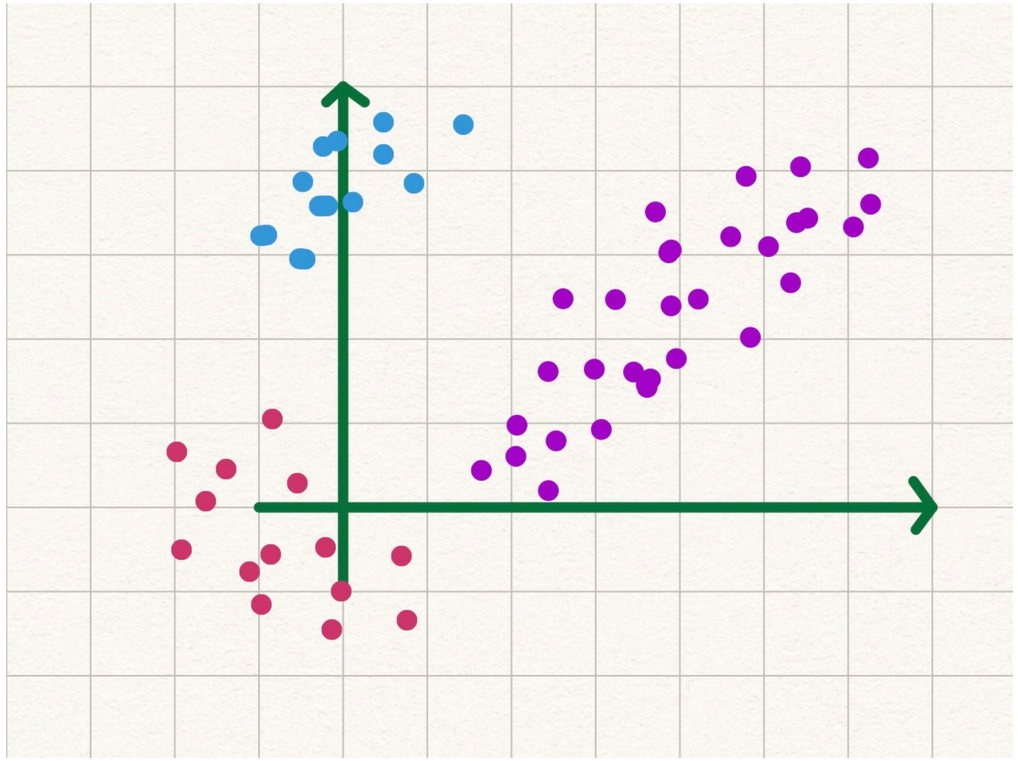
Applied Machine Learning

k-Means Clustering

k-Means Clustering

- Cluster optimization
- k-Means clustering algorithm
- Tuning of k-Means

Clustering

- Group similar data into Clusters of Blobs
- Agglomerative Clustering
  - start with a cluster per item
  - repetitive merging of clusters
- Cluster parameters
  - Cluster representation
  - Assignment of data items to cluster
  - Number of clusters
  - Evaluation of clusters
- 

Clustering

- Dataset X with items $\mathbf{x}_i$
- Optimize distances between items $\mathbf{x}_i$ and cluster centers $\mathbf{c}_j$
  - $\Phi(\delta, c) = \sum_{i,j} \delta_{i,j} \left[(\mathbf{x}_i - \mathbf{c}_j)^\top (\mathbf{x}_i - \mathbf{c}_j) \right]$
    - $\delta_{i,j} = \begin{cases} 1 & \mathbf{x_i} \text{ belongs to cluster } j \\ 0 & \text{otherwise} \end{cases}$
    - $\sum_j \delta_{i,j} = 1 \qquad \sum_i \delta_{i,j} > 0$

k-Means Clustering

1. Initialization: choose k data items as cluster centers $\mathbf{c}_j$
2. While (cluster centers have significant changes)
    1. For each data item $\mathbf{x}_i$
        - closest_center_from($\mathbf{x}_i$).assign($\mathbf{x}_i$)
    2. For each empty cluster center $\mathbf{c}_j$
        - $\mathbf{c}_j$.assign(item_far_from_its_center())
    - For each cluster center $\mathbf{c}_j$
        - $\mathbf{c}_j$.center = $\mathbf{c}_j$.mean()

k-Means Clustering: Tuning

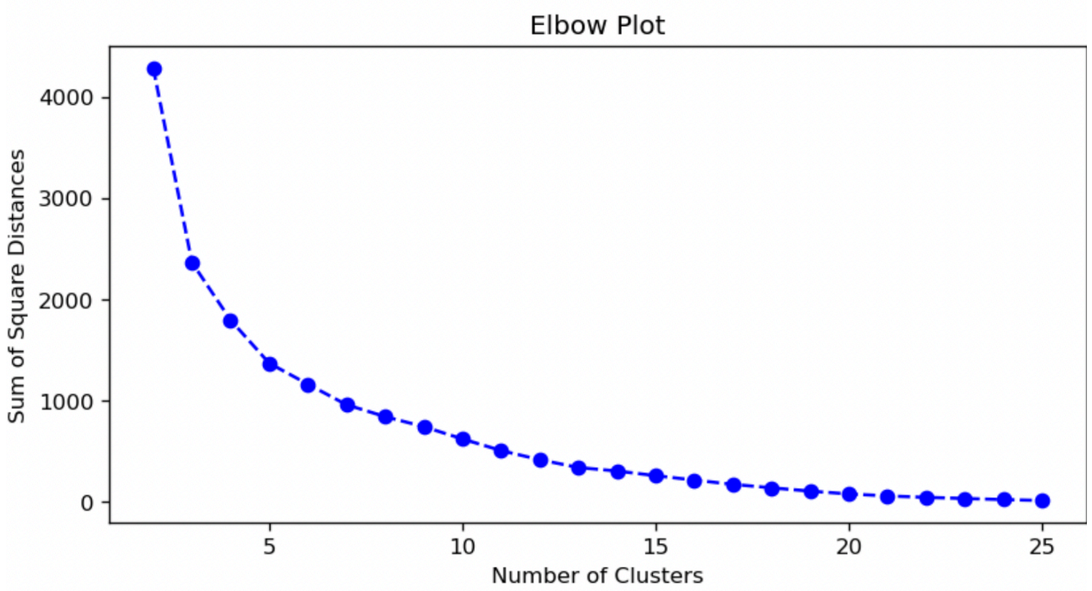
1. Initialization: choose k data items as cluster centers $\mathbf{c}_j$

- Poor choice of initial centers may lead to a poor clustering
- k-means++
  - first cluster center: item $\mathbf{x}$. random selection from dataset with uniform distribution
  - remaining $k-1$ centers: random selection from probability distribution:
    - $\frac{d_i^2(\mathbf{x})}{\sum_u d_u^2(\mathbf{x})}$, with $d_i^2(\mathbf{x})$ the squared distance between point i and center $\mathbf{X}$

k-Means Clustering: Tuning

1. Initialization: choose k data items as cluster centers $\mathbf{c}_j$

- Selection of k
  - too big: many isolated clusters
  - too low: spread clusters: high cost function Φ
  - cost function Φ decreases as k increases (0 when 1 cluster per item)
  - "knee" on Φ vs k
  - Other criteria
    - silhouette score to evaluate similarity of objects in each cluster compared to other clusters
    - performance in application



k-Means Clustering

- Cluster optimization
- k-Means clustering algorithm
- Tuning of k-Means

Applied Machine Learning

k-Means Clustering