Daily Inspiration

Sleep more,
perish less.

I'm InspiroBot.

I am an artificial intelligence dedicated to generating unlimited amounts of unique inspirational quotes for endless enrichment of pointless human existence.

Today

- Recap yesterday
- Clustering:
  - Why clustering is (logically) impossible
  - Basics:
    - Prototype clustering: k-nearest neighbours
    - Agglomerative/hierarchical clustering: how we make phylogenies

Recap yesterday (neural networks)

- In simplest form: hierarchically ordered logistic regressors, distilling most distinguishing features from data and then predicting.

- Use backpropagation to train: partial derivatives showing how weights and biases of current layer should change to reduce cost, and how output from previous layer should change. Latter propagates the error back → recursively look how previous layer's weights and biases should change.

- Convolutional neural networks reduce number of parameters massively *and* take local structure into account by convolving filters over images resulting in feature maps. These filters become sensitive to certain image features useful for classification.

Clustering

- Want to find some structure in data automatically.
- Unsupervised learning, don't have true or correct clustering.
- In fact, correct clustering **doesn't exist**.
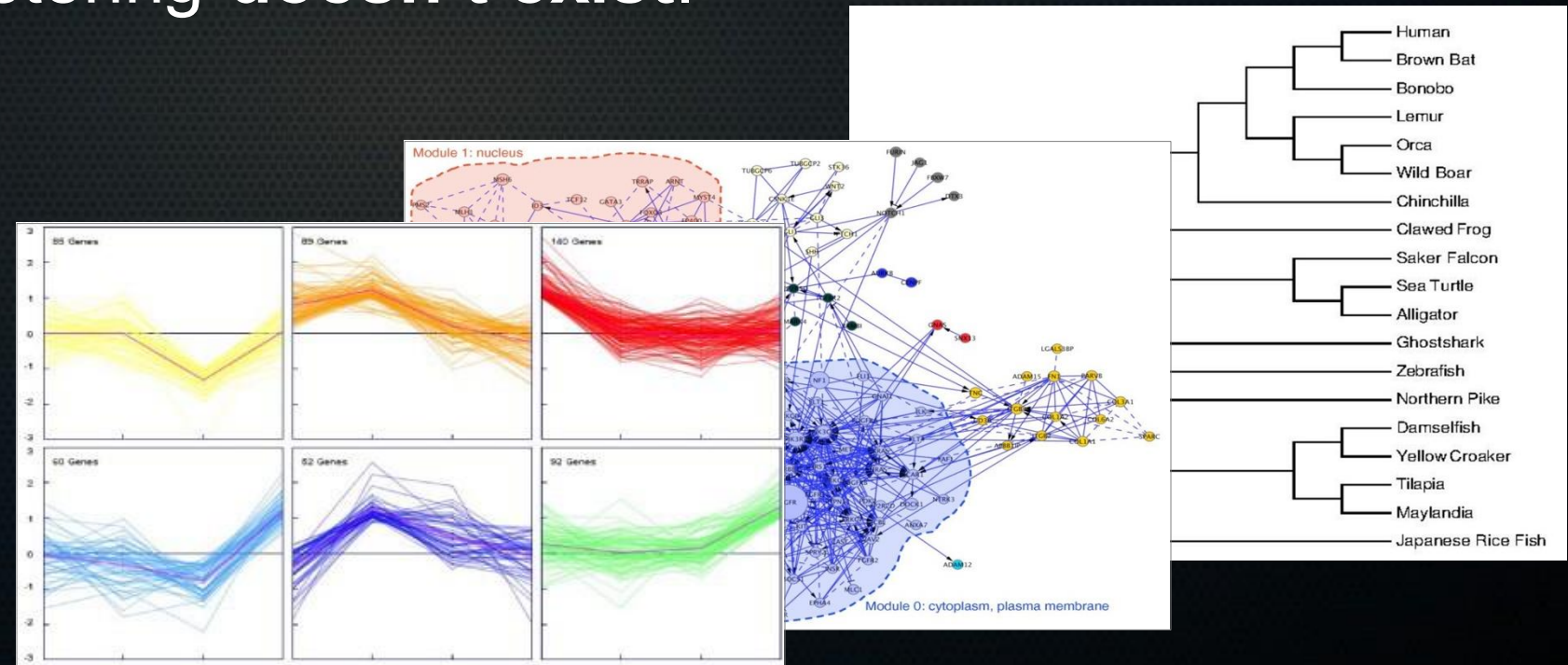
Ugly duckling theorem

- Who is the odd-one-out?
- Who is the ugly rubber duckling?

Ugly duckling theorem

- Who is the odd-one-out?

- Who is the ugly rubber duckling?

Yellow	1	1	0
Left-facing	1	0	1

Ugly duckling theorem

Yellow	1	1	0
Left-facing	1	0	1

Ugly duckling theorem

Yellow	1	1	0
Left-facing	1	0	1

Ugly duckling theorem

Yellow	1	1	0
Left-facing	1	0	1

Perhaps these Boolean features we measured were somewhat arbitrary.
Perhaps combinations of them are more informative?

Ugly duckling theorem

Yellow	1	1	0
Left-facing	1	0	1

Perhaps these Boolean features we measured were somewhat arbitrary.
Perhaps combinations of them are more informative?
→ Only fair way to do that is to make all logical combinations of these two features using Boolean functions.

Ugly duckling theorem

Yellow	1	1	0
Left-facing	1	0	1

Perhaps these Boolean features we measured were somewhat arbitrary.
Perhaps combinations of them are more informative?
→ Only fair way to do that is to make all logical combinations of these two features using Boolean functions.

Boolean Functions

A	B	OR	Not OR	And	Not And	XOR	Not XOR
0	0	0	1	0	1	0	1
0	1	1	0	0	1	1	0
1	0	1	0	0	1	1	0
1	1	1	0	1	0	0	1

Ugly duckling theorem

Yellow	1	1	0
Left-facing	1	0	1

Example: Yellow XOR Left-facing

Yellow XOR left-facing	0	1	1

Boolean Functions

A	B	OR	Not OR	And	Not And	XOR	Not XOR
0	0	0	1	0	1	0	1
0	1	1	0	0	1	1	0
1	0	1	0	0	1	1	0
1	1	1	0	1	0	0	1

13

Ugly duckling theorem

Yellow	1	1	0
Left-facing	1	0	1
Yellow AND left-facing	1	0	0
Yellow OR left-facing	1	1	1
Yellow XOR left-facing	0	1	1
Yellow NOT AND left-facing	0	1	1
Yellow NOT OR left-facing	0	0	0
Yellow NOT XOR left-facing	1	0	0
Yellow AND NOT left-facing	0	1	0
Yellow OR NOT left-facing	1	1	0
Yellow XOR NOT left-facing	1	0	0
NOT yellow	0	0	1
NOT yellow AND left-facing	0	0	1
NOT yellow OR left-facing	1	0	1
NOT yellow XOR left-facing	0	1	0
NOT left-facing	0	1	0

Boolean Functions

A	B	OR	Not OR	And	Not And	XOR	Not XOR
0	0	0	1	0	1	0	1
0	1	1	0	0	1	1	0
1	0	1	0	0	1	1	0
1	1	1	0	1	0	0	1

Ugly duckling theorem

Yellow	1	1	0
Left-facing	1	0	1
Yellow AND left-facing	1	0	0
Yellow OR left-facing	1	1	1
Yellow XOR left-facing	0	1	1
Yellow NOT AND left-facing	0	1	1
Yellow NOT OR left-facing	0	0	0
Yellow NOT XOR left-facing	1	0	0
Yellow AND NOT left-facing	0	1	0
Yellow OR NOT left-facing	1	1	0
Yellow XOR NOT left-facing	1	0	0
NOT yellow	0	0	1
NOT yellow AND left-facing	0	0	1
NOT yellow OR left-facing	1	0	1
NOT yellow XOR left-facing	0	1	0
NOT left-facing	0	1	0

3

Okay, so is there a correct clustering now?
Let's tally how similar each group of objects is

Boolean Functions

A	B	OR	Not OR	And	Not And	XOR	Not XOR
0	0	0	1	0	1	0	1
0	1	1	0	0	1	1	0
1	0	1	0	0	1	1	0
1	1	1	0	1	0	0	1

5

Ugly duckling theorem

Yellow	1	1	0
Left-facing	1	0	1
Yellow AND left-facing	1	0	0
Yellow OR left-facing	1	1	1
Yellow XOR left-facing	0	1	1
Yellow NOT AND left-facing	0	1	1
Yellow NOT OR left-facing	0	0	0
Yellow NOT XOR left-facing	1	0	0
Yellow AND NOT left-facing	0	1	0
Yellow OR NOT left-facing	1	1	0
Yellow XOR NOT left-facing	1	0	0
NOT yellow	0	0	1
NOT yellow AND left-facing	0	0	1
NOT yellow OR left-facing	1	0	1
NOT yellow XOR left-facing	0	1	0
NOT left-facing	0	1	0

Okay, so is there a correct clustering now?
Let's tally how similar each group of objects is

3

3

Boolean Functions

A	B	OR	Not OR	And	Not And	XOR	Not XOR
0	0	0	1	0	1	0	1
0	1	1	0	0	1	1	0
1	0	1	0	0	1	1	0
1	1	1	0	1	0	0	1

16

Ugly duckling theorem

Yellow	1	1	0
Left-facing	1	0	1
Yellow AND left-facing	1	0	0
Yellow OR left-facing	1	1	1
Yellow XOR left-facing	0	1	1
Yellow NOT AND left-facing	0	1	1
Yellow NOT OR left-facing	0	0	0
Yellow NOT XOR left-facing	1	0	0
Yellow AND NOT left-facing	0	1	0
Yellow OR NOT left-facing	1	1	0
Yellow XOR NOT left-facing	1	0	0
NOT yellow	0	0	1
NOT yellow AND left-facing	0	0	1
NOT yellow OR left-facing	1	0	1
NOT yellow XOR left-facing	0	1	0
NOT left-facing	0	1	0

Okay, so is there a correct clustering now?
Let's tally how similar each group of objects is

3

3

3

Boolean Functions

A	B	OR	Not OR	And	Not And	XOR	Not XOR
0	0	0	1	0	1	0	1
0	1	1	0	0	1	1	0
1	0	1	0	0	1	1	0
1	1	1	0	1	0	0	1

Ugly duckling theorem

Yellow	1	1	0
Left-facing	1	0	1
Yellow AND left-facing	1	0	0
Yellow OR left-facing	1	1	1
Yellow XOR left-facing	0	1	1
Yellow NOT AND left-facing	0	1	1
Yellow NOT OR left-facing	0	0	0
Yellow NOT XOR left-facing	1	0	0
Yellow AND NOT left-facing	0	1	0
Yellow OR NOT left-facing	1	1	0
Yellow XOR NOT left-facing	1	0	0
NOT yellow	0	0	1
NOT yellow AND left-facing	0	0	1
NOT yellow OR left-facing	1	0	1
NOT yellow XOR left-facing	0	1	0
NOT left-facing	0	1	0

3

3

3

Okay, so is there a correct clustering now?
Let's tally how similar each group of objects is

→ **Still arbitrary who we cluster together!**
→ **No rubber duckling is ugly.**

Boolean Functions

A	B	OR	Not OR	And	Not And	XOR	Not XOR
0	0	0	1	0	1	0	1
0	1	1	0	0	1	1	0
1	0	1	0	0	1	1	0
1	1	1	0	1	0	0	1

8

What does this mean?

- You cannot cluster *anything* without *some sort of (inductive) bias* → what you consider to be important for some reason.

What does this mean?

- You cannot cluster *anything* without *some sort of (inductive) bias* → what you consider to be important for some reason.

- Clusters cannot be *correct* or the best clusters, they can only be good for whatever purpose you want to use them for.

What does this mean?

- In biology, we easily assay expression of 20,000 genes.

  Could think: genes are not Boolean, they are continuous values.

	Sample 1
Gene 1 [1,2>	0
Gene 1 [2,3>	1
Gene 2 [-10,-9>	0
Gene 2 [-9, -8>	1

	Expr. Gene 1	Expr. Gene 2
Sample 1	2.45	-8.677

What does this mean?

- In practice: bias is introduced by the features we don't include:

  - Morphological characteristics

  - Metabolite concentrations

What does this mean?

- In practice: bias is introduced by the features we don't include:

  - Morphological characteristics

  - Metabolite concentrations

  In clustering algorithms:

  - What distance metric do you use?

  - What are the assumptions of the algorithm?

What does this mean?

▪ However: *even if* you know every atom, every quantum state of the objects to be clustered there is to know:
→ no *correct* clustering. Only useful.

What does this mean?

- So: clustering always a dialogue between you and the data. Trying to find structure that is useful to you.

Today

- ~~Recap yesterday~~

- Clustering:
  - ~~Why clustering is (logically) impossible~~
  - Basics:
    - Prototype clustering: k-nearest neighbours
    - Agglomerative/hierarchical clustering: how we make phylogenies
      - Hierarchical clustering and phylogeny

K-means clustering: prototype method
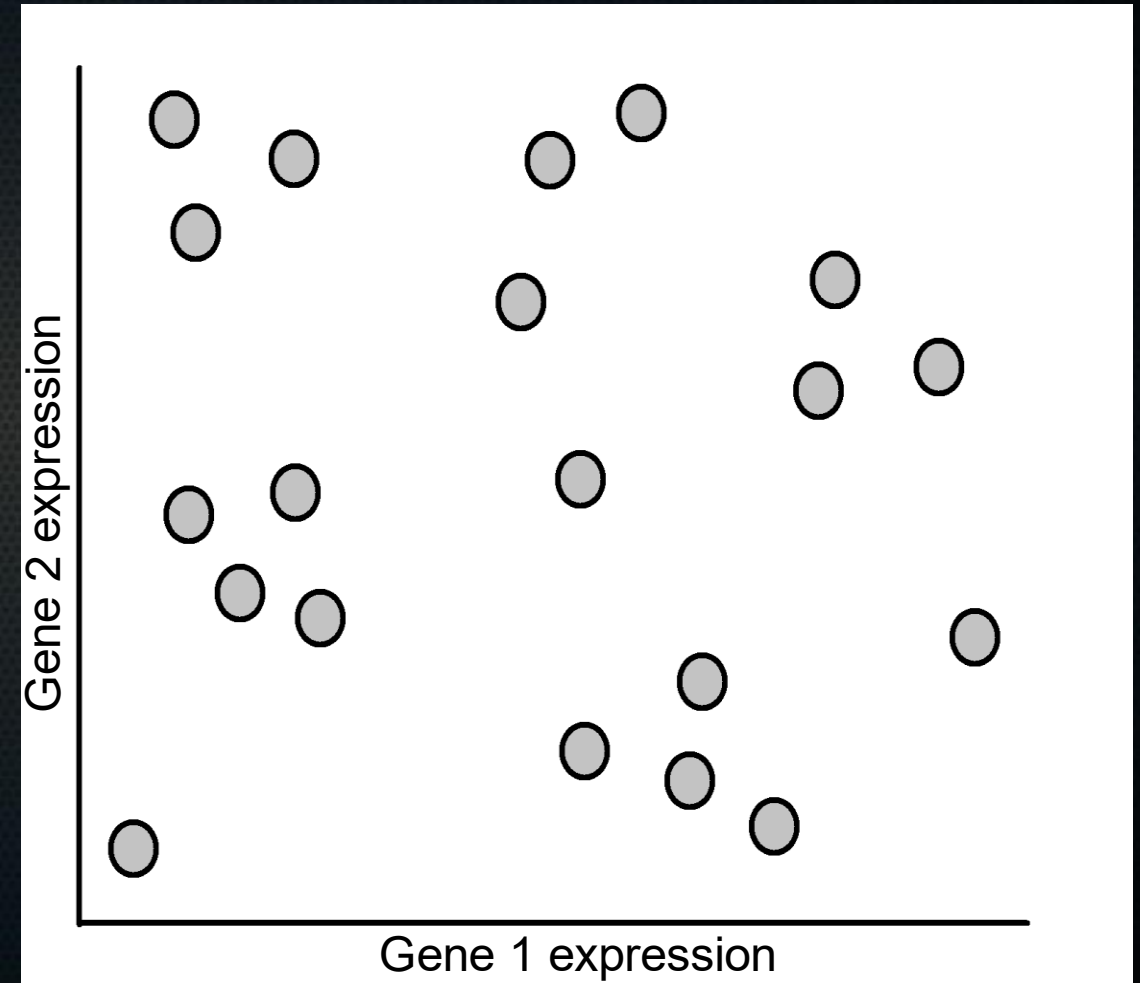
K-means clustering: prototype method

- Want to form clusters of like samples
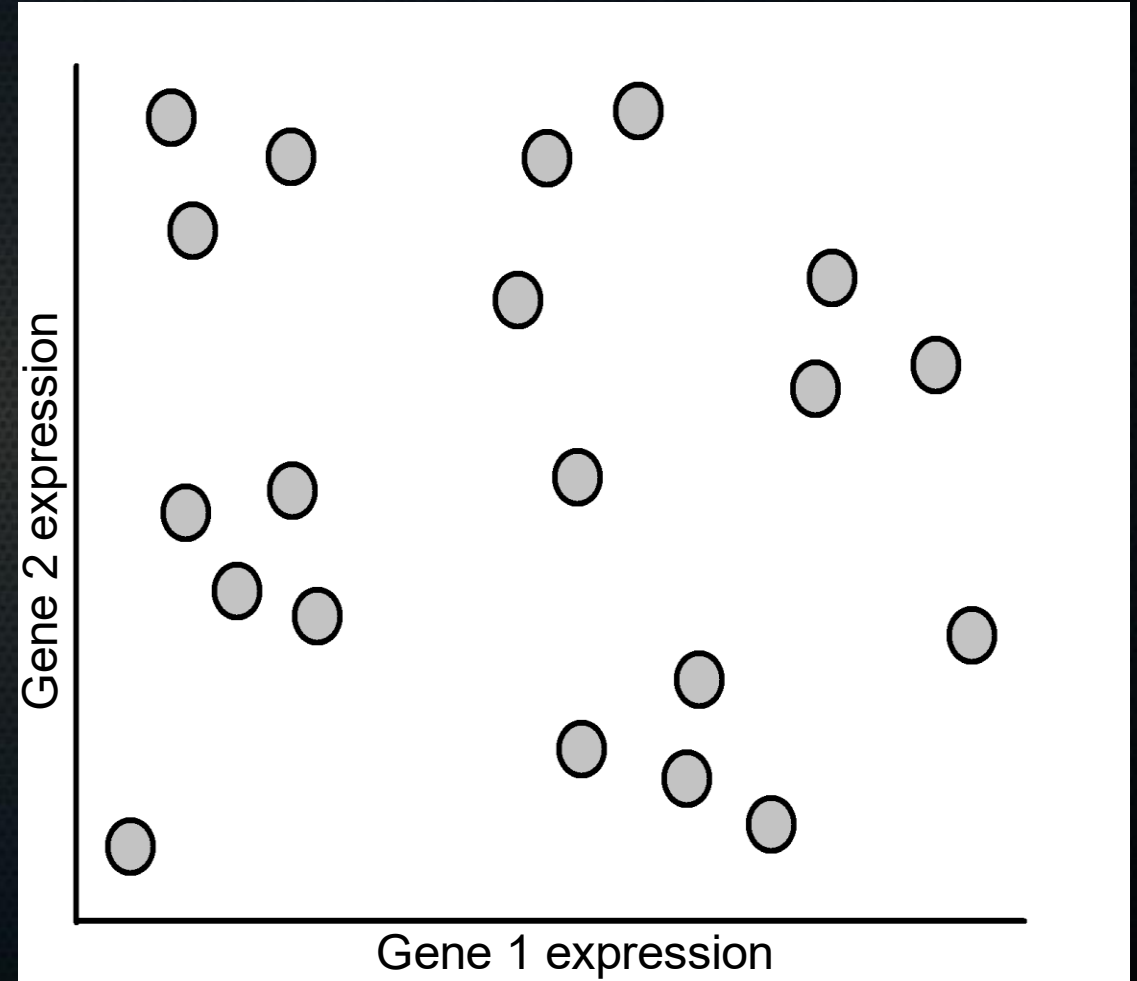
K-means clustering: prototype method

▪ Want to form clusters of like samples
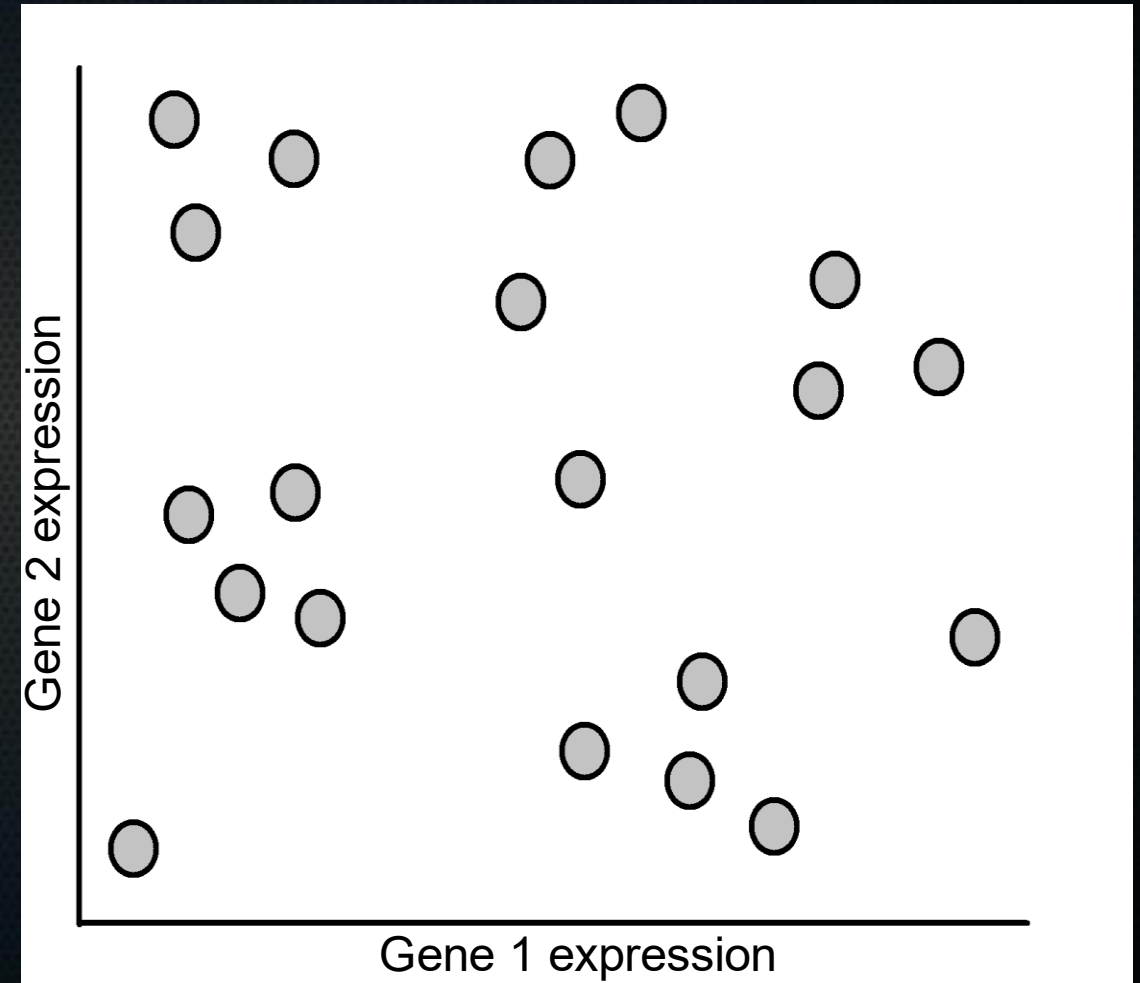
▪ Need two things:

K-means clustering: prototype method

- Want to form clusters of like samples

- Need two things:

  - How different is each point from each other point? → *distance metric*
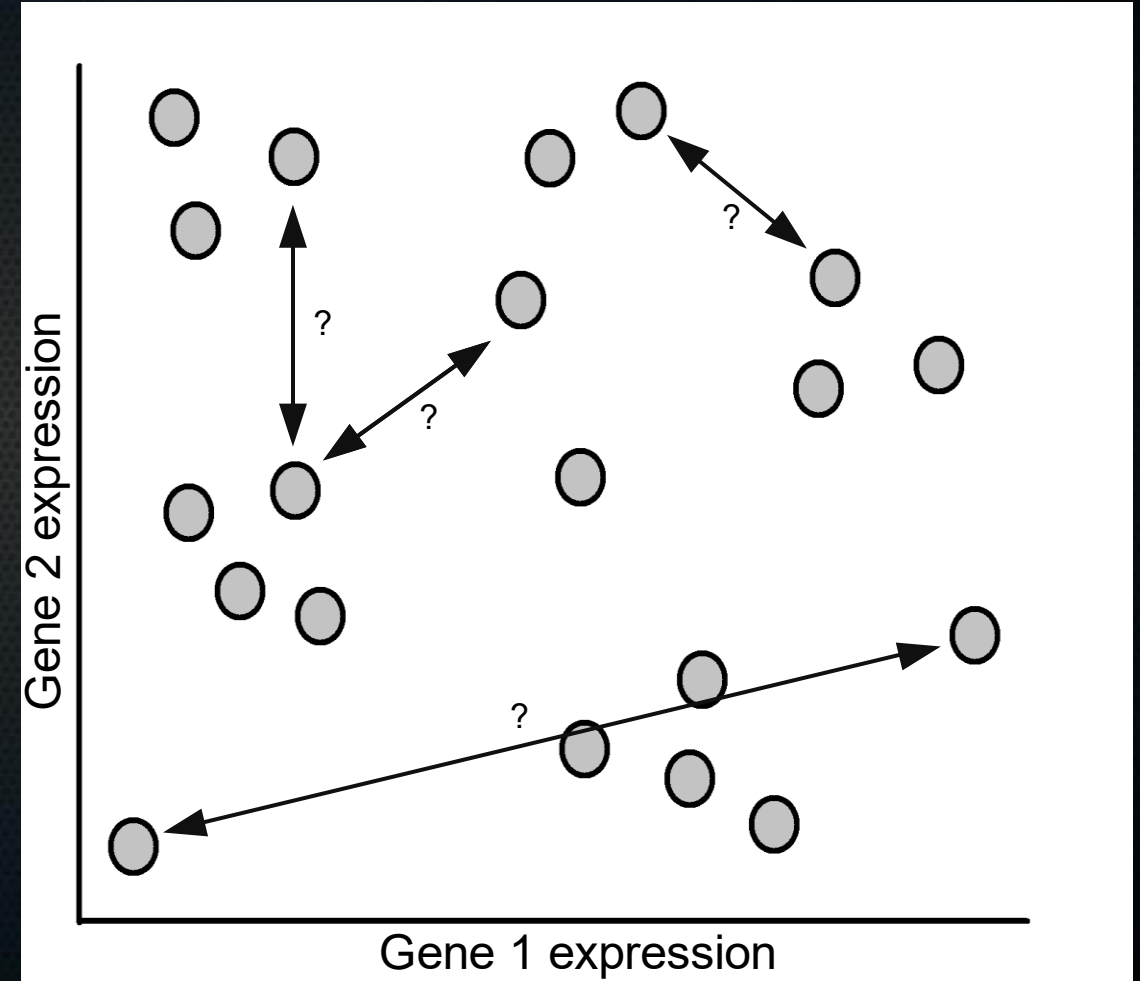
K-means clustering: prototype method

- Want to form clusters of like samples

- Need two things:

  - How different is each point from each other point? → *distance metric*

  - How many clusters do we want to form? → *K*
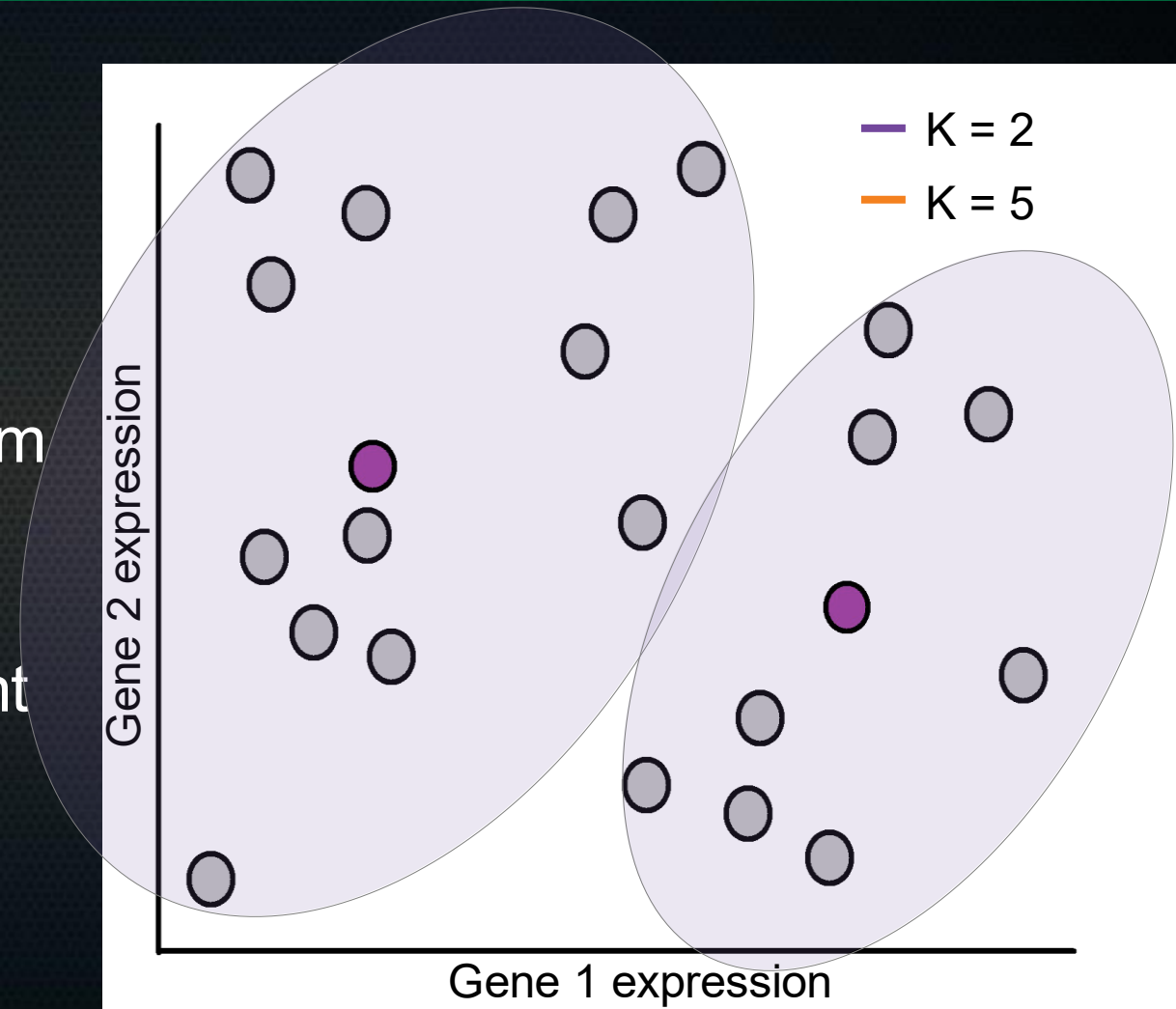
K-means clustering: prototype method

- Want to form clusters of like samples

- Need two things:
    - How different is each point from each other point? → *distance metric*
    - How many clusters do we want to form? → *K*

- Call the bright orange dots the *cluster centroids*.
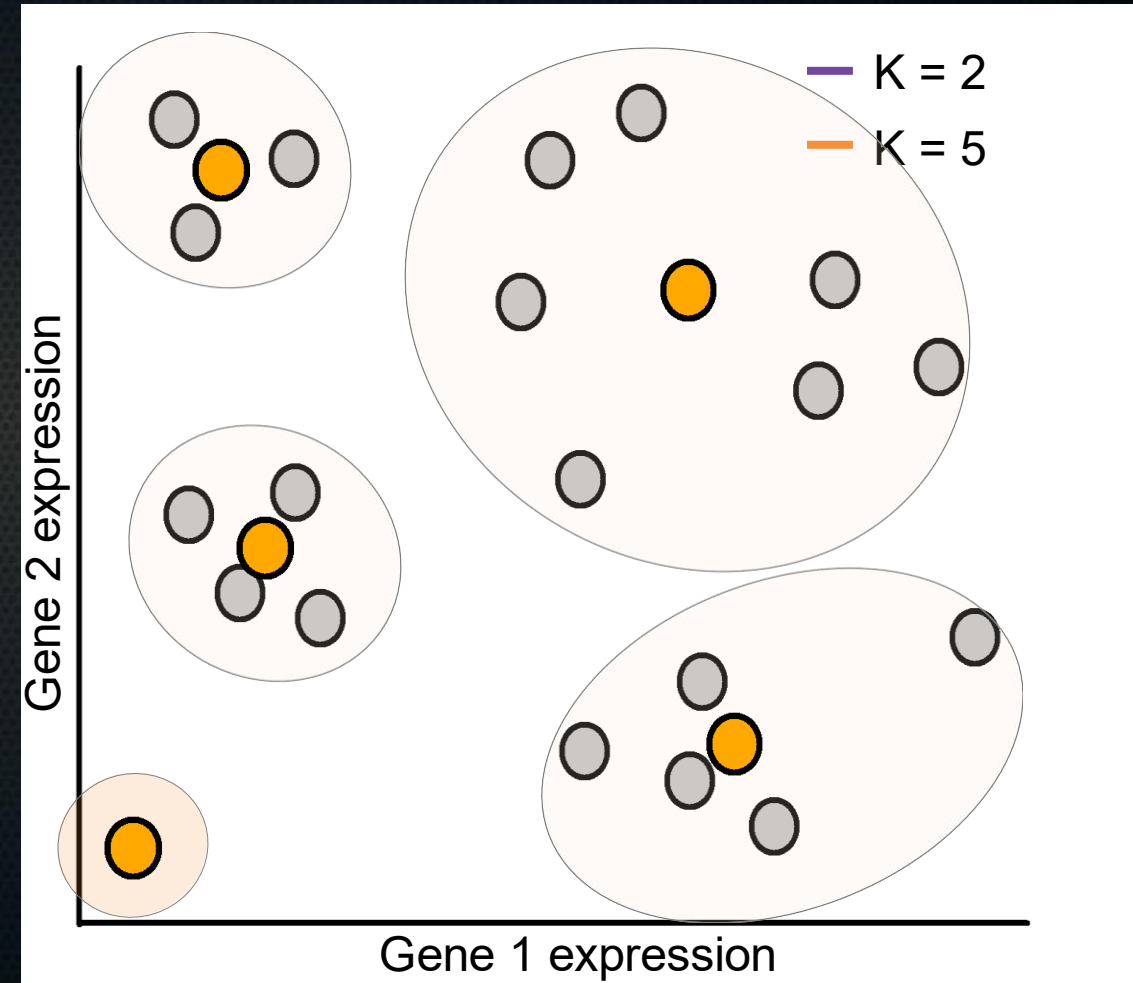
Calculating distances
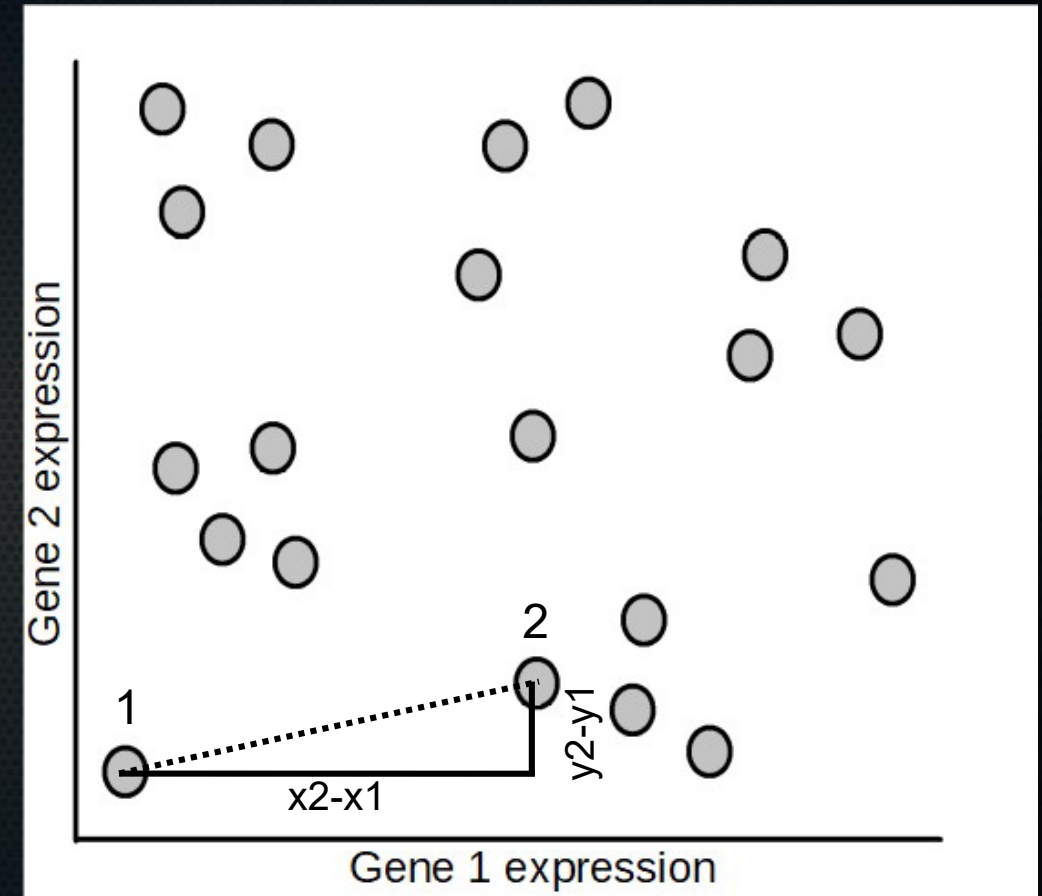
- Euclidian distance: simply the shortest line between two points

Calculating distances

- Euclidian distance: simply the shortest line between two points

$$A^2 + B^2 = C^2$$
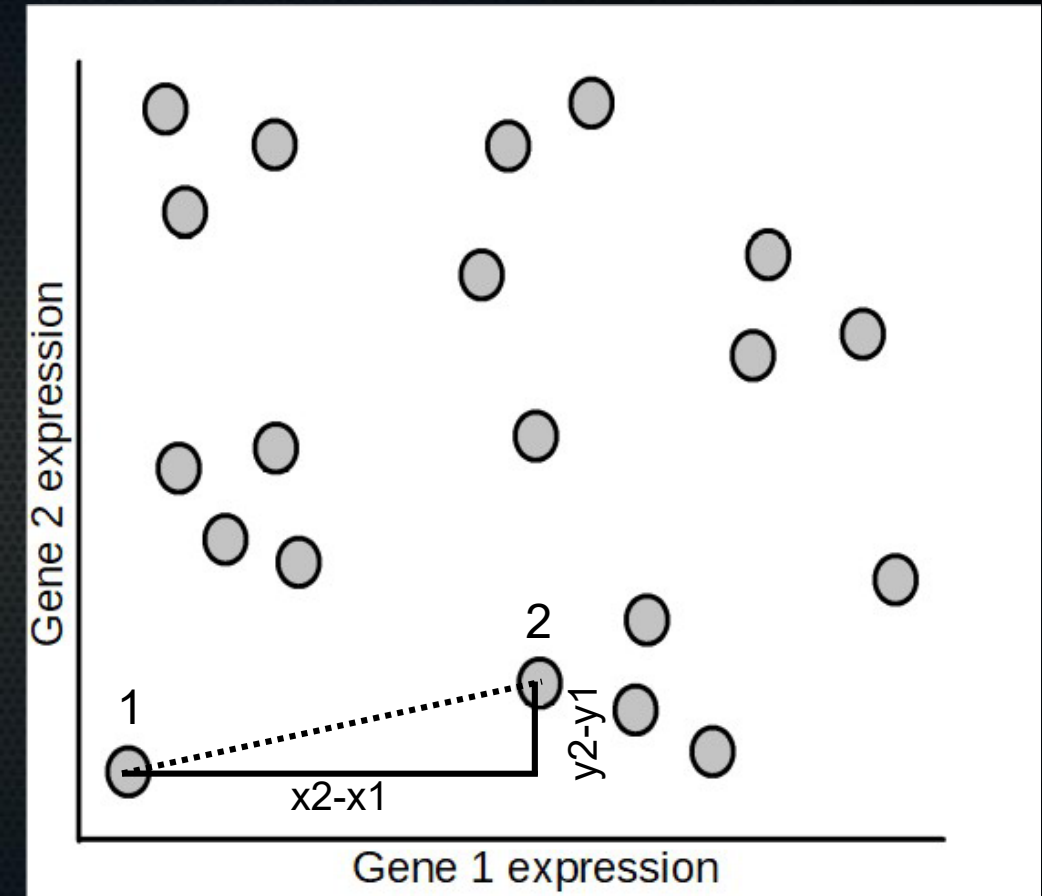
$$C = \sqrt{(A^2 + B^2)} = \sqrt{((x1 - x2)^2 + (y2 - y1)^2)}$$

Calculating distances

- Euclidian distance: simply the shortest line between two points

$$A^2 + B^2 = C^2$$

$$C = \sqrt{(A^2 + B^2)} = \sqrt{((x1 - x2)^2 + (y2 - y1)^2)}$$

- Also works when you have many more genes, i.e. high-dimensional data
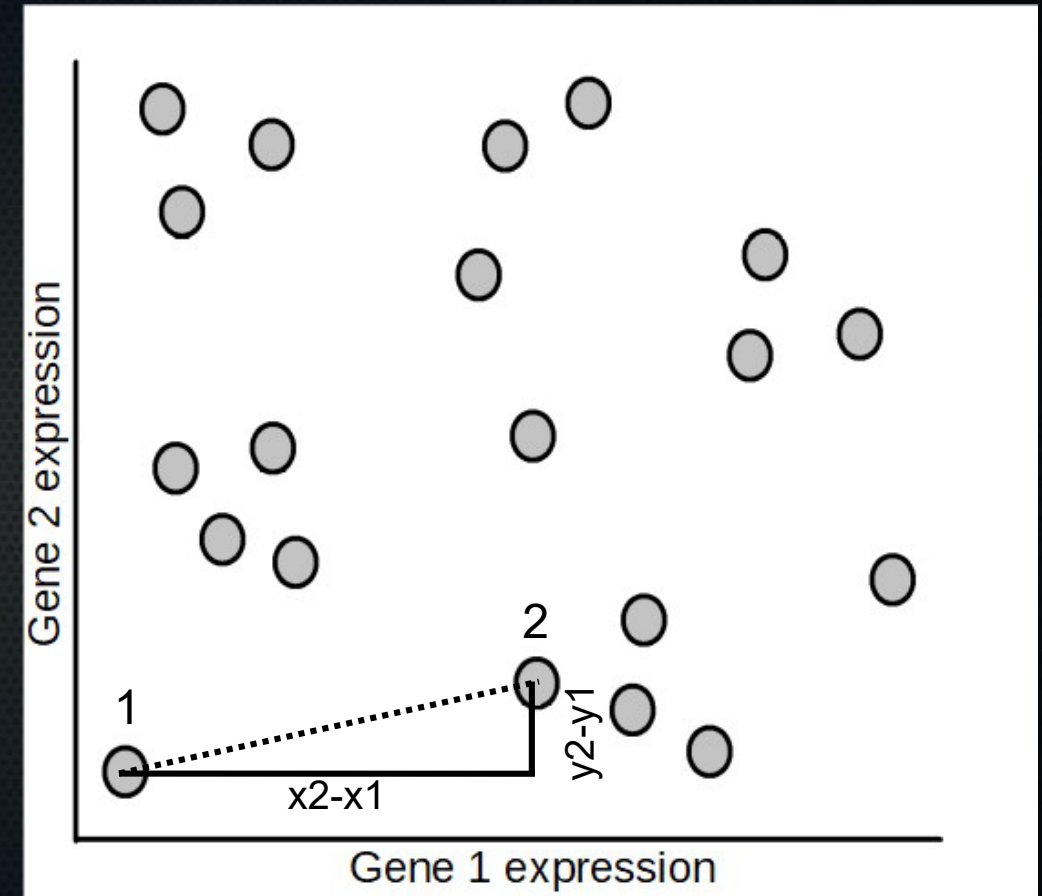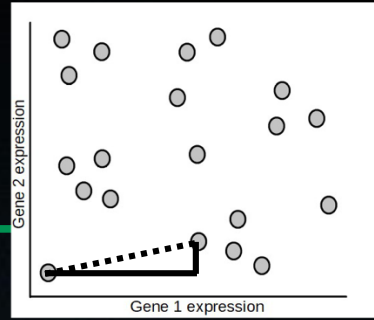
Calculating distances

- Other distance metrics abound, for example: Manhattan distance:
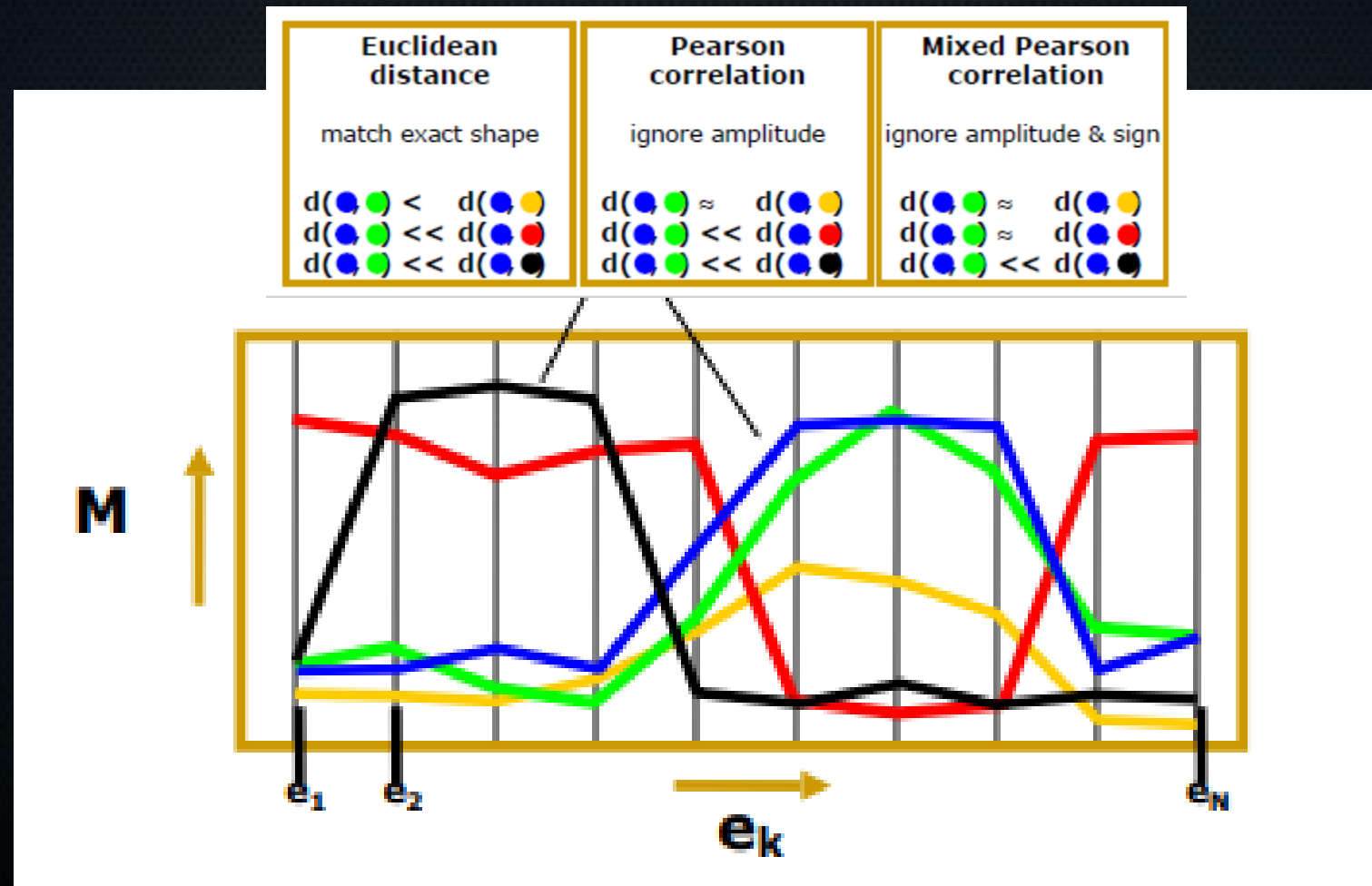
$$C = |x2 - x1| + |y2 - y1|$$

$$|x_1 - x_2| + |y_1 - y_2|$$
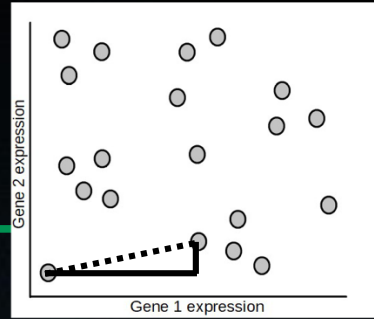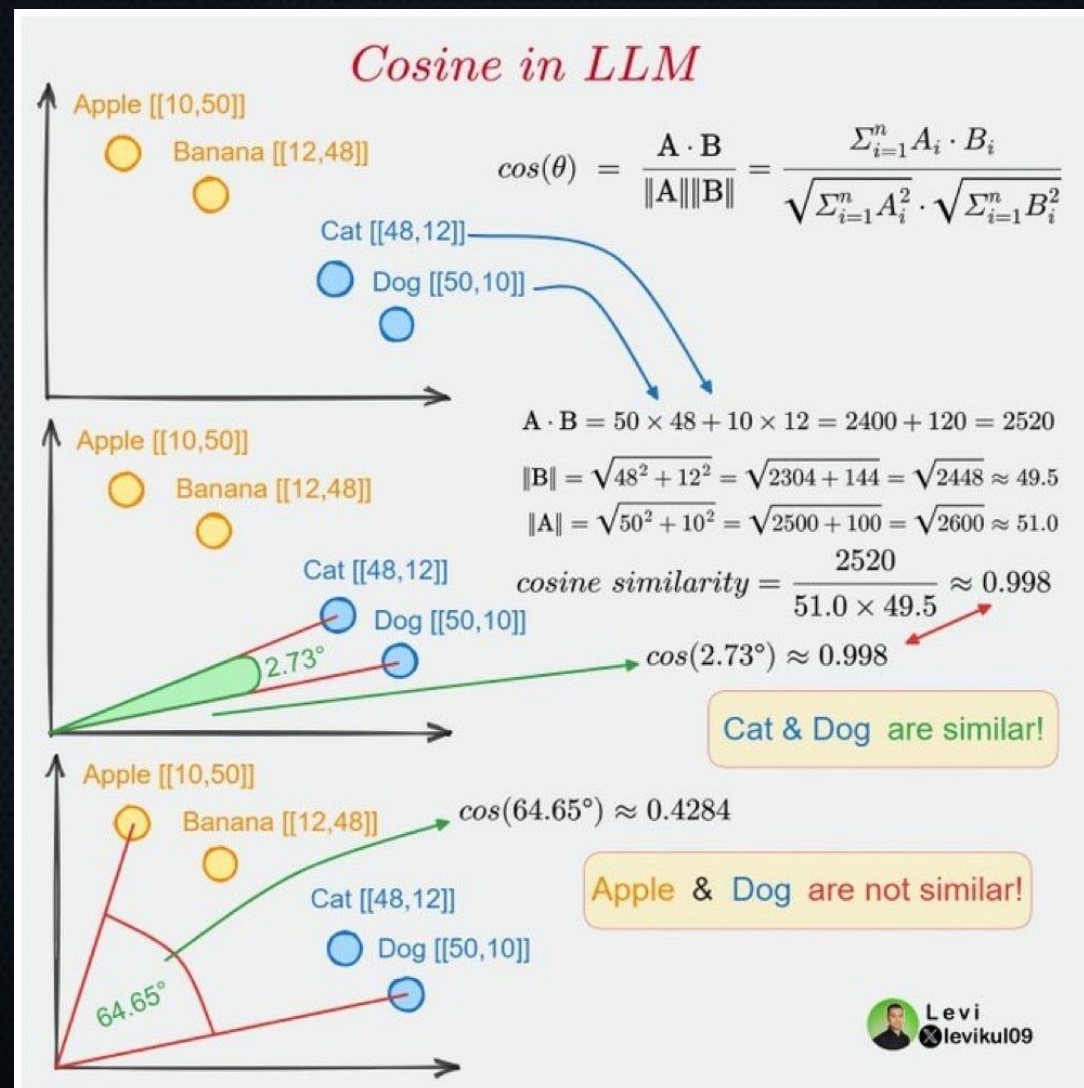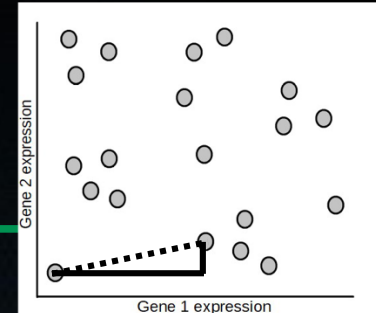
Which genes are close?
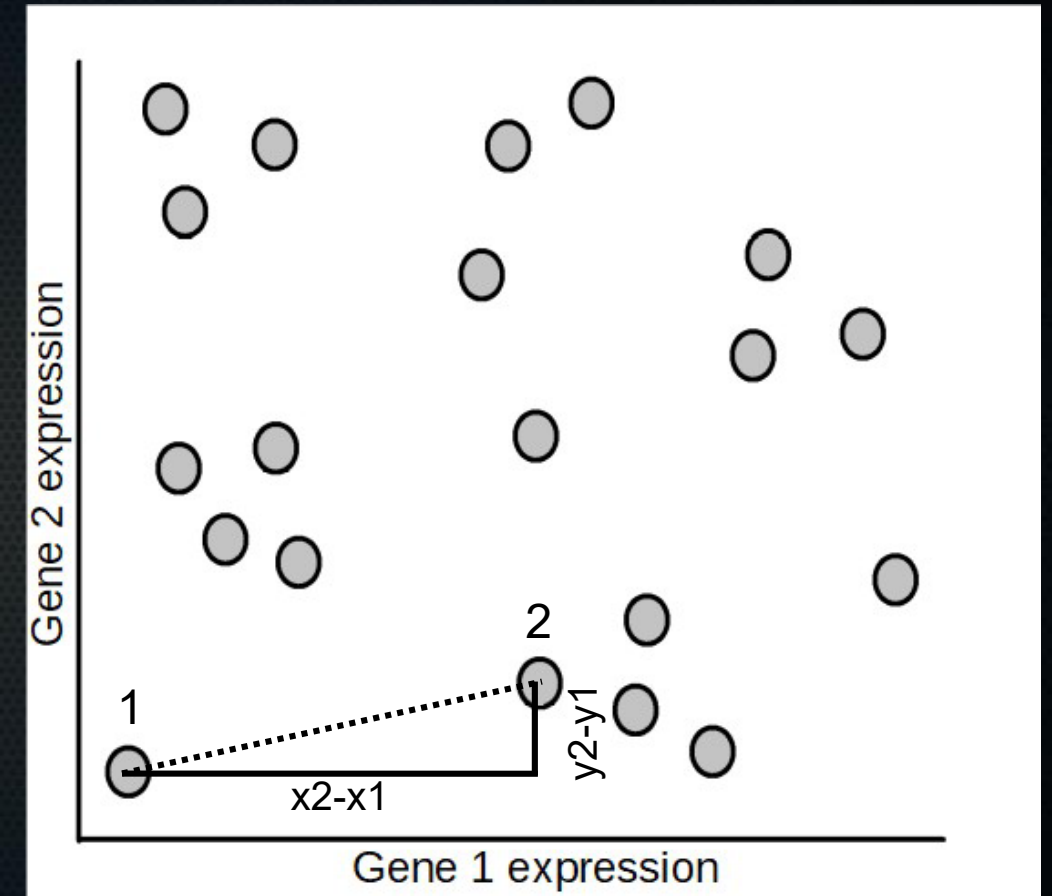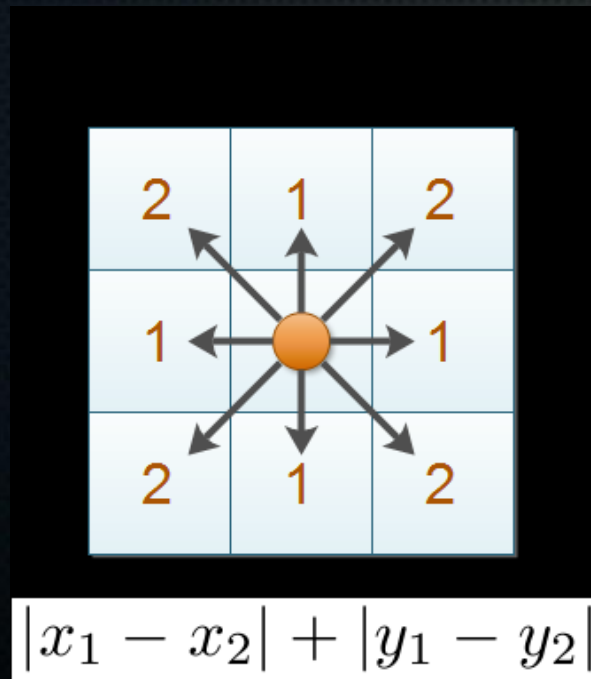
A bit more about distances

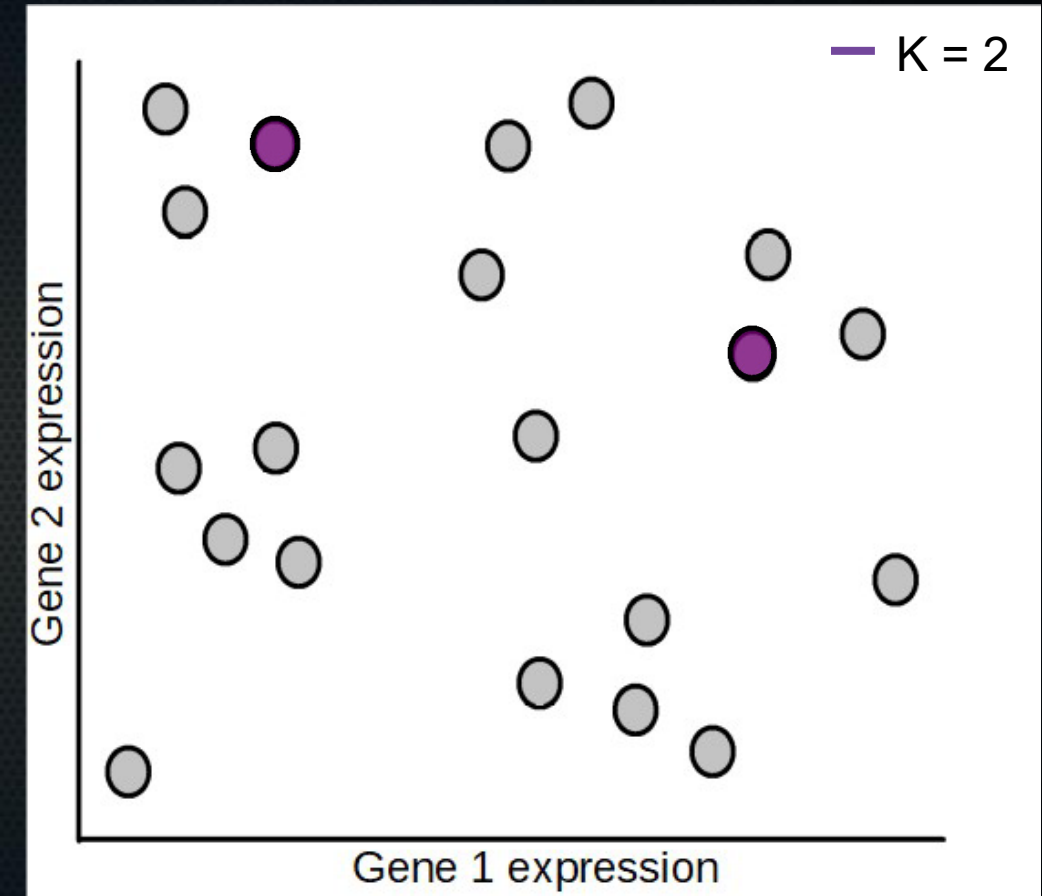## Which genes are close?

A bit more about distances
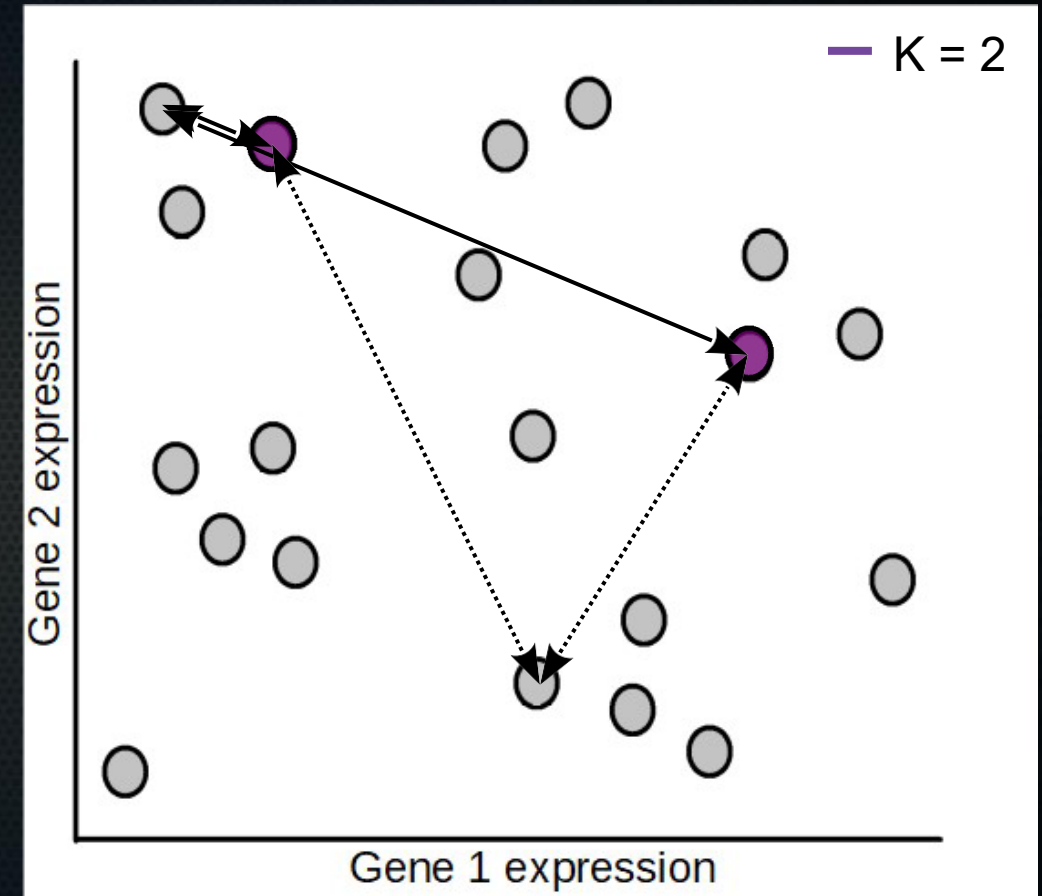
Back to clustering

1

2

x2-x1

y2-y1

K-means clustering in practice

- Start with *K* random prototypes →
pick *K* random data points to start

K-means clustering in practice

- Start with *K* random prototypes →
  pick *K* random data points to start

- Calculate the distance of each point
  to each prototype.

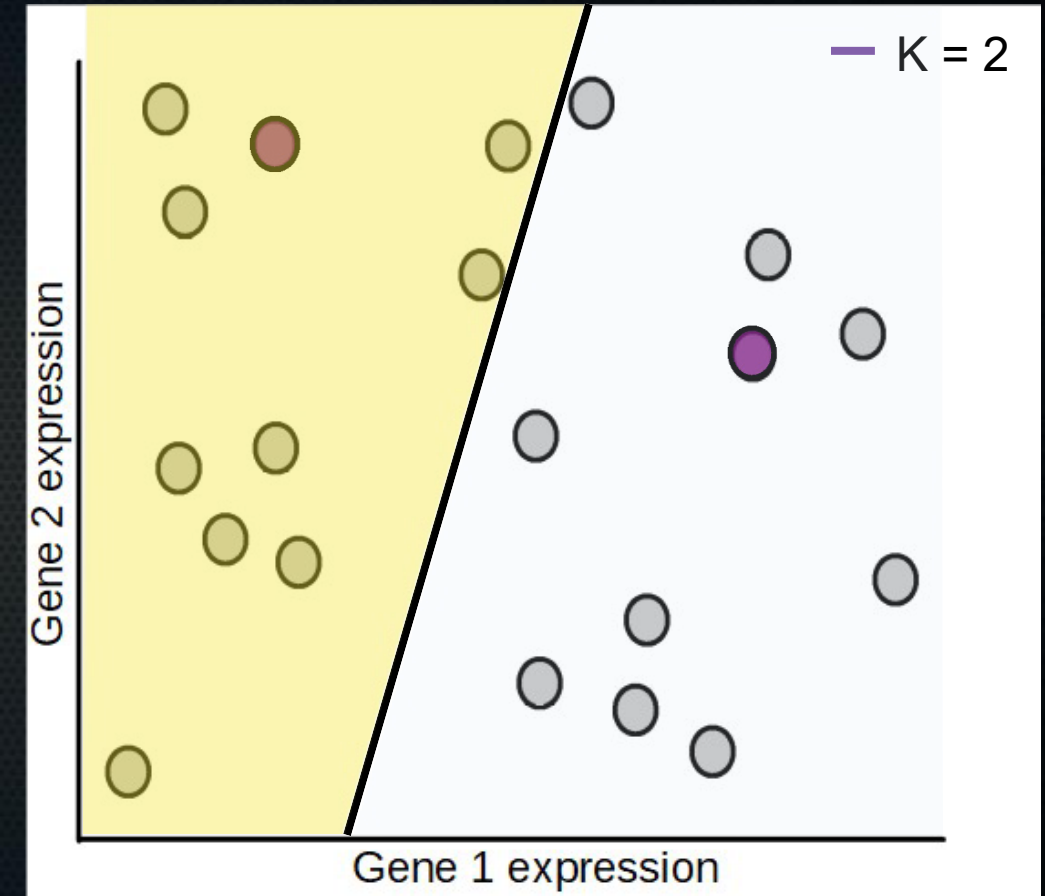# K-means clustering in practice

- Start with *K* random prototypes → pick *K* random data points to start

- Calculate the distance of each point to each prototype.

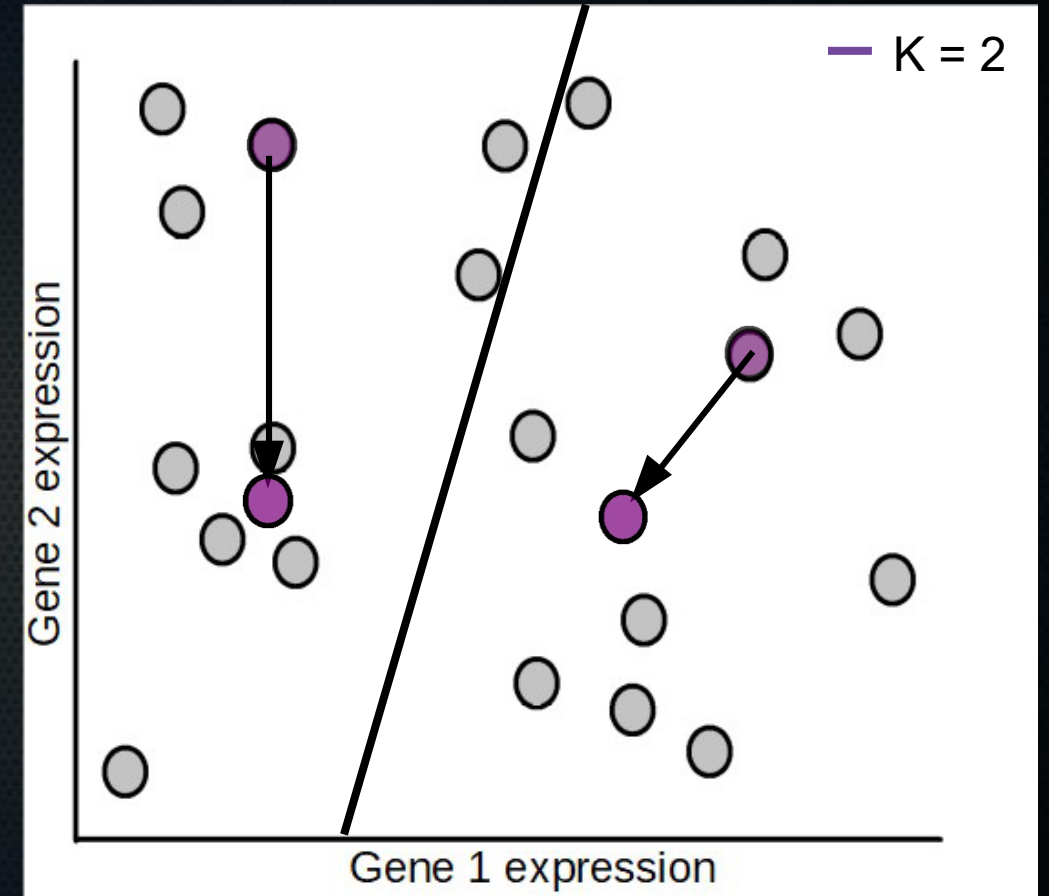- Assign each point to the closest prototype.

K-means clustering in practice

- Start with *K* random prototypes →
  pick *K* random data points to start

- Calculate the distance of each point
  to each prototype.
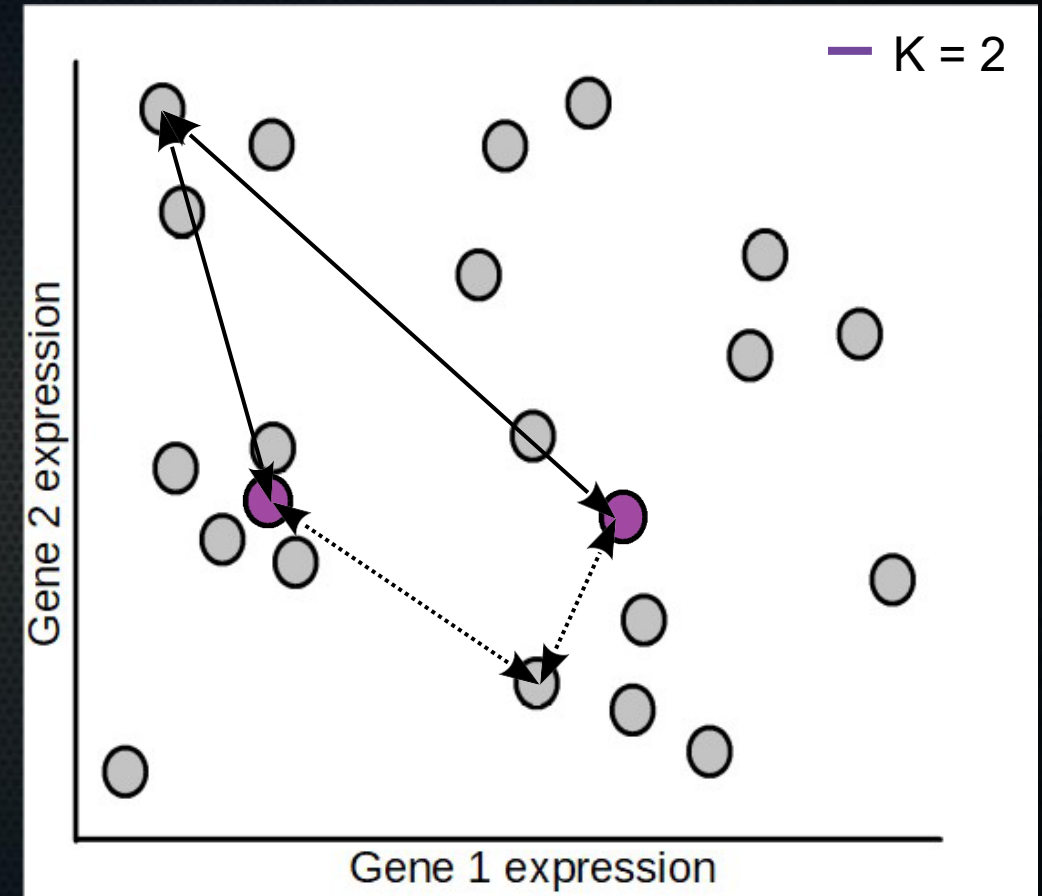
- Assign each point to the closest
  prototype.

- Move the cluster centroid to the
  mean of all points in the cluster

K-means clustering in practice

- Calculate the distance of each point to each cluster centroid (= cluster prototype)

- Assign each point to the closest prototype.

- Iterate until convergence

K-means clustering in practice

- ▪ Calculate the distance of each point to each cluster centroid (= cluster prototype)

- ▪ Assign each point to the closest prototype.
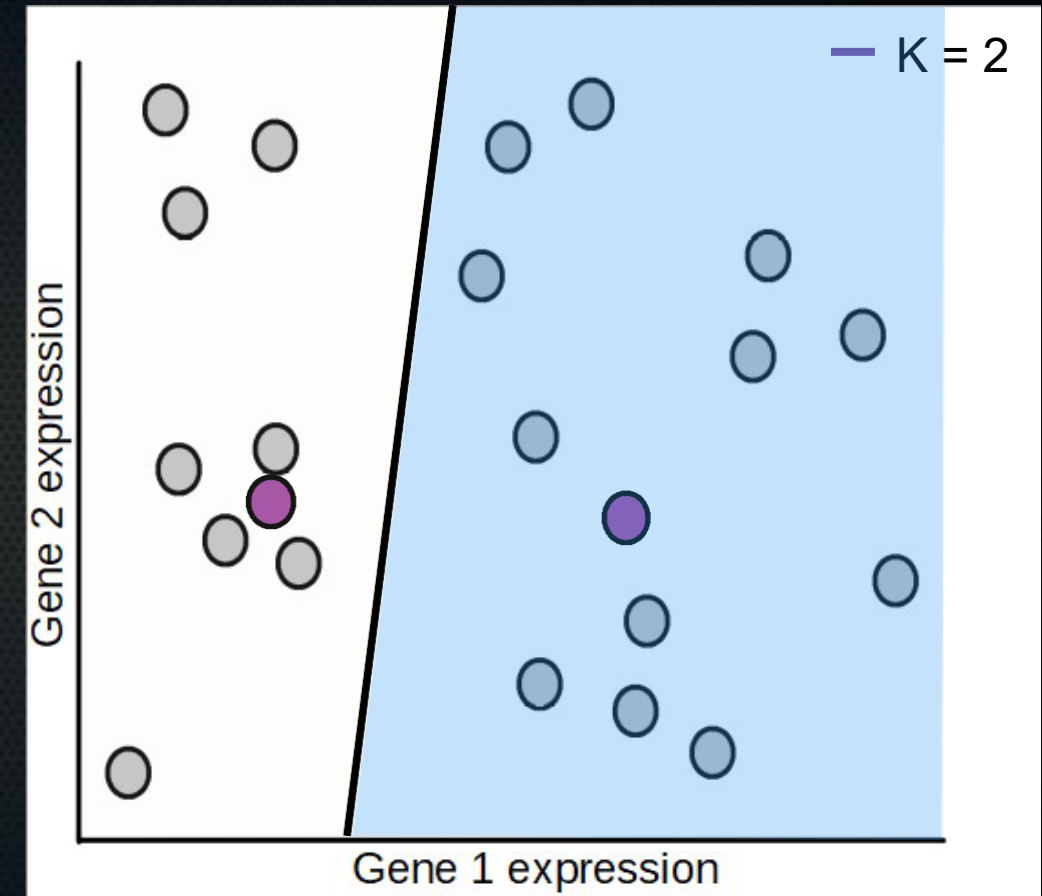
- ▪ Iterate until convergence

K-means clustering in practice

- Calculate the distance of each point to each cluster centroid (= cluster prototype)

- Assign each point to the closest prototype.

- Iterate until convergence

K-means clustering in practice

- Calculate the distance of each point to each cluster centroid (= cluster prototype)

- Assign each point to the closest prototype.

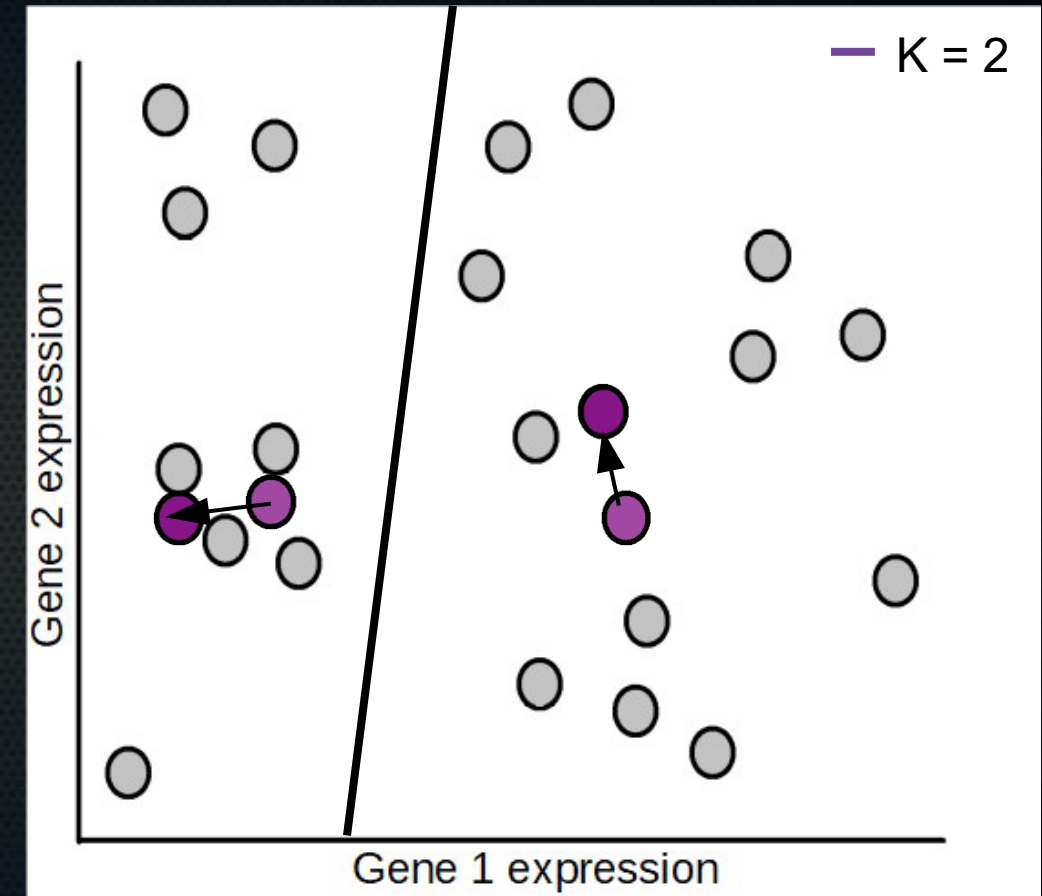- Iterate until convergence

K-means clustering in practice

- Calculate the distance of each point to each cluster centroid (= cluster prototype)

- Assign each point to the closest prototype.
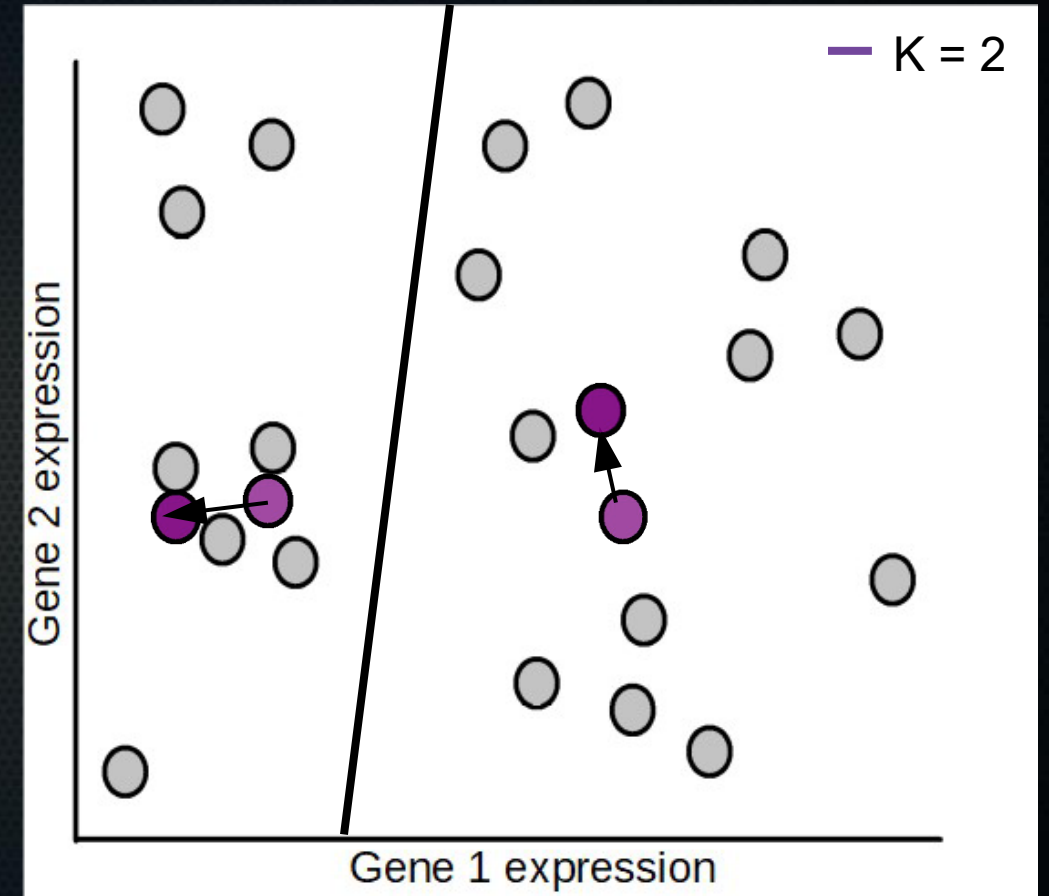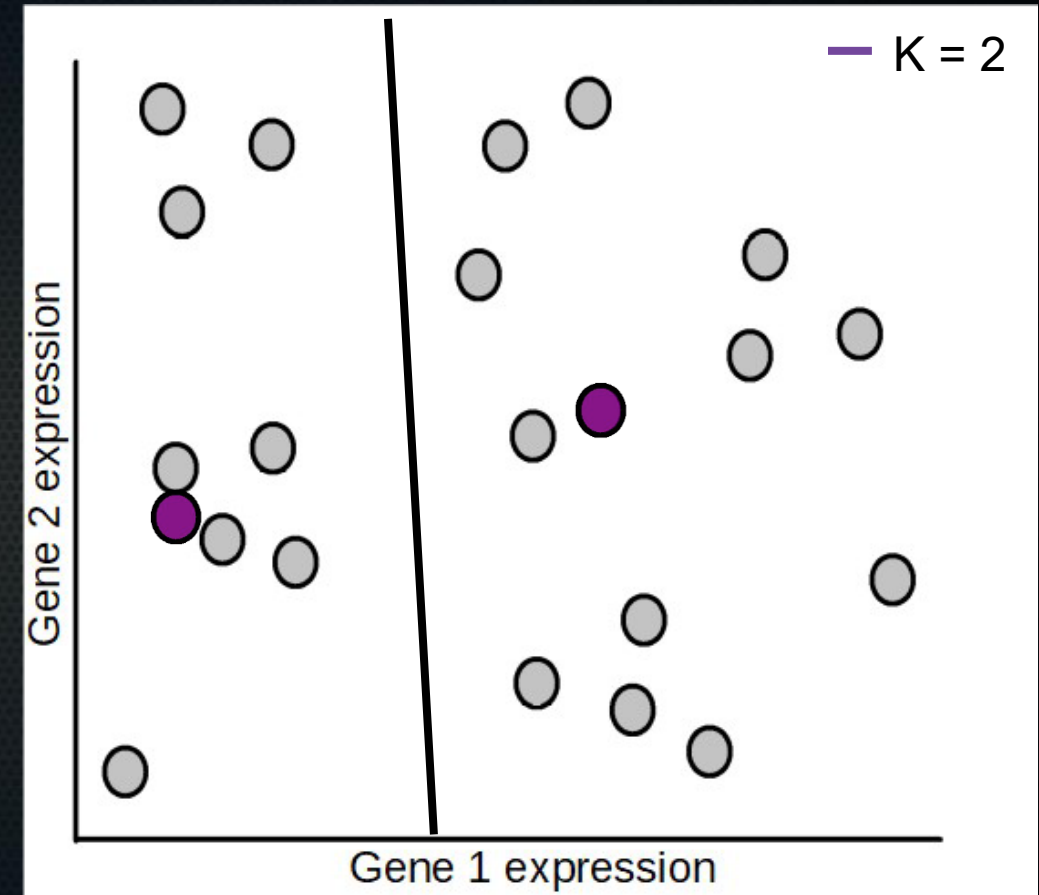
- Iterate until convergence

K-means clustering in practice

- Calculate the distance of each point to each cluster centroid (= cluster prototype)

- Assign each point to the closest prototype.

- Iterate until convergence → now no point changes cluster anymore.

K-means clustering in practice

- Two questions:
  - We start with random points as prototypes, does that matter?
  - How do we choose *k*?

K-means clustering in practice

- Two questions:

  - **We start with random points as prototypes, does that matter?**

  - How do we choose *k*?

K-means clustering in practice

- We start with random points as prototypes, does that matter?

- Yes!

K-means clustering in practice

- We start with random points as prototypes, does that matter?

- Yes!

K-means clustering in practice

- We start with random points as prototypes, does that matter?

- Yes!

K-means clustering in practice

- We start with random points as prototypes, does that matter?

- Yes!

K-means clustering in practice

- We start with random points as prototypes, does that matter?

- Yes!

K-means clustering in practice
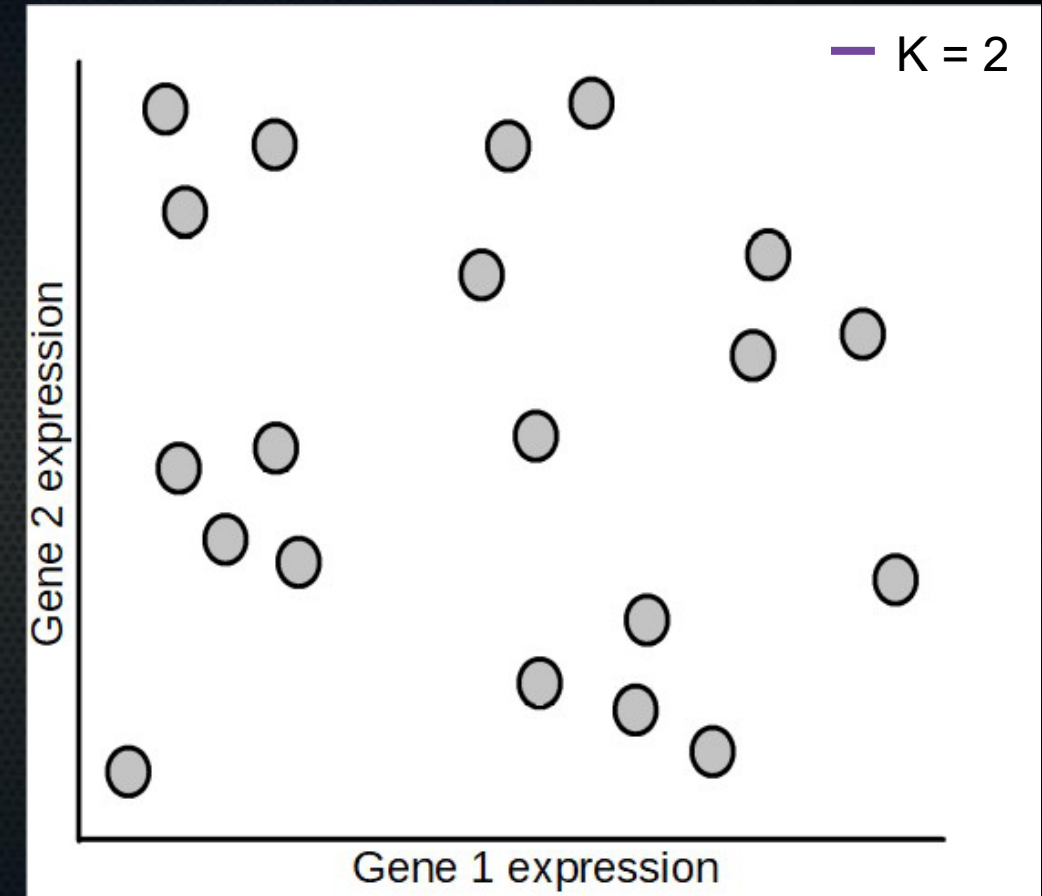
▪ We start with random points as prototypes, does that matter?

▪ Yes!

K-means clustering in practice

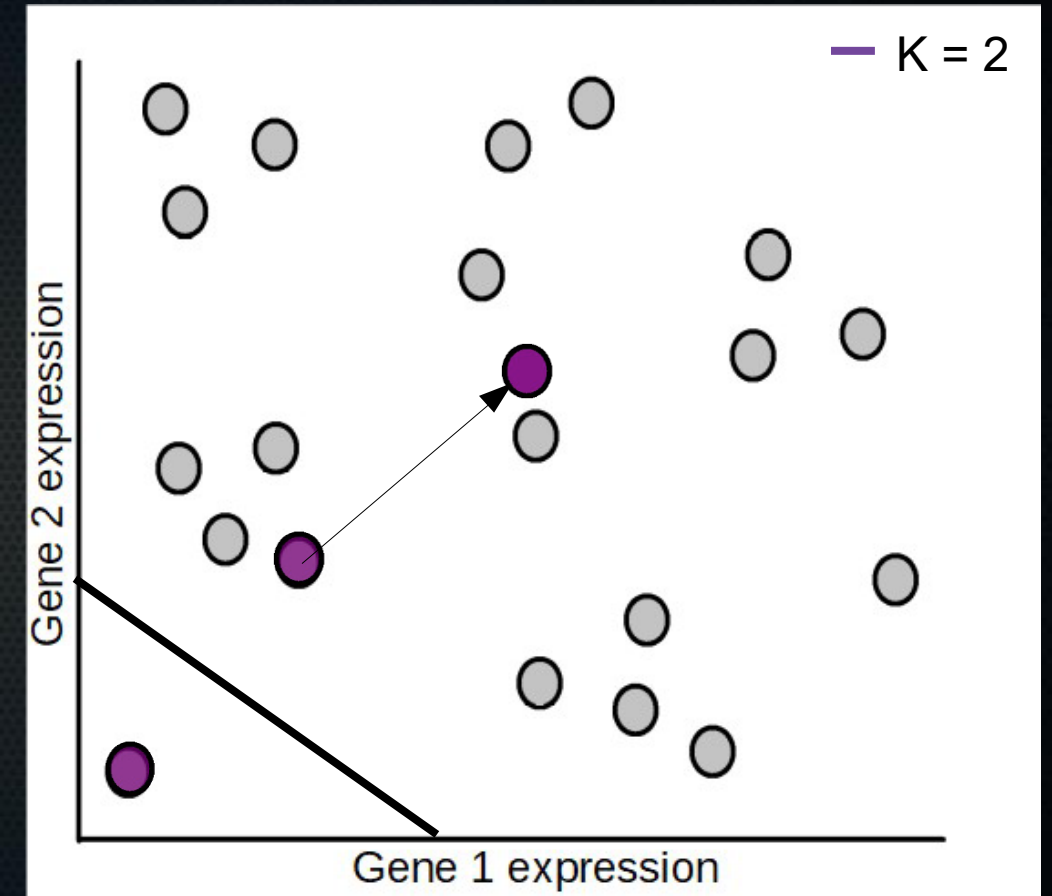- We start with random points as prototypes, does that matter?
- Yes!

K-means clustering in practice

- We start with random points as prototypes, does that matter?
- Yes!
- So what do we do?

K-means clustering in practice

- We start with random points as prototypes, does that matter?
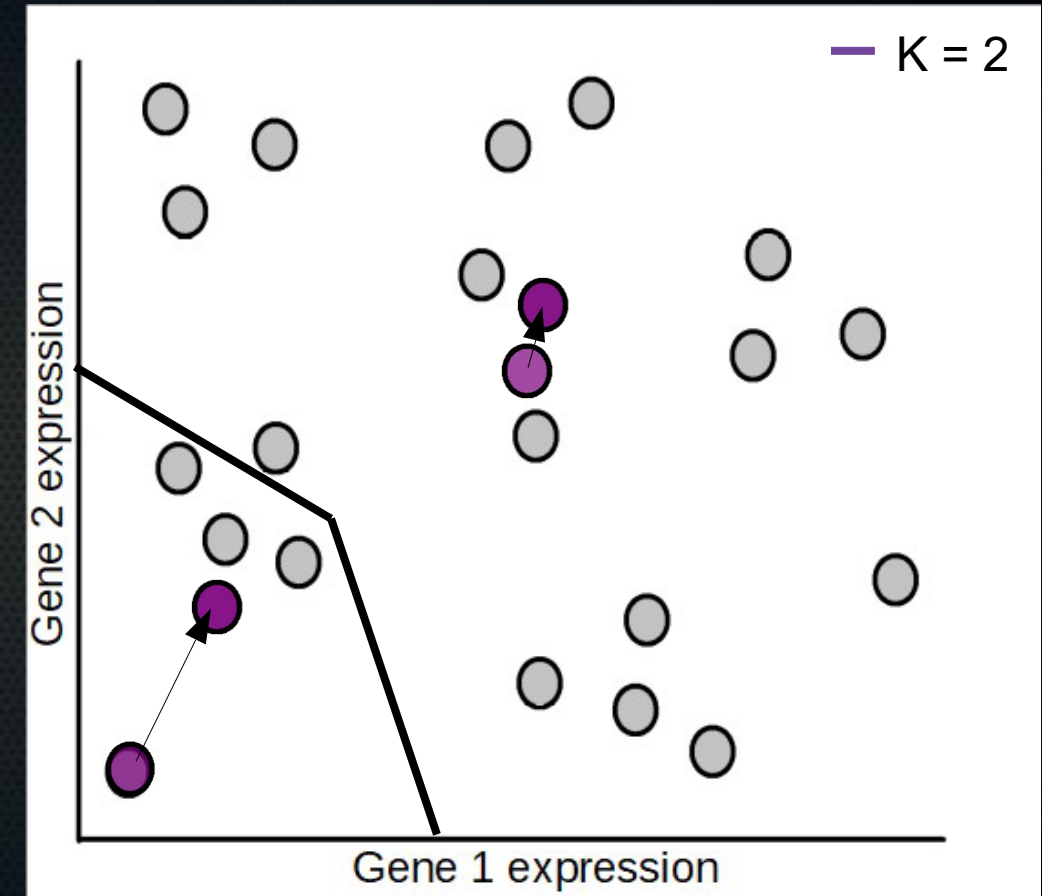
- Yes!

- So what do we do?

  - What do you think?
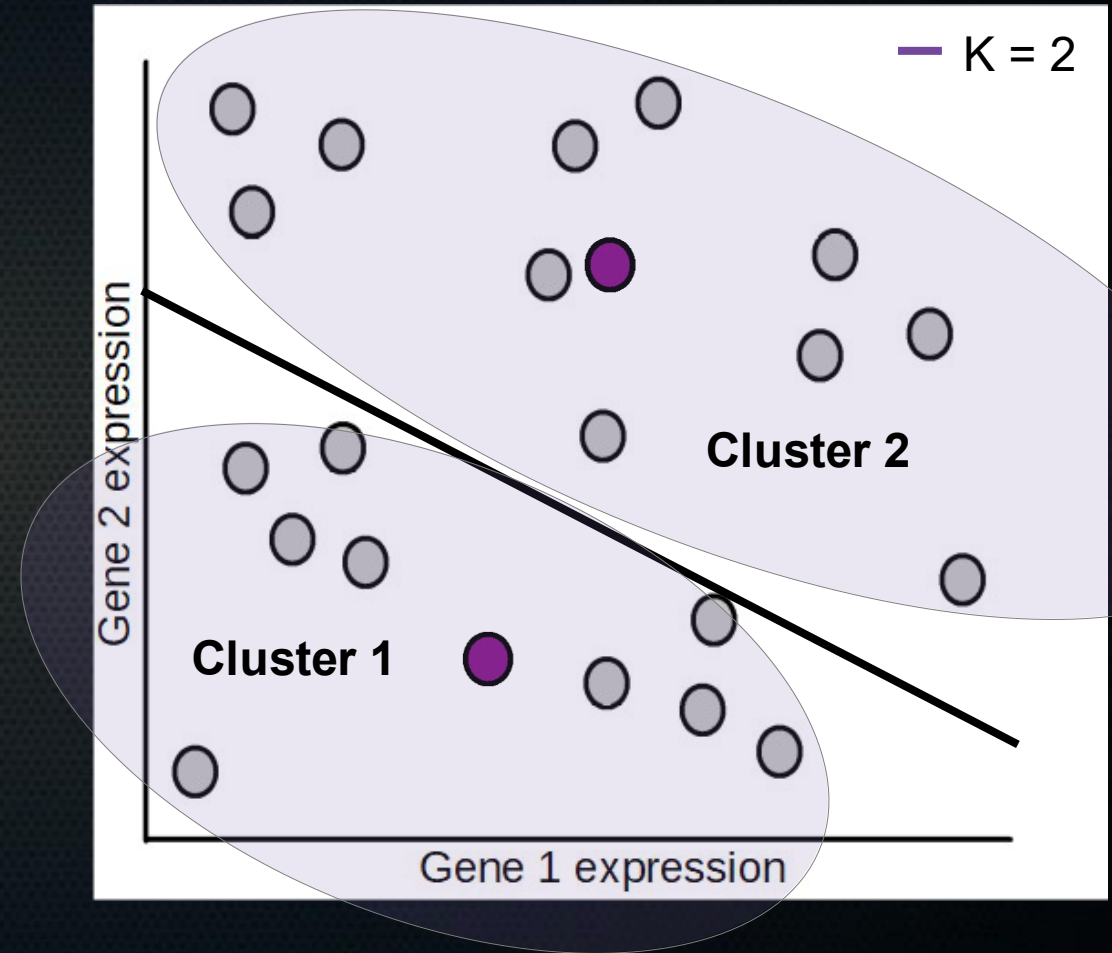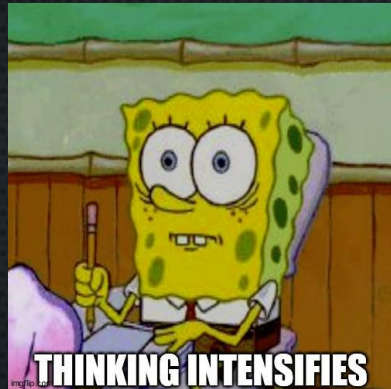
K-means clustering in practice

▪ The *total clustering* has a cost: mean square distance of every point to the cluster centroid of the cluster to which it is assigned
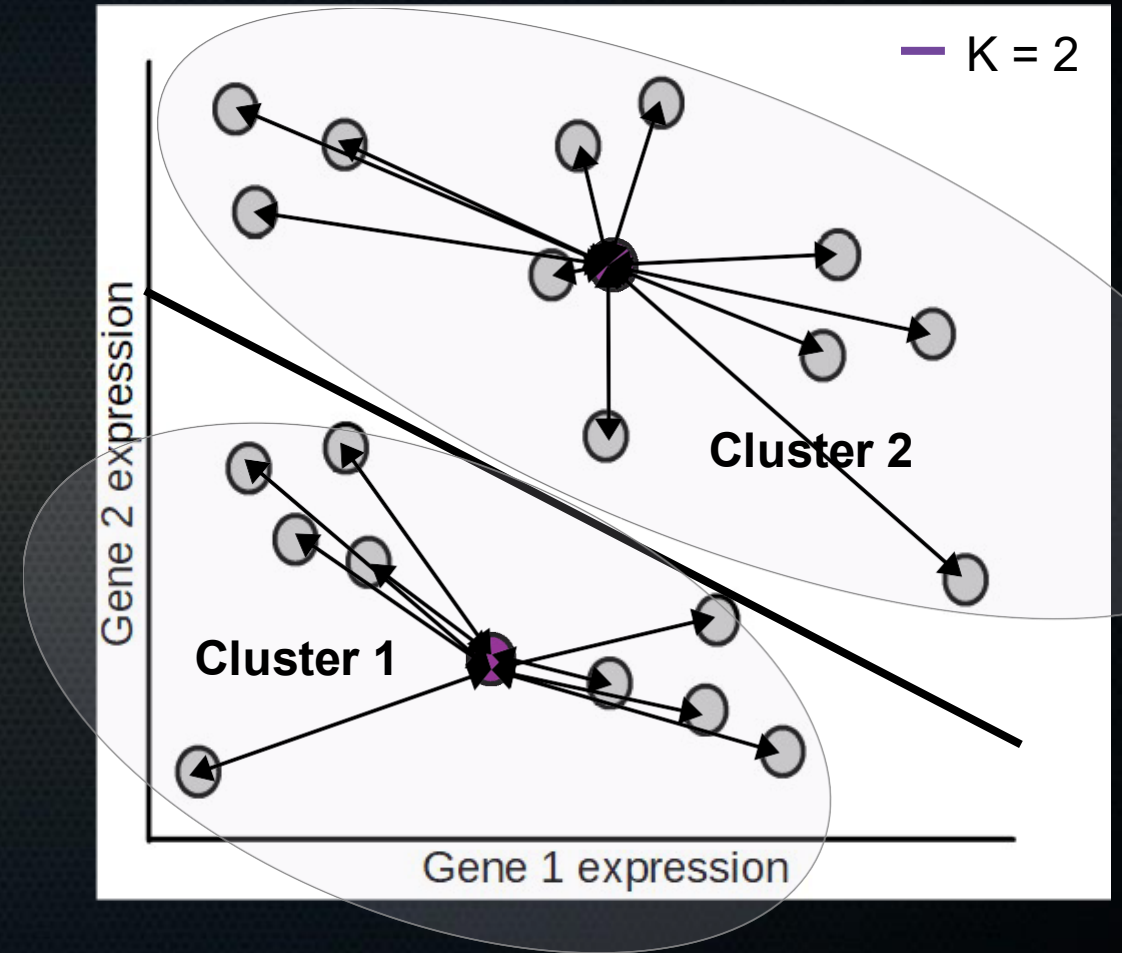
K-means clustering in practice

- The *total clustering* has a cost: mean square distance of every point to the cluster centroid of the cluster to which it is assigned

$$J(\ldots) = \frac{1}{m} \sum_{i=1}^{m} (x^{(i)} - \mu_{c^{(i)}})^2$$

$\underbrace{\qquad\qquad\qquad}_{\text{Distortion}}$
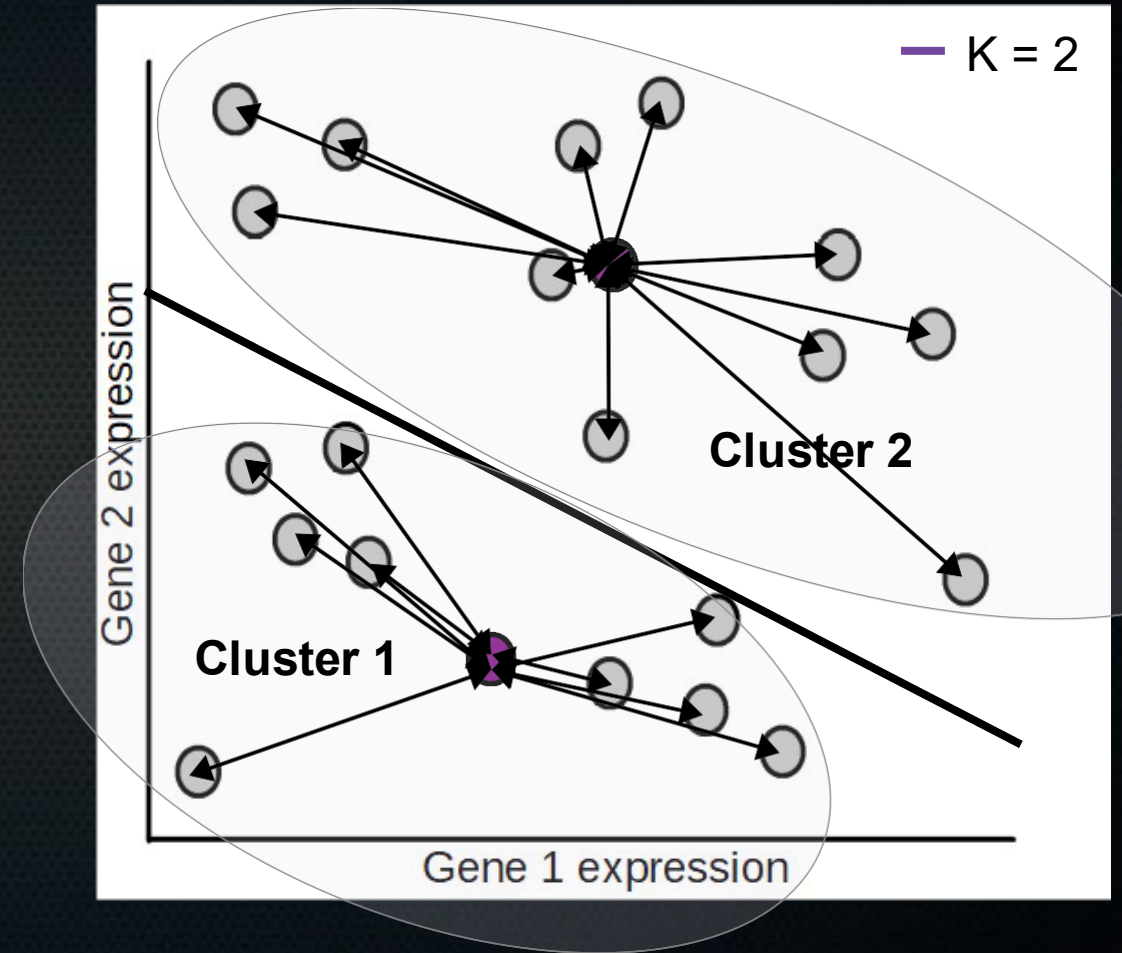
K-means clustering in practice

▪ The *total clustering* has a cost: mean square distance of every point to the cluster centroid of the cluster to which it is assigned

$$J(\ldots) = \frac{1}{m} \sum_{i=1}^{m} (x^{(i)} - \mu_{c^{(i)}})^2$$

▪ We've just seen that this cost depends on initialisation → Do this many times, pick clustering with lowest cost!

K-means clustering in practice

- Two questions:
    - ~~We start with random points as prototypes, does that matter?~~
    - **How do we choose *K*?**
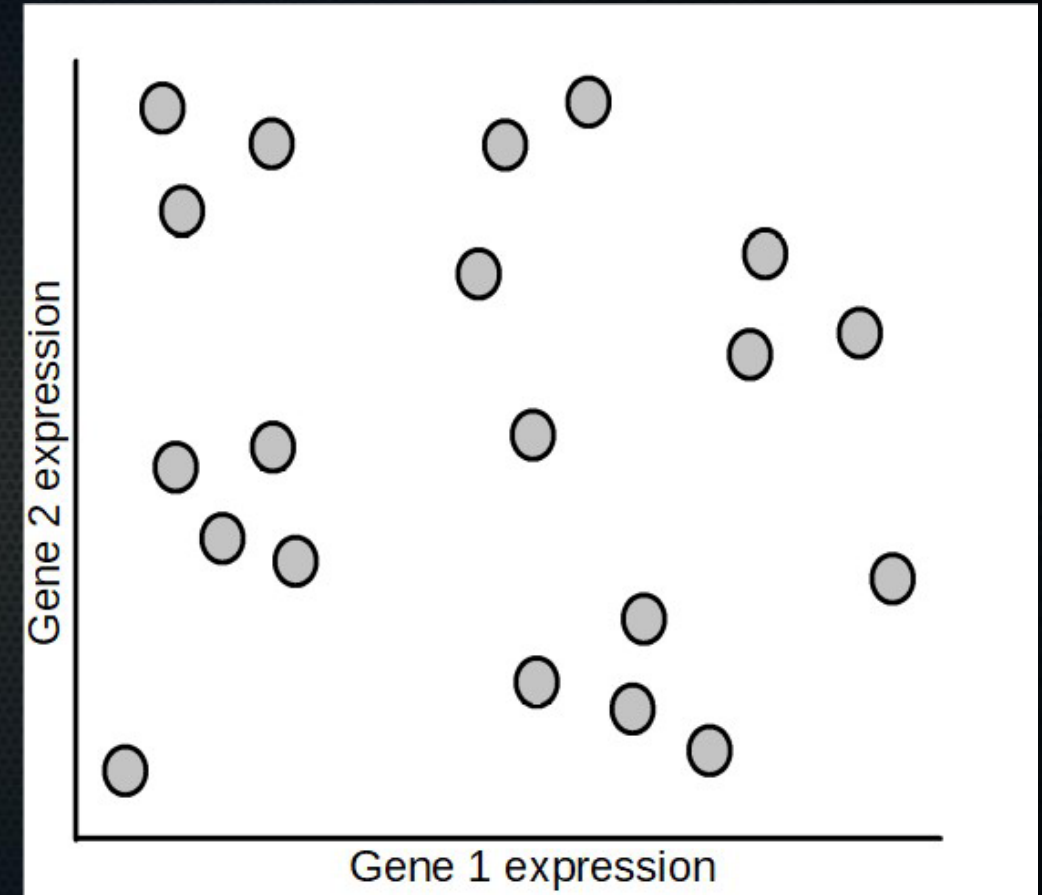
K-means clustering in practice

- How do we choose *K*?

K-means clustering in practice

- How do we choose *K*?

  - There is no correct K, because no correct amount of clusters exists.
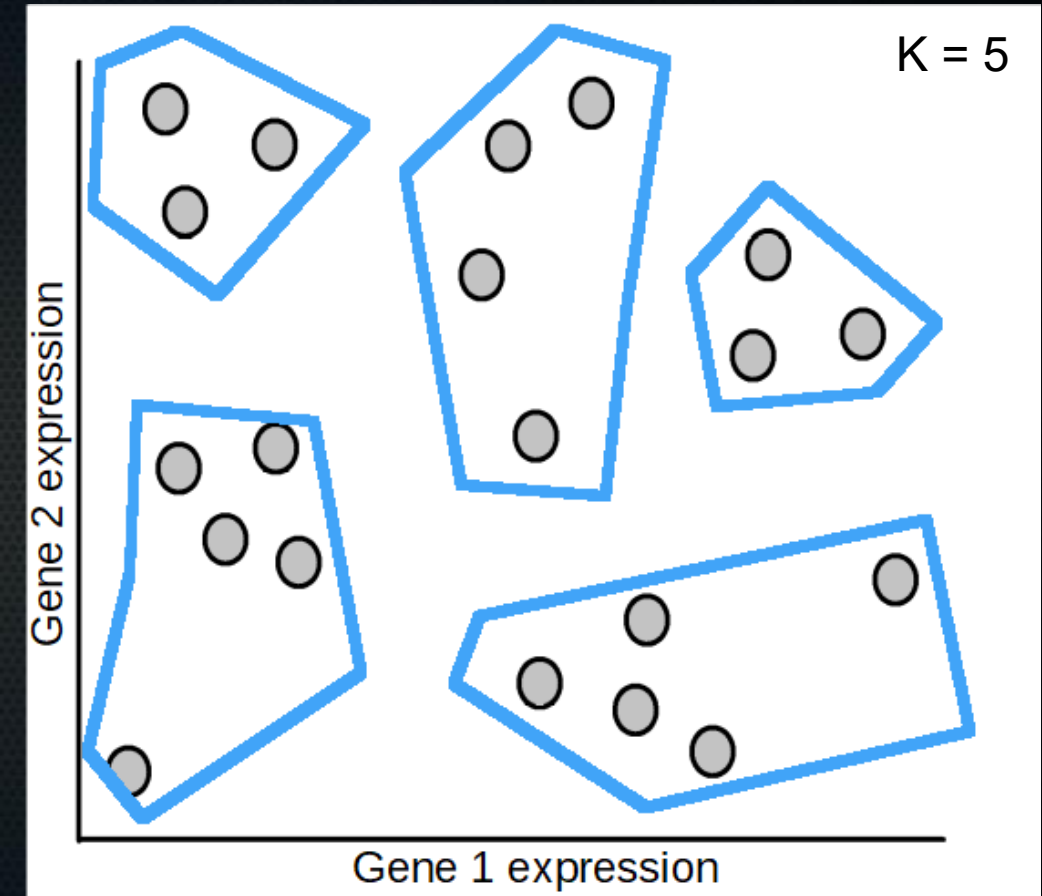
K-means clustering in practice

- How do we choose *K*?
  - There is no correct K, because no correct amount of clusters exists.
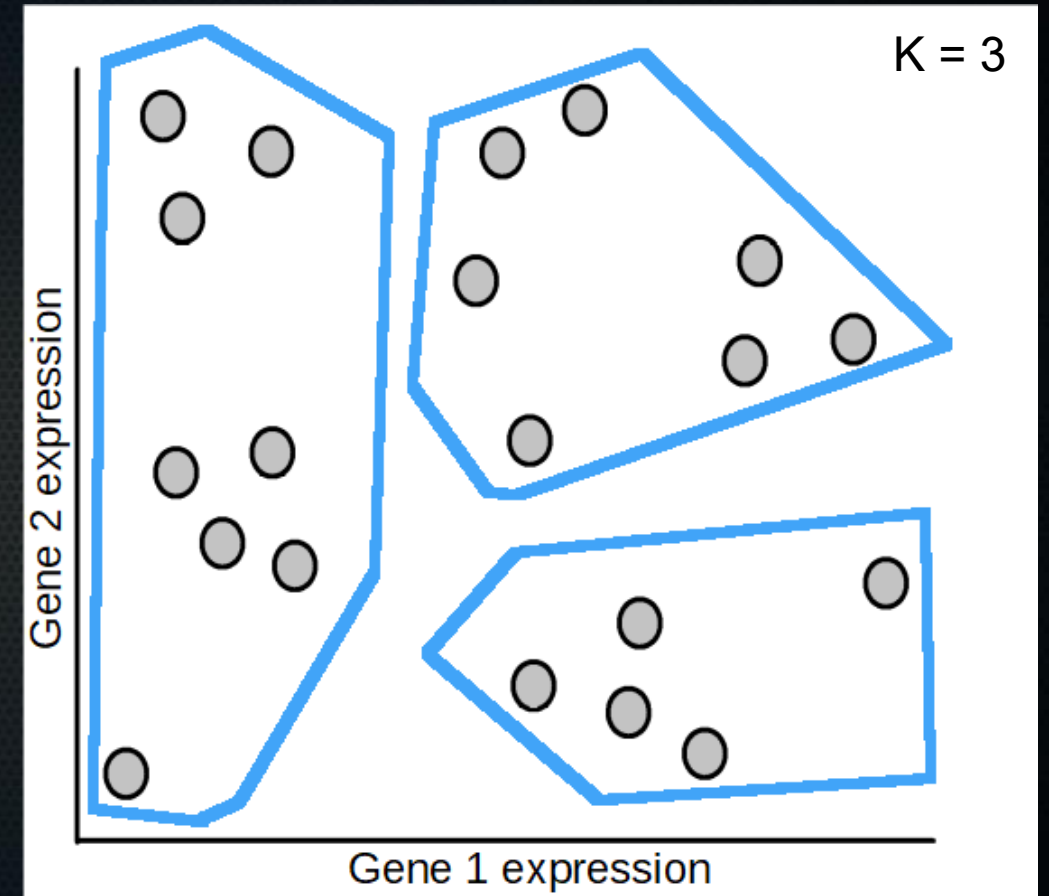
K-means clustering in practice

- How do we choose *K*?

  - There is no correct K, because no correct amount of clusters exists.
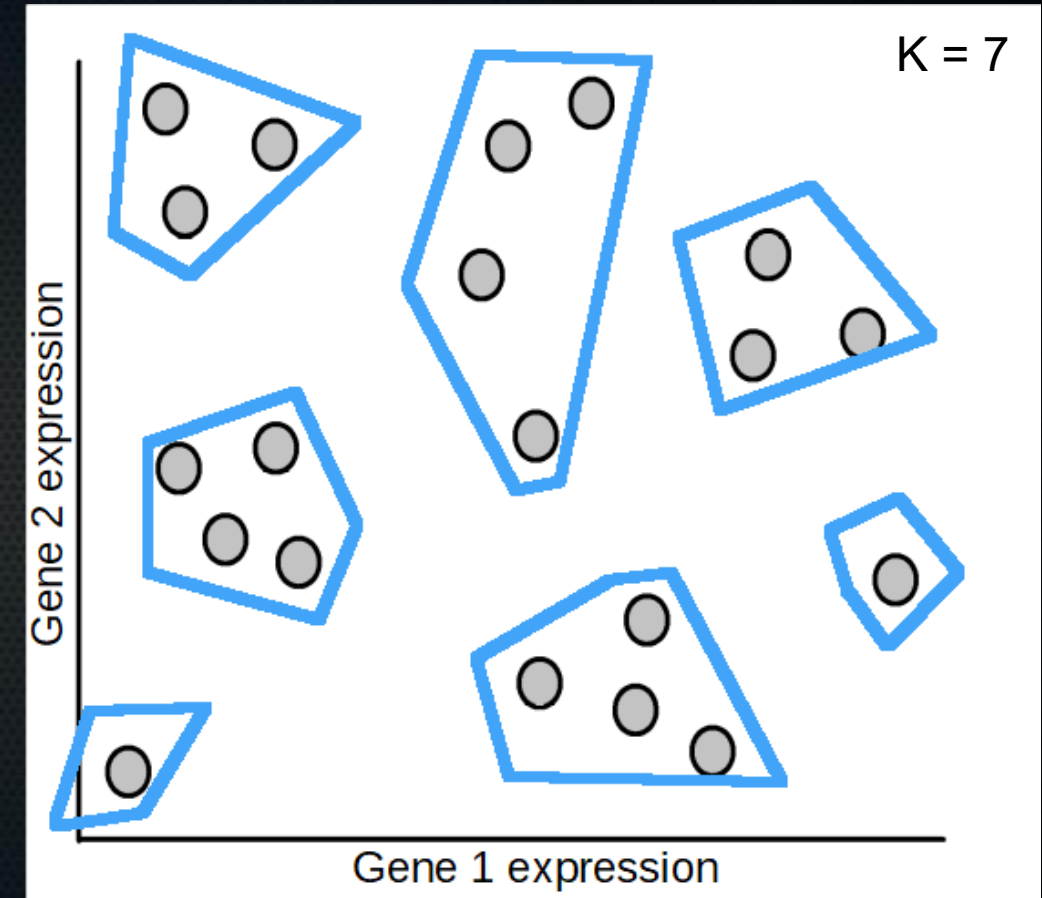
K-means clustering in practice

- How do we choose *K*?

  - There is no correct K, because no correct amount of clusters exists.

- In theory a so-called elbow method, but in practice often doesn't work:
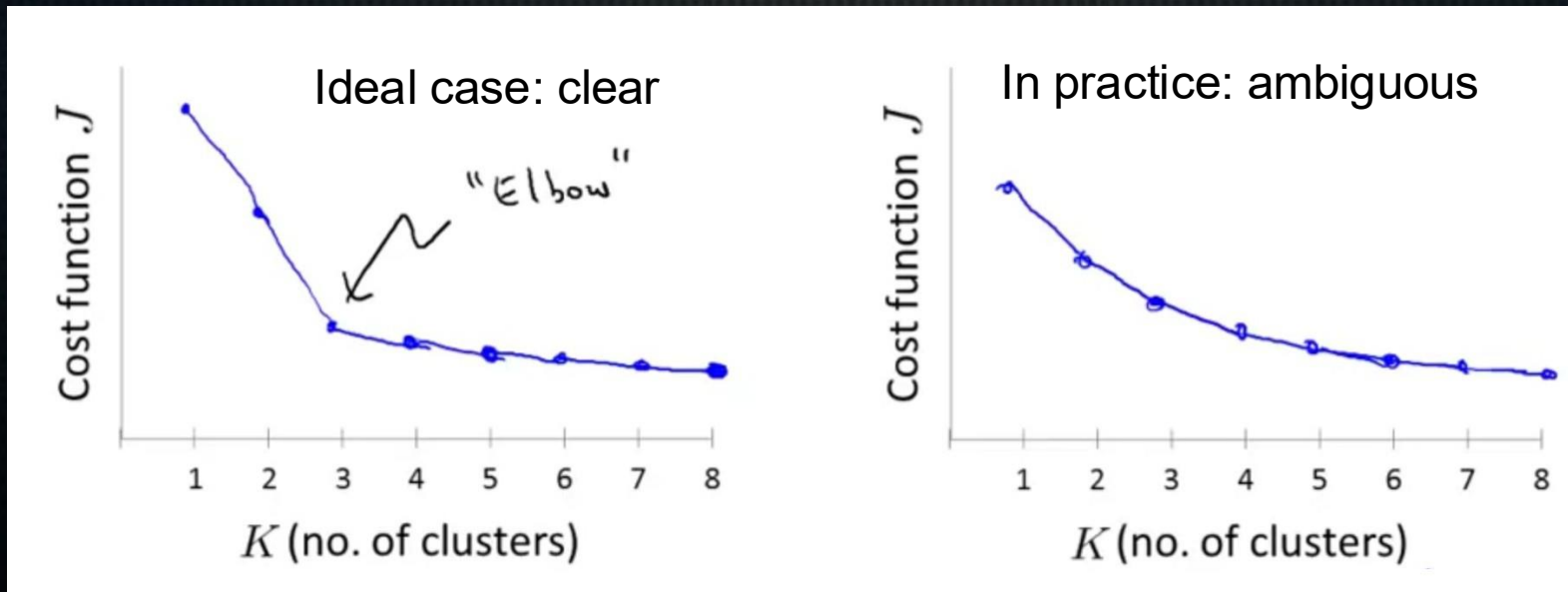
Source: Andrew Ng, Coursera

K-means clustering in practice

- How do we choose *K*?

  - There is no correct K, because no correct amount of clusters exists.

- In practice: done manually. What looks good in (dimension-reduced) visualisation?
  What is useful or manageable?
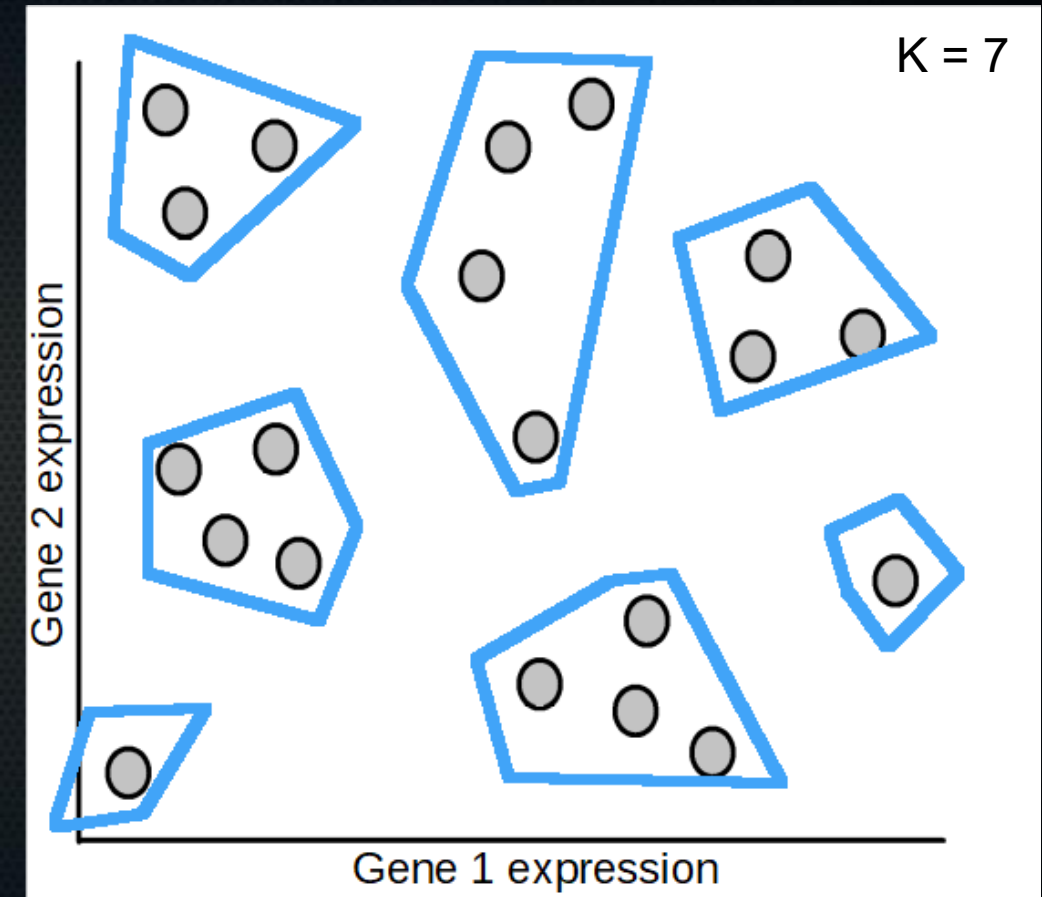
K-means clustering in practice

- How do we choose *K*?

  - There is no correct K, because no correct amount of clusters exists.

- Can be motivated by downstream use:
If I have gene expression data for Alzheimers patients and non-patients →
cluster into 3 or 4 groups to find healthy, diseased, and *pre-clinical diseased* (i.e. something has already gone awry but we don't diagnose that in current clinical practice) → early intervention?
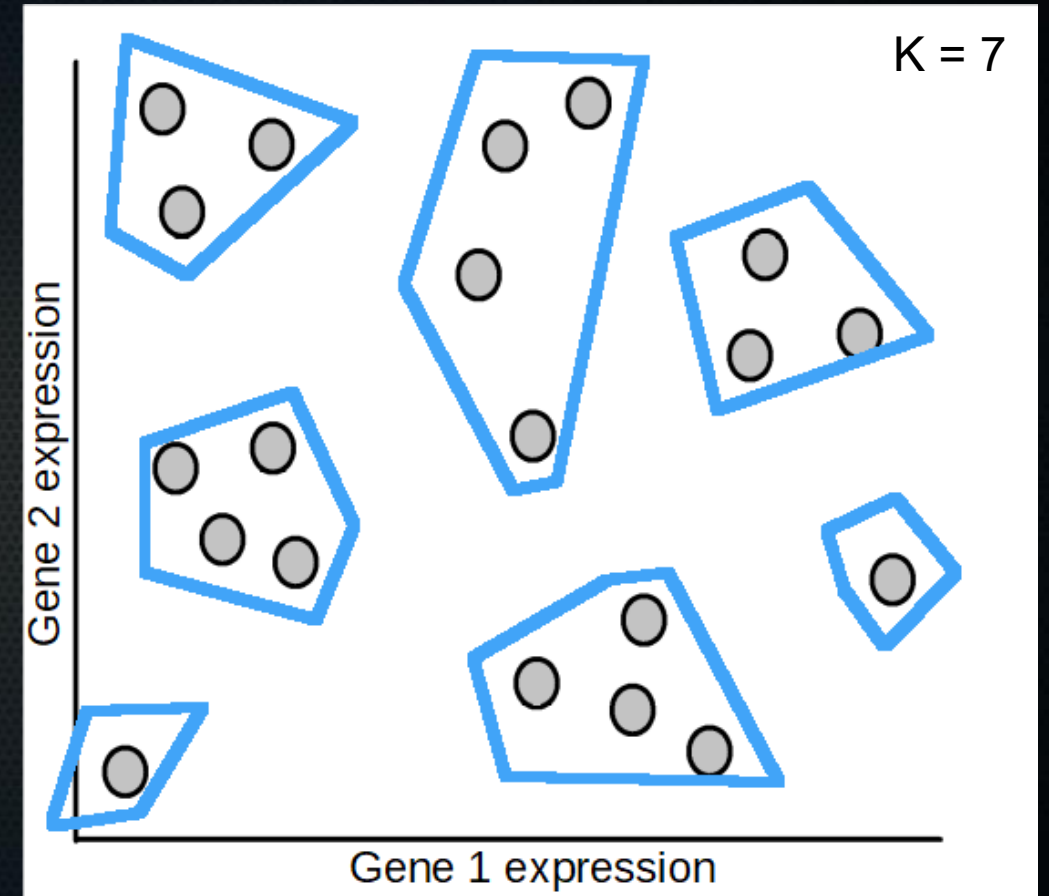
K-means clustering in practice

- Two questions:
  - ~~We start with random points as prototypes, does that matter?~~
  - ~~How do we choose *K*?~~
- All done! → now let's formalise
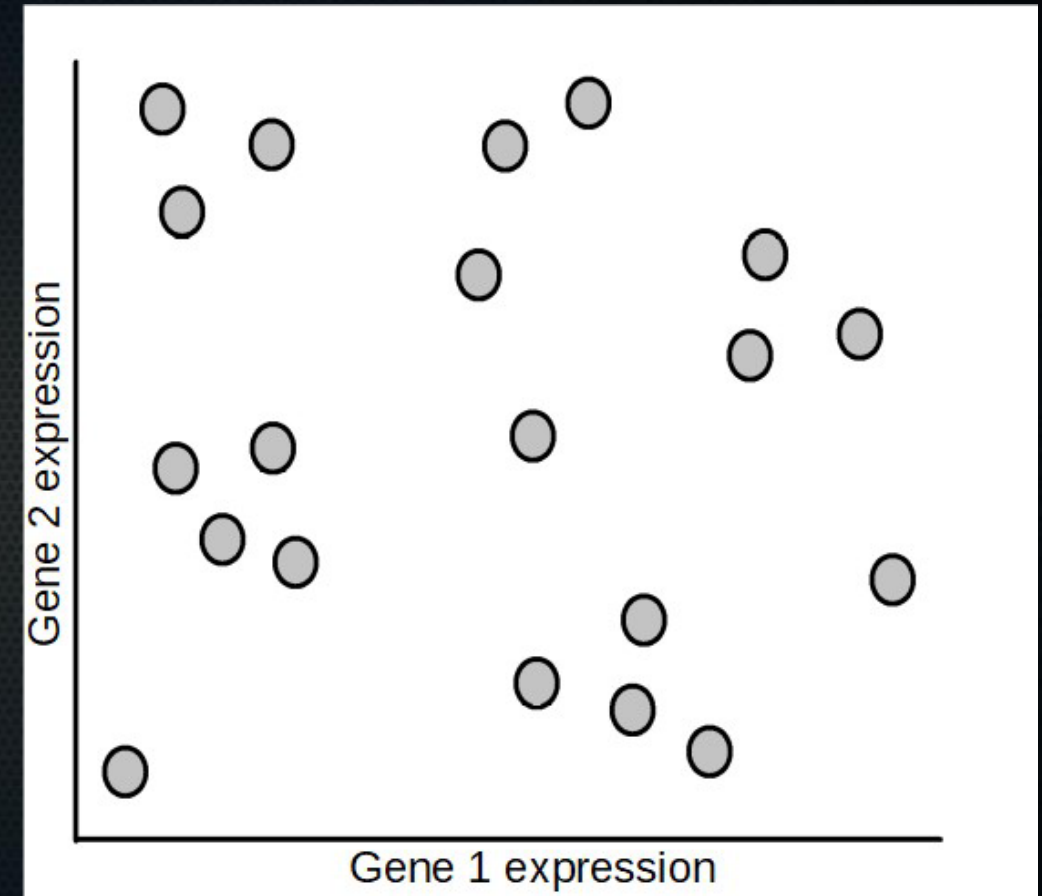
K-means clustering formally

- Formally:

K-means algorithm

Randomly initialize K cluster centroids $\mu_1, \mu_2, \ldots, \mu_K \in \mathbb{R}^n$

Repeat {

    for i = 1 to m

        $c^{(i)}$:= index (from 1 to K) of cluster centroid

                closest to $x^{(i)}$

    for k = 1 to K

        μ_k := average (mean) of points assigned to cluster k

}   ➡ Stop when no change

Source: Andrew Ng, Coursera

K-means clustering formally

- Formally:

K-means optimization objective

$c^{(i)}$ = index of cluster $(1,2,...,K)$ to which example $x^{(i)}$ is currently assigned

μ_k = cluster centroid k $(\mu_k \in \mathbb{R}^n)$ $\qquad k \in \{1,2,...,k\}$

$\mu_{c^{(i)}}$ = cluster centroid of cluster to which example $x^{(i)}$ has been assigned $\qquad x^{(i)} \rightarrow 5 \qquad c^{(i)} = 5 \qquad \mu_{c^{(i)}} = \mu_5$

Source: Andrew Ng, Coursera

78

K-means clustering formally

- Formally:

K-means optimization objective

$c^{(i)}$ = index of cluster (1,2,...,K) to which example $x^{(i)}$ is currently
assigned

μ_k = cluster centroid k ($\mu_k \in \mathbb{R}^n$) $k \in \{1, 2, ..., k\}$

$\mu_{c^{(i)}}$ = cluster centroid of cluster to which example $x^{(i)}$ has been
assigned $x^{(i)} \to 5$ $c^{(i)} = 5$ $\mu_{c^{(i)}} = \mu_5$

Optimization objective:

$$J(c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K) = \frac{1}{m} \sum_{i=1}^{m} ||x^{(i)} - \mu_{c^{(i)}}||^2$$

$$\min_{\substack{c^{(1)}, \ldots, c^{(m)}, \\ \mu_1, \ldots, \mu_K}} J(c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K)$$

Source: Andrew Ng, Coursera

K-means clustering formally

- Formally:

K-means optimization objective

$c^{(i)}$ = index of cluster (1,2,...,K) to which example $x^{(i)}$ is currently assigned

μ_k = cluster centroid k ($\mu_k \in \mathbb{R}^n$)

$k \in \{1, 2, ..., k\}$

$\mu_{c^{(i)}}$ = cluster centroid of cluster to which example $x^{(i)}$ has been assigned

$x^{(i)} \rightarrow 5$ $c^{(i)} = 5$ $\mu_{c^{(i)}} = \mu_5$

Optimization objective:

$$J(c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K) = \frac{1}{m} \sum_{i=1}^{m} \left\| x^{(i)} - \mu_{c^{(i)}} \right\|^2$$

$$\min_{\substack{c^{(1)}, \ldots, c^{(m)}, \\ \mu_1, \ldots, \mu_K}} J(c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K)$$

Source: Andrew Ng, Coursera

K-means clustering formally

- Formally:

$$\|x^{(i)} - \mu_{c^{(i)}}\|^2 \longrightarrow \text{What does this mean?}$$

K-means optimization objective

$c^{(i)}$ = index of cluster (1,2,…,K) to which example $x^{(i)}$ is currently assigned

μ_k = cluster centroid k ($\mu_k \in \mathbb{R}^n$) $k \in \{1,2,..,K\}$

$\mu_{c^{(i)}}$ = cluster centroid of cluster to which example $x^{(i)}$ has been assigned $x^{(i)} \to 5$ $c^{(i)} = 5$ $\mu_{c^{(i)}} = \mu_5$

Optimization objective:

$$J(c^{(1)},\ldots,c^{(m)},\mu_1,\ldots,\mu_K) = \frac{1}{m}\sum_{i=1}^{m}\|x^{(i)} - \mu_{c^{(i)}}\|^2$$

$$\min_{\substack{c^{(1)},\ldots,c^{(m)}, \\ \mu_1,\ldots,\mu_K}} J(c^{(1)},\ldots,c^{(m)},\mu_1,\ldots,\mu_K)$$

K-means clustering formally

- Formally:

$$\left\|x^{(i)} - \mu_{c^{(i)}}\right\|^2 \longrightarrow \text{What does this mean?}$$

- Some linear algebra notation. Called L2-norm. Means: take the square of each element in a vector, sum that, take the square root.

K-means optimization objective

$c^{(i)}$ = index of cluster (1,2,...,K) to which example $x^{(i)}$ is currently assigned

μ_k = cluster centroid k ($\mu_k \in \mathbb{R}^n$) $k \in \{1, 2, ..., k\}$

$\mu_{c^{(i)}}$ = cluster centroid of cluster to which example $x^{(i)}$ has been assigned $x^{(i)} \to 5$ $c^{(i)} = 5$ $\mu_{c^{(i)}} = \mu_5$

Optimization objective:

$$J(c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K) = \frac{1}{m} \sum_{i=1}^{m} \left\|x^{(i)} - \mu_{c^{(i)}}\right\|^2$$

$$\min_{\substack{c^{(1)}, \ldots, c^{(m)}, \\ \mu_1, \ldots, \mu_K}} J(c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K)$$

K-means clustering formally

- Formally:

$$\left\|x^{(i)} - \mu_{c^{(i)}}\right\|^2$$ ⟶ What does this mean?

- Really just the Euclidean distance that works for any amount of dimensions (features).

K-means optimization objective

$c^{(i)}$ = index of cluster (1,2,…,K) to which example $x^{(i)}$ is currently assigned

μ_k = cluster centroid k $(\mu_k \in \mathbb{R}^n)$ $k \in \{1,2,..,k\}$

$\mu_{c^{(i)}}$ = cluster centroid of cluster to which example $x^{(i)}$ has been assigned $x^{(i)} \to 5$ $c^{(i)} = 5$ $\mu_{c^{(i)}} = \mu_5$

Optimization objective:

$$J(c^{(1)},\dots,c^{(m)},\mu_1,\dots,\mu_K) = \frac{1}{m}\sum_{i=1}^{m}\left\|x^{(i)} - \mu_{c^{(i)}}\right\|^2$$

$$\min_{\substack{c^{(1)},\dots,c^{(m)}, \\ \mu_1,\dots,\mu_K}} J(c^{(1)},\dots,c^{(m)},\mu_1,\dots,\mu_K)$$

K-means clustering formally

- Formally:

$$\|x^{(i)} - \mu_{c^{(i)}}\|^2$$ → What does this mean?

- Example: let's say we have 5 genes

K-means optimization objective

$c^{(i)}$ = index of cluster (1,2,…,K) to which example $x^{(i)}$ is currently assigned

μ_k = cluster centroid k ($\mu_k \in \mathbb{R}^n$) $k \in \{1,2,..,k\}$

$\mu_{c^{(i)}}$ = cluster centroid of cluster to which example $x^{(i)}$ has been assigned $x^{(i)} \to 5$ $c^{(i)} = 5$ $\mu_{c^{(i)}} = \mu_5$

Optimization objective:

$$J(c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K) = \frac{1}{m} \sum_{i=1}^{m} \|x^{(i)} - \mu_{c^{(i)}}\|^2$$

$$\min_{\substack{c^{(1)},\ldots,c^{(m)}, \\ \mu_1,\ldots,\mu_K}} J(c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K)$$

K-means clustering formally

- Formally:

$$\left\| x^{(i)} - \mu_{c^{(i)}} \right\|^2 \longrightarrow \text{What does this mean?}$$

- Example: let's say we have 5 genes

K-means optimization objective

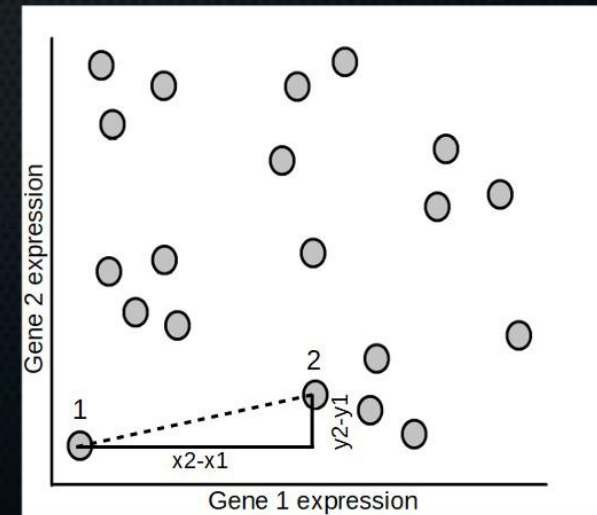$c^{(i)}$ = index of cluster (1,2,...,K) to which example $x^{(i)}$ is currently assigned

μ_k = cluster centroid k ($\mu_k \in \mathbb{R}^n$) $k \in \{1,2,...,k\}$

$\mu_{c^{(i)}}$ = cluster centroid of cluster to which example $x^{(i)}$ has been assigned $x^{(i)} \to 5$ $c^{(i)} = 5$ $\mu_{c^{(i)}} = \mu_5$

Optimization objective:

$$J(c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K) = \frac{1}{m} \sum_{i=1}^{m} \| x^{(i)} - \mu_{c^{(i)}} \|^2$$

$$\min_{\substack{c^{(1)}, \ldots, c^{(m)}, \\ \mu_1, \ldots, \mu_K}} J(c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K)$$

Gene 1
Gene 2

$$\begin{bmatrix} 3 \\ 4 \\ -2 \\ 9 \\ 3 \end{bmatrix}$$

Gene 5

Expression of genes
in a sample

K-means clustering formally

- Formally:

$$\|x^{(i)} - \mu_{c^{(i)}}\|^2 \longrightarrow$$ What does this mean?

- Example: let's say we have 5 genes

Gene 1 $\begin{bmatrix} 3 \\ 4 \\ -2 \\ 9 \\ 3 \end{bmatrix}$ $\begin{bmatrix} 4.48 \\ 2.6 \\ 8 \\ 10.3 \\ 4.22 \end{bmatrix}$

Gene 2

Gene 5

Mean expression of genes for the cluster
that sample is currently assigned to

K-means optimization objective

$c^{(i)}$ = index of cluster $(1,2,...,K)$ to which example $x^{(i)}$ is currently assigned

μ_k = cluster centroid k ($\mu_k \in \mathbb{R}^n$) $k \in \{1,2,...,K\}$

$\mu_{c^{(i)}}$ = cluster centroid of cluster to which example $x^{(i)}$ has been assigned $x^{(i)} \to 5$ $c^{(i)} = 5$ $\mu_{c^{(i)}} = \mu_5$

Optimization objective:

$$J(c^{(1)},...,c^{(m)}, \mu_1,...,\mu_K) = \frac{1}{m}\sum_{i=1}^{m}\|x^{(i)} - \mu_{c^{(i)}}\|^2$$

$$\min_{\substack{c^{(1)},...,c^{(m)}, \\ \mu_1,...,\mu_K}} J(c^{(1)},...,c^{(m)}, \mu_1,...,\mu_K)$$

K-means clustering formally

- Formally:

$$\left\| x^{(i)} - \mu_{c^{(i)}} \right\|^2$$ ⟶ What does this mean?

- Example: let's say we have 5 genes

K-means optimization objective

$c^{(i)}$ = index of cluster (1,2,...,K) to which example $x^{(i)}$ is currently assigned

μ_k = cluster centroid k ($\mu_k \in \mathbb{R}^n$) $k \in \{1,2,..,k\}$

$\mu_{c^{(i)}}$ = cluster centroid of cluster to which example $x^{(i)}$ has been assigned $x^{(i)} \to 5$ $c^{(i)} = 5$ $\mu_{c^{(i)}} = \mu_5$

Optimization objective:

$$J(c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K) = \frac{1}{m} \sum_{i=1}^{m} \| x^{(i)} - \mu_{c^{(i)}} \|^2$$

$$\min_{\substack{c^{(1)},\ldots,c^{(m)}, \\ \mu_1,\ldots,\mu_K}} J(c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K)$$

Gene 1
Gene 2

Gene 5

$$\begin{bmatrix} 3 \\ 4 \\ -2 \\ 9 \\ 3 \end{bmatrix} - \begin{bmatrix} 4.48 \\ 2.6 \\ 8 \\ 10.3 \\ 4.22 \end{bmatrix} \begin{bmatrix} 1.48 \\ 1.4 \\ -10 \\ -1.3 \\ -1.22 \end{bmatrix}$$

K-means clustering formally

- Formally:

$$\|x^{(i)} - \mu_{c^{(i)}}\|^2 \longrightarrow \text{What does this mean?}$$

- Example: let's say we have 5 genes

K-means optimization objective

- $c^{(i)}$ = index of cluster (1,2,…,K) to which example $x^{(i)}$ is currently assigned
- μ_k = cluster centroid k ($\mu_k \in \mathbb{R}^n$) $\qquad k \in \{1,2,...,k\}$
- $\mu_{c^{(i)}}$ = cluster centroid of cluster to which example $x^{(i)}$ has been assigned $\qquad x^{(i)} \to 5 \qquad c^{(i)} = 5 \qquad \mu_{c^{(i)}} = \mu_5$

Optimization objective:

$$J(c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K) = \frac{1}{m}\sum_{i=1}^{m} \|x^{(i)} - \mu_{c^{(i)}}\|^2$$

$$\min_{\substack{c^{(1)},\ldots,c^{(m)}, \\ \mu_1,\ldots,\mu_K}} J(c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K)$$

Gene 1
Gene 2

Gene 5

$$\begin{bmatrix} 3 \\ 4 \\ -2 \\ 9 \\ 3 \end{bmatrix} - \begin{bmatrix} 4.48 \\ 2.6 \\ 8 \\ 10.3 \\ 4.22 \end{bmatrix} \begin{bmatrix} 1.48 \\ 1.4 \\ -10 \\ -1.3 \\ -1.22 \end{bmatrix} \xrightarrow{\text{Square}} \begin{bmatrix} 2.19 \\ 1.96 \\ 100 \\ 1.69 \\ 1.49 \end{bmatrix}$$

K-means clustering formally

- Formally:

$$\|x^{(i)} - \mu_{c^{(i)}}\|^2 \longrightarrow \text{What does this mean?}$$

- Example: let's say we have 5 genes

Gene 1
Gene 2
Gene 5

$$\begin{bmatrix} 3 \\ 4 \\ -2 \\ 9 \\ 3 \end{bmatrix} - \begin{bmatrix} 4.48 \\ 2.6 \\ 8 \\ 10.3 \\ 4.22 \end{bmatrix} \begin{bmatrix} 1.48 \\ 1.4 \\ -10 \\ -1.3 \\ -1.22 \end{bmatrix} \xrightarrow{\text{Square}} \begin{bmatrix} 2.19 \\ 1.96 \\ 100 \\ 1.69 \\ 1.49 \end{bmatrix} \xrightarrow{\text{Sum}} 107.33$$

K-means clustering formally

- Formally:

$$\|x^{(i)} - \mu_{c^{(i)}}\|^2 \longrightarrow$$ What does this mean?

- Example: let's say we have 5 genes

K-means optimization objective

$c^{(i)}$ = index of cluster $(1,2,...,K)$ to which example $x^{(i)}$ is currently assigned

μ_k = cluster centroid k $(\mu_k \in \mathbb{R}^n)$ $k \in \{1,2,...,K\}$

$\mu_{c^{(i)}}$ = cluster centroid of cluster to which example $x^{(i)}$ has been assigned $x^{(i)} \to 5$ $c^{(i)} = 5$ $\mu_{c^{(i)}} = \mu_5$

Optimization objective:

$$J(c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K) = \frac{1}{m} \sum_{i=1}^{m} \|x^{(i)} - \mu_{c^{(i)}}\|^2$$

$$\min_{\substack{c^{(1)}, \ldots, c^{(m)}, \\ \mu_1, \ldots, \mu_K}} J(c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K)$$

$$
\begin{array}{l}
\text{Gene 1} \\
\text{Gene 2} \\
\\
\\
\text{Gene 5}
\end{array}
\begin{bmatrix} 3 \\ 4 \\ -2 \\ 9 \\ 3 \end{bmatrix}
-
\begin{bmatrix} 4.48 \\ 2.6 \\ 8 \\ 10.3 \\ 4.22 \end{bmatrix}
=
\begin{bmatrix} 1.48 \\ 1.4 \\ -10 \\ -1.3 \\ -1.22 \end{bmatrix}
\xrightarrow{\text{Square}}
\begin{bmatrix} 2.19 \\ 1.96 \\ 100 \\ 1.69 \\ 1.49 \end{bmatrix}
\xrightarrow{\text{Sum}}
107.33
$$

Square root → 10.36

K-means clustering formally

- Formally:

K-means optimization objective

$c^{(i)}$ = index of cluster (1,2,...,K) to which example $x^{(i)}$ is currently assigned

μ_k = cluster centroid k ($\mu_k \in \mathbb{R}^n$) $k \in \{1, 2, .., k\}$

$\mu_{c^{(i)}}$ = cluster centroid of cluster to which example $x^{(i)}$ has been assigned $x^{(i)} \rightarrow \underline{5}$ $\underline{c^{(i)} = 5}$ $\underline{\mu_{c^{(i)}}} = \mu_5$

Optimization objective:

$$J(c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K) = \frac{1}{m} \sum_{i=1}^{m} ||x^{(i)} - \mu_{c^{(i)}}||^2$$

$$\min_{\substack{c^{(1)}, \ldots, c^{(m)}, \\ \mu_1, \ldots, \mu_K}} J(c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K)$$
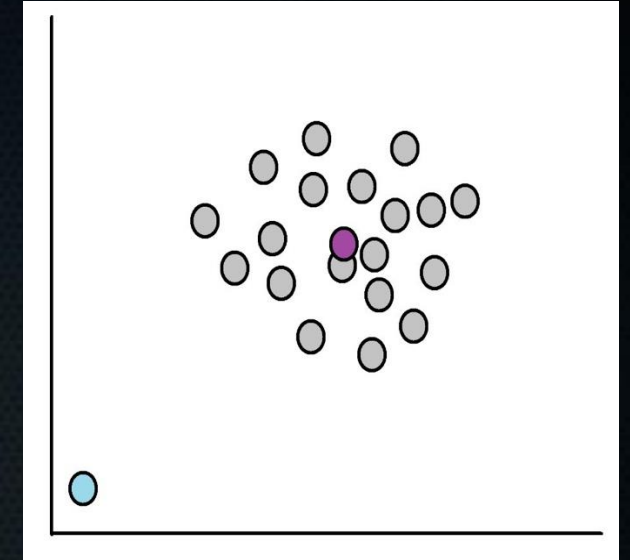
Minimise the mean squared distance of every point to the centroid of the cluster it is assigned to
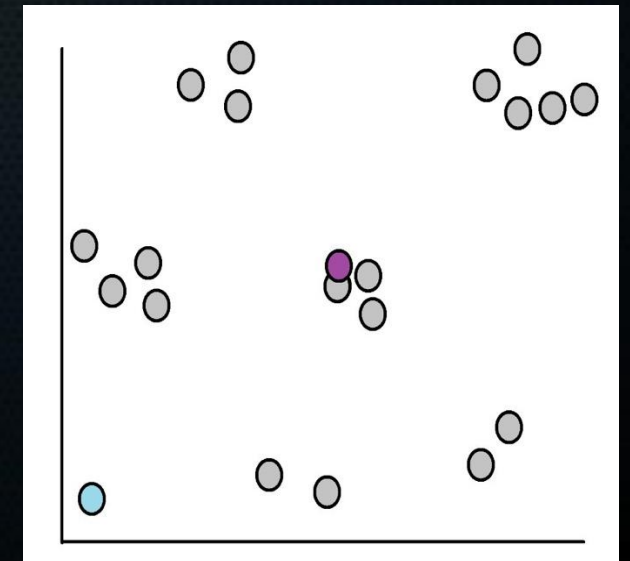
K-means extension

▪ Can take into account spread, rather than solely distance:

Not inclined to see the blue dot as part of the cluster

Same centroid, but more inclined to assign it to that cluster because of spread

K-means extension

- Can take into account spread, rather than solely distance

- Can take into account *covariance*: if gene A expression increases with gene B expression, they co-vary.

- Distance metric taking both into account: *Mahalanobis distance* (yes, really)
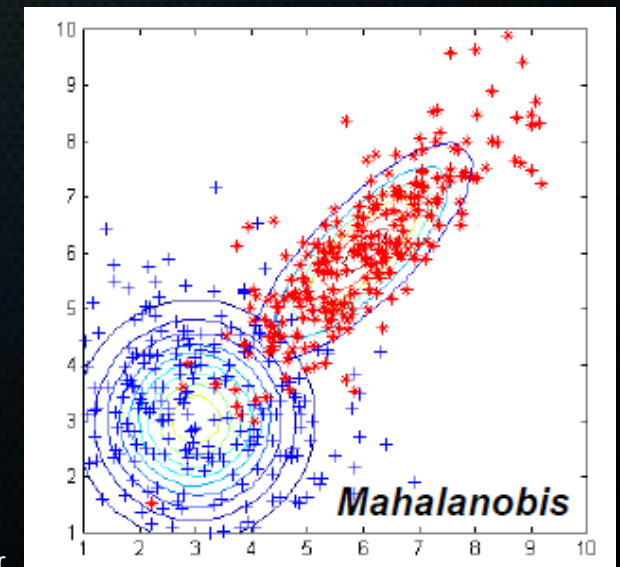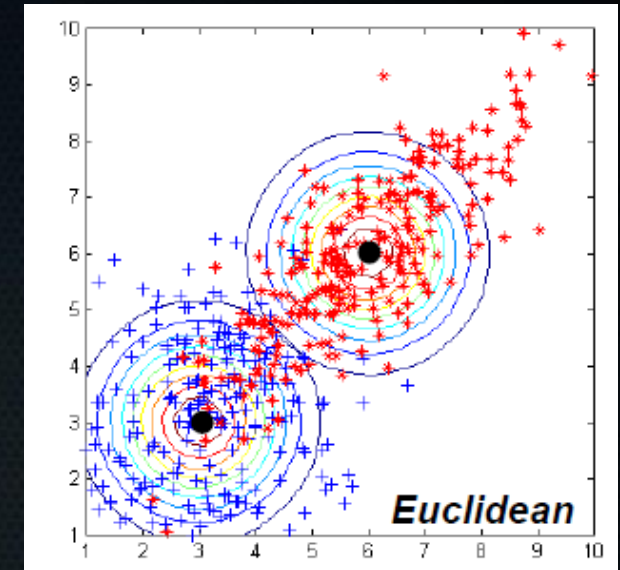
K-means clustering summary

- Works by randomly choosing K data points as cluster prototypes (centroids) and assigning each data point to a cluster.

- Then: iteratively update cluster centroids and assign points. Stop when no change.

- Depends on random initialisation: run many times, pick clustering with lowest cost (lowest distortion).

- Picking K non-trivial.

The other way around: bisectional K-means

- Start from one big cluster.

- Each time, randomly split the cluster with the highest SSE.

$$to \ split = cluster \ for \ which \ (x - \mu)^2 == maximal$$
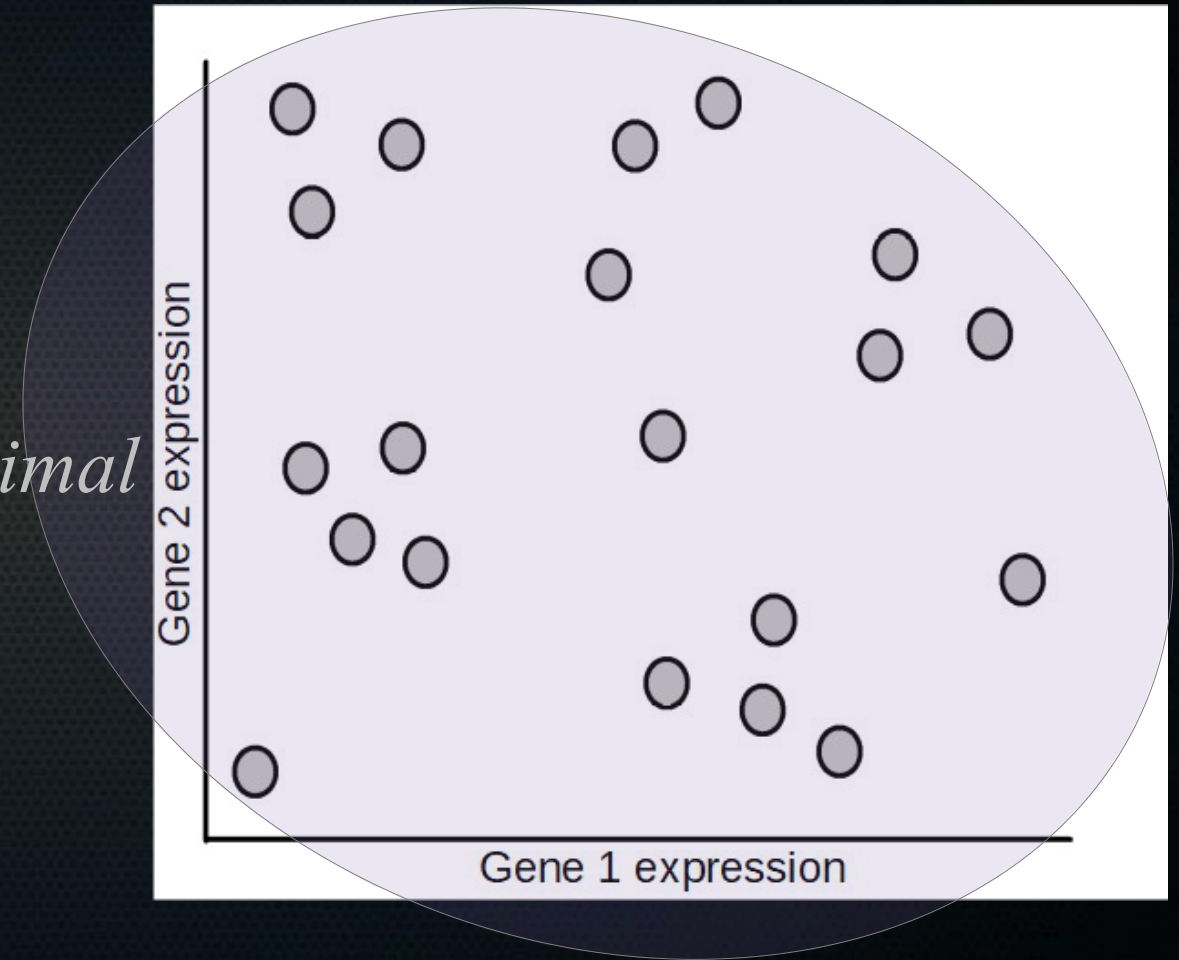
The other way around: bisectional K-means

- Start from one big cluster.

- Each time, randomly split the cluster with the highest SSE.

$$to\ split = cluster\ for\ which\ (x - \mu)^2 == maximal$$

- Done by selecting random new centroids *n* times and then picking the two that lower distortion most.

$$J(\ldots) = \frac{1}{m} \sum_{i=1}^{m} (x^{(i)} - \mu_{c^{(i)}})^2$$
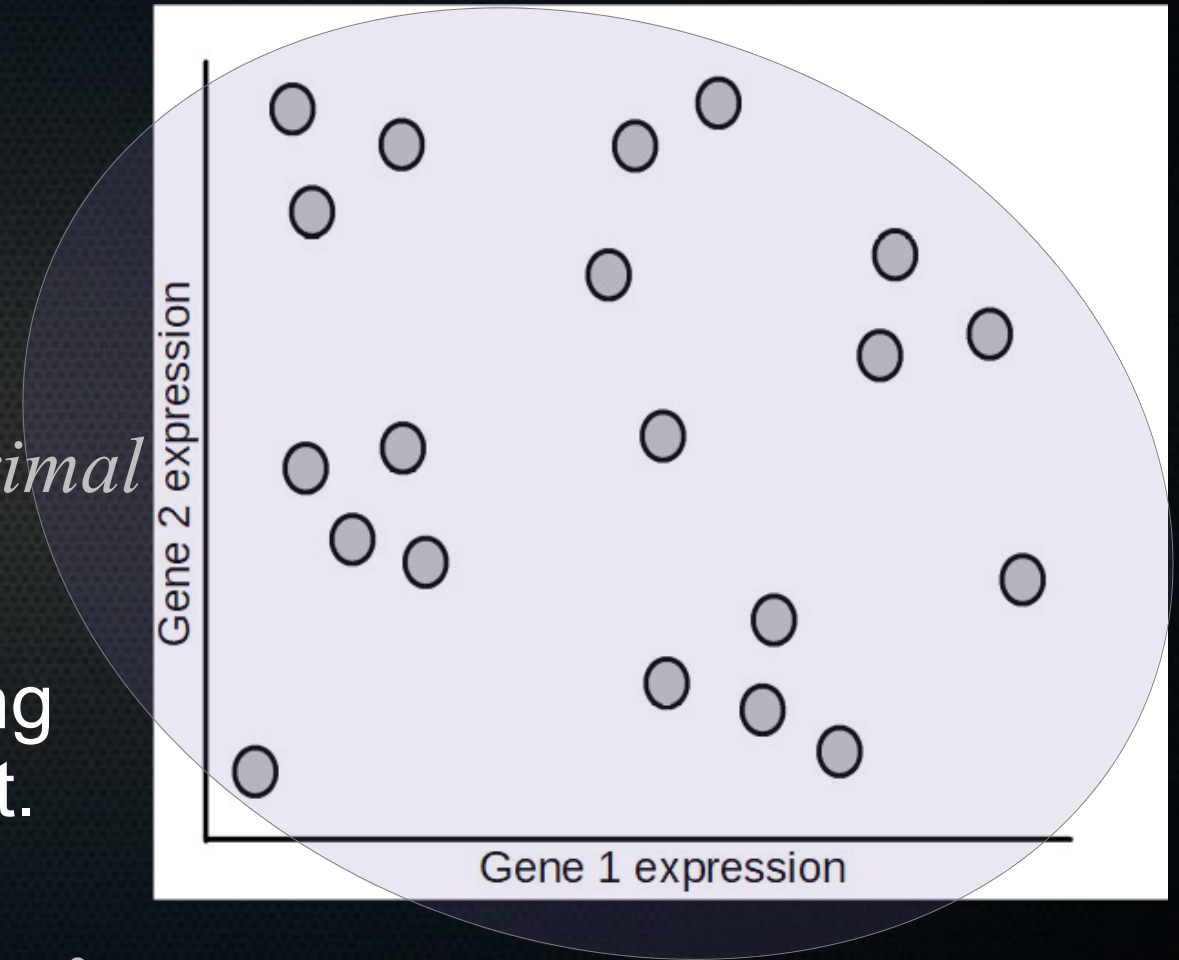
Gene 2 expression

Gene 1 expression

The other way around: bisectional K-means

- Start from one big cluster.

- Each time, randomly split the cluster with the highest SSE.

$$\text{to split=cluster for which}\ (x-\mu)^2 == maximal$$

- Done by selecting random new centroids n times and then picking the two that lower distortion most.

- Let's say we set $n=2$, $K = 3$.

$$J(\ldots) = \frac{1}{m}\sum_{i=1}^{m}(x^{(i)}-\mu_{c^{(i)}})^2$$

n = 1

Gene 2 expression

Gene 1 expression

The other way around: bisectional K-means

- Start from one big cluster.

- Each time, randomly split the cluster with the highest SSE.

$$\text{to split=cluster for which}\ (x - \mu)^2 == maximal$$

- Done by selecting random new centroids *n* times and then picking the two that lower distortion most.

- Let's say we set *n=2, K = 3*.

$$J(\ldots) = \frac{1}{m} \sum_{i=1}^{m} (x^{(i)} - \mu_{c^{(i)}})^2$$
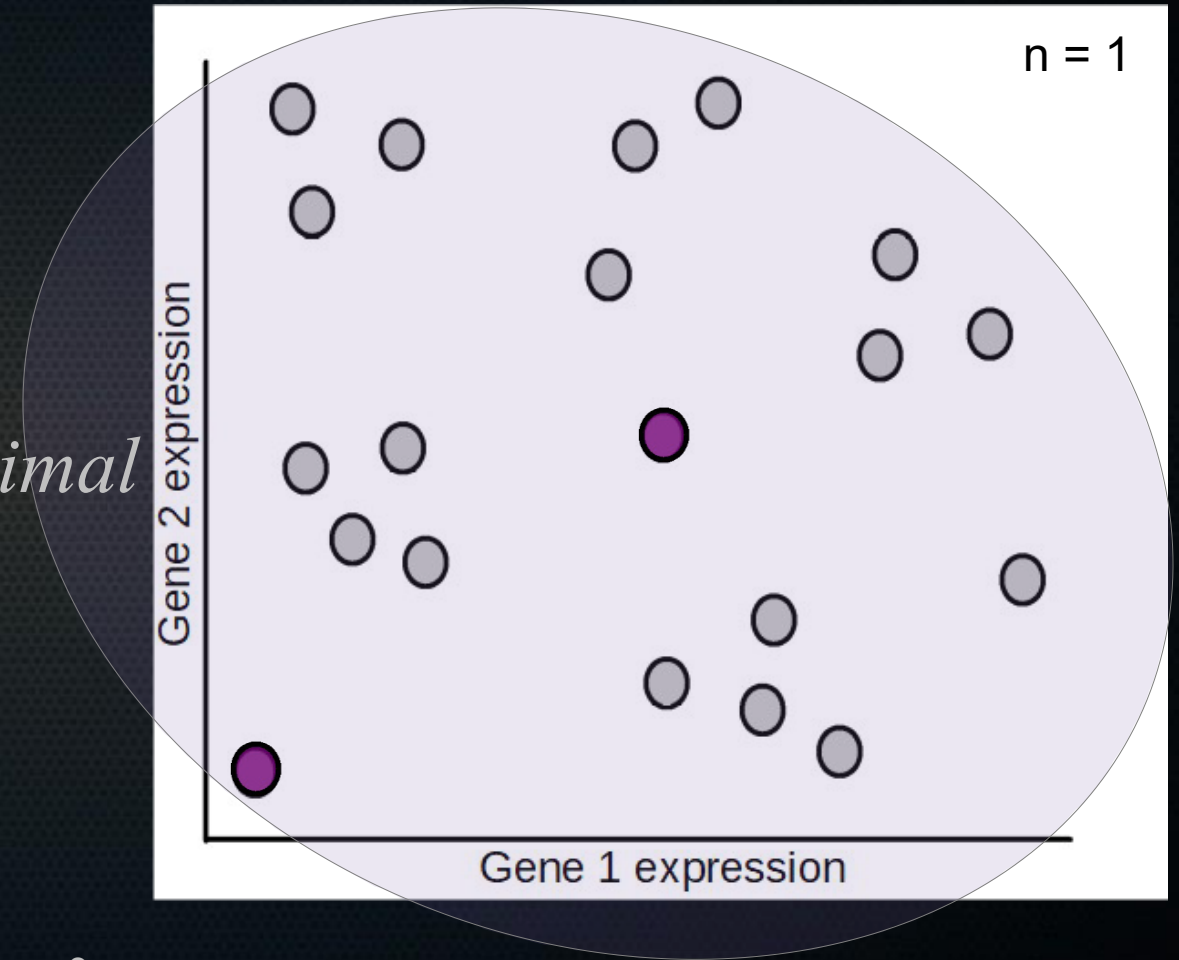
n = 1
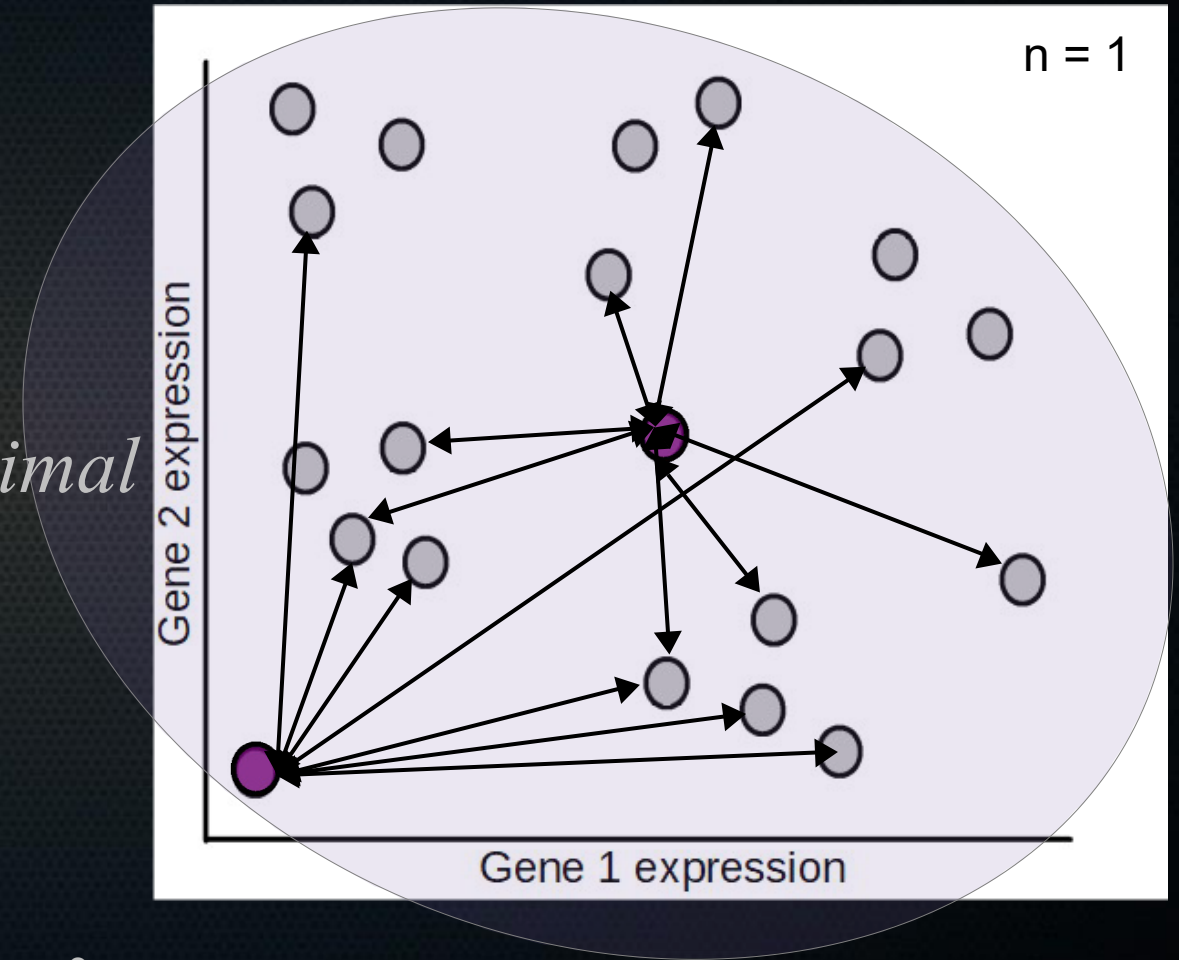
Gene 2 expression

Gene 1 expression

The other way around: bisectional K-means

- Start from one big cluster.

- Each time, randomly split the cluster with the highest SSE.

to split=cluster for which $(x-\mu)^2 == maximal$

- Done by selecting random new centroids n times and then picking the two that lower distortion most.

- Let's say we set $n=2$, $K = 3$.

$$J(...) = \frac{1}{m} \sum_{i=1}^{m} (x^{(i)} - \mu_{c^{(i)}})^2$$

Distortion = high

n = 1

Gene 2 expression

Gene 1 expression

Lower SSE, but still quite large distances

High SSE: many points, also very far away ones
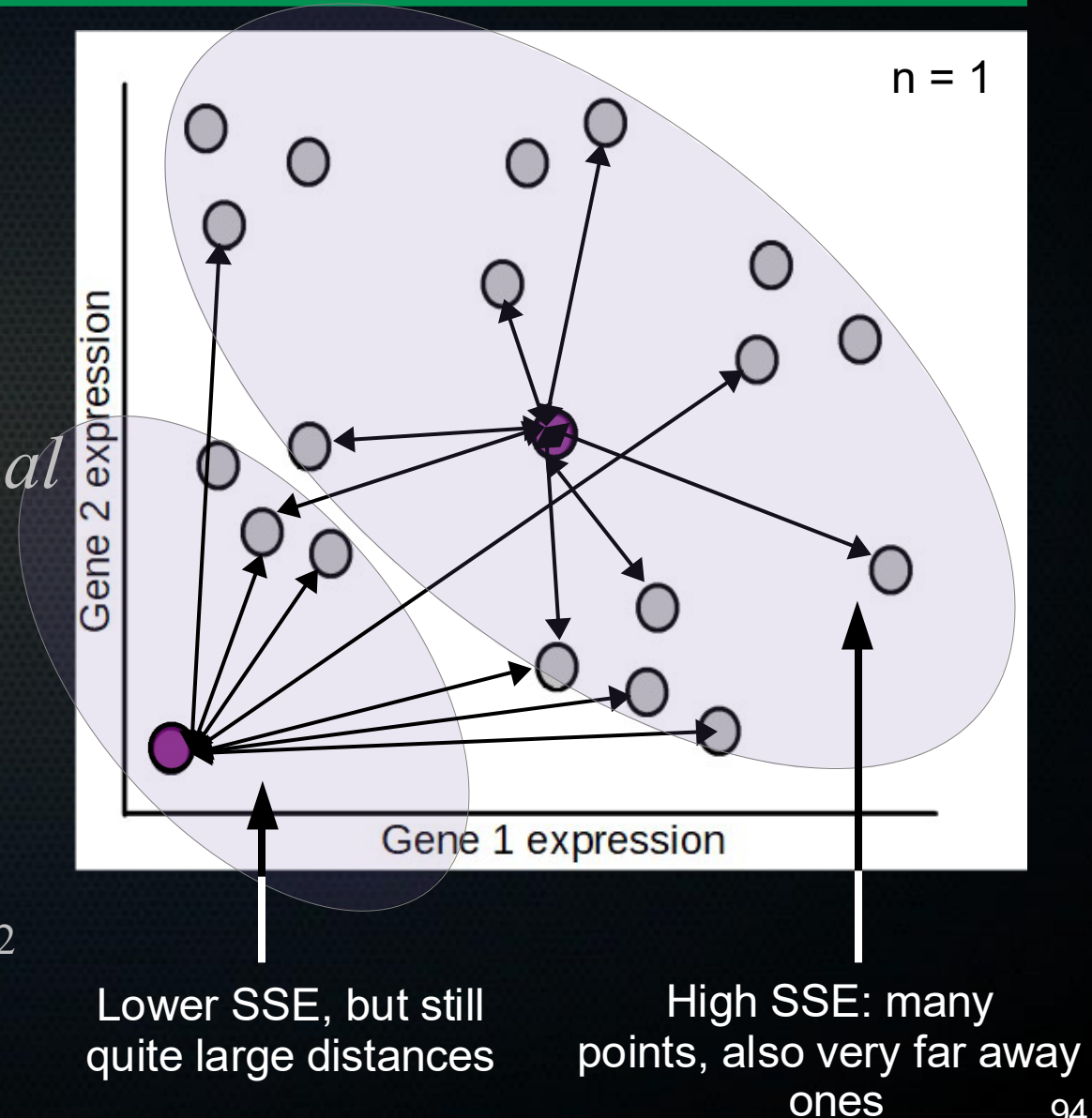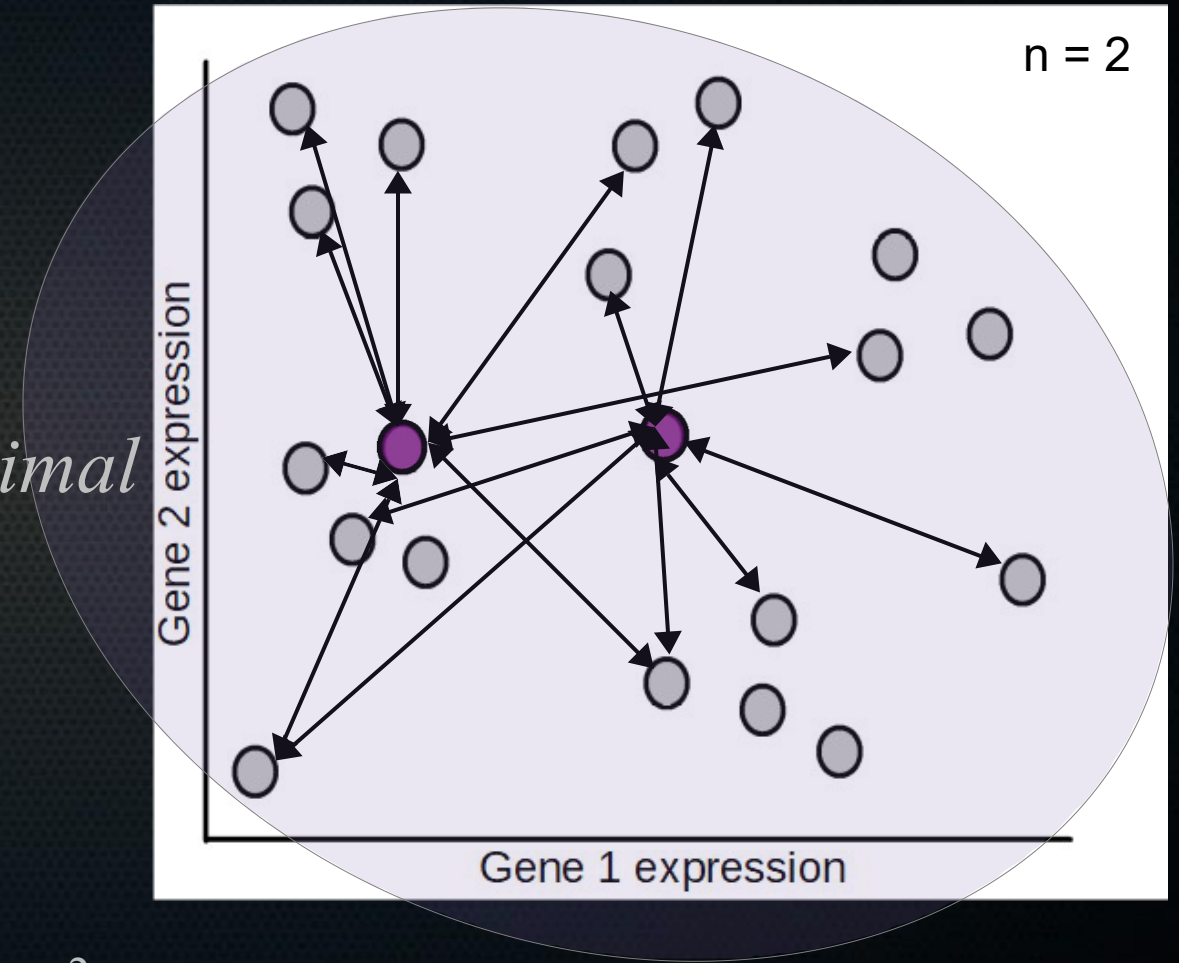
The other way around: bisectional K-means

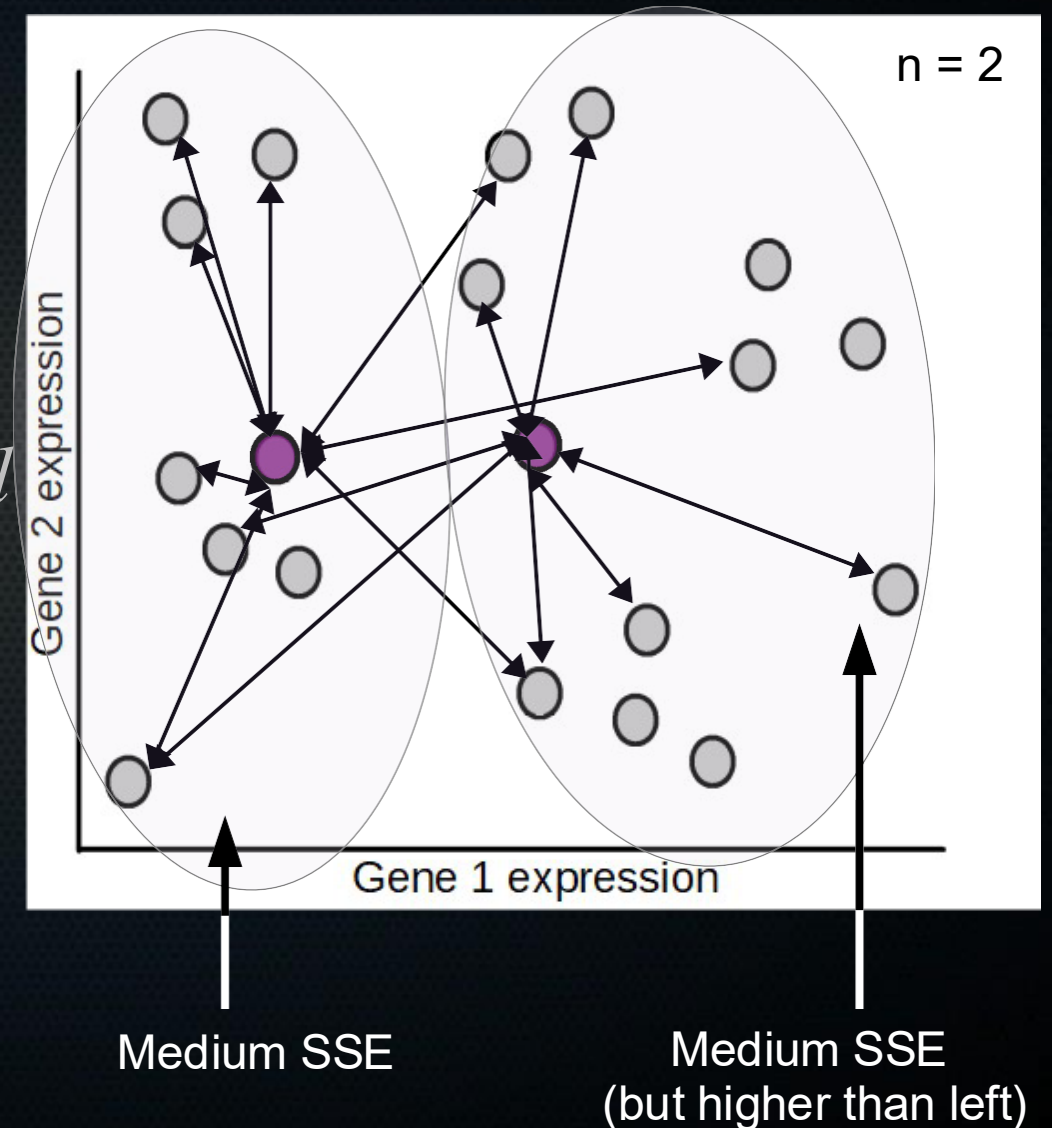- Start from one big cluster.

- Each time, randomly split the cluster with the highest SSE.

to split=cluster for which $(x-\mu)^2 == maximal$

- Done by selecting random new centroids n times and then picking the two that lower distortion most.

- Let's say we set $n=2$, $K = 3$.

$$J(...) = \frac{1}{m} \sum_{i=1}^{m} (x^{(i)} - \mu_{c^{(i)}})^2$$

n = 2

Gene 2 expression

Gene 1 expression

The other way around: bisectional K-means

- Start from one big cluster.

- Each time, randomly split the cluster with the highest SSE.

$$\text{to split} = \text{cluster for which } (x - \mu)^2 == maximal$$

- Done by selecting random new centroids n times and then picking the two that lower distortion most.

- Let's say we set $n=2$, $K = 3$.

$$J(\ldots) = \frac{1}{m} \sum_{i=1}^{m} (x^{(i)} - \mu_{c^{(i)}})^2$$

Distortion = lower

n = 2

Gene 2 expression

Gene 1 expression

Medium SSE

Medium SSE (but higher than left)

The other way around: bisectional K-means

- Start from one big cluster.

- Each time, randomly split the cluster with the highest SSE.

$$\text{to split} = \text{cluster for which} \ (x - \mu)^2 == maximal$$

- Done by selecting random new centroids n times and then picking the two that lower distortion most.

- Let's say we set $n=2$, $K = 3$.

$$J(\ldots) = \frac{1}{m} \sum_{i=1}^{m} (x^{(i)} - \mu_{c^{(i)}})^2$$

n = 1

Gene 2 expression

Gene 1 expression

Medium SSE
(but higher than left)
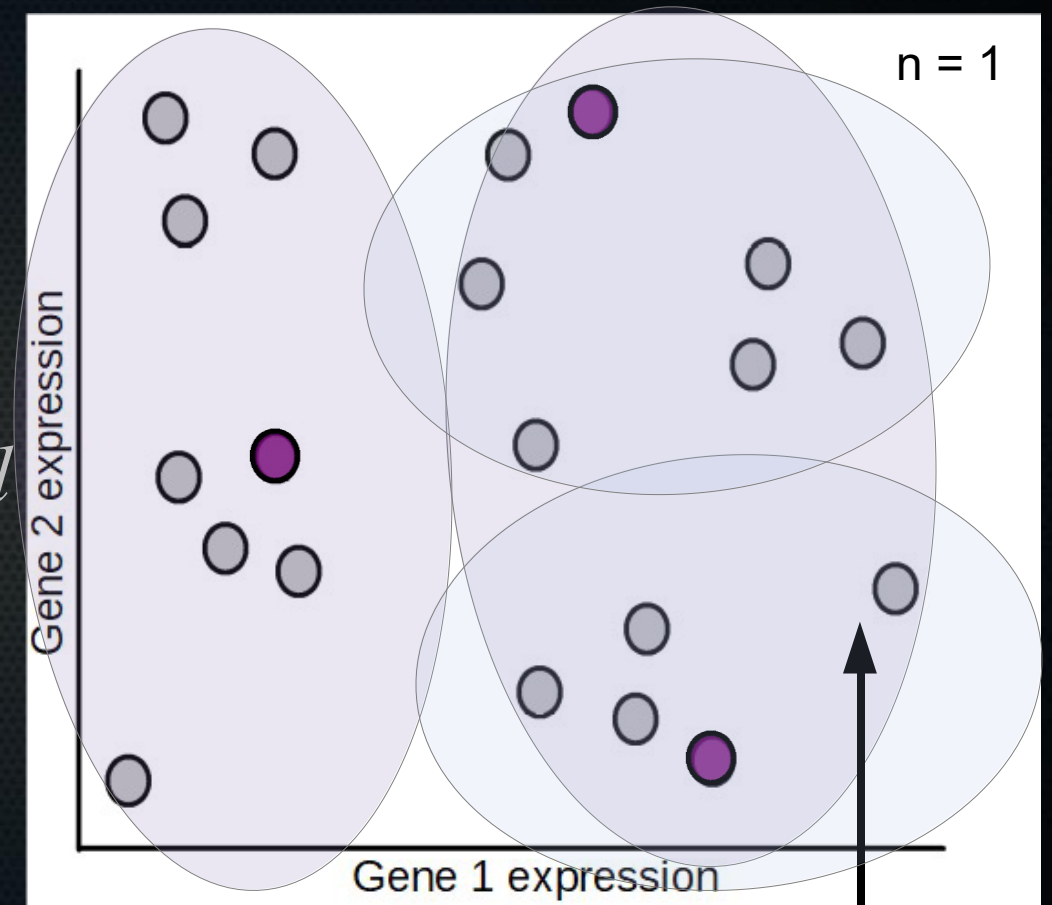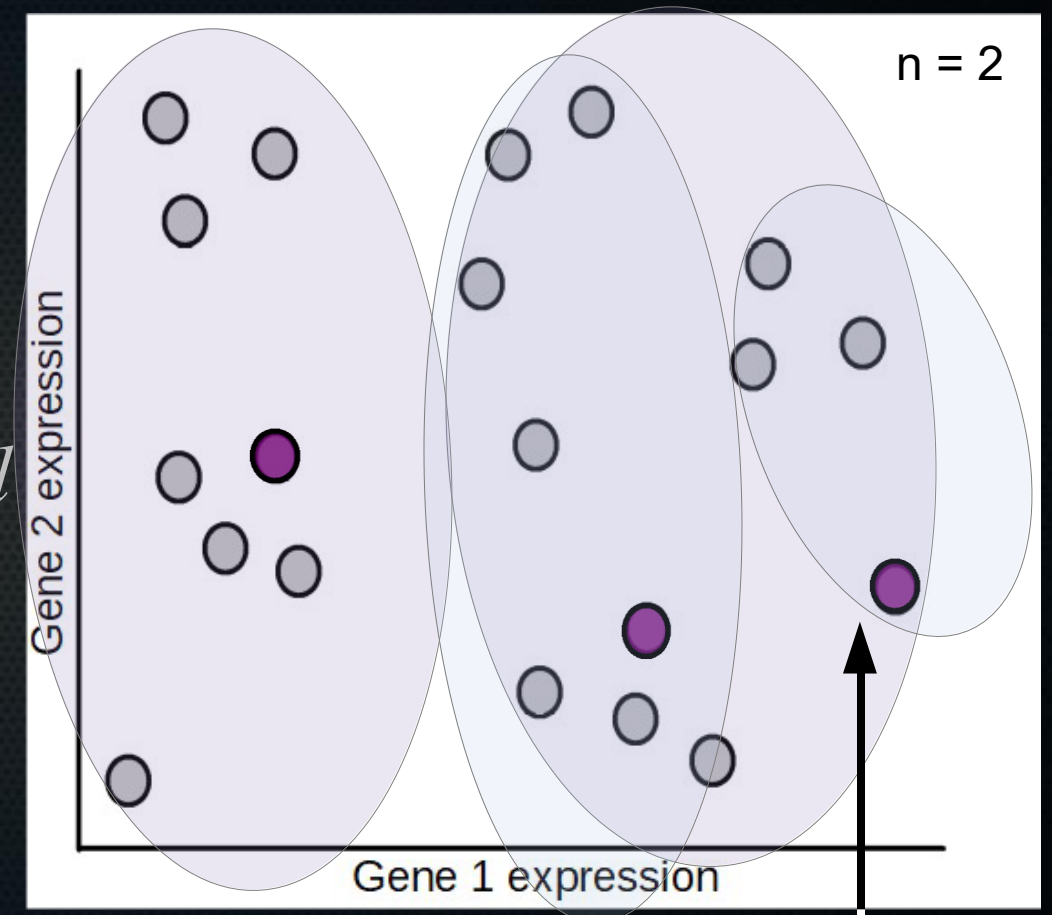Now split this one

The other way around: bisectional K-means

- Start from one big cluster.

- Each time, randomly split the cluster with the highest SSE.

to split=cluster for which $(x - \mu)^2 == maximal$

- Done by selecting random new centroids n times and then picking the two that lower distortion most.

- Let's say we set $n=2$, $K = 3$.

$$J(\ldots) = \frac{1}{m} \sum_{i=1}^{m} (x^{(i)} - \mu_{c^{(i)}})^2$$

n = 2

Gene 2 expression

Gene 1 expression

Medium SSE
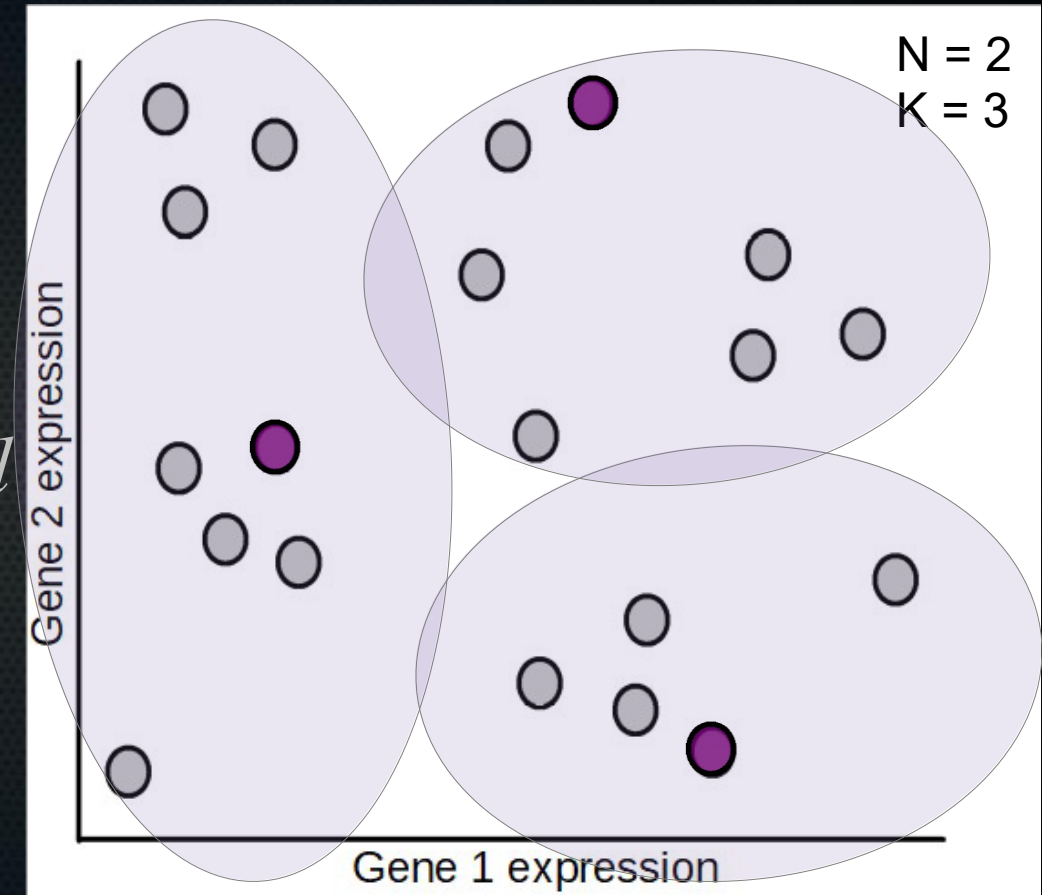(but higher than left)
Now split this one

The other way around: bisectional K-means

- Start from one big cluster.

- Each time, randomly split the cluster with the highest SSE.

to split=cluster for which $(x-\mu)^2 == maximal$

- Done by selecting random new centroids n times and then picking the two that lower distortion most.

- Let's say we set $n=2$, $K = 3$.

$$J(\ldots) = \frac{1}{m} \sum_{i=1}^{m} (x^{(i)} - \mu_{c^{(i)}})^2$$

N = 2
K = 3

Gene 2 expression

Gene 1 expression

Done!

Summary bisectional K-means

- Start with everything in one cluster

- Give number of clusters you want (K) and times to try random split per cluster (n).

- For cluster with highest SSE of members → randomly choose two data points as new centroids n times.

  - Calculate new SSE for this split

  - Pick split with lowest SSE

- Continue until you have K clusters.

Break for practical