K-means algorithm

select K points \((m_1, \ldots, m_K)\) randomly

do

\((w_1, \ldots, w_K) = (m_1, \ldots, m_K)\)

all clusters \(C_i = \{\}\)

for each row \(w\) in \(M\)

find the closest point in \((w_1, \ldots, w_K)\) to \(w\)

assign \(w\) to the corresponding cluster:

\[C_i = C_i \cup \{w\}\] (if \(w_i\) is closest point)

end

for each cluster \(C_i\)

   calculate the mean point \(\bar{m}_i\)

while exists \(m_i \neq w_i\)
K-means clustering

• Input: $M$ (set of points), $K$ (number of clusters) $m_1, \ldots, m_k$ (Initial centroids)

• Choosing $K$
  - Study the data
  - Measure how squared error decreases as more clusters are added

• Choosing centroids
  - Typically randomly
K-means clustering

• Pros:
  - Easy
  - Scalable

• Cons:
  - Works only for certain clusters
  - Sensitive to outliers and noise
K-means clustering
K-means clustering

Bad initial points
K-means clustering
Non-spherical clusters
Questions

• Using the euclidean distance one gets spherical clusters, what types of clusters does one get using the manhattan distance?

• If we assume that the K-means algorithm converges in I iterations, with N points and X characteristics for each point give an approximation of the complexity of the algorithm expressed in K, I, N and X

• Can the K-means algorithm be parallellized? if yes how?
I want to cluster this class into 5 different clusters. Assume that I know:

- Your Age
- What row you are sitting in
- Whether you handed in the first assignment on time or not
- How many years you have studied at university

Design a method to use K-means to create these clusters
DB Scan

- Density based clustering
- Connected regions with sufficiently high density
- Clusters with arbitrary shape
- Avoids outliers, noise
DB Scan
- key concepts

- **ε-neighbourhood**
  - the neighbourhood within a radius $\varepsilon$ of an object

- **core object**
  - an object is a core object iff there are more than $\text{MinPts}$ objects in its $\varepsilon$-neighbourhood

- **directly density reachable (ddr)**
  - An object $p$ is ddr from $q$ iff $q$ is a core object and $p$ is inside the $\varepsilon$-neighbourhood of $q
DB Scan
- key concepts

• density reachable (dr)
  - an object \( q \) is \( \text{dr} \) from \( p \) iff there exists a chain of objects \( p_1, \ldots, p_n \) such that \( p_1 \) is \( \text{ddr} \) from \( p \), \( p_2 \) is \( \text{ddr} \) from \( p_1 \), \( p_3 \) is \( \text{ddr} \) from ... and \( q \) is \( \text{ddr} \) from \( p_n \).

• density connected (dc)
  - \( p \) is \( \text{dc} \) to \( q \) iff exist an object \( o \) such that \( p \) is \( \text{dr} \) from \( o \) and \( q \) is \( \text{dr} \) from \( o \)
DB Scan
- How to use DB scan to cluster

- Idea:
  - If object $p$ is density connected to $q$, then $p$ and $q$ should belong to the same cluster
  - If an object is not density connected to any other object it is considered as noise
DB Scan
- How to use DB scan to cluster

• Naïve Algorithm:

\[
i = 0
\]

\[
do
\]

\[
take a point p from M
\]

\[
find the set of points P which are density connected to p
\]

\[
if P = {}
\]

\[
M = M / \{p\}
\]

\[
else C_i = P, i = i+1, M = M / P
\]

\[
end
\]

\[
while M \neq {}
\]
More practical Algorithm:

\[ i = 0, \text{Find the core points CP in M} \]

do

\[ \text{take a point } p \text{ from CP} \]

\[ \text{find the set of points } P \text{ which are density} \]

\[ \text{reachable from } p \]

\[ C_i = P, \quad i = i+1, \quad \text{CP} = \text{CP} / (\text{CP} \cap P) \]

while \( \text{CP} \neq {} \)
DB Scan

- How to use DB scan to cluster

find the set of points $P$ which are density reachable from $p$

$C = \{p\}, P = \{p\}$

do

    Remove a point $p'$ from $C$
    Find all of the points $X$ that are directly density reachable from $p'$

    $C = C \cup (X \setminus (P \cap X))$

    $P = P \cup X$

while $C \neq {}$
Questions

• Why is the density connected criterion useful to define a cluster, instead of density reachable or directly density reachable?

• For which points are density reachable symmetric?

• Express using only core objects and directly density reachable, which objects will belong to a cluster.
Practical db scan

Try to use the db scan algorithm with the following parameters:

MinPts:
Eps:

To determine if you are a core point, if you belong to a cluster.