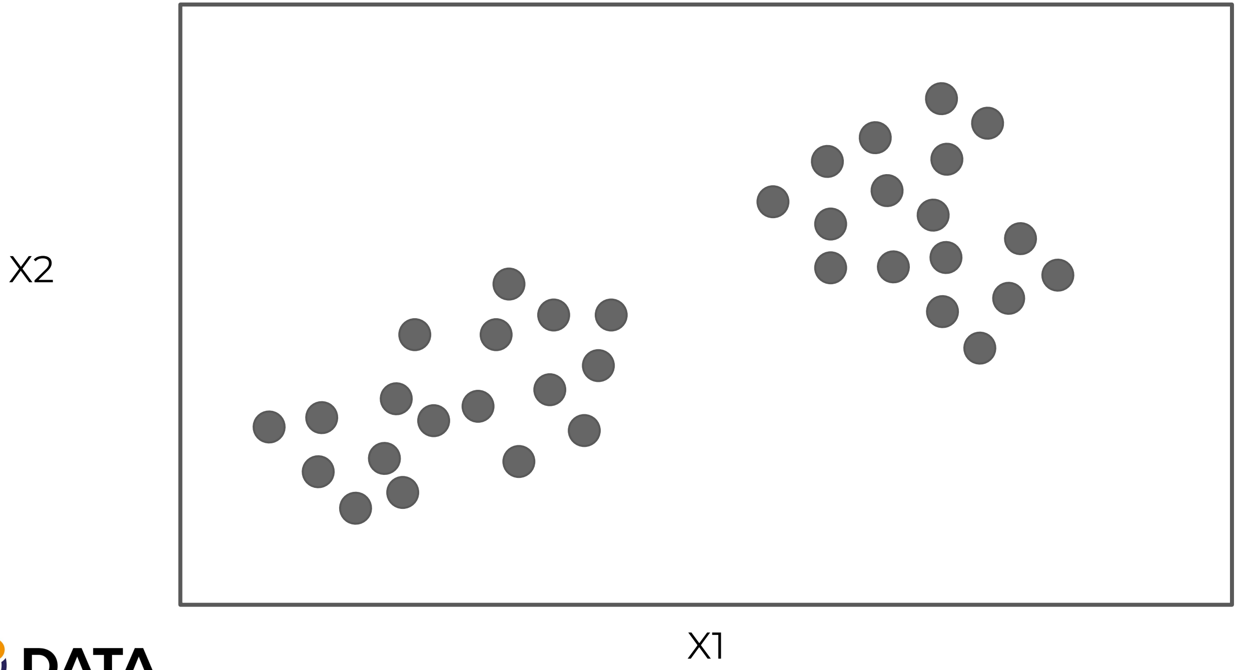
Clustering

- Could simply plot and discover patterns:



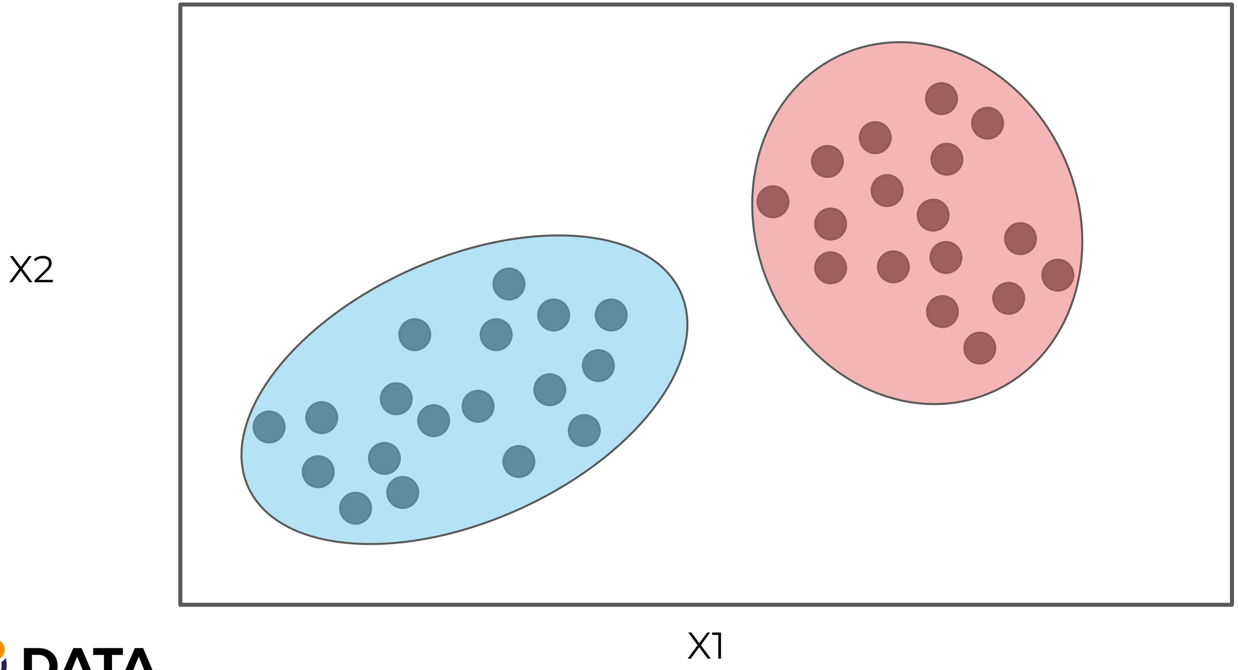
Clustering

- Here we intuitively see 2 groupings:



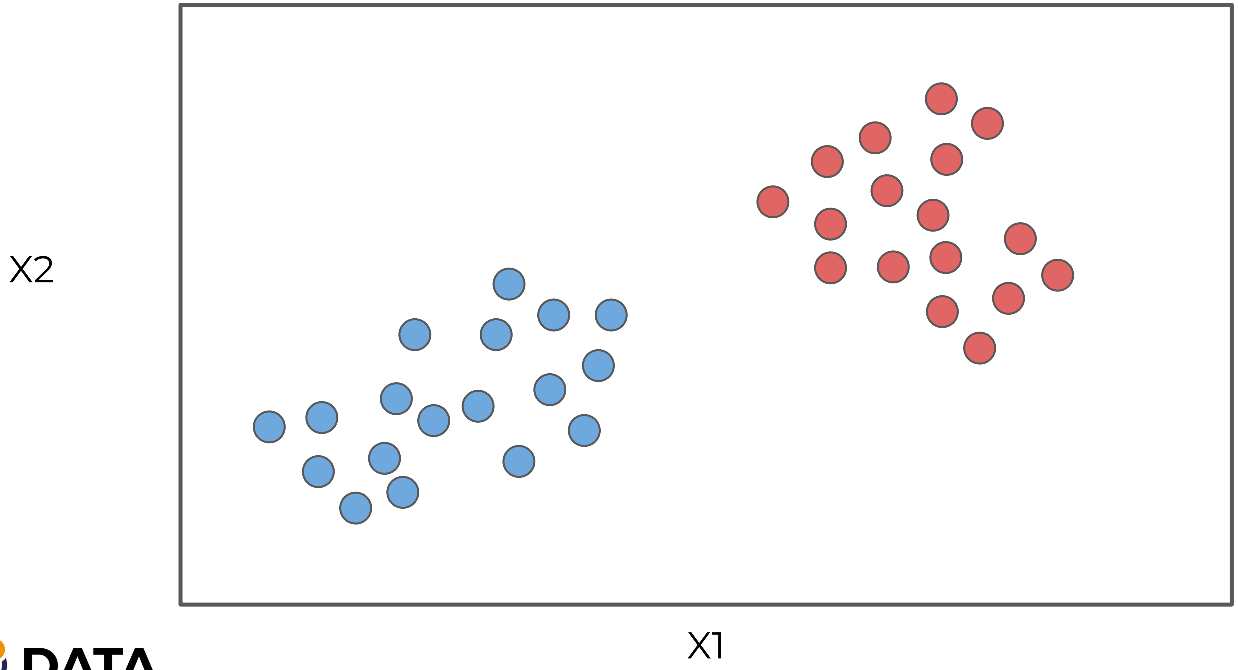
Clustering

- Note how distance is the intuitive metric:



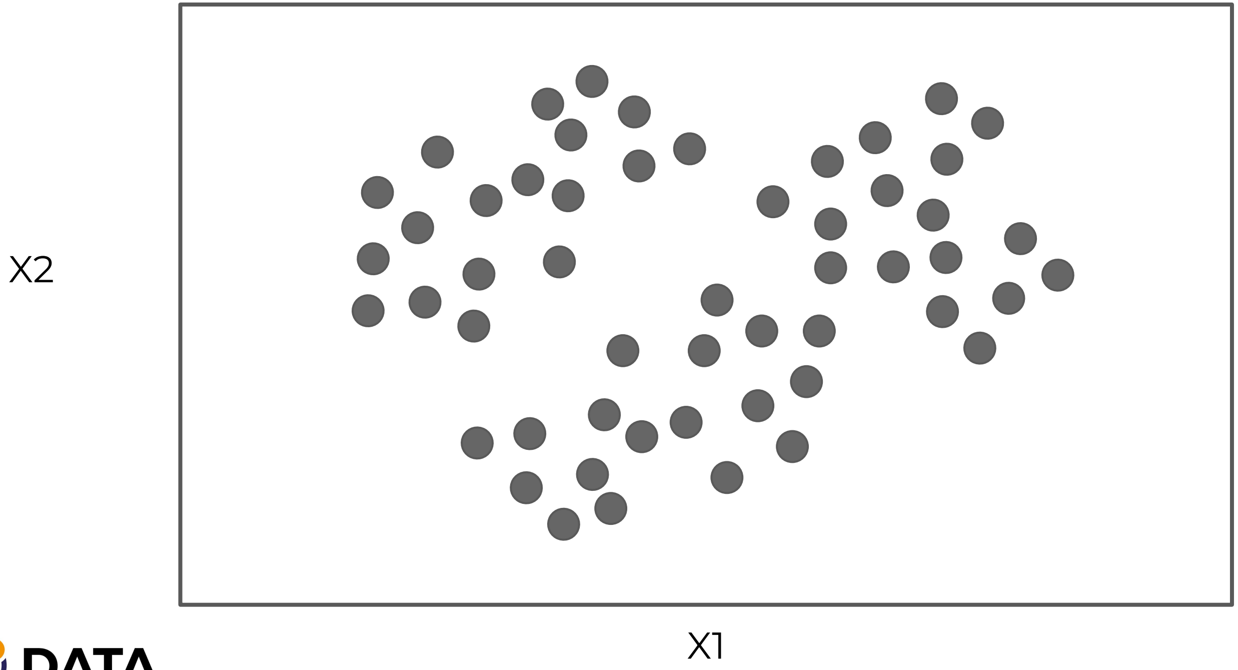
Clustering

- We could then assign clusters:



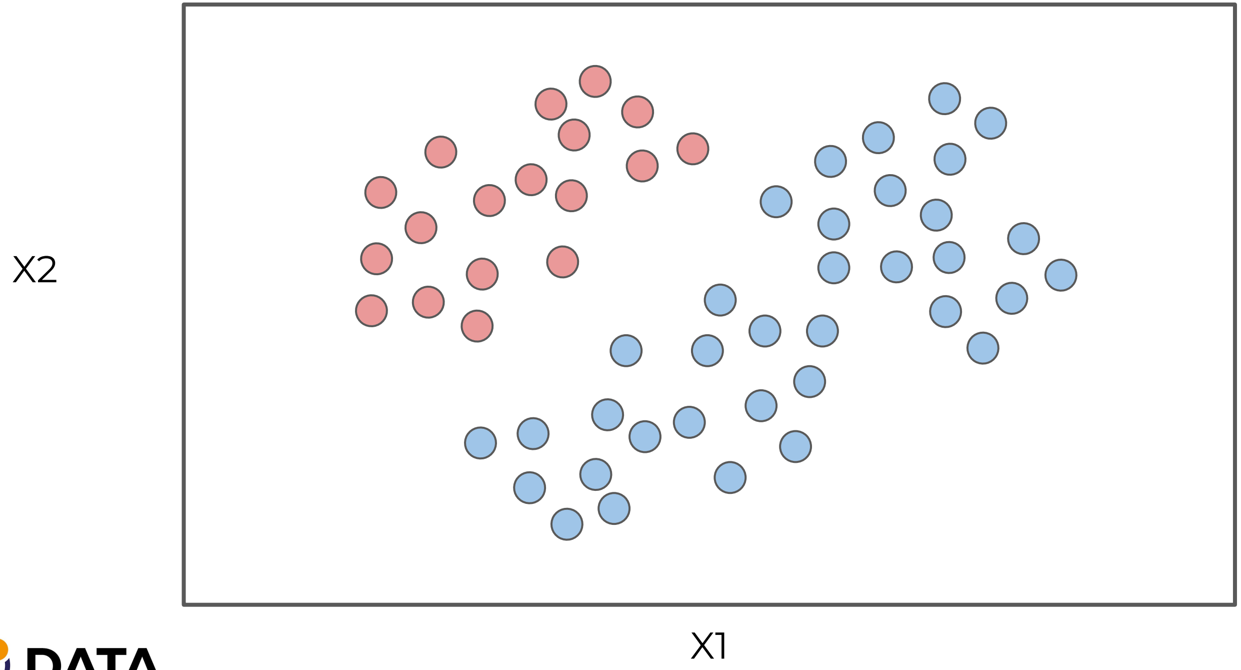
Clustering

- 2 or 3 clusters could both be reasonable:



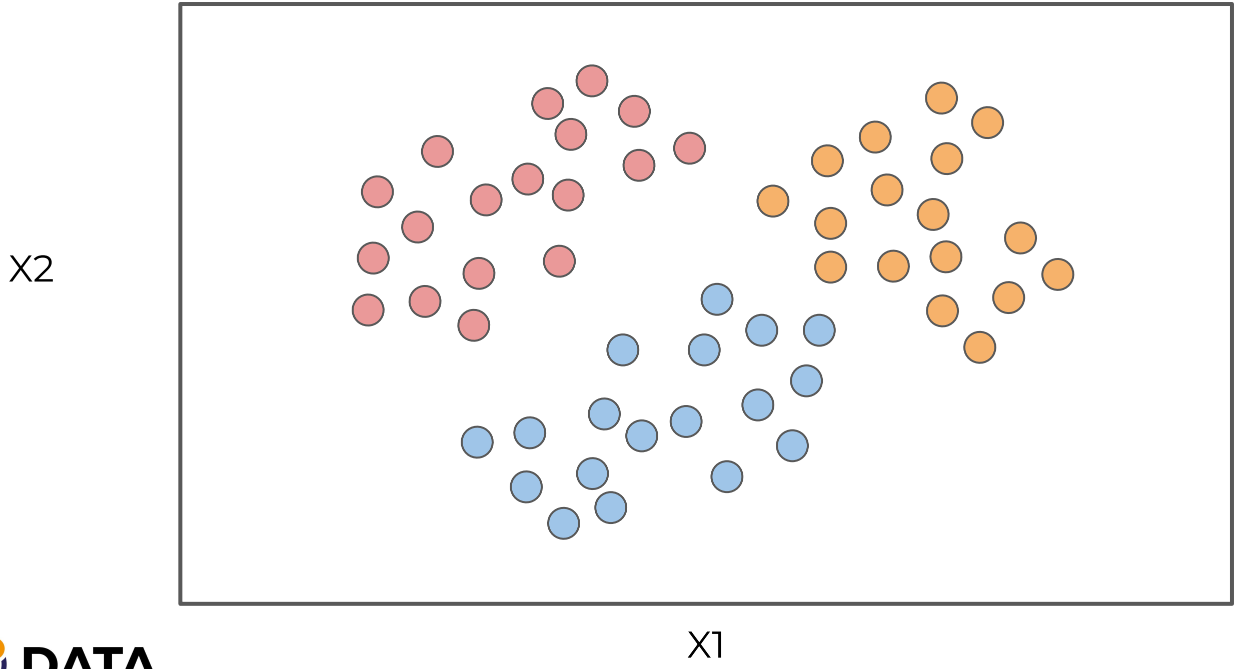
Clustering

- 2 or 3 clusters could both be reasonable:



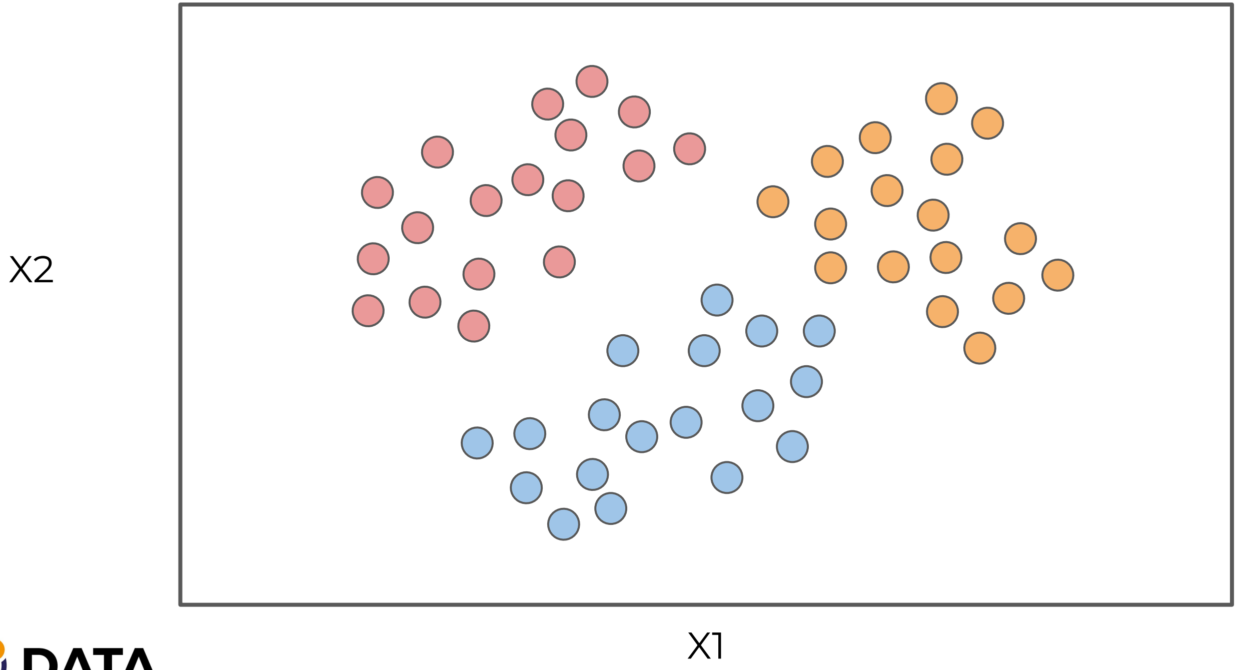
Clustering

- 2 or 3 clusters could both be reasonable:



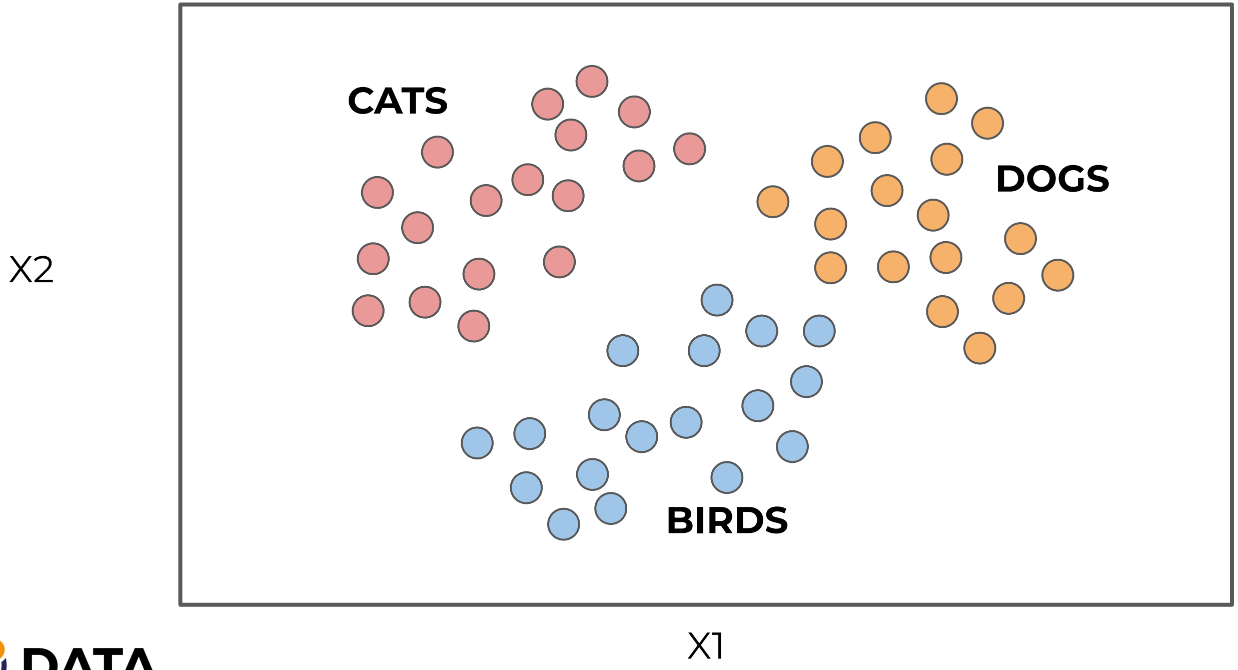
Clustering

- Different methods can be used to decide!



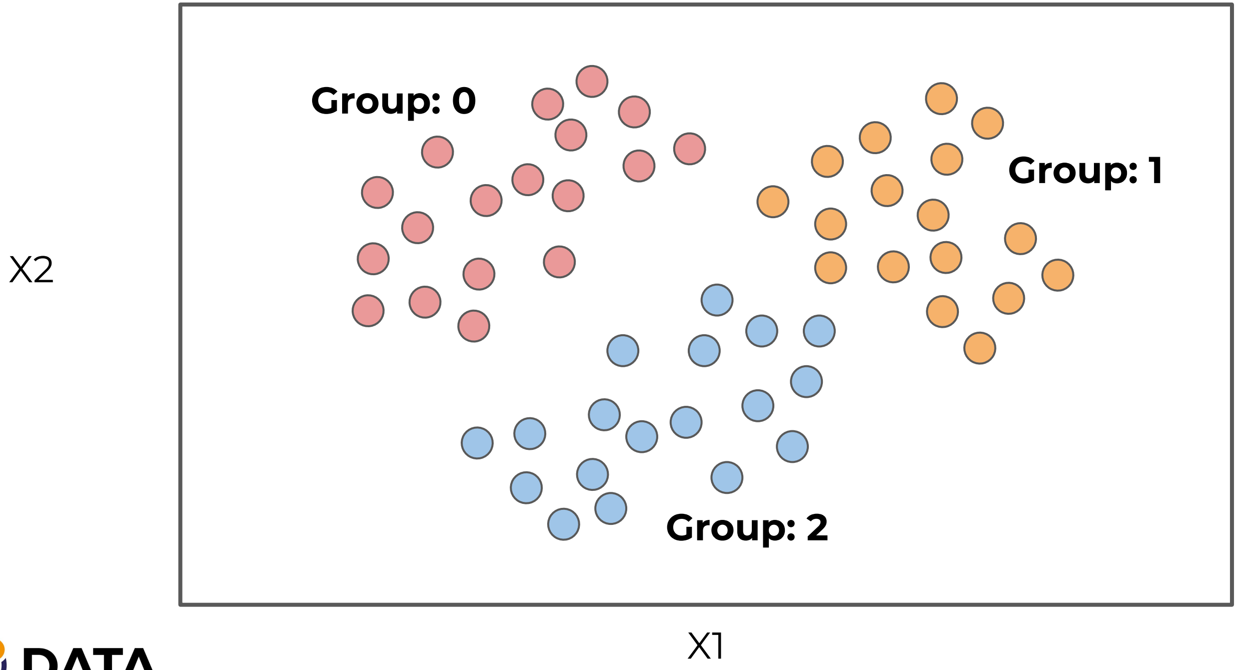
Clustering

- Clustering doesn't "label" these for you!



Clustering

- Clustering doesn't "label" these for you!



Clustering

- Unsupervised Learning Paradigm Shift:
  - *If we've discovered these new cluster labels, could we use that as a* ***y*** *for supervised training?*
    - Yes! We can use unsupervised learning to discover possible labels, then apply supervised learning on new data points.

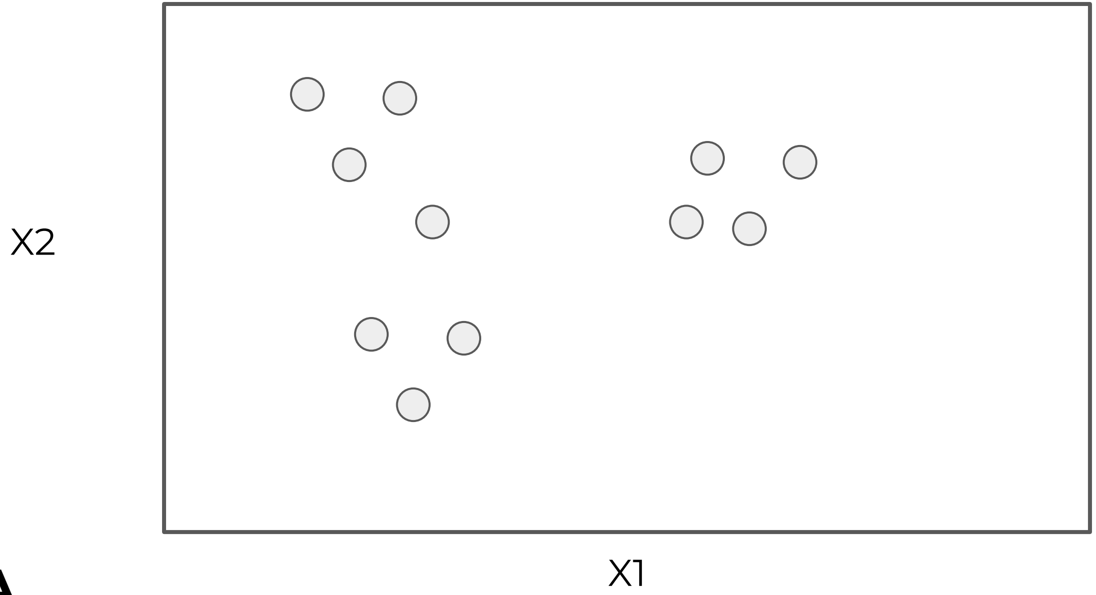
Clustering

- It’s much harder to compare unsupervised algorithms against each other due to the lack of ground truth based performance metrics (e.g. can’t use accuracy or RMSE).

K-Means Clustering

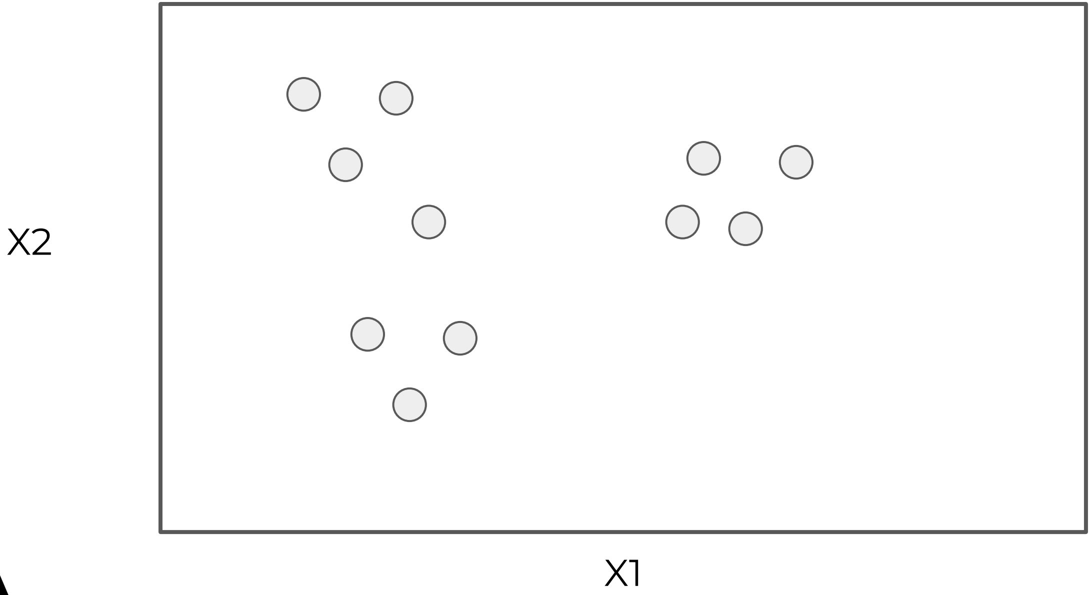
K-Means Clustering

- Step 1: Choose the number of clusters to create (this is the K value).



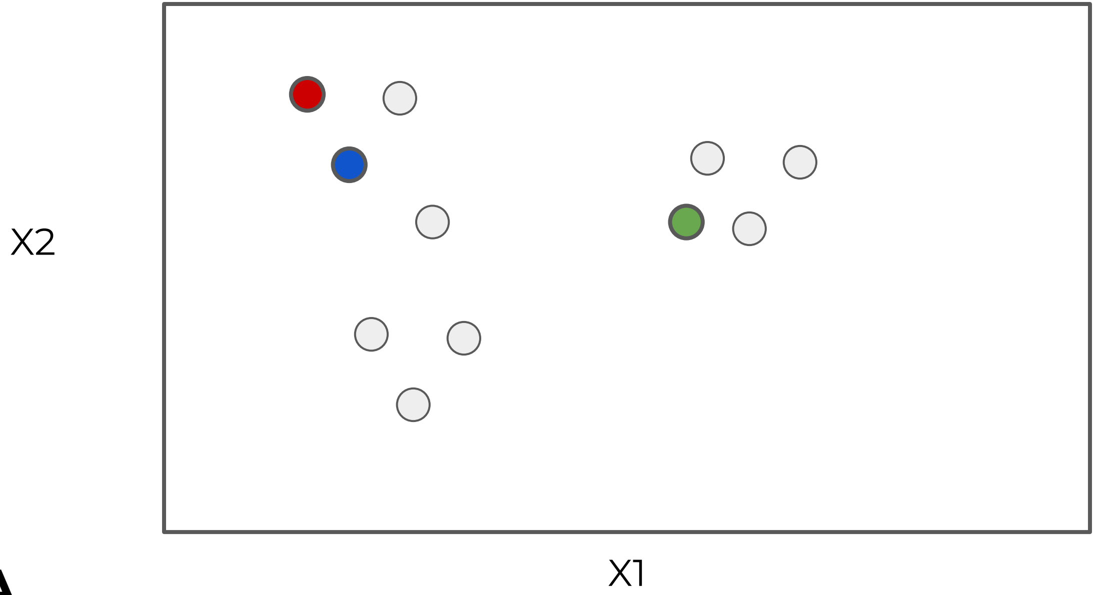
K-Means Clustering

- Step 1: We'll choose K=3. Note in most situations you won't visualize the data.



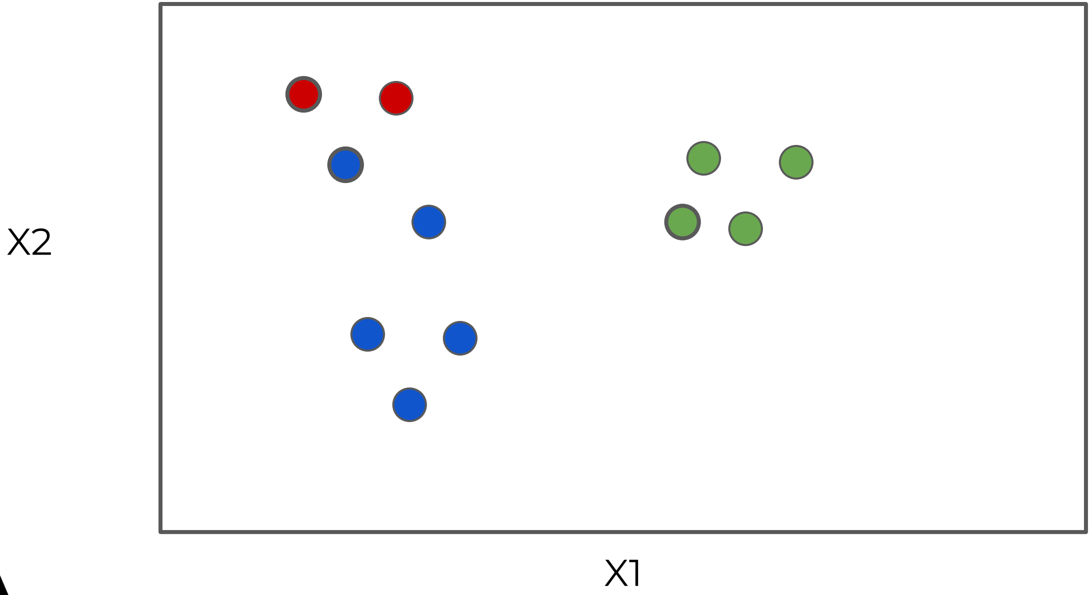
K-Means Clustering

- Step 2: Randomly select K=3 distinct data points.



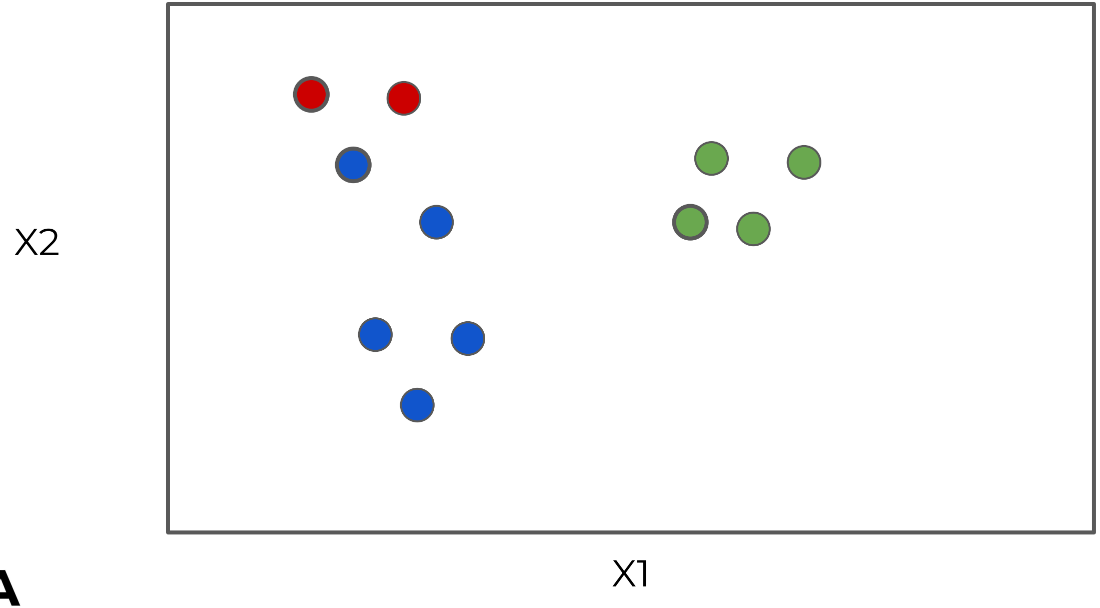
K-Means Clustering

- Step 3: Assign each remaining point to the nearest "cluster" point.



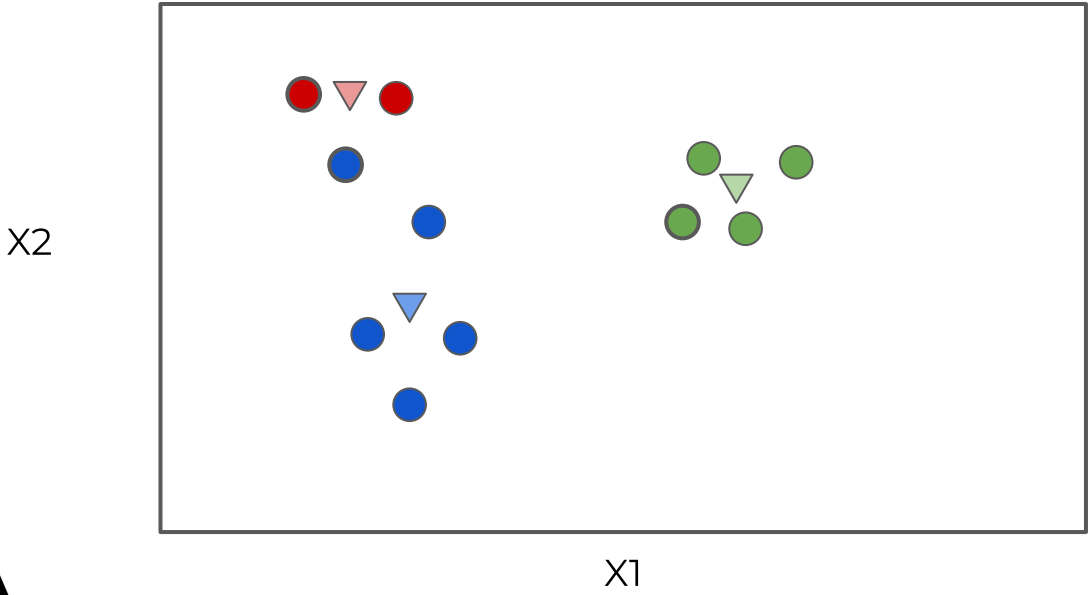
K-Means Clustering

- Step 3: Note how this is using a distance metric to judge the nearest point.



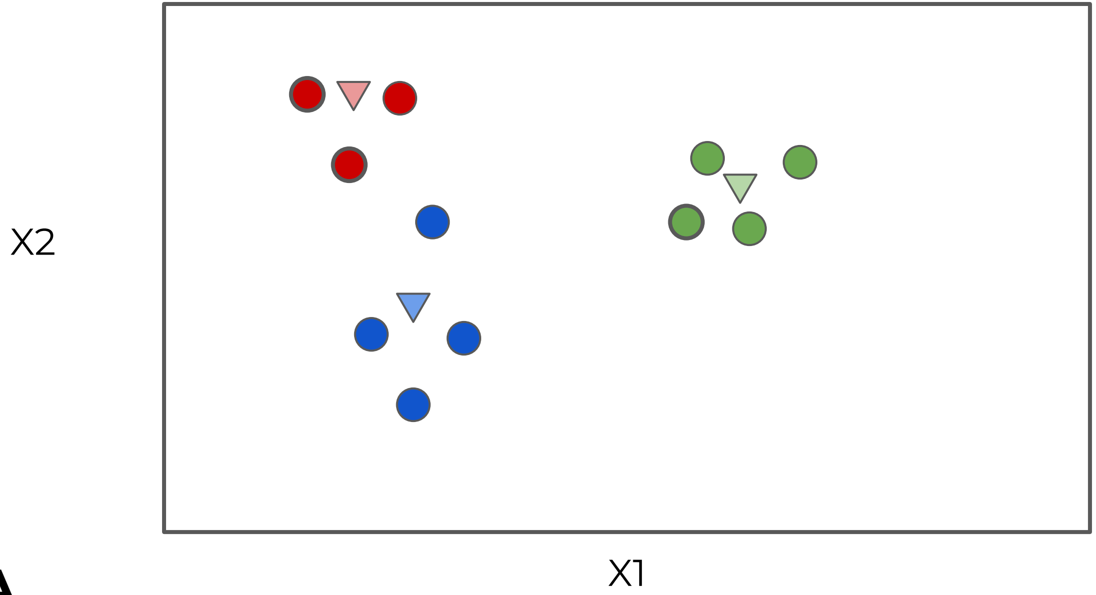
K-Means Clustering

- Step 4: Calculate the center of the cluster points (mean value of point vectors).



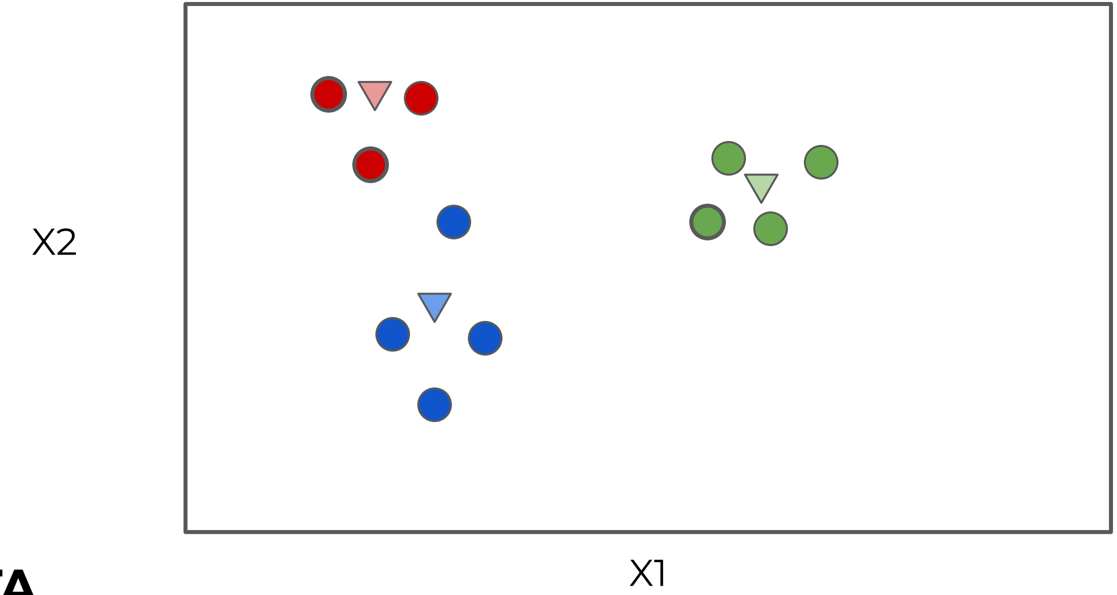
K-Means Clustering

- Step 5: Now assign each point to the nearest cluster center.



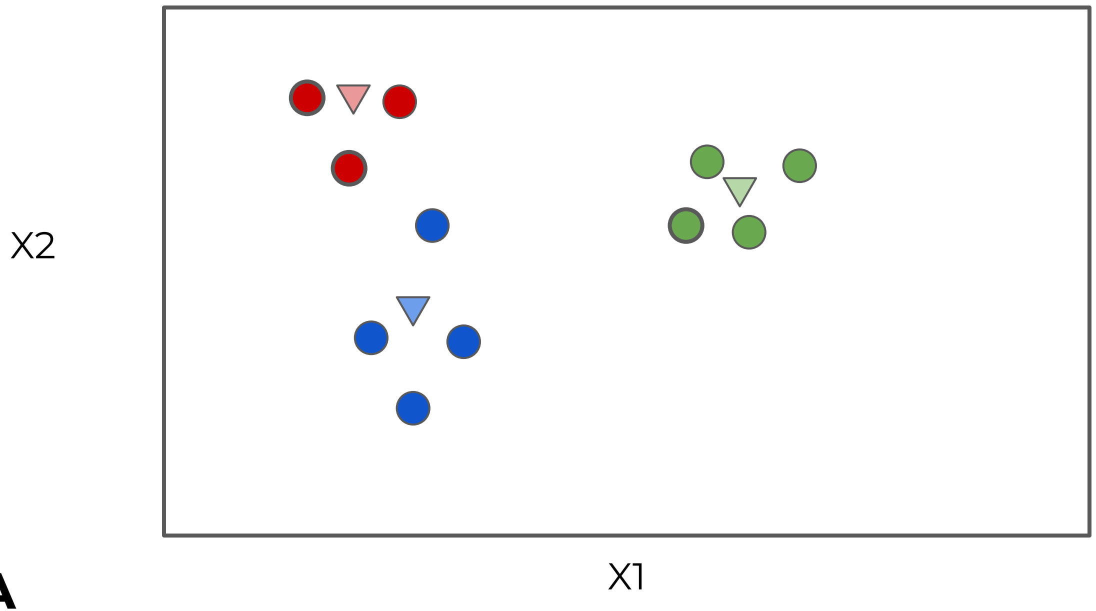
K-Means Clustering

- We repeat steps 4 and 5 until there are no more cluster reassignments.



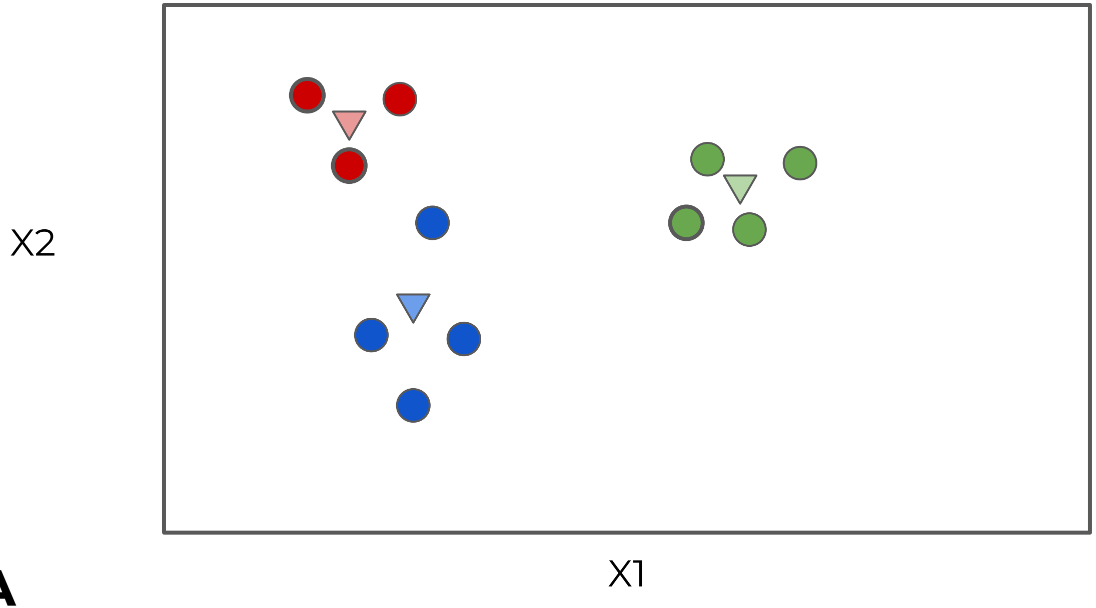
K-Means Clustering

- Step 4b: Recalculate new cluster centers:



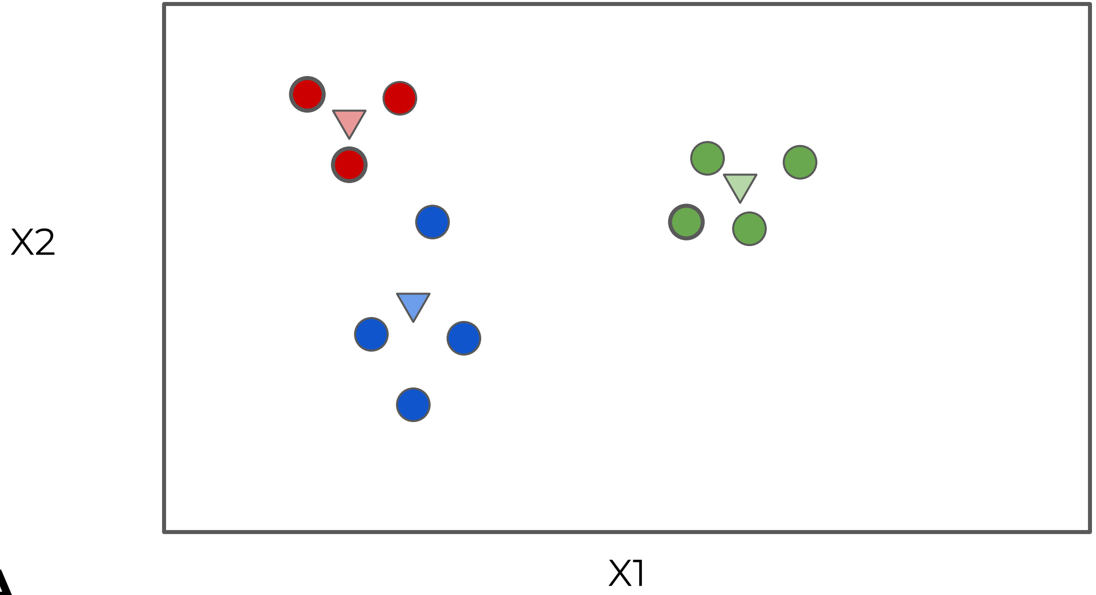
K-Means Clustering

- Step 4b: Recalculate new cluster centers:



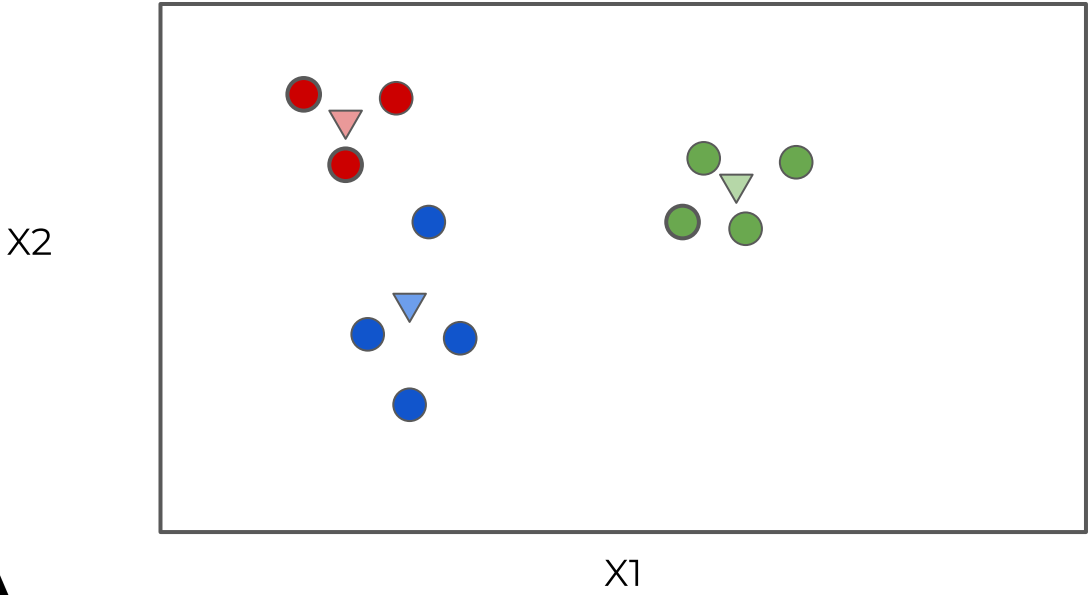
K-Means Clustering

- Step 5b: Assign points to nearest cluster center.



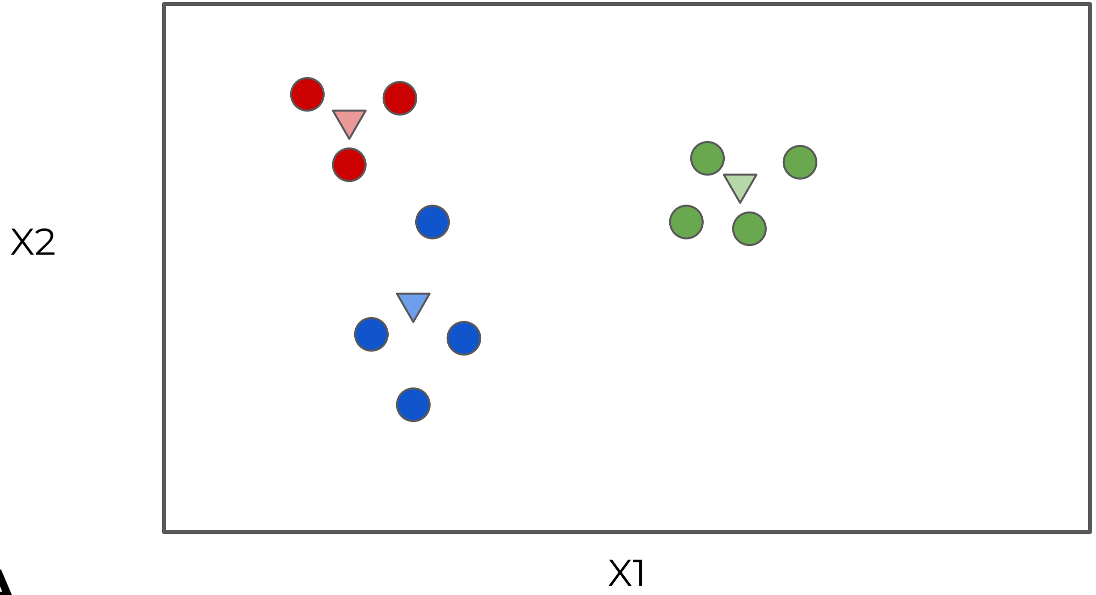
K-Means Clustering

- If there are no more reassignments, we're done! The clusters have been found.



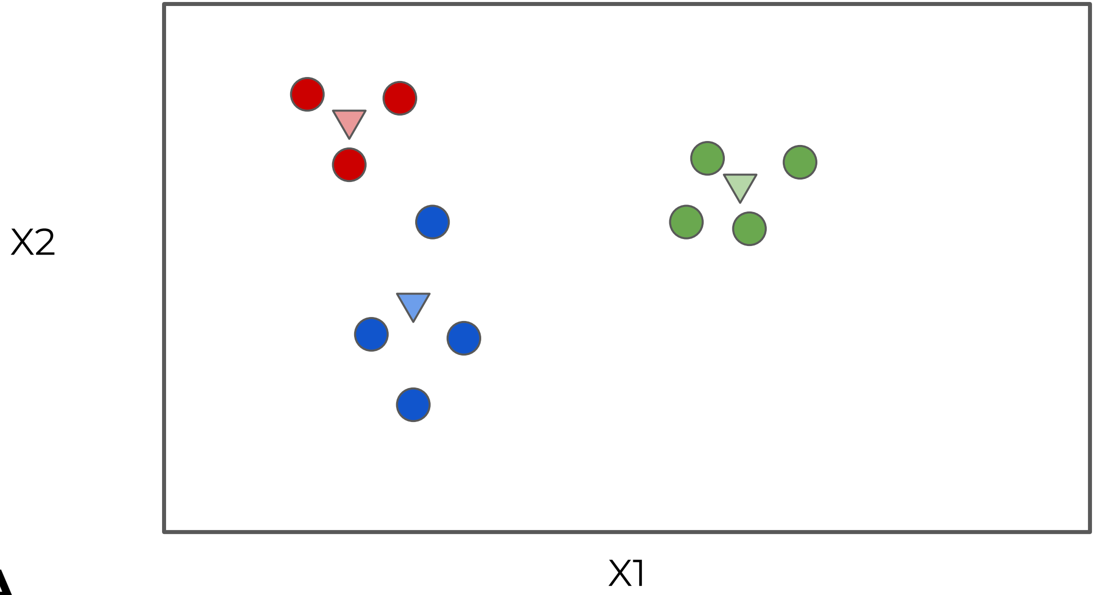
K-Means Clustering

- How can we measure “goodness of fit”?



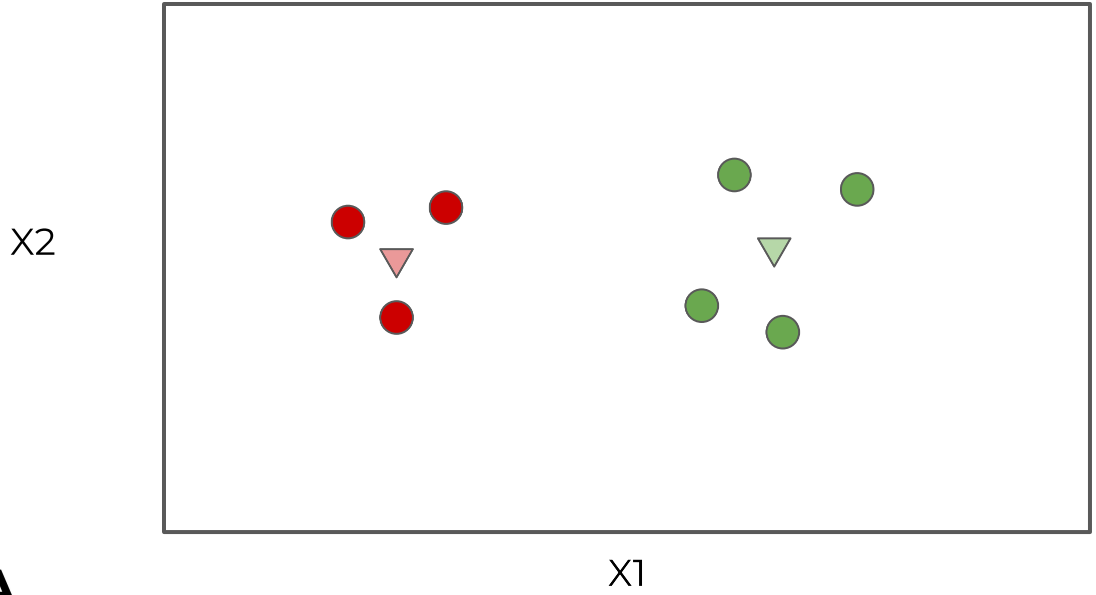
K-Means Clustering

- We could measure the sum of the distances from points to cluster centers.



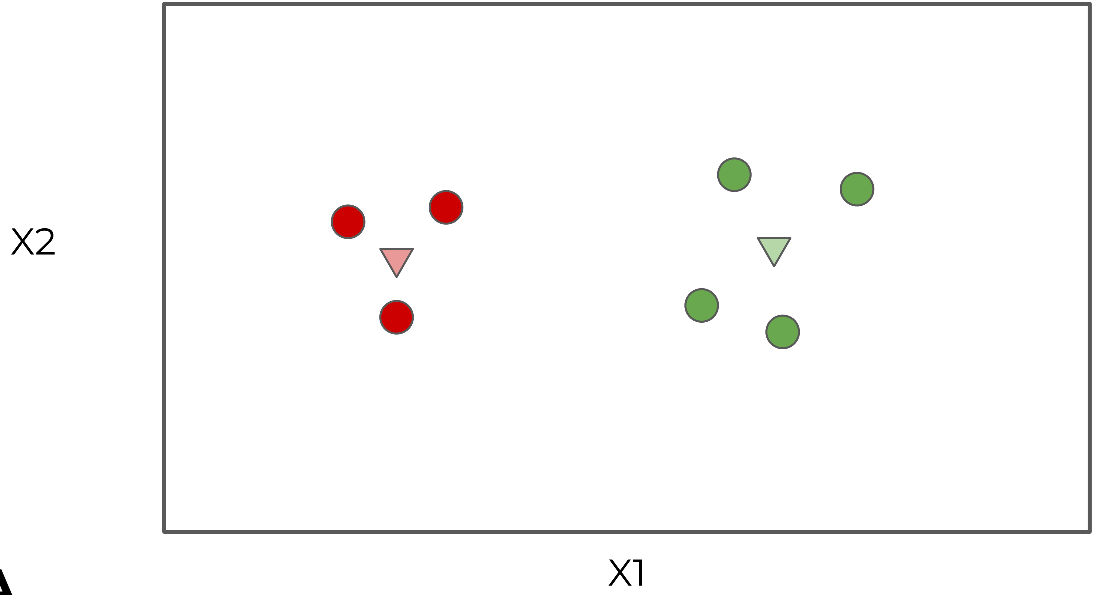
K-Means Clustering

- Imagine a simple example starting with K=2.



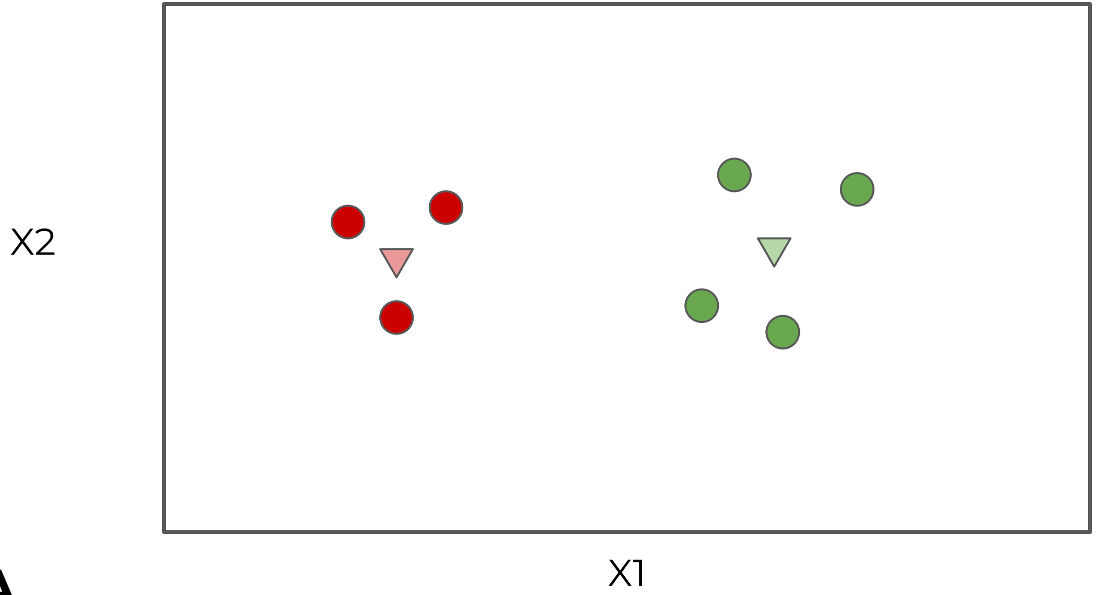
K-Means Clustering

- We measure the sum of the squared distances from points to the cluster center:



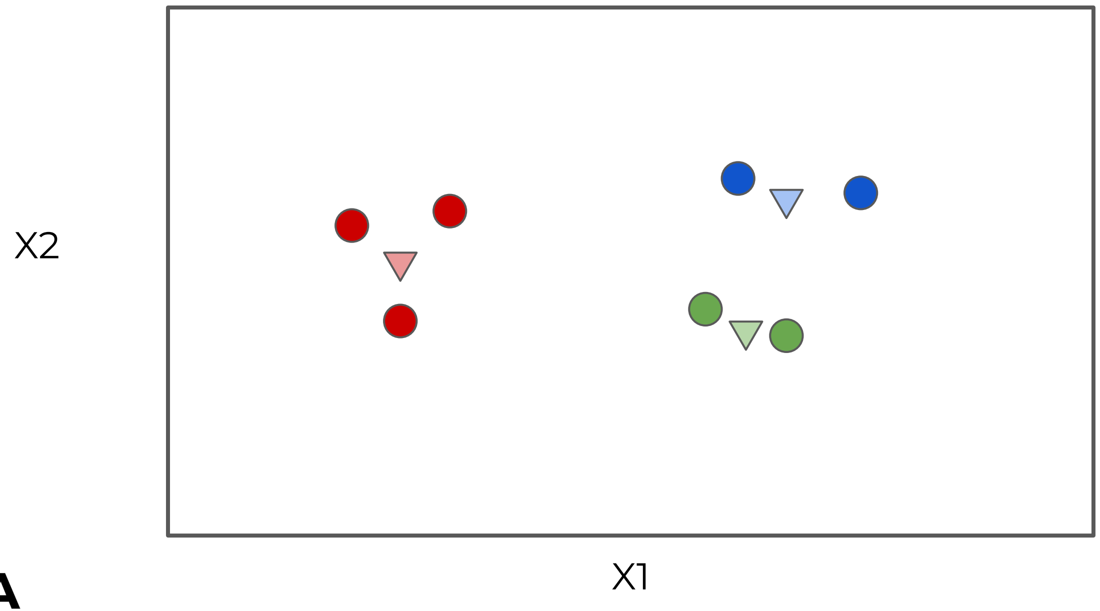
Clustering

- Then we fit an entirely new KMeans model with K+1:



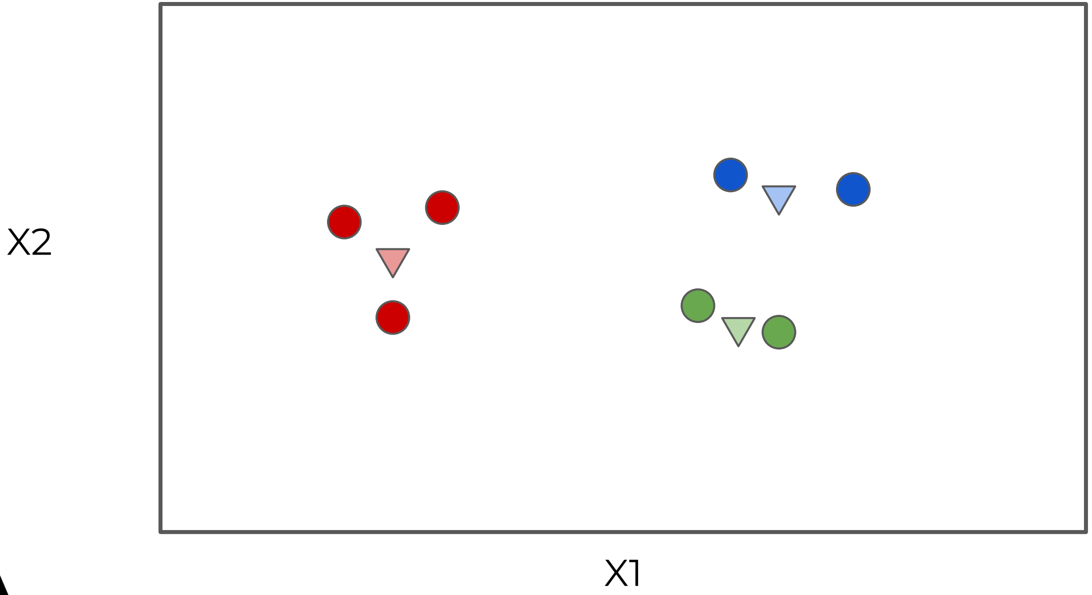
Clustering

- Then we fit an entirely new KMeans model with K+1:



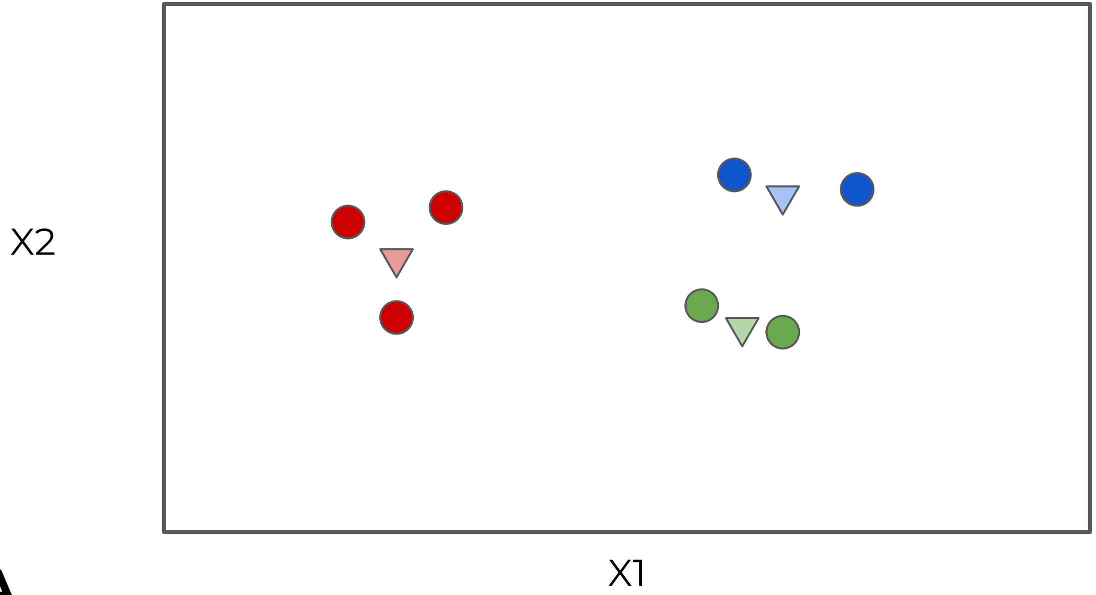
Clustering

- Then measure again the sum of the squared distance (SSD) to center.



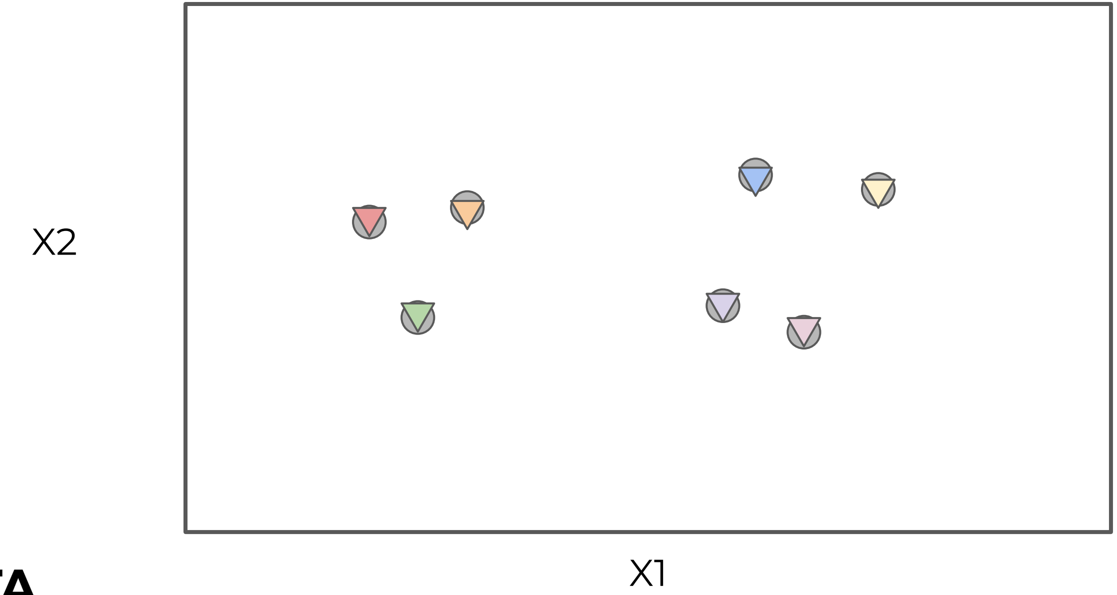
Clustering

- In theory this SSD would go to zero once K is equal to the number of points.



Clustering

- You would have a cluster for each point! SSD would be perfect at 0!



Clustering

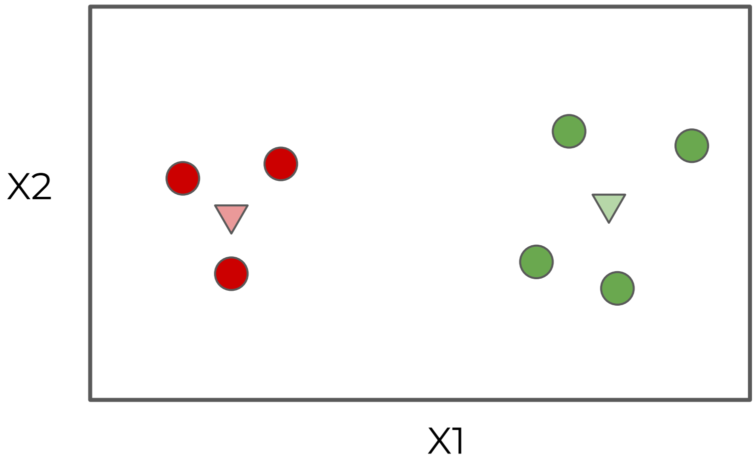
- We keep track of this SSD value for a range of different K values.
- We then look for a K value where **rate of reduction in SSD** begins to decline.
- This signifies that adding an extra cluster is **not** obtaining enough clarity of cluster separation to justify increasing K.

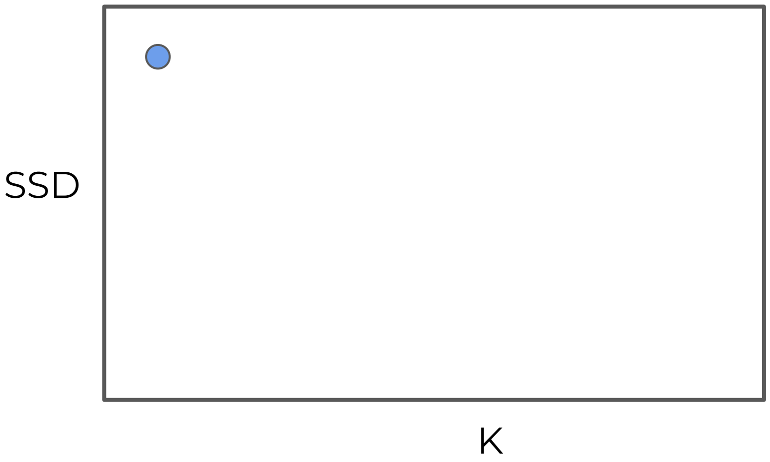
Clustering

- This is known as the “elbow” method since we will track where decrease in SSD begins to flatten out compared to increasing K values.
- Let’s walk through what this chart would look like...

Clustering

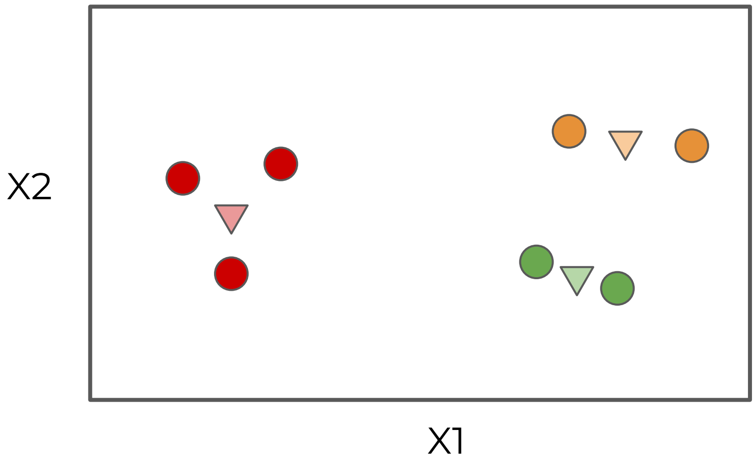
- Start with K=2:

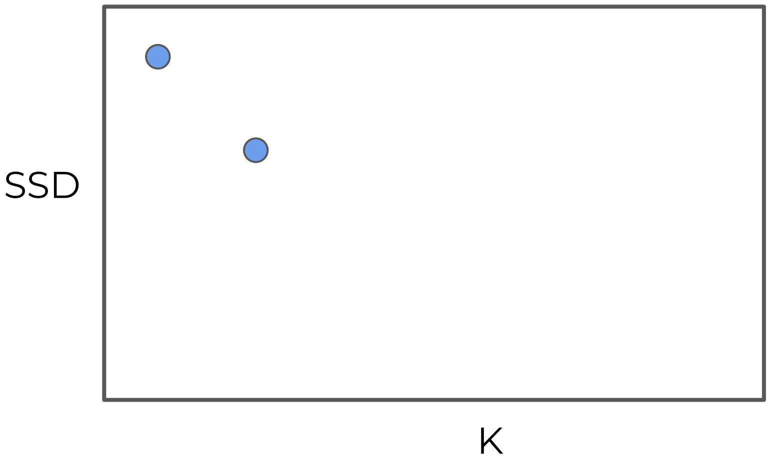




Clustering

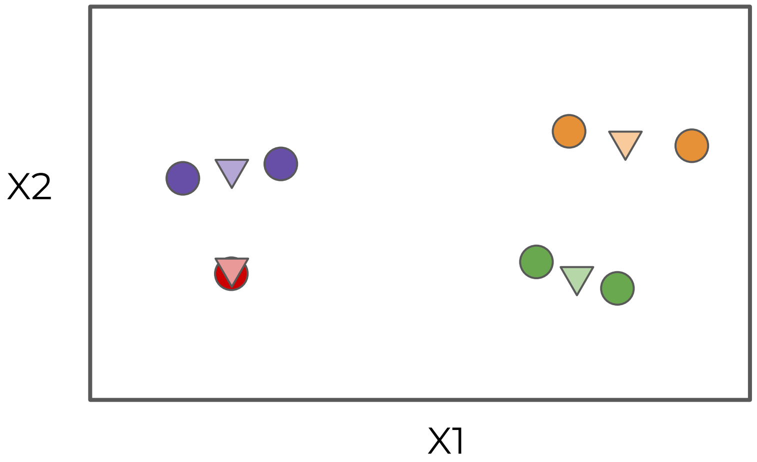
- Increase K and measure SSD:

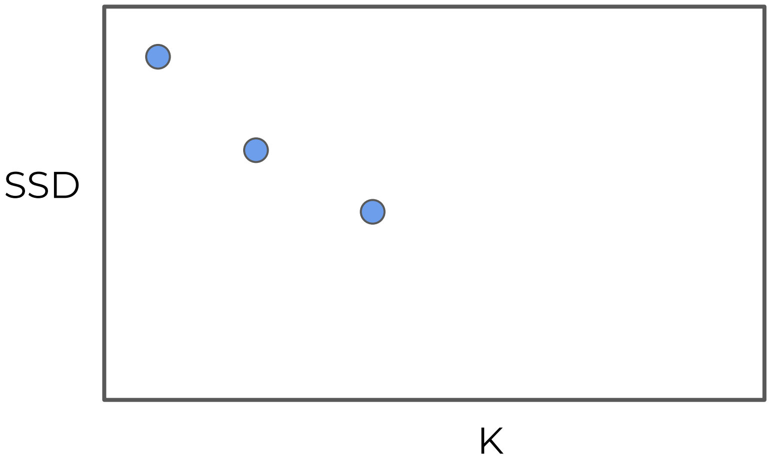




Clustering

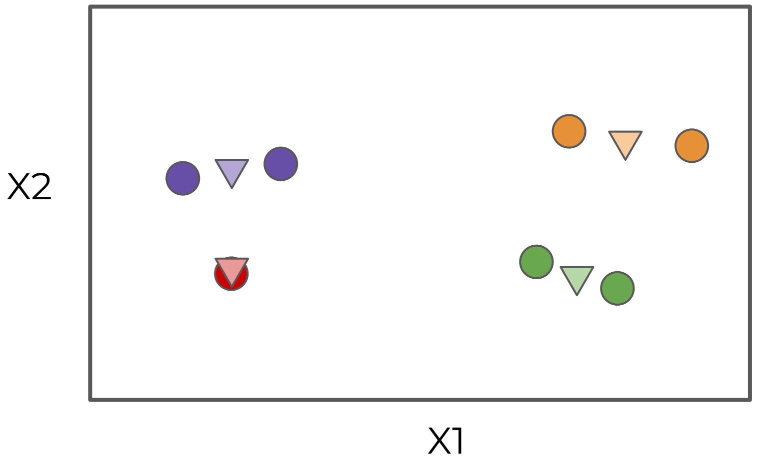
- Increase K and measure SSD:

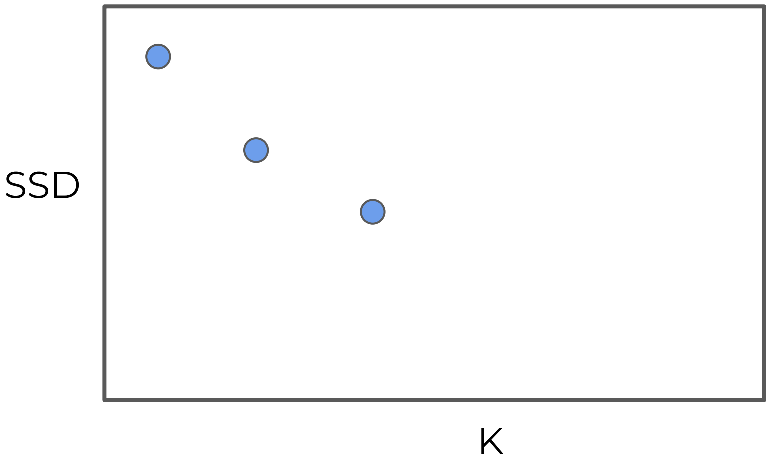




Clustering

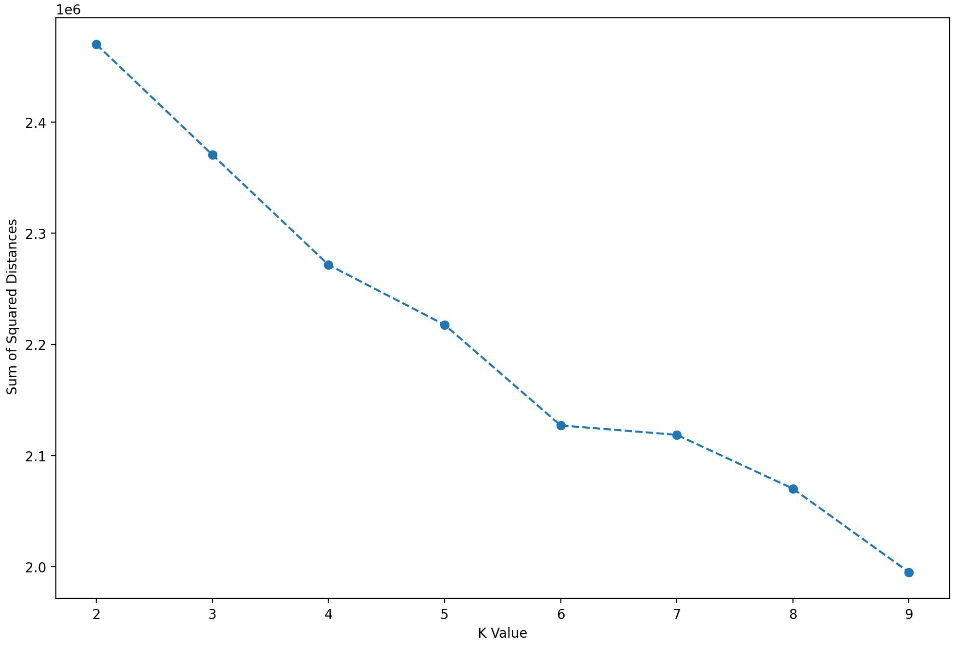
- Repeat this process for some set number of K values:





K Means Clustering

- You will see a continuous decline.



PIERIAN DATA

K Means Clustering

- Eventually you will see “elbow” points:
- These points are strong indicators that increasing K further is no longer justified as it is not revealing more “signal”.