Machine Learning for Data Science (CS4786)
Lecture 9

Single Link Clustering, Spectral Clustering

Course Webpage :
http://www.cs.cornell.edu/Courses/cs4786/2016fa/

Lets build an Algorithm

$$M_5 = \sum_{j=1}^{K} \sum_{t \in C_j} \left\| \mathbf{x}_t - \mathbf{r}_j \right\|_2^2$$

$$\text{where } \mathbf{r}_j = \frac{1}{|C_j|} \sum_{t \in C_j} \mathbf{x}_t$$

Demo

K-MEANS CLUSTERING

- For all $j \in [K]$, initialize cluster centroids $\hat{\mathbf{r}}_j^1$ randomly and set $m = 1$
- Repeat until convergence (or until patience runs out)
  1. For each $t \in \{1, \ldots, n\}$, set cluster identity of the point
  $$\hat{c}^m(\mathbf{x}_t) = \underset{j \in [K]}{\operatorname{argmin}} \|\mathbf{x}_t - \hat{\mathbf{r}}_j^m\|$$
  2. For each $j \in [K]$, set new representative as
  $$\hat{\mathbf{r}}_j^{m+1} = \frac{1}{|\hat{C}_j^m|} \sum_{t \in \hat{C}_j^m} \mathbf{x}_t$$
  3. $m \leftarrow m + 1$

K-means objective

$$O\left(c; \mathbf{r}_1, \ldots, \mathbf{r}_K\right) = \sum_{j=1}^{K} \sum_{c(\mathbf{x}_t)=j} \|\mathbf{x}_t - \mathbf{r}_j\|_2^2$$

Minimize above objective over c and $\mathbf{r}_1, \ldots, \mathbf{r}_K$

$$\sum_{j=1}^{K} \sum_{t \in C_j} \left\| \mathbf{x}_t - \frac{1}{|C_j|} \sum_{s \in C_j} \mathbf{x}_s \right\|^2 = \min_{\mathbf{r}_1, \ldots, \mathbf{r}_K} \sum_{j=1}^{K} \sum_{t \in C_j} \|\mathbf{x}_t - \mathbf{r}_j\|^2$$

$$||$$

$$M_5 = \min_{\mathbf{r}_1, \ldots, \mathbf{r}_K} O(c; \mathbf{r}_1, \ldots, \mathbf{r}_K)$$

Fact: Centroid is Minimizer

$$\forall \mathbf{r}_j, \quad \sum_{t \in C_j} \left\| \mathbf{x}_t - \frac{1}{|C_j|} \sum_{s \in C_j} \mathbf{x}_s \right\|^2 \leq \sum_{t \in C_j} \|\mathbf{x}_t - \mathbf{r}_j\|^2$$

Proof

$$\sum_{t\in C_j} \|\mathbf{x}_t - \mathbf{r}_j\|^2$$

$$\mu_j = \frac{1}{|C_j|}\sum_{t\in C_j}\mathbf{x}_t$$

$$= \sum_{t\in C_j} \|\mathbf{x}_t - \mu_j + \mu_j - \mathbf{r}_j\|^2$$

$$= \sum_{t\in C_j} \|\mathbf{x}_t - \mu_j\|^2 + \sum_{t\in C_j} \|\mu_j - \mathbf{r}_j\|^2 + 2\sum_{t\in C_j} (\mathbf{x}_t - \mu_j)^\top (\mu_j - \mathbf{r}_j)$$

$$= \sum_{t\in C_j} \|\mathbf{x}_t - \mu_j\|^2 + \sum_{t\in C_j} \|\mu_j - \mathbf{r}_j\|^2 + 2\left(\sum_{t\in C_j}\mathbf{x}_t - |C_j|\mu_j\right)^\top (\mu_j - \mathbf{r}_j)$$

$$= \sum_{t\in C_j} \|\mathbf{x}_t - \mu_j\|^2 + \sum_{t\in C_j} \|\mu_j - \mathbf{r}_j\|^2$$

$$\geq \sum_{t\in C_j} \|\mathbf{x}_t - \mu_j\|^2$$

K-means Convergence

- K-means algorithm converges to local minima of objective

$$O(c; \mathbf{r}_1, \ldots, \mathbf{r}_K) = \sum_{j=1}^{K} \sum_{c(\mathbf{x}_t)=j} \|\mathbf{x}_t - \mathbf{r}_j\|_2^2$$

- Proof:
  Clustering assignment improves objective:

$$O\left(\hat{c}^{m-1}; \mathbf{r}_1^m, \ldots, \mathbf{r}_K^m\right) \geq O\left(\hat{c}^m; \mathbf{r}_1^m, \ldots, \mathbf{r}_K^m\right)$$

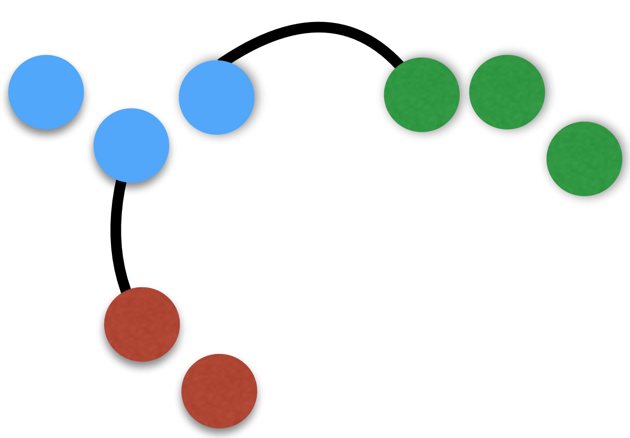
  (By definition of $\hat{c}^m(\mathbf{x}_t)$)
  Computing centroids improves objective:

$$O\left(\hat{c}^m; \mathbf{r}_1^m, \ldots, \mathbf{r}_K^m\right) \geq O\left(\hat{c}^m; \mathbf{r}_1^{m+1}, \ldots, \mathbf{r}_K^{m+1}\right)$$

  (By the fact about centroid)

Lets build an Algorithm

$$M_3 = \min_{\mathbf{x}_s, \mathbf{x}_t : c(\mathbf{x}_s) \neq c(\mathbf{x}_t)} \text{dissimilarity}(x_t, x_s)$$



Single Link Clustering

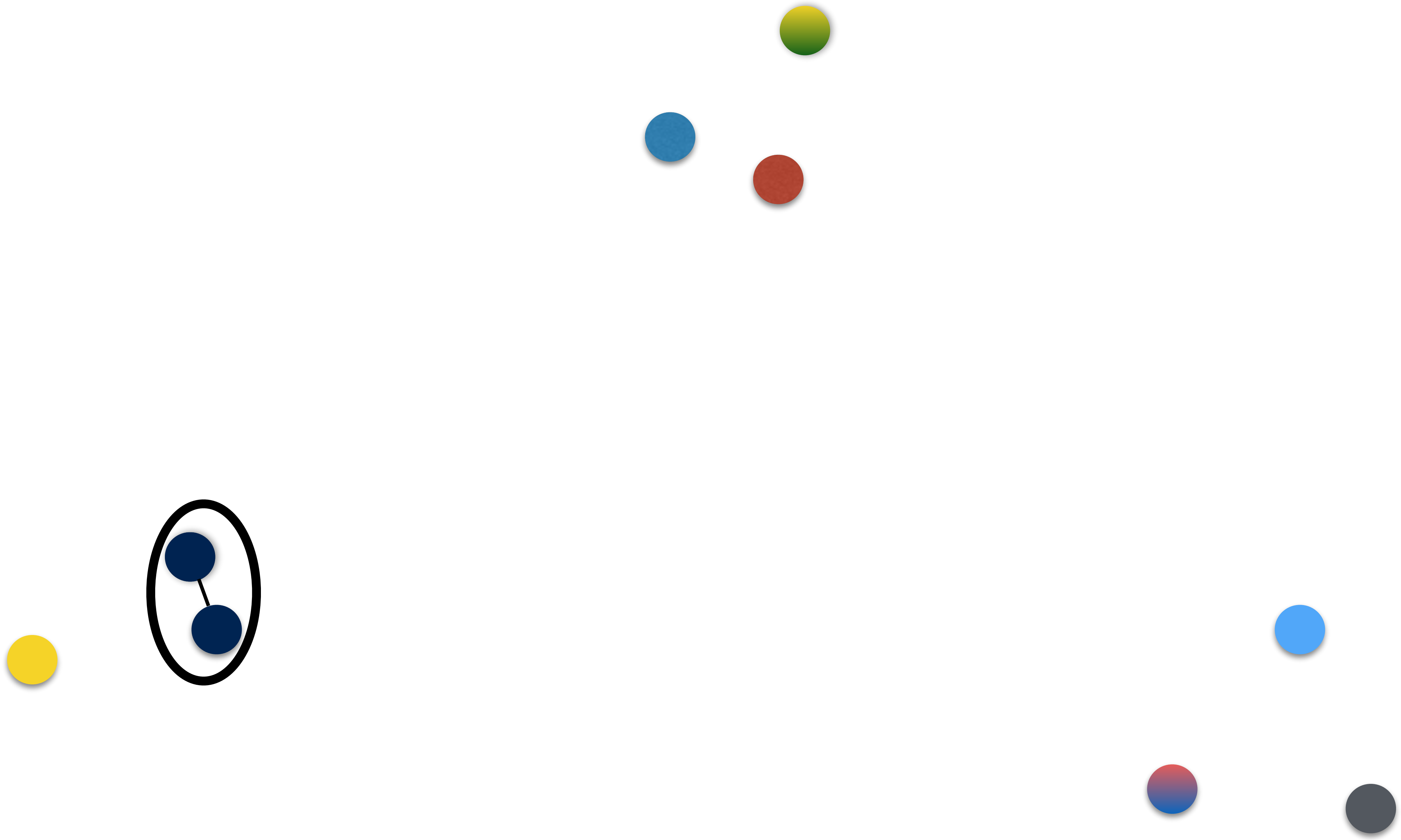
- Initialize n clusters with each point $\mathbf{x}_t$ to its own cluster
- Until there are only K clusters, do
  1. Find closest two clusters and merge them into one cluster

$$\text{dissimilarity}(C_i, C_j) = \min_{t \in C_i, s \in C_j} \text{dissimilarity}(\mathrm{x}_t, \mathrm{x}_s)$$

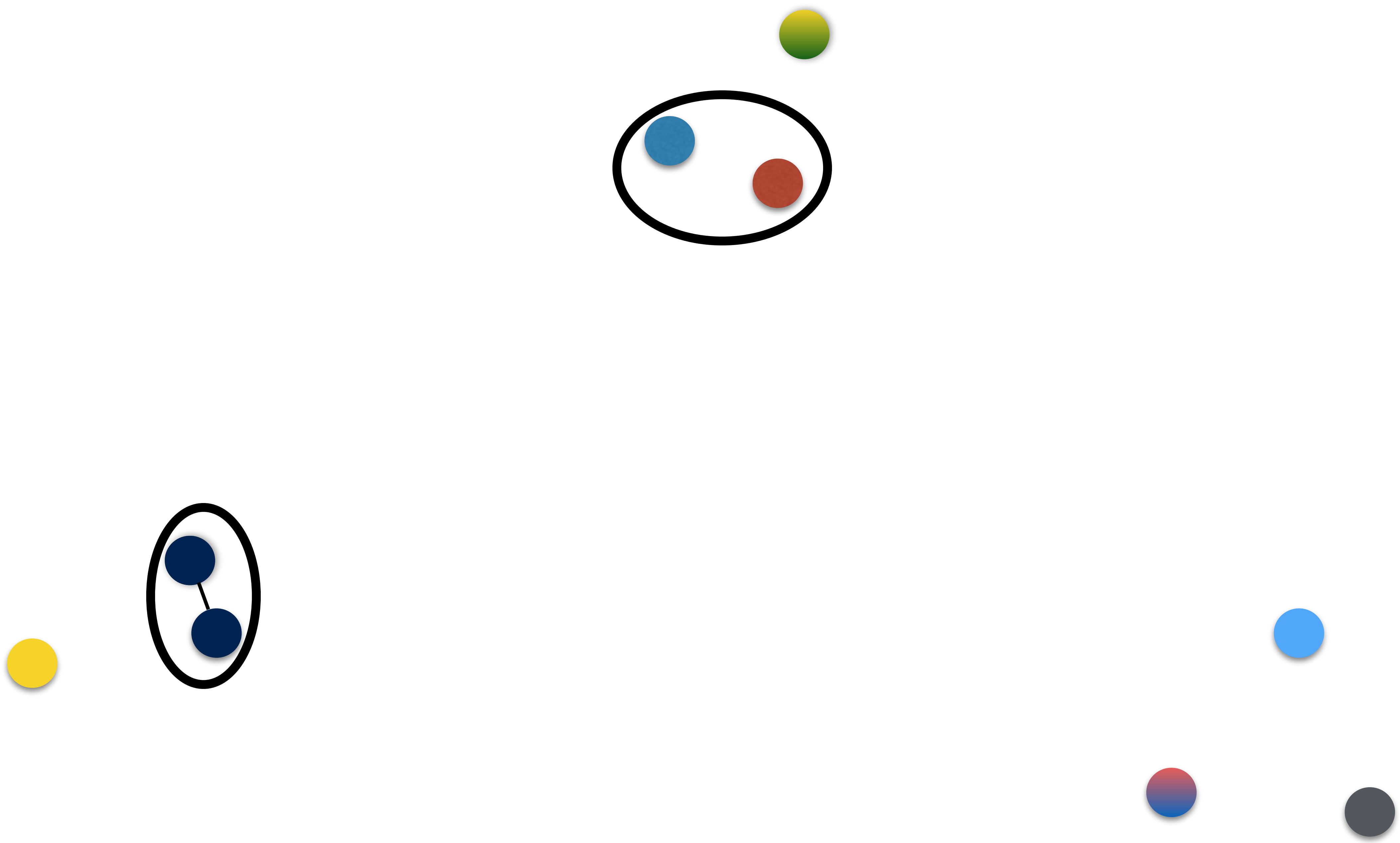
Demo

Demo

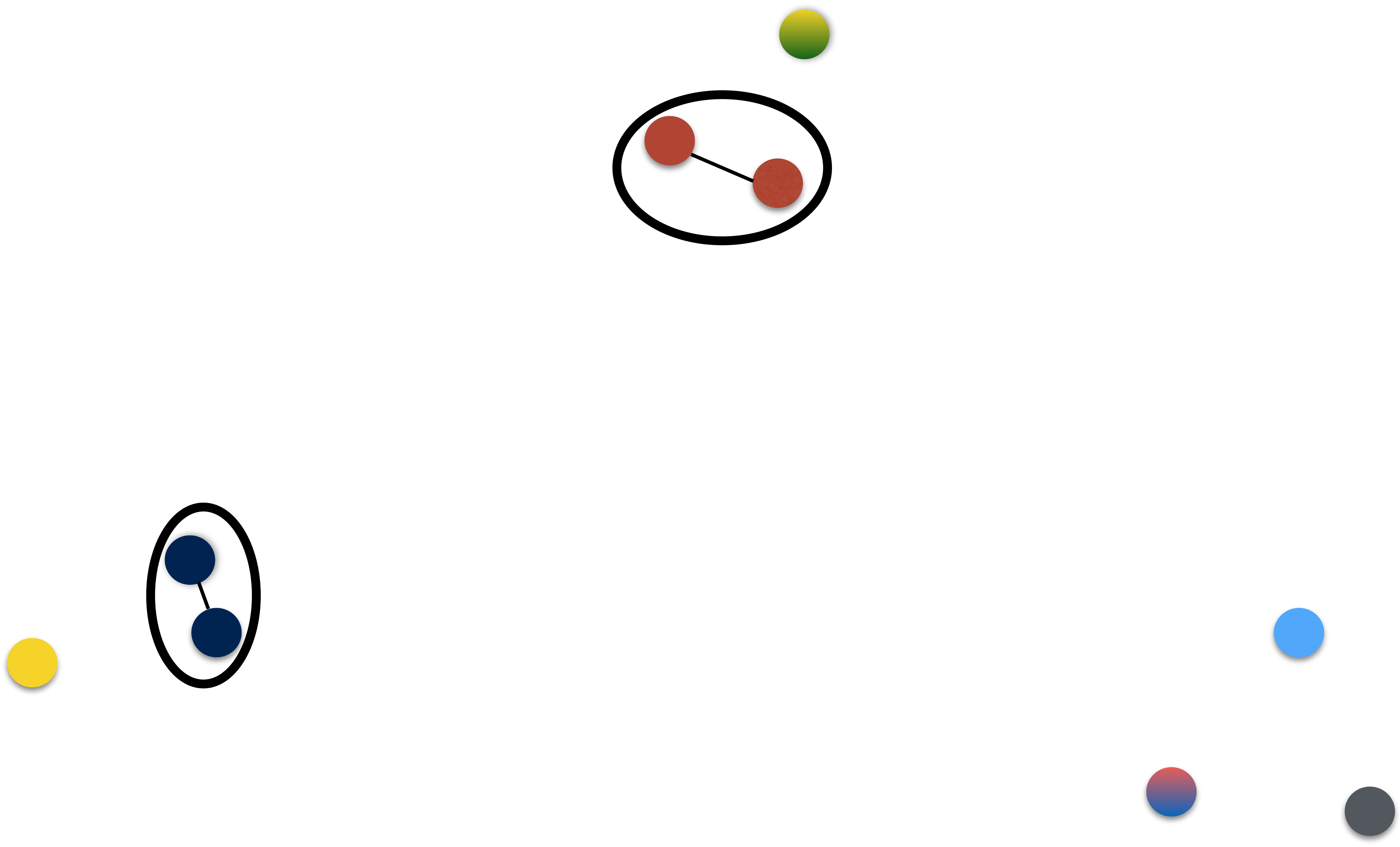
Demo



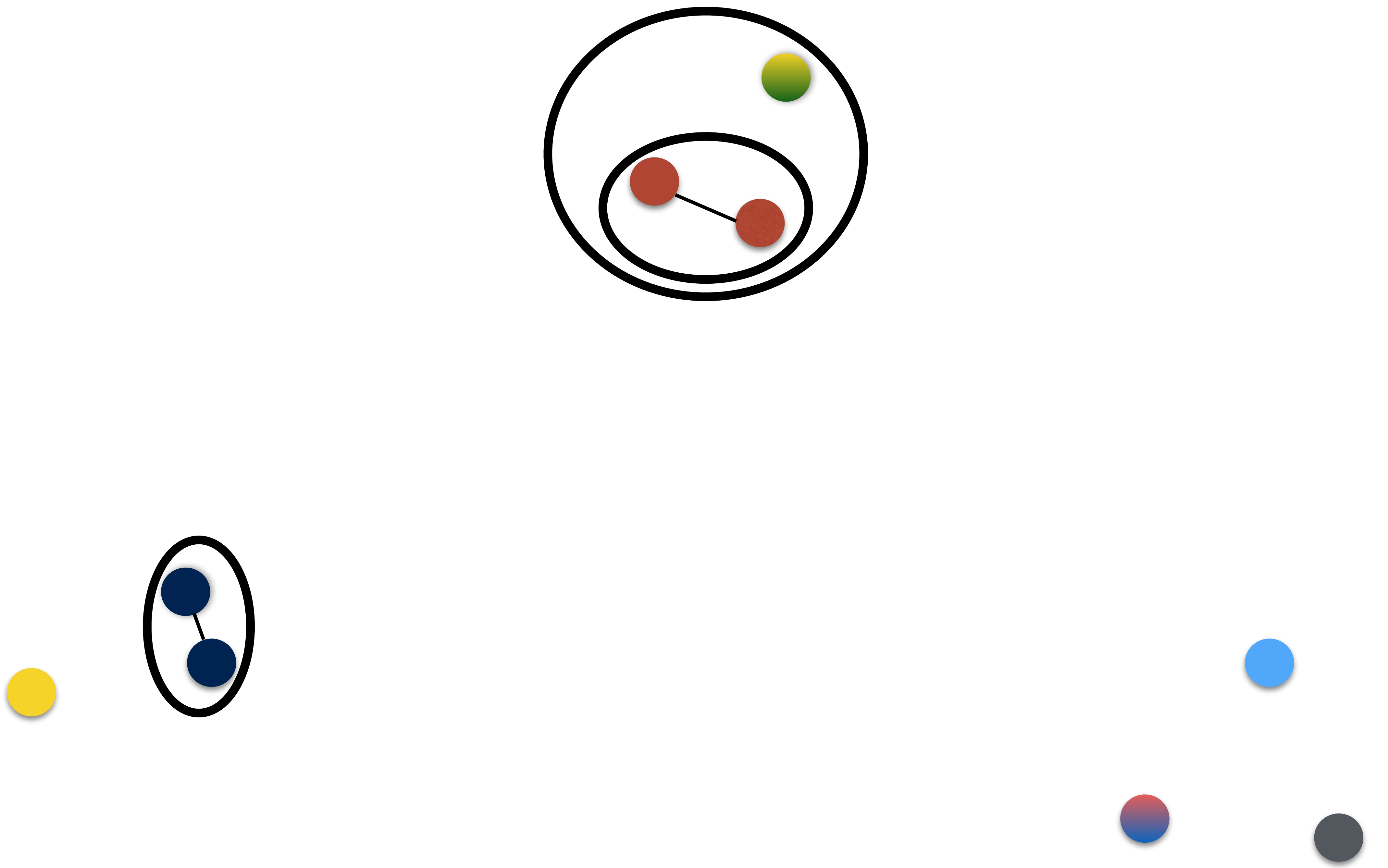
Demo



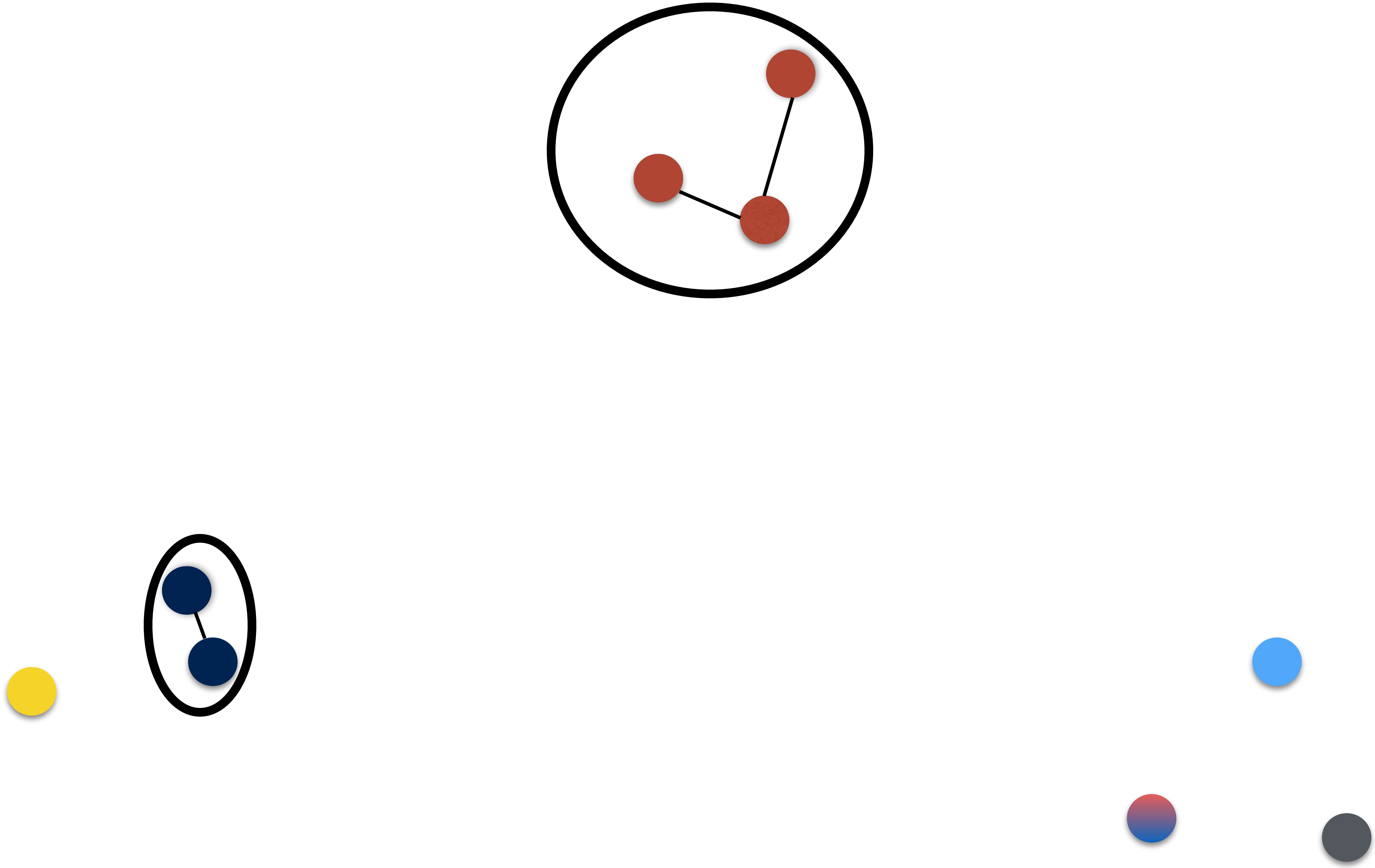
Demo



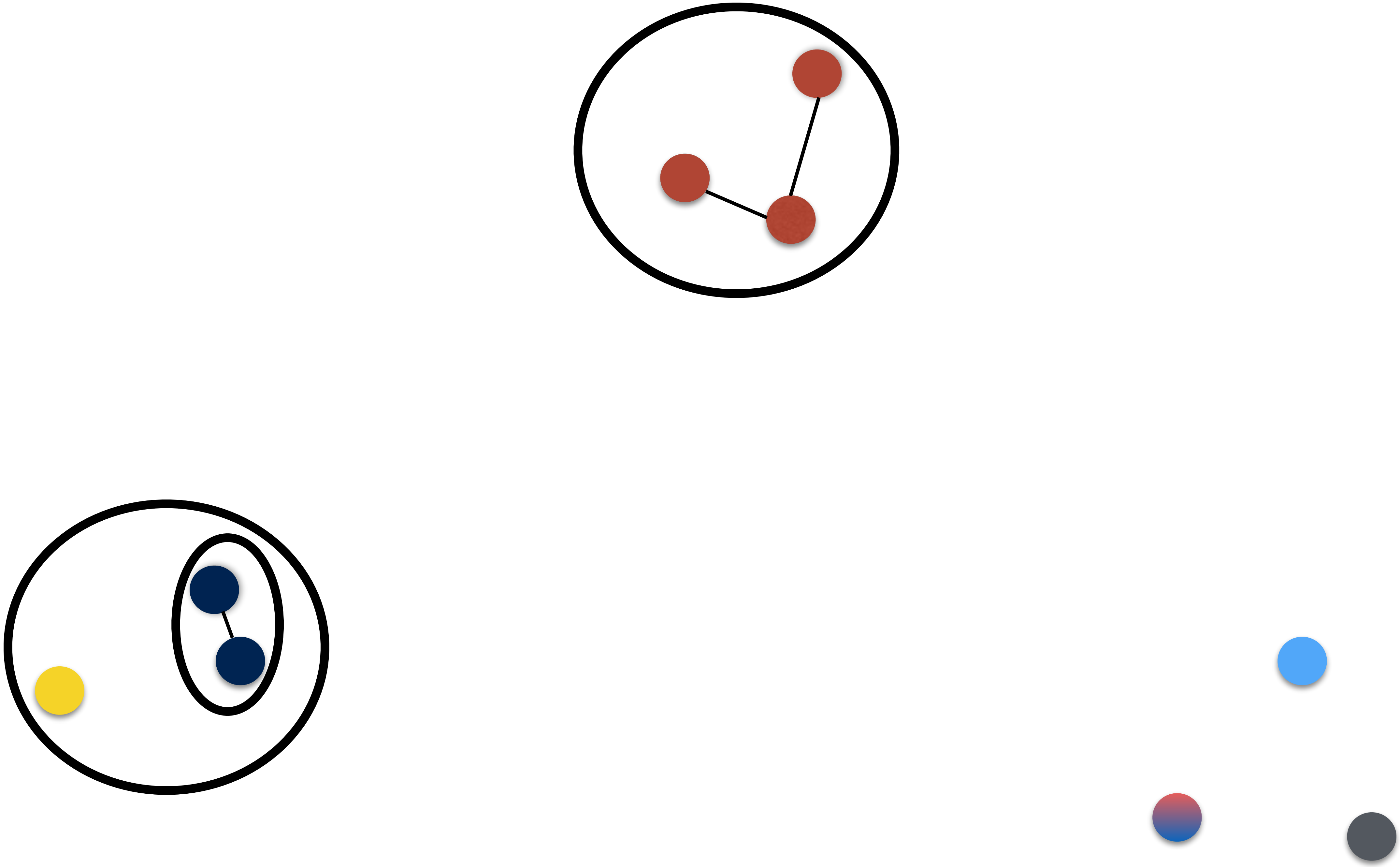
Demo



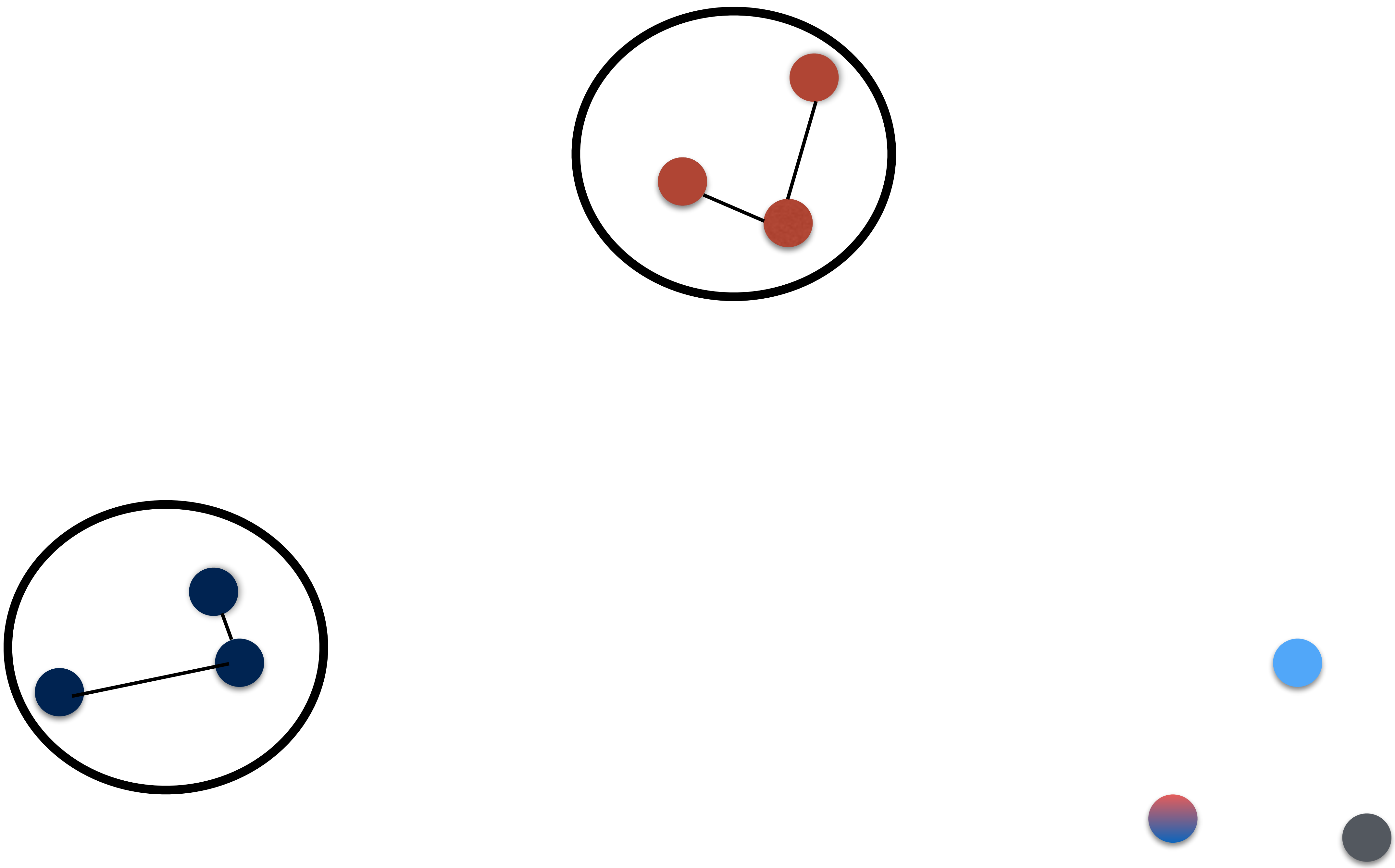
Demo



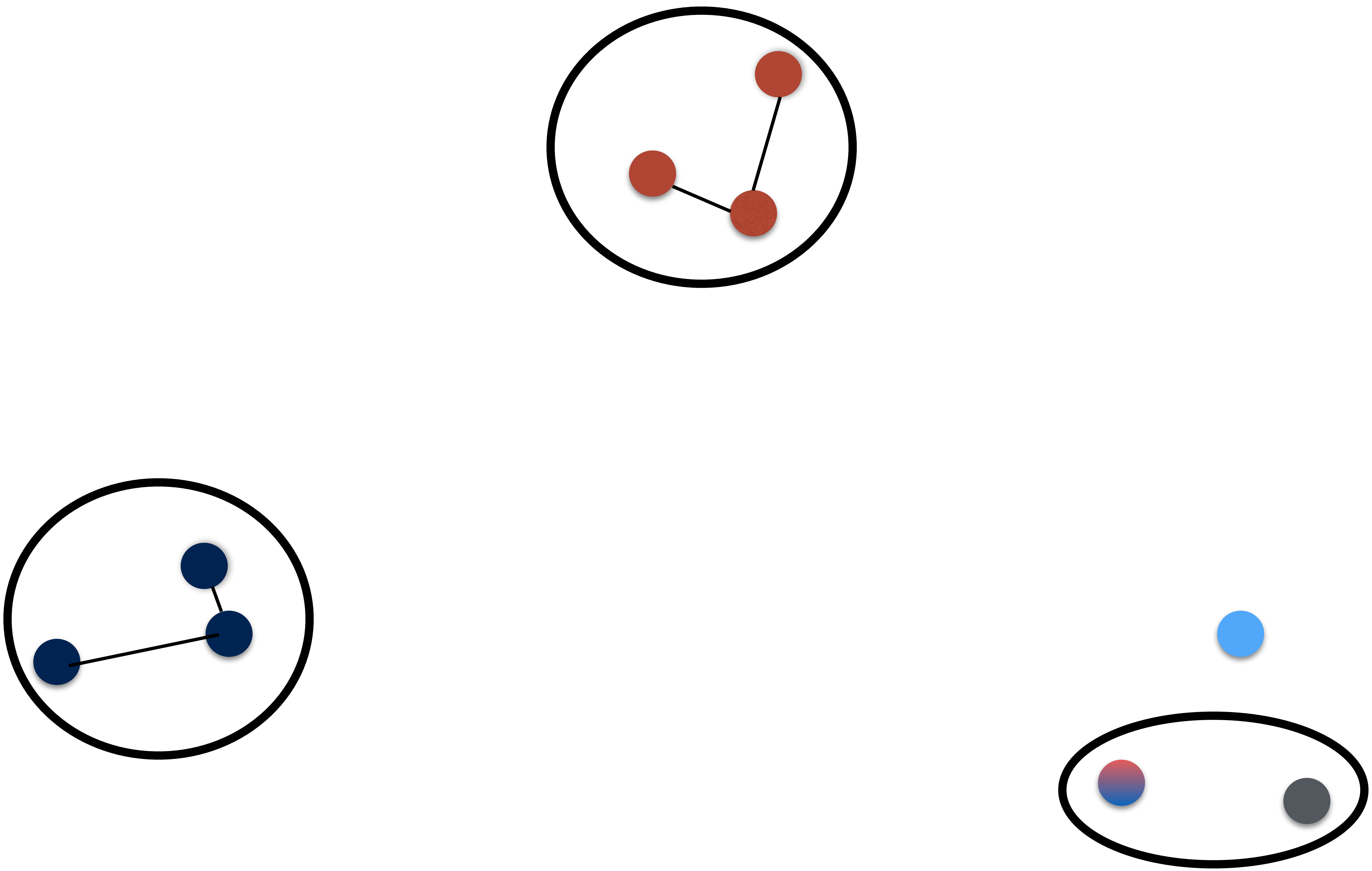
Demo



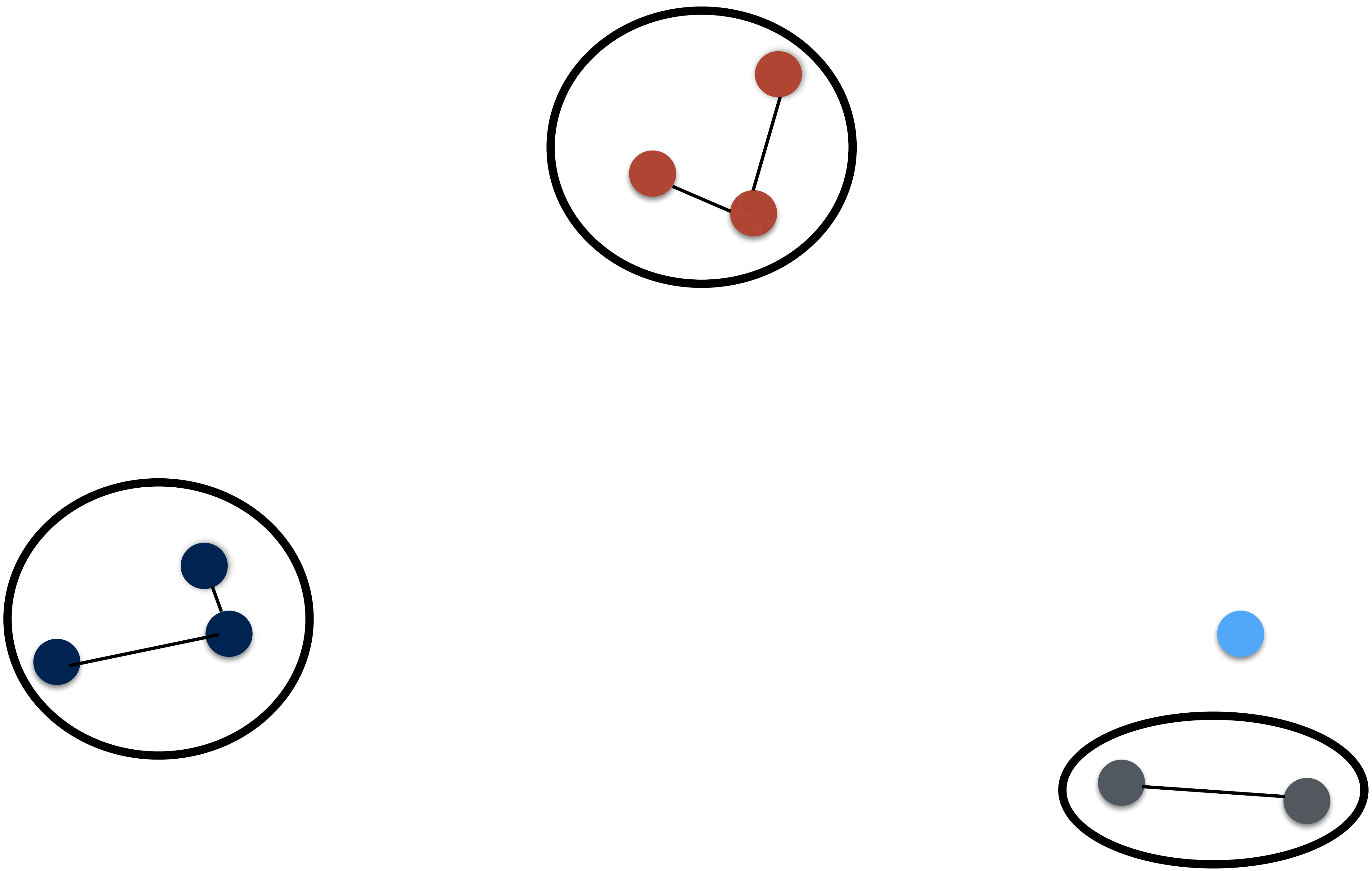
Demo



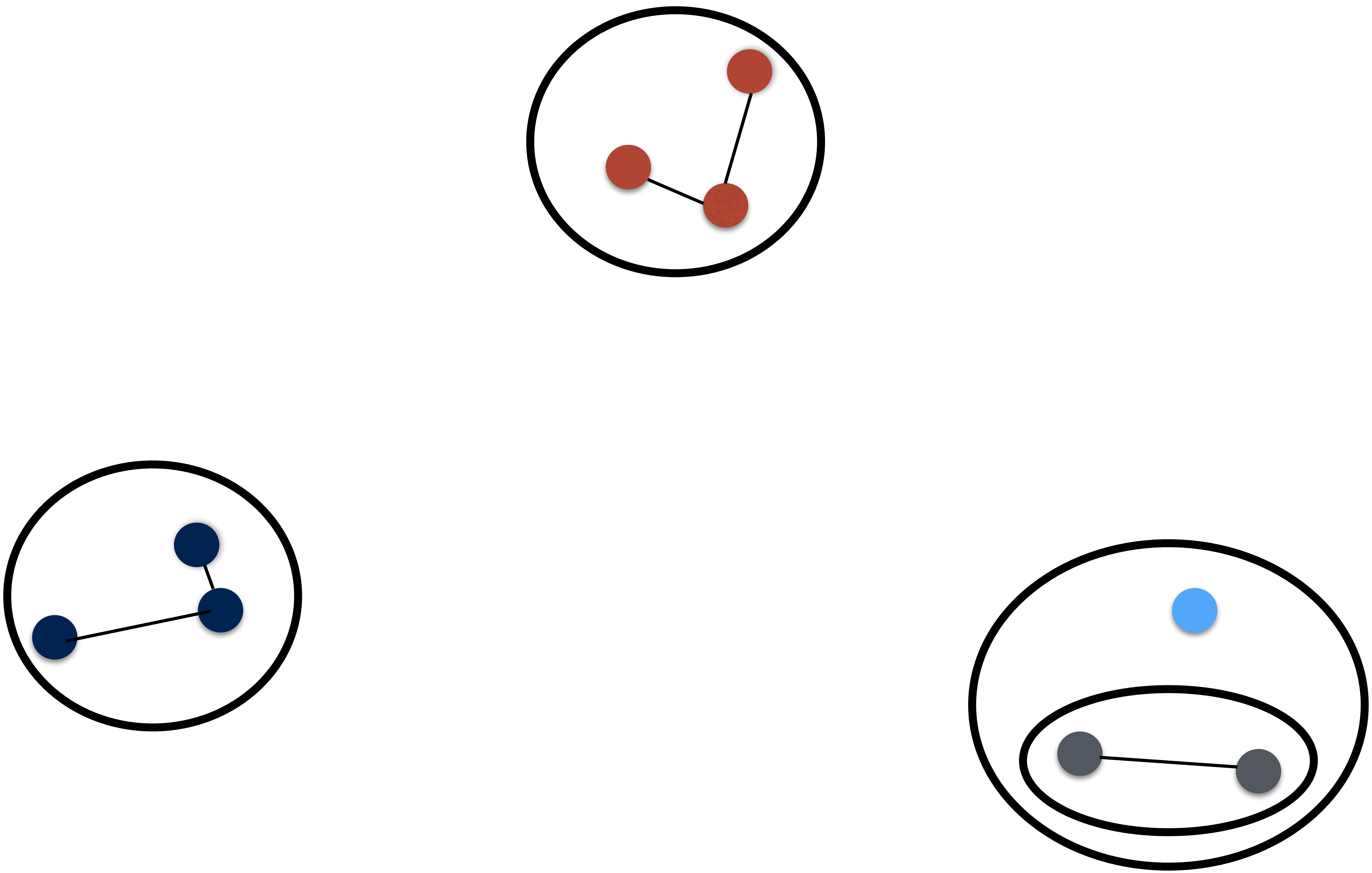
Demo



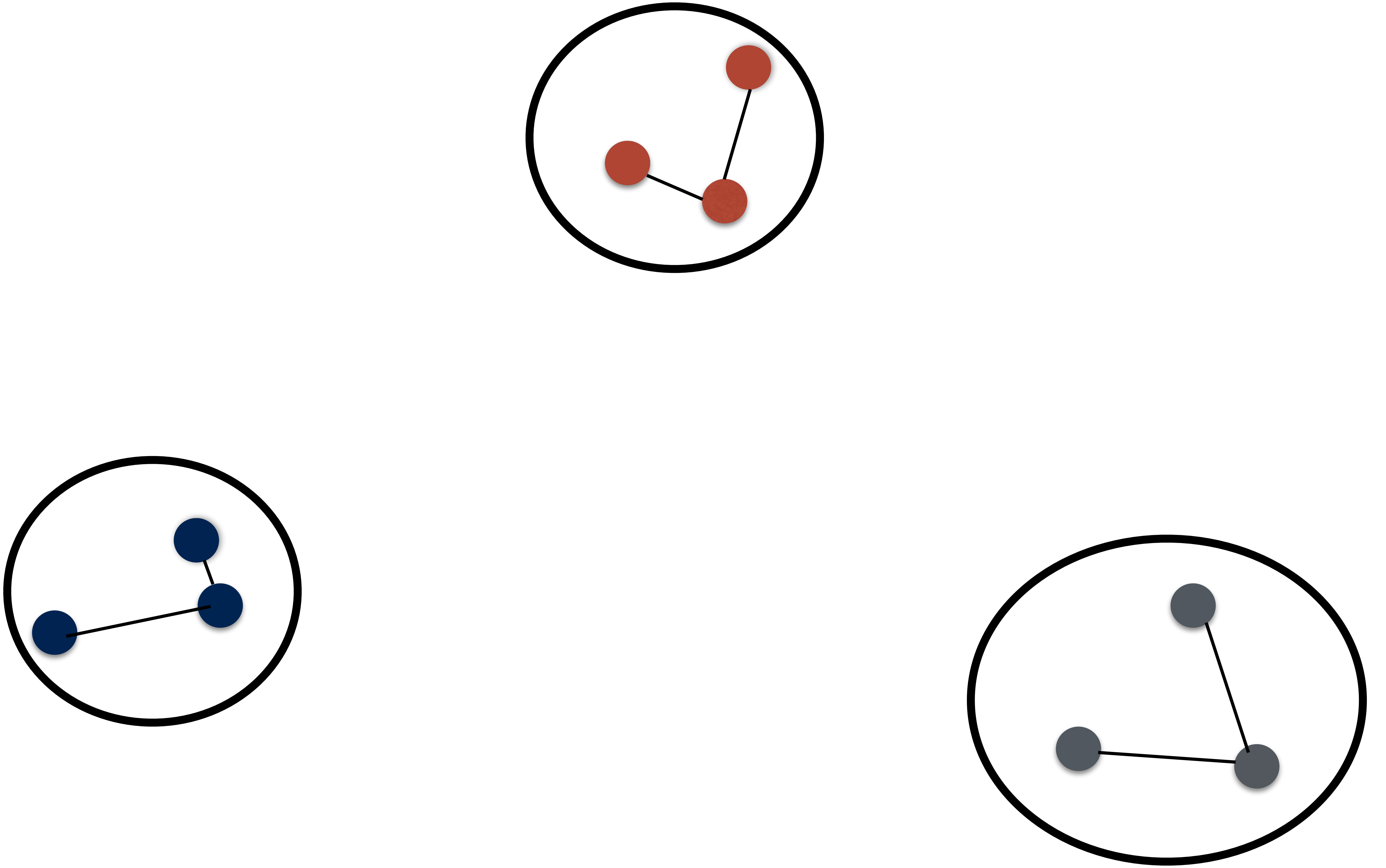
Demo



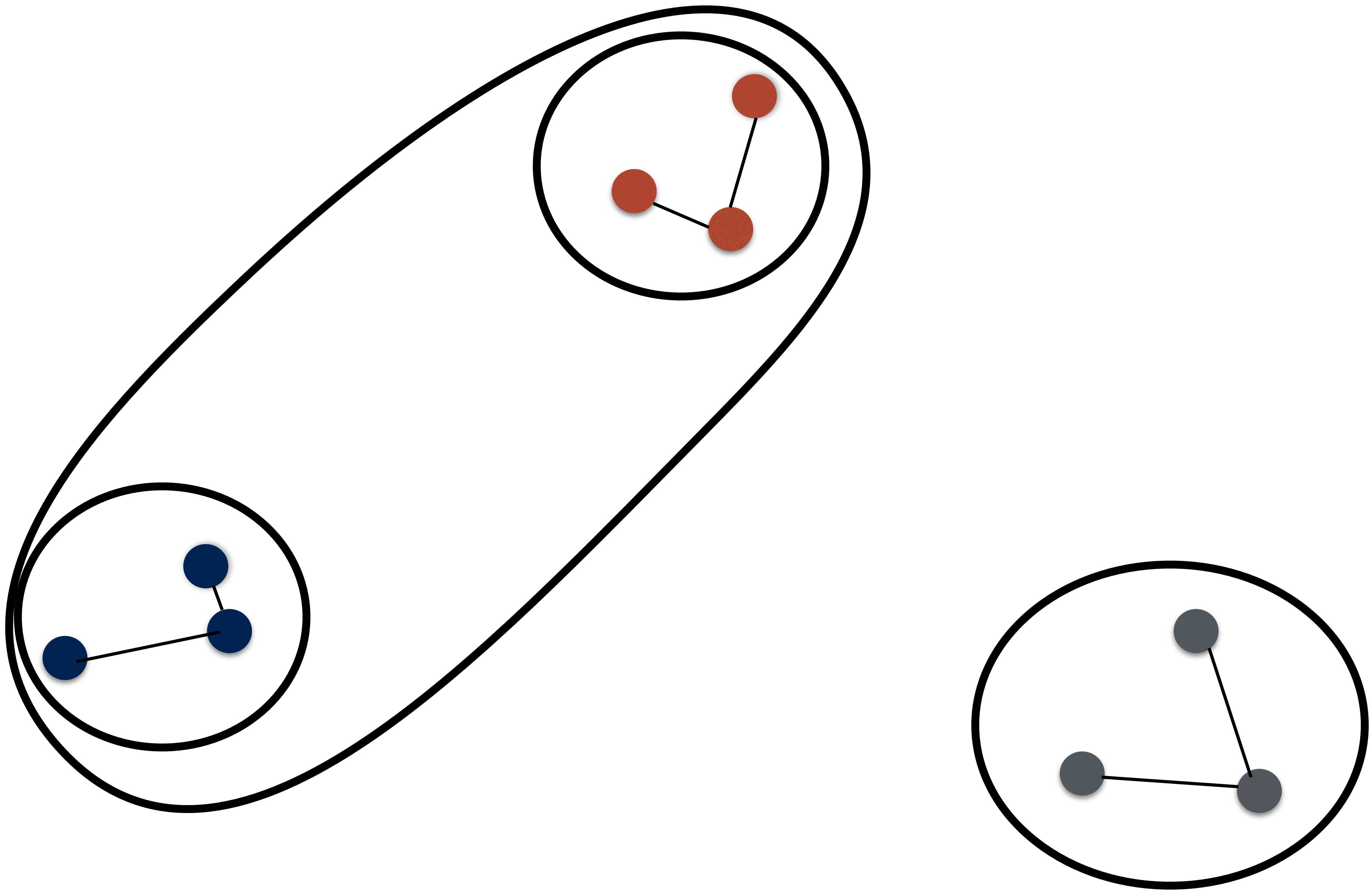
Demo



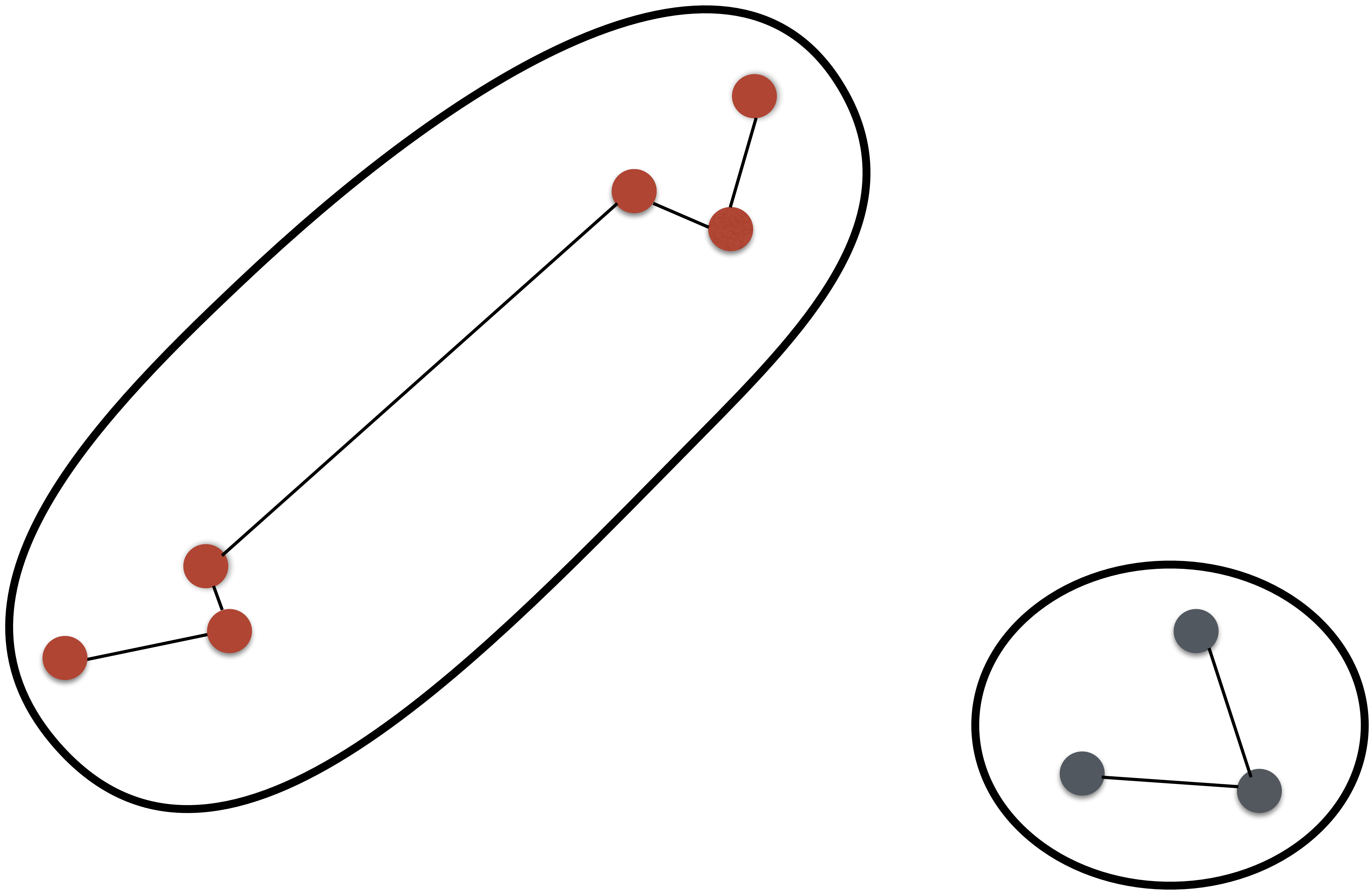
Demo



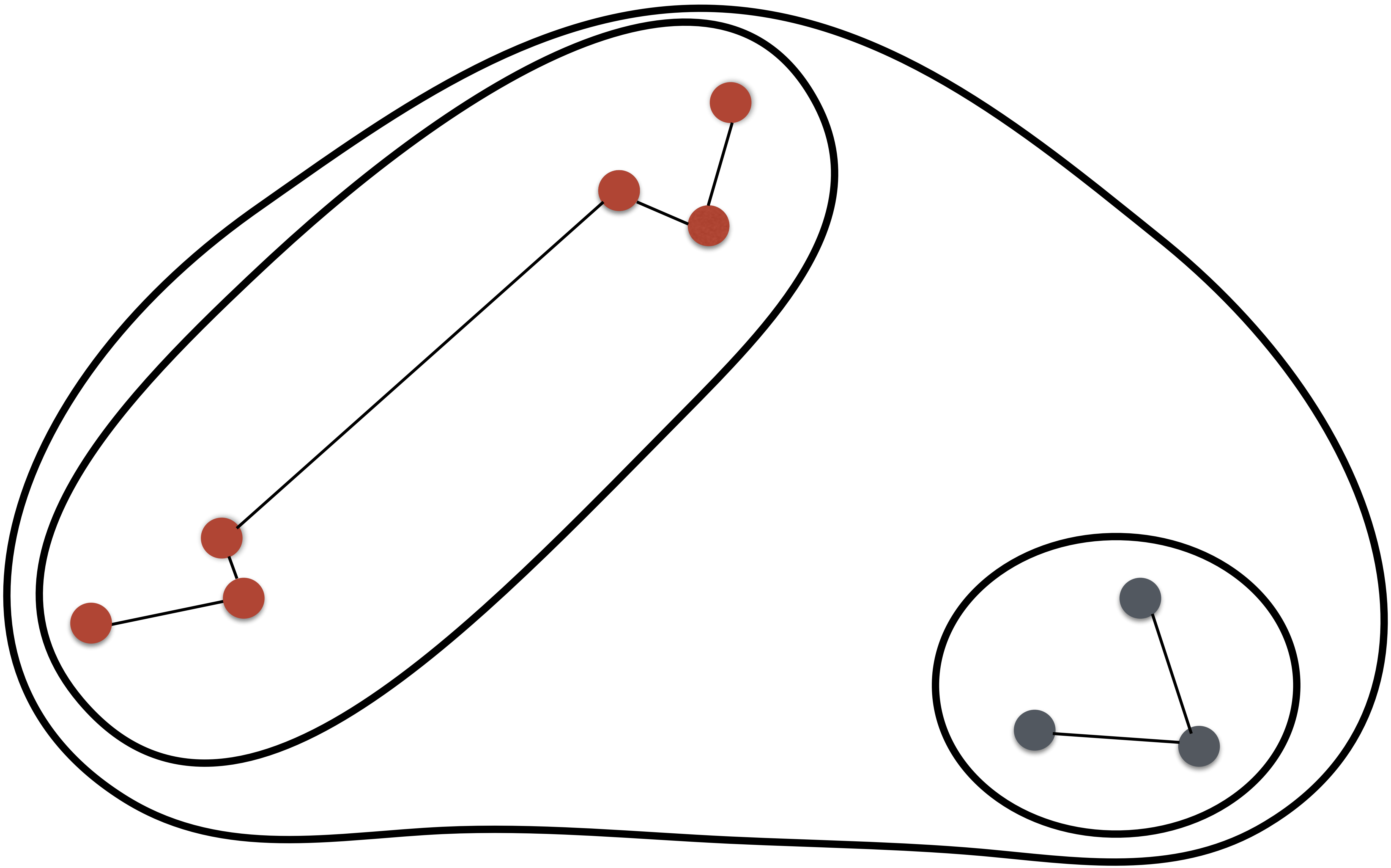
Demo



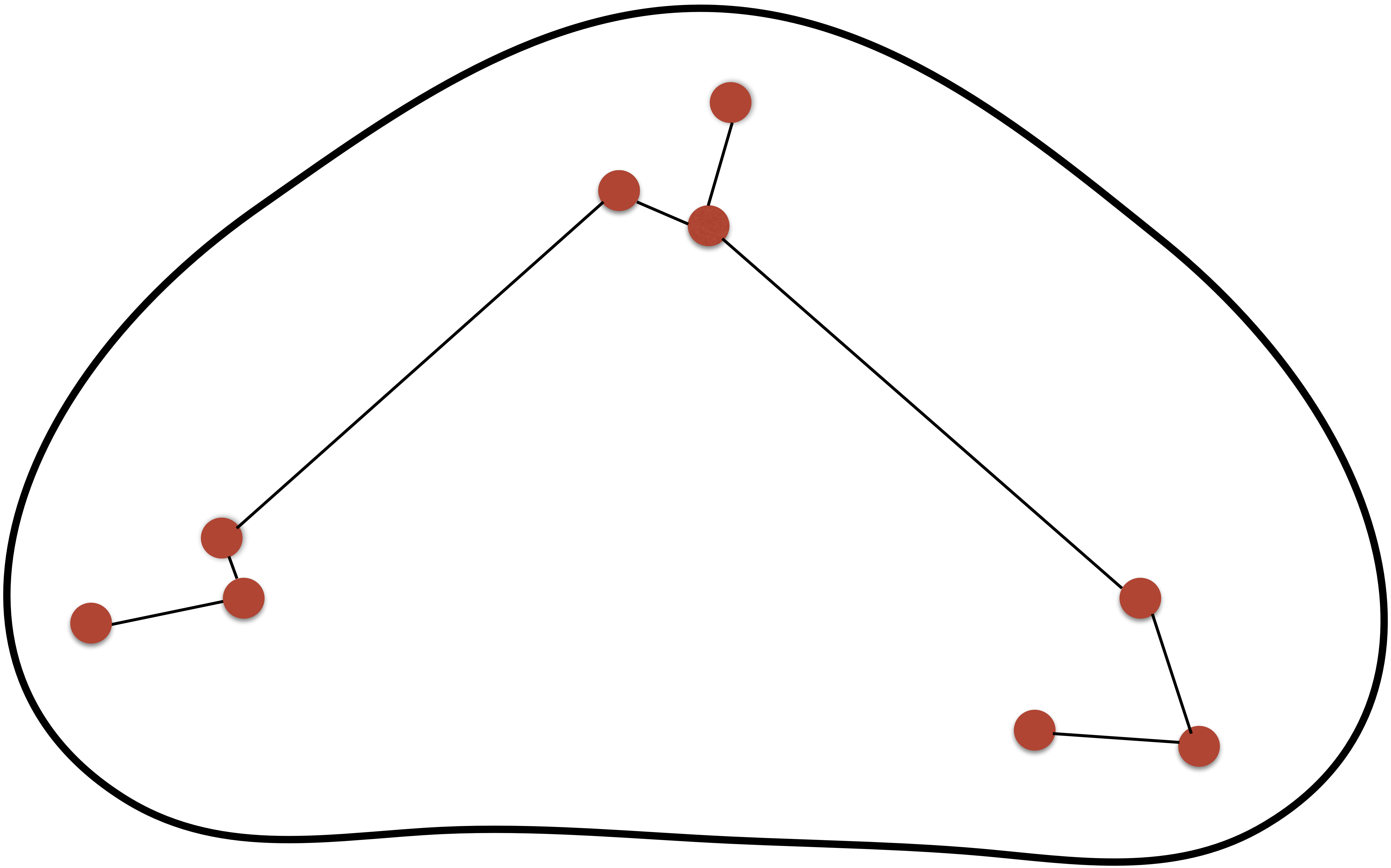
Demo



Demo



Demo



Single Link Objective

Objective for single-link:

$$M_3 = \min_{\mathbf{x}_s, \mathbf{x}_t : c(\mathbf{x}_s) \neq c(\mathbf{x}_t)} \text{dissimilarity}(x_t, x_s)$$

Single link clustering is optimal for above objective!

SINGLE LINK OBJECTIVE

Proof:

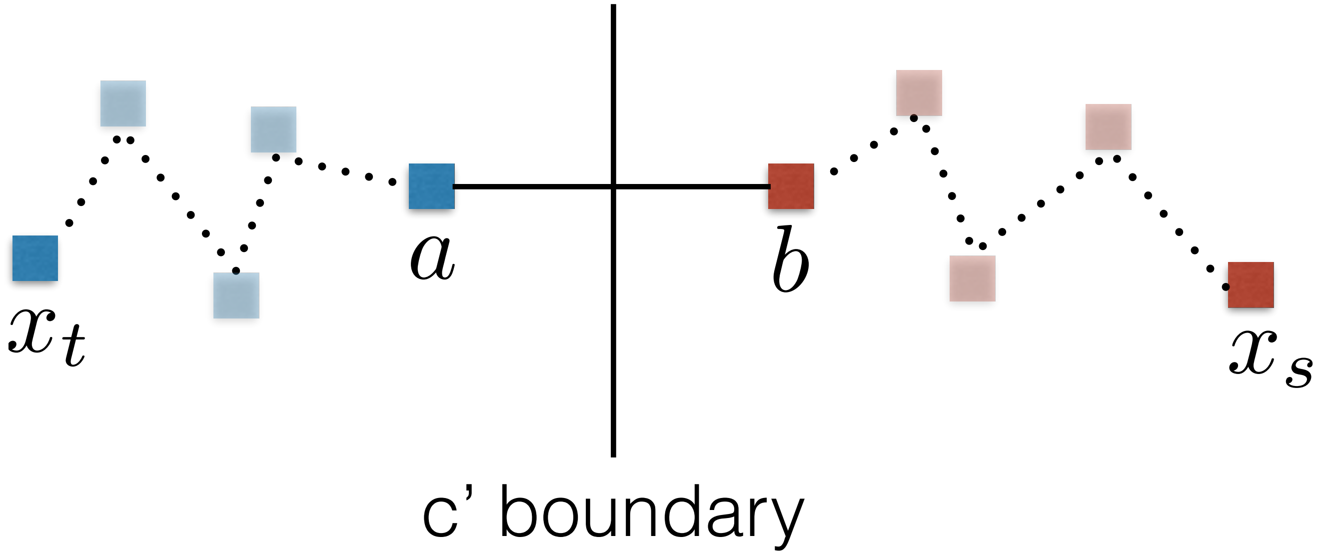
Say c is solution produced by single-link clustering

Key observation:

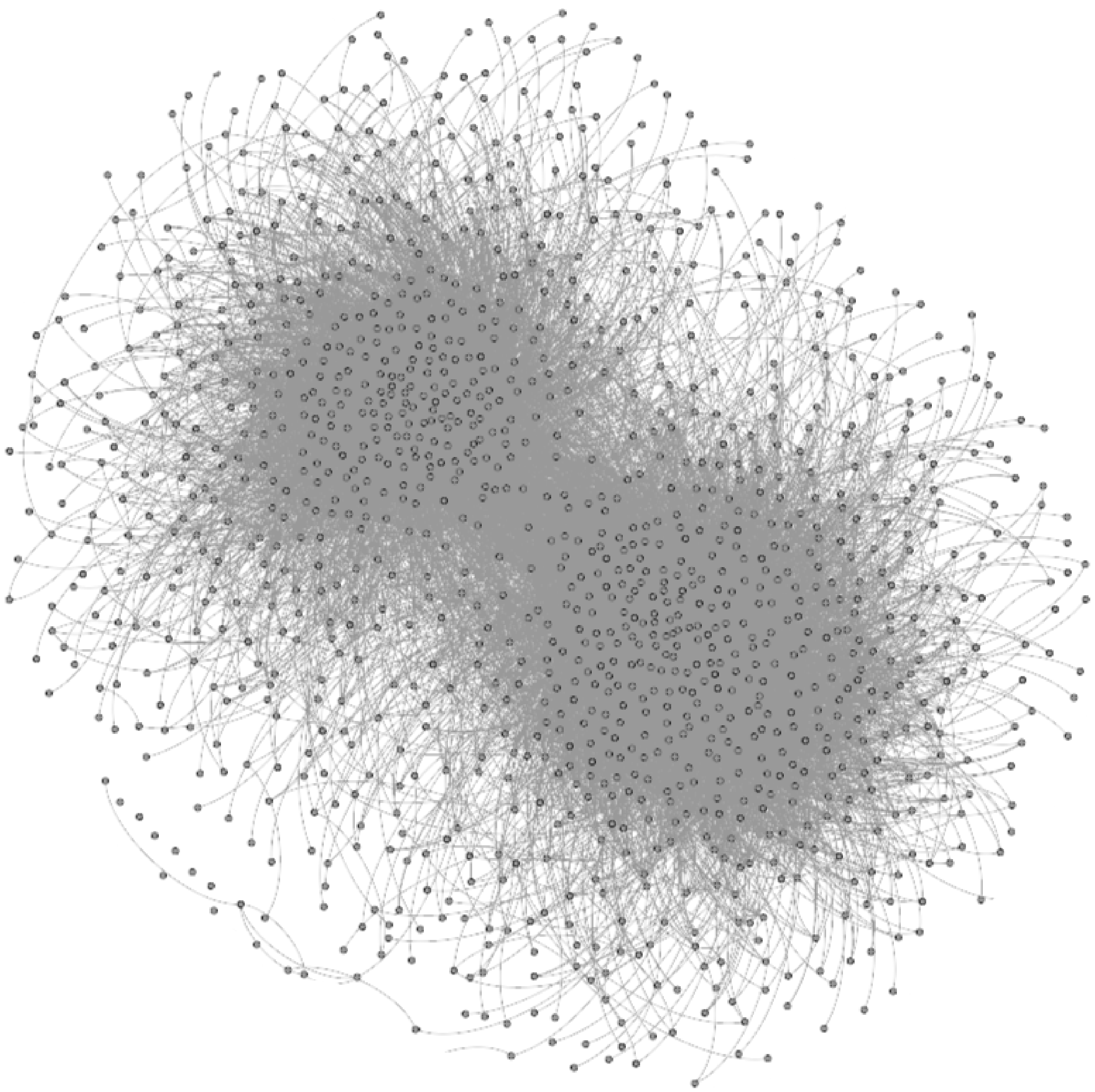
$$\min_{t,s:c(x_i)\neq c(x_j)} \text{dissimilarity}(x_i, x_j) >$$ Distance of points merged (on the tree)

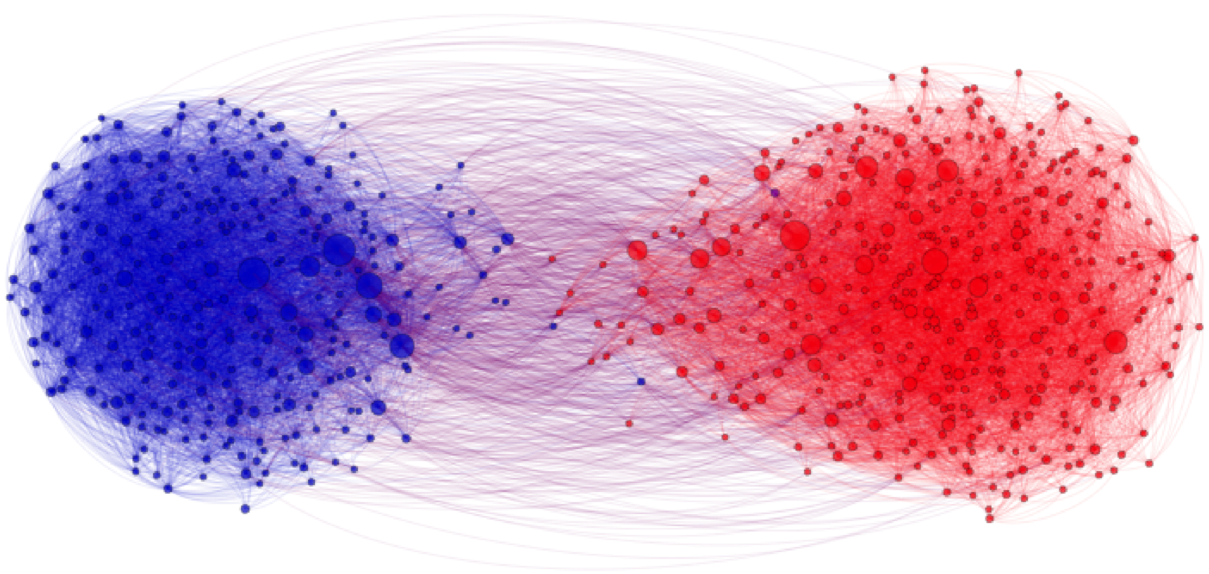
Say $c' \neq c$ then,

$$\exists\ t, s \text{ s.t. } c'(x_t) \neq c'(x_s) \text{ but } c(x_t) = c(x_s)$$



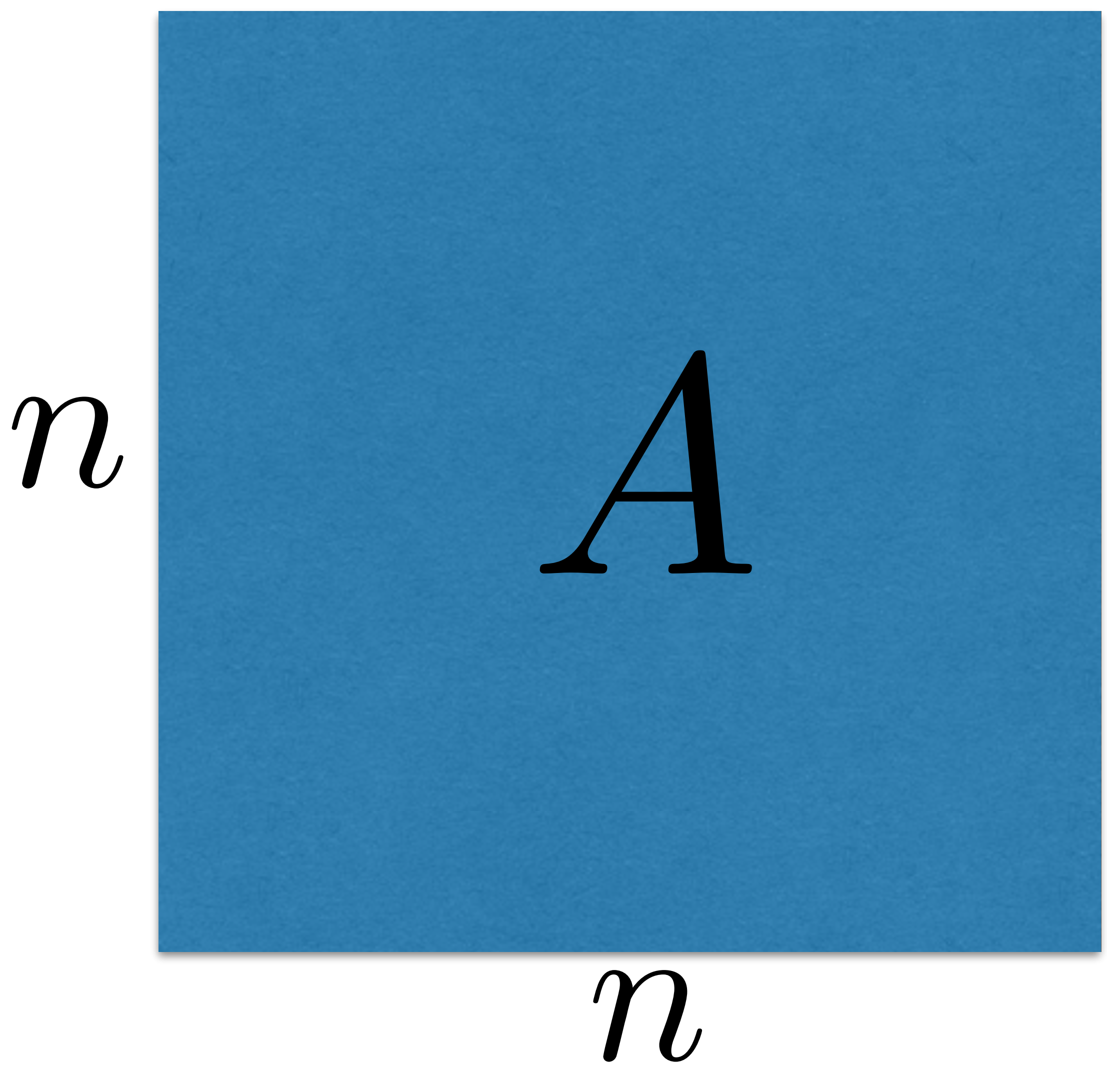
SPECTRAL CLUSTERING





- Cluster nodes in a graph.
- Analysis of social network data.

SPECTRAL CLUSTERING



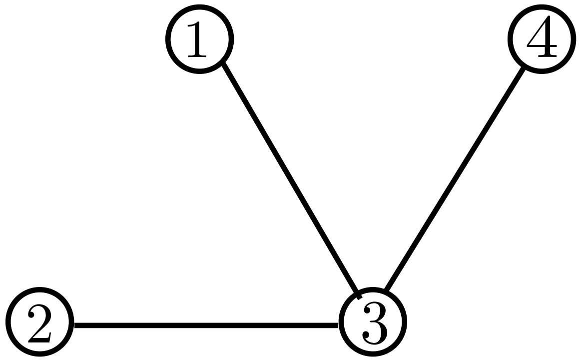
A is adjacency matrix of a graph

Spectral Clustering Algorithm (Unnormalized)

1. Given matrix A calculate diagonal matrix D s.t. $D_{i,i} = \sum_{j=1}^{n} A_{i,j}$
2. Calculate the Laplacian matrix $L = D - A$
3. Find eigen vectors $\mathbf{v}_1, \ldots, \mathbf{v}_n$ of L (ascending order of eigenvalues)
4. Pick the K eigenvectors with smallest eigenvalues to get $\mathbf{y}_1, \ldots, \mathbf{y}_n \in \mathbb{R}^K$
5. Use K-means clustering algorithm on $\mathbf{y}_1, \ldots, \mathbf{y}_n$

Example

Graph Clustering



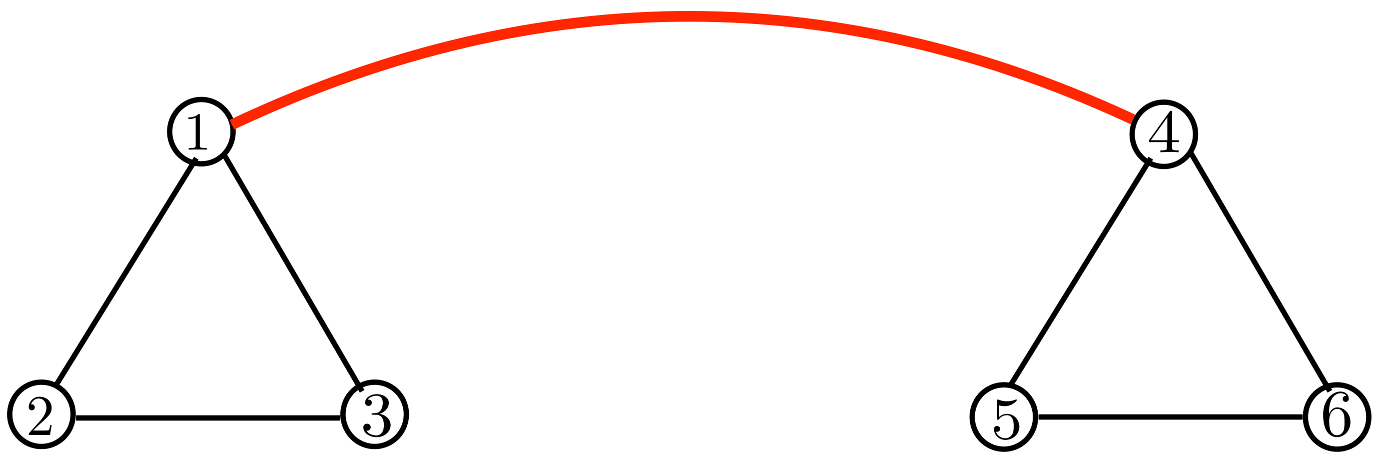
- Fact: For a connected graph, exactly one, the smallest of eigenvalues is 0, corresponding eigenvector is $\mathbf{1} = (1, \ldots, 1)^\top$

  Proof: Sum of each row of L is 0 because $D_{i,i} = \sum_{j=1}^{n} A_{i,j}$ and $L = D - A$

- Fact: For general graph, number of 0 eigenvalues correspond to number of connected components. The corresponding eigenvectors are all 1's on the nodes of connected components

  Proof: L is block diagonal. Use connected graph result on each component.

Graph Clustering



- Fact: For general graph, number of 0 eigenvalues correspond to number of connected components. The corresponding eigenvectors are all 1's on the nodes of connected components

  Proof: L is block diagonal. Use connected graph result on each component.

Graph Clustering: Cuts

- Partition nodes so that as few edges are cut (Mincut)
- What has this got to do with the Laplacian matrix?

Let $c_k \in \mathbb{R}^{|V|}$ be s.t. each coordinate indicates if the corresponding node belongs to cluster k

$$\text{cut}(c) = \sum_{j=1}^{K} c_k^\top L c_k$$

$$
\begin{aligned}
\mathrm{Cut}(c) &= \frac{1}{2}\sum_{k=1}^{K}\sum_{(i,j)\in E}\left(c_k[j]-c_k[i]\right)^2 \\
&= \frac{1}{2}\sum_{k=1}^{K}\sum_{(i,j)\in E}\big(c_k[j]^2+c_k[i]^2-2c_k[i]c_k[j]\big) \\
&= \frac{1}{2}\sum_{k=1}^{K}\sum_{i\in V}\sum_{j\in V}\big(A_{i,j}c_k[j]^2+A_{i,j}c_k[i]^2-2A_{i,j}c_k[i]c_k[j]\big) \\
&= \frac{1}{2}\sum_{k=1}^{K}\sum_{j\in V}\left(\sum_{i\in V}A_{i,j}\right)c_k[j]^2+\frac{1}{2}\sum_{i\in V}\left(\sum_{j\in V}A_{i,j}\right)c_k[i]^2-\sum_{i\in V}\sum_{j\in V}A_{i,j}c_k[i]c_k[j] \\
&= \frac{1}{2}\sum_{k=1}^{K}\sum_{j\in V}\left(\sum_{i\in V}A_{i,j}\right)c_k[j]^2+\frac{1}{2}\sum_{k=1}^{K}\sum_{i\in V}\left(\sum_{j\in V}A_{i,j}\right)c_k[i]^2-\sum_{k=1}^{K}\sum_{i\in V}\sum_{j\in V}A_{i,j}c_k[i]c_k[j] \\
&= \sum_{k=1}^{K}\sum_{i\in V}D_{i,i}c_k[i]^2-\sum_{k=1}^{K}\sum_{i\in V}\sum_{j\in V}A_{i,j}c_k[i]c_k[j] \\
&= \sum_{k=1}^{K}c_k^\top Dc_k-\sum_{k=1}^{K}c_k^\top Ac_k=\sum_{k=1}^{K}c_k^\top Lc_k
\end{aligned}
$$

Spectral Clustering Algorithm (Unnormalized)

Find Clustering c to minimize

$$\sum_{j=1}^{K} c_k^\top L c_k$$