I'm from Pensacola
Pensacola
before
Hurricane IVAN
Pensacola after 9/16/04
What IS Pattern Recognition?

Where will this bee go?
Numerical Pattern Recognition

Process Description

Feature Analysis

Cluster Analysis

Classifier Design

Humans: "Knowledge Tidbits"

Sensors

Process Description

Feature Nomination

X = Object Data

R = Relation Data

Design

Test
2 kinds of data for Pattern Recognition

Objects
\[ \text{O} = \{o_1, \ldots, o_n\} : o_i = \text{i-th physical object} \]

Object Data
\[ X = \{x_1, \ldots, x_n\} : x_i = \text{feature vector for } o_i \]
\[ x_{ji} = \text{j-th (measured) feature of } x_i : 1 \leq j \leq p \]

Relational data
\[ R = [r_{ij}] = \text{relationship}(o_i, o_j) \text{ or } (x_i, x_j) \]
\[ s_{ij} = \text{similarity}(o_i, o_j) \text{ or } (x_i, x_j) \]
\[ d_{ij} = \text{dissimilarity}(o_i, o_j) \text{ or } (x_i, x_j) \]

Typically (\( R = D \))

(i) \[ d_{ij} \geq 0 \quad 1 \leq i \neq j \leq n \]
(ii) \[ d_{ii} = 0 \quad 1 \leq i \leq n \]
(iii) \[ d_{ij} = d_{ji} \quad 1 \leq i \neq j \leq n \]

When \( X \rightarrow D \), \( d_{ij} = \|x_i - x_j\|_E \), \( D \) is “Euclidean”
\[ x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{pmatrix} \]

\[ y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_p \end{pmatrix} \]

Perceived Relation

Mathematical Similarity

Perceived Similarity

Relational Data \( R(x,y) \)

Object → Object Data

HUMANS – COMPUTERS
Measuring (Dis)Similarity: The 5 Good Norms

**Mahalonobis**

\[ \delta_{M^{-1}} = \| x - v \|_{M^{-1}} = \sqrt{(x - v)^T M^{-1} (x - v)} \]

**Diagonal**

\[ \delta_{D^{-1}} = \| x - v \|_{D^{-1}} = \sqrt{(x - v)^T D^{-1} (x - v)} \]

**Euclidean**

\[ \delta_2 = \| x - v \|_{I^{-1}} = \sqrt{(x - v)^T I^{-1} (x - v)} \]

**City Block**

\[ \delta_1 = \| x - v \|_1 = \sum_{j=1}^{p} |x_j - v_j| \]

**Sup or Max**

\[ \delta_\infty = \| x - v \|_\infty = \max_{j=1}^{p} \{ |x_j - v_j| \} \]
Parameters of the Statistical Norms

**Mean Vector**
\[ \bar{v} = \frac{1}{n} \sum_{k=1}^{n} x_k \]

**Covariance Matrix**
\[ M = [m_{ij}] = \frac{1}{n} \sum_{k=1}^{n} (x_k - \bar{v})(x_k - \bar{v})^T \]

**Variance (Matrix)**
\[ D = \begin{bmatrix} m_{11} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & m_{pp} \end{bmatrix} = \begin{bmatrix} s_1^2 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & s_p^2 \end{bmatrix} \]
Level Sets and Open balls of Norms in $\mathbb{R}^2$

Inner Product Norms induce elliptical balls via eigenstructure of $A$

$$\langle x, x \rangle_A = \|x\|^2_A = x^T A x$$
Level Sets and Open Balls of 1-norm

\[ z = \| x \|_1 = x + y \]
Level Sets and Open Balls of 2-norm

\[ z = \left\| x \right\|_2 = \sqrt{x^2 + y^2} \]
Level Sets and Open Balls of 50-norm

\[ z = \| x \|_{50} = \left( x^{50} + y^{50} \right)^{1/50} \]
Feature Analysis

OBJECTS

NOMINATE

SENSOR MEASURES

OBJECT DATA

OBJECTS

O

Pattern Recognition
Feature Nomination: Issues

- What are the problems to solve?
- What variables are important?
- Which variables can be measured?
- What data analysis will be done?
- Are more features better than less?

Propose and collect raw (object) data.
Feature Extraction: Why?

Find $\phi : P(\mathbb{R}^p) \rightarrow P(\mathbb{R}^q)$ to:

1. Increase dimension: $p < q$
   - Functional link networks
   - Support Vector Machines

2. Decrease dimension: $p > q$
   - Decrease time and space
   - Eliminate “redundant” features
   - Assess tendencies visually ($q=2$)

3. $Y$ should have the “same information” as $X$ for:
   - Clustering and Classification
   - Prediction and Control
Feature Extraction: How?

1. Functions

\[ f : \mathbb{R}^p \rightarrow \mathbb{R}^q \]

- Linear
  - Axial ON Projection
  - Derived ON Projection (PCA)
  - Derived non-OG Projection

- Non-Linear

2. Algorithms

\[ A : P(\mathbb{R}^p) \rightarrow P(\mathbb{R}^q) \]

- Sammon's Algorithm
- FFBP Neural Nets
- SVMs (in theory!)

3. Use your...
Feature Analysis Example: Iris Data

Each object vector represents an Iris flower

- **n = 150** vectors
- **p = 4** features
- **c = 3** physical subspecies
  - \((n_i = 50\) vectors each)

\[
x_k = \begin{pmatrix}
x_{1k} \leftarrow \text{sepal length} \\
x_{2k} \leftarrow \text{sepal width} \\
x_{3k} \leftarrow \text{petal length} \\
x_{4k} \leftarrow \text{petal width}
\end{pmatrix}
\]
Visual Examination of Feature Ranges

F1 : Sepal Length
5 6 7

F2 : Sepal Width
2 3 4

F3 : Petal Length
1 2 3 4 5 6

F4 : Petal Width
0 1 2

A = Sestosa
B = Versicolor
C = Virginica

F3 or F4 separates A from BUC
(F1, F2) *linearly* separates A from B ∪ C

---and---
suggests that Iris has only *c = 2* geometric clusters
(F1, F2) linearly separates A from B∪C and suggests that Iris has only $c = 2$ geometric clusters.
(y₁, y₂) \textit{linearly} separates A from B \textcup C \text{ and } suggests that Iris has only c = 2 \textit{geometric} clusters
NL Extraction by 4:2:4 FFBP NN

Extracted feature pair (NN\textsubscript{21}, NN\textsubscript{22})

\[(\text{NN}\textsubscript{21}, \text{NN}\textsubscript{22})\]

linearly separates \(A\) from \(B \cup C\)

suggests that Iris has only \(c = 2\) geometric clusters
What is Clustering?

(Unsupervised Learning)

Label the similar objects

What is “similar”?

How many groups (c)?
Data sets usually contain *mixtures* of objects

Clustering *groups* and *labels* objects
Easy for Humans!  

Hard for computers!

Confusing Objects

Similarity Mistakes

What is “similar”?  

Wrong features?

Non-linear boundaries

How many groups (c)?

What kind of labels?

Mystery Data (Noise)

Incomplete data

Yipes!
Unipolar Label Vectors @ $c = 3$

- **Crisp** = Vertices
- **Fuzzy or Probabilistic** = Triangle
- **Possibilistic** = Cube-$\{-0\}$

\[ N_{h3} = \{e_1, e_2, e_3\} \]

\[

e_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix},
\]

\[

e_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix},
\]

\[

e_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix},
\]

\[ y = \begin{pmatrix} 0.1 \\ 0.6 \\ 0.3 \end{pmatrix} \]

\[ z = \begin{pmatrix} 0.7 \\ 0.2 \\ 0.7 \end{pmatrix} \]

\[ N_{f3} = \text{conv}(N_{h3}) \]

\[ N_{p3} = [0, 1]^3 - \{0\} \]
The matrix $U$ represents the membership functions for objects in clusters.

**Objects**

- $o_1$, $o_k$, $o_n$

**Clusters**

- $c_1$, $c_k$, $c_n$

**Membership Functions**

- $u_i: O \rightarrow [0, 1]$

- $u_i(o_k) = u_{ik} = \text{M'ship of } o_k \text{ in cluster } i$

**Partition Matrices**

- $u_{i1}$ to $u_{in}$
- $u_{c1}$ to $u_{cn}$

**Row $i$**

- Represents the membership of all $o_k$'s in cluster $i$

**Column $j$**

- Represents the membership of $o_k$ in each cluster

**Definition**

- $u = \{ u_{i1}, \ldots, u_{ik}, \ldots, u_{in}; u_{c1}, \ldots, u_{ck}, \ldots, u_{cn} \}$
<table>
<thead>
<tr>
<th></th>
<th>Crisp</th>
<th>Fuzzy/Prob</th>
<th>Possibilistic</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Row sums</strong></td>
<td>$\sum_{k} u_{ik} &gt; 0$</td>
<td>same</td>
<td>same</td>
</tr>
<tr>
<td><strong>Col sums</strong></td>
<td>$\sum_{i} u_{ik} = 1$</td>
<td>same</td>
<td>$\sum_{i=1}^{c} u_{ik} \leq c$</td>
</tr>
<tr>
<td><strong>M'ships</strong></td>
<td>$u_{ik} \in {0,1}$</td>
<td>$u_{ik} \in [0,1]$</td>
<td>same</td>
</tr>
<tr>
<td><strong>Set Name</strong></td>
<td>$M_{hcn}$</td>
<td>$M_{fcn}$</td>
<td>$M_{pcn}$</td>
</tr>
<tr>
<td><strong>Example</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 0 0 0 0</td>
<td>1 0.07 0.44</td>
<td>1 0.07 1.44</td>
</tr>
<tr>
<td></td>
<td>0 1 0 0</td>
<td>0 0.91 0.06</td>
<td>0 0.91 0.52</td>
</tr>
<tr>
<td></td>
<td>0 0 1 1</td>
<td>0 0.02 1.50</td>
<td>0 0.02 1.38</td>
</tr>
</tbody>
</table>

Take a 2nd Example
Partition Set Theory

\[ M_{fcn} = \text{conv}(M_{fcn0}) = \text{convex polytope} \]

Each face of \( M_{fcn0} \) looks like \( N_{fc} \)

\[ U = [1/c] = \text{centroid of } M_{fcn} \]

\[ \left| M_{hc} \right| = \left( \frac{1}{c!} \right) \sum_{j=1}^{c} \binom{c}{j} (-1)^{c-j} j^n \approx \left( \frac{c^n}{c!} \right) \]

\[ \dim(M_{fcn}) = n(c-1) \]
Defuzzification (Deprobabilization = Bayes Rule for $U \in M_{fcn}$ !)

$U = \begin{bmatrix} 1 & 0.07 & 0.6 & 0.5 \\ 0 & 0.81 & 0.2 & 0 \\ 0 & 0.12 & 0.7 & 0.5 \end{bmatrix} \in M_{pcn}$

$H_1(U) = U_{MM} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \in M_{hcn}$

resolve ties randomly

$\alpha$ -cut thresholding ($0 < \alpha < 1$) filters chosen levels of m'ship

$H_9(U) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \in M_{hcn}$

$H_8(U) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \in M_{hcn}$
Clusters represented by partition matrix \((U)\) or prototypes \(V = \{v_k\}\)

Crisp: hard boundaries

\[
U_c = \begin{bmatrix}
1 & 1 & \ldots & 1 \\
0 & 0 & \ldots & 0 \\
0 & 0 & \ldots & 0 \\
0 & 0 & \ldots & 0
\end{bmatrix}
\]

Fuzzy: soft boundaries

\[
U_f = \begin{bmatrix}
.9 & 1 & \ldots & .6 \\
.3 & 0 & \ldots & 0 \\
.1 & 0 & \ldots & 1 \\
.7 & 1 & \ldots & .4 \\
.8 & 1 & \ldots & .7
\end{bmatrix}
\]

4 types of clustering models

- \(U\) only models
- \((U, V)\) models
- \((U, V, +)\) models
- \(V\) only models
What is Classification?

(Supervised Learning)

Find regions that contain similar objects

What are “(decision) regions”?
Classifier Functions

\[ D : \mathbb{R}^p \mapsto N_{pc} = [0, 1]^c \]

3 Kinds of Training Data!

Feature Vectors in \( \mathbb{R}^p \)

Label Vectors in \( N_{pc} \)

Crisp

Fuzzy (or) Prob.

Poss.
Classifier Training

1. Choose a family of models $\mathcal{F} = \{ D_1(\theta) \ldots D_i(\theta) \ldots \}$

2. Training Data $X_t = \{ x_{t,1} \ldots x_{t,k} \ldots x_{t,n} \}$

3. Training Labels $U_t = \begin{bmatrix} 1 \\ 0 \\ .4 \\ .6 \\ .7 \\ .5 \end{bmatrix}$

4. Algorithm A "looks for" an optimal ($\theta^*$) of $D$

5. Use $(X, U_x, A)$ to find ($\theta^*$) of $D$ ("learning")
\[ d_{HP}(x) = \langle x, \bar{w} \rangle + \alpha + X_t + U_t + \text{Hyperplane Algorithm} \]

\[ \theta = \begin{bmatrix} (w_1), \alpha \end{bmatrix} \]

\[ d_{HP}(x) = x + y - 1 \]

\[ \theta^* = \begin{bmatrix} (1), -1 \end{bmatrix} \]

\( \bigcirc = \text{class 1} \)

\( \square = \text{class 2} \)
The classifier function $D_{HP}^*$ crisply labels every point in $\mathbb{R}^p$. For a point $x$, we have:

$$D_{HP}^*(x) = \begin{cases} e_1 = (1, 0)^T; d_{HP}(x) > \alpha^* & \Rightarrow x \in 1 \\ e_1 \text{ or } e_2; d_{HP}(x) = \alpha^* & \Rightarrow \text{Tie} \\ e_2 = (0, 1)^T; d_{HP}(x) < \alpha^* & \Rightarrow x \in 2 \end{cases}$$

Mistakes can be costly: a doctor treats patient $x$ with A or B

Use ($A| x \in 1$) $\Rightarrow$ cures $x$ 

Use ($B| x \in 1$) $\Rightarrow$ no cure, no harm

Use ($B| x \in 2$) $\Rightarrow$ cures $x$ 

Use ($A| x \in 2$) $\Rightarrow$ a cure?

Patient $x$ wants $D_{HP}^*$ to be soft. For $x$, we have:

$$D_{HP}^*(x) = \begin{pmatrix} 0.95 \\ 0.05 \end{pmatrix} \Rightarrow \begin{cases} \text{use A} \\ \text{more tests} \end{cases}$$
3 Kinds of Training and Operation
3 Data Sets needed to Train and Test a Classifier

Training: $X_{tr} = \{x_{tr,1} \ldots x_{tr,n}\}$ with labels $u_{tr} \in N_{pc}$

Validation: $X_v = \{x_{va,1} \ldots x_{va,s}\}$ with labels $u_{va} \in N_{hc}$

Test: $X_t = \{x_{te,1} \ldots x_{te,q}\}$ with labels $u_{te} \in N_{hc}$

Data can be uncooperative!

Non-Linear Clusters
Noise & Outliers
Mixtures of Points
High Dimensional
Wrong Features
Hardening SOFT Classifier Labels

$H(y) = e_i \iff \| y - e_i \| = \min_k \{ \| y - e_k \| \}$

$e_3 = H(z) = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$

$y = D(x) = \begin{pmatrix} 0.1 \\ 0.6 \\ 0.3 \end{pmatrix}$

$y \in i \uparrow$

$\forall i > y_k$

$z = D(\tilde{x}) = \begin{pmatrix} 0.5 \\ 0.6 \\ 0.7 \end{pmatrix}$

Same if $z = D(x)$ is possibilistic

$e_2 = H(y) = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$

$e_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$
Error rate estimation

\[ D(X) = \text{classifier learned by applying algorithm A to } X \]

\[ U_{D(X)} = \text{(possibly) soft labels given to } X \text{ by classifier } D \]

\[ H(U_{D(X)}) = \text{crisp labels given to } X \text{ by hardening } U_{D(X)} \]

\[ U_X = \text{(known) crisp labels of data set } X \text{ (} X_{va} \text{ or } X_{te} \text{)} \]

**Empirical Error Rate (generalization error)**

\[
E_D(X) = \left( \frac{\# \text{ wrong}}{\# \text{ tried}} \right) = \frac{\| U_X - H(U_{D(X)}) \|}{2X}
\]

Minimize \( E_D \) iteratively by learning with \( X_{tr} \) and then computing \( E_D(X_{va}) \)

Estimate \( E_D \) \text{ ONCE by computing } \( E_D(X_{te}) \)
Clustering vs Classification: Summary

Data $X$ + Other parameters $V, \ldots$

Clustering labels only the submitted data

Classifiers can label all the data in their domain space
# Building a PR system - the big picture

<table>
<thead>
<tr>
<th>Process Description</th>
<th>Feature Analysis</th>
<th>Cluster Analysis</th>
<th>Classifier Design</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Model Type</strong></td>
<td><strong>Operation</strong></td>
<td><strong>Criterion</strong></td>
<td><strong>Model Basis</strong></td>
</tr>
<tr>
<td>Numerical</td>
<td>Extraction</td>
<td>Dissimilarity</td>
<td>Deterministic</td>
</tr>
<tr>
<td>Rule-Based</td>
<td>Filtering</td>
<td>Connectivity</td>
<td>Fuzzy</td>
</tr>
<tr>
<td>Syntactic</td>
<td>2D-Displays</td>
<td>Frequency</td>
<td>Statistical</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Data Needed</th>
<th>Model + Algorithm</th>
<th>Model + Algorithm</th>
<th>Model + Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Projection</td>
<td>c Means</td>
<td>FFBP NN</td>
</tr>
<tr>
<td>Structure</td>
<td>PCA/LDA</td>
<td>SAHN</td>
<td>1-np rules</td>
</tr>
<tr>
<td>Nomination</td>
<td>Sammon</td>
<td>Entropy</td>
<td>K-nn rules</td>
</tr>
<tr>
<td>Collection</td>
<td>SOMs</td>
<td>GMD</td>
<td>SVMs</td>
</tr>
<tr>
<td>Sensors</td>
<td>VAT-coVAT</td>
<td>ART NN</td>
<td>Bayes Quad.</td>
</tr>
</tbody>
</table>
Numerical Pattern Recognition


Machine Learning & SVMs


Syntactic Pattern Recognition

