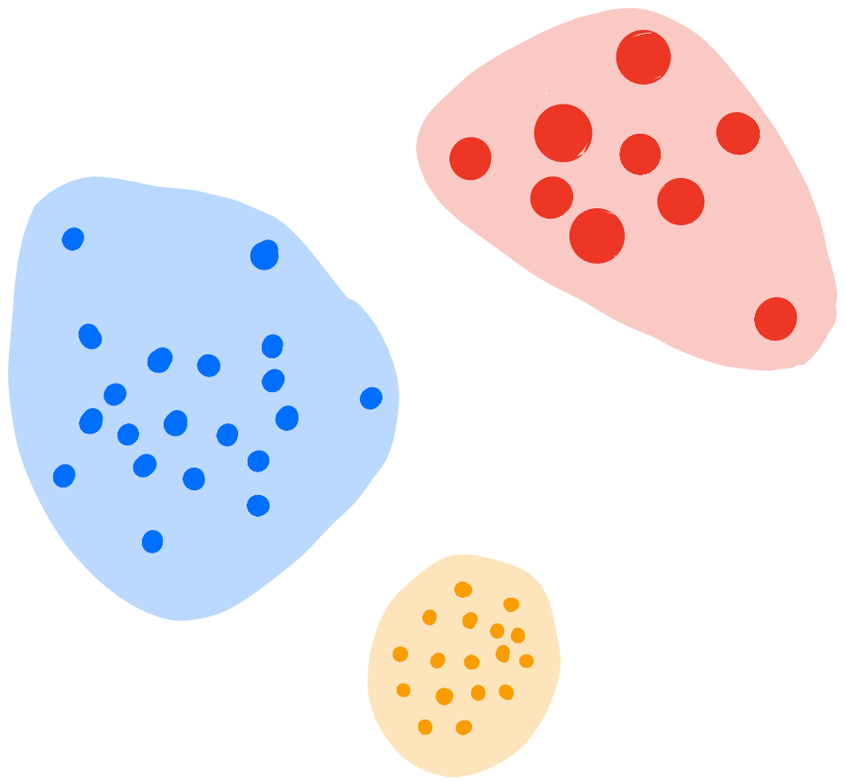
Introduction to clustering methods

Epid 814 - Marisa Eisenberg

Cluster Analysis

- What is a cluster?
    - A set of objects/data points, such that the objects in the set are more similar to one another than they are to the objects outside the set/other clusters.



Cluster Analysis

- Broadly used in data analysis, including machine learning
- **Clustering** (unsupervised) vs. **classification** (supervised)
- **Hard clustering** (every element belongs to only one cluster) vs. **fuzzy clustering** (every element has various probabilities of belonging to a given cluster)
- Some methods find the number of clusters, others use a predefined number of clusters

Cluster Analysis

- Wide range of methods—which is best depends on the data to be clustered. Not really one ‘best’ method across all settings.

- In general, we want:
  - High intra-cluster similarity, low inter-cluster similarity (how to determine similarity?)
  - Potential to discover hidden features (especially in high dimensional data)

Some general classes (or clusters haha) of clustering methods:

- **Partitioning** methods (e.g. k-means clustering & other centroid methods)
- **Hierarchical** clustering methods
- **Density-based** methods
- **Model or distribution-based** methods (e.g. Gaussian mixture models, latent class analysis)
- Network clustering methods (**community detection** methods)
- & many others!

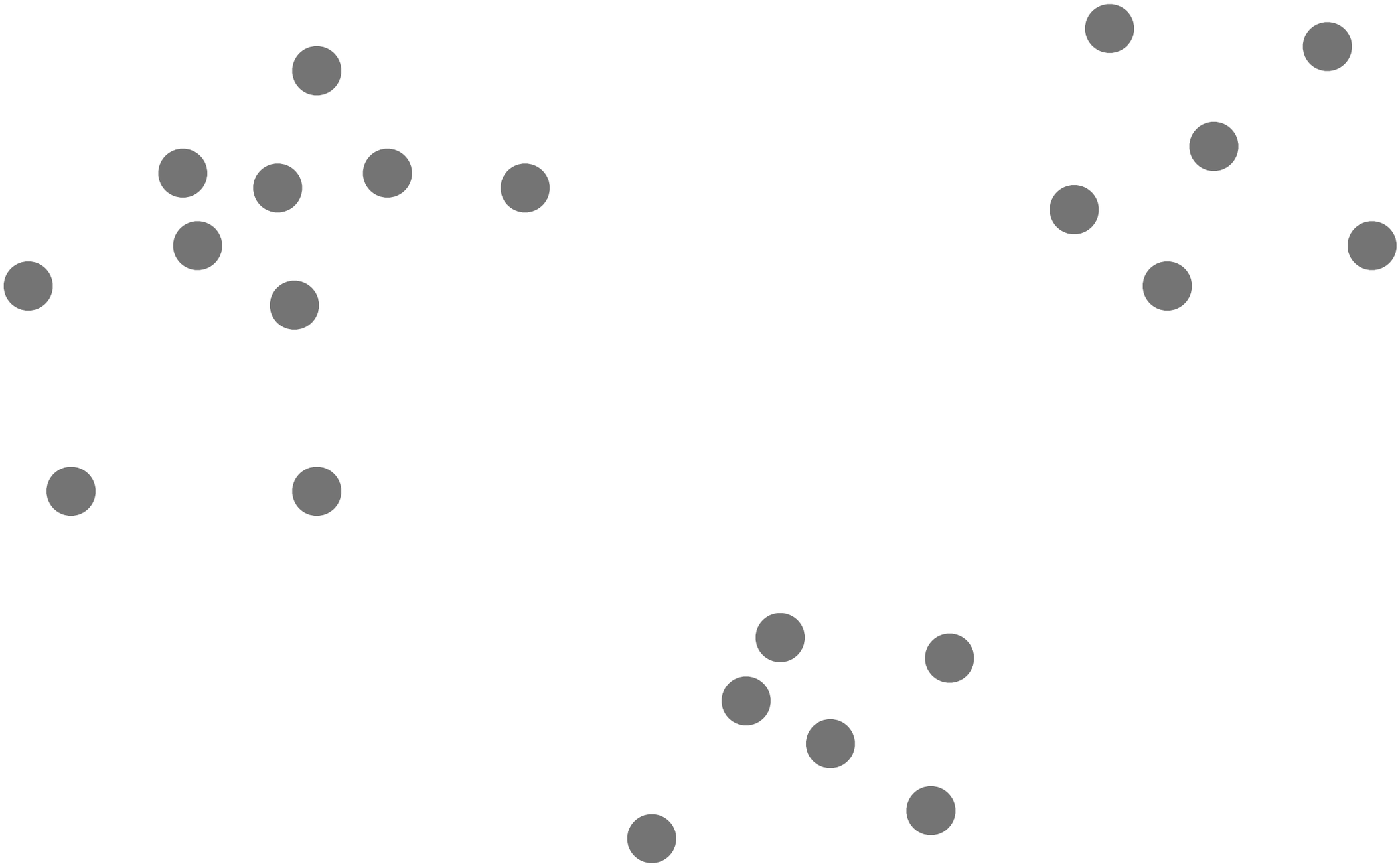
Partitioning methods

- General idea is often:
  - Construct a partition of the data into *k* clusters
  - Evaluate the resulting clusters and improve the partition
  - Repeat until optimal partition/clusters found
- Examples: k-means, k-medioids, k-modes (among many others)

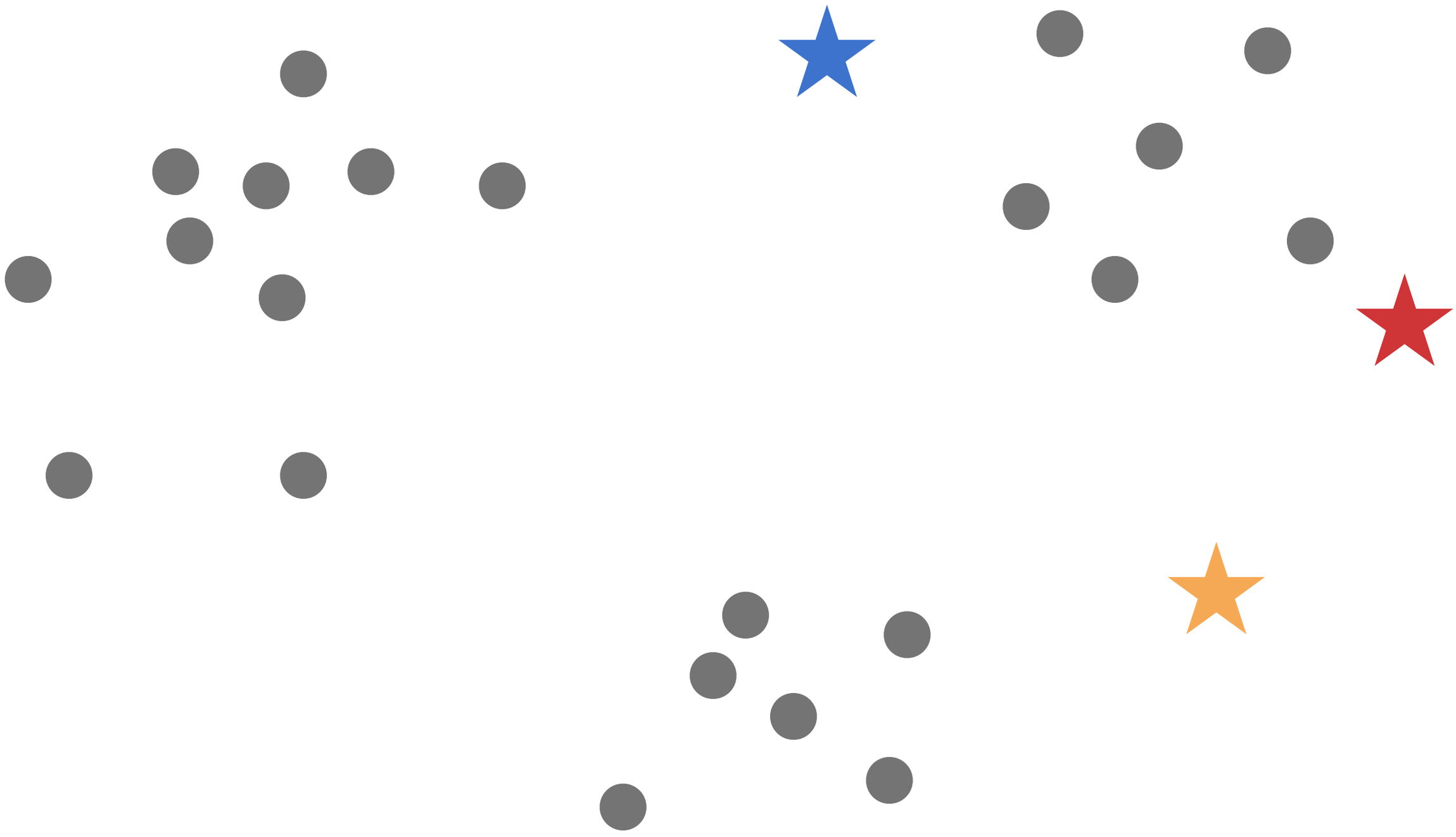
K-means clustering

- Select k centroids (means), and each data point is assigned to the nearest centroid
- This partitions the space into Voronoi cells, which are our clusters
- For each cluster, calculate the centroid of all points
- These become the new cluster centroids
- Reassign points to nearest centroid and repeat

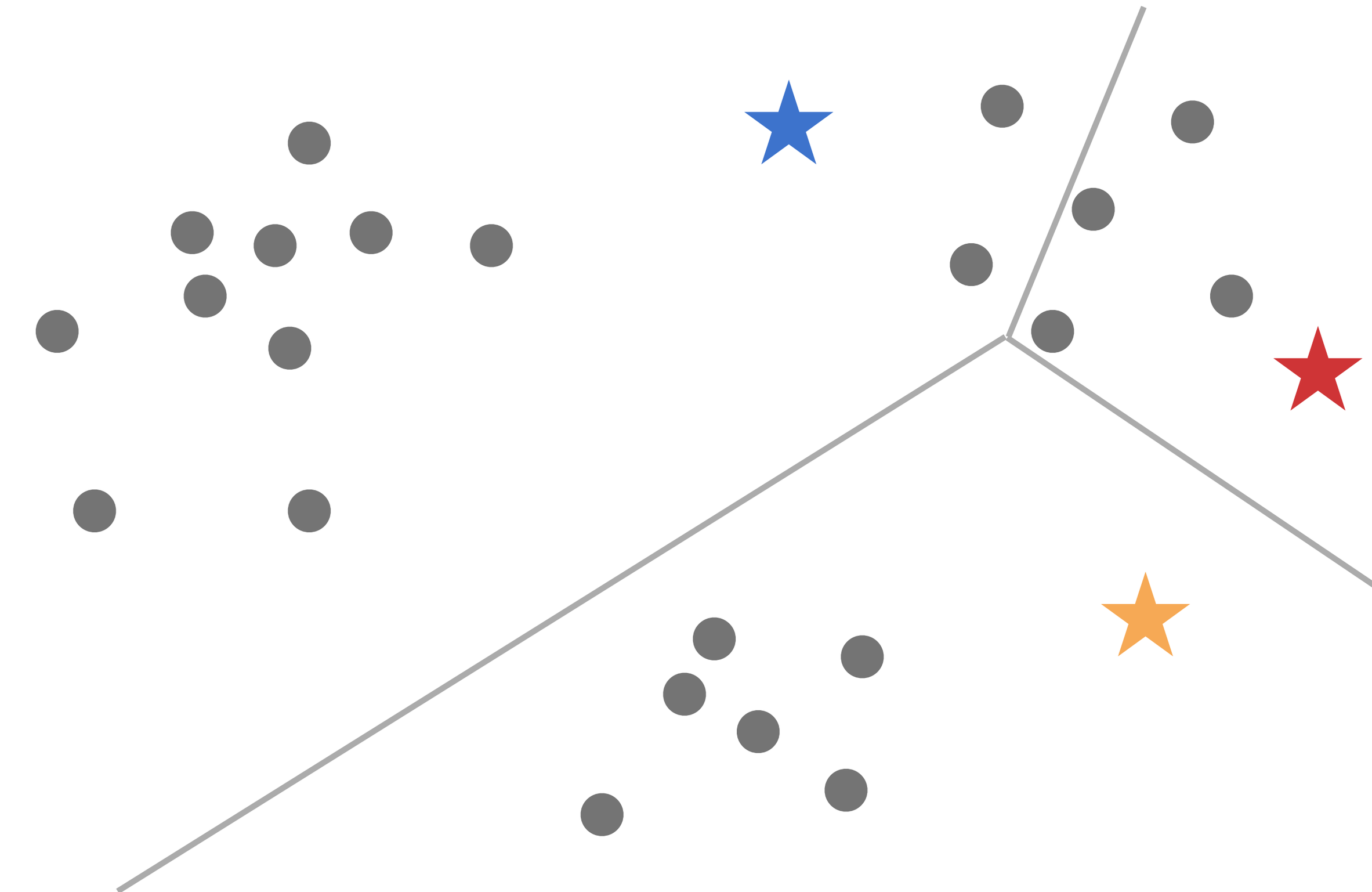
K-means clustering example



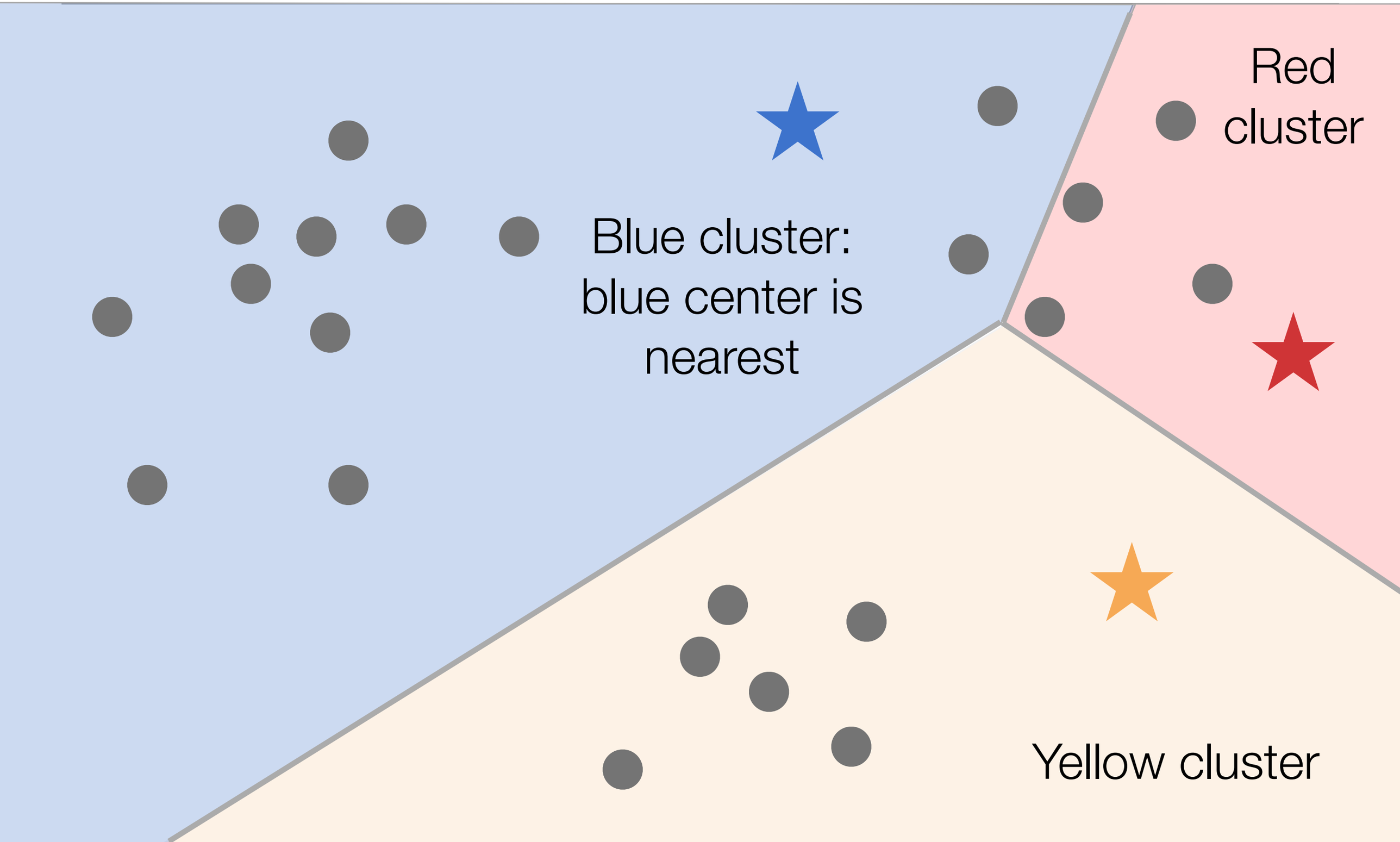
Randomly choose 3 cluster centers to start



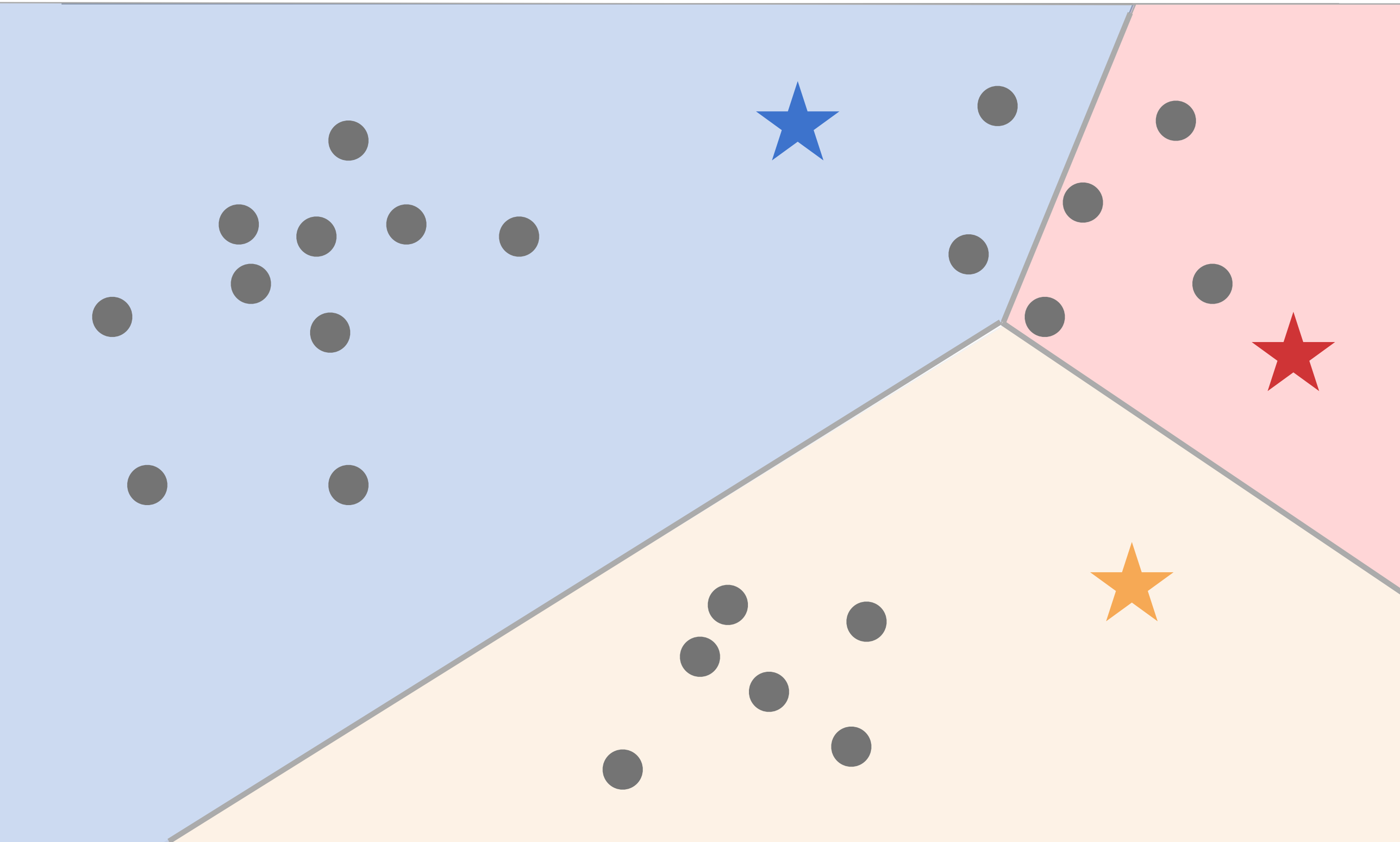
The cluster centers partition the space based on which center is nearest



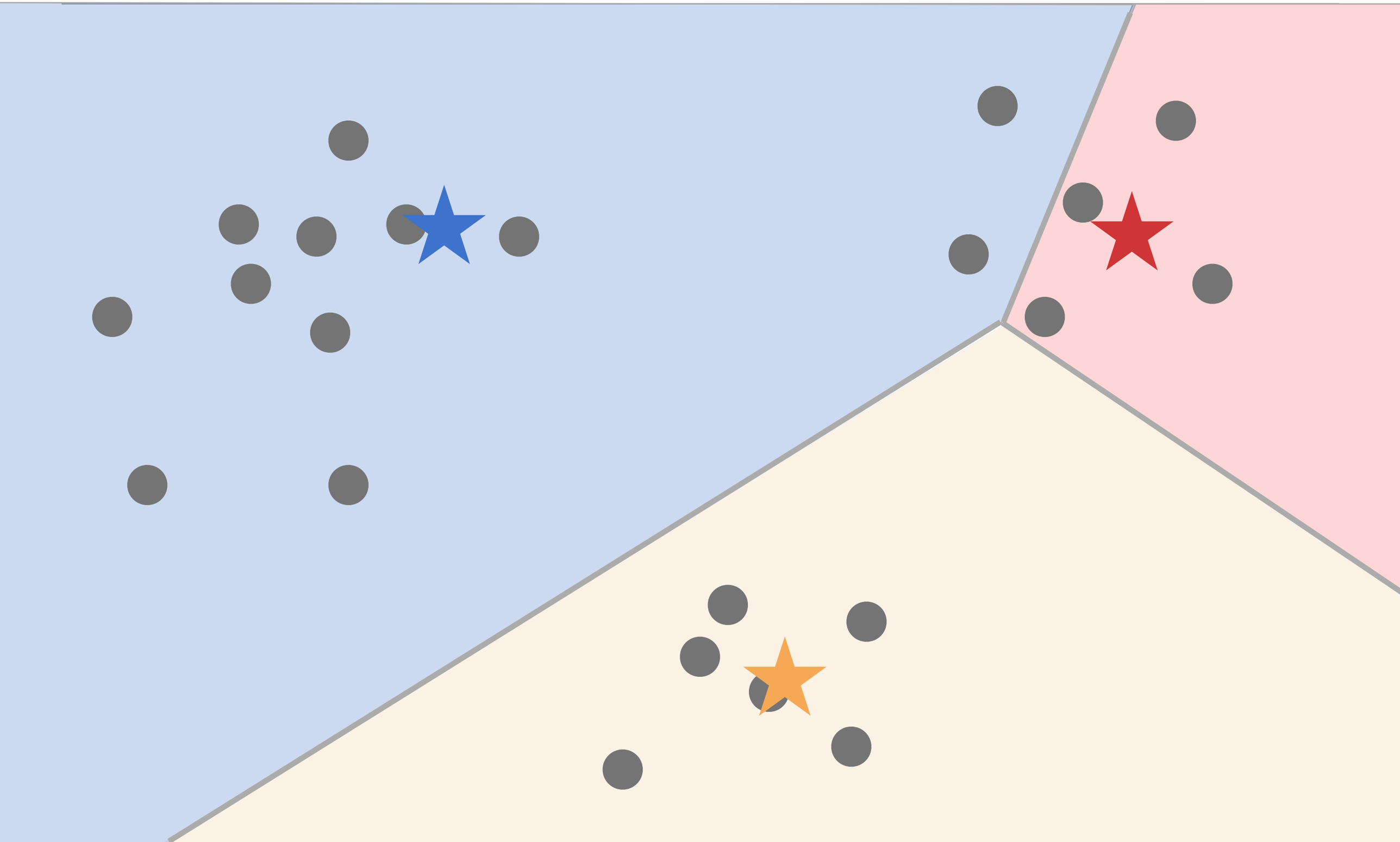
These are our starting **clusters**



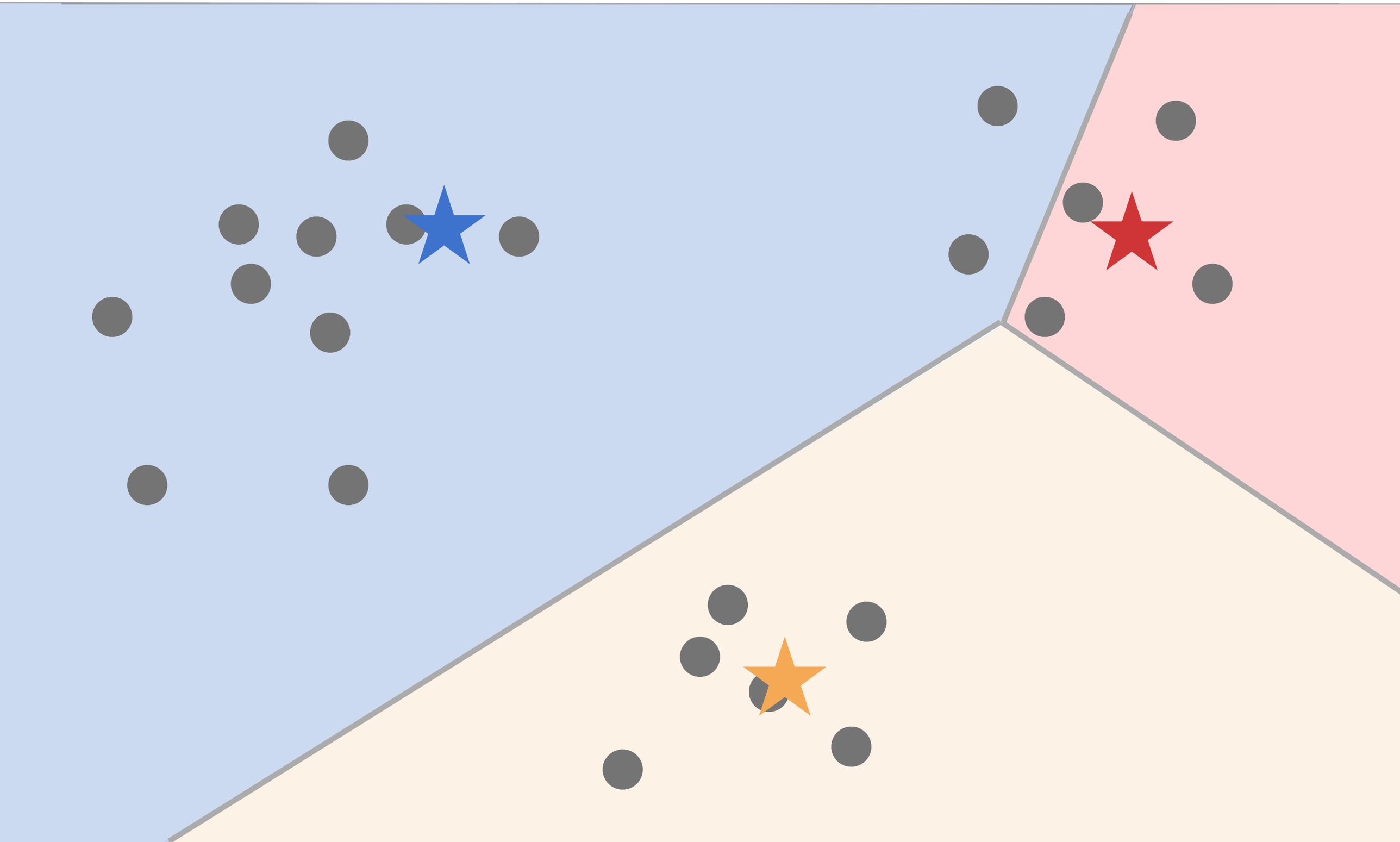
What are the means of the data points in each cluster?



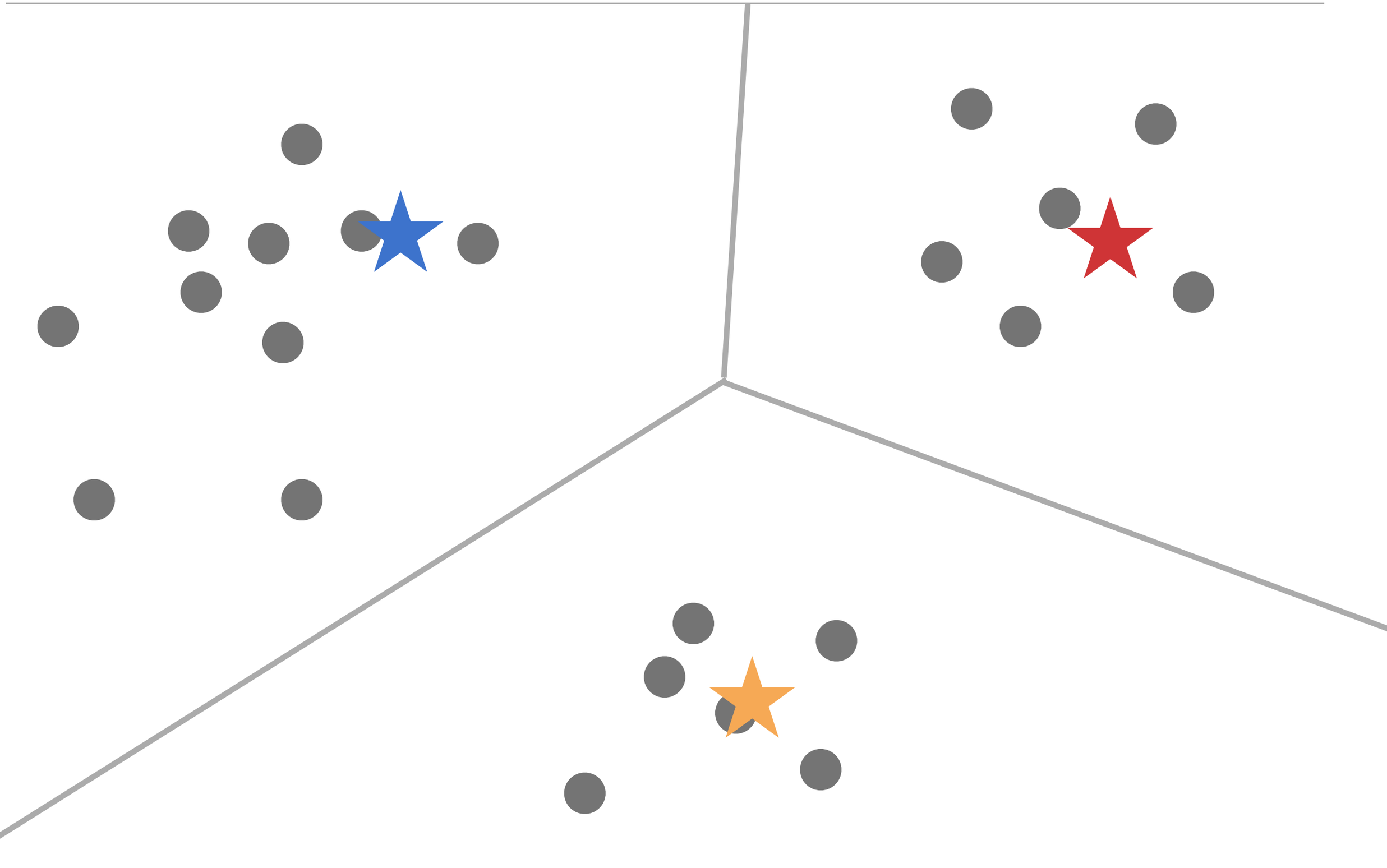
These are the new centers.



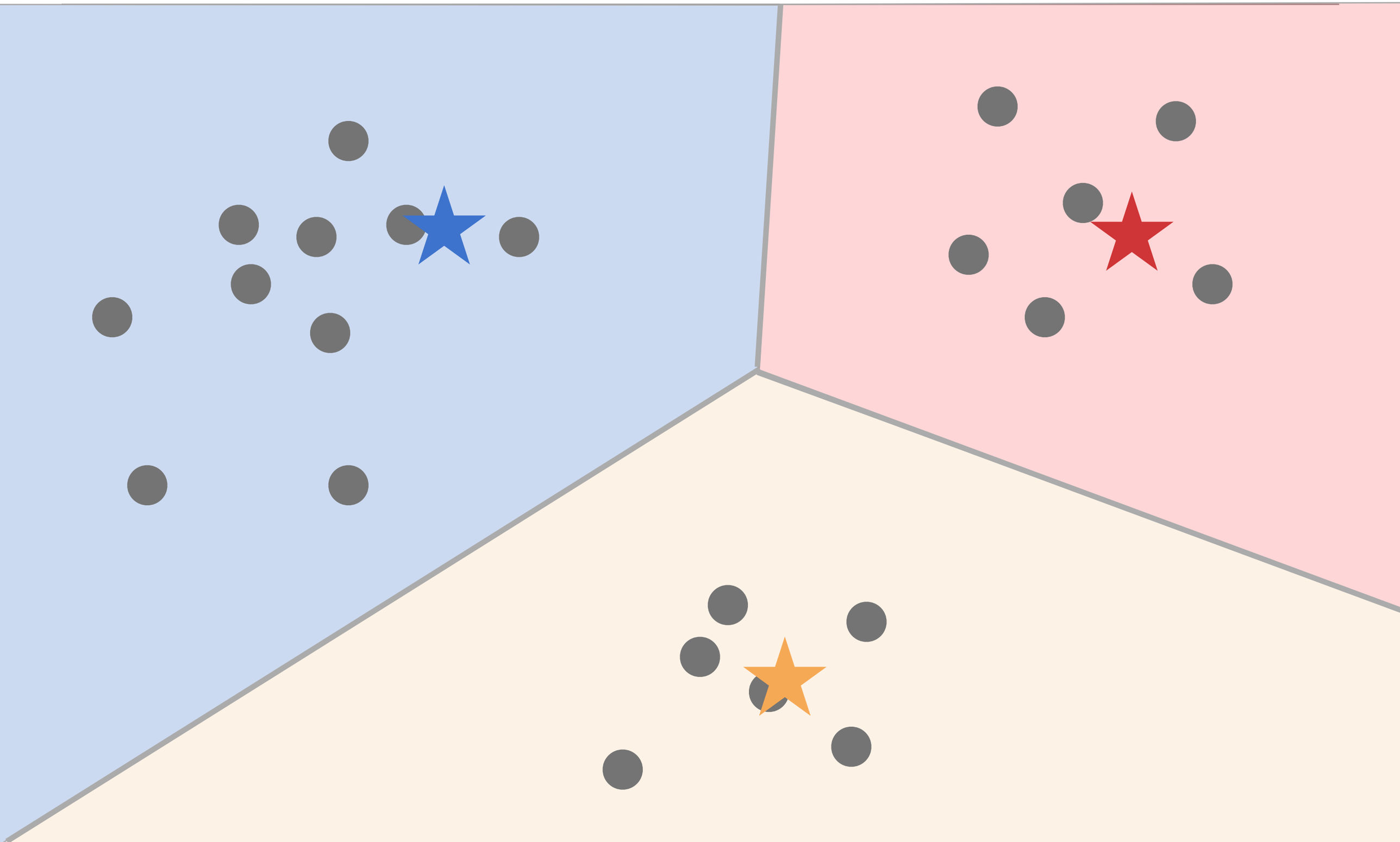
Now which data points are closest to each center?



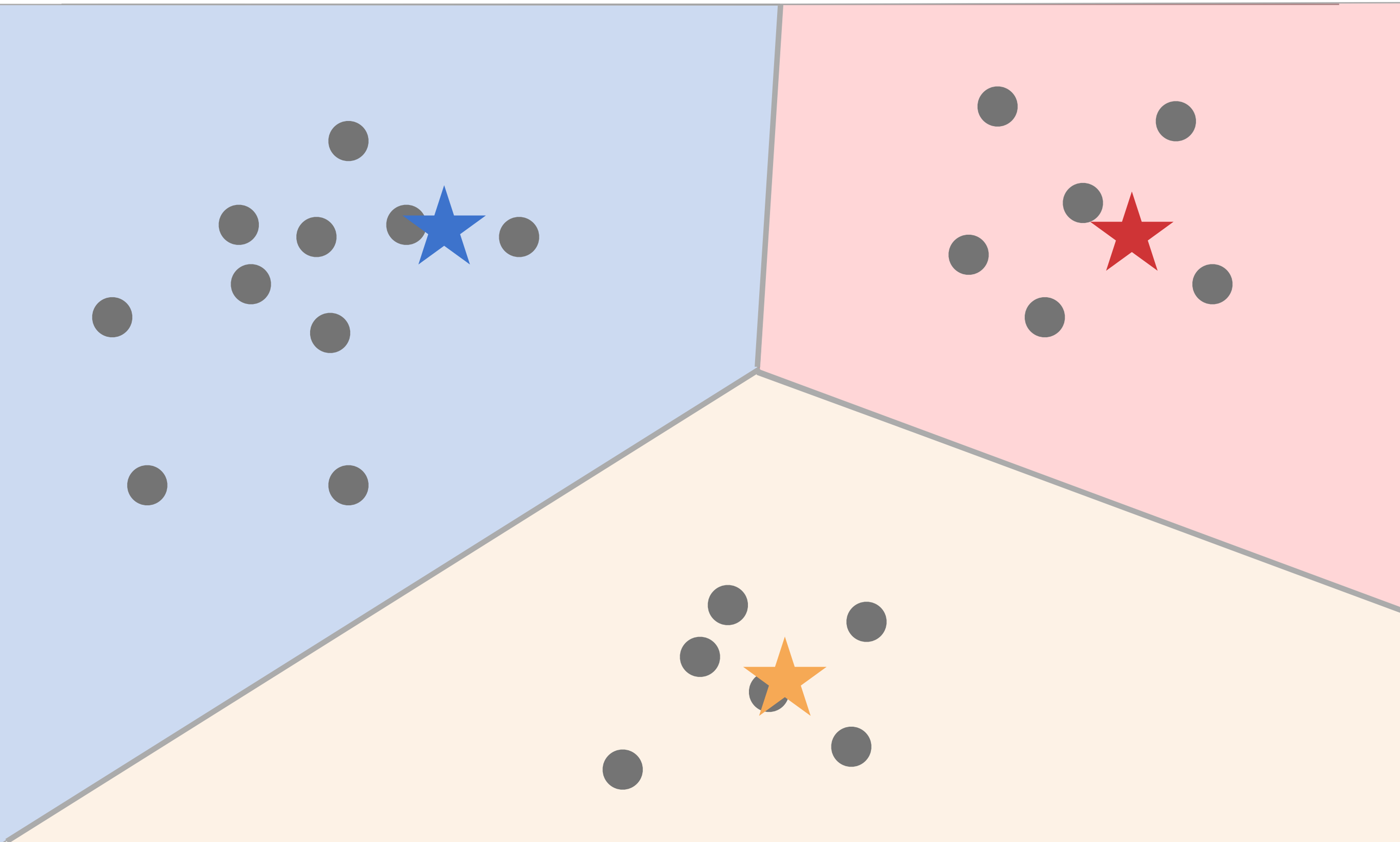
Now which data points are closest to each center?



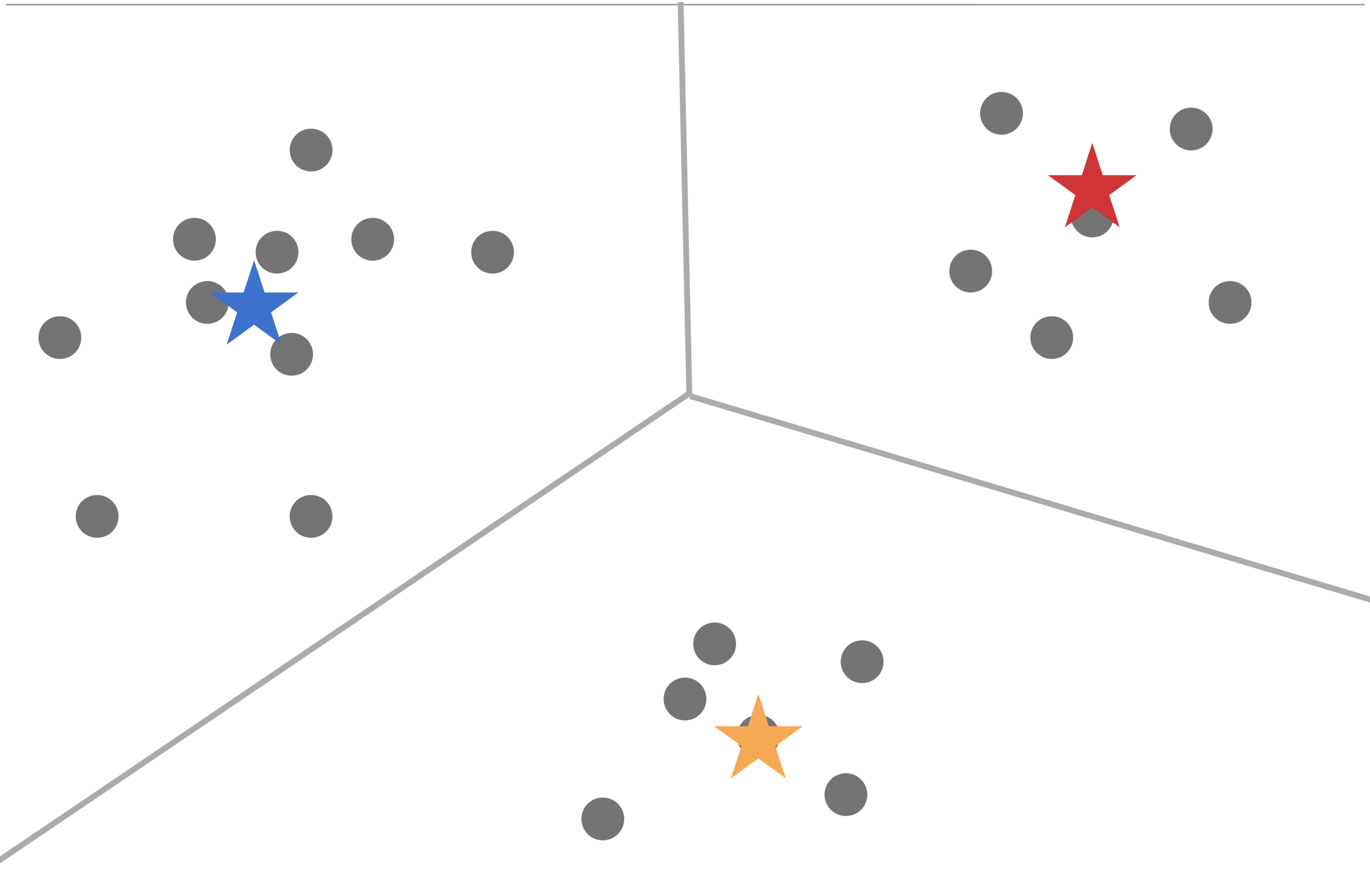
Redefine the clusters based on which center they're nearest



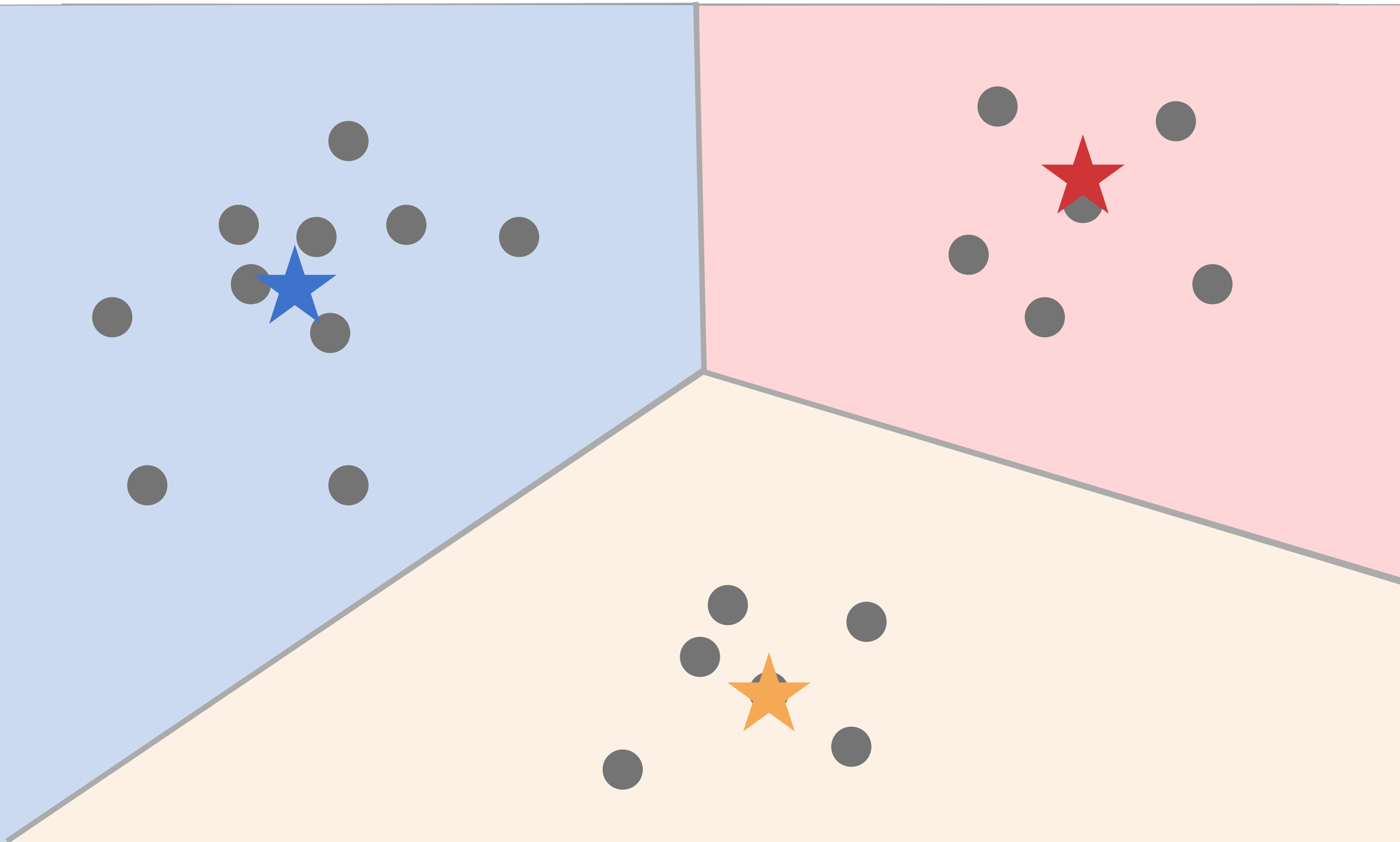
And repeat! Keep calculating the centers and redefining the clusters until they stop changing.



And repeat! Keep calculating the centers and redefining the clusters until they stop changing.

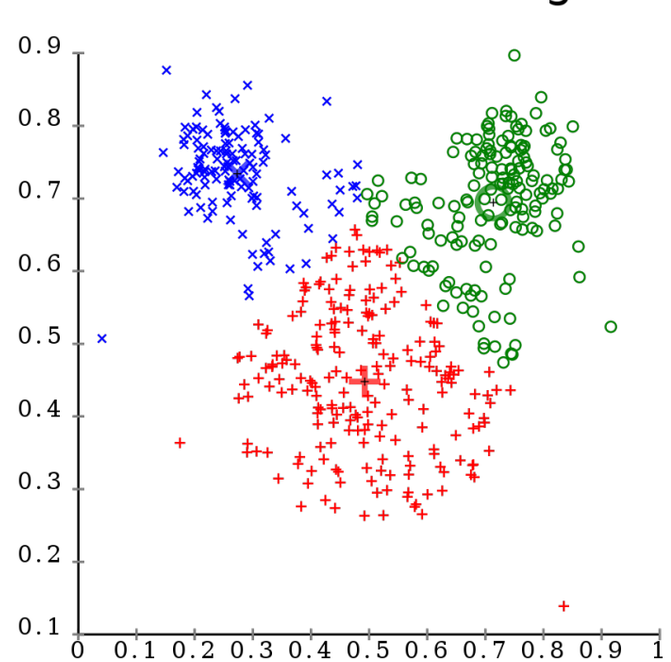


The results once the clusters and centers are fixed are your final k-means clusters.



K-means clustering

- Relatively efficient
- Can converge to local optima (e.g. depending on starting points)
- Have to specify k (number of clusters)
- Cannot make clusters with non-convex shapes
- Tends toward equal sized clusters
- How to handle categorical data? (e.g. can use k-modes)

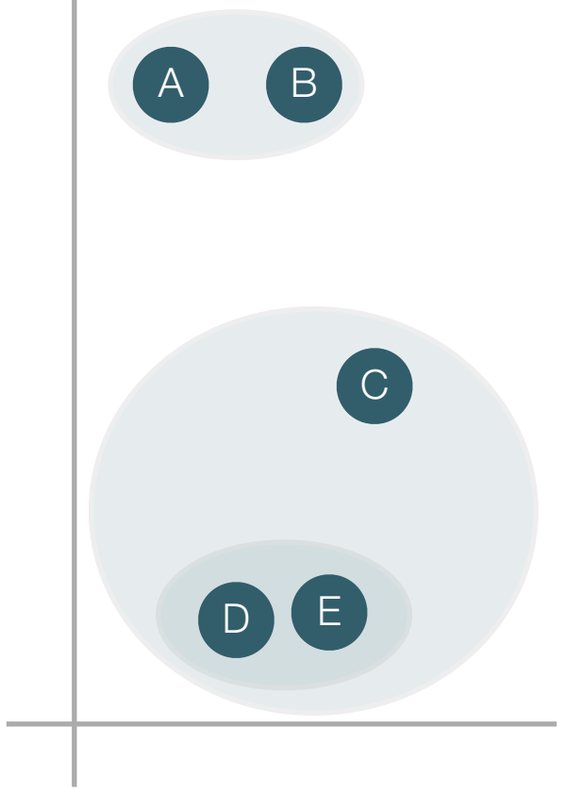


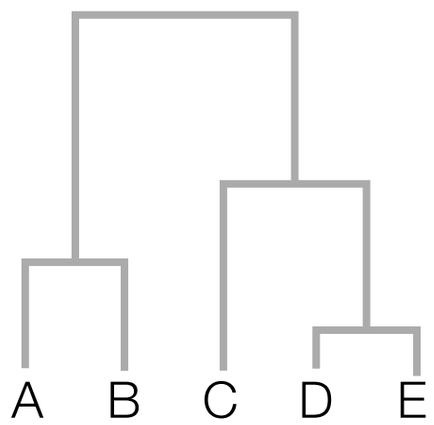
By Chire - Own work, Public Domain, https://commons.wikimedia.org/w/index.php?curid=11765684

Hierarchical clustering methods

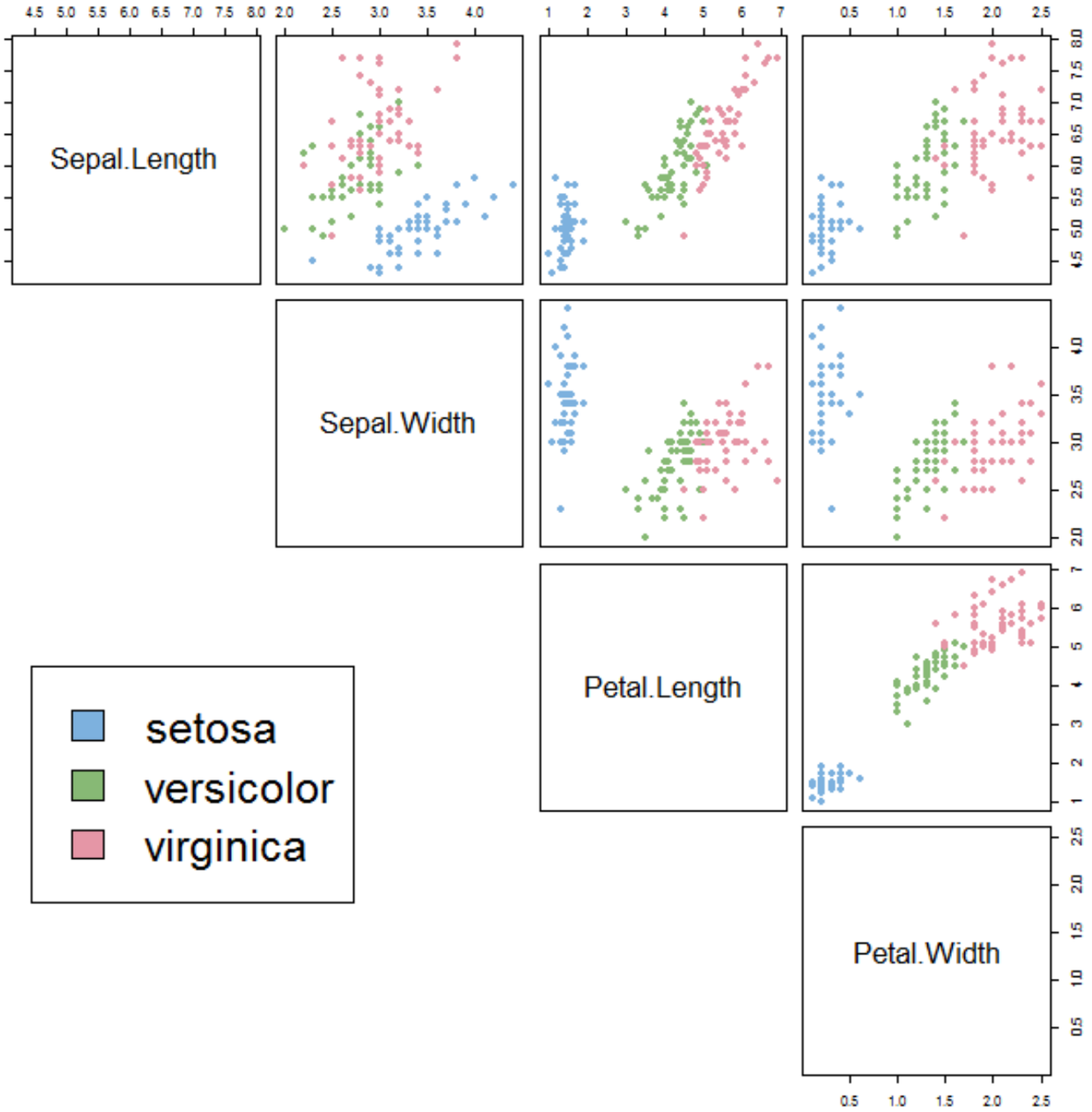
- Agglomerative approach to clustering
    - Starts with small clusters (e.g. individual points) and then merges based on distance
- Divisive approach does the reverse (all one cluster then split into smaller ones)
- Many different approaches with different distance measurements, etc.

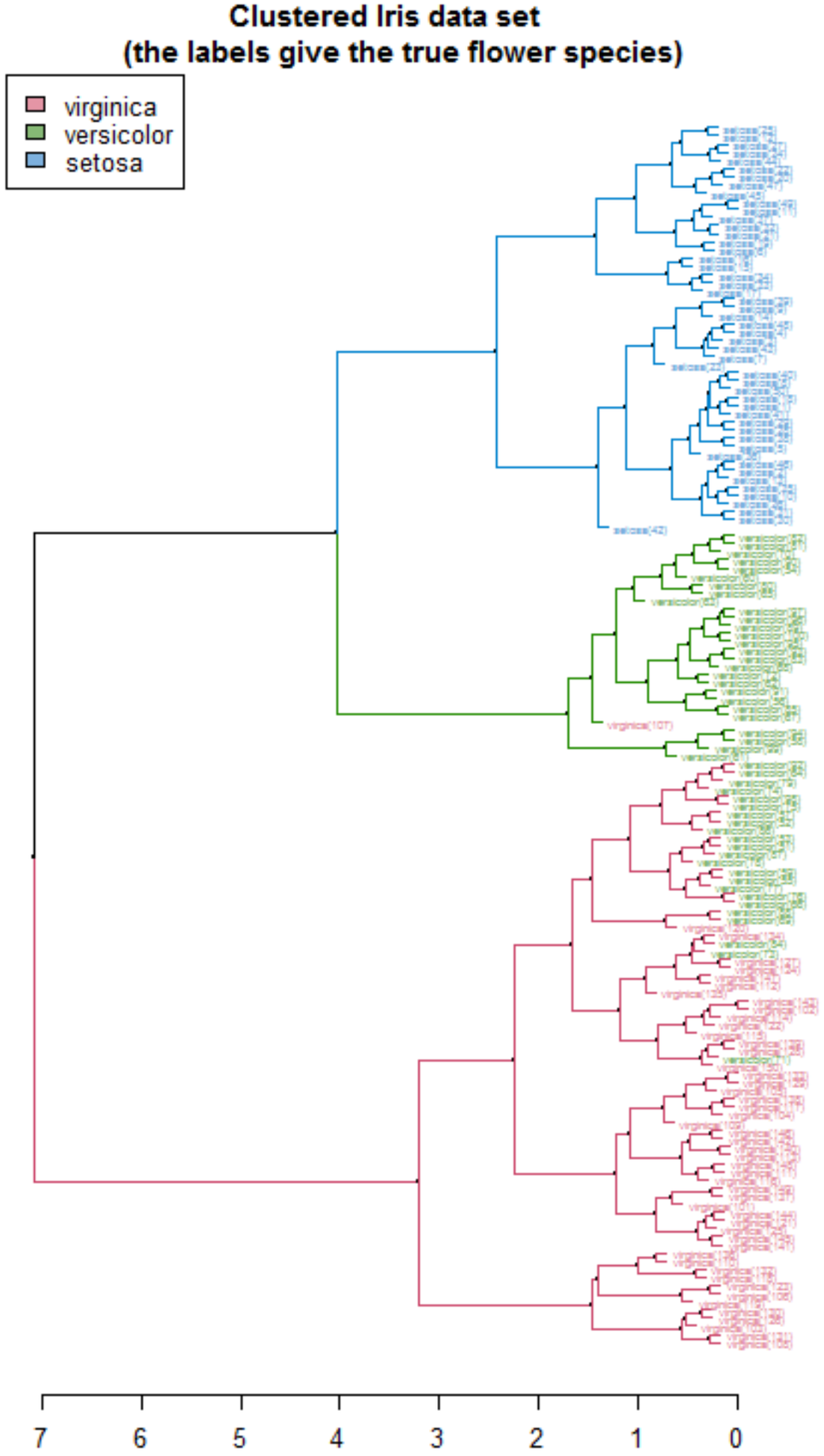
Hierarchical clustering example





- Start with all single point clusters
- Merge the two nearest clusters—forms a new cluster
- Merge the next two nearest clusters, etc.
- How to decide cluster distances? (What metric, do we use nearest point distance, furthest, centroid?)
- Capture clusters as a dendrogram—can choose resolution of clusters as desired





https://cran.r-project.org/web/packages/dendextend/vignettes/Cluster_Analysis.html

Hierarchical clustering

- Slow for larger data sets
- Useful for finding substructures/subclusters in data
- Assumes every data point is relevant/part of the clusters
- How to choose level of granularity?

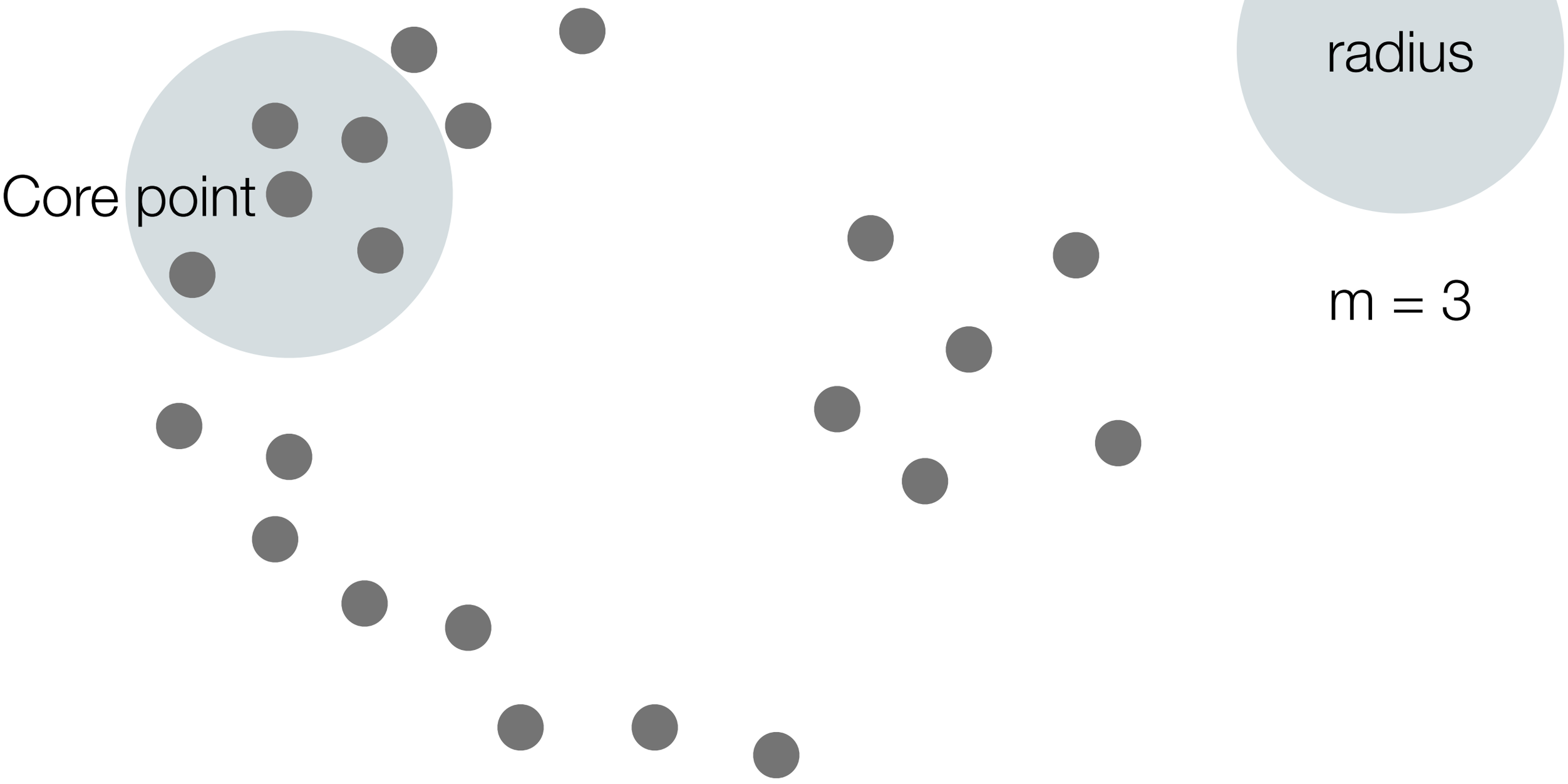
Density-based clustering

- Decides clusters based on density of points
- Not every point need be assigned a cluster—some can be considered noise or outliers
- One of the most commonly used algorithms is DBSCAN (Density-Based Spatial Clustering of Applications with Noise)

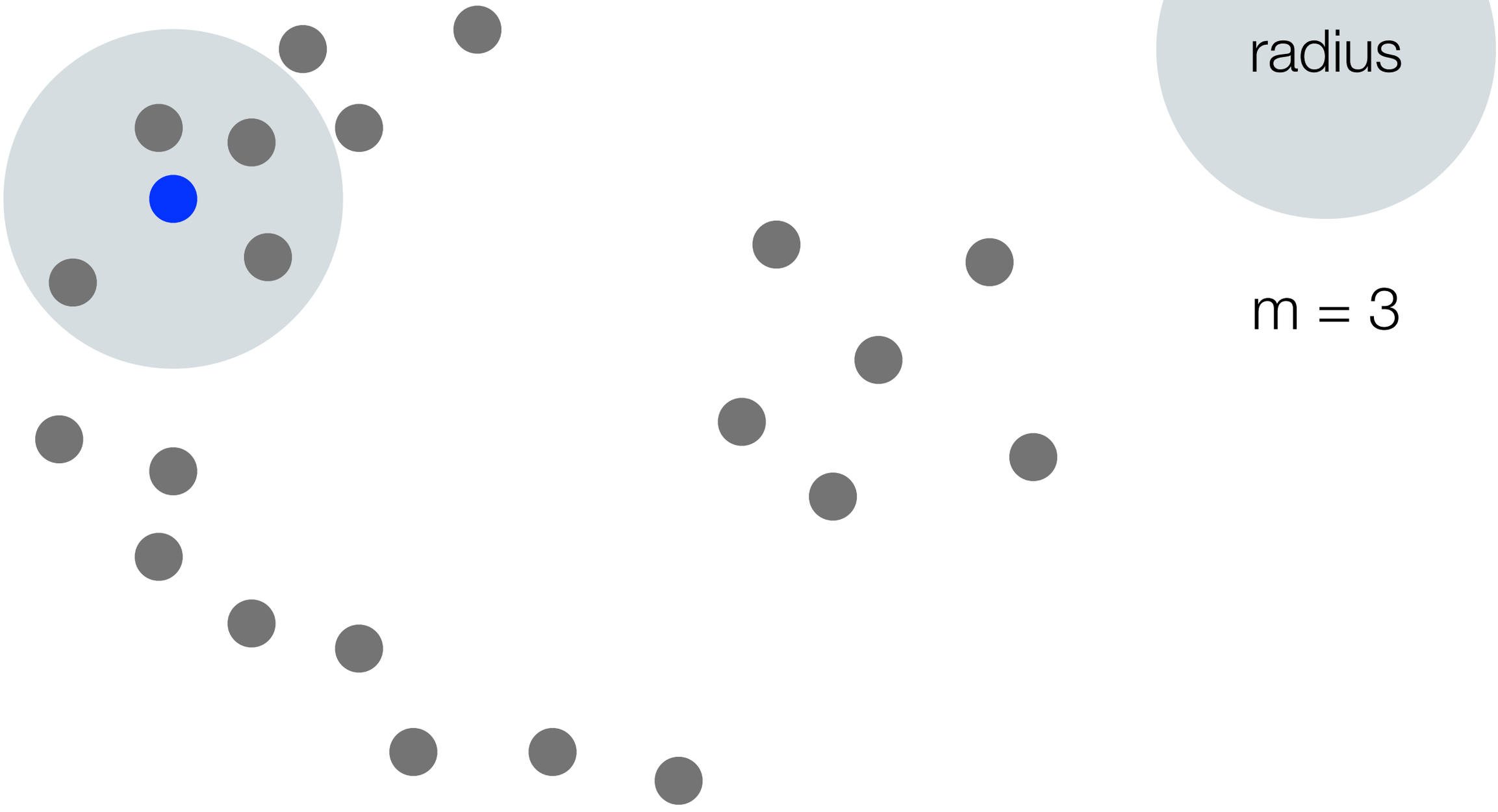
DBSCAN

- Choose a radius *r* and a minimum number of points *m*
- Classify each point as a:
  - **Core point** - has at least *m* other points within radius *r*
  - **Border point** - does not have *m* points within radius *r*, but is **reachable** a core point *p* - i.e. can be connected to data point *p* by a chain of core points each within radius *r* of the next point
  - **Outlier** - neither core nor border

DBSCAN example



DBSCAN example

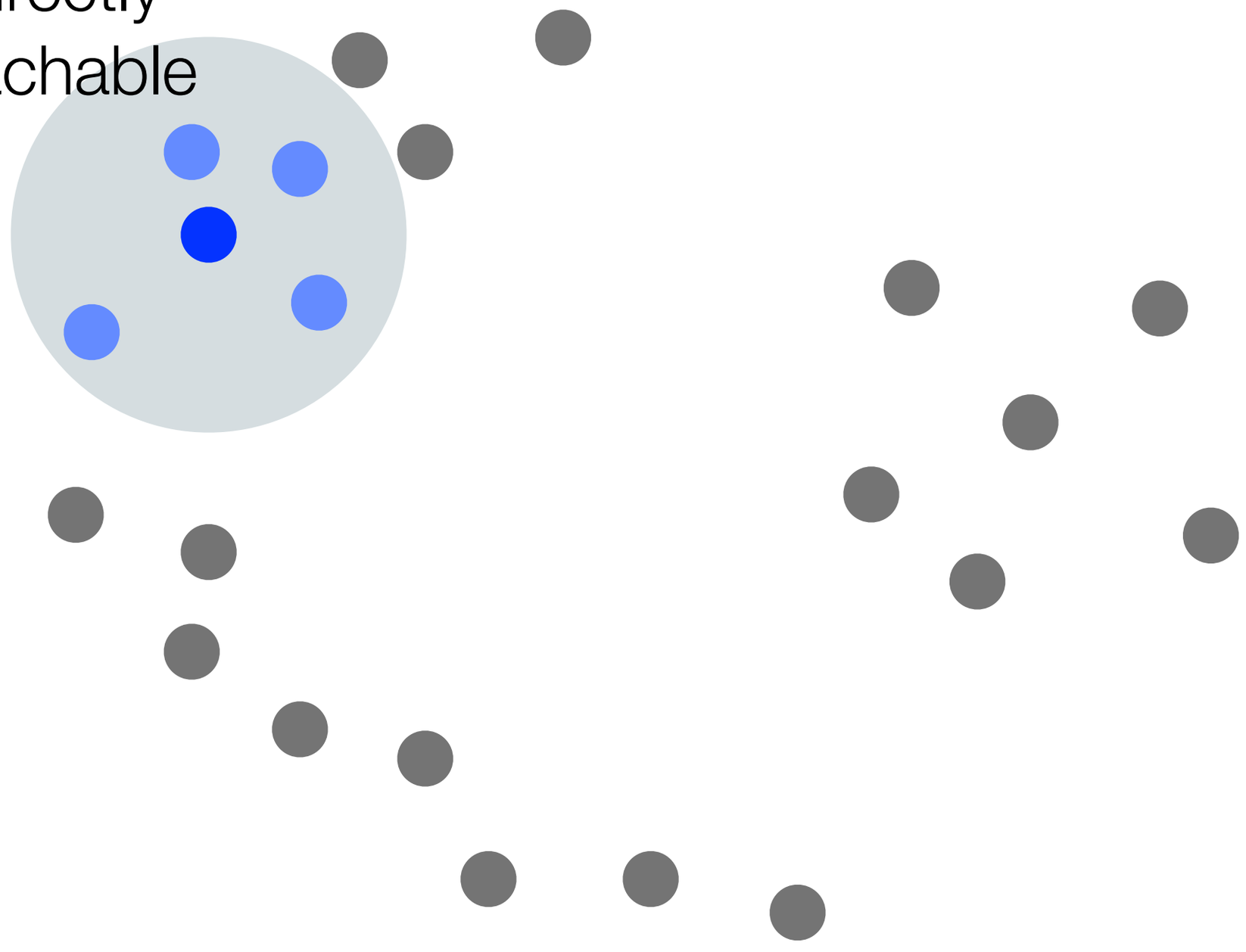


DBSCAN example

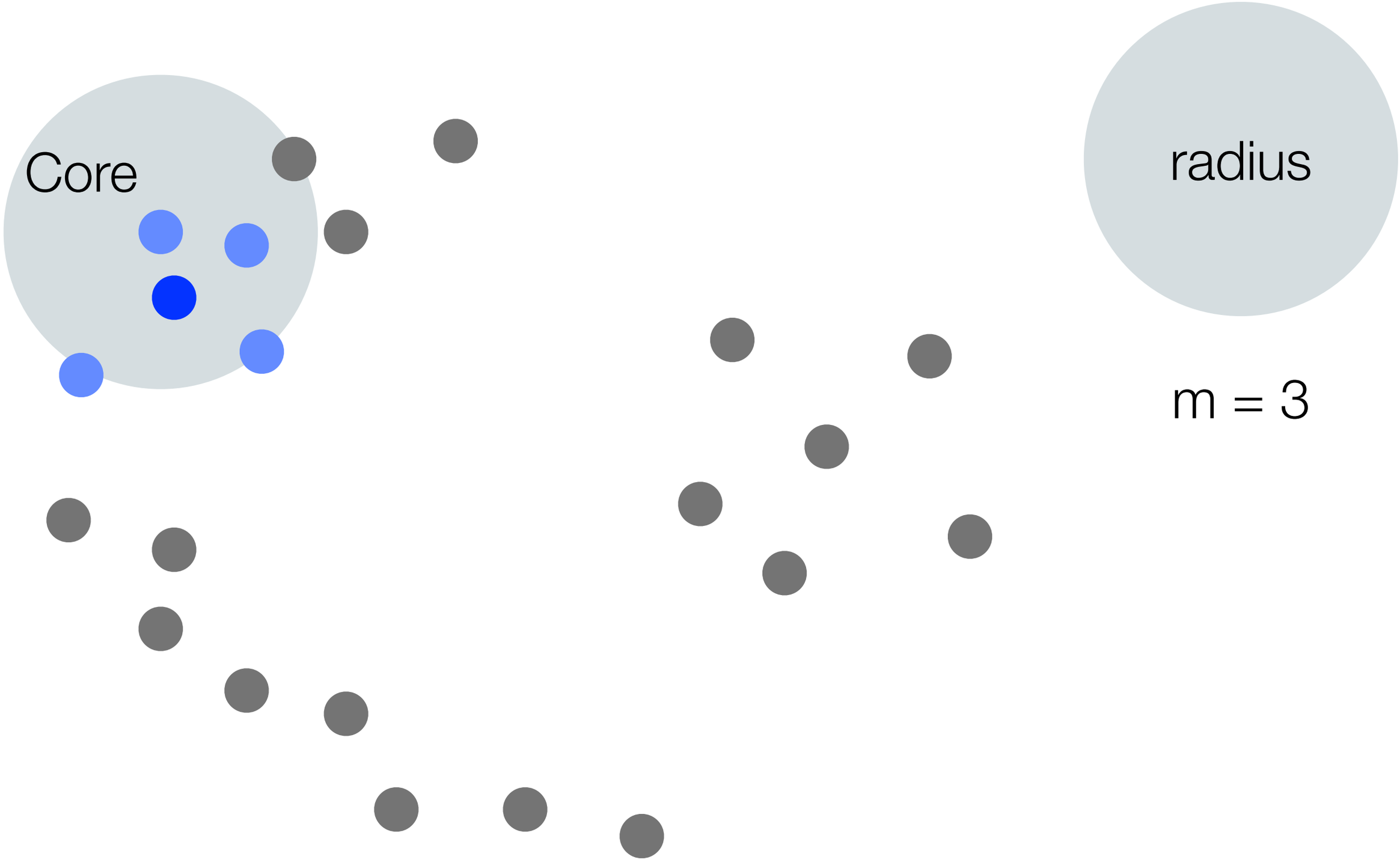
Directly reachable

radius

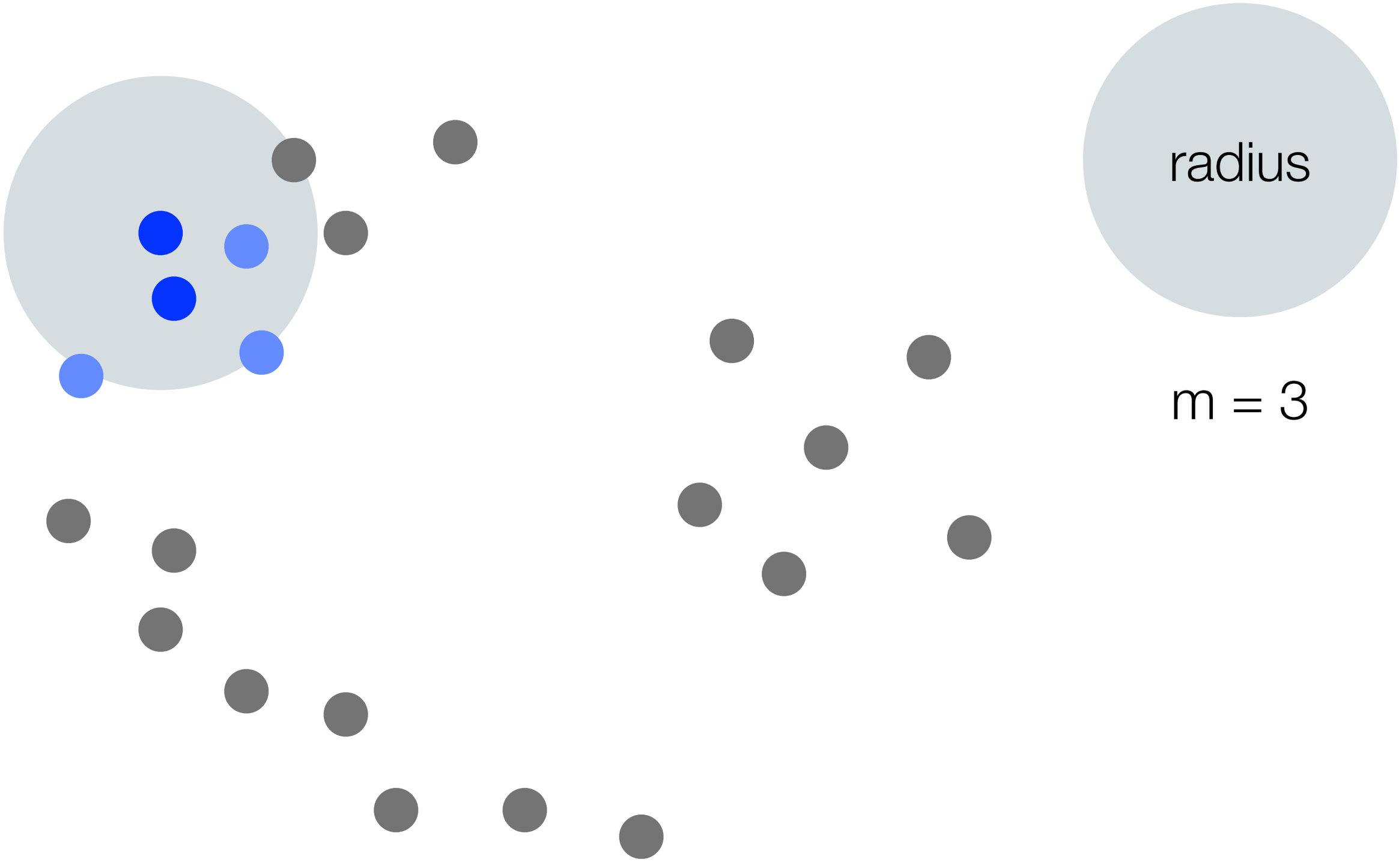
m = 3



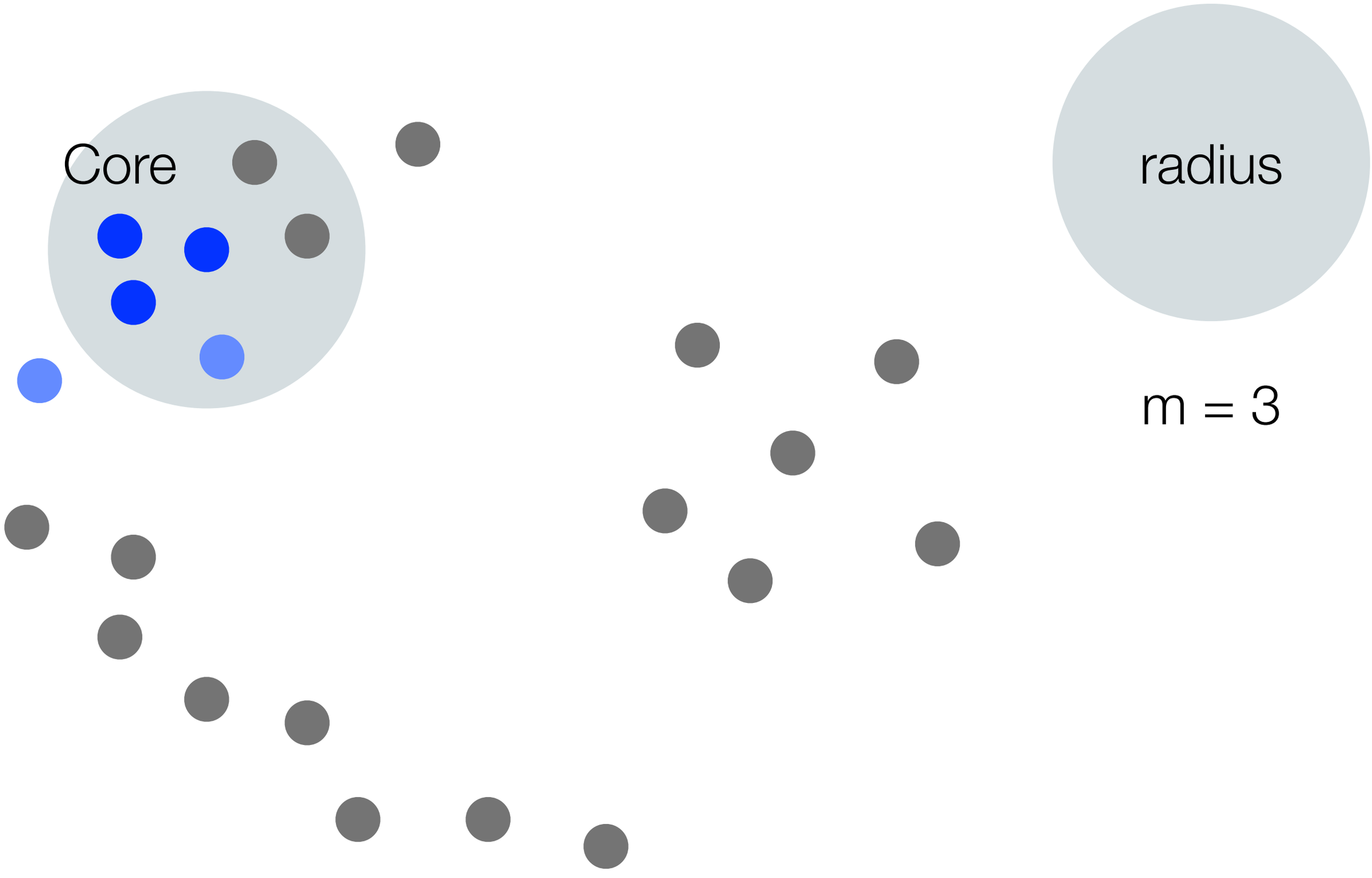
DBSCAN example



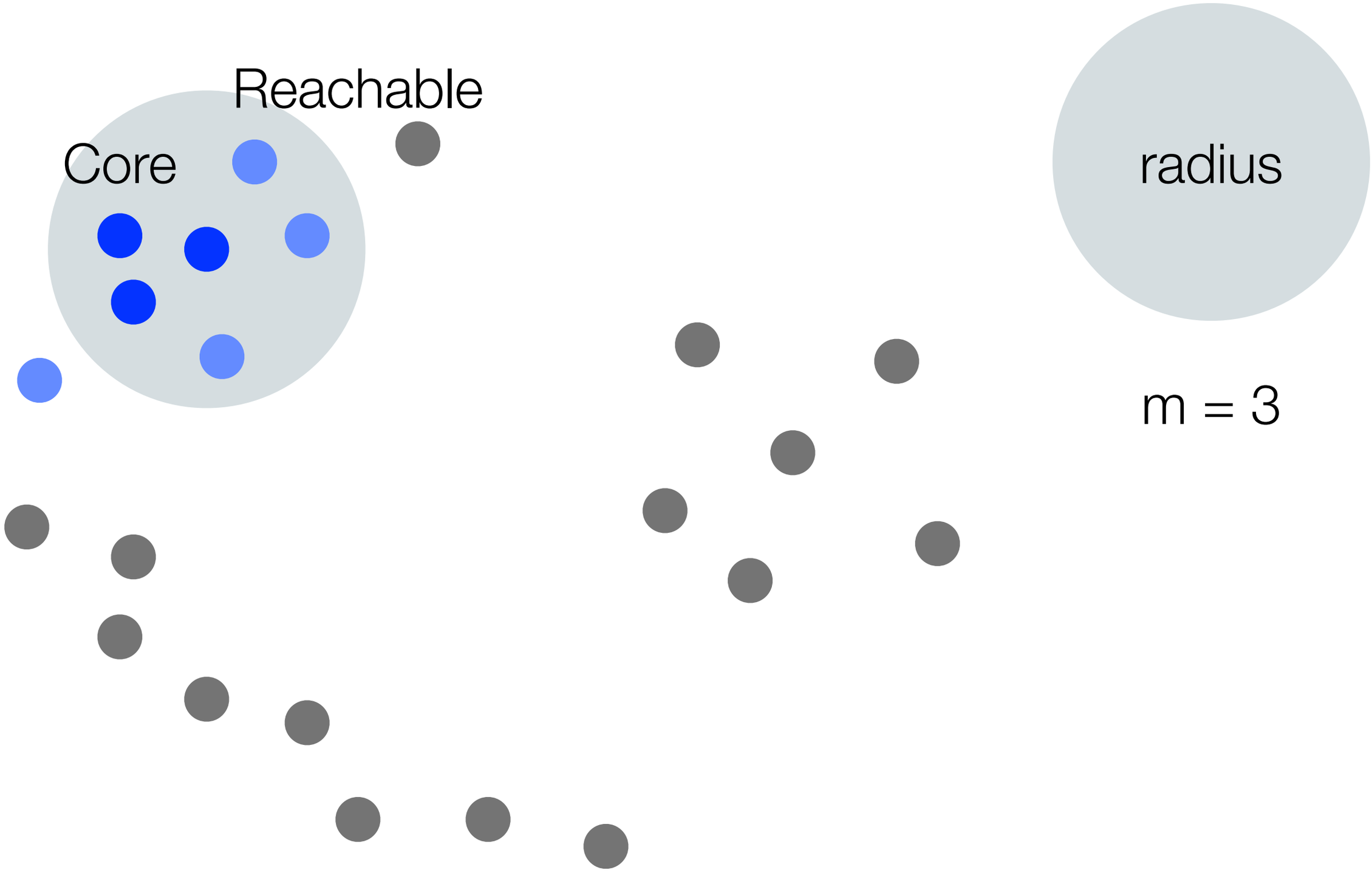
DBSCAN example



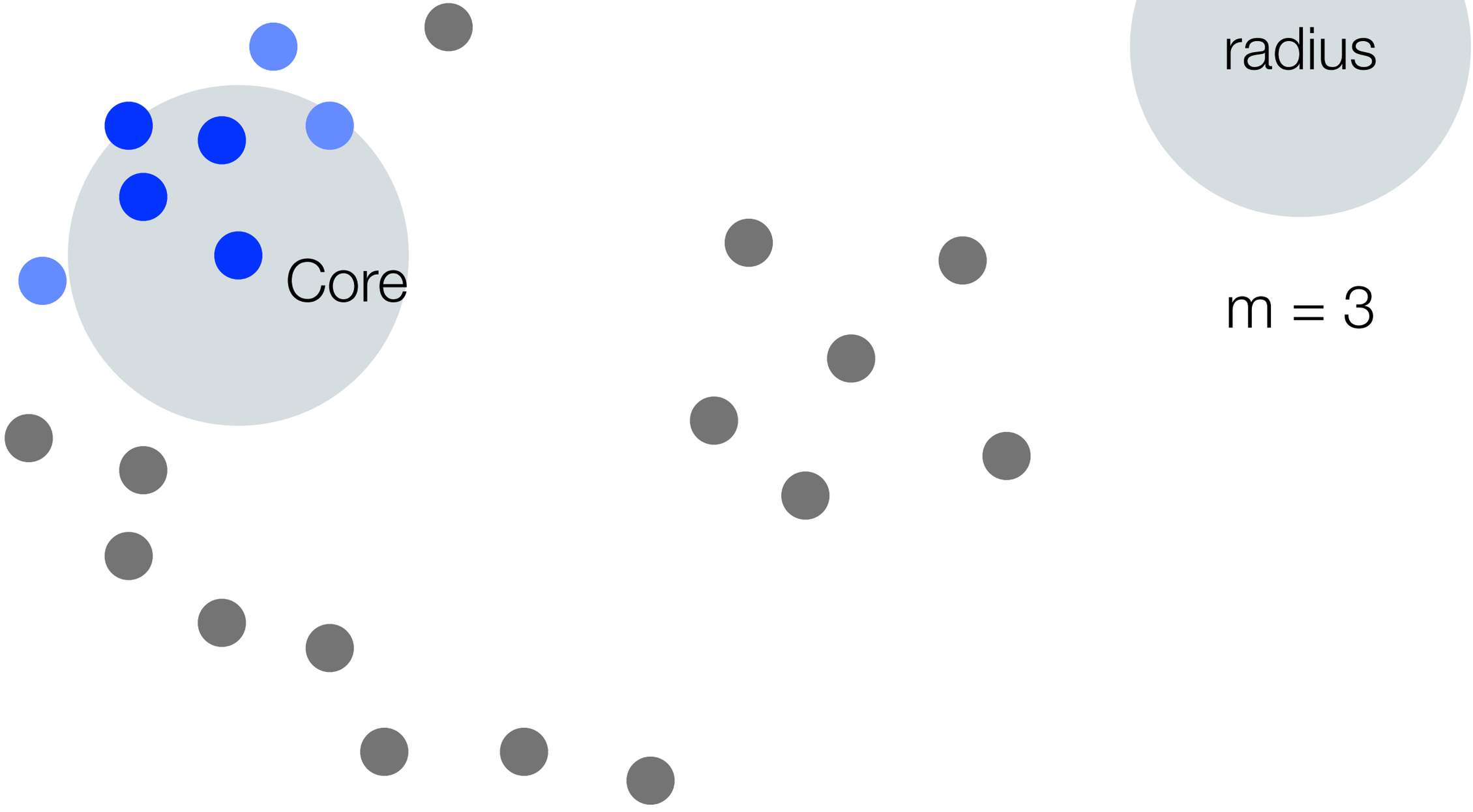
DBSCAN example



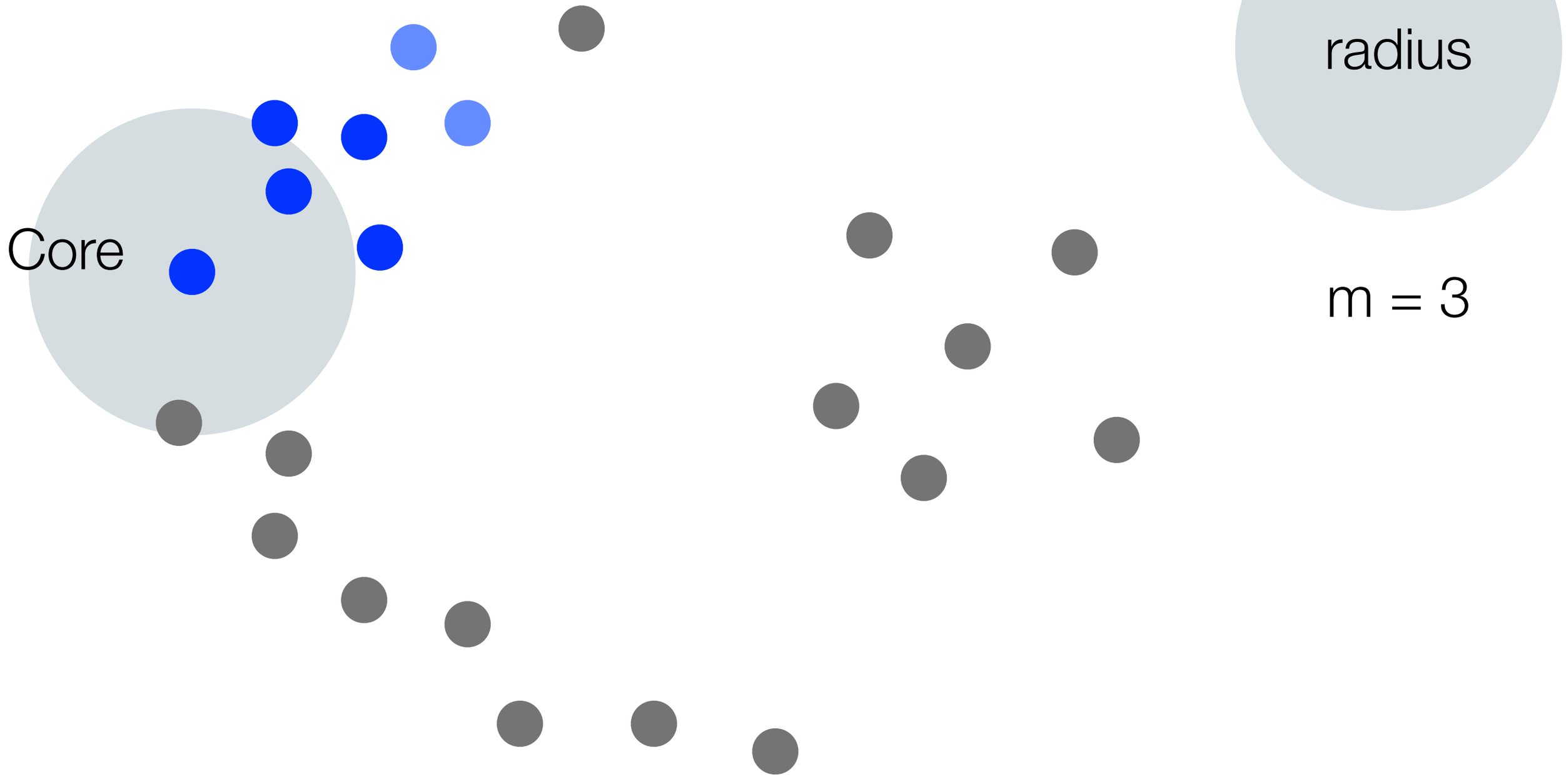
DBSCAN example



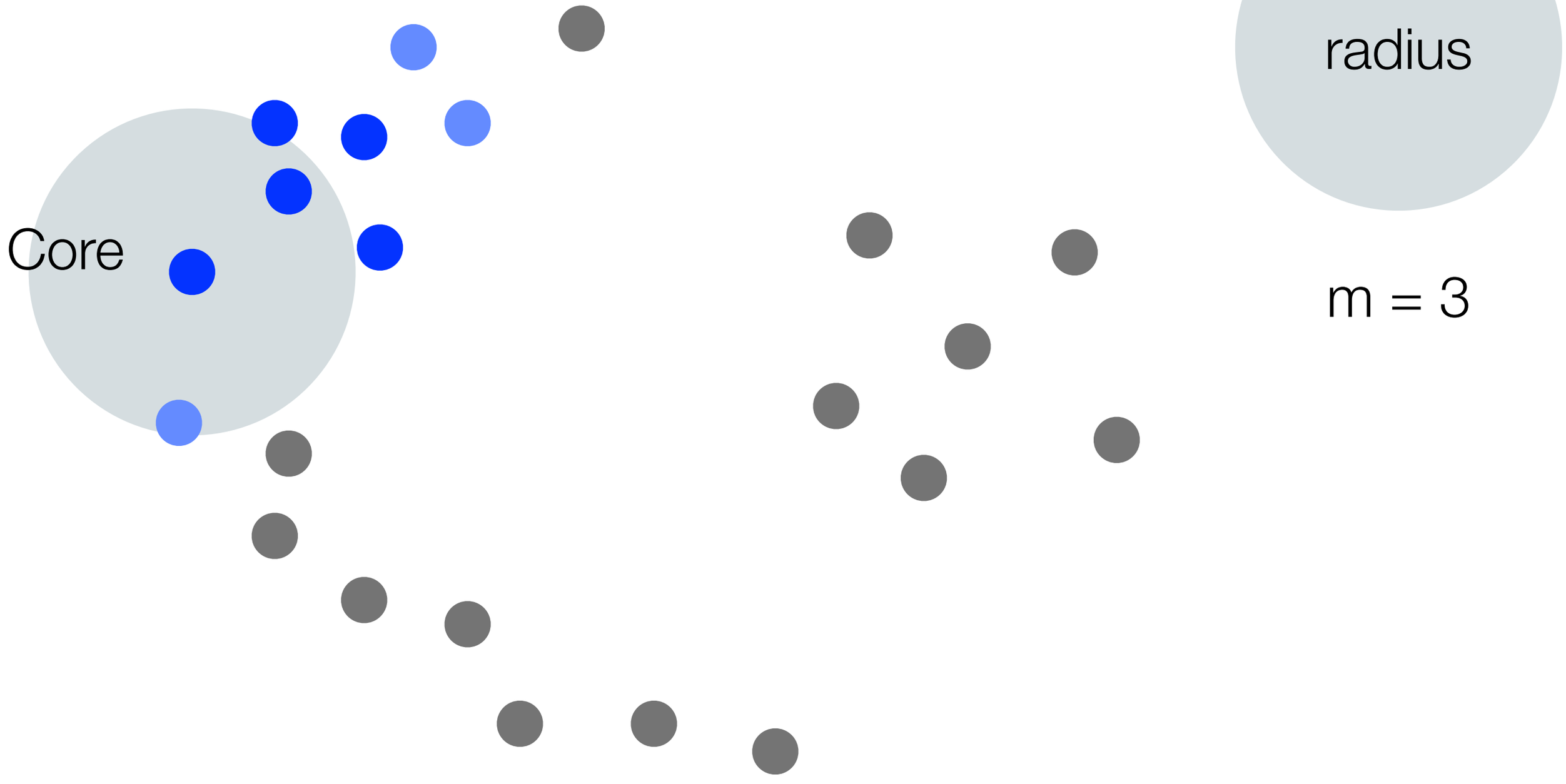
DBSCAN example



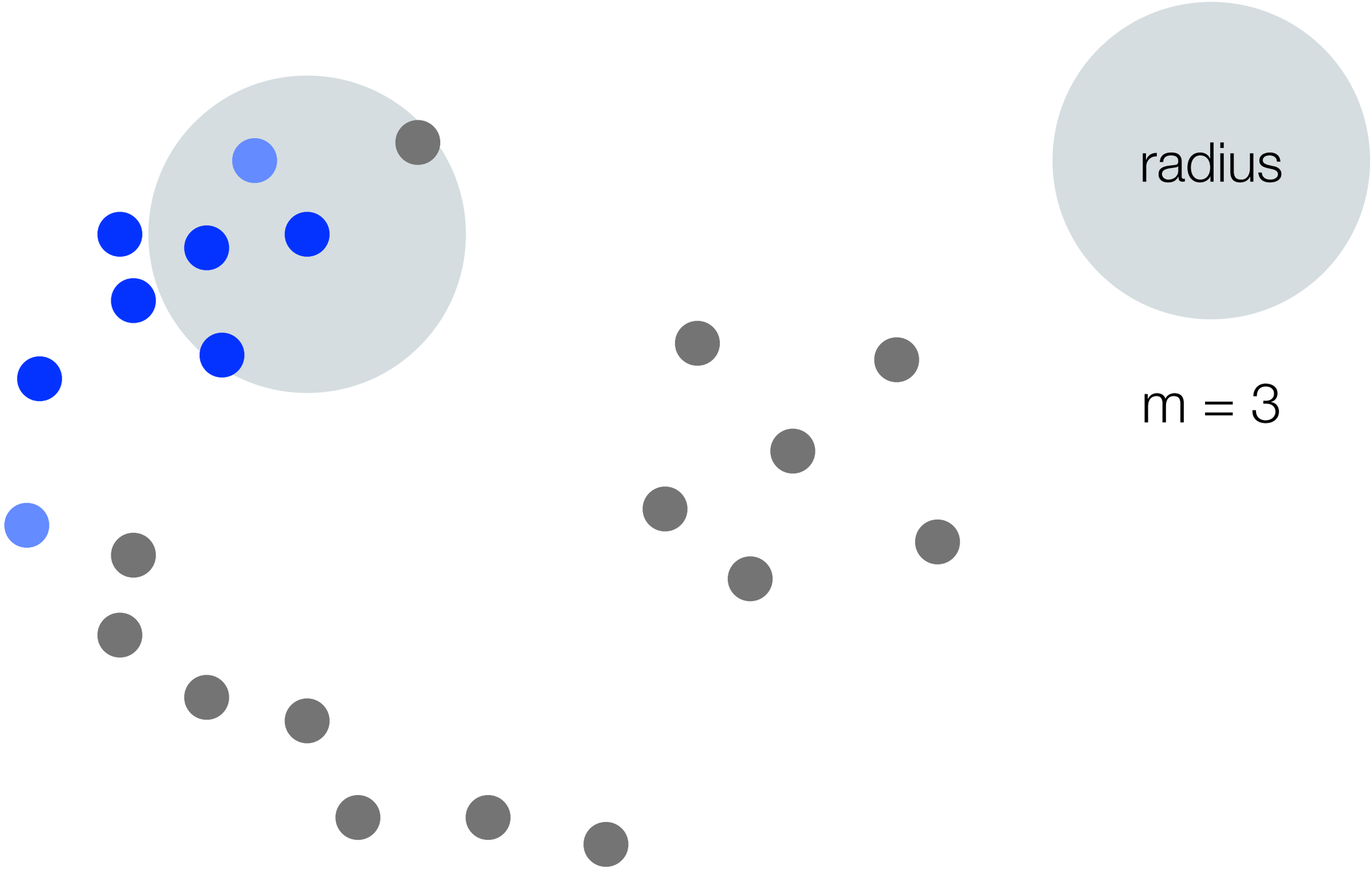
DBSCAN example



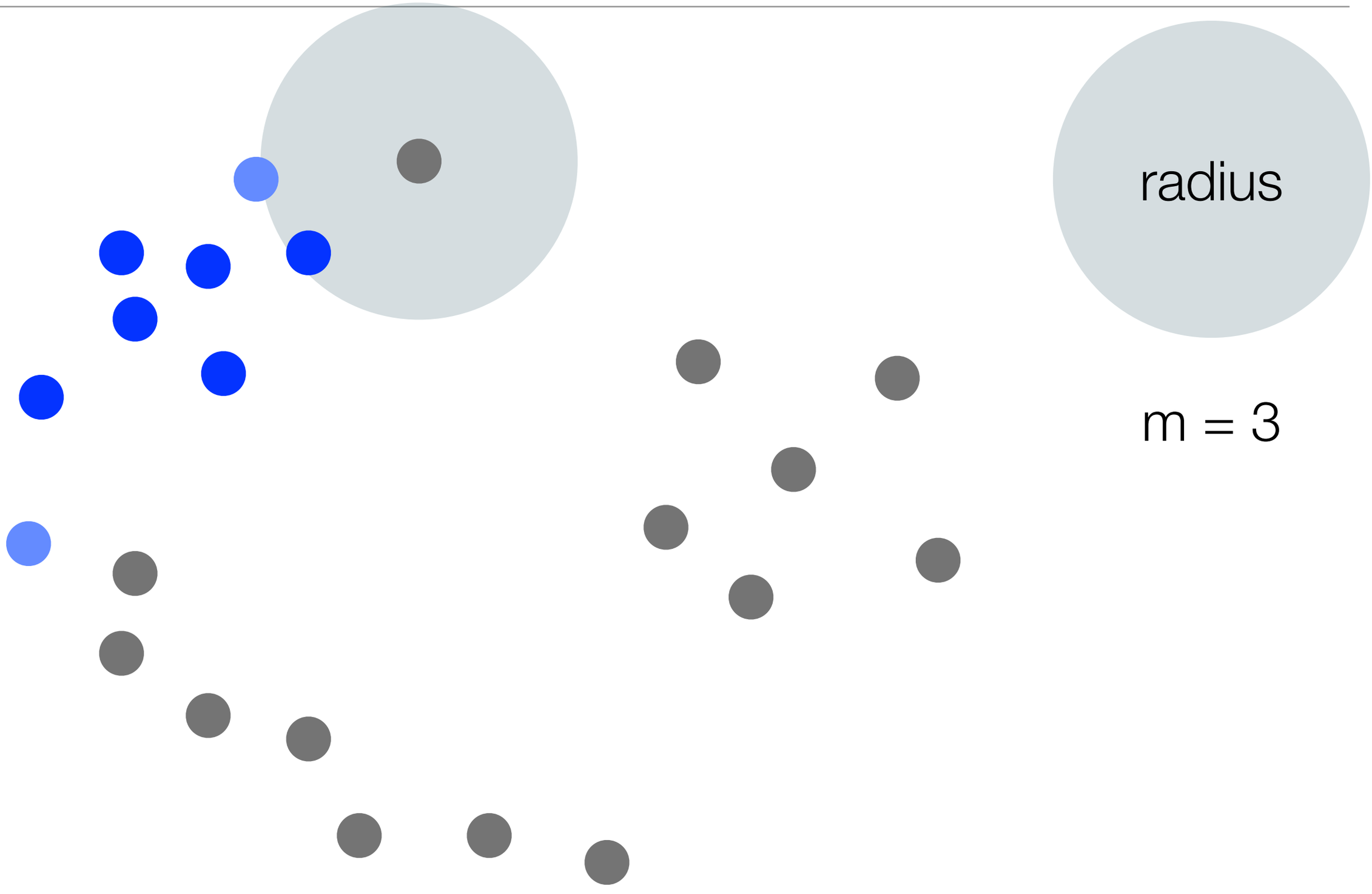
DBSCAN example



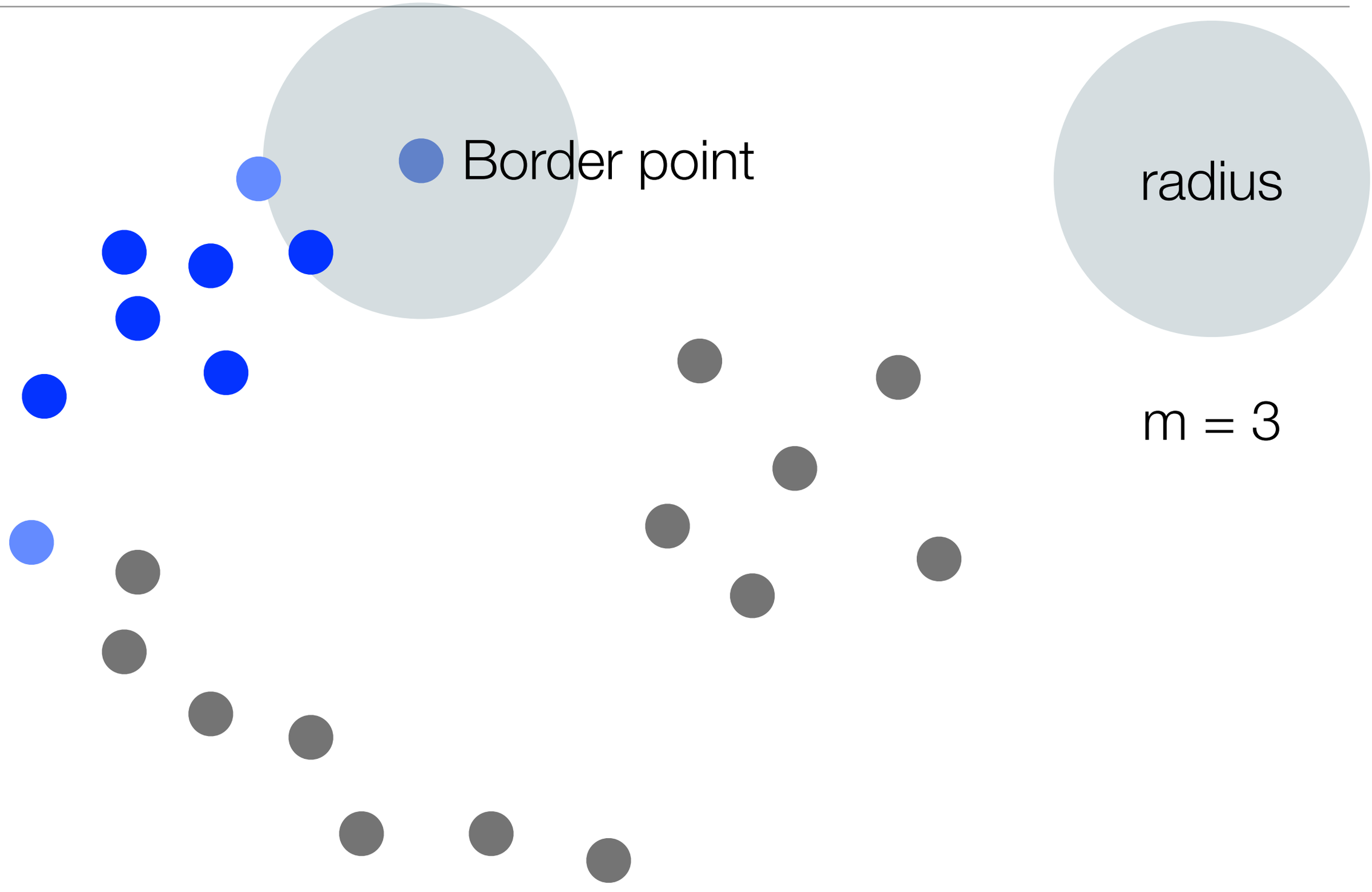
DBSCAN example



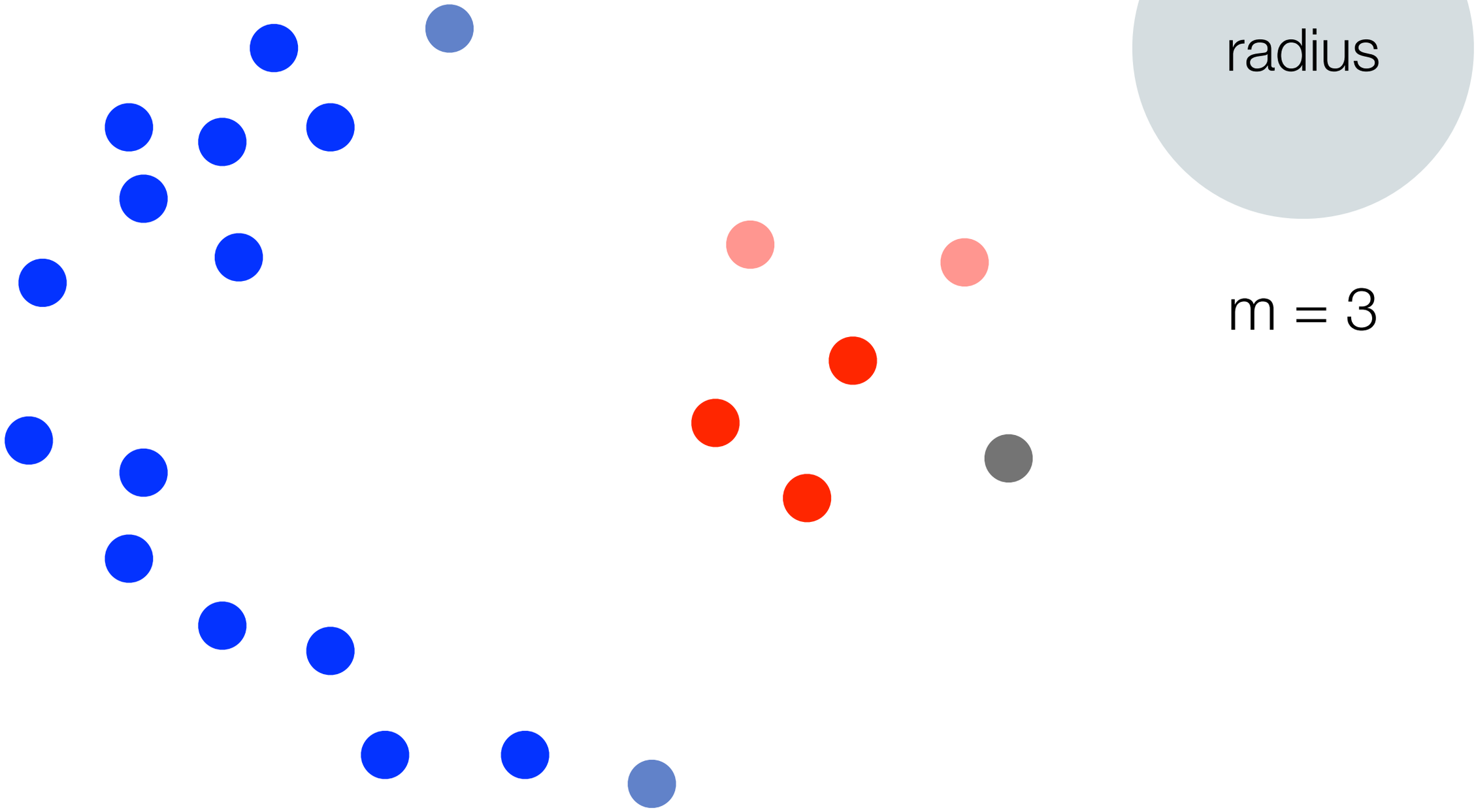
DBSCAN example



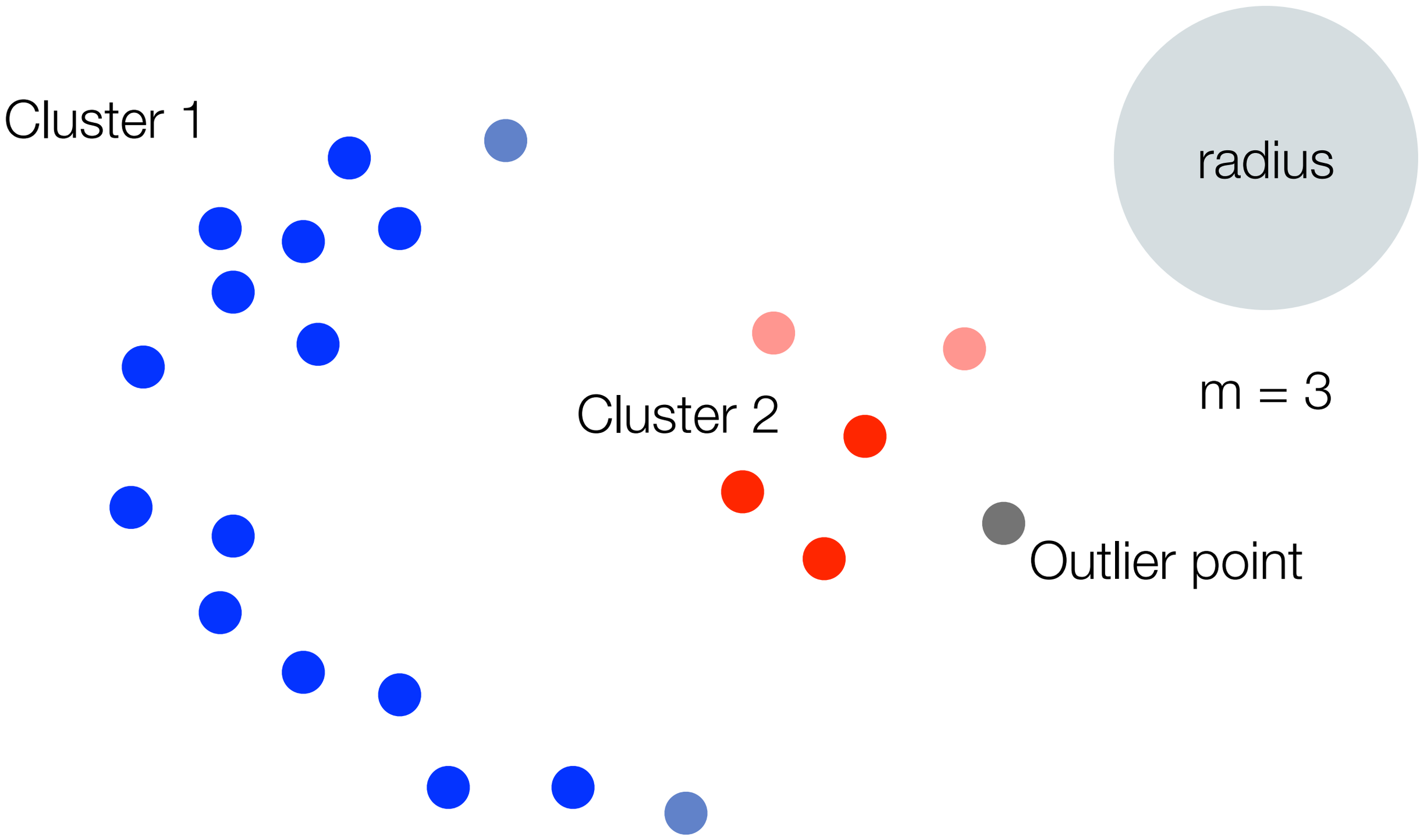
DBSCAN example



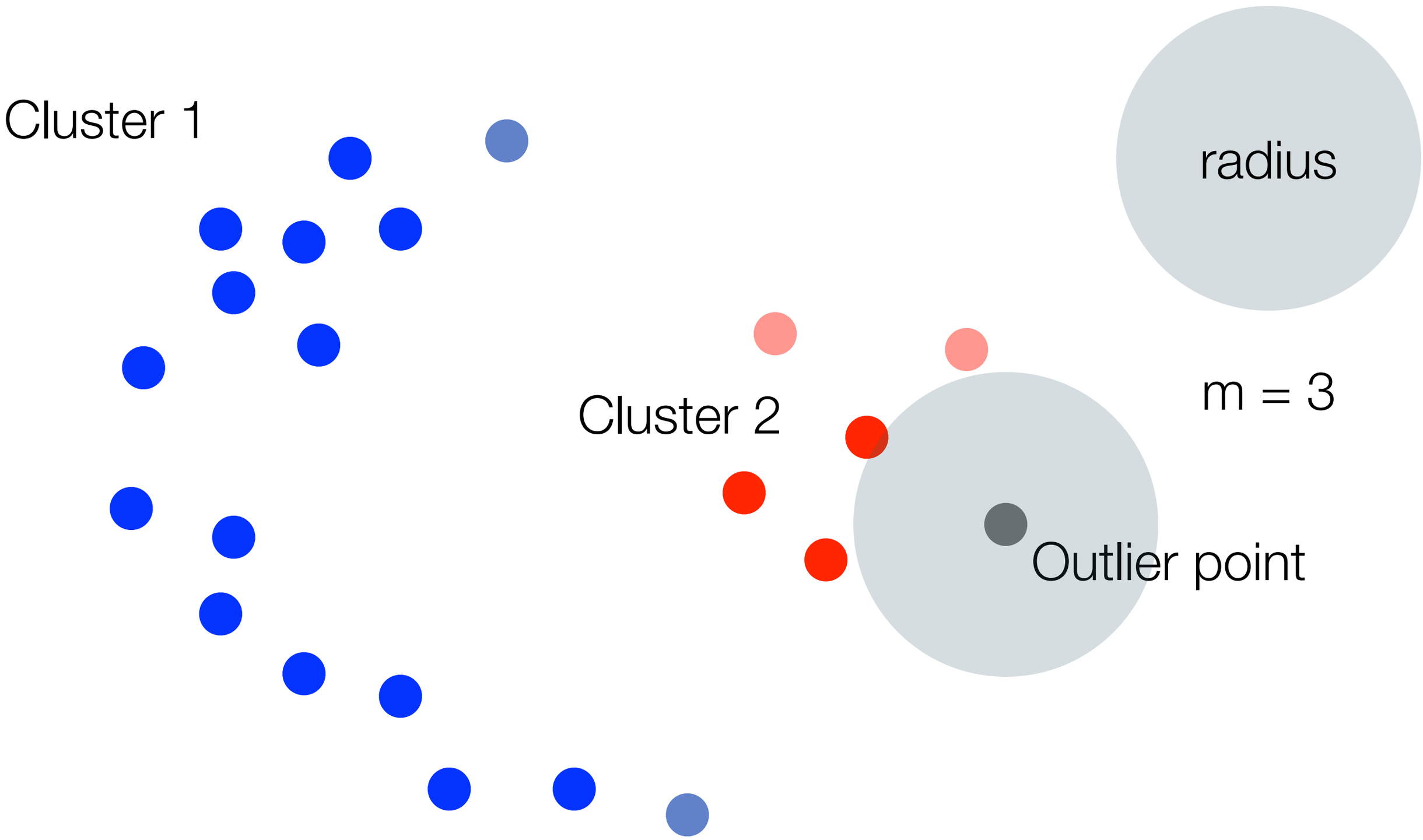
DBSCAN example



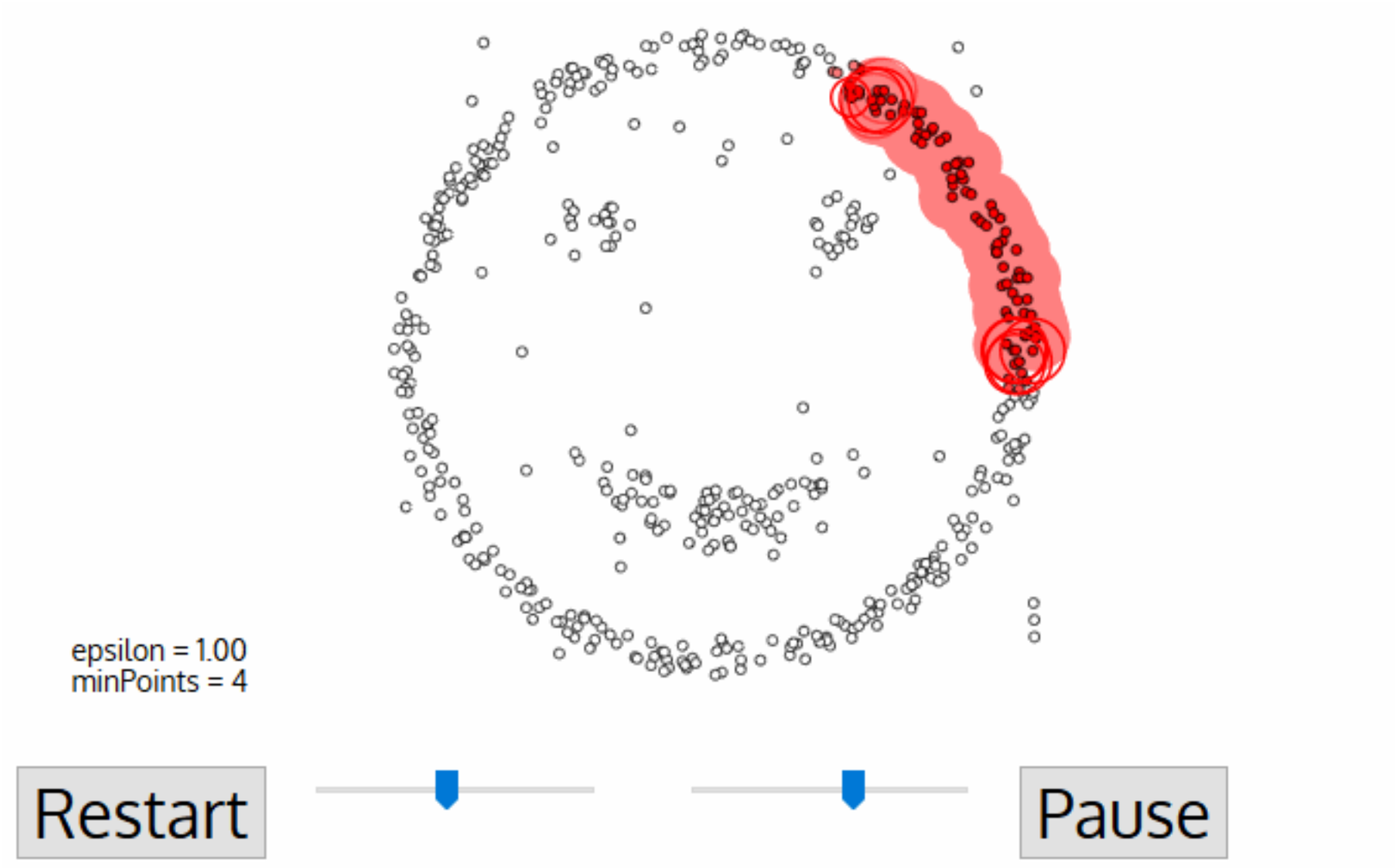
DBSCAN example



DBSCAN example



DBSCAN



https://towardsdatascience.com/the-5-clustering-algorithms-data-scientists-need-to-know-a36d136ef68

DBSCAN

- Can find non-convex clusters
- Automatically determines number of clusters needed
- Not every point goes into a cluster (handles outliers/noise; however can be a drawback if you want to assign all points to a cluster)
- Tends to find/work best with clusters of similar density
- How to choose radius & min points? There are rules of thumb but can be tricky! Often use min points = 2 x dim, for radius, can us elbow plot of a k-distance graph, but harder to say)

Model based methods: Gaussian Mixture Models

- Assumes the data points come from a combination of multivariate gaussians

- This seems restrictive but is often no more so than other methods (e.g. k-means in some sense assumes a centroid and resulting Voronoi diagram govern the data)

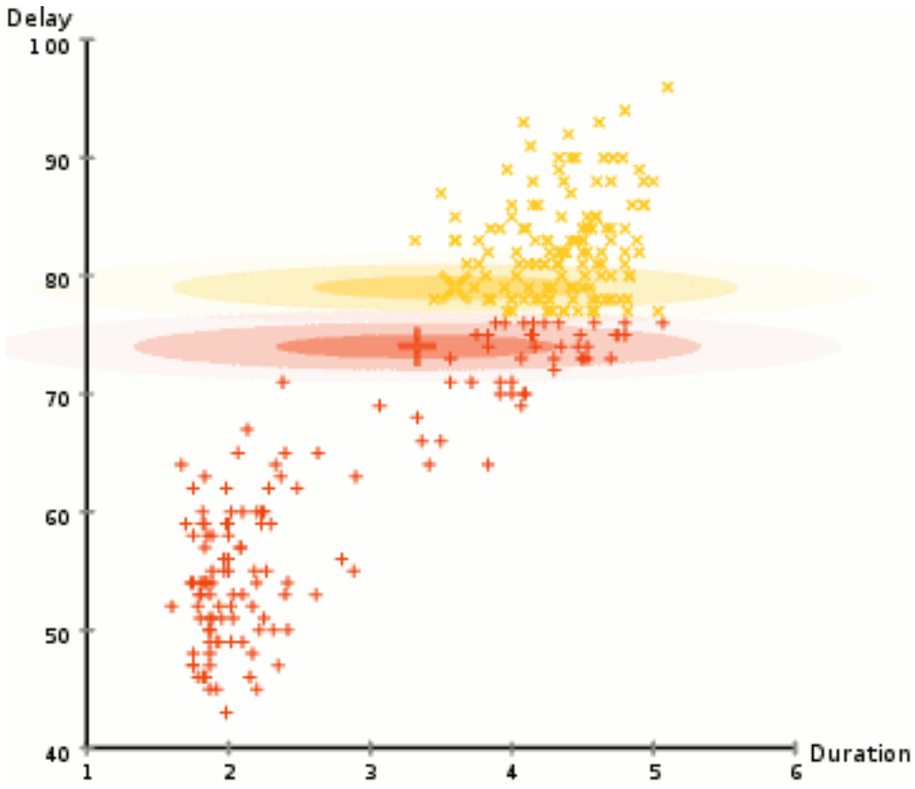
- Each data point has a probability of belonging to each cluster

- Often fit via expectation maximization (a type of maximum likelihood approach)

Model based methods: Gaussian Mixture Models

- Select number of clusters (number of gaussians to fit)
- Randomly initialize them (or better yet, use a method to pick a good starting guess)
- Compute the probability that each data point is in each cluster (based on the value of the gaussian at that point)
- Compute new parameters (μ,σ) for each gaussian that maximize this probability
- Repeat last two steps above until convergence

Model based methods: Gaussian Mixture Models



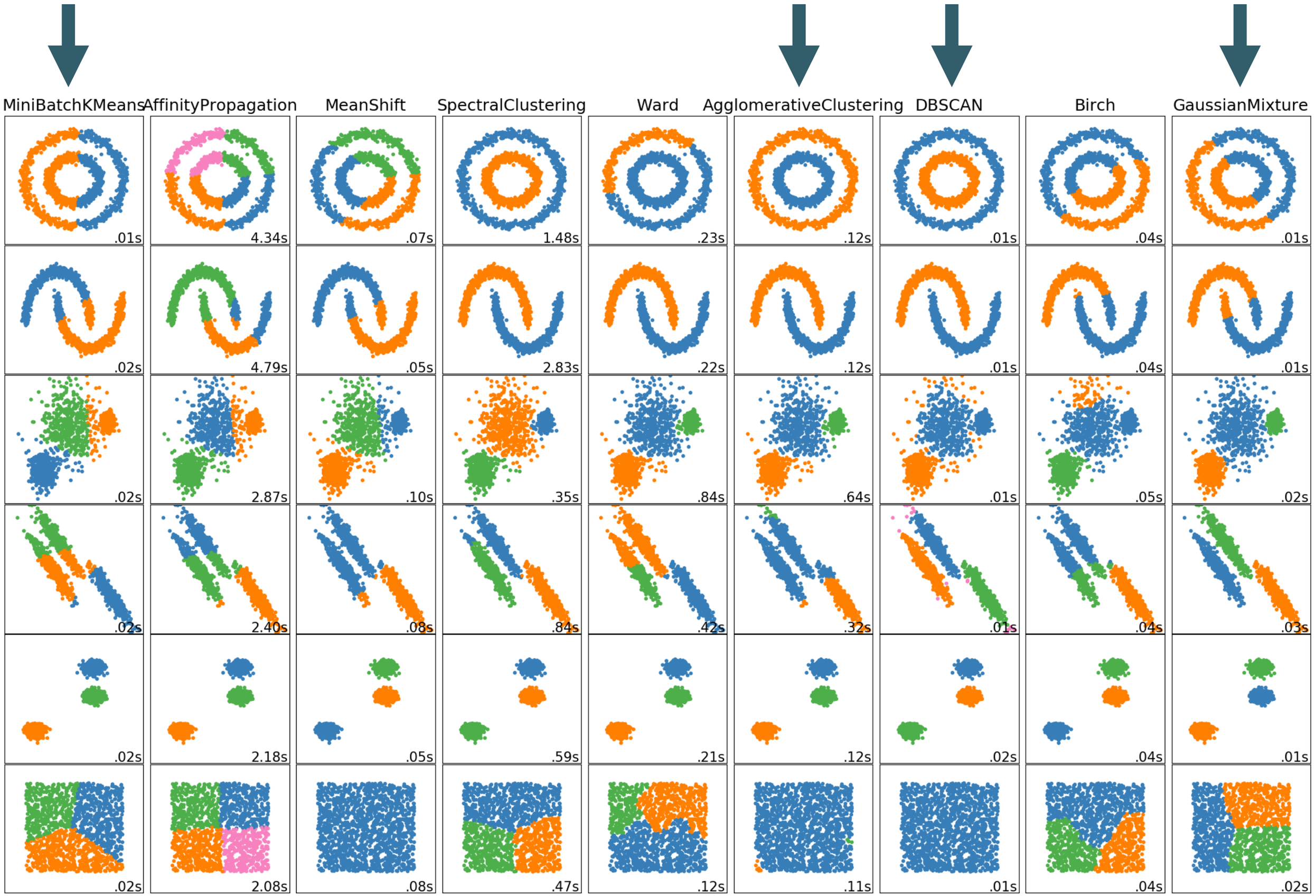
https://towardsdatascience.com/the-5-clustering-algorithms-data-scientists-need-to-know-a36d136ef68

Clustering methods

- Many different approaches! These are just a few examples

- Different methods behave better/worse on different data sets

- Testing how well a clustering method behaves can be difficult, especially in high dimensions and/or without ground truth information



https://towardsdatascience.com/the-5-clustering-algorithms-data-scientists-need-to-know-a36d136ef68

Resources

- https://en.wikipedia.org/wiki/Cluster_analysis
- https://en.wikipedia.org/wiki/DBSCAN
- https://medium.com/predict/three-popular-clustering-methods-and-when-to-use-each-4227c80ba2b6
- https://blog.dominodatalab.com/topology-and-density-based-clustering/
- https://shapeofdata.wordpress.com/2014/03/04/k-modes/
- https://towardsdatascience.com/the-5-clustering-algorithms-data-scientists-need-to-know-a36d136ef68

Clustering in networks

- Many different ways to look at clustering

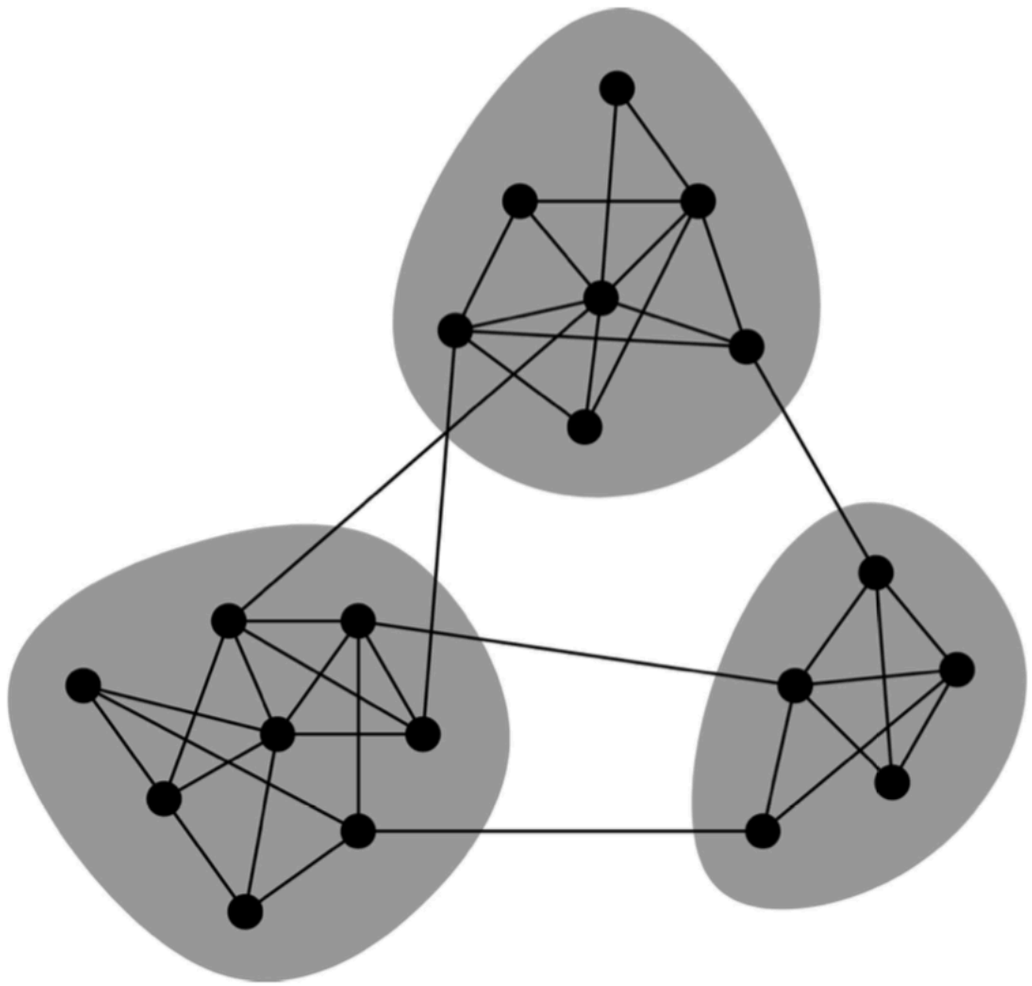
- Community detection - finding clusters (groups) of nodes that are highly connected within the group and less connected between groups (i.e. clustering, where similarity is based on connectivity)

- How do node traits (degree, covariates) cluster based on edges? E.g. do smokers tend to be friends with other smokers? Do individuals cluster by popularity

Network methods: **modularity** maximization

- Community (cluster) detection approach for networks

- Looks for groups of nodes that have more within-group edges than would be expected from a random graph with the same degree for each node



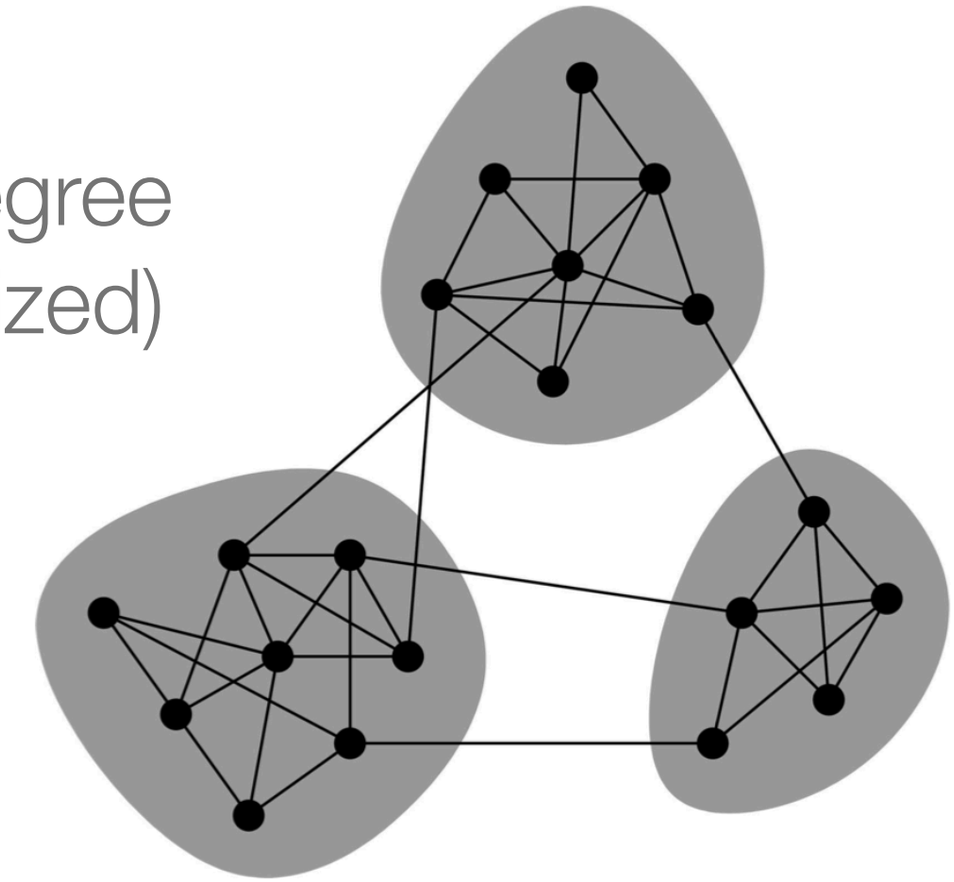
Modularity and community structure in networks. M. E. J. Newman. PNAS 2006, 103 (23) 8577-8582; DOI: 10.1073/pnas.0601602103

Modularity

- Modularity compares observed community edges to what would be expected at random

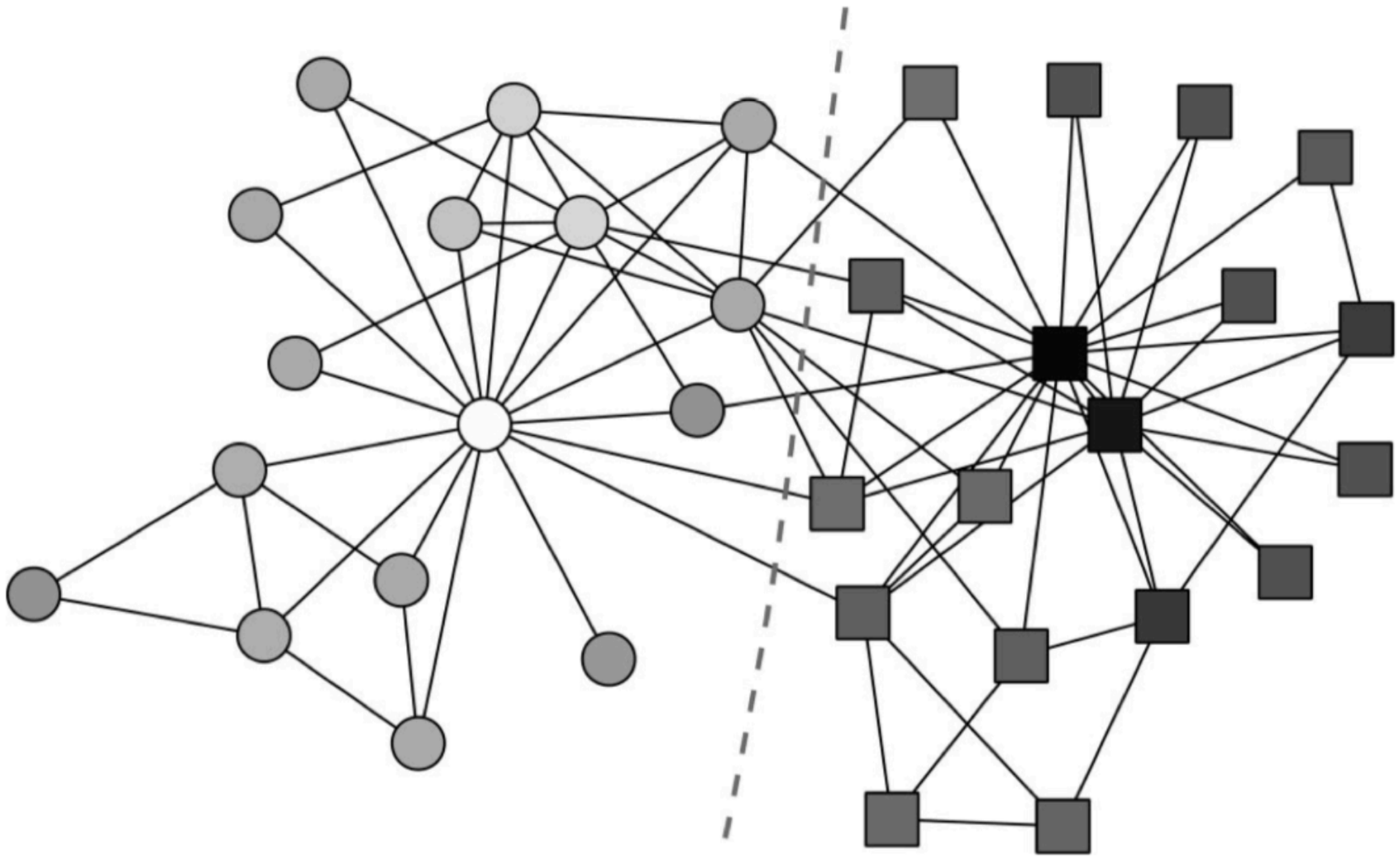
- Modularity is the fraction of within-group edges minus the fraction expected at random (if degree conserved but edges are randomized)

- Modularity-based community detection: find community groupings that maximize modularity



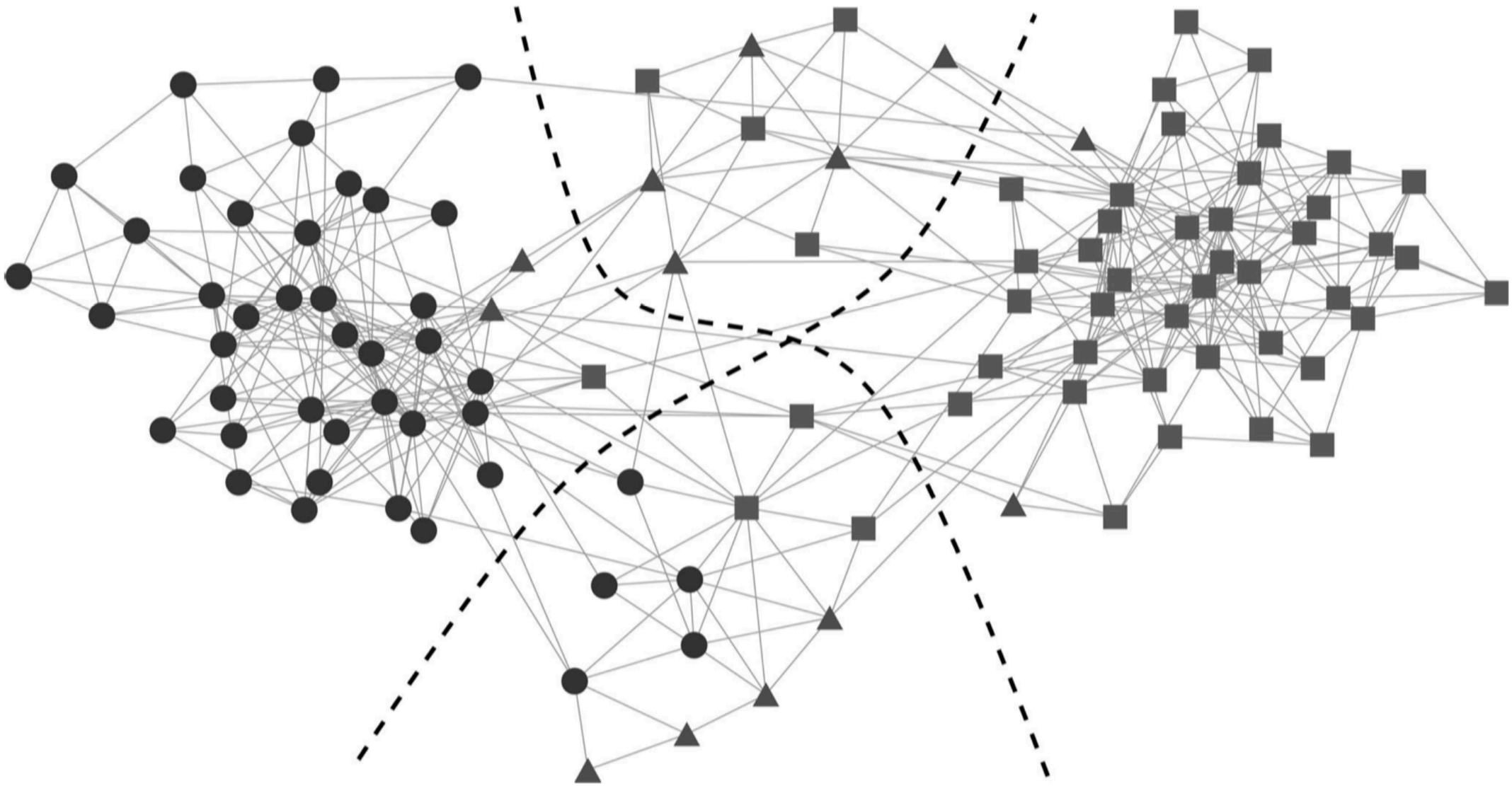
Newman, Mark EJ. "Modularity and community structure in networks." *PNAS* 103.23 (2006): 8577-8582.

Karate club example



Newman, Mark EJ. "Modularity and community structure in networks." *PNAS* 103.23 (2006): 8577-8582.

Political books



Newman, Mark EJ. "Modularity and community structure in networks." *PNAS* 103.23 (2006): 8577-8582.

Modularity

- Can be slow/difficult to maximize—spectral methods have made much faster

- Resolution limit - as the network grows larger, it is harder for modularity-based community detection methods to find small communities

Network methods: **assortativity**

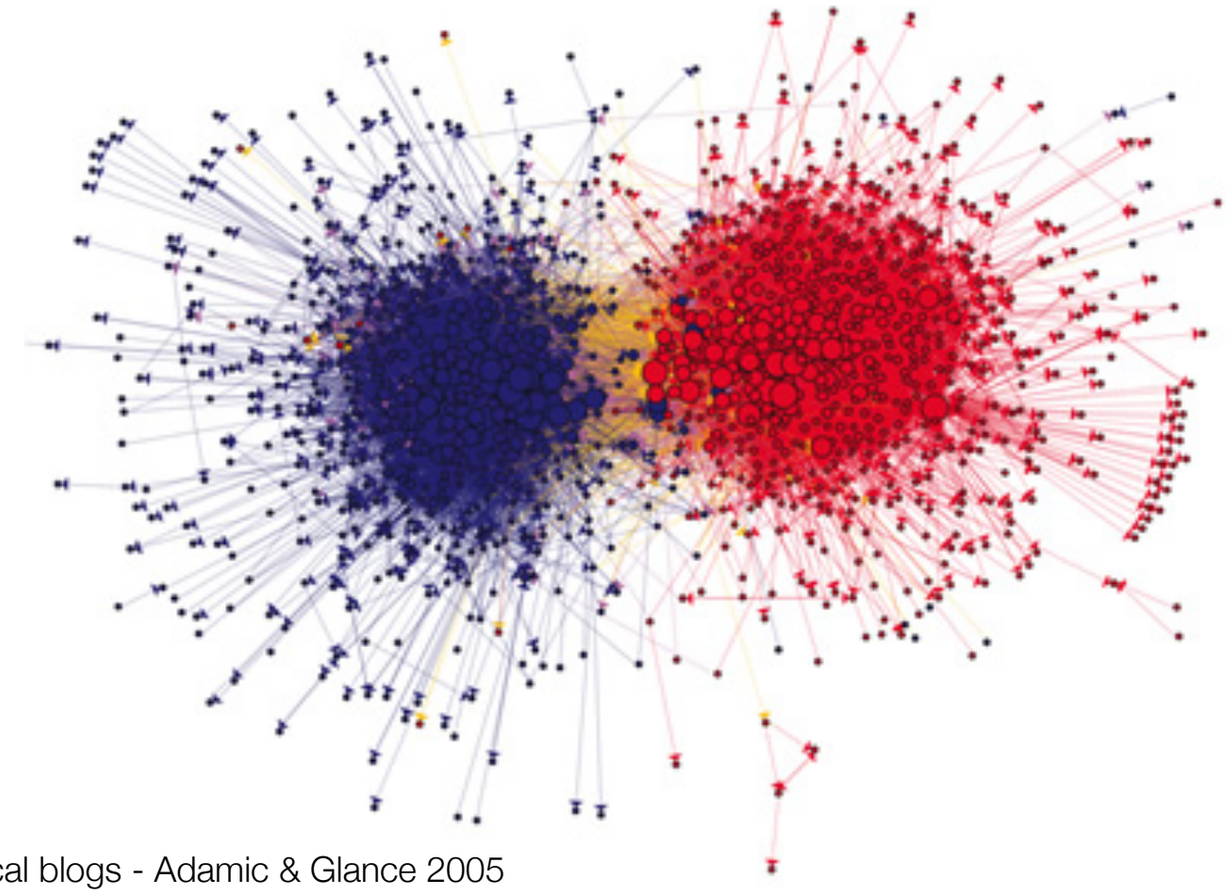
- **Assortativity** - measures network-level tendency for nodes to to attach to similar nodes

- Not really for cluster (community) detection, so much as to evaluate how clustered a given property is on the network

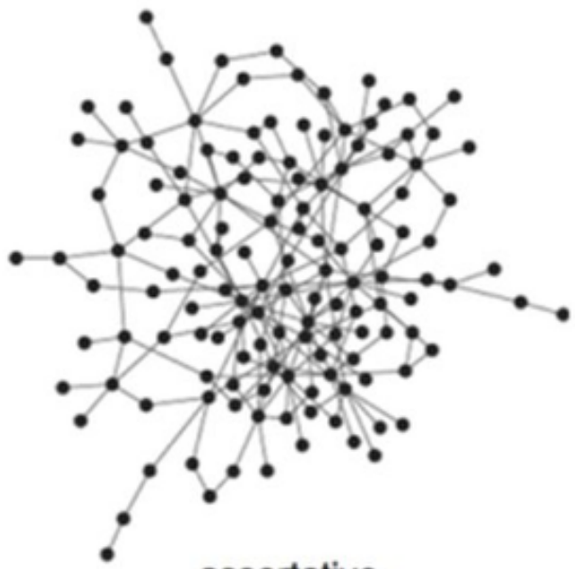
- Often look at clustering of degree, but can be other properties (e.g. how is the network clustered by gender, vaccination, smoking behaviors, etc.)

Assortativity

- Heterosexual networks - highly dissassortative by gender
- Social/sexual networks often assortative on a range of demographic, degree, behavioral traits - ‘birds of a feather flock together’



Political blogs - Adamic & Glance 2005



assortative

disassortative

Assortativity

- Calculate fraction of edges between nodes of the same type/value, compare to what would be expected from a random network

- Ranges from -1 (dissassortative) to 1 (assortative)

    - But min value (most dissassortative) is between -1 and 0 depending on the composition of the network

M. E. J. Newman, Mixing patterns in networks Physical Review E, 67 026126, 2003

Assortativity

- Consider a case where we have discrete characteristics on the nodes

- Define a mixing matrix with entries e_{ij} given by the fraction of the total edges linking type *i* to type *j*

- Let a_i and b_i be the total fractions of each end type that we have ($a_i = b_i$ for undirected graphs)

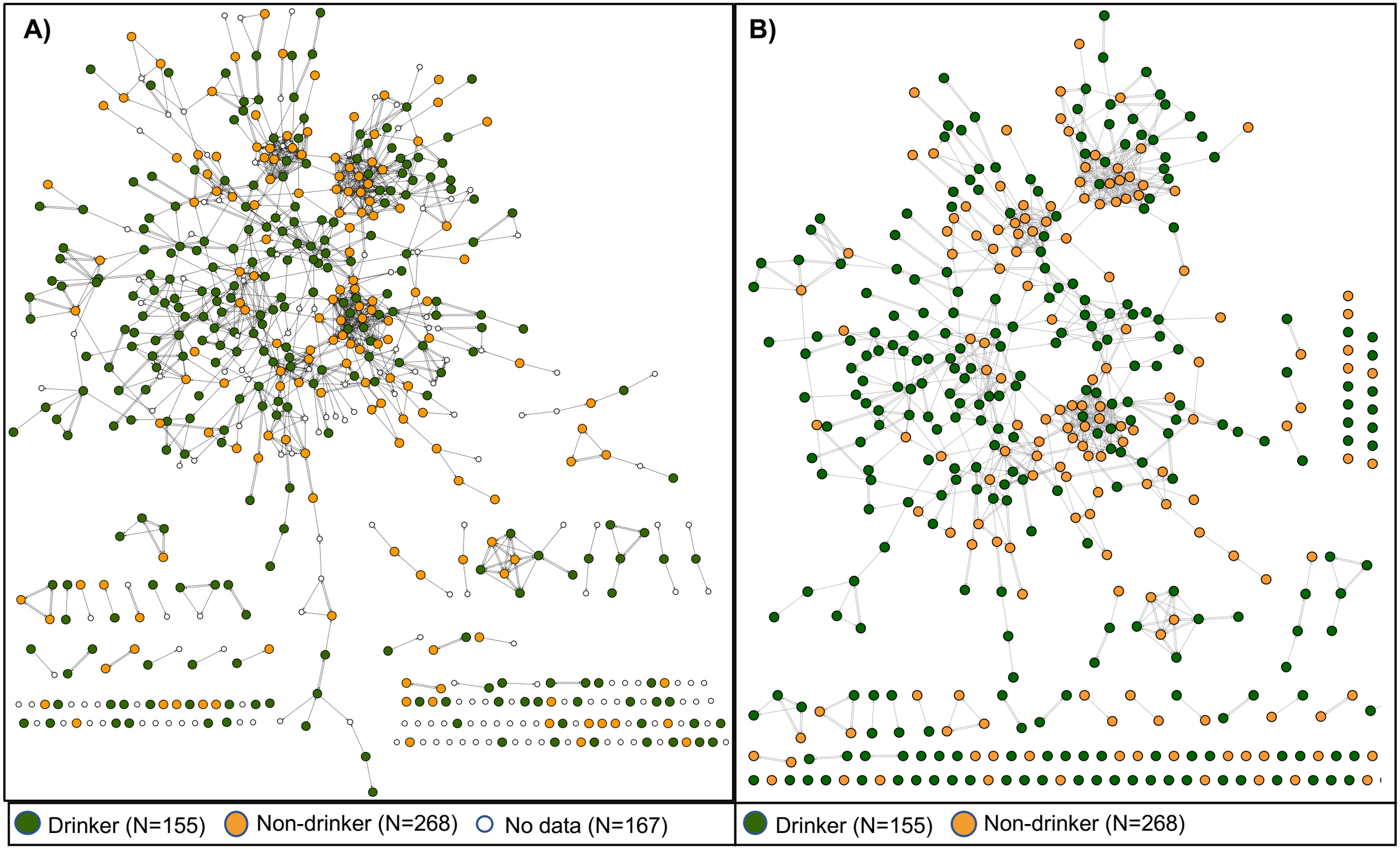
- Note that $\sum_{ij} e_{ij} = 1, \quad \sum_{j} e_{ij} = a_i, \quad \sum_{i} e_{ij} = b_j$

Assortativity

- Defined based on a mixing matrix - entries are the fraction of edges in a network linking type i to type j

$$r = \frac{\sum_i e_{ii} - \sum_i a_i b_i}{1 - \sum_i a_i b_i} = \frac{\mathbf{Tre} - ||\mathbf{e}^2||}{1 - ||\mathbf{e}^2||},$$

- For degree assortativity (and other scalar variables), assortativity is the Pearson correlation coefficient of degree between pairs of linked nodes



Ali Walsh Dissertation, 2019 (assortativity 0.2)