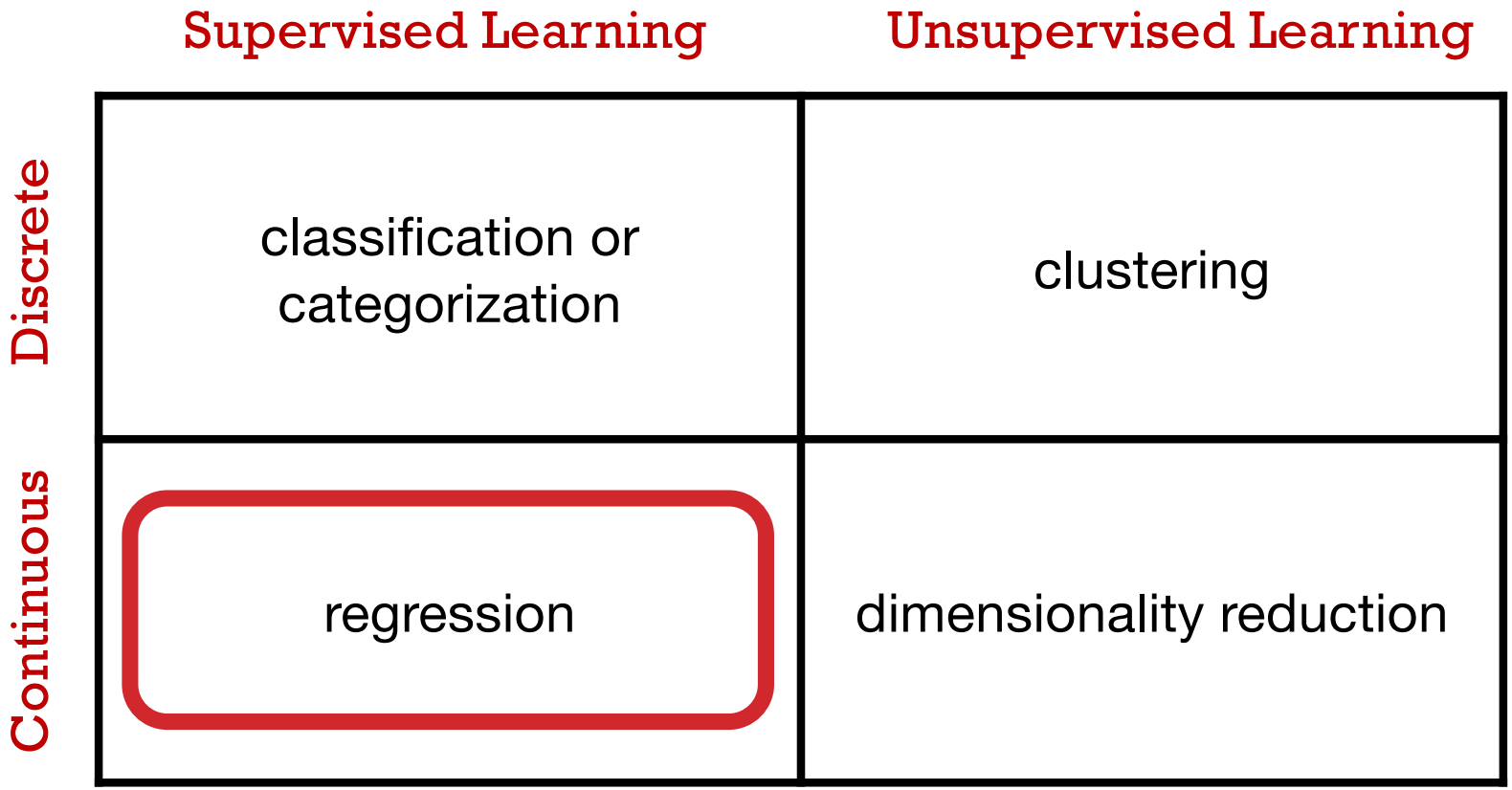
MACHINE LEARNING OVERVIEW

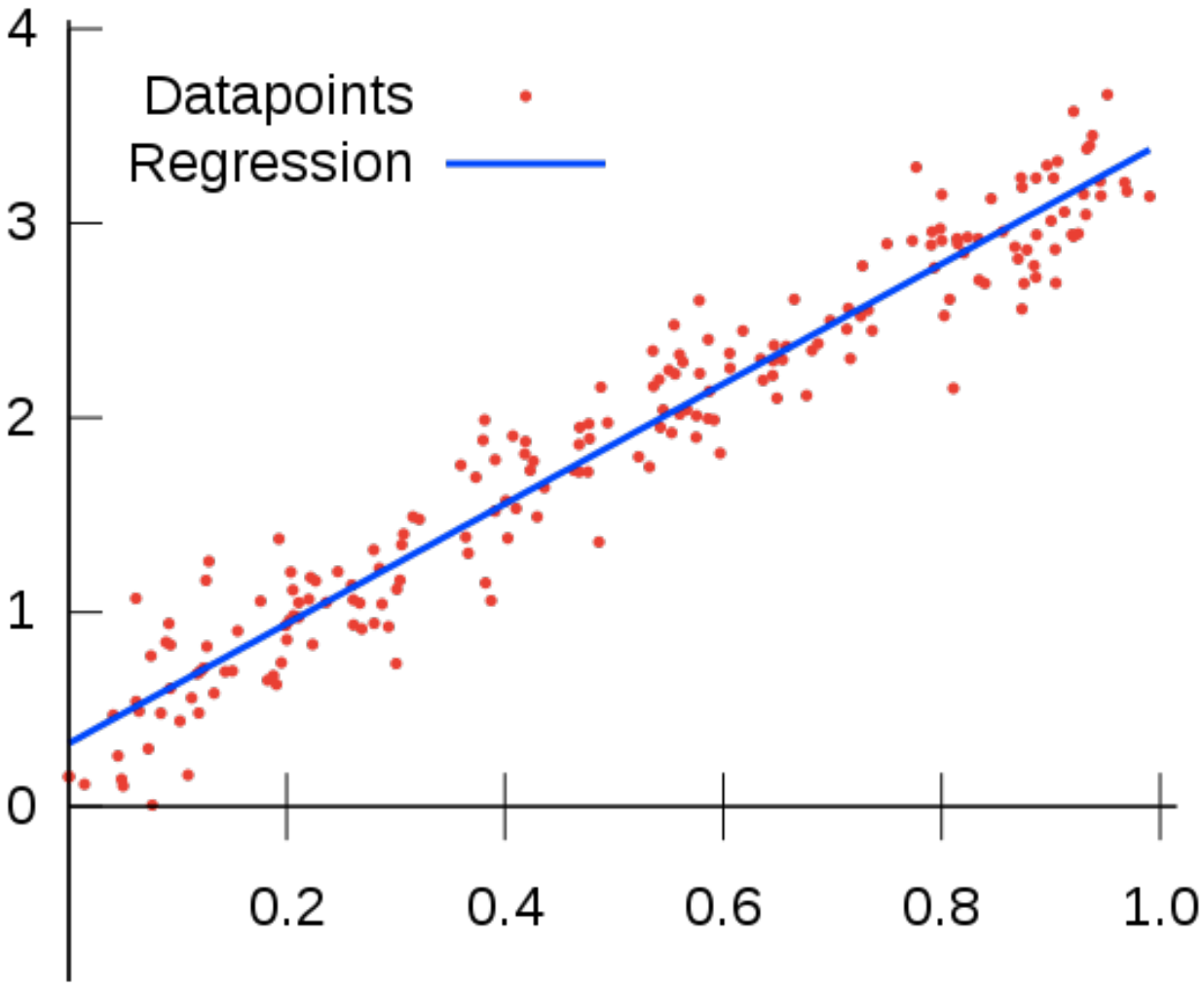
MACHINE LEARNING PROBLEMS

	Supervised Learning	Unsupervised Learning
Discrete	classification or categorization	clustering
Continuous	regression	dimensionality reduction

MACHINE LEARNING PROBLEMS

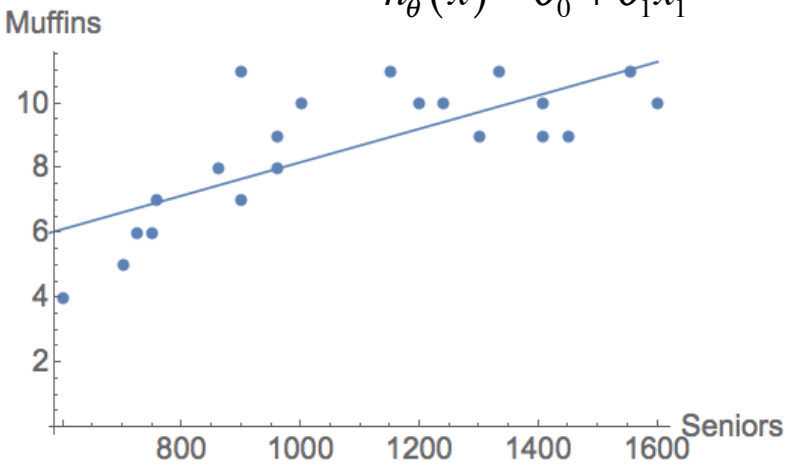


LINEAR REGRESSION

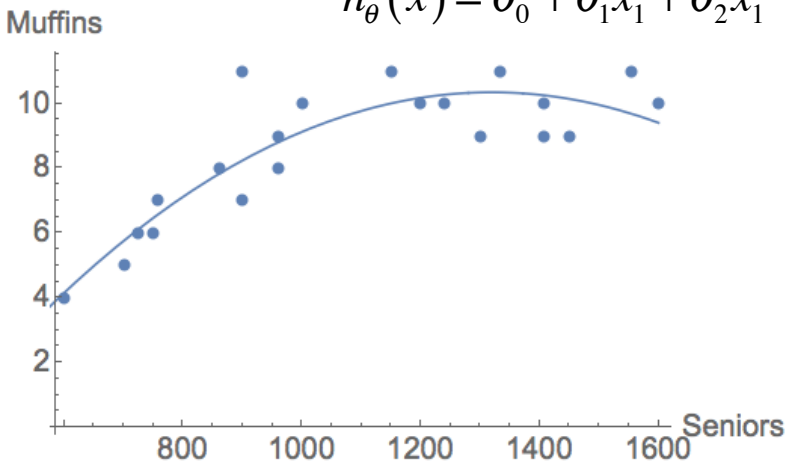


POLYNOMIAL REGRESSION

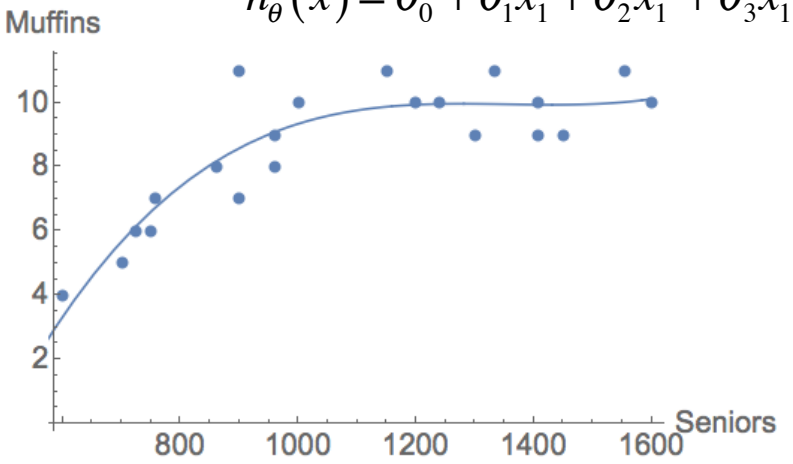
$$h_\theta(x) = \theta_0 + \theta_1 x_1$$



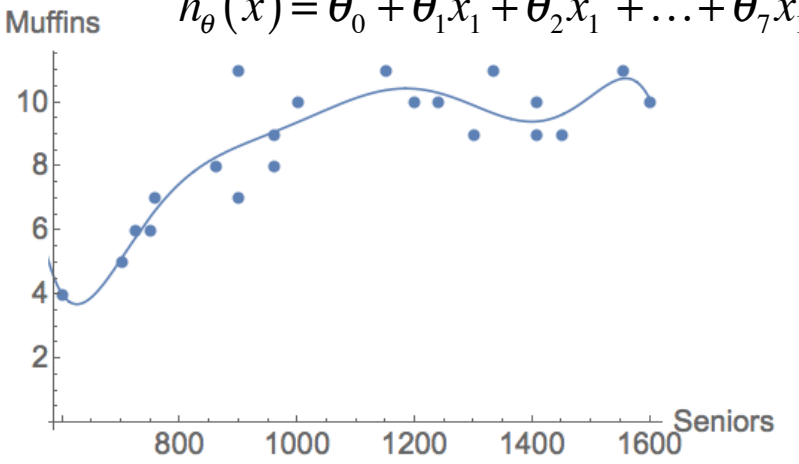
$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_1^2$$



$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_1^2 + \theta_3 x_1^3$$

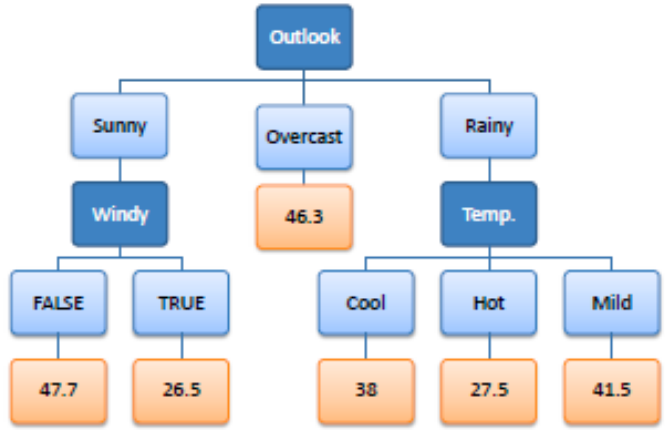


$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_1^2 + \ldots + \theta_7 x_1^7$$

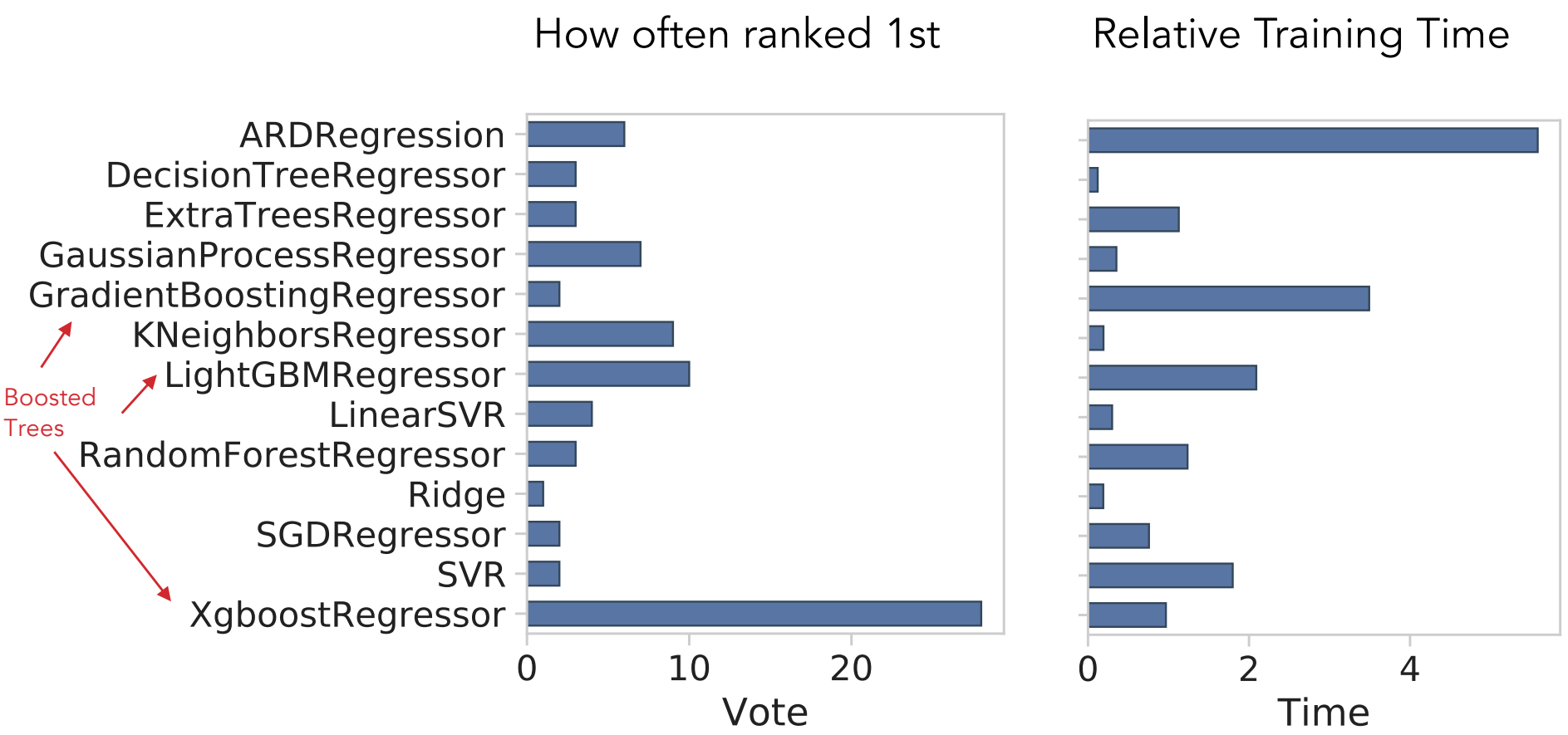


DECISION TREE - REGRESSION

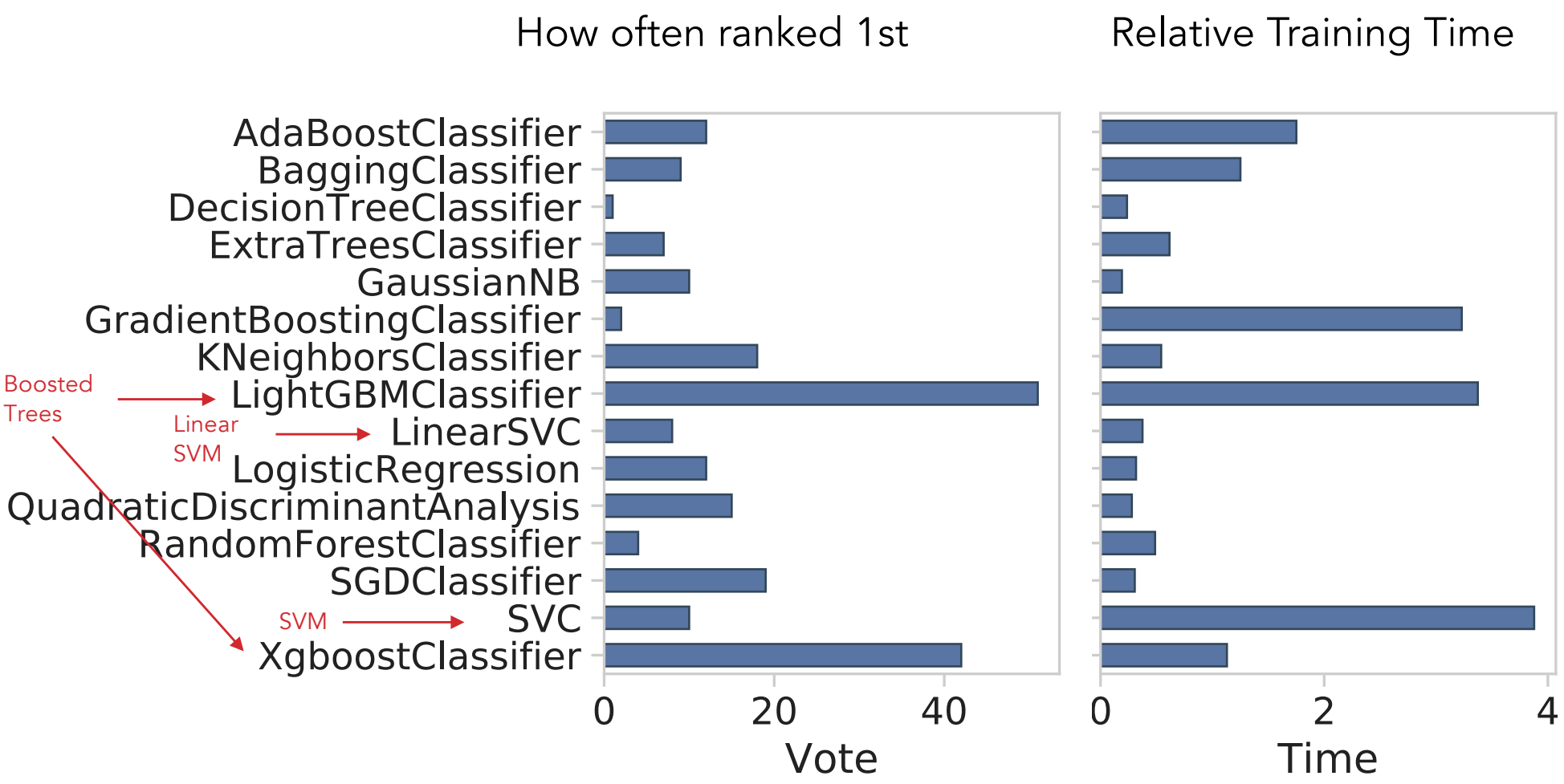
Predictors				Target
Outlook	Temp	Humidity	Windy	Hours Played
Rainy	Hot	High	False	25
Rainy	Hot	High	True	30
Overcast	Hot	High	False	48
Sunny	Mild	High	False	46
Sunny	Cool	Normal	False	62
Sunny	Cool	Normal	True	23
Overcast	Cool	Normal	True	43
Rainy	Mild	High	False	36
Rainy	Cool	Normal	False	38
Sunny	Mild	Normal	False	48
Rainy	Mild	Normal	True	48
Overcast	Mild	High	True	62
Overcast	Hot	Normal	False	44
Sunny	Mild	High	True	30



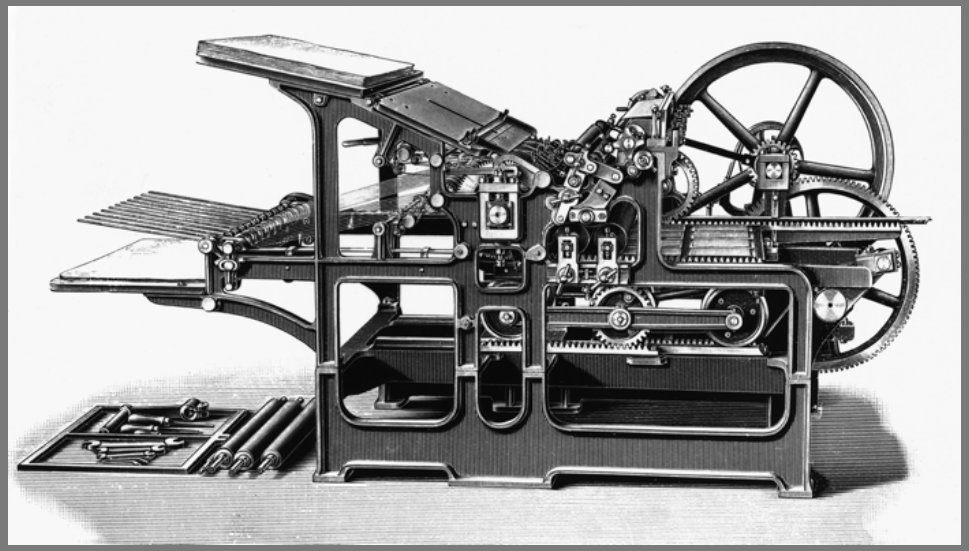
REGRESSION PERFORMANCE



CLASSIFIER PERFORMANCE



VS



Clustering

CLUSTERING STRATEGIES

K-means

- Iteratively re-assign points to the nearest cluster center

Agglomerative clustering

- Start with each point as its own cluster and iteratively merge the closest clusters

Mean-shift clustering

- Estimate modes of PDF (i.e., the value x at which its probability mass function takes its maximum value)

Spectral clustering

- Split the nodes in a graph based on assigned links with similarity weights

DBSCAN (Density-based spatial clustering of applications with noise)

As we go down this chart, the clustering strategies have more tendency to transitively group points even if they are not nearby in feature space

CLUSTERING STRATEGIES

K-means

- Iteratively re-assign points to the nearest cluster center

Agglomerative clustering

- Start with each point as its own cluster and iteratively merge the closest clusters

Mean-shift clustering

- Estimate modes of PDF (i.e., the value x at which its probability mass function takes its maximum value)

Spectral clustering

- Split the nodes in a graph based on assigned links with similarity weights

DBSCAN (Density-based spatial clustering of applications with noise)

As we go down this chart, the clustering strategies have more tendency to transitively group points even if they are not nearby in feature space

K-MEANS ALGORITHM

Select K random data points $\{s_1, s_2, \ldots s_K\}$ as centroids c_j.

Until clustering converges or other stopping criterion {

For each data point x_i:

Assign x_i to the closes centroid such that

$dist(x_i, c_j)$ is minimal.

For each cluster c_j, update the centroids

$c_j = \mu(c_j)$

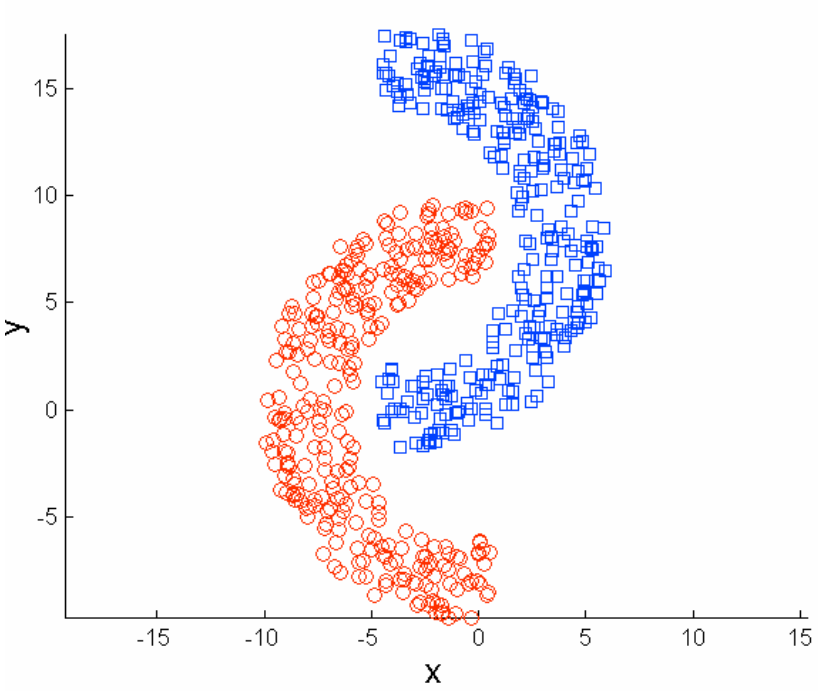
}

Several possible stopping criterions, e.g.,

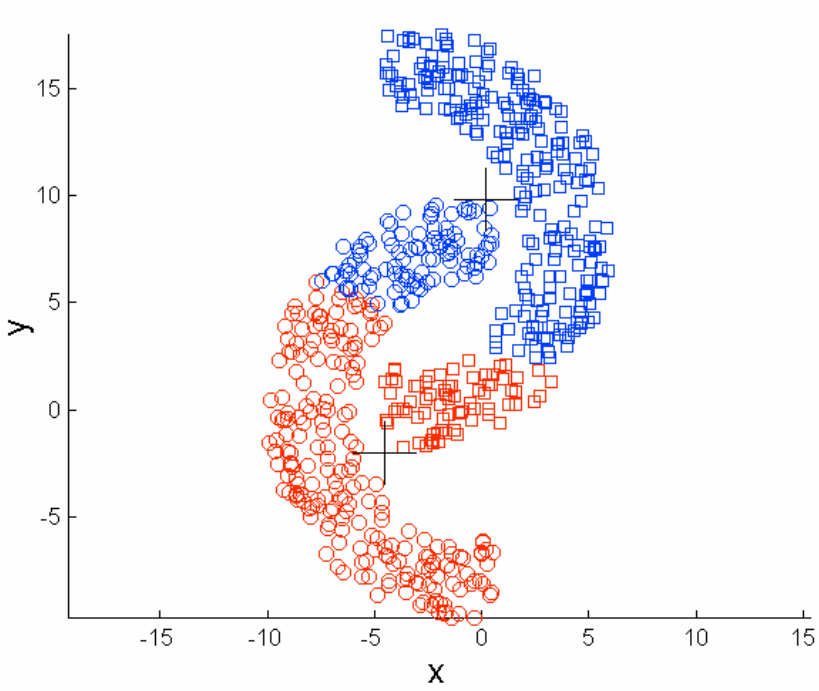
- A fixed number of iterations.
- Partition unchanged.
- Centroid positions don't change.

CAN YOU THINK OF AN EXAMPLE FOR WHICH K-MEANS IS NOT A GOOD CLUSTERING ALGORITHMS?

Non-globular Shapes

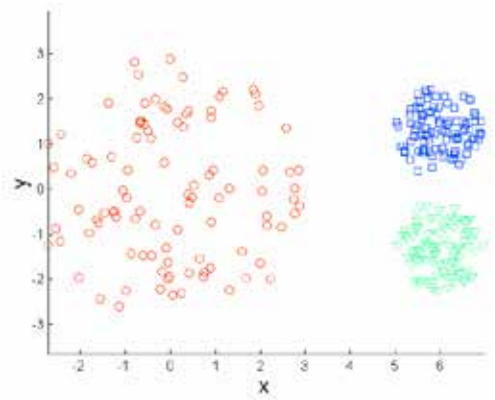


Original Points

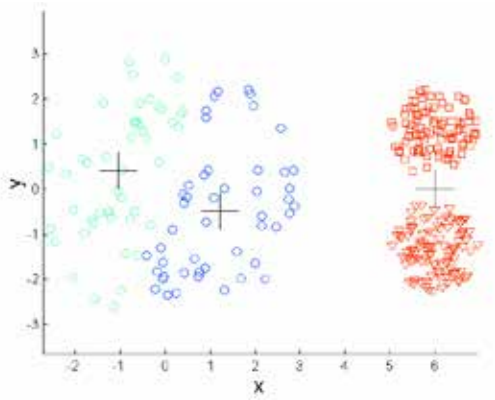


K-means (2 Clusters)

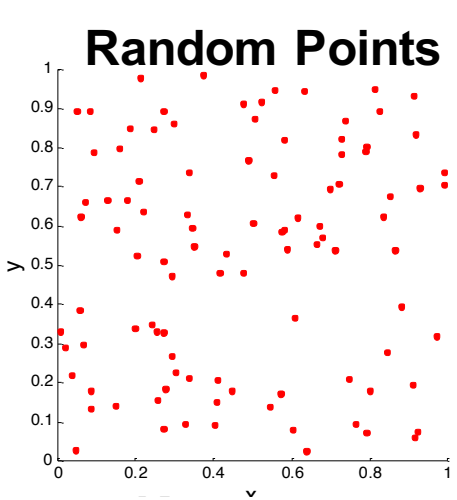
How do I know how good the clustering is?

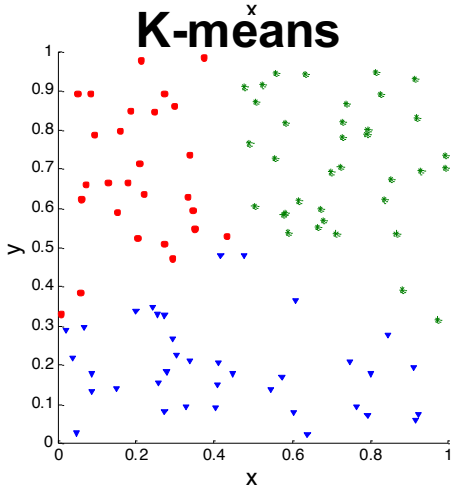


Original Points



K-means (3 Clusters)

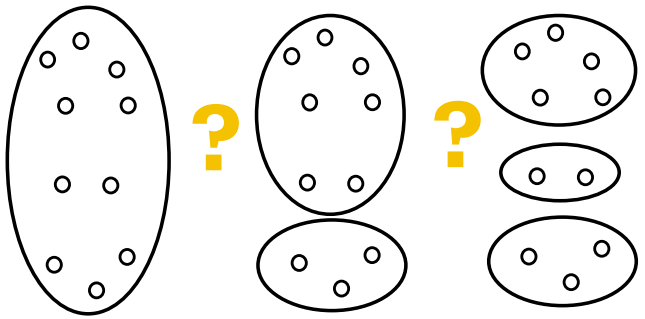




Measuring clustering validity

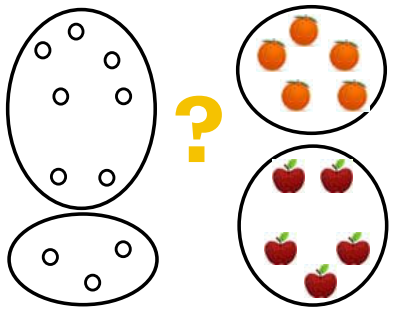
Internal Index:

- Validate *without* external info
- With different number of clustersl



External Index

Validate against ground truth



INTERNAL INDEXES

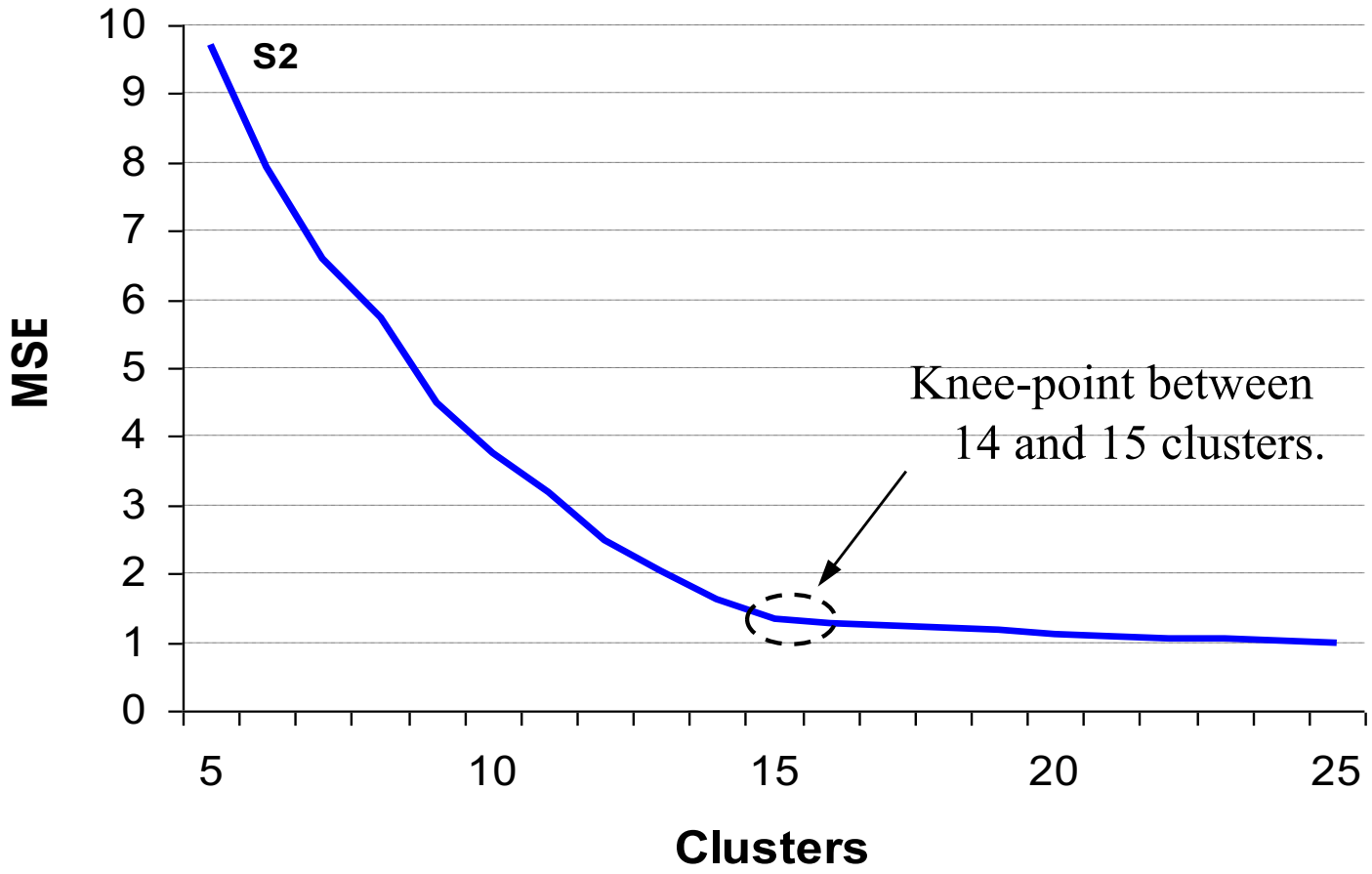
Ground truth is rarely available but unsupervised validation must be done.

Minimizes (or maximizes) internal index:

- Variances of within cluster and between clusters
- Rate-distortion method
- F-ratio
- Davies-Bouldin index (DBI)
- Bayesian Information Criterion (BIC)
- Silhouette Coefficient
- Minimum description principle (MDL)
- Stochastic complexity (SC)

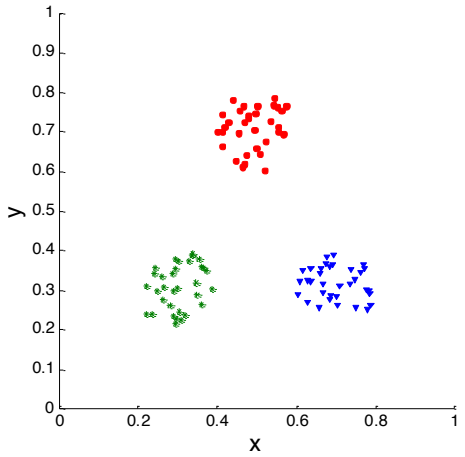
IDEA: MEAN SQUARE ERROR (MSE)

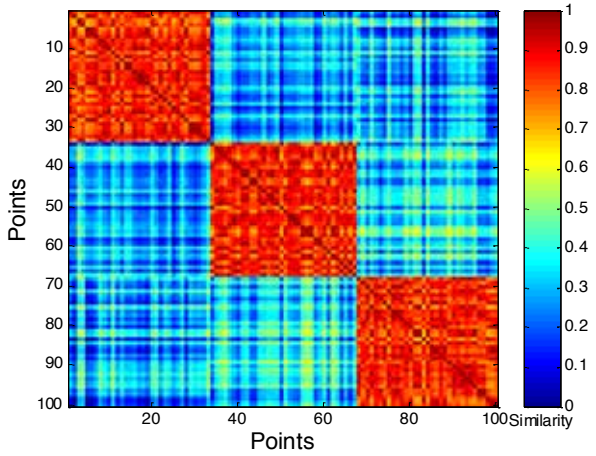
- The more clusters the smaller the MSE.
- Small knee-point near the correct value.
- But how to detect?

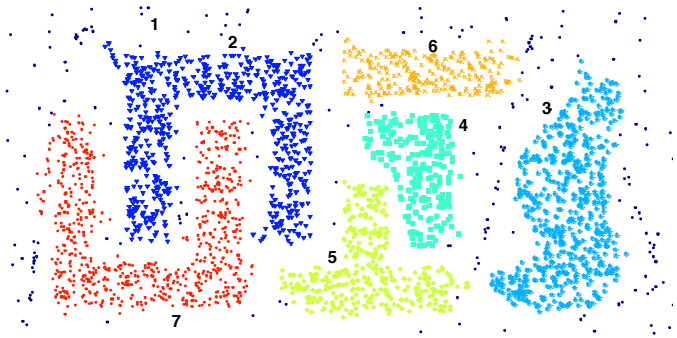


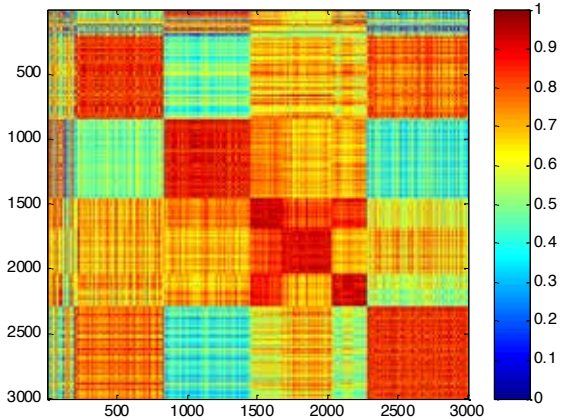
USING SIMILARITY MATRIX FOR CLUSTER VALIDATION

Order the similarity matrix with respect to cluster labels and inspect visually.









CLUSTERING STRATEGIES

K-means

- Iteratively re-assign points to the nearest cluster center

Agglomerative clustering

- Start with each point as its own cluster and iteratively merge the closest clusters

Mean-shift clustering

- Estimate modes of PDF (i.e., the value x at which its probability mass function takes its maximum value)

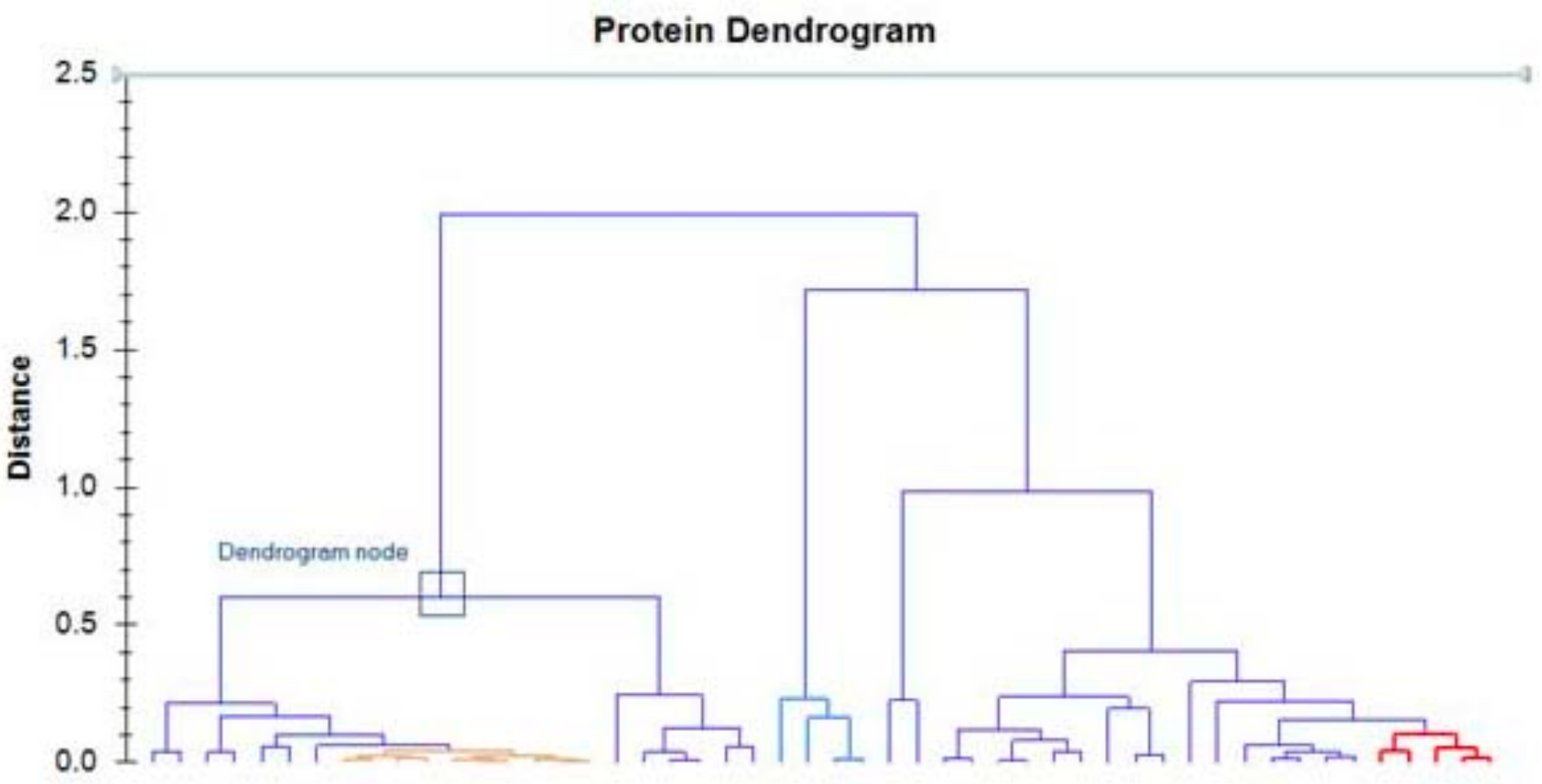
Spectral clustering

- Split the nodes in a graph based on assigned links with similarity weights

DBSCAN (Density-based spatial clustering of applications with noise)

As we go down this chart, the clustering strategies have more tendency to transitively group points even if they are not nearby in feature space

DENDROGRAM EXAMPLE

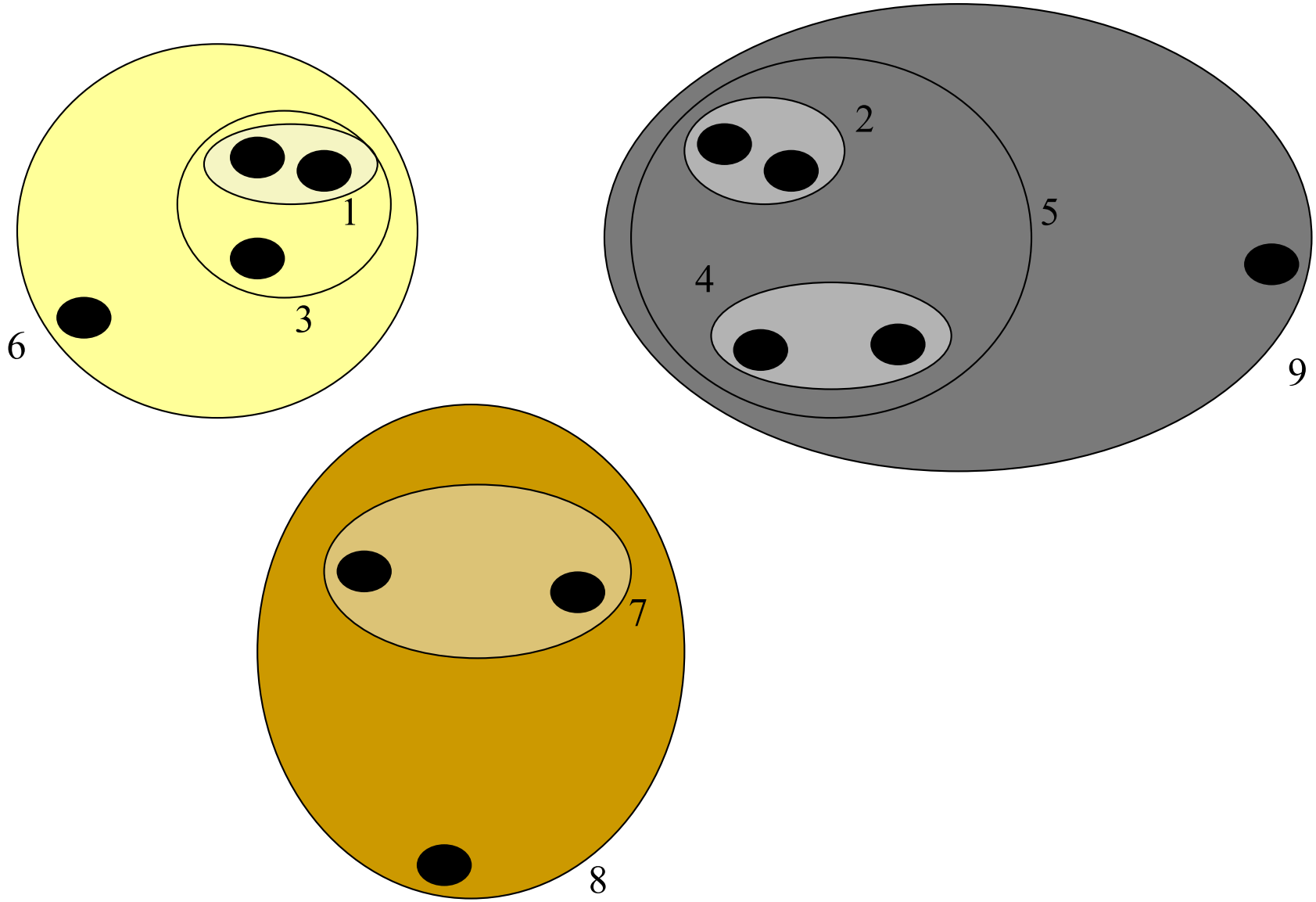


HIERARCHICAL AGGLOMERATIVE CLUSTERING METHODS

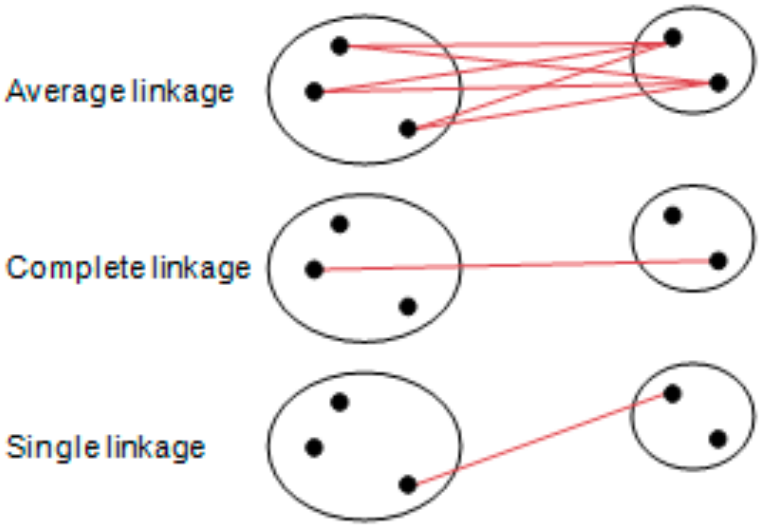
Generic Agglomerative Procedure (Salton '89):

1. Compute all pairwise document-document similarity coefficients
2. Place each of *n* documents into a class of its own
3. Merge the two most similar clusters into one;
   - replace the two clusters by the new cluster
   - recompute intercluster similarity scores w.r.t. the new cluster
4. Repeat the above step until there are only k clusters left (note k could = 1).

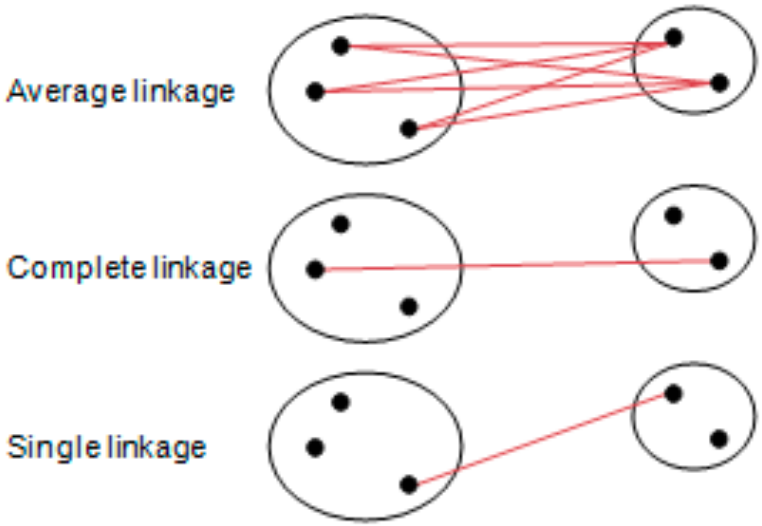
Group Agglomerative Clustering

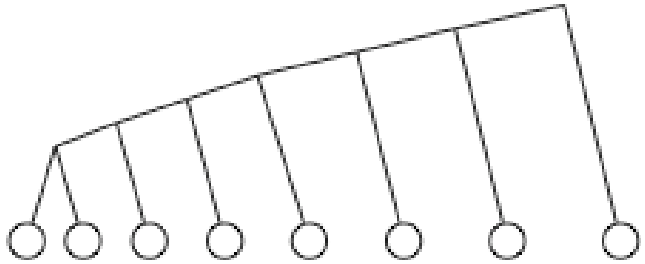


LINKAGE TYPES



CLICKER - HTTPS://CLICKER.CSAIL.MIT.EDU/6.S080/

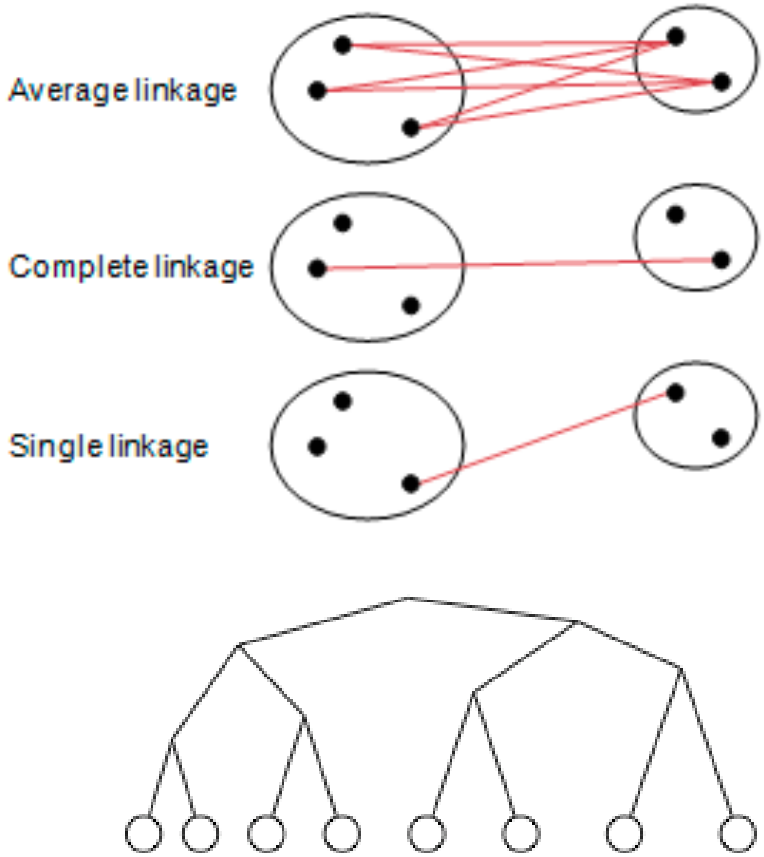




Which linkage type was used for this clustering?

a) Average
b) Complete
c) Single

CLICKER - HTTPS://CLICKER.CSAIL.MIT.EDU/6.S080/



Which linkage type was used for this clustering?

a) Average
b) Complete
c) Single

CLUSTERING STRATEGIES

K-means

- Iteratively re-assign points to the nearest cluster center

Agglomerative clustering

- Start with each point as its own cluster and iteratively merge the closest clusters

Mean-shift clustering

- Estimate modes of PDF (i.e., the value x at which its probability mass function takes its maximum value)

Spectral clustering

- Split the nodes in a graph based on assigned links with similarity weights

DBSCAN (Density-based spatial clustering of applications with noise)

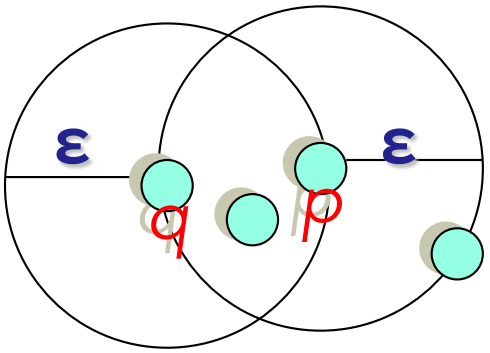
As we go down this chart, the clustering strategies have more tendency to transitively group points even if they are not nearby in feature space

ε-NEIGHBORHOOD

ε-Neighborhood – Objects within a radius of ε from an object.

$$N_{\varepsilon}(p) : \{q \mid d(p,q) \leq \varepsilon\}$$

"High density" - ε-Neighborhood of an object contains at least *MinPts* of objects.



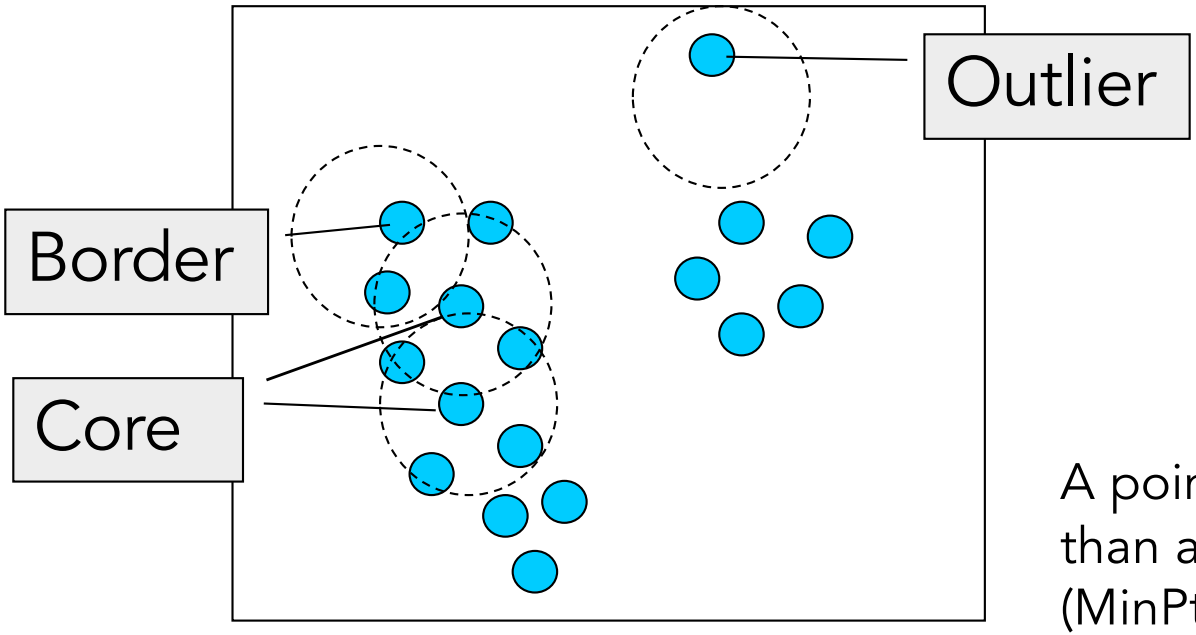
ε-Neighborhood of *p*
ε-Neighborhood of *q*

Density of p is "high" (MinPts = 4)

Density of q is "low" (MinPts = 4)

Slides on DBSCAN partially taking and modified from:
https://www.cse.buffalo.edu/faculty/azhang/cse601/density-based.ppt

CORE, BORDER & OUTLIER



ε = 1unit, MinPts = 5

Given *ε* and *MinPts*, categorize the objects into three exclusive groups.

A point is a core point if it has more than a specified number of points (MinPts) within Eps These are points that are at the interior of a cluster.

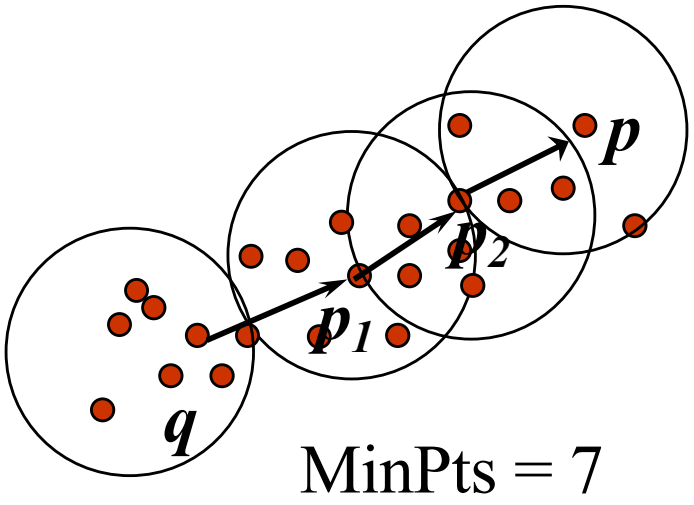
A border point has fewer than MinPts within Eps, but is in the neighborhood of a core point..

A noise point (outlier) is any point that is not a core point nor a border point.

DENSITY-REACHABILITY

Density-Reachable (directly and indirectly):

- A point p is directly density-reachable from p2;
- p2 is directly density-reachable from p1;
- p1 is directly density-reachable from q;
- p←p2←p1←q form a chain.



p is (indirectly) density-reachable from q

q is not density- reachable from p?

DBSCAN ALGORITHM

Input: The data set D

Parameter: ε, MinPts

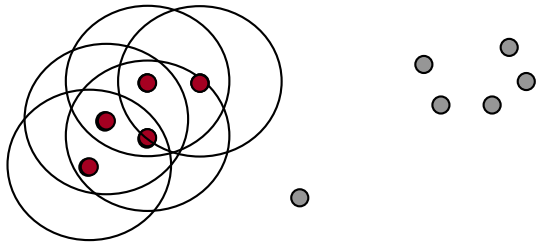
For each object p in D
    if p is a core object and not processed then
        C = retrieve all objects density-reachable from p
        mark all objects in C as processed
        report C as a cluster
    else mark p as outlier
    end if

End For

DBSCAN ALGORITHM: EXAMPLE

Parameter

- ε = 2 cm
- *MinPts* = 3

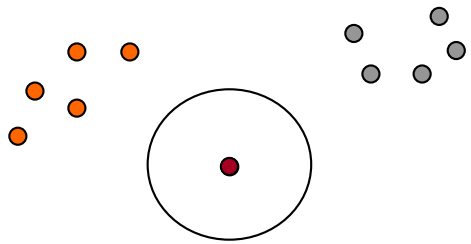


```
for each o ∈ D do
    if o is not yet classified then
        if o is a core-object then
            collect all objects density-reachable from o
            and assign them to a new cluster.
        else
            assign o to NOISE
```

DBSCAN ALGORITHM: EXAMPLE

Parameter

- ε = 2 cm
- *MinPts* = 3

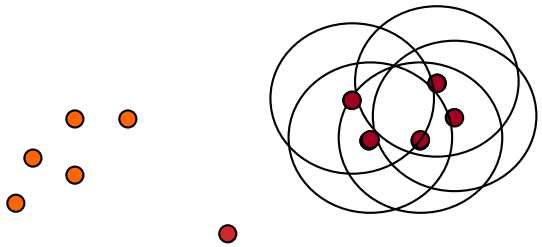


```
for each o ∈ D do
    if o is not yet classified then
        if o is a core-object then
            collect all objects density-reachable from o
            and assign them to a new cluster.
        else
            assign o to NOISE
```

DBSCAN ALGORITHM: EXAMPLE

Parameter

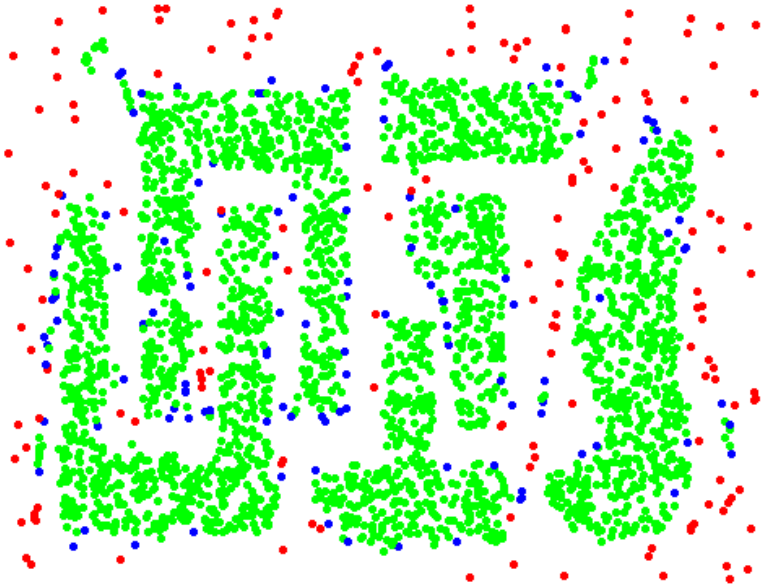
- ε = 2 cm
- *MinPts* = 3



```
for each o ∈ D do
    if o is not yet classified then
        if o is a core-object then
            collect all objects density-reachable from o
            and assign them to a new cluster.
        else
            assign o to NOISE
```

EXAMPLE

Original Points

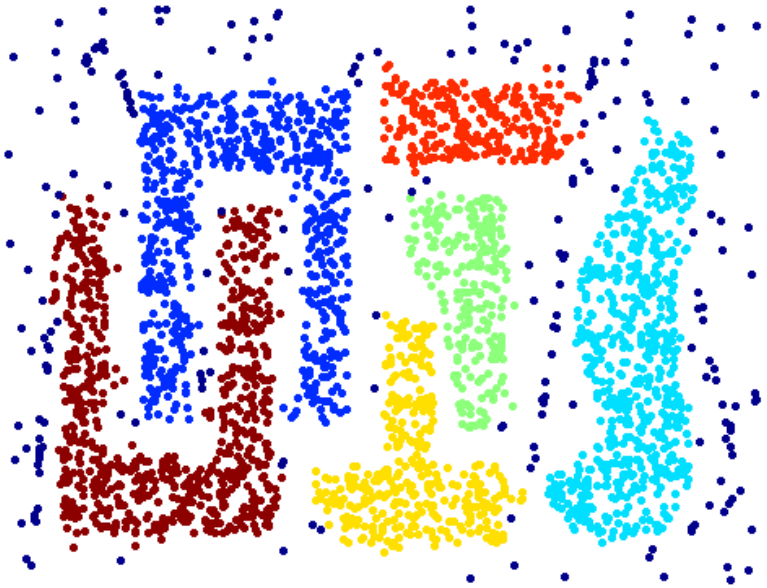


Point types: core, border and outliers

ε = 10, MinPts = 4

WHEN DBSCAN WORKS WELL

Original Points

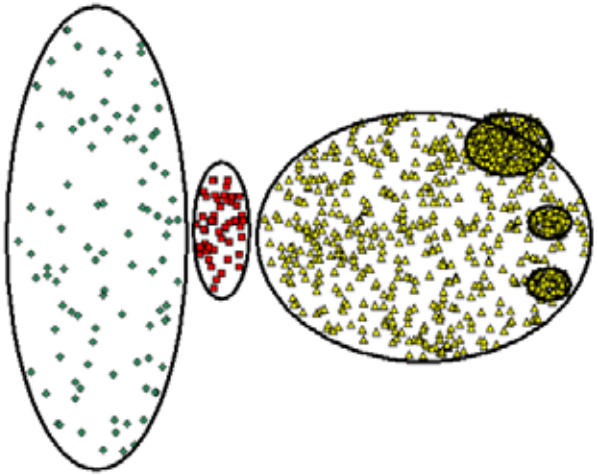


Clusters

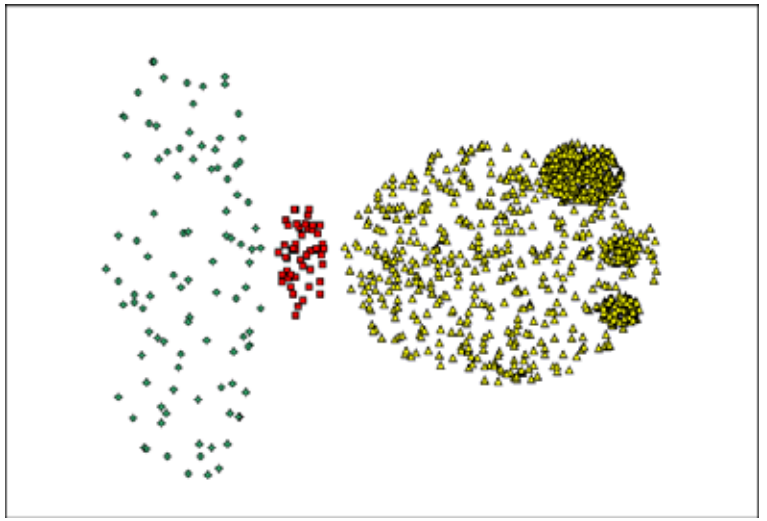
- **Resistant to Noise**
- **Can handle clusters of different shapes and sizes**

CAN YOU CREATE AN EXAMPLE FOR WHICH DBSCAN WILL NOT WORK WELL

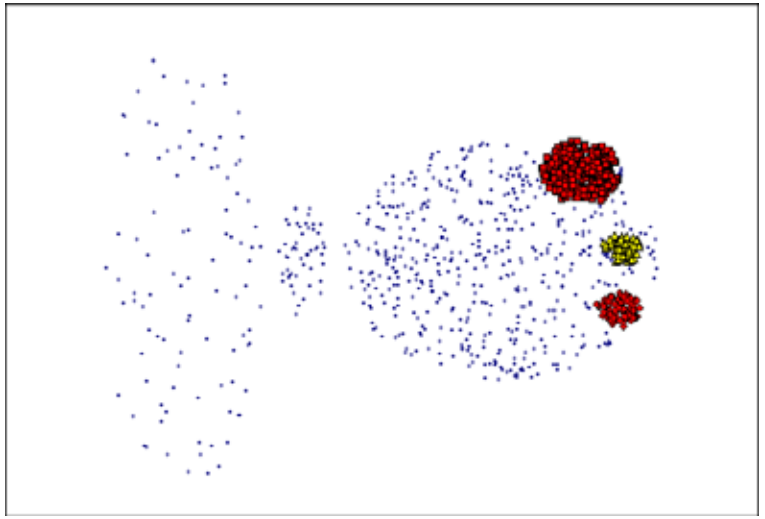
WHEN DBSCAN DOES NOT WORK WELL



Original Points



(MinPts=4, Eps=9.92).

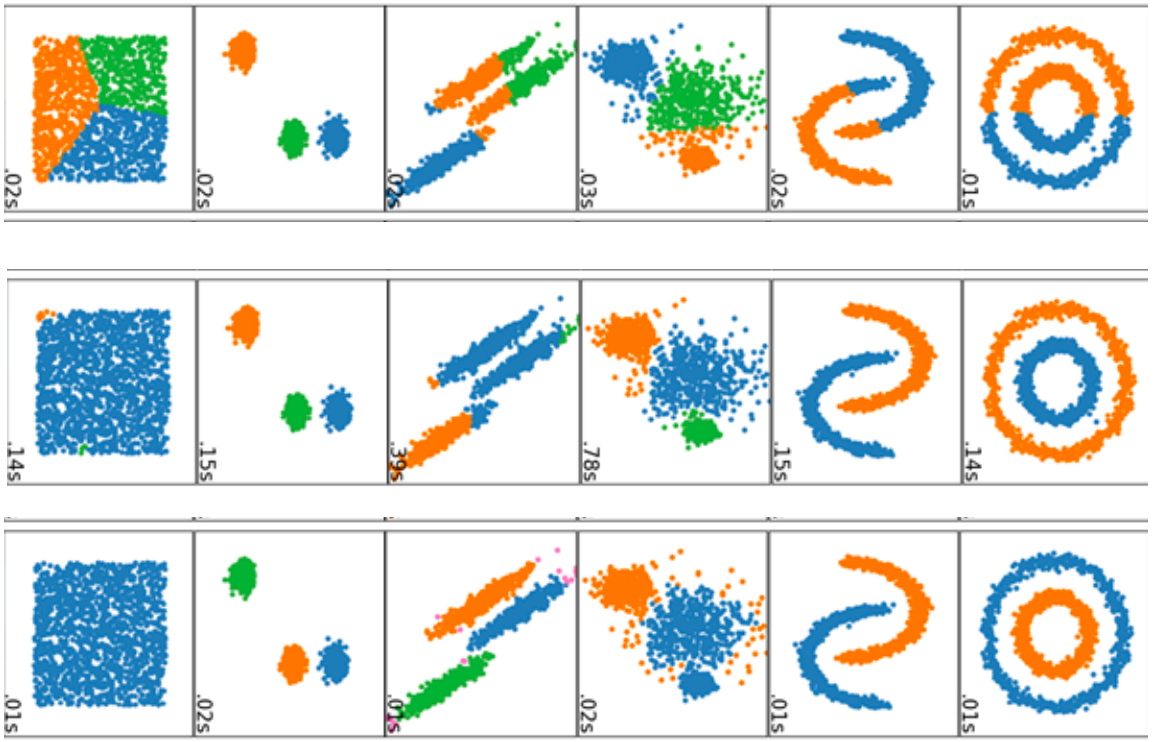


(MinPts=4, Eps=9.75)

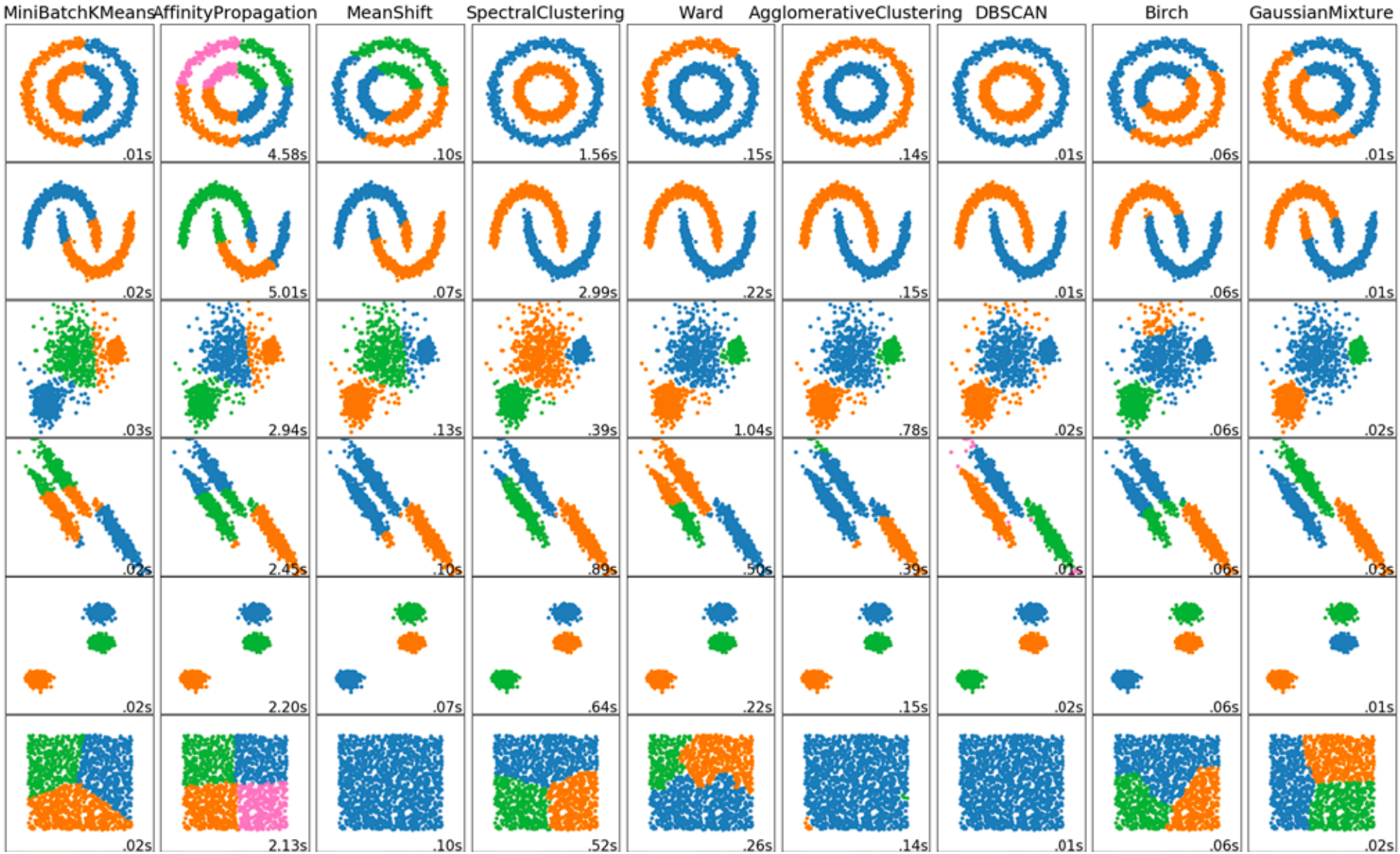
- Cannot handle Varying densities
- Sensitive to parameters

CLICKER - HTTPS://CLICKER.CSAIL.MIT.EDU/6.S080/

A	B	C
KMeans	Aggl. clustering	KMeans
DBScan	KMeans	Aggl. clustering
Aggl. clustering	DBScan	DBScan



CLUSTERING


MiniBatchKMeans
AffinityPropagation
MeanShift
SpectralClustering
Ward
AgglomerativeClustering
DBSCAN
Birch
GaussianMixture
.01s
4.58s
.10s
1.56s
.15s
.14s
.01s
.06s
.01s
.02s
5.01s
.07s
2.99s
.22s
.15s
.01s
.06s
.01s
.03s
2.94s
.13s
.39s
1.04s
.78s
.02s
.06s
.02s
.02s
2.45s
.10s
.89s
.50s
.39s
.01s
.06s
.03s
.02s
2.20s
.07s
.64s
.22s
.15s
.02s
.06s
.01s
.02s
2.13s
.10s
.52s
.26s
.14s
.01s
.06s
.02s

OUTLIER DETECTION

Slides on outlier detection based on Tan,Steinbach, Kumar: Introduction to Data Mining

ANOMALY/OUTLIER DETECTION

What are anomalies/outliers?

- The set of data points that are considerably different than the remainder of the data

Variants of Anomaly/Outlier Detection Problems

- Given a database D, find all the data points $\mathbf{x} \in D$ with anomaly scores greater than some threshold t
- Given a database D, find all the data points $\mathbf{x} \in D$ having the top-n largest anomaly scores f($\mathbf{x}$)
- Given a database D, containing mostly normal (but unlabeled) data points, and a test point $\mathbf{x}$, compute the anomaly score of $\mathbf{x}$ with respect to D

Applications:

- Credit card fraud detection, telecommunication fraud detection, network intrusion detection, fault detection

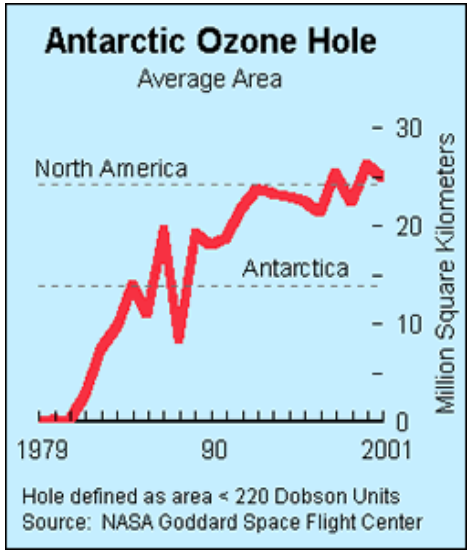
OUTLIER REMOVAL CAN BE DANGEROUS

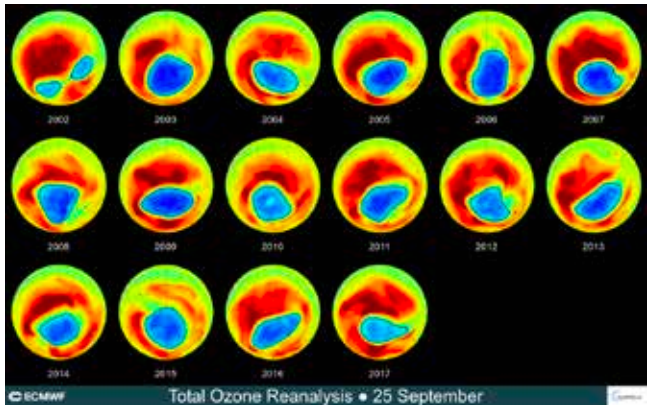
Ozone Depletion History

- In 1985 three researchers (Farman, Gardinar and Shanklin) were puzzled by data gathered by the British Antarctic Survey showing that ozone levels for Antarctica had dropped 10% below normal levels

- Why did the Nimbus 7 satellite, which had instruments aboard for recording ozone levels, not record similarly low ozone concentrations?

- The ozone concentrations recorded by the satellite were so low they were being treated as outliers by a computer program and discarded!





Sources:
http://exploringdata.cqu.edu.au/ozone.html
http://www.epa.gov/ozone/science/hole/size.html

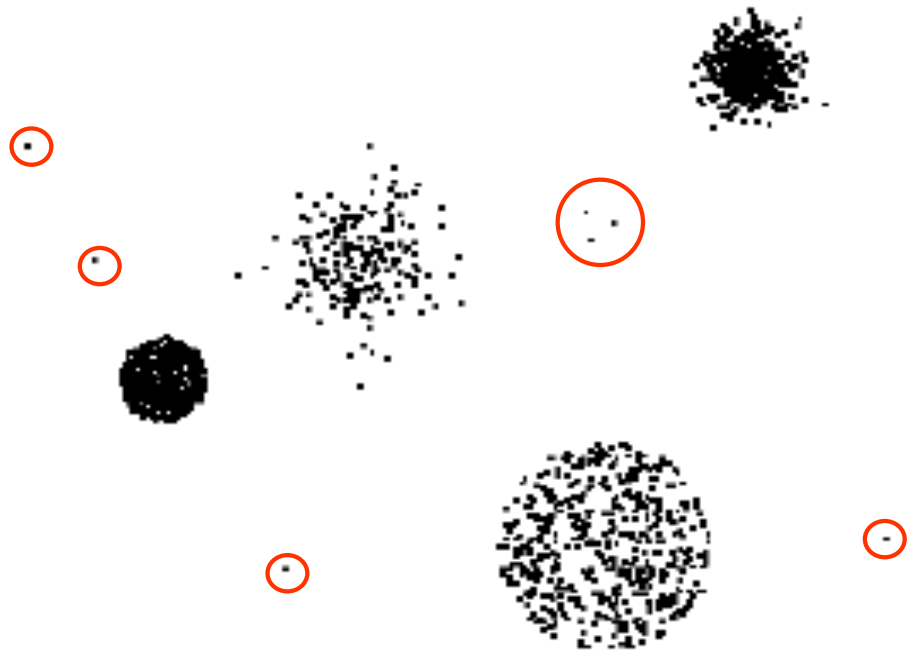
ANOMALY DETECTION SCHEMES

General Steps

- Build a profile of the "normal" behavior
  - Profile can be patterns or summary statistics for the overall population
- Use the "normal" profile to detect anomalies
  - Anomalies are observations whose characteristics differ significantly from the normal profile

Types of anomaly detection schemes

- Graphical
- Model-based
- Distance-based
- Clustering-based

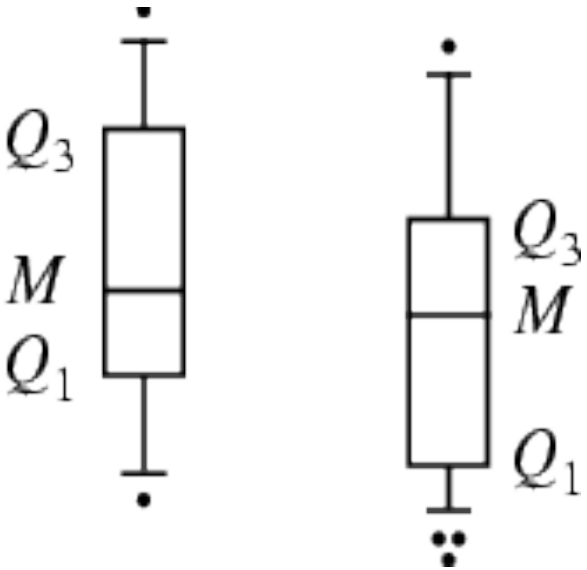


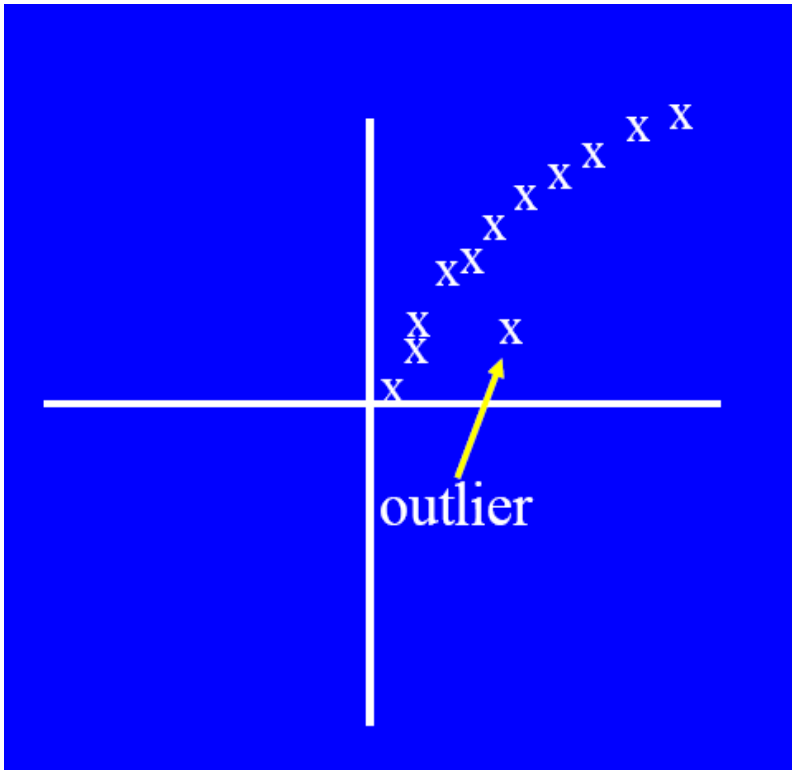
GRAPHICAL APPROACHES

Boxplot (1-D), Scatter plot (2-D), Spin plot (3-D)

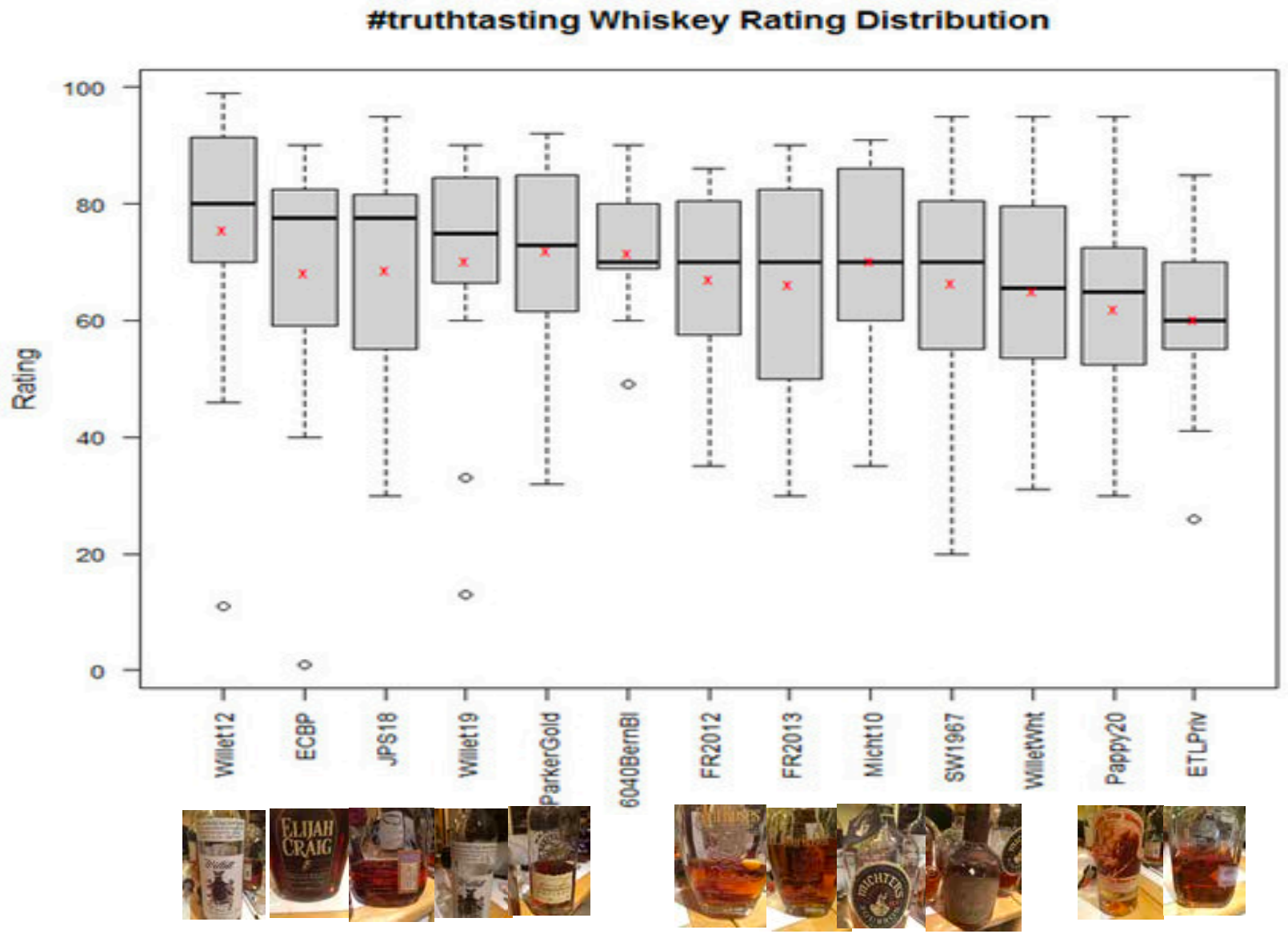
Limitations

- Time consuming
- Subjective





WHISKEY RATING – HOW DO YOU READ THE BOX PLOTS?



UNDERSTANDING BOX PLOTS

median (Q2/50th Percentile): the middle value of the dataset.

first quartile (Q1/25th Percentile): the middle number between the smallest number (not the "minimum") and the median of the dataset.

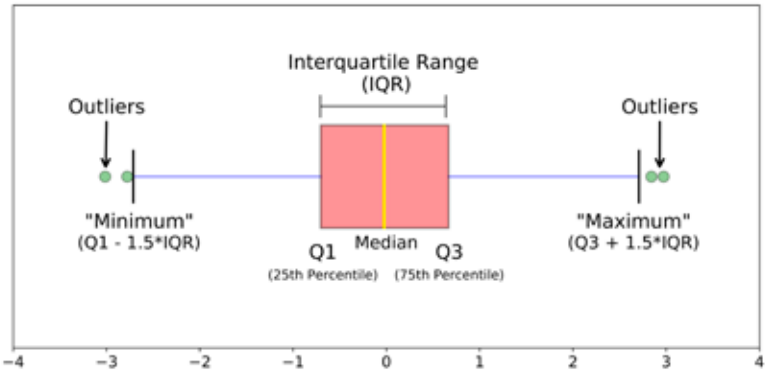
third quartile (Q3/75th Percentile): the middle value between the median and the highest value (not the "maximum") of the dataset.

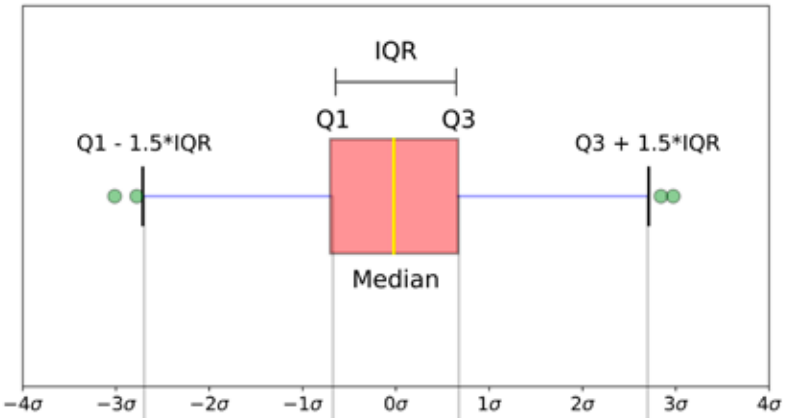
interquartile range (IQR): 25th to the 75th percentile.

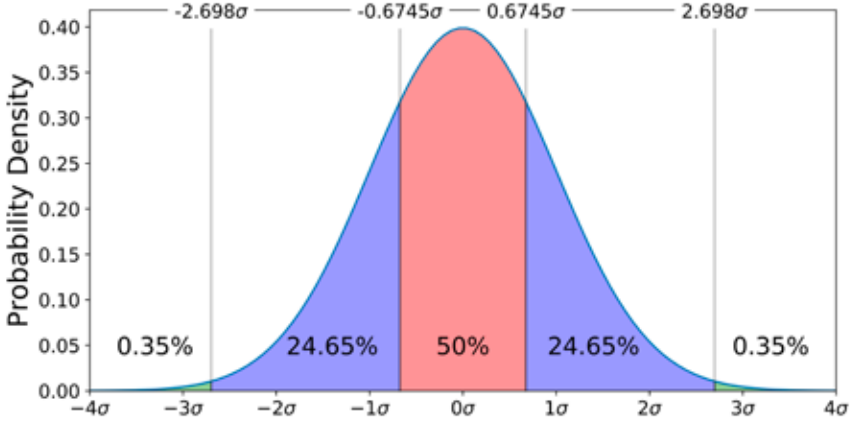
outliers (shown as green circles)

End of whiskers (shown in blue):

- **"maximum"**: Q3 + 1.5* IQR and **"minimum"**: Q1 -1.5* IQR (example right)
- the minimum and maximum of all of the data
- the lowest datum still within 1.5 IQR of the lower quartile, and the highest datum still within 1.5 IQR of the upper quartile (often called the **Tukey boxplot**)
- one standard deviation above and below the mean of the data
- the 9th percentile and the 91st percentile
- the 2nd percentile and the 98th percentile.





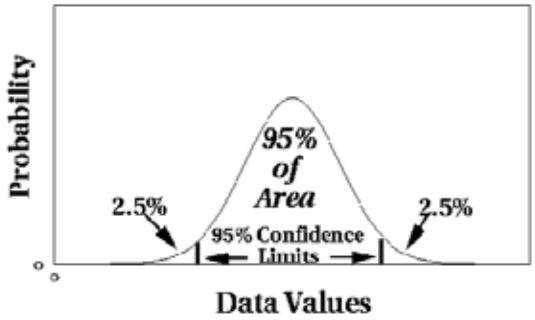


STATISTICAL APPROACHES---MODEL-BASED

Assume a parametric model describing the distribution of the data (e.g., normal distribution)

Apply a statistical test that depends on

- Data distribution
- Parameter of distribution (e.g., mean, variance)
- Number of expected outliers (confidence limit)



Common Approaches:

- Grubbs' Test (assumes normal distribution)
- Likelihood Approaches / EM-Algorithm

Limitations:

- Most of the tests are for a single attribute
- In many cases, data distribution/model may not be known
- For high dimensional data, it may be difficult to estimate the true distribution

DISTANCE-BASED APPROACHES

Data is represented as a vector of features

Three major approaches

- Nearest-neighbor based
- Density based
- Clustering based

Approach:

- Compute the distance between every pair of data points
- There are various ways to define outliers:
  - Data points for which there are fewer than *p* neighboring points within a distance *D*
  - The top n data points whose distance to the kth nearest neighbor is greatest
  - The top n data points whose average distance to the k nearest neighbors is greatest

CLUSTERING-BASED

Idea: Use a clustering algorithm that has some notion of outliers!

Problem what parameters should I choose for the algorithm; e.g. DBSCAN?

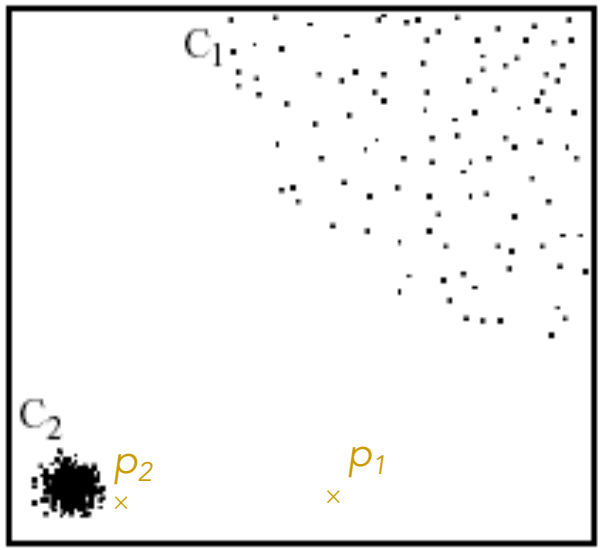
Rule of Thumb: Less than x% of the data should be outliers (with x typically chosen between 0.1 and 10); x might be determined with other methods; e.g. statistical tests.

DENSITY-BASED

For each point, compute the density of its local neighborhood; e.g. use Kmeans, DBSCAN's approach

Compute local outlier factor (LOF) of a sample *p* as the average of the ratios of the density of sample *p* and the density of its nearest neighbors

Outliers are points with largest LOF value



Alternative approach: directly use density function; e.g. DENCLUE's density function

Mining Association Rules in Large Databases

Slides partially taken from http://cs-people.bu.edu/evimaria/cs565-10/lect2.pdf

Association rules

- Given a set of transactions D, find rules that will predict the occurrence of an item (or a set of items) based on the occurrences of other items in the transaction

Market-Basket transactions

TID	*Items*
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

Examples of association rules

{Diaper} → {Beer},
{Milk, Bread} → {Diaper,Coke},
{Beer, Bread} → {Milk},

- Task 1: Methods for finding all frequent itemsets efficiently

- Task 2: Methods for finding association rules efficiently

An even simpler concept: frequent itemsets

- Given a set of transactions D, find combination of items that occur frequently

Market-Basket transactions

TID	*Items*
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

Examples of frequent itemsets

{Diaper, Beer},
{Milk, Bread}
{Beer, Bread, Milk},

Why are frequent itemsets interesting on their own?

Definition: Frequent Itemset

- Itemset
  - A set of one or more items
    - E.g.: {Milk, Bread, Diaper}
  - k-itemset
    - An itemset that contains k items
- Support count (σ)
  - Frequency of occurrence of an itemset (number of transactions it appears)
  - E.g. **σ({Milk, Bread,Diaper}) = 2**
- Support
  - Fraction of the transactions in which an itemset appears
  - E.g. **s({Milk, Bread, Diaper}) = 2/5**
- Frequent Itemset
  - An itemset whose support is greater than or equal to a *minsup* threshold

TID	*Items*
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

CLICKER - HTTPS://CLICKER.CSAIL.MIT.EDU/6.S080/

TID	date	items_bought
100	10/10/99	{F,A,D,B}
200	15/10/99	{D,A,C,E,B}
300	19/10/99	{C,A,B,E}
400	20/10/99	{B,A,D}

How many frequent itemsets (k>= 2) are there with a min support of 75%?

A) 1

B) 2

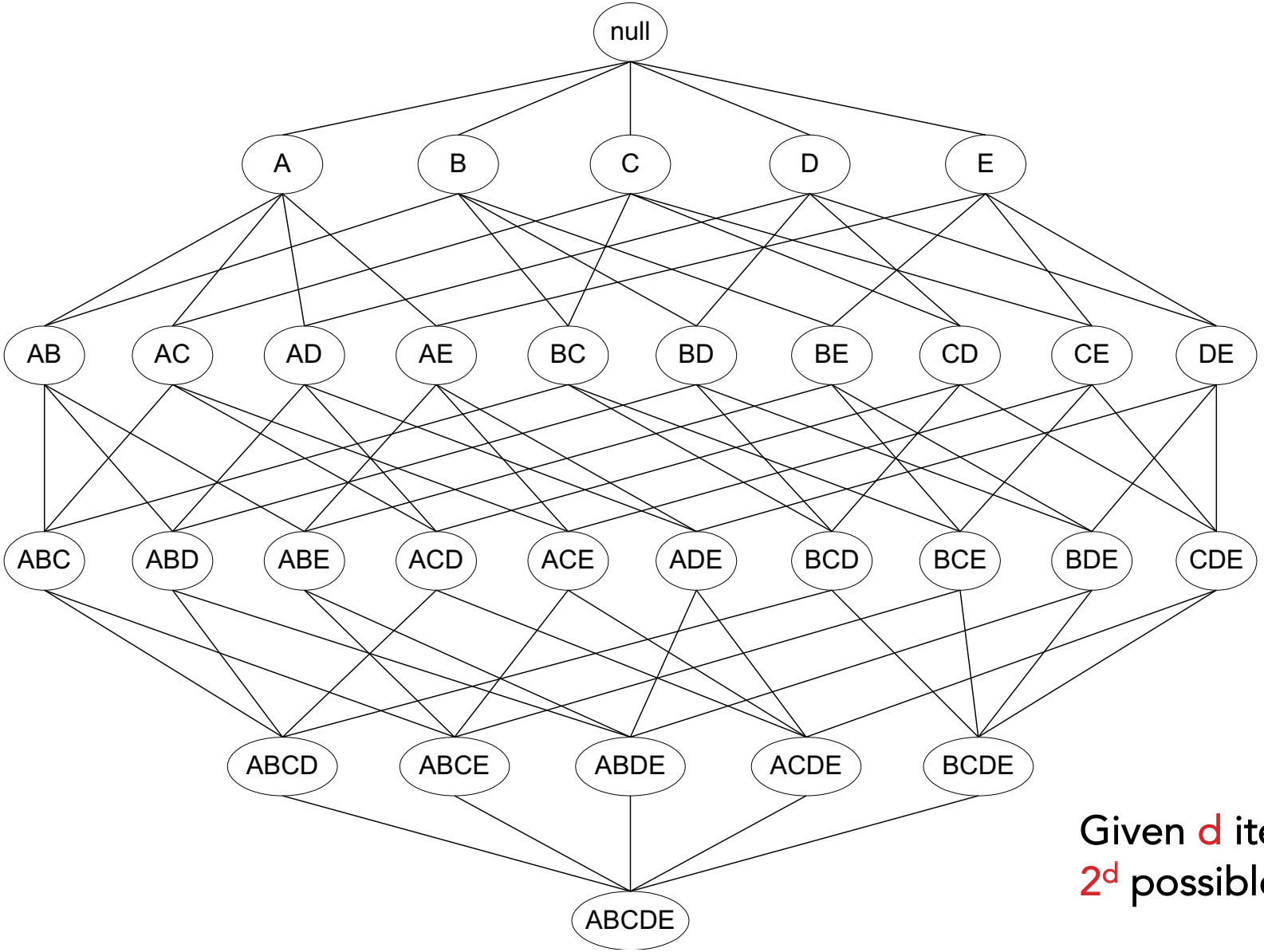
C) 3

D) 4

E) 5

F) 6

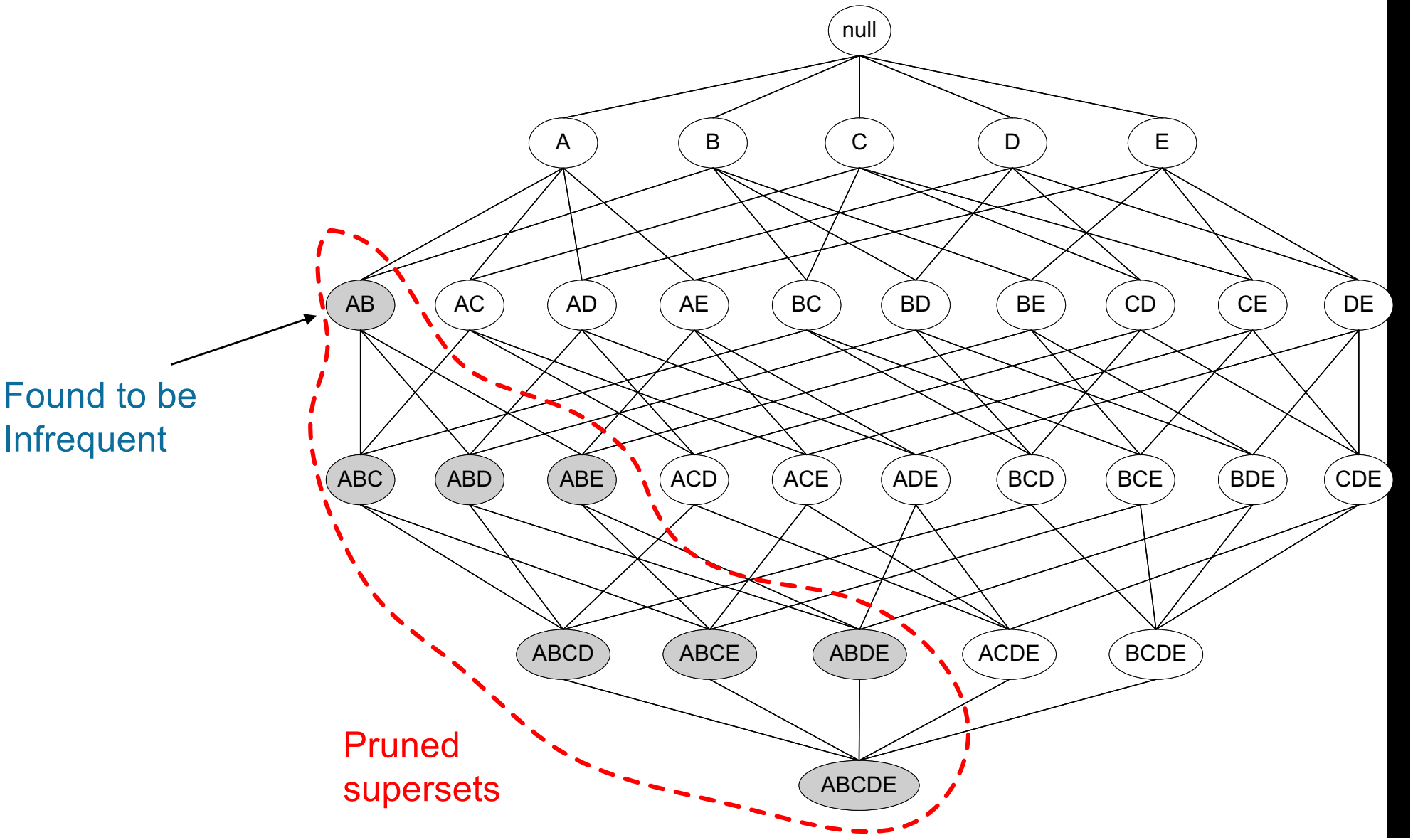
Expensive to Compute!



Given d items, there are 2^d possible itemsets

Can you think of a better algorithm?

Illustrating the Apriori principle



Reduce the number of candidates

- Apriori principle (Main observation):
  - If an itemset is frequent, then all of its subsets must also be frequent

- Apriori principle holds due to the following property of the support measure:

$$\forall X, Y : (X \subseteq Y) \Rightarrow s(X) \geq s(Y)$$

  - The support of an itemset ***never exceeds*** the support of its subsets
  - This is known as the *anti-monotone* property of support

Exploiting the Apriori principle

1. Find frequent 1-items and put them to L_k ($k=1$)
2. Use L_k to generate a collection of *candidate* itemsets C_{k+1} with size ($k+1$)
3. Scan the database to find which itemsets in C_{k+1} are frequent and put them into L_{k+1}
4. If L_{k+1} is not empty
    1. $k=k+1$
    2. Goto step 2

R. Agrawal, R. Srikant: "Fast Algorithms for Mining Association Rules", *Proc. of the 20th Int'l Conference on Very Large Databases*, 1994.

- Task 1: Methods for finding all frequent itemsets efficiently

- Task 2: Methods for finding association rules efficiently

Definition: Association Rule

- Association Rule
  - An implication expression of the form X → Y, where X and Y are non-overlapping itemsets

TID	Items
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

- Rule Evaluation Metrics
  - Support (s)
    - Fraction of transactions that contain both X and Y
  - Confidence (c)
    - Measures how often items in Y appear in transactions that contain X

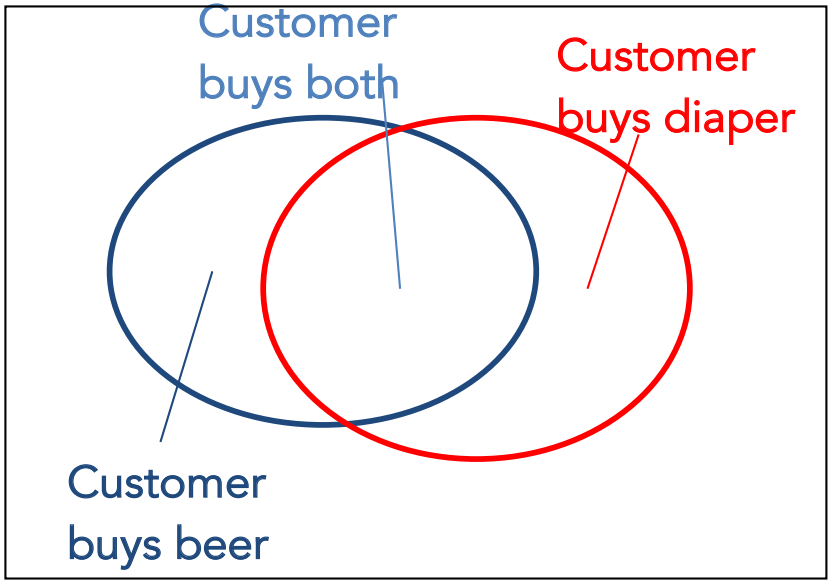
Example:

$$\{\text{Milk}, \text{Diaper}\} \rightarrow \text{Beer}$$

$$s = \frac{\sigma(\text{Milk}, \text{Diaper}, \text{Beer})}{|\text{T}|} = \frac{2}{5} = 0.4$$

$$c = \frac{\sigma(\text{Milk}, \text{Diaper}, \text{Beer})}{\sigma(\text{Milk}, \text{Diaper})} = \frac{2}{3} = 0.67$$

Rule Measures: Support and Confidence



Find all the rules $X \rightarrow Y$ with minimum confidence and support

- support, *s*, probability that a transaction contains {X and Y}
- confidence, *c*, ***conditional probability*** that a transaction having X also contains Y

TID	Items
100	A,B,C
200	A,C
300	A,D
400	B,E,F

Let minimum support 50%, and minimum confidence 50%, we have

- $A \rightarrow C$ (50%, 66.6%)
- $C \rightarrow A$ (50%, 100%)

CLICKER - HTTPS://CLICKER.CSAIL.MIT.EDU/6.S080/

TID	date	items_bought
100	10/10/99	{F,A,D,B}
200	15/10/99	{D,A,C,E,B}
300	19/10/99	{C,A,B,E}
400	20/10/99	{B,D}

What is the ***support*** and ***confidence*** of the rule: {B,D} → {A}

a) support = 75%, confidence = 66.6%

b) support = 50%, confidence = 66.6%

c) support = 50%, confidence = 75%

CLICKER - HTTPS://CLICKER.CSAIL.MIT.EDU/6.S080/

TID	date	items_bought
100	10/10/99	{F,A,D,B}
200	15/10/99	{D,A,C,E,B}
300	19/10/99	{C,A,B,E}
400	20/10/99	{B,D}

What is the ***support*** and ***confidence*** of the rule:

{B,D} → {A}

- Support:
  - percentage of tuples that contain {A,B,D} = 50%
- Confidence:

$$\frac{\text{number of tuples that contain } \{A,B,D\}}{\text{number of tuples that contain } \{B,D\}} = 66.6\%$$

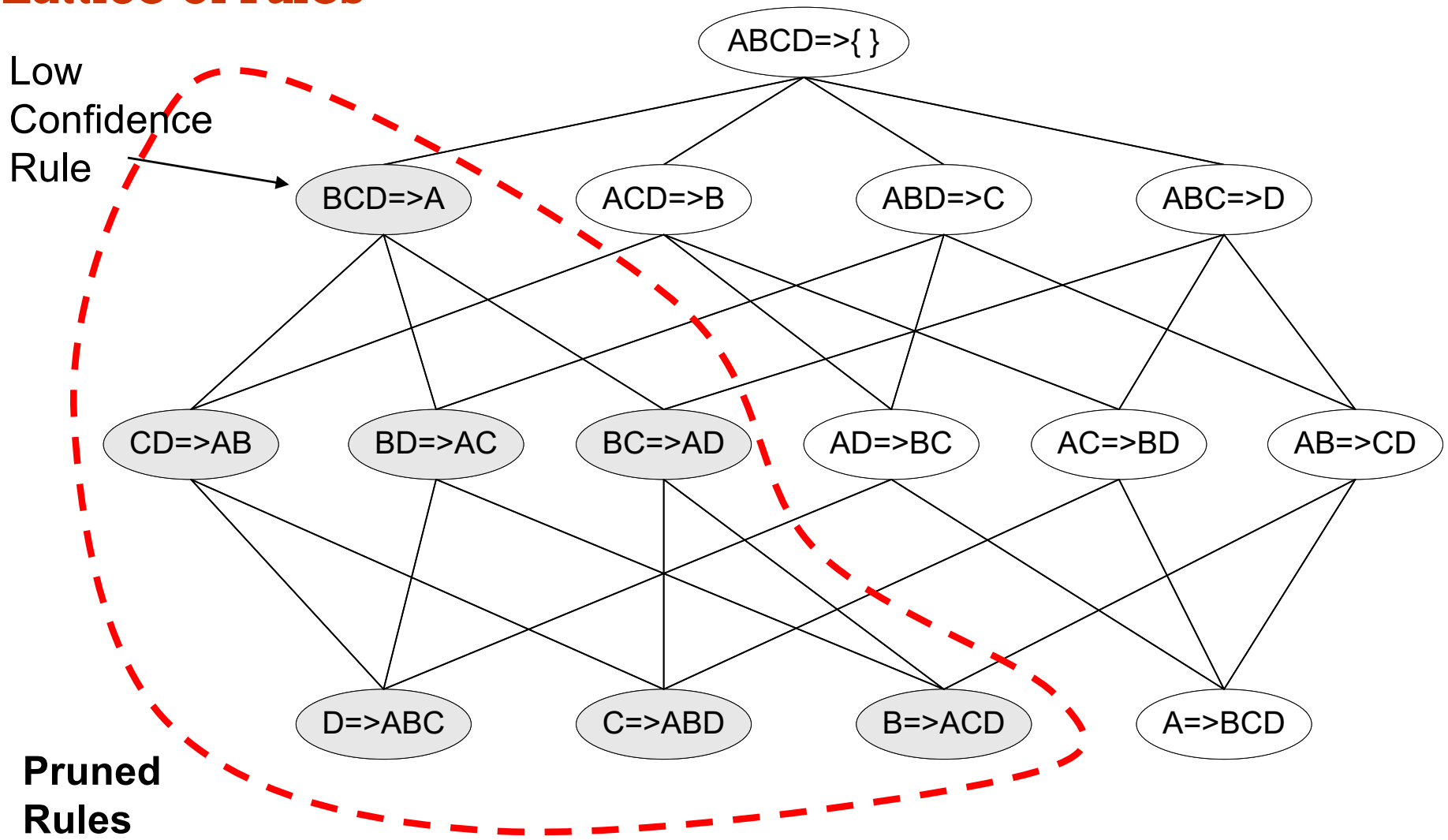
Association-rule mining task

Two-step approach:

- Frequent Itemset Generation
  - Generate all itemsets whose support >= minsup
- Rule Generation
  - Generate high confidence rules from each frequent itemset, where each rule is a binary partition of a frequent itemset

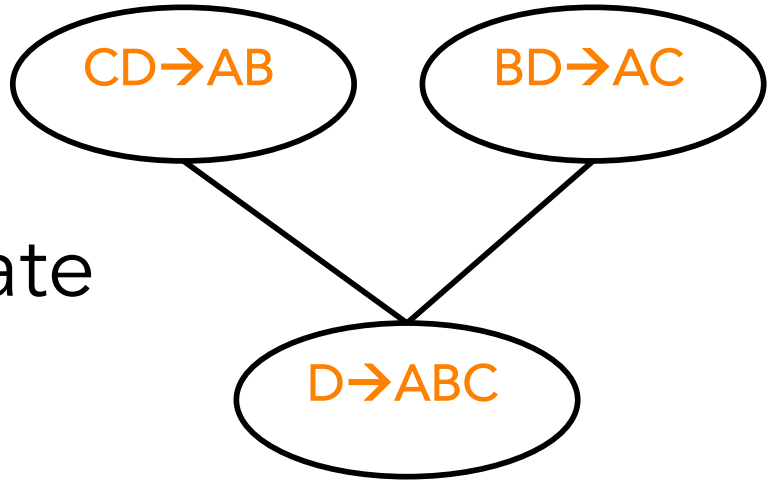
Rule Generation for Apriori Algorithm

Lattice of rules



Apriori algorithm for rule generation

- Candidate rule is generated by merging two rules that share the same prefix in the rule consequent



- join(CD→AB,BD—>AC) would produce the candidate rule D →ABC

- Prune rule D→ABC if there exists a subset (e.g., AD→BC) that does not have high confidence