Lecture Notes on

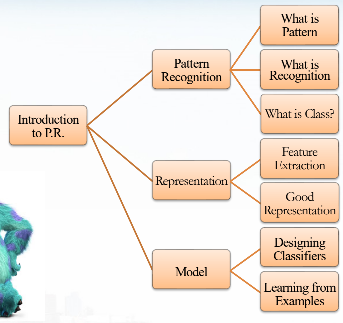
Pattern Recognition

Session 01(B): Introduction to P.R.

Gang Li
School of Information Technology
Deakin University, VIC 3125, Australia

1

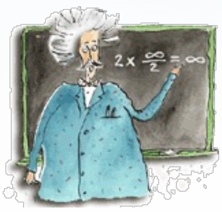
Road map



TULIP FLIP: Pattern Recognition (G. Li) 3

3

Pattern Recognition

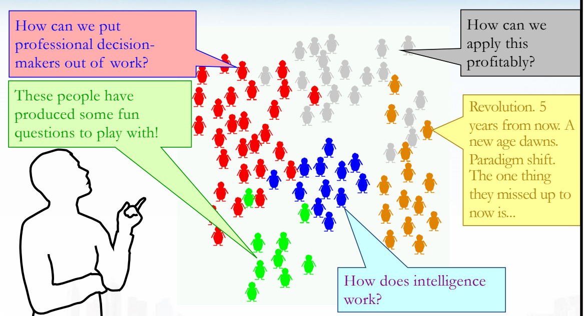


- What is Pattern?
- What is Recognition?
- What is Pattern Class?

TULIP FLIP: Pattern Recognition (G. Li) 4

4

An AI cocktail party



TULIP FLIP: Pattern Recognition (G. Li) 5

5

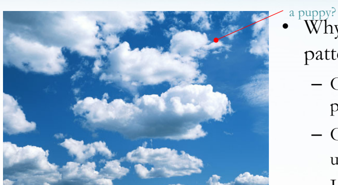
What Kinds of AI research?

Natural AI	**Algorithmic AI**
• Can we make something that is as intelligent as a human (or a bee)? • Can we get something that is evolutionary and self improving and autonomous and flexible?	• Can we save this bank $50billion by auto fraud detection? • Can we start a new industry of automated negotiation/voice translation etc.?

TULIP FLIP: Pattern Recognition (G. Li) 6

6

Pattern Discovery from Data



- How do we 'teach' a computer to do this?
  - This is a form of learning.
  - Also called unsupervised learning.
  - Key component of exploratory data analysis.

- Why do we tend to patterns in the cloud?
  - Our mind prefers patterns.
  - Our brains do 'clustering' unconsciously.
  - In fact, we are 'encoded' to see patterns in everything:
    - shopping, traffic,
    - what to eat, wear, etc. …

TULIP FLIP: Pattern Recognition (G. Li) 7

7

Pattern Recognition

Pattern Recognition

- The act of taking in raw data and taking an action based on the "category" of the pattern

The real power of human thinking is based on recognizing patterns. The better computers get at pattern recognition, the more human-like they will become.

Ray Kurzweil, NY Times, Nov 24, 2003

The problem of searching for patterns in data is a fundamental one and has a long and successful history.

Christopher M. Bishop, 2004

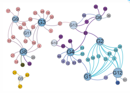
TULIP Team for Universal Learning and Intelligent Processing FLIP: Pattern Recognition (G. Li) 8

8

Pattern Recognition

- What digit is it?
  - zip code in US Post
- Where are the faces?
  - face detection
- Whose face is it?
  - face recognition
- Where are the groups?
  - Community detection





TULIP Team for Universal Learning and Intelligent Processing FLIP: Pattern Recognition (G. Li) 9

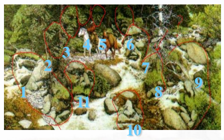
9

Pattern Recognition

Pattern

- "A pattern is the ***opposite of a chaos***; it is an entity vaguely defined, that could be given a name. " (Watanabe)
- A pattern is an abstract object, such as a set of measurements describing a physical object.





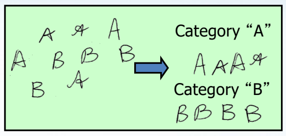
TULIP Team for Universal Learning and Intelligent Processing FLIP: Pattern Recognition (G. Li) 10

10

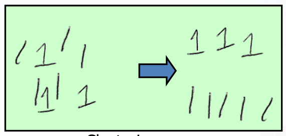
Pattern Recognition

Recognition

- Identification of a pattern as a member of a category we already know, or we are familiar with
  - **Classification** (known categories)
  - **Clustering** (learning categories)



Classification



Clustering

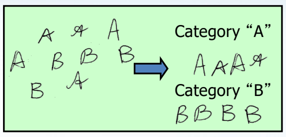
TULIP Team for Universal Learning and Intelligent Processing FLIP: Pattern Recognition (G. Li) 11

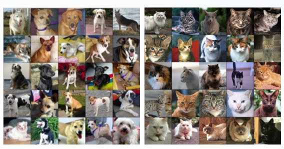
11

Pattern Recognition

Pattern Class

- A collection of similar (not necessarily identical) objects
- A class is defined by class samples
  - paradigms, exemplars, prototypes, training or learning samples





TULIP Team for Universal Learning and Intelligent Processing FLIP: Pattern Recognition (G. Li) 12

12

Pattern Recognition

- A basic attribute of people: ~~categorization of sensory input~~

pattern → **HUMAN BRAIN** → Class label

- Examples:
  - Diagnosis from medical images
  - Face detection from photo
  - etc.

TULIP Team for Universal Learning and Intelligent Processing FLIP: Pattern Recognition (G. Li) 13

13

Pattern Recognition

- A basic attribute of people: categorization of sensory input

pattern → Preprocessing → Feature extraction → Classification → Class label

- Examples:
  - Diagnosis from medical images
  - Face detection from photo
  - etc.

TULIP Team for Universal Learning and Intelligent Processing — FLIP: Pattern Recognition (G. Li) — 14

14

Pattern Recognition

- Machine Recognition of Patterns

pattern → Preprocessing → Feature extraction → Classification → Class label

- **Preprocessing**
  - remove noise, normalize, etc.
- **Feature Extraction**
  - makes some "problem-specific" measurement on the input pattern
- **Classifier**
  - maps each feature vector to a class label

TULIP Team for Universal Learning and Intelligent Processing — FLIP: Pattern Recognition (G. Li) — 15

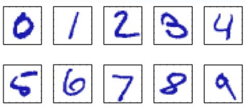
15

Pattern Recognition

- Machine Recognition of Patterns

pattern → Preprocessing → Feature extraction → Classification → Class label

- **Examples: Character Recognition**
  - **Pattern** → Image
  - **Class** → Identify of Character
  - **Features** → Binary Image, Projection, Moments, etc.



TULIP Team for Universal Learning and Intelligent Processing — FLIP: Pattern Recognition (G. Li) — 16

16

Pattern Recognition

- Machine Recognition of Patterns

pattern → Preprocessing → Feature extraction → Classification → Class label

- **Examples: Speech Recognition**
  - **Pattern** → 1D signal or its samples
  - **Class** → the semantics
  - **Features** → Spectral Info, LPC model, cepstrum, etc.

It is getting colder and colder. You must wear young more say, young more cool, me more say, me more cool, and wall sure say

TULIP Team for Universal Learning and Intelligent Processing — FLIP: Pattern Recognition (G. Li) — 17

17

Pattern Recognition

- Machine Recognition of Patterns

pattern → Preprocessing → Feature extraction → Classification → Class label

- **Examples: Speech Recognition**
  - **Pattern** → 1D signal or its sample
  - **Class** → the semantics
  - **Features** → Spectral Info, LPC model, cepstrum, etc.

TULIP Team for Universal Learning and Intelligent Processing — FLIP: Pattern Recognition (G. Li) — 18

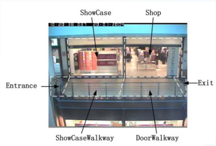
18

Pattern Recognition

- Machine Recognition of Patterns

pattern → Preprocessing → Feature extraction → Classification → Class label

- **Examples: Video Surveillance**
  - **Pattern** → Video sequence
  - **Class** → levels of alertness
  - **Features** → Motion Trajectories, parameters of a prefixed model, etc.



TULIP Team for Universal Learning and Intelligent Processing — FLIP: Pattern Recognition (G. Li) — 19

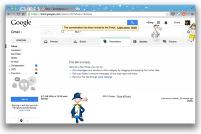
19

Pattern Recognition

- Machine Recognition of Patterns

pattern → Preprocessing → Feature extraction → Classification → Class label

- **Examples: Spam Detection**
  - **Pattern** → An Email
  - **Class** → Spam or Legitimate
  - **Features** → TF-IDF, BOW, word context, IP addresses, etc.



TULIP Team for Universal Learning and Intelligent Processing — FLIP: Pattern Recognition (G. Li) — 20

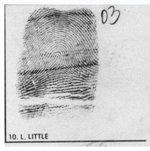
20

Pattern Recognition

- Machine Recognition of Patterns

pattern → Preprocessing → Feature extraction → Classification → Class label

- **Examples: Fingerprint (iPhone5S-)**
  - **Pattern** → Image plus an identity claim
  - **Class** → Yes or No
  - **Features** → Position of Minutiae, Orientation fields of Ridge lines, etc.



TULIP Team for Universal Learning and Intelligent Processing — FLIP: Pattern Recognition (G. Li) — 21

21

Pattern Recognition

- Machine Recognition of Patterns

pattern → Preprocessing → Feature extraction → Classification → Class label

- **Examples: Credit Screening**
  - **Pattern** → Details of an Applicant
  - **Class** → Yes or No
  - **Features** → income, job history, levels of credits, credits history etc.

TULIP Team for Universal Learning and Intelligent Processing — FLIP: Pattern Recognition (G. Li) — 22

22

Pattern Recognition

- Pattern Recognition System

Recognition: pattern → Preprocessing → Feature extraction → Classification → Class label

Representation | Model

Training: Patterns + Class labels → Preprocessing → Feature selection → Learning

TULIP Team for Universal Learning and Intelligent Processing — FLIP: Pattern Recognition (G. Li) — 23

23

Representation

- Feature Engineering
- Difficulties
- Representation

TULIP Team for Universal Learning and Intelligent Processing — FLIP: Pattern Recognition (G. Li) — 24

24

Feature Engineering

- Features depend on the problem

pattern → Preprocessing → Feature extraction →X→ Classification → Class label

- After ***feature extraction***, each pattern is represented as a ***vector***
  - A representation could consist of a vector of real-valued numbers, ordered list of attributes, parts and their relations….
- Some techniques available to extract more relevant quantities from the initial measurements (like PCA)

TULIP Team for Universal Learning and Intelligent Processing — FLIP: Pattern Recognition (G. Li) — 25

25

Difficulties of Representation

- **Feature Selection** vs. **Feature Extraction**
- How many and which subset of features to use in constructing the decision boundary?
  - Some features may be redundant
- Curse of dimensionality
  - Error rate may in fact increase with too many features in the case of small number of training samples

26

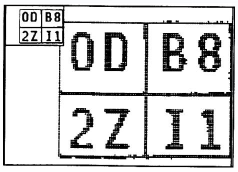
Intra-Class Variability

"T" in different typefaces

Same face under different expression, pose, illumination

28

Inter-Class Similarity



Characters that look similar

Twins

29

Good Representation

- Account for *intra-class variations*
- Ability to *discriminate* pattern classes of interest;
  - low inter-class similarity
- Should have some *invariant* properties
  - e.g., w.r.t. rotation, translation, scale…
- *Robustness* to noise, occlusion,..
- Lead to simple matching or decision-making strategies (e.g., linear decision boundary)
- Low measurement cost; real-time

30

Example
(Salmon vs Bass Classification)

- Preprocessing
  - image enhancement and segmentation
    1. separate touching or occluding fishes;
    2. extract fish contour

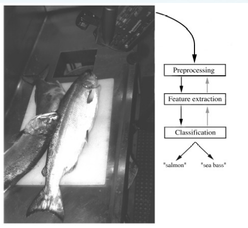


FIGURE 1.1. The objects to be classified are first sensed by a transducer (camera), whose signals are preprocessed. Next the features are extracted and finally the classification is emitted, here either "salmon" or "sea bass." Although the information flow is often chosen to be from the source to the classifier, some systems employ information flow in which earlier levels of processing can be altered based on the tentative or preliminary response in later levels (gray arrows). Yet others combine two or more stages into a unified step, such as simultaneous segmentation and feature extraction. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

31

Example
(Salmon vs Bass Classification)

- *Fish length* as a feature

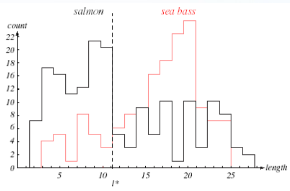


FIGURE 1.2. Histograms for the length feature for the two categories. No single threshold value of the length will serve to unambiguously discriminate between the two categories; using length alone, we will have some errors. The value marked l^* will lead to the smallest number of errors, on average. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

32

Example
(Salmon vs Bass Classification)

- *Fish lightness* as a feature

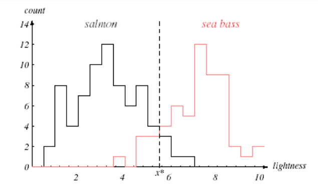


FIGURE 1.3. Histograms for the lightness feature for the two categories. No single threshold value x^* (decision boundary) will serve to unambiguously discriminate between the two categories; using lightness alone, we will have some errors. The value x^* marked will lead to the smallest number of errors, on average. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

33

Example
(Salmon vs Bass Classification)

- There are two possible classification errors.
  1. *Deciding the fish was a sea bass when it was a salmon.*
  2. *Deciding the fish was a salmon when it was a sea bass.*
- Which error is more important ?
  - It depends; e.g., if the fish packing company knows that:
    - Customers who buy salmon will object vigorously if they see sea bass in their cans.
    - Customers who buy sea bass will not be unhappy if they occasionally see some expensive salmon in their cans.
- How does this knowledge affect our decision?

34

Example
(Salmon vs Bass Classification)

- 2D Features leads to better separation

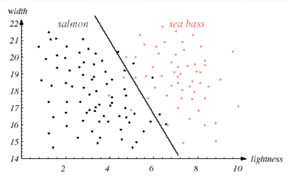


FIGURE 1.4. The two features of lightness and width for sea bass and salmon. The dark line could serve as a decision boundary of our classifier. Overall classification error on the data shown is lower than if we use only one feature as in Fig. 1.3, but there will still be some errors. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

35

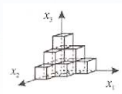
How Many Features

- Does adding more features always improve the results?
  - Correlated features do not improve performance.
  - It might be difficult to extract certain features.
  - It might be computationally expensive to extract many features.
  - "Curse" of dimensionality …

36

How Many Features

- Adding too many features can, paradoxically, lead to a worsening of performance.
  - Divide each of the input features into a number of intervals, so that the value of a feature can be specified approximately by saying in which interval it lies.

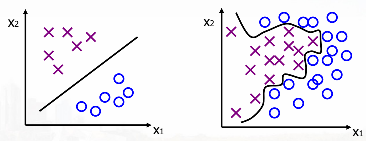


  - If each input feature is divided into **M** divisions, then the total number of cells is $\mathbf{M}^d$ (***d:*** # of features) which grows exponentially with ***d***.
  - Since each cell must contain at least one point, the number of training data grows ***exponentially!***

37

Representation

- Each pattern is represented as a point in d-dimensional feature space
  - Choice of features and their invariance properties are domain-specific
- Good representation implies
  1. *small* intra-class variation,
  2. *large* inter-class separation and
  3. *simple* decision boundary



38

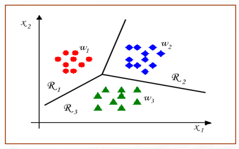
Classifiers

- Designing Classifiers
- Learning from Examples
- Issues on Classifiers
- Statistical Pattern Recognition

39

Classifiers

- After Representation, patterns are represented in a feature space
  - **Feature Space**: set of all possible feature vectors
  - **Classifier**: a decision model or function that maps the feature space to class labels {1, 2, … M}
    - We consider 2-class problem
    - For M-class, design M 2-class problem '*one vs rest*'



40

Designing Classifiers

- Need to decide how feature vector values determine the class
  - In most applications, it is impossible to design classifiers from "*physics of the problem*"
- The difficulties are
  - Variability in patterns of a single class
  - Variability in feature vector values
  - Feature vectors of patterns from different classes can be arbitrarily close
  - Noise in measurements

41

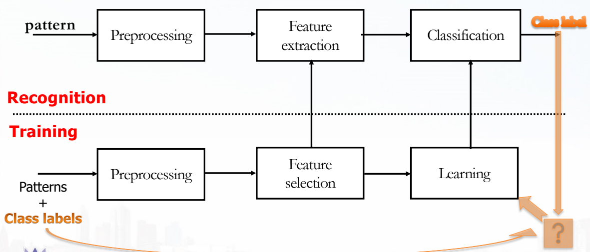
Designing Classifiers

- Often the only information available for the design is a training set of example patterns
  - **Training Set**: $\{(X_i, Y_i), I = 1, \ldots, n\}$
    - Take representative patterns of known category and obtain the feature vectors
  - Now learn an appropriate function $\boldsymbol{h}$ as classifier
    - Model Choice
  - Test and validate the classifier on more data
    - Noise in measurements

42

Learning from Examples

- Designing a classifier is learning from examples



43

Learning from Examples

- Nature of feedback determines the capability of the learning problem
  - **Supervised Learning**
    - The true label for each feature vector
  - **Reinforcement Learning**
    - Noisy assessment of performance
  - **Unsupervised Learning**
    - No feedback on the true labels

44

Learning from Examples
(Generalization)

- ***Errors*** on the ***training set*** do not necessarily tell how good is the classifier
  - Any classifier that amounts to only store the training set is useless
- Interested in the '***generalization ability***'
  - The ability of a classifier to produce correct results on ***novel*** patterns.

45

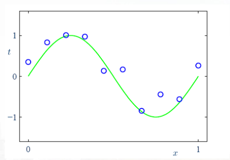
Learning from Examples
(Generalization)

- How can we ***improve generalization ability***?
  - More training examples
    - i.e., better model estimates
  - Simpler models usually yield better performance.
    - Occam's Razor

46

Learning from Examples
(Generalization)

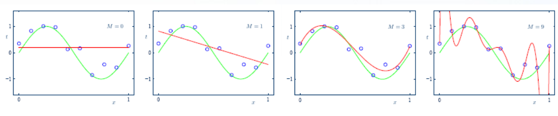
- **Regression** example:
  - plot of 10 sample points for the input variable x along with the corresponding target variable t (assuming some noise). Green curve is the true function that generated the data.



47

Learning from Examples
(Generalization)

- **Regression** example:
  - plots of polynomials having various orders, shown as red curves, fitted to the set of 10 sample points.

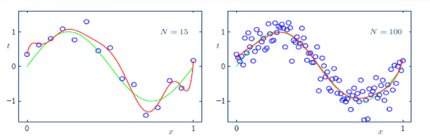


(a) 0'th order polynomial (b) 1'st order polynomial (c) 3'rd order polynomial (d) 9'th order polynomial

48

Learning from Examples
(Generalization)

- **Regression** example:
  - plots of 9th order polynomials fitted to 15 and 100 sample points



(a) 15 sample points (b) 100 sample points

49

Example
(Salmon vs Bass Classification)

- 2D Features leads to better separation

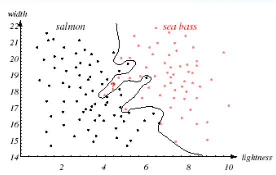


FIGURE 1.5. Overly complex models for the fish will lead to decision boundaries that are complicated. While such a decision may lead to perfect classification of our training samples, it would lead to poor performance on future patterns. The novel test point marked ? is evidently most likely a salmon, whereas the complex decision boundary shown leads it to be classified as a sea bass. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

50

Example
(Salmon vs Bass Classification)

- 2D Features leads to better separation

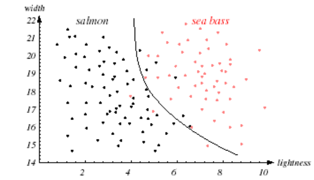


FIGURE 1.6. The decision boundary shown might represent the optimal tradeoff between performance on the training set and simplicity of classifier, thereby giving the highest accuracy on new patterns. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

TULIP Team for Universal Learning and Intelligent Processing — FLIP: Pattern Recognition (G. Li) 51

51

Overfitting

- Models complex than necessary lead to **overfitting**
  - i.e., ***good*** performance on the ***training (seen)*** data but ***poor*** performance on ***novel (unseen)*** data
- How can we adjust the complexity of the model ?
  - not very complex or simple
  - Are there principled methods for finding the best complexity ?

TULIP Team for Universal Learning and Intelligent Processing — FLIP: Pattern Recognition (G. Li) 52

52

Domain Knowledge

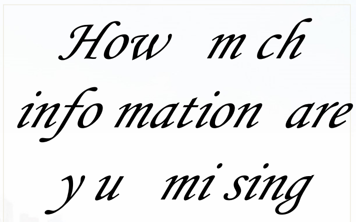
- When there is insufficient training data, ***incorporate domain knowledge***:
  - Model how each pattern is generated
    - analysis by synthesis
    - not easy! (e.g., recognize all types of chairs).
  - Incorporate some knowledge about the pattern generation method.
    - e.g., optical character recognition (OCR) assuming characters are sequences of strokes

TULIP Team for Universal Learning and Intelligent Processing — FLIP: Pattern Recognition (G. Li) 53

53

Context

- Utilizing **Side Information**



TULIP Team for Universal Learning and Intelligent Processing — FLIP: Pattern Recognition (G. Li) 54

54

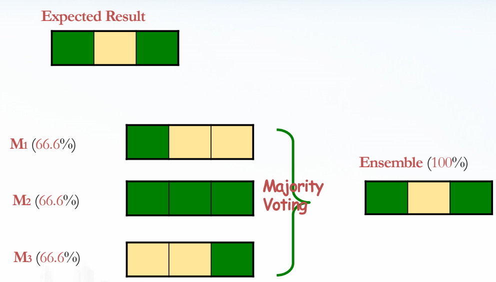
Classifier Combination

- Performance can be improved using *a "**pool**" of classifiers.*
  - How should we combine multiple classifiers ?

TULIP Team for Universal Learning and Intelligent Processing — FLIP: Pattern Recognition (G. Li) 55

55

Why Classifier Combination works? A simplified explanation



TULIP Team for Universal Learning and Intelligent Processing — FLIP: Pattern Recognition (G. Li) 56

56

Classification Error

- Each classification is associated with a risk
  - e.g., classification error involving costs for different errors
- How can we incorporate knowledge about such risks?
- Can we estimate the lowest possible risk of any classifier ?

57

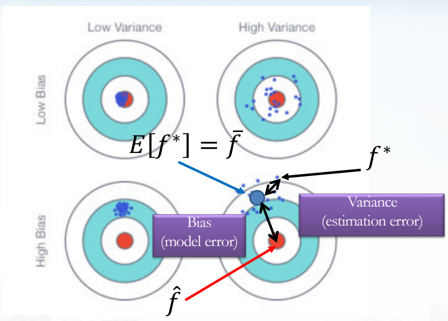
Main Sources of Error in Classifier Design

$$P(\omega_i / \mathbf{x}, D_i) = \frac{p(\mathbf{x}/\omega_i, D_i)P(\omega_i)}{\sum_j p(\mathbf{x}/\omega_j, D_j)P(\omega_j)}$$

- **Bayes error**
  - The error due to overlapping densities $p(x \mid \omega_i)$
- **Model error**
  - The error due to choosing an incorrect model.
- **Estimation error**
  - The error due to incorrectly estimated parameters
  - Inadequate training samples

58

Bias and Variance



59

Computational Complexity

- How does an algorithm scale with
  - the number of feature dimensions
  - the number of patterns
  - the number of categories
- Brute-force approaches might lead to perfect classifications results but usually have impractical time and memory requirements.
- Consider tradeoffs between computational complexity and performance.

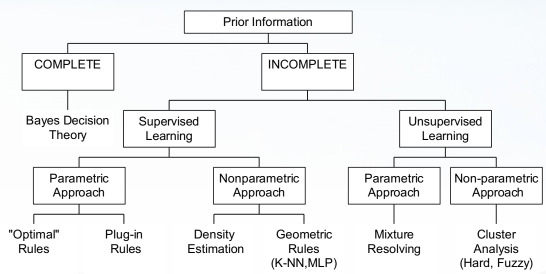
60

Statistical Pattern Recognition

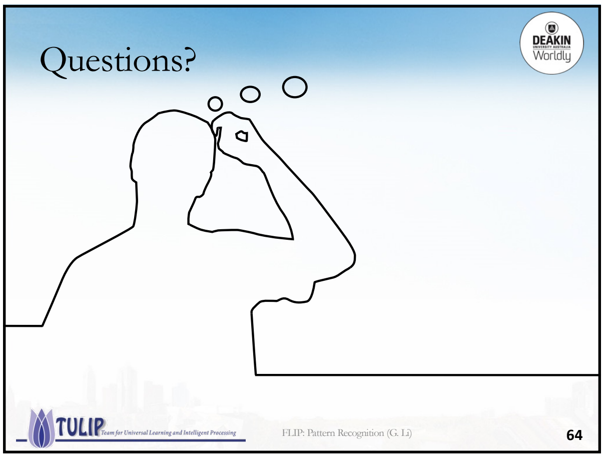
- The classifier should perform well despite inherent variability of patterns and noise in feature extraction and/or in class labels as given in training set
- **Statistical Pattern Recognition**
  - An approach where the variabilities are captured through probabilistic models

61

Statistical Pattern Recognition



62


Questions?
DEAKIN
Worldly
TULIP
FLIP: Pattern Recognition (G. Li)
64

64