Clustering

77B Recommender Systems
What if not all rating are observed?

\[
PCC(\bar{x}, \bar{y}) = \frac{\sum_{i:x_i \& y_i \text{observed}} (x_i - m_x)(y_i - m_y)}{\sqrt{\sum_{i:x_i \& y_i \text{observed}} (x_i - m_x)^2} \sqrt{\sum_{i:x_i \& y_i \text{observed}} (y_i - m_y)^2}}
\]

\[
CSM(\bar{x}, \bar{y}) = \frac{\sum_{i=x_i \& y_i \text{observed}} x_iy_i}{\sqrt{\sum_{i:x_i \& y_i \text{observed}} x_i^2} \sqrt{\sum_{i:x_i \& y_i \text{observed}} y_i^2}}
\]
Correction to Book

\[ x_{iu} = m_u + \frac{\sum_{v=1}^{U} pcc(u,v) \cdot (y_{iv} - m_v)}{\sum_{v=1}^{U} pcc(u,v)} \quad \rightarrow \quad x_{iu} = m_u + \frac{\sum_{v=1}^{U} pcc(u,v) \cdot (y_{iv} - m_v)}{\sum_{v=1}^{U} |pcc(u,v)|} \]

\[ x_{iu} = \frac{\sum_{j=1}^{I} csm(i,j) \cdot y_{ju}}{\sum_{j=1}^{I} csm(i,j)} \quad \rightarrow \quad x_{iu} = \frac{\sum_{j=1}^{I} csm(i,j) \cdot y_{ju}}{\sum_{j=1}^{I} |csm(i,j)|} \]
K Means Clustering

Idea:
Cluster the items OR users
For new item-user pair
Assign it to one of the clusters
Predict by using the average rating cluster
Clustering: K-means

• We iterate two operations:
  1. Update the assignment of data-cases to clusters
  2. Update the location of the cluster.

• Denote $z_i = [1, 2, 3, \ldots, K]$ the assignment of data-case “i” to cluster “c”.

• Denote $\mu_c \in \mathbb{R}^d$ the position of cluster “c” in a d-dimensional space.

• Denote $x_i \in \mathbb{R}^d$ the location of data-case i

• Then iterate until convergence:

  1. For each data-case, compute distances to each cluster and the closest one:

       $$z_i = \arg \min_c \| x_i - \mu_c \|_2 \quad \forall i$$

  2. For each cluster location, compute the mean location of all data-cases assigned to it:

       $$\mu_c = \frac{1}{N_c} \sum_{i \in S_c} x_i \quad \forall c$$

Nr. of data-cases in cluster $c$  \hspace{2cm}  Set of data-cases assigned to cluster $c$
K-means

- Cost function: \[ C = \sum_{i=1}^{N} || x_i - \mu_{z_i} ||^2 \]
- Each step in k-means decreases this cost function.
- Often initialization is very important since there are very many local minima in C. Relatively good initialization: place cluster locations on K randomly chosen data-cases.
- How to choose K?
  Add complexity term: \[ C \rightarrow C + \frac{1}{2} \times [\# \text{parameters}] \times \log(N) \] and minimize also over K
K-medians

Cost function:
\[ C = \sum_{i=1}^{N} \| \mathbf{x}_i - \bar{\mu}_{z_i} \|_1 = \sum_{i=1}^{N} \sum_{d=1}^{D} | x_{d,i} - \mu_{d,z_i} | \]

Algorithm: iterate until convergence:

1. For each data-case, compute distances to each cluster and the closest one:
   \[ z_i = \arg\min_{c} \| \mathbf{x}_i - \mu_c \|_1 \quad \forall i \]

2. For each cluster location, compute the mean location of all data-cases assigned to it:
   \[ \mu_{d,c} = \sum_{i \in S_c} median(\mathbf{x}_{d,i}) \quad \forall c \]

( the median is middle point (n=odd) or halfway between the middle 2 points (n=even) )
Vector Quantization

- K-means divides the space up in a Voronoi tessellation.
- Every point on a tile is summarized by the code-book vector “+”. This clearly allows for data compression!