# [320] Unsupervised ML Recap

## ????

| A | B | C | D |
|---|---|---|---|
| 50 | 50 | 110 | 90 |
| ... | ... | ... | ... |

*...hundreds of rows...*

## Tree



## ????

| A | B | C | D |
|---|---|---|---|
| 50 | 50 | 110 | 90 |
| ... | ... | ... | ... |

*...hundreds of rows...*

## Centroids

| A | B | C | D |
|---|---|---|---|
| 250 | 0 | 15 | 12 |
| 45 | 45 | 100 | 102 |
| 0 | 95 | 24 | 80 |

## ????

| A | B | C | D |
|---|---|---|---|
| 50 | 50 | 110 | 90 |
| ... | ... | ... | ... |

*...hundreds of rows...*

## Component Rows

50

10
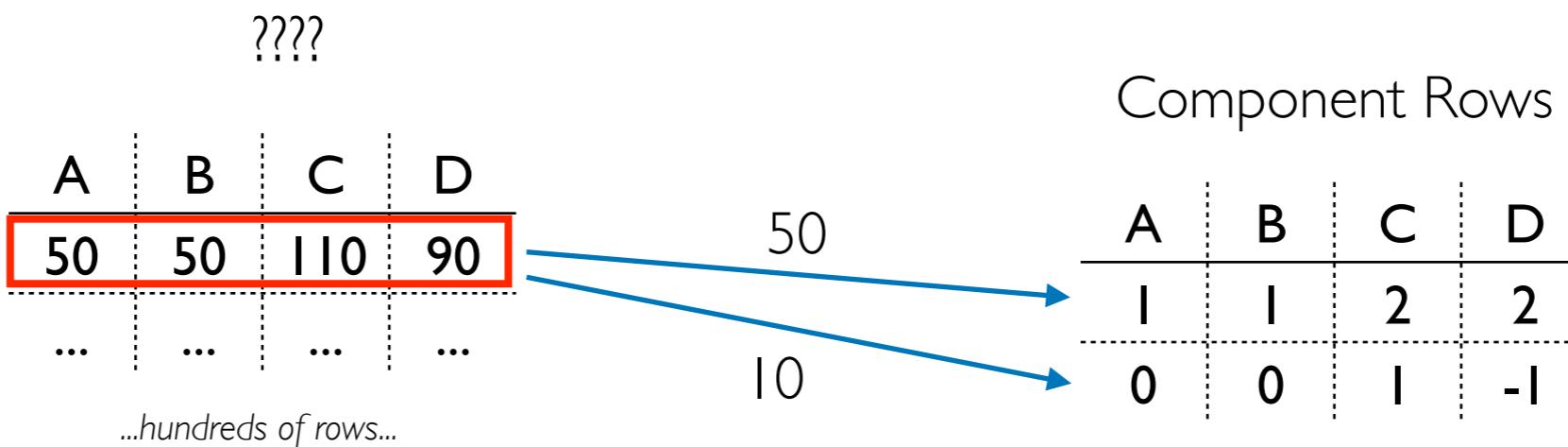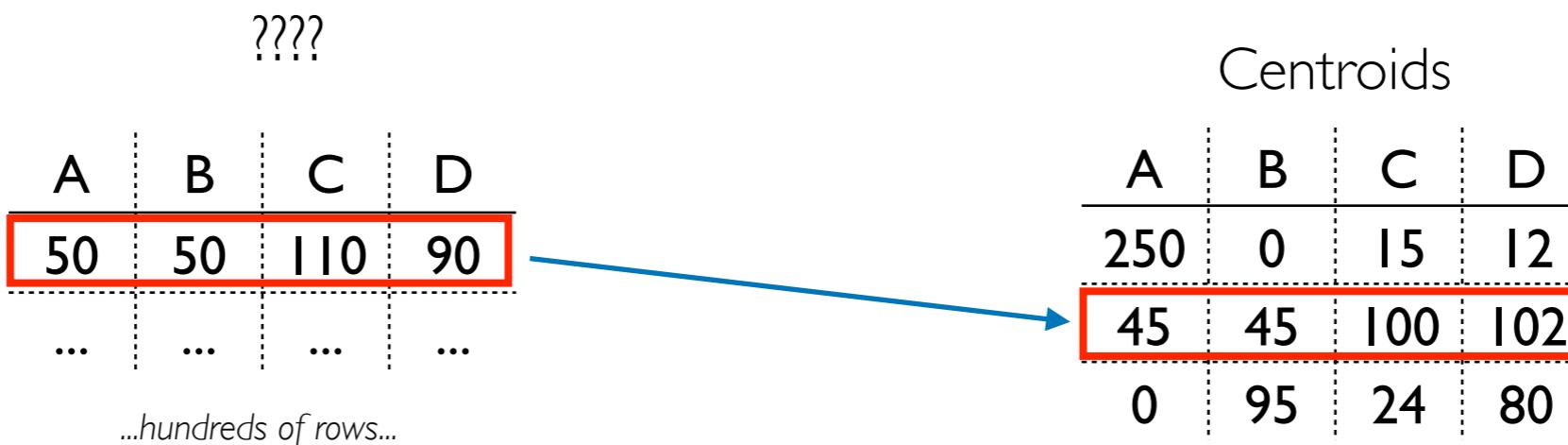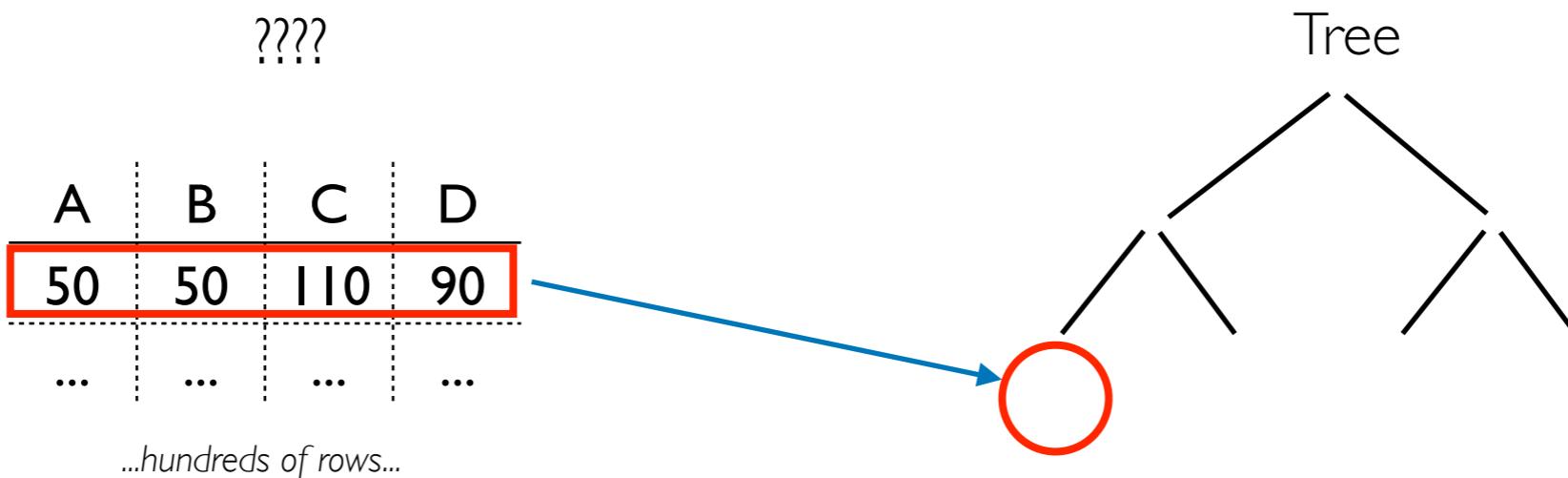
| A | B | C | D |
|---|---|---|---|
| 1 | 1 | 2 | 2 |
| 0 | 0 | 1 | -1 |

# Hierarchical Clustering
(for example, AgglomerativeClustering)

|  A  |  B  |  C  |  D  |
| --- | --- | --- | --- |
| 50  | 50  | 110 | 90  |
| ... | ... | ... | ... |

*...hundreds of rows...*

## Tree
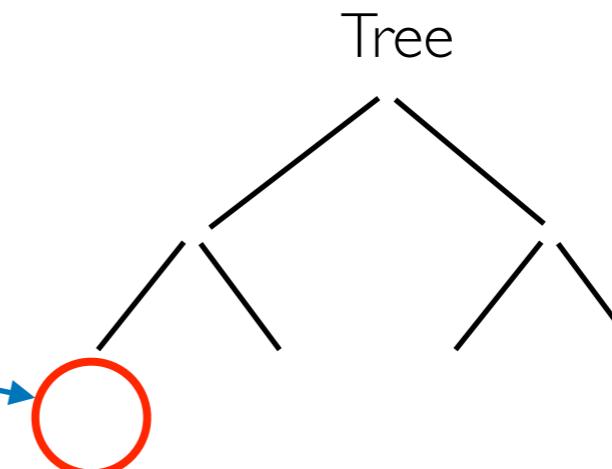


# Non-Hierarchical Clustering
(for example, KMeans)

|  A  |  B  |  C  |  D  |
| --- | --- | --- | --- |
| 50  | 50  | 110 | 90  |
| ... | ... | ... | ... |

*...hundreds of rows...*

## Centroids

|  A  |  B  |  C  |  D  |
| --- | --- | --- | --- |
| 250 |  0  | 15  | 12  |
| 45  | 45  | 100 | 102 |
|  0  | 95  | 24  | 80  |

# Decomposition
(for example, PCA)

|  A  |  B  |  C  |  D  |
| --- | --- | --- | --- |
| 50  | 50  | 110 | 90  |
| ... | ... | ... | ... |

*...hundreds of rows...*

## Component Rows

|  A  |  B  |  C  |  D  |
| --- | --- | --- | --- |
|  1  |  1  |  2  |  2  |
|  0  |  0  |  1  | -1  |

50

10

# Hierarchical Clustering
(for example, AgglomerativeClustering)

## Tree

| A | B | C | D |
|---|---|---|---|
| 50 | 50 | 110 | 90 |
| ... | ... | ... | ... |
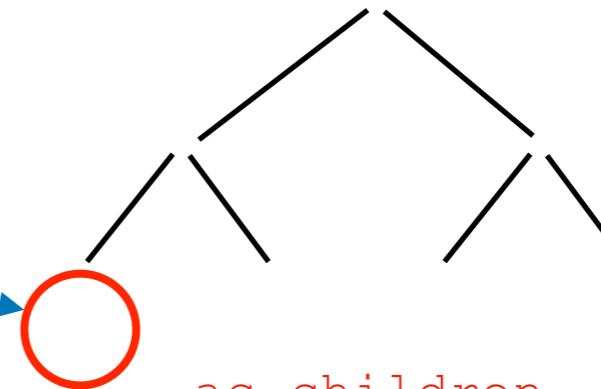
*...hundreds of rows...*

`ac.children_`

# Non-Hierarchical Clustering
(for example, KMeans)

## Centroids

| A | B | C | D |
|---|---|---|---|
| 50 | 50 | 110 | 90 |
| ... | ... | ... | ... |

*...hundreds of rows...*

| A | B | C | D |
|---|---|---|---|
| 250 | 0 | 15 | 12 |
| 45 | 45 | 100 | 102 |
| 0 | 95 | 24 | 80 |

`km.cluster_centers_`

# Decomposition
(for example, PCA)

## Component Rows

| A | B | C | D |
|---|---|---|---|
| 50 | 50 | 110 | 90 |
| ... | ... | ... | ... |

*...hundreds of rows...*

50

10

| A | B | C | D |
|---|---|---|---|
| 1 | 1 | 2 | 2 |
| 0 | 0 | 1 | -1 |

`pca.components_`

# Preprocessing: Clustering or Decomposition?
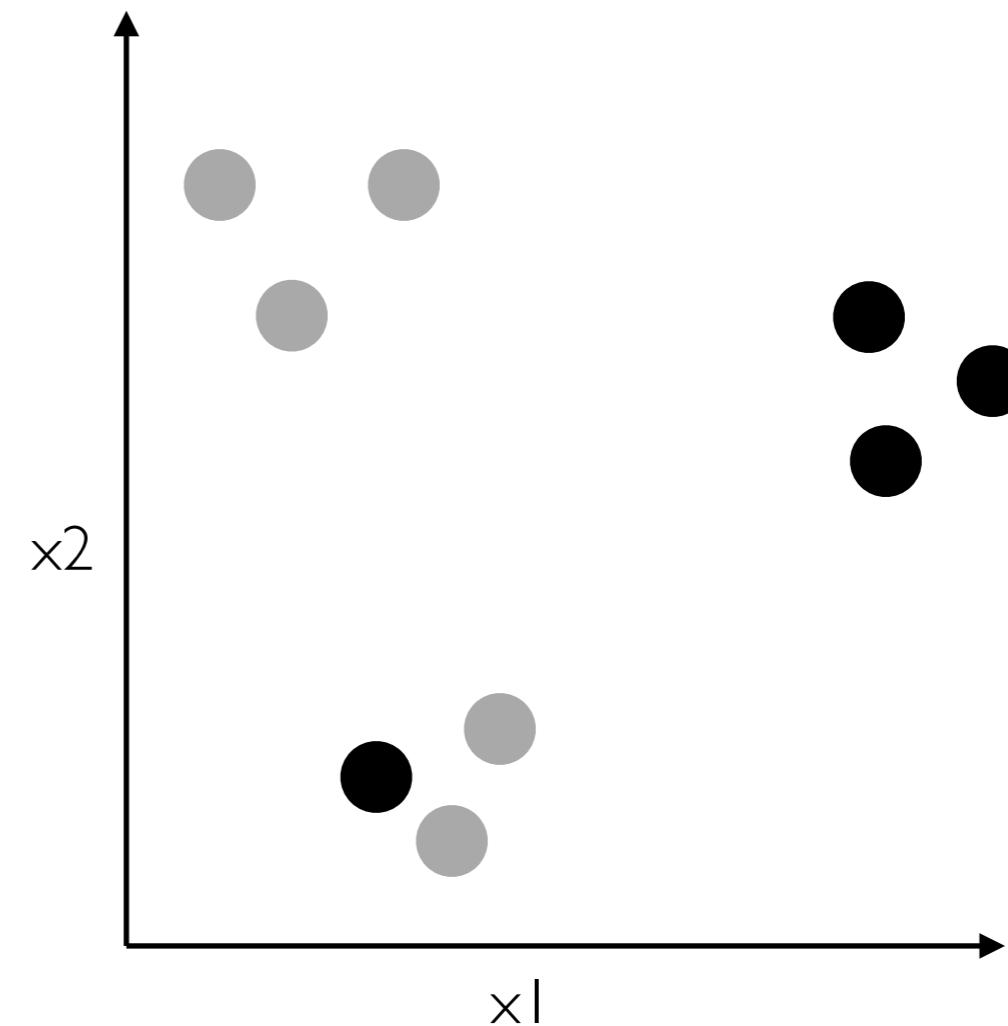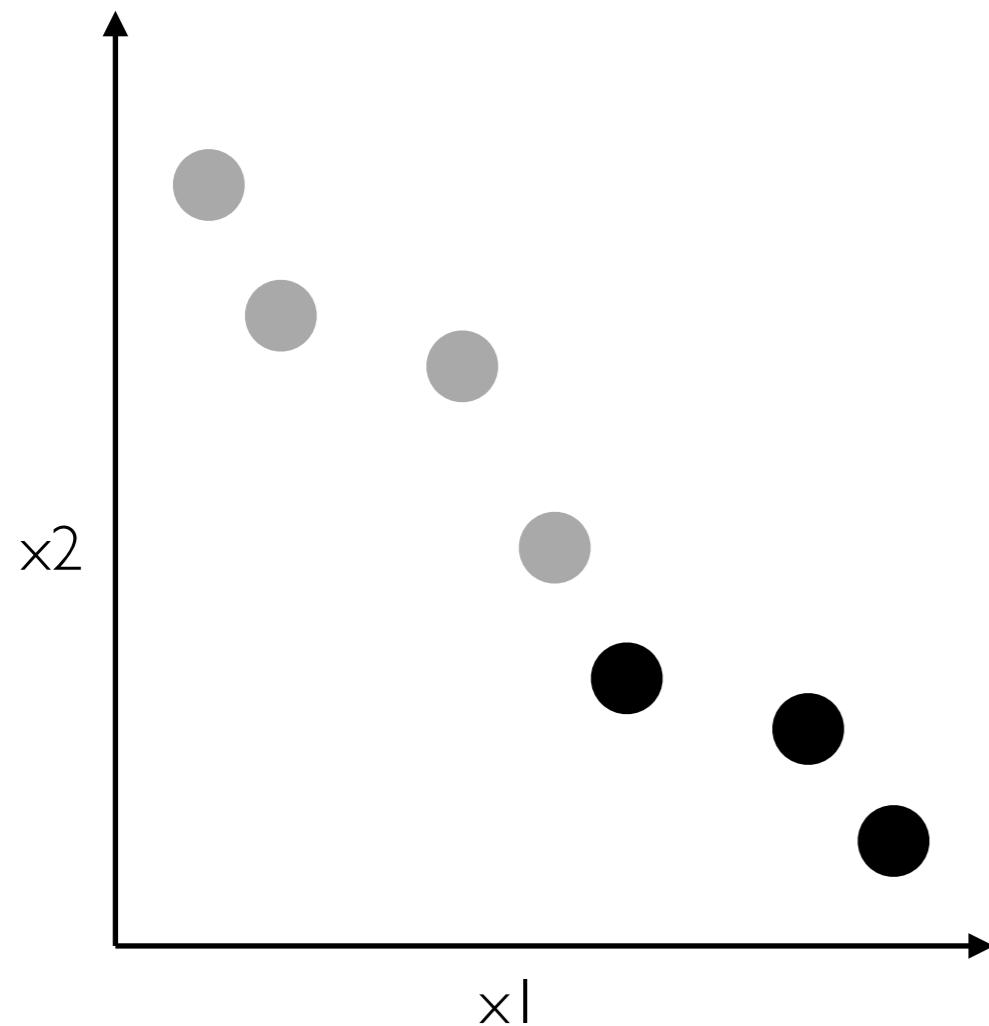


| x1 | x2 | y |
|----|----|----|
| 10 | 5 | TRUE |
| ... | ... | ... |

```
model = Pipeline([
    ????,
    ("lr", LogisticRegression())
])
```

# Preprocessing: Clustering or Decomposition?

PCA(1)

pc0

x2

x1

KMeans(3)

x2

x1
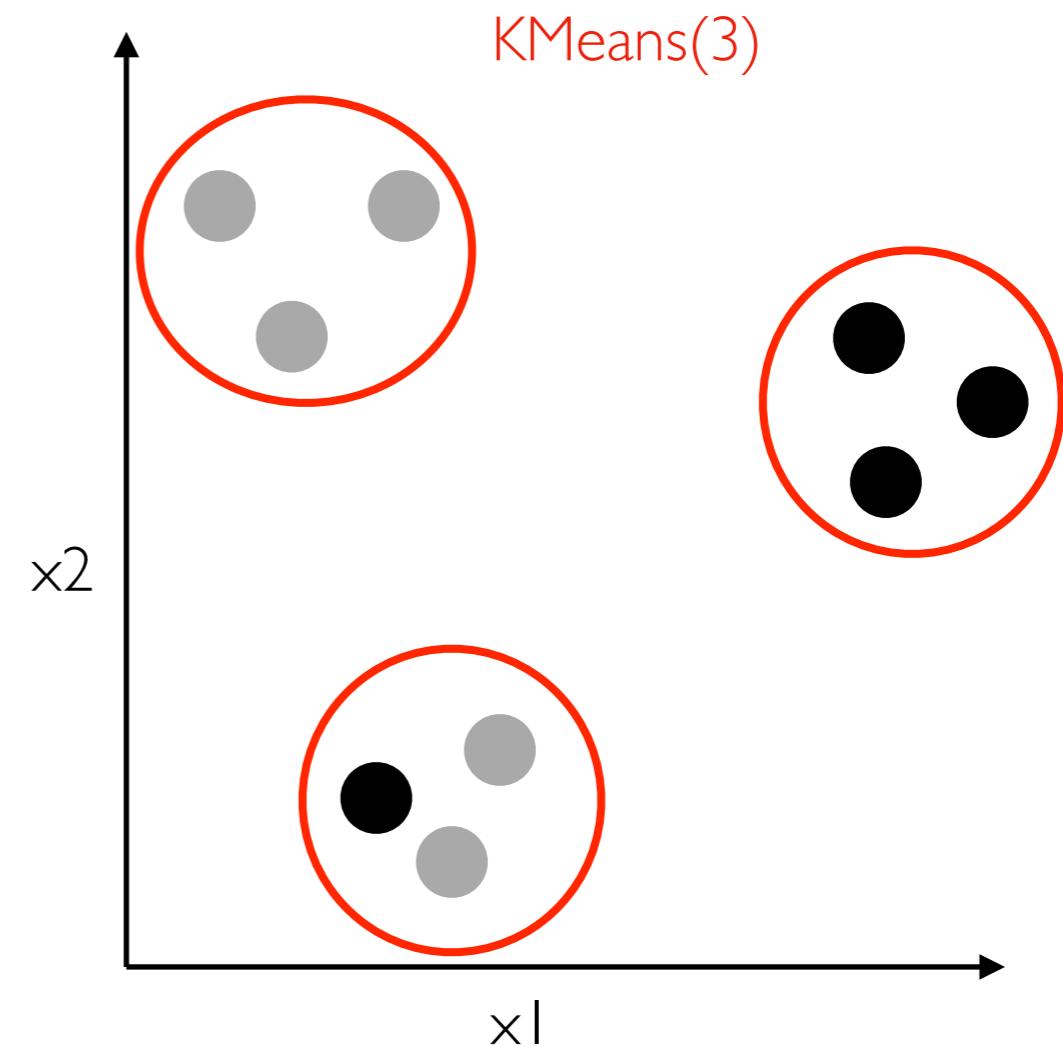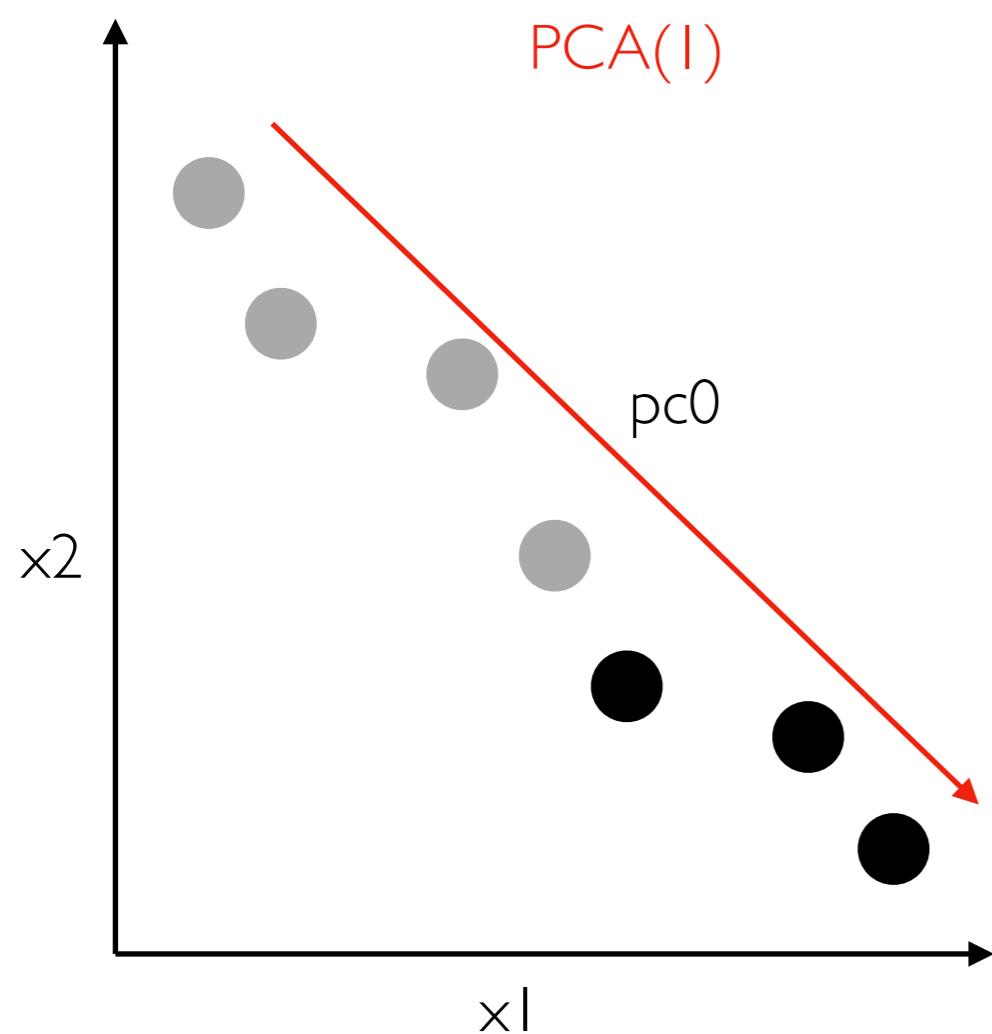
| x1 | x2 | y |
|----|----|-----|
| 10 | 5 | TRUE |
| ... | ... | ... |

```
model = Pipeline([
    ????,
    ("lr", LogisticRegression())
])
```

KMeans or Agglomerative Clustering?
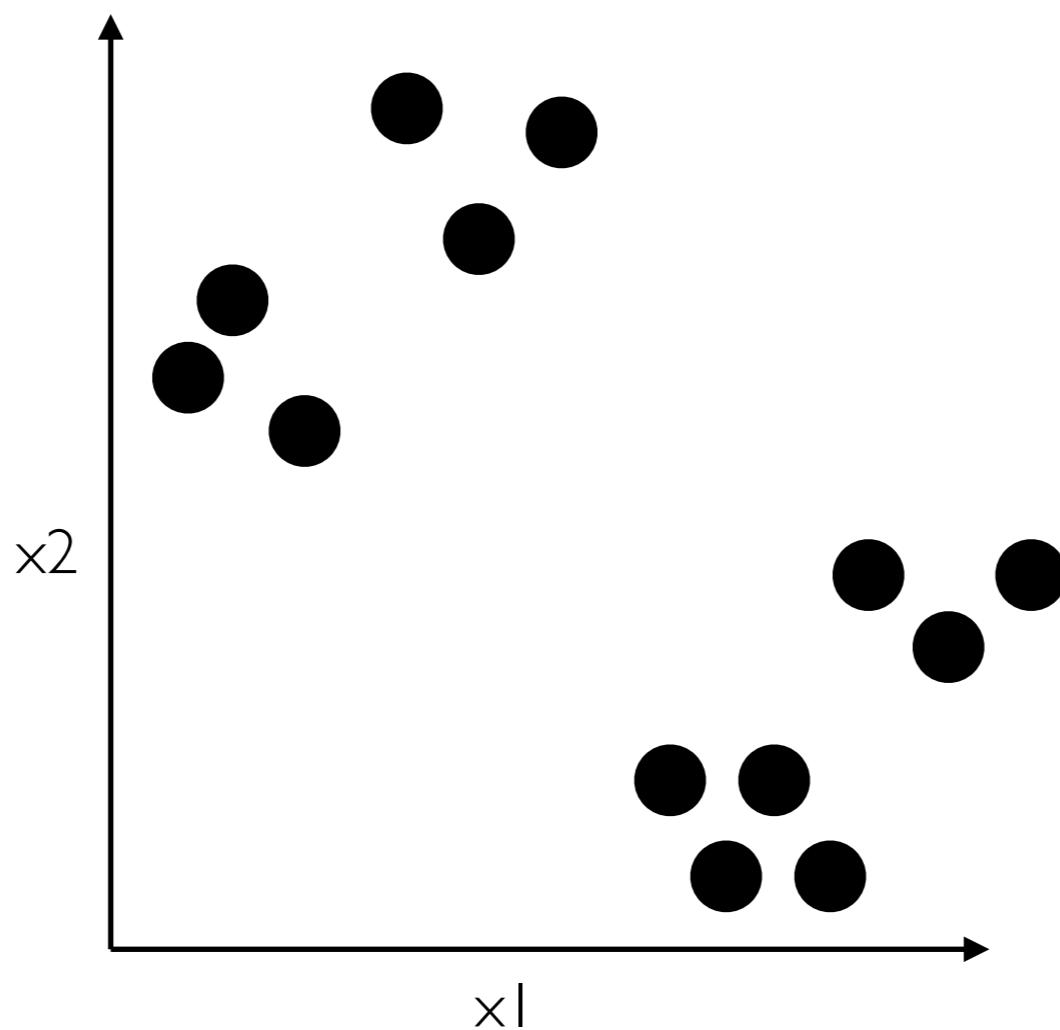
# KMeans or Agglomerative Clustering?
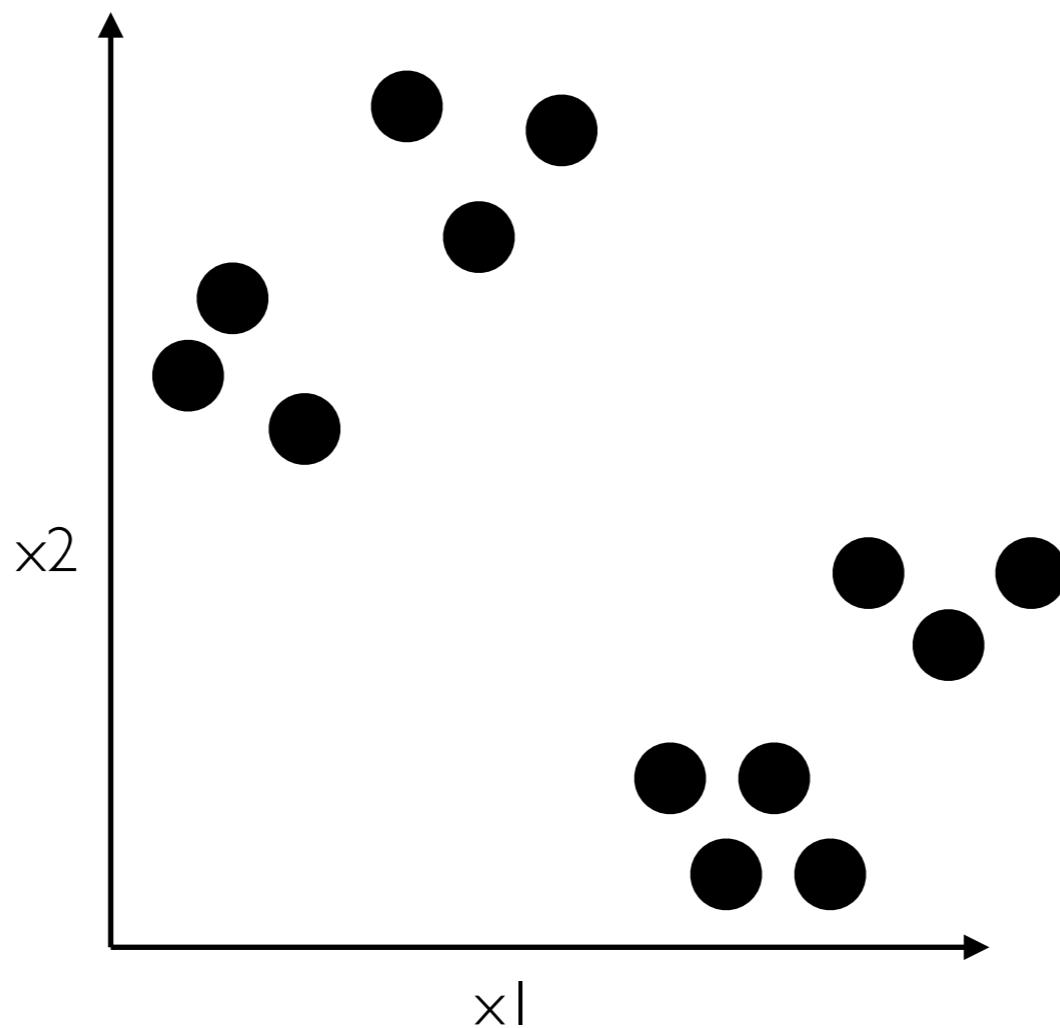


**AgglomerativeClustering** can show us that the two big clusters contain sub clusters.

# KMeans or Agglomerative Clustering?
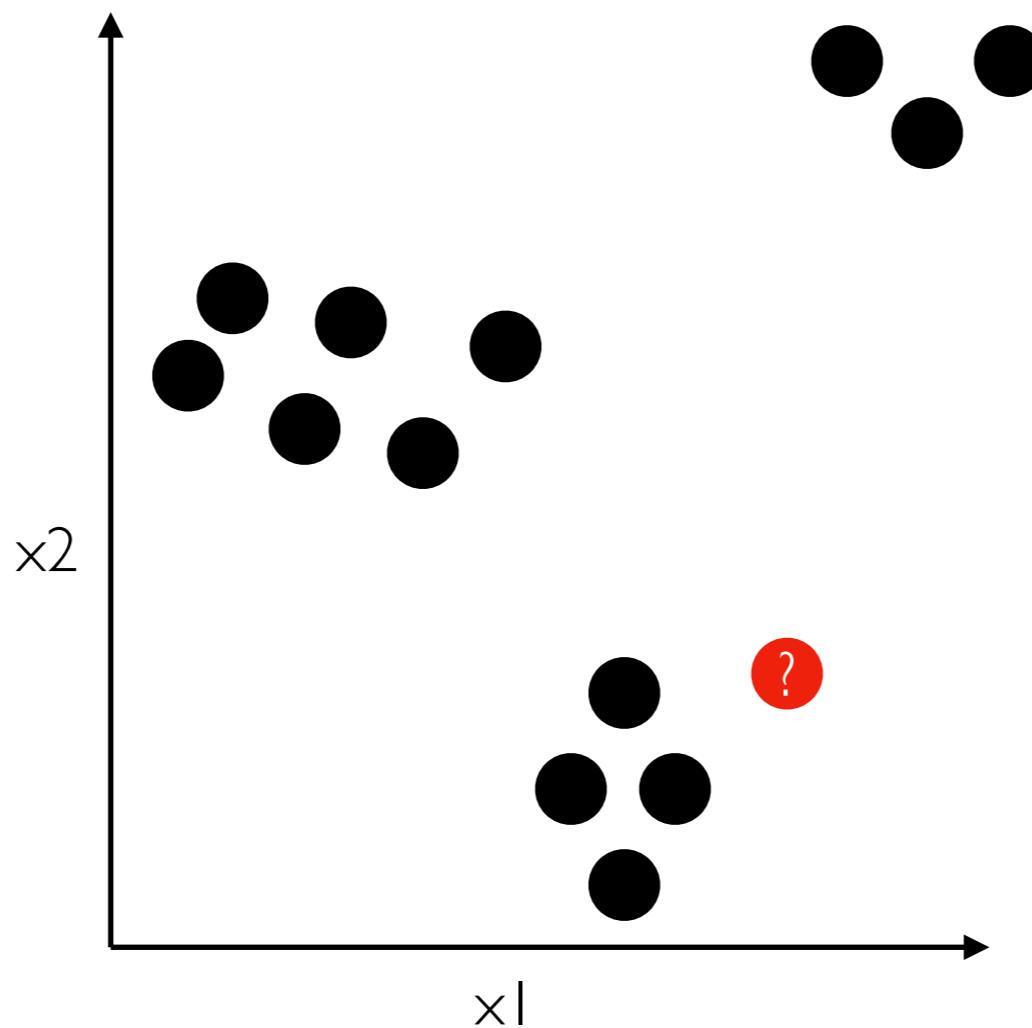


After identifying some clusters from initial data, we will need to look at new data points and find what cluster is the best match

# KMeans or Agglomerative Clustering?
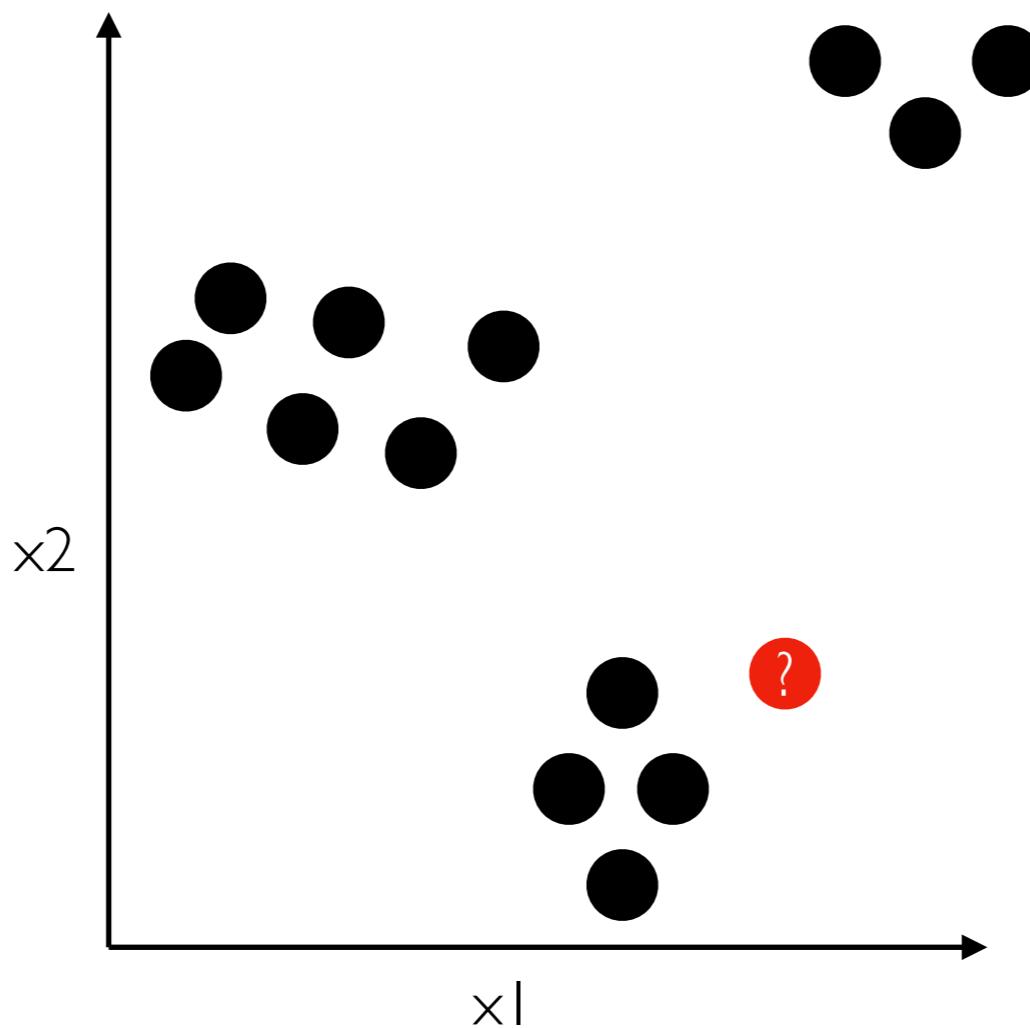


Use **KMeans**, because it can do fit and predict on separate datasets.
AgglomerativeClustering can only do fit_predict on a single dataset.