# The Metric Space of Partitions and Its Applications in Data Mining

Dan Simovici

University of Massachusetts Boston,
Department of Computer Science,
Boston, Massachusetts 02125, USA

# Applications:

- Building better classifiers
- Better discretization algorithms
- Stable incremental clustering categorical data
- Metric study of genetic codes

# Collaborators:

Lucila Ohno-Machado
Szymon Jaroszewicz
Winston Kuo
Richard Butterworth
Namita Singla

# Metrics and Partitions

# Metrics

A metric on a set $S$ is a mapping $d : S \times S \longrightarrow \mathbb{R}$ that satisfies the following:

- $d(p, q) = 0$ if and only if $p = q$;
- $d(p, q) = d(q, p)$;
- $d(p, q) + d(q, r) \geq d(p, r)$,

for every $p, q, r \in S$.

# Popular Examples ...

- Standard distance on real line:

$$d(p,q) = |p - q|$$

- Minkowski distance in $\mathbb{R}^n$:

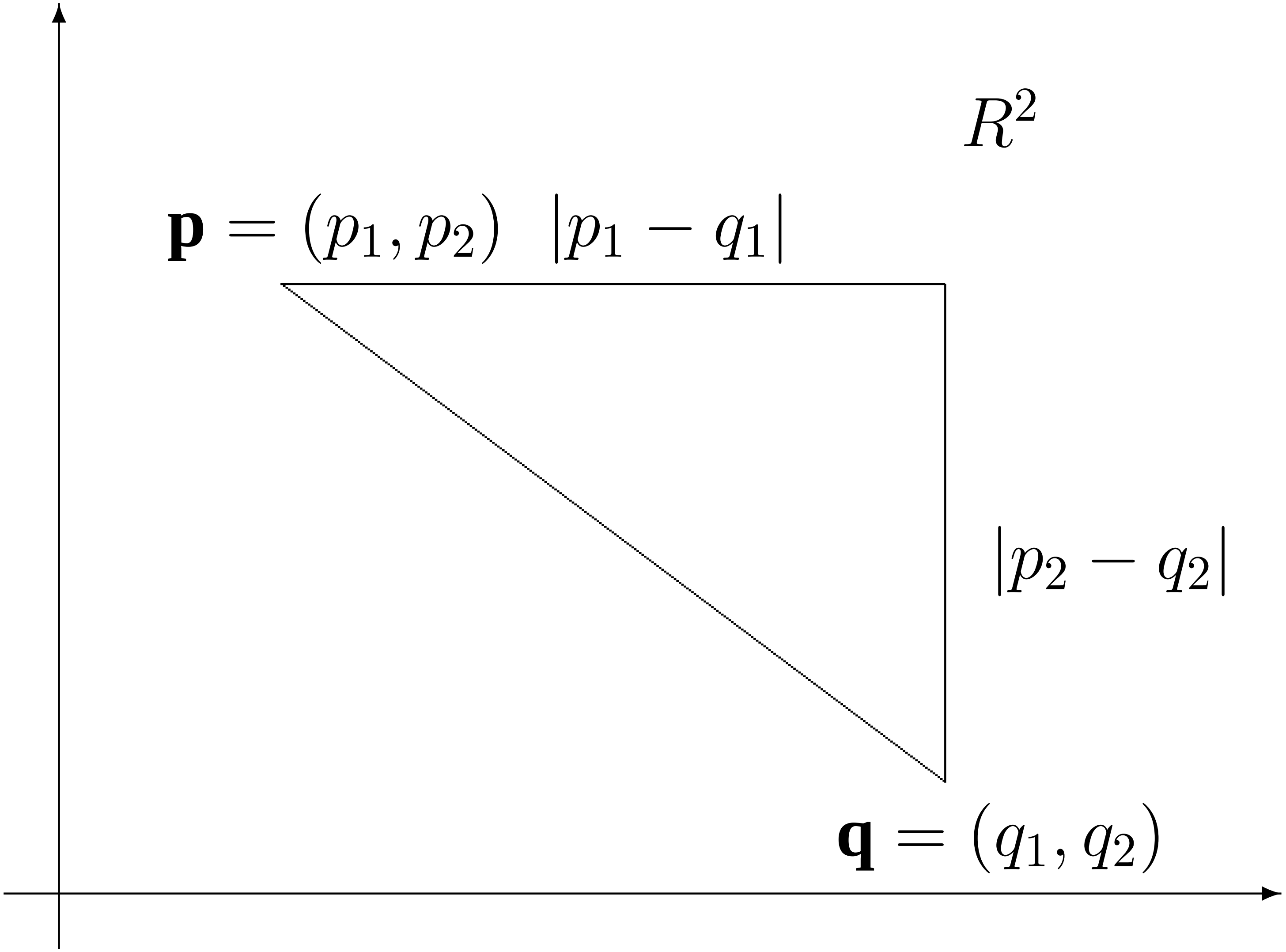
$$d_k(\mathbf{p},\mathbf{q}) = \left( \sum_{i=1}^{n} |p_i - q_i|^k \right)^{\frac{1}{k}}$$

for $\mathbf{p} = (p_1, \ldots, p_n)$ and $\mathbf{q} = (q_1, \ldots, q_n) \in \mathbb{R}^n$.

# Examples

In $\mathbb{R}^2$:

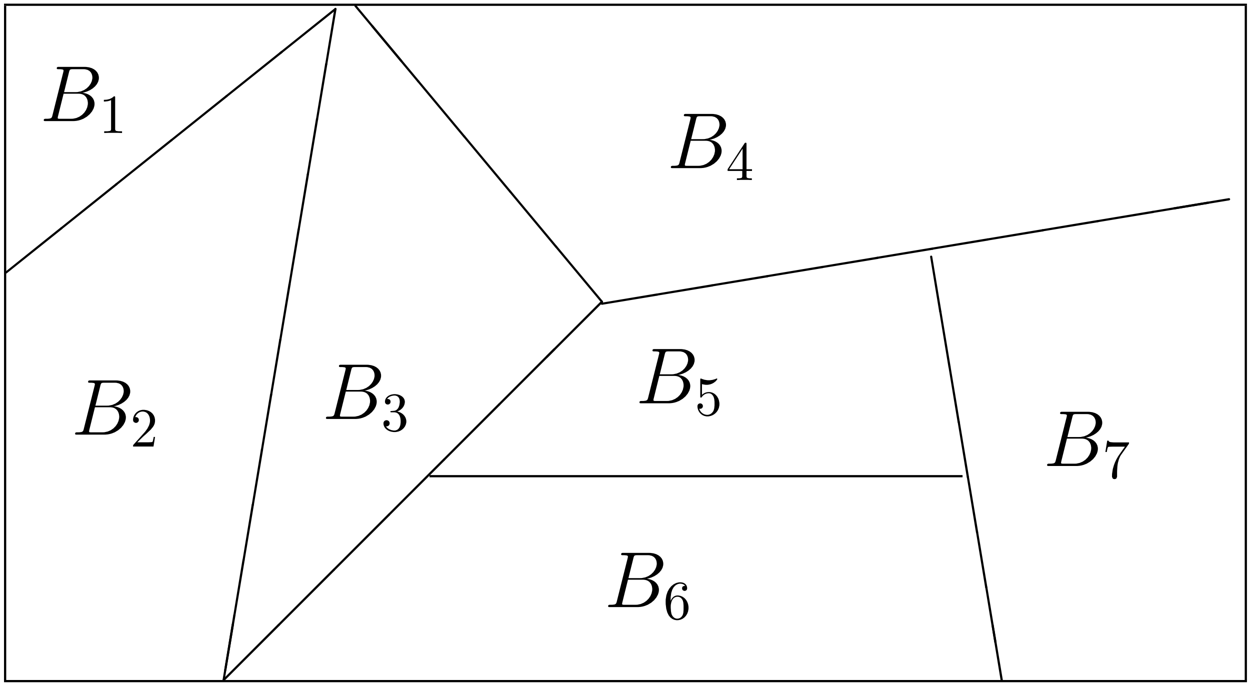
$$\begin{aligned}
d_1(\mathbf{p},\mathbf{q}) &= |p_1 - q_1| + |p_2 - q_2| \\
&\quad \text{(Manhattan distance)} \\
d_2(\mathbf{p},\mathbf{q}) &= \sqrt{|p_1 - q_1|^2 + |p_2 - q_2|^2} \\
&\quad \text{(Euclidean distance)} \\
d_\infty(\mathbf{p},\mathbf{q}) &= \lim_{k\to\infty} d_k(\mathbf{p},\mathbf{q}) \\
&= \max\{|p_1 - q_1|, |p_2 - q_2|\} \\
&\quad \text{(Canberra distance)}
\end{aligned}$$


$R^2$
$\mathbf{p} = (p_1, p_2)$
$|p_1 - q_1|$
$|p_2 - q_2|$
$\mathbf{q} = (q_1, q_2)$

# Partitions

PART$(S)$: set of partitions of set $S$

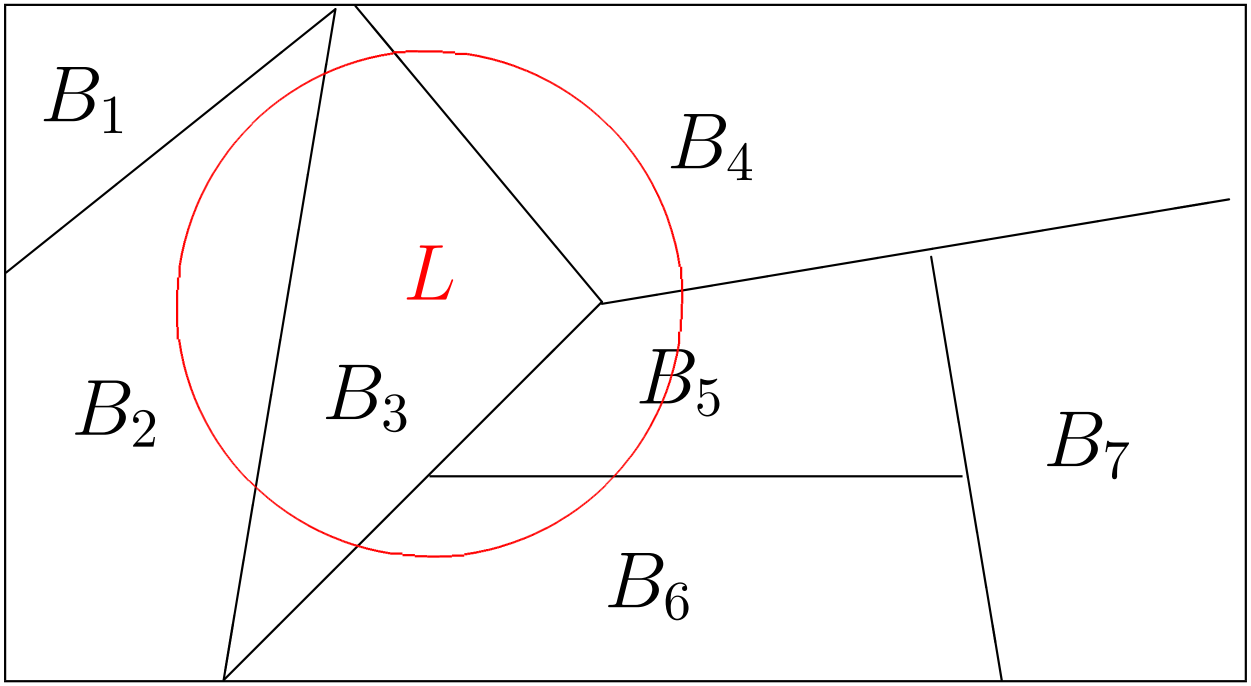
$$\text{Partition } \pi = \{B_1, \ldots, B_7\}$$



Let $L \subseteq S$ and $\pi = \{B_1, \ldots, B_n\}$. The *trace of the partition* $\pi$ on $L$ is:

$$\pi_L = \{B_i \cap L \,|\, 1 \leq i \leq k \text{ and } B_i \cap L \neq \emptyset\}.$$

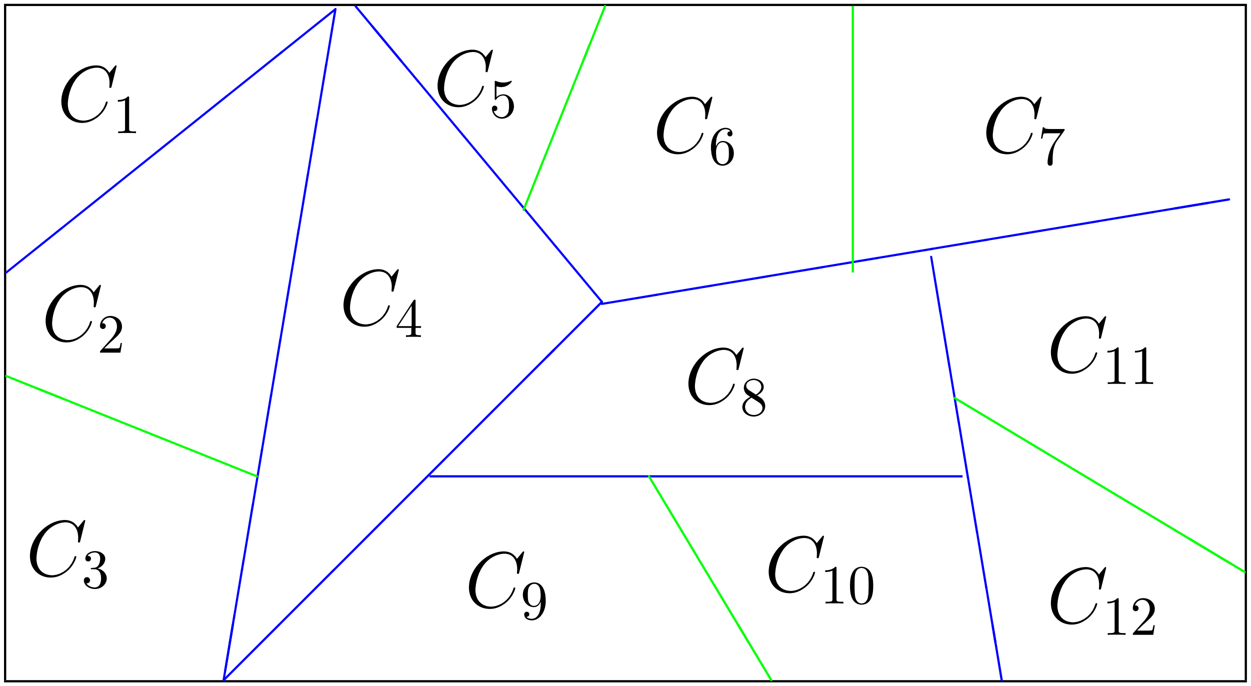
Trace of partition $\pi = \{B_1, \ldots, B_7\}$ on set $L$



# Partitions Partial Order

$\sigma \leq \pi$ if each block $C$ of $\sigma$ is included in a block of $\pi$.

Partition $\sigma = \{C_1, \ldots, C_{12}\} \leq \pi$



# Tables

A database table $\tau$
is a triple $\tau = (T, H, \rho)$
The header:
$H = A_1 \cdots A_n$
Dom$(A_i)$: domain of $A_i$

$T$

| | $A_1$ | $A_2$ | $\cdots$ | $A_n$ |
|---|---|---|---|---|
| $t_1$ | $a_{11}$ | $a_{12}$ | $\cdots$ | $a_{1n}$ |
| $t_2$ | $a_{21}$ | $a_{22}$ | $\cdots$ | $a_{2n}$ |
| $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ |
| $t_m$ | $a_{m1}$ | $a_{m2}$ | $\cdots$ | $a_{mn}$ |

The content of the table: $\rho = \{t_1, \ldots, t_m\}$ where $\rho \subseteq$ Dom$(A_1) \times \cdots \times$ Dom$(A_n)$.

# Partitions induced by Attribute Sets

Every attribute set $K \subseteq H$ induces a partition $\pi_K$: same as:

select K,count(K)
from T
group by K

$T$

| | $\cdots$ | $\longleftarrow K \longrightarrow$ | $\cdots$ |
|---|---|---|---|
| $t_1$ | $\cdots$ | $k_1$ | $\cdots$ |
| $t_2$ | $\cdots$ | $k_1$ | $\cdots$ |
| $t_3$ | $\cdots$ | $k_1$ | $\cdots$ |
| $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ |
| $t_l$ | $\cdots$ | $k_p$ | $\cdots$ |
| $t_{l+1}$ | $\cdots$ | $k_p$ | $\cdots$ |
| $t_{l+2}$ | $\cdots$ | $k_p$ | $\cdots$ |
| $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ |
| $t_{n-1}$ | $\cdots$ | $k_r$ | $\cdots$ |
| $t_n$ | $\cdots$ | $k_r$ | $\cdots$ |

# Shannon's Entropy

For random variables...

The Shannon entropy is introduced for a random variable distribution

$$X : \begin{pmatrix} x_1 & \cdots & x_n \\ p_1 & \cdots & p_n \end{pmatrix}$$

is $\mathcal{H}(X) = -\sum_{i=1}^{n} p_i \log_2 p_i$.

# Shannon entropy

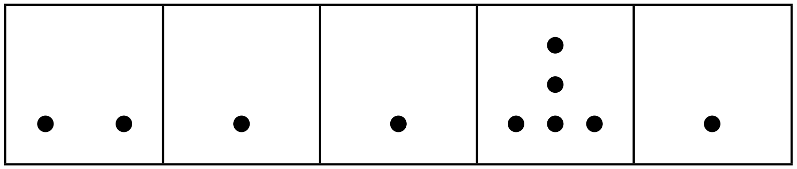
... for partitions

A partition $\pi = \{B_1, \ldots, B_m\}$ on a finite, nonempty set $A$ generates naturally a random variable:

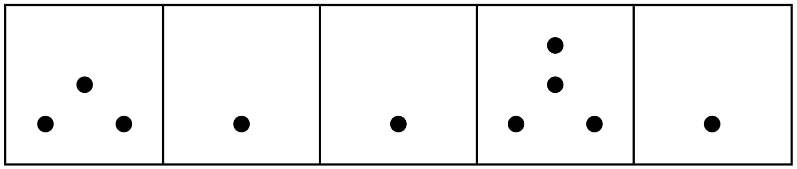
$$X_\pi : \begin{pmatrix} B_1 & \cdots & B_m \\ \frac{|B_1|}{|S|} & \cdots & \frac{|B_m|}{|S|} \end{pmatrix}$$

We define the Shannon entropy of $\pi$ as the Shannon entropy of $X_\pi$.

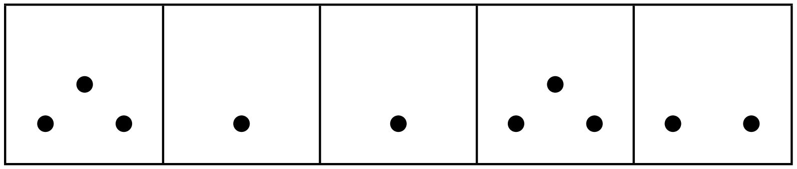
# Measuring concentration of values



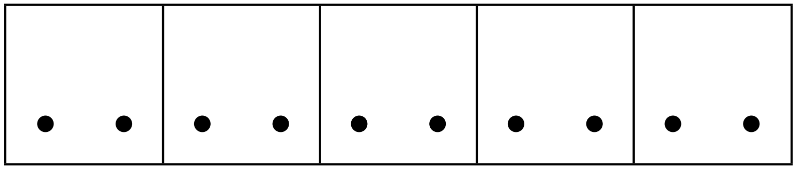
$$\mathcal{H}_1(\pi_4) = 1.9609$$



$$\mathcal{H}_1(\pi_3) = 2.0464$$



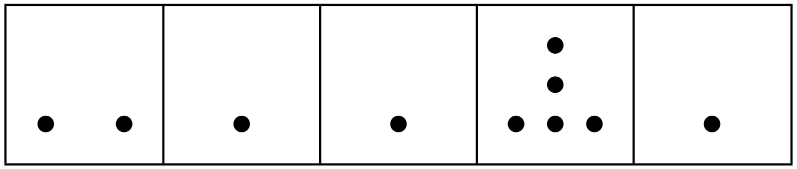
$$\mathcal{H}_1(\pi_2) = 2.1709$$



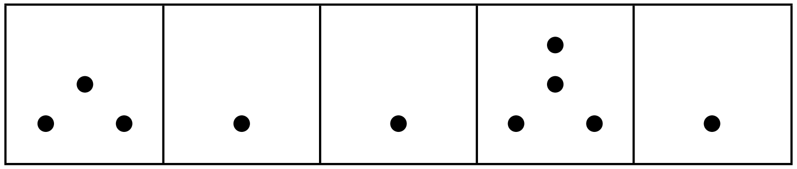
$$\mathcal{H}_1(\pi_1) = 2.3219$$

# Gini's Index

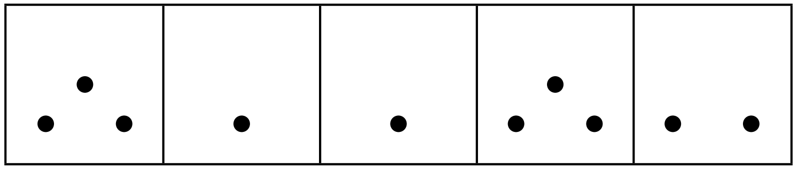
$$\mathcal{H}_2(\pi) = 1 - \sum_{i=1}^{n} p_i^2$$



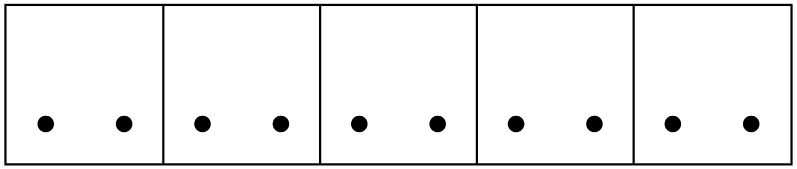
$\mathcal{H}_1(\pi_4) = 0.68$



$\mathcal{H}_1(\pi_3) = 0.72$



$\mathcal{H}_1(\pi_2) = 0.79$



$\mathcal{H}_1(\pi_1) = 0.80$

# Generalized Entropy of Partitions

Daróczy's $\beta$-generalized entropy of $\pi = \{B_1, \ldots, B_n\}$:

$$\mathcal{H}_\beta(\pi) = \frac{1}{1 - 2^{1-\beta}} \left( 1 - \sum_{i=1}^{n} \left( \frac{|B_i|}{|S|} \right)^\beta \right).$$

For $\beta = 2$ we obtain the Gini index. Also, $\lim_{\beta \to 1} \mathcal{H}_\beta(\pi)$ is Shannon's entropy

$$\mathcal{H}(\pi) = -\sum_{i=1}^{n} \frac{|B_i|}{|S|} \log_2 \frac{|B_i|}{|S|}$$

UMASS BOSTON

# Set Purity and Entropy

$\mathcal{H}(\pi_L)$ measures the impurity of the set $L$ relative to the partition $\pi$: the larger the entropy, the more $L$ is scattered among the blocks of $\pi$.

If $\pi, \sigma \in \mathsf{PART}(S)$, the average impurity of the blocks of $\sigma$ relative to $\pi$ is the *conditional entropy of* $\pi$ *relative to* $\sigma$:

$$\mathcal{H}(\pi|\sigma) = \sum_{j=1}^{m} \frac{|Q_j|}{|S|} \mathcal{H}(\pi_{Q_j}),$$

where $\sigma = \{Q_1, \ldots, Q_m\}$.

# Generalized Conditional Entropy

For $\pi, \sigma \in \mathsf{PART}(S)$ such that

$$\begin{aligned}\pi &= \{P_1, \ldots, P_k\}\\ \sigma &= \{Q_1, \ldots, Q_m\}\end{aligned}$$

the conditional $\beta$-entropy $\mathcal{H}_\beta(\pi|\sigma)$ is:

$$\mathcal{H}_\beta(\pi|\sigma) = \sum_{j=1}^{m} \left(\frac{|Q_j|}{|S|}\right)^\beta \mathcal{H}_\beta(\pi_{Q_j})$$
$$= \frac{1}{(2^{1-\beta}-1)|S|^\beta}\left(\sum_{i=1}^{k}\sum_{j=1}^{m} |P_i \cap Q_j|^\beta - \sum_{j=1}^{m} |Q_j|^\beta\right)$$

# Metrics on Partitions Sets

López de Mántaras:

$$d(\pi, \sigma) = \mathcal{H}(\pi|\sigma) + \mathcal{H}(\sigma|\pi)$$

Simovici and Jaroszewicz:

$$\begin{aligned} d_\beta(\pi, \sigma) &= \mathcal{H}_\beta(\pi|\sigma) + \mathcal{H}_\beta(\sigma|\pi) \\ &= \frac{1}{(2^{1-\beta}-1)|S|^\beta} \left(2 \cdot \sum_{i=1}^{k} \sum_{j=1}^{m} |P_i \cap Q_j|^\beta \right. \\ &\qquad \left. - \sum_{i=1}^{n} |P_i|^\beta - \sum_{j=1}^{m} |Q_j|^\beta \right). \end{aligned}$$

# Special Cases ...

De Mántaras' Metric:

$$\lim_{\beta \to 1} d_\beta(\pi, \sigma) = d(\pi, \sigma)$$

The $\beta = 2$ case:

$$\begin{aligned} d_2(\pi, \sigma) &= \frac{2}{\sqrt{|S|}} \Big( \textstyle\sum_{i=1}^{n} |P_i|^2 + \sum_{j=1}^{m} |Q_j|^2 \\ &\quad -2 \cdot \textstyle\sum_{i=1}^{k} \sum_{j=1}^{m} |P_i \cap Q_j|^2 \Big) \end{aligned}$$

# GK Classification Rule

Let $X$, $Y$ be two discrete random variables.

- $P(Y = b_j | X = a_i)$: the probability of predicting the value $b_j$ for $Y$ when $X = a_i$
  Classification rule: An event that has the component $X = a_i$ is classified in the $Y$-class $b_j$ if $j$ is the number for which $P(Y = b_j | X = a_i)$ has the largest value.
- The probability of misclassification:

$$1 - \max_{1 \leq j \leq k} P(Y = b_j | X = a_i).$$

# The Goodman-Kruskal Coefficient

The *Goodman-Kruskal coefficient* of $X$ and $Y$ is defined by

$$\begin{aligned}
&\mathsf{GK}(X,Y)\\
&= \sum_{i=1}^{l} P(X=a_i)\left(1 - \max_{1\le j\le k} P(Y=b_j|X=a_i)\right)\\
&= 1-\sum_{i=1}^{l} P(X=a_i)\max_{1\le j\le k} P(Y=b_j|X=a_i).
\end{aligned}$$

$\mathsf{GK}(X,Y)$ is the expected probability that in a randomly chosen case the value of $Y$ will be incorrectly predicted from $X$.
$\lambda_{Y|X}$ is the relative reduction in the probability of prediction error:

$$\lambda_{Y|X} = 1 - \frac{\mathsf{GK}(X,Y)}{1 - \max_{1 \leq j \leq k} P(Y = b_j)}$$

$\lambda_{Y|X}$ is the proportion of the relative error in predicting the value of $Y$ that can be eliminated by knowledge of the $X$-value.

# The Goodman-Kruskal Coefficient for Partitions

Consider two partitions

$$\pi = \{B_1, \ldots, B_l\} \text{ and } \sigma = \{C_1, \ldots, C_k\}.$$

*The Goodman-Kruskal coefficient* of $\pi, \sigma$:

$$\mathsf{GK}(\pi, \sigma) = 1 - \sum_{i=1}^{l} \max_{1 \leq j \leq k} \frac{|C_j \cap B_i|}{|S|}.$$

# Interpretation of GK

For a fixed $i$, the largest error in predicting $Y$ is:

$$1 - \max_{1 \leq j \leq k} P(Y = j | X = i) = 1 - \max_{1 \leq j \leq k} \frac{|C_j \cap B_i|}{|B_i|}.$$

Expected value of the largest error in predicting $Y$ is GK$(X, Y)$:

$$\begin{aligned} & \sum_{i=1}^{l} \frac{|B_i|}{|S|} \cdot \left(1 - \max_{1 \leq j \leq k} \frac{|C_j \cap B_i|}{|B_i|}\right) \\ = \; & 1 - \sum_{i=1}^{l} \max_{1 \leq j \leq k} \frac{|C_j \cap B_i|}{|S|}, \end{aligned}$$

# Properties of GK

- $\mathsf{GK}(\pi, \sigma) = 0$ if and only if $\pi \leq \sigma$.
- GK is monotonic in its first argument and dually monotonic in its second:
- GK satisfies a triangular inequality:

$$\mathsf{GK}(\pi, \sigma) \leq \mathsf{GK}(\pi, \tau) + \mathsf{GK}(\tau, \sigma).$$

# Metric Associated to GK

The Goodman-Kruskal coefficient generates a metric on $\mathsf{PART}(S)$.

Let $d_{GK} : \mathsf{PART}(S) \times \mathsf{PART}(S) \longrightarrow \mathbb{R}$ be

$$d_{GK}(\pi, \sigma) = \mathsf{GK}(\pi, \sigma) + \mathsf{GK}(\sigma, \pi).$$

for $\pi, \sigma \in \mathsf{PART}(S)$.

The function $d_{GK}$ is a metric on the set $\mathsf{PART}(S)$.

# Goodman-Kruskal Coefficient for Attribute Sets

Let $K, L$ be two sets of attributes of a table. Define $\mathsf{GK}(K, L) = \mathsf{GK}(\pi_K, \pi_L)$: the expected error that occurs when we try to predict the value of $t[L]$ from the value of $t[K]$.

- If $K_1 \subseteq K_2$, then $\pi_{K_2} \leq \pi_{K_1}$, so $\mathsf{GK}(K_2, L) \leq \mathsf{GK}(K_1, L)$.
- If $L_1 \subseteq L_2$, then $\mathsf{GK}(K, L_2) \leq \mathsf{GK}(K, L_1)$.

# Goodman-Kruskal Metric on Attribute Sets

$$d_{GK}(K, L) = d_{GK}(\pi_K, \pi_L)$$

The new metric can be used for:

- constructing classifiers;
- discretization of continuous attributes;
- attribute clustering, feature selection and data compression.

# Data Mining Applications

# Clustering Generic Codes

## A Proof-of-Concept Experiment

- Aminoacids in proteins are created according to a DNA blueprint, the *genetic code* (GC).
- Each GC is a function $c : \{A, G, C, T\}^3 \longrightarrow \mathcal{A} \cup \{\textbf{Ter}\}$; thus, each GC defines a partition on the set $\{A, G, C, T\}^3$.
- The NCBI site lists 16 genetic codes: 6 nuclear and 10 mitochondrial.

# An Example: The “Universal” GC

**Trp**
TGG

**Met**
ATG

| **Lys** | | **Phe** | **Pro** | **Ser** | **Ter** | **Thr** | | **Tyr** | **Val** | **Ile** |
|---|---|---|---|---|---|---|---|---|---|---|
| AAA<br>AAG | | TTT<br>TTC | CCT<br>CCC<br>CCA<br>CCG | TCT<br>TCC<br>TCA<br>TGG<br>AGT<br>AGC | TAA<br>TAG<br>TGA | ACT<br>ACC<br>ACA<br>ACG | | TAT<br>TAC | GTT<br>GTC<br>GTA<br>GTG | ATT<br>ATC<br>ATA |

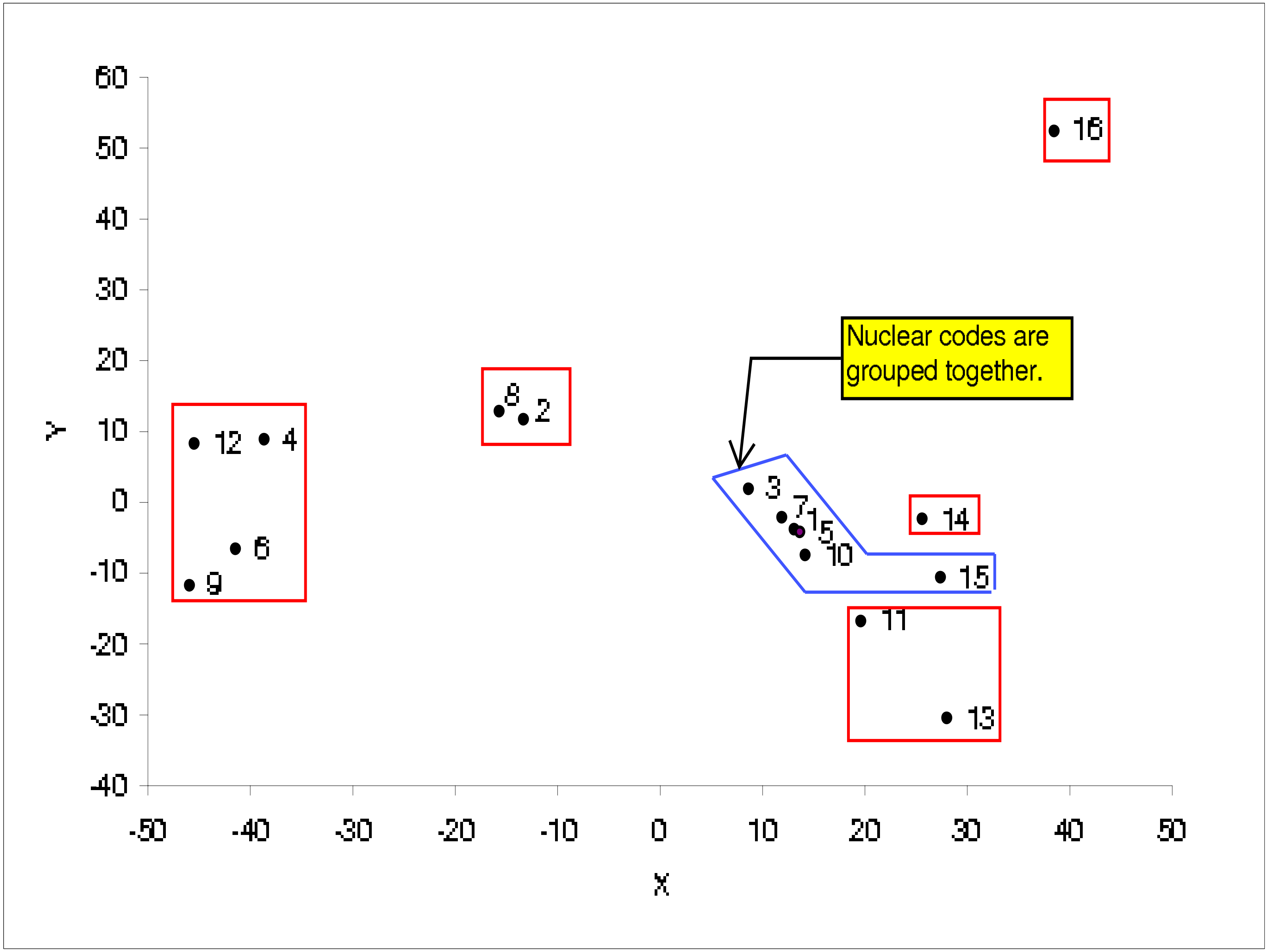
| **Ala** | **Arg** | **Asn** | **Asp** | **Cys** | **Gln** | **Glu** | **Gly** | **His** | **Leu** |
|---|---|---|---|---|---|---|---|---|---|
| GCT<br>GCC<br>GCA<br>GCG | CGT<br>CGC<br>CGA<br>CGG<br>AGA<br>AGG | AAT<br>AAC | GAT<br>GAC | TGT<br>TGC | CAA<br>CAG | GAA<br>GAG | GGT<br>GGC<br>GGA<br>GGG | CAT<br>CAC | TTA<br>TTG<br>CTT<br>CTC<br>CTA<br>CTG |

# Visualizing Genetic Codes

Visualization process:

- codes are viewed as partitions on the set of codons $\{A, G, C, T\}^3$;
- inter-code distances are computed using the entropic distance $d_2$;
- codes are represented as points in $\mathbb{R}^2$ using the “classical multidimensional scaling”.

# Scaling of Genetic Codes



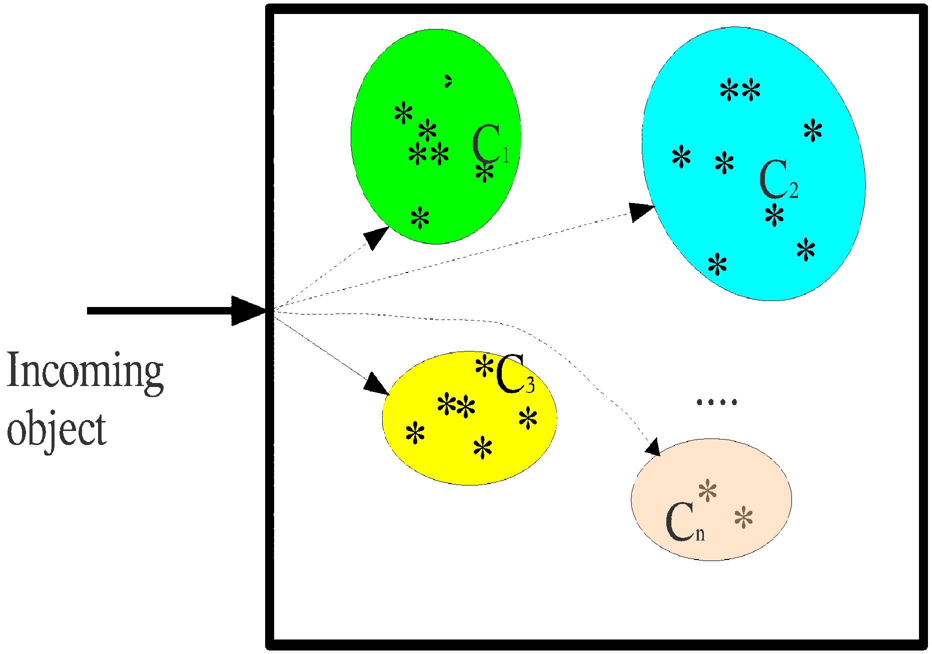
# Incremental Clustering

# Main focus

- Nominal data
- Incremental clustering

Main Feature of IC: Incremental clustering forms clusterings gradually by a sequential process of adding objects to clusters or initiating new clusters.

# Incremental Clustering



# The interest in incremental clustering

- Main memory usage is minimal.
- Algorithms are scalable with the size of the set of objects.

# Valuations and Metrics

- $v : \mathsf{PART}(S) \longleftarrow \mathbb{R}$ is $v(\pi) = \sum_{i=1}^{n} |B_i|^2$, where $\pi = \{B_1, \ldots, B_n\}$ is a lower valuation on $\mathsf{PART}(S)$:

$$v(\pi \vee \sigma) + v(\pi \wedge \sigma) \geq v(\pi) + v(\sigma) \qquad (1)$$

  for $\pi, \sigma \in \mathsf{PART}(S)$.

- For every lower valuation $v$, $d : (\mathsf{PART}(S))^2 \longleftarrow \mathbb{R}$ defined by $d(\pi, \sigma) = v(\pi) + v(\sigma) - 2v(\pi \wedge \sigma)$ is a metric on $\mathsf{PART}(S)$.

# Clusterings as Partitions

We seek a clustering $\kappa = \{C_1, \ldots, C_n\} \in \mathsf{PART}(S)$ such that the total distance from $\kappa$ to the partitions of the attributes:

$$D(\kappa) = \sum_{i=1}^{n} d(\kappa, \pi^{A_i})$$

is minimal.

# Distance between clustering and attribute partitions

$$d(\kappa, \pi^A) = \sum_{i=1}^{n} |C_i|^2 + \sum_{j=1}^{m_A} |B^A_{a_j}|^2 - 2 \sum_{i=1}^{n} \sum_{j=1}^{m_A} |C_i \cap B^A_{a_j}|^2,$$

# AMICA

A Metric Incremental Clustering Algorithm)

If $t \notin S$, and let $Z = S \cup \{t\}$. The following may occur:

1. the object $t$ is added to an existing cluster $C_k$, or
2. a new cluster, $C_{n+1}$ is created that consists only of $t$.

Relative to $\pi^A$, $t$ is added to the block $B^A_{t[A]}$.

# Object is added to existing cluster

$$\begin{aligned}
\kappa_{(k)} &= \{C_1, \ldots, C_{k-1}, C_k \cup \{t\}, C_{k+1}, \ldots, C_n\} \\
\pi^{A'} &= \{B^A_{a_1}, \ldots, B^A_{t[A]} \cup \{t\}, \ldots, B^A_{a_{m_A}}\}
\end{aligned}$$

$$\begin{aligned}
&d(\kappa_{(k)}, \pi^{A'}) - d(\kappa, \pi^A) \\
&\quad = (|C_k| + 1)^2 - |C_k|^2 + (|B^A_{t[A]}| + 1)^2 \\
&\qquad -|B^A_{t[A]}|^2 - 2(2|C_k \cap B^A_{t[A]}| + 1) \\
&\quad = 2|C_k| + 1 + 2|B^A_{t[A]}| + 1 - 4|C_k \cap B^A_{t[A]}| - 2 \\
&\quad = 2|C_k \oplus B^A_{t[A]}|.
\end{aligned}$$

The minimal increase of $d(\kappa_{(k)}, \pi^{A'})$ is given by:

# Object forms a new cluster

$$\begin{aligned} \kappa' &= \{C_1, \ldots, \ldots, C_n, \{t\}\} \\ \pi^{A'} &= \{B^A_{a_1}, \ldots, B^A_{t[A]} \cup \{t\}, \ldots, B^A_{a_{m_A}}\} \end{aligned}$$

$$d(\kappa', \pi^{A'}) - d(\kappa, \pi^A) = 2|B^A_{t[A]}|.$$

# Course of Action

$$D(\kappa') - D(\kappa) = \begin{cases} 2 \cdot \sum_A |C_k \oplus B^A_{t[A]}| & \text{in Case 1} \\ 2 \cdot \sum_A |B^A_{t[A]}| & \text{in Case 2.} \end{cases}$$

If $\min_k \sum_A |C_k \oplus B^A_{t[A]}| < \sum_A |B^A_{t[A]}|$ add $t$ to a cluster $C_k$ for which $\sum_A |C_k \oplus B^A_{t[A]}|$ is minimal; otherwise, create a new one-object cluster.

# Difficulties of IC

- Incremental clustering algorithms are affected, in general, by the order in which objects are processed by the clustering algorithm.
- Each such algorithm proceeds typically in a hill-climbing fashion that yields local minima rather than global ones.

# Limiting the Effect of Ordering

The “not-yet” technique introduced by Roure and Talavera:

In our framework: A new cluster is created only when

$$r(t) = \frac{\sum_A |B^A_{t[A]}|}{\min_k \sum_A |C_k \oplus B^A_{t[A]}|} < \alpha,$$

is satisfied, that is, only when the effect $r(t)$ of adding the object $t$ on the total distance is significant enough.

$\alpha \leq 1$ is a parameter provided by the user (no buffer if $\alpha = 1$

# The AMICA Algorithm:

**Input:** data set $S$ and threshold $\alpha$

**Output:** clustering $C_1, \ldots, C_{\mathsf{nc}}$

**Method:**

```
$\mathsf{nc} = 0; \ell = 1;$
while $S \neq \emptyset$ do
    select an object $t$; $S = S - \{t\}$;
    if $\quad_{A} |B^{A}_{t[A]}| \leq \alpha \min_{1 \leq k \leq \mathsf{nc}} \quad_{A} |C_k \oplus B^{A}_{t[A]}|$
      then
          $\mathsf{nc}$ ++; create a new single-object cluster $C_{\mathsf{nc}} = \{t\}$;
      else
          $r(t) = \quad_{A} |B^{A}_{t[A]}| / \min_{1 \leq k \leq \mathsf{nc}} \quad_{A} |C_k \oplus B^{A}_{t[A]}|$
    if $r(t) > 1$
      then $k = \mathtt{arg\ min}_k \quad_{A} |C_k \oplus B^{A}_{t[A]}|$
          add $t$ to cluster $C_k$;
      else /* this means $\alpha < r(t) \leq 1$ */
          place $t$ in NOT-YET buffer;
    end if;
endwhile;
process objects in the NOT-YET buffer as above with $\alpha = 1$;
```

# Experiments on Synthetic Data

- Synthetic data sets: produced by an algorithm that generates clusters of objects having real-numbered components grouped around a specified number of centroids.
- Data was discretized using a specified number of discretization intervals which allowed us to treat the data as nominal.
- The experiments were applied to several data sets with an increasing number of tuples and increased dimensionality and using several permutations of the set of objects.
- All experiments describe use $\alpha = 0.95$.

# Cluster Stability

- A data set that consists of 10,000 objects (grouped by the synthetic data algorithm around 6 centroids)
- A first pass of the algorithm produced 11 clusters.
- Most objects (9895) are concentrated in the top 6 clusters, a good approximation of the “natural” clusters produced by the synthetic algorithm.

# Insensitivity to Orderings

| Initial Run | | Random Permutation | | |
|---|---|---|---|---|
| Cluster | Size | Cluster | Size | Distribution (Original cluster) |
| 1 | 1548 | 1 | 1692 | 1692 (2) |
| 2 | 1693 | 2 | 1552 | 1548 (1), 3 (3), 1 (2) |
| 3 | 1655 | 3 | 1672 | 1672 (5) |
| 4 | 1711 | 4 | 1711 | 1711 (4) |
| 5 | 1672 | 5 | 1652 | 1652 (3) |
| 6 | 1616 | 6 | 1616 | 1616 (6) |
| 7 | 1 | 7 | 85 | 85 (8) |
| 8 | 85 | 8 | 10 | 10 (9) |
| 9 | 10 | 9 | 8 | 8 (10) |
| 10 | 8 | 10 | 1 | 1 (11) |
| 11 | 1 | 11 | 1 | 1 (7) |

# Scalability

| Number of objects | Time for 3 permutations (ms) | | | Average time (ms) |
|---|---|---|---|---|
| 2000 | 131 | 140 | 154 | 141.7 |
| 5000 | 410 | 381 | 432 | 407.7 |
| 10000 | 782 | 761 | 831 | 794.7 |
| 20000 | 1103 | 1148 | 1061 | 1104 |

# The Mushrooms Data Set

- The data set contains 8124 mushroom records and is typically used as test set for classification algorithms.
- Classifiers seek to predict the poisonous/edible character of the mushrooms.
- The class attribute (poisonous/edible) was removed and AMICA was applied to the remaining data set.

# Experimental Results

| Cl. | Poisonous/Edible | Total | Percentage of |
|---|---|---|---|
| num. | | | dominant group |
| 1 | 825/2752 | 3577 | 76.9% |
| 2 | 8/1050 | 1058 | 99.2% |
| 3 | 1304/0 | 1304 | 100% |
| 4 | 0/163 | 163 | 100% |
| 5 | 1735/28 | 1763 | 98.4% |
| 6 | 0/7 | 7 | 100% |
| 7 | 0/192 | 192 | 100% |
| 8 | 36/16 | 52 | 69% |
| 9 | 8/0 | 8 | 100% |

# Cluster Stability

| $C_i$ | Computed Clusters | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | First Random Permutation | | | | | | | | | |
| | $C'_1$ | $C'_2$ | $C'_3$ | $C'_4$ | $C'_5$ | $C'_6$ | $C'_7$ | $C'_8$ | $C'_9$ | $C'_{10}$ |
| | 3540 | 1797 | 1095 | 192 | 1296 | 8 | 36 | 7 | 137 | 16 |
| 3577 | 3540 | 0 | 37 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1058 | 0 | 0 | 1058 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1304 | 0 | 8 | 0 | 0 | 1296 | 0 | 0 | 0 | 0 | 0 |
| 163 | 0 | 26 | 0 | 0 | 0 | 0 | 0 | 0 | 137 | 0 |
| 1763 | 0 | 1763 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 7 | 0 | 0 |
| 192 | 0 | 0 | 0 | 192 | 0 | 0 | 0 | 0 | 0 | 0 |
| 52 | 0 | 0 | 0 | 0 | 0 | 0 | 36 | 0 | 0 | 16 |
| 8 | 0 | 0 | 0 | 0 | 0 | 8 | 0 | 0 | 0 | 0 |

# Analysis of Microarray Data

# $\epsilon$-predictors

An $\epsilon$-*predictor* for a set of attributes $Y$ is a set of attributes $K$ such that $\mathsf{GK}(K, Y) \leq \epsilon$.

- If $K$ is an $\epsilon$-predictor for $Y$, then any superset $K'$ of $K$ is also a $\epsilon$-predictor for $Y$.
- An $\epsilon$-predictor such that no of its proper subsets is an $\epsilon$-predictor is called *minimal*.

# An Algorithm for $\epsilon$-predictors

**Input:** Set of attributes $H$,
a target attribute $Y$, $Y \not\in H$ and an error level $\epsilon$.
**Output:** Set $P$ of all minimal $\epsilon$-predictors from $H$.

```
(1)   Cand = $\{\{A\} : A \in H\}$;
(2)   $P$ = $\emptyset$;
(3)   P = $\mathsf{P} \cup \{K \in \mathsf{Cand} : \mathsf{GK}(K, Y) \leq \epsilon\}$;
(4)   Cand = $\mathsf{Cand} \setminus \mathsf{P}$;
(5)   Cand = $\{L \subseteq H :$ for all $K \subset L$,
             $|K| = |L| - 1$ we have $K \in \mathsf{Cand}\}$;
(6)   goto (3);
```

- If a set is a non-minimal predictor, so are all of its supersets, which can thus be skipped.
- Initialize candidate set of predictors Cand to include one-set attributes.
- The set of minimal predictors P is constructed starting from Cand.

- Initialize P to include all singleton predictors whose error is below the threshold $\epsilon$. Remove those from C and the search for minimal two-attribute predictors makes use of the remaining candidate attributes, etc.
- The stopping condition could be exceeding the maximum predictor size or finding a predictor with desired prediction error.

# Experimental Results – KHAN

J. Khan et.al.: Classification and Diagnostic Prediction of Cancers using gene expression profiling and artificial neural networks, Nature Medicine, vol 7., 2001

Differential diagnosis of four small round blue cell tumors of childhood (SRBCTs) :

| | |
|---|---|
| NB: | neuroblastoma |
| RMS: | rhabdomyosarcoma |
| BL: | Burkitt lymphoma |
| EWS: | Ewing family of sarcomas |

# Previous work:

single layer neural networks (Khan)
logistic regression model (Weber)
SVMs (Mukerjee)
combined classifiers (Yeo)

# Khan Data

- 2308 genes were measured using cDNA microarrays
- Training Data: 63 cases (12 NB, 20 RMS, 8 BL, and 23 EWS)
- Test Data: 25 cases (6 EWS, 5 RMS, 6 NB, 3 BL, and 5 non-SRBCTs)
- The test cases include 5 cases which do not belong to any of the predicted SRBCT types. Such cases are not present in the training set.

# Preprocessing

Replace each class attribute with $4$ binary attributes, one for each cancer type.

| original attribute | computed attributes | | | |
|---|---|---|---|---|
| Cancer type | NB | RMS | BL | EWS |
| NB | 1 | 0 | 0 | 0 |
| EWS | 0 | 0 | 0 | 1 |
| RMS | 0 | 1 | 0 | 0 |
| other | 0 | 0 | 0 | 0 |

- A separate predictor is built for each binary attribute to allow for handling of cases of type ‘other’ present in the test set, but absent in the training set.
- We expect that for ‘other’ cancer type all of the predictors will give the value of $0$ thus indicated that none of the $4$ cancer types is present.

- Predictors may contradict each other (infrequently, because low error rate of individual classifiers).
- If presence of more than one cancer type is predicted consider it misclassified.
- Small predictors decrease the risk of overfitting (small number of training cases!)

# Limitations on the Computation

- We find all predictors with $1$ or $2$ attributes, allowing up to one misclassified instance on the training set.
- The stopping rule: reaching the maximum prescribed size of the predictor, or obtaining an error rate less than to $\frac{1}{t}$, where $t$ is the size of the training set.
- All but $30$ most predictive attributes are discarded.
- For each cancer type the first predictor with minimal training error is manually picked at random (without looking at its test set performance to avoid bias in the choice).

| Cancer type | selected predictor | image ids | mtr | mte | 1GP | 2GP |
|---|---|---|---|---|---|---|
| BL | WAS $\leq 0.69 \Rightarrow$ BL | 236282 | 0 | 1 | 15 | 5 |
| EWS | FCGRT $\leq 1.59 \Rightarrow$ EWS | 770394 | 1 | 3 | 2 | 10 |
| NB | MAP1B $> 2.17$<br>or RCV1 $> 1.98 \Rightarrow$ NB | 629896 - 383188 | 0 | 0 | 2 | 28 |
| RMS | TNNT2 $> 0.55$<br>or SGCA $> 0.44 \Rightarrow$ RMS | 298062 - 796258 | 0 | 2 | 0 | 25 |

Legend:

| | |
|---|---|
| mte | misclassified cases in test set |
| mtr | misclassified cases in training set |
| 1GP | number of one-gene predictors |
| 2GP | number of two-gene predictors |

- A fairly large number (12–30) of very simple predictors have been found for each cancer type.
- Each of those predictors has very good classification rate on the training set: up to one misclassified case is allowed.
- The results show that there are many genes based on which a diagnosis can be made for each cancer type.
- All genes except for the one that predicts BL were reported among the 96 selected in Khan.

- If a classifier for only one type of tumor gave a positive prediction, then the instance was classified as this type of tumor.
- If none of them gave positive prediction we declared the case as ‘other tumor type’.
- If more than one classifier was active the case was considered a prediction error.
- The combined classifier used a total of $6$ genes and classified correctly $19$ out of $25$ test cases.
- Out of the $6$ misclassified cases, $2$ gave classifications when the real outcome was ‘other’, $3$ SRBCT cases were undetected, and there was $1$ conflict.

# Experimental Results - GOLUB

- Training data: 38 cases (27 acute lymphoblastic leukemia and 11 acute myelocytic leukemia)
  Test data: 34 cases (20 ALL and 14 AML);
- Data involves 6817 genes.
- We discretized the gene expression levels using Fayyad-Irani
- 20 genes were retained for which the Goodman-Kruskal coefficient was below 0.04.

- Five single-genes predictors and 66 two-gene predictors were identified.
- We identified two two-genes predictors (MGST1, APLP2 and CD33, CystatinA) for which the errors on the test set are $0$ and $0.0294118$, respectively.
- CD33 was among the 50 genes selected by Golub et al.

# Voting Mechanism

- We retained 19 one-attribute predictors whose prediction error on the training set did not exceed 5.3% (that is, two errors out of the 38 training cases).
- A vote was taken, and the instance was classified according to the majority vote.
- We obtained 3 errors on the test set of 34 cases. Namely, the errors occurred on the 57th, 60th and 66th cases of the original Golub test set ("unclassifiable" in the original study (Golub)).

- The Goodman-Kruskal dissimilarity GK is a simple, but powerful measure of predictive power that can be used to produce robust classifiers.
- The small number of training cases makes reliable construction of more complex models like Bayesian networks or C4.5 trees very hard or even impossible.
- Naive Bayesian classifiers suffer from independence assumptions which may not be satisfied in the microarray setting where most genes are correlated with each other.

# A New Metric Discretization Algorithm

# From numerical to nominal

Previous work on discretization:

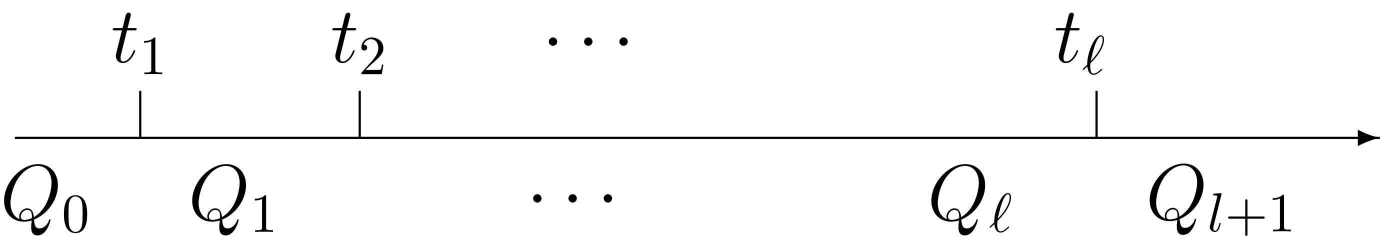
- fixed $k$-interval discretization (J. Dougherty, R. Kohavi, M. Sahami, 1995)
- fuzzy discretization (Kononenko 1992-1993)
- Shannon-entropy discretization (Fayyad and Irani, 1993)
- proportional $k$-interval discretization (Yang and Web, 2001, 2003)
- highly dependent attributes (M. Robnik and I. Kononenko, 1995)

# Basic Results

- a generalization of Fayyad-Irani discretization technique
- a geometric criterion for halting the discretization process
- better results in building
  - naive Bayes classifiers
  - decision trees

# Discretization of a numeric attribute $B$

Set of cutpoints: $S = \{t_1, \ldots, t_\ell\}$ in $\mathsf{aDom}(B)$, where $t_1 < t_2 < \cdots < t_\ell$.



Discretization partition of $\mathsf{aDom}(B)$:

$$\pi^S = \{Q_0, \ldots, Q_\ell\}$$

# Boundary Points

$t_1, \ldots, t_n$: the list of tuples sorted on the values of an attribute $B$.

$\pi_{B,A}$ is the partition of $\mathsf{aDom}(B)$ that consists of the longest runs of *consecutive* $B$-components of the tuples in this list that belong to the *same block* $K$ of the partition $\pi_A$.

The *boundary points* of the partition $\pi_{B,A}$ are the least and the largest elements of each of the blocks of the partition $\pi_{B,A}$.

We have $\pi_{B,A*} \leq \pi_A$ for any attribute $B$.

# Main Result

Theorem: Let $\beta \in (1, 2]$.
If $S$ is a set of cutpoints such that the distance $d_\beta(\pi_A, \pi_*^S)$ is minimal among the set of cutpoints with the same number of elements, then $S$ consists of boundary points of the partition $\pi_{B,A}$ of $\mathsf{aDom}(B)$.

To discretize $\mathsf{aDom}(B)$ we seek a set of cutpoints such that

$$d_\beta(\pi_A, \pi_*^S) = \mathcal{H}_\beta(\pi_A|\pi_*^S) + \mathcal{H}_\beta(\pi_*^S|\pi_A)$$

is minimal.
Seek a set of cutpoints $S$ such that the partition $\pi_*^S$ induced on the set of rows is as close as possible to the target partition $\pi_A$.

# Discretization Algorithm

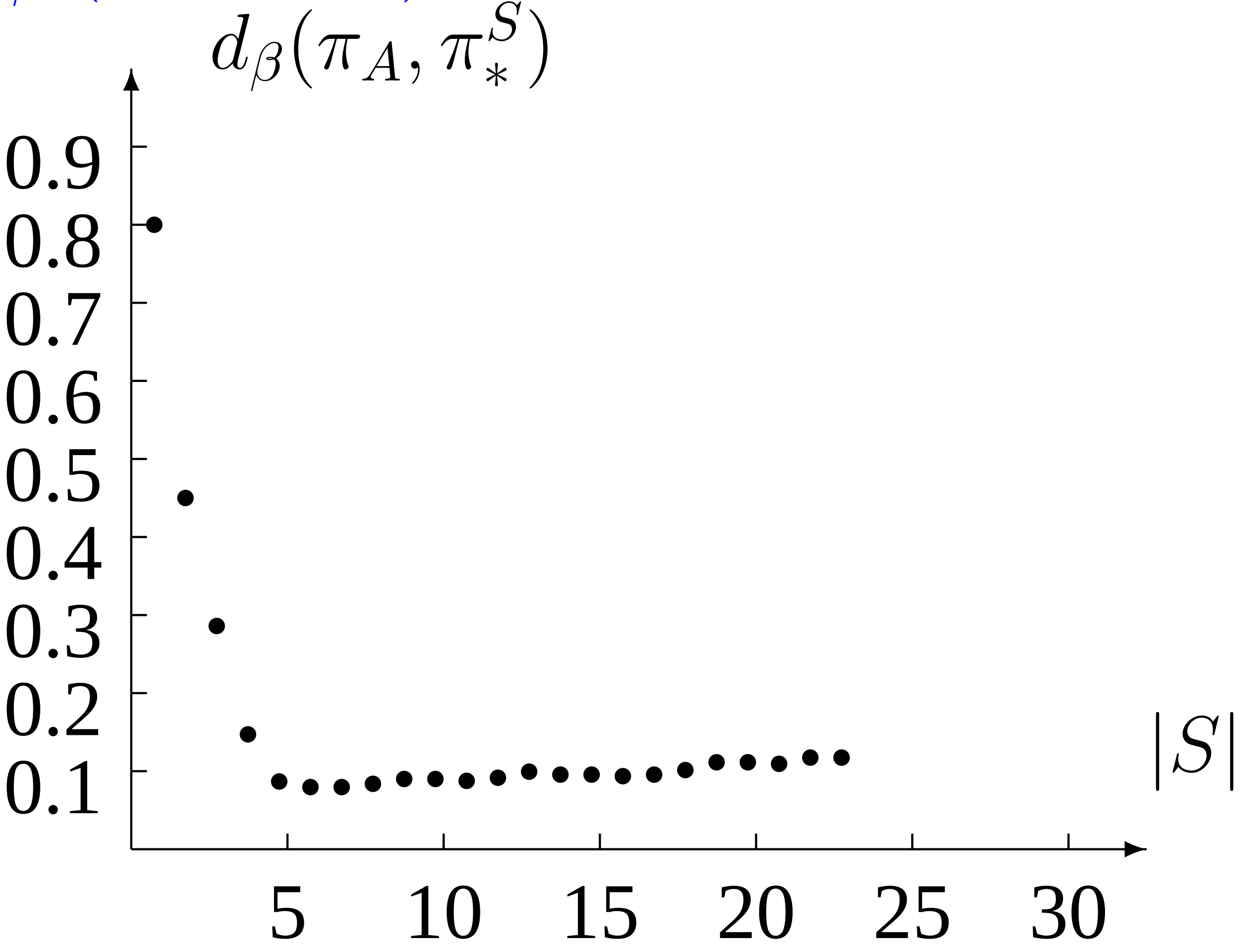
**Input:** `A table` $T$`, a class attribute` $A$
`and a real-valued attribute` $B$`.`
**Output:** `A discretized attribute` $B$`.`

BP is the set of boundary points of partition $\pi_{B,A*}$

# Method:

```
sort $T$ on $B$;
compute BP;
$S = \emptyset$; $d = \infty$;
while BP $\neq \emptyset$ do
        let $t = \arg\min_{t \in \text{BP}} d_\beta(\pi_A, \pi_*^{S \cup \{t\}})$;
        if $d \geq d_\beta(\pi_A, \pi_*^{S \cup \{t\}})$ then
            begin
                $S = S \cup \{t\}$; BP $=$ BP $- \{t\}$;
                $d = d_\beta(\pi_A, \pi_*^S)$
            end
        else exit while loop;
end while
for $\pi_*^S = \{Q_0, \ldots, Q_\ell\}$ replace
every value in $Q_i$ by $i$ for $0 \leq i \leq \ell$.
```

# $d_\beta(\pi_A, \pi_*^S)$ as a function of $|S|$



78% of the total time is spent on decreasing the distance by the last 1%

$$d_\beta(\pi_A, \pi_*^S) = \mathcal{H}_\beta(\pi_A|\pi_*^S) + \mathcal{H}_\beta(\pi_*^S|\pi_A)$$

If $S \subseteq S'$ then $\pi^S \geq \pi^{S'}$ and

$$\begin{aligned} \mathcal{H}_\beta(\pi_A|\pi_*^S) &\geq \mathcal{H}_\beta(\pi_A|\pi_*^{S'}) \\ \mathcal{H}_\beta(\pi_*^S|\pi_A) &\leq \mathcal{H}_\beta(\pi_*^{S'}|\pi_A). \end{aligned}$$

Process starts with $S = \emptyset$, so $\pi_*^S = \omega$.
Practical halting criterion:

$$|d - d_\beta(\pi_A, \pi_*^{S\cup\{t\}})| > 0.01d.$$

# Experimental Results

- Accuracy measured in stratified 10-fold cross-validation
- UCI datasets with $\beta \in \{1.5, 1.8.1.9, 2\}$

# Experimental Results - I

heart-c:

| Method | Size | Leaves | Accuracy |
|---|---|---|---|
| standard | 51 | 30 | 79.20 |
| $\beta = 1.5$ | 20 | 14 | 77.36 |
| $\beta = 1.8$ | 28 | 18 | 77.36 |
| $\beta = 1.9$ | 35 | 22 | 76.01 |
| $\beta = 2.0$ | 54 | 32 | 76.01 |

glass:

| | | | |
|---|---|---|---|
| standard | 57 | 30 | 57.28 |
| $\beta = 1.5$ | 32 | 24 | 71.02 |
| $\beta = 1.8$ | 56 | 50 | 77.10 |
| $\beta = 1.9$ | 64 | 58 | 67.57 |
| $\beta = 2.0$ | 92 | 82 | 66.35 |

# Experimental Results - II

ionosphere:

| | | | |
|---|---|---|---|
| standard | 35 | 18 | 90.88 |
| $\beta = 1.5$ | 15 | 8 | 95.44 |
| $\beta = 1.8$ | 19 | 12 | 88.31 |
| $\beta = 1.9$ | 15 | 10 | 90.02 |
| $\beta = 2.0$ | 15 | 10 | 90.02 |

iris:

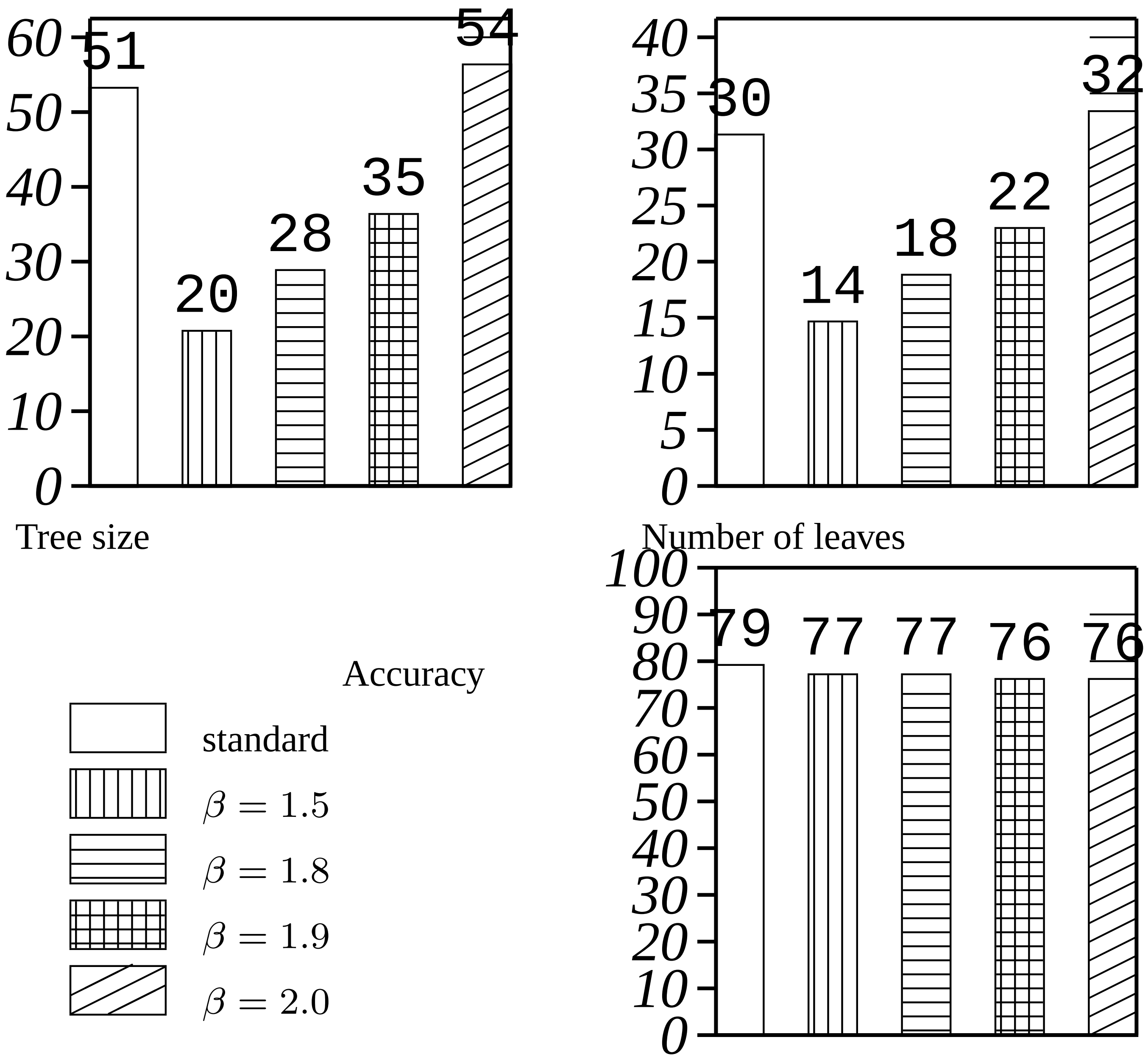
| | | | |
|---|---|---|---|
| standard | 9 | 5 | 95.33 |
| $\beta = 1.5$ | 7 | 5 | 96 |
| $\beta = 1.8$ | 7 | 5 | 96 |
| $\beta = 1.9$ | 7 | 5 | 96 |
| $\beta = 2.0$ | 7 | 5 | 96 |

# Experimental Results - III

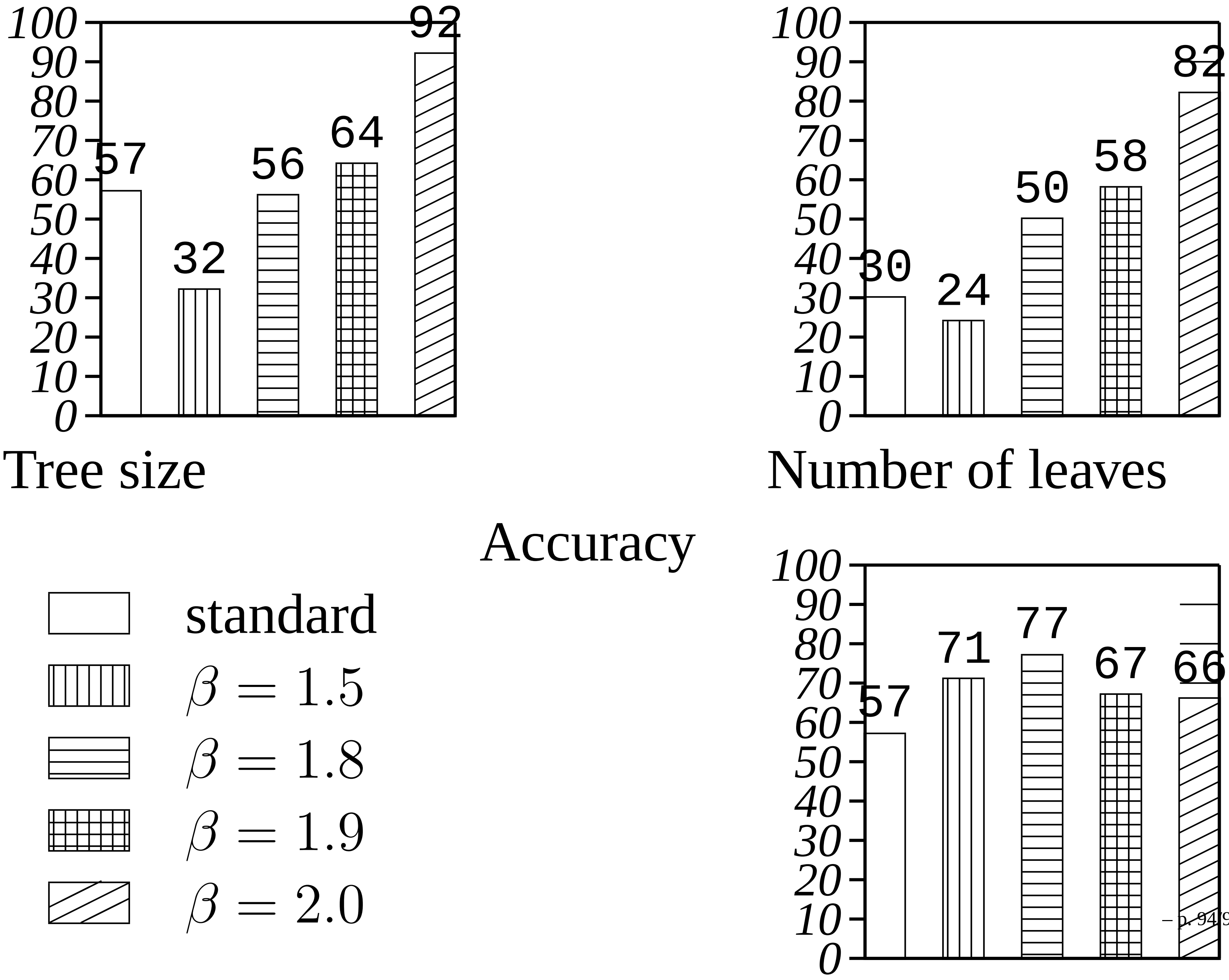
diabetes:

| | | | |
|---|---|---|---|
| standard | 43 | 22 | 74.08 |
| $\beta = 1.8$ | 5 | 3 | 75.78 |
| $\beta = 1.9$ | 7 | 4 | 75.39 |
| $\beta = 2.0$ | 14 | 10 | 76.30 |

# Heart-c


Tree size
0
10
20
30
40
50
60
51
20
28
35
54
Number of leaves
0
5
10
15
20
25
30
35
40
30
14
18
22
32
Accuracy
0
10
20
30
40
50
60
70
80
90
100
79
77
77
76
76
standard
$\beta = 1.5$
$\beta = 1.8$
$\beta = 1.9$
$\beta = 2.0$

# Glass



Tree size

Number of leaves

Accuracy

# Naive Bayes Classifiers

Error Rate

| Discretization Method | Diabetes | Glass | Ionosphere | Iris |
|---|---|---|---|---|
| $\beta = 1.5$ | 34.9 | 25.2 | 4.8 | 2.7 |
| $\beta = 1.8$ | 24.2 | 22.4 | 8.3 | 4 |
| $\beta = 1.9$ | 24.9 | 23.4 | 8.5 | 4 |
| $\beta = 2.0$ | 25.4 | 24.3 | 9.1 | 4.7 |
| weighted prop | 25.5 | 38.4 | 10.3 | 6.9 |
| prop. | 26.3 | 33.6 | 10.4 | 7.5 |

An appropriate choice of $\beta$ that defines the metric used in discretization, yields better classifiers (decision trees and naive Bayes)

Open issues:

- identifying simple parameters of data sets that inform the best choice of $\beta$;
- metric discretization for data with missing values.

# Future Directions of Work

- The metric space of attributes can be used to cluster attributes.
  - Similar attribute are grouped in clusters, that may have biological significance.
  - Retaining one attribute per cluster (e.g., the centroid) allows for data compression and for simplification of decision techniques.
- Study dynamic properties of clusterings.
- Classification of complex objects (that include graphs, histograms as components).
- Using wavelet transforms for studying total orderings on archeological data.