Introduction to Cognitive Robotics

Module 5: Robot Vision

Lecture 7: K-nearest neighbour, minimum distance, linear, maximum likelihood and Bayes classifiers

David Vernon
Carnegie Mellon University Africa

www.vernon.eu

Classification

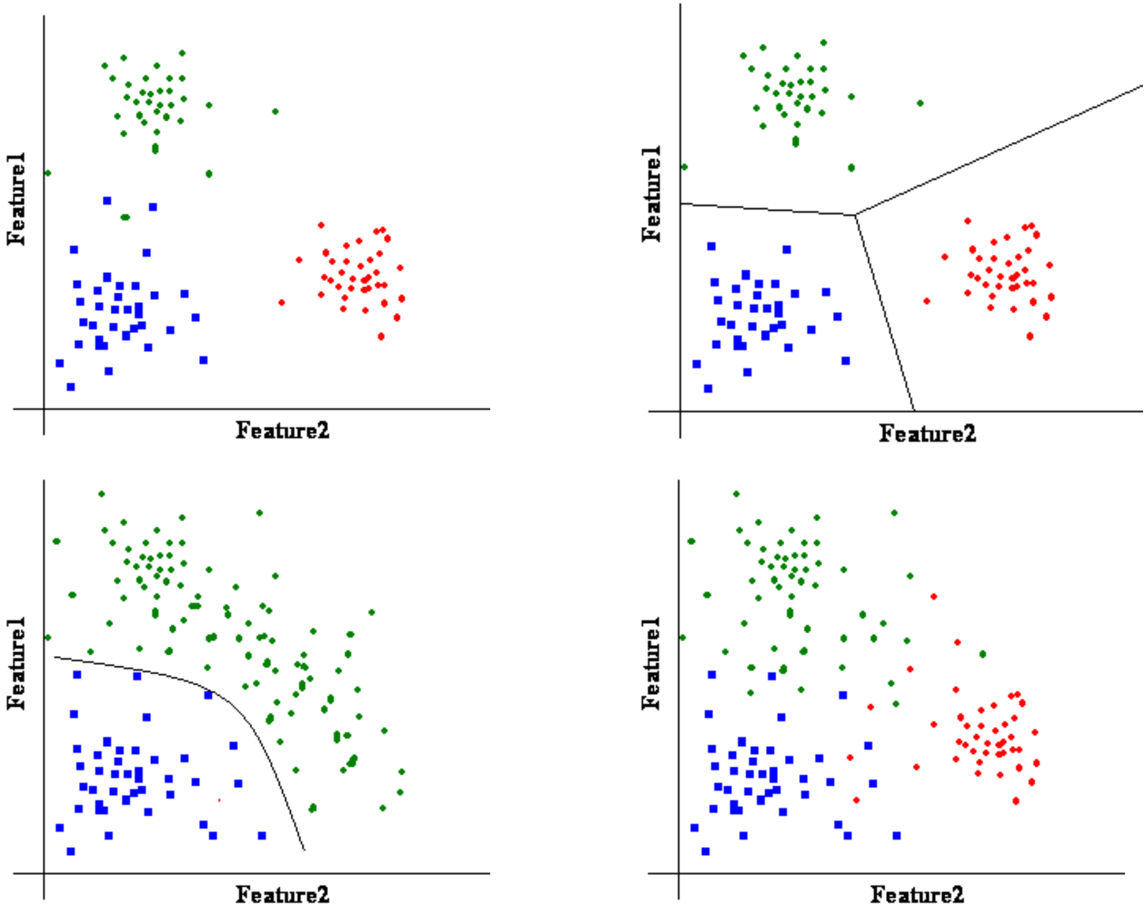
- The final stage of the statistical pattern recognition exercise
  - Classification of the objects on the basis of the set of features we have just computed, *i.e.* on the basis of the feature vector
  - Feature values are viewed as 'co-ordinates' of a point in n-dimensional space
  - Goal of classification: determine the sub-space to which the feature vector belongs
- Since each sub-space corresponds to a distinct object, the classification essentially accomplishes the object identification

Classification

- Object recognition
  - R classes: $w_1, w_2, \dots w_R$
- Classifier
  - n features: input pattern / feature vector: $x_1, x_2, \dots x_\mathrm{n}$
- Feature space
  - Choosing the features
  - Clusters in feature space
- Separability
  - Linear separability
  - Hyper-surfaces
  - Inseparable classes
- Classifiers
  - Nearest Neighbour Classifier / Minimum Distance Classifier
  - Linear Classifier
  - (Naive) Maximum Likelihood Classifier / Naive Bayes Classifier
  - …

Credit: Kenneth Dawson-Howe, A Practical Introduction to Computer Vision with OpenCV, © Wiley & Sons Inc. 2014

Classification

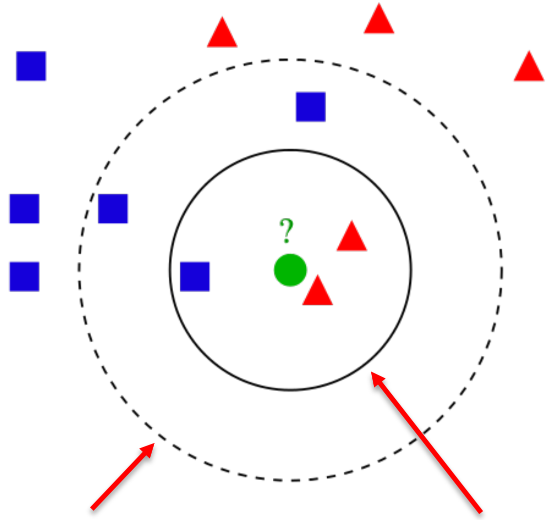


Credit: Kenneth Dawson-Howe, A Practical Introduction to Computer Vision with OpenCV, © Wiley & Sons Inc. 2014

Classification

k-nearest neighbour classifier

- Each class represented by a set of exemplar feature vectors
- For an unknown object
  - Assigning the label which is most frequent among the *k* training samples nearest to that query point
  - *k* is defined by the user
- Advantages
  - Training
  - Computational complexity



5-nearest neighbour:
classify as a rectangle

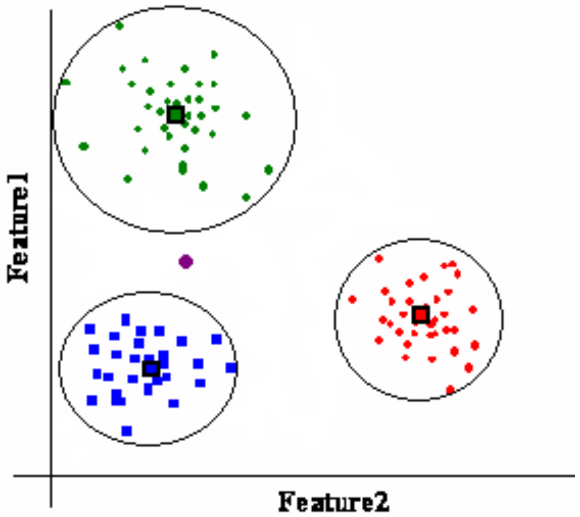
3-nearest neighbour:
classify as a triangle

Credit: https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm

Classification

Minimum distance classifier

- Each class represented by the mean of vectors for that class
- For an unknown object
  - Determine the distance to the exemplars
  - Pick the class with the smallest distance
- Unknown class?
  - Distance must be less than some threshold
- Advantages
  - Training
  - Computational complexity

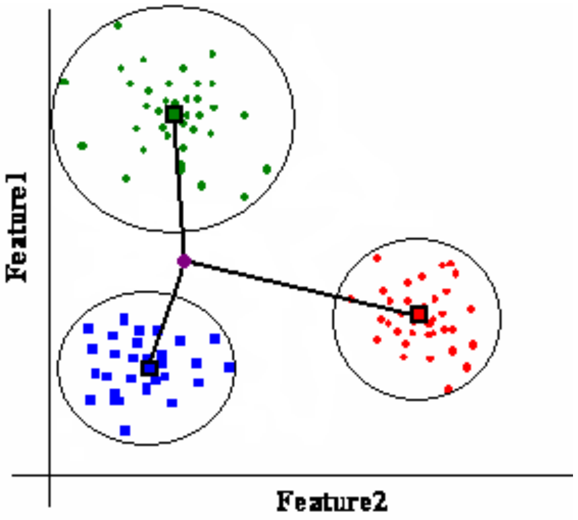


Credit: Kenneth Dawson-Howe, A Practical Introduction to Computer Vision with OpenCV, © Wiley & Sons Inc. 2014

Classification

Minimum distance classifier

- Each class represented by the mean of vectors for that class
- For an unknown object
  - Determine the distance to the exemplars
  - Pick the class with the smallest distance
- Unknown class?
  - Distance must be less than some threshold
- Advantages
  - Training
  - Computational complexity



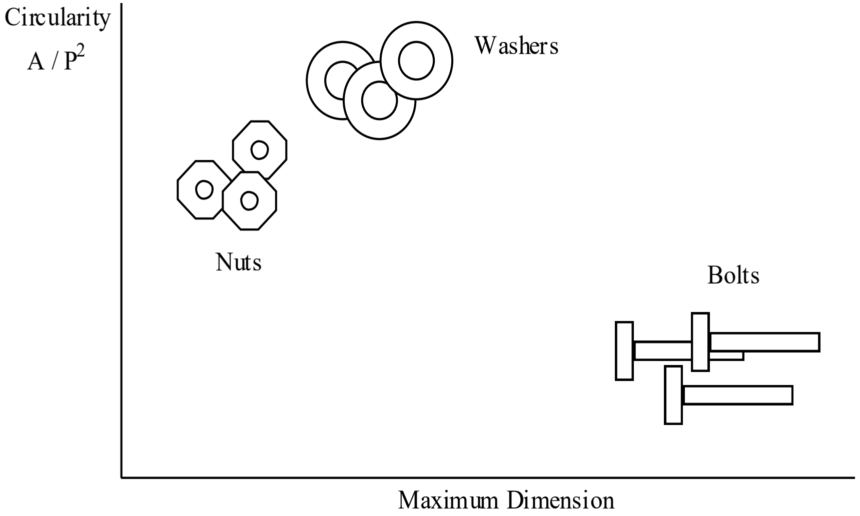
Credit: Kenneth Dawson-Howe, A Practical Introduction to Computer Vision with OpenCV, © Wiley & Sons Inc. 2014

Classification

Example

- Classify nuts, bolts, and washers

- Use two features: circularity and maximum dimension



Classification

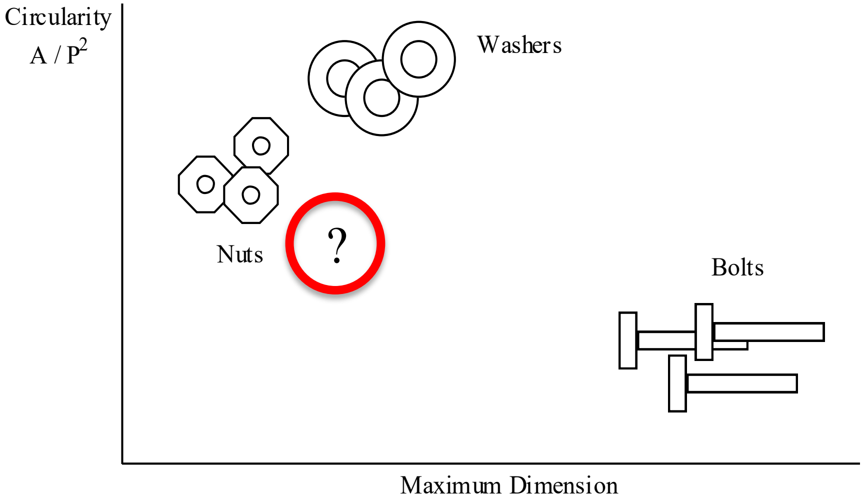
Example

- Classify nuts, bolts, and washers
- Use two features: circularity and maximum dimension
- Learn the distribution of these features for the three objects
  - Need a training set
- Measure the features for an unknown object
- Classify it on the basis of its position in the feature space
- Which sub-space does it belong to?

Classification

Example

- Which sub-space does it belong to?



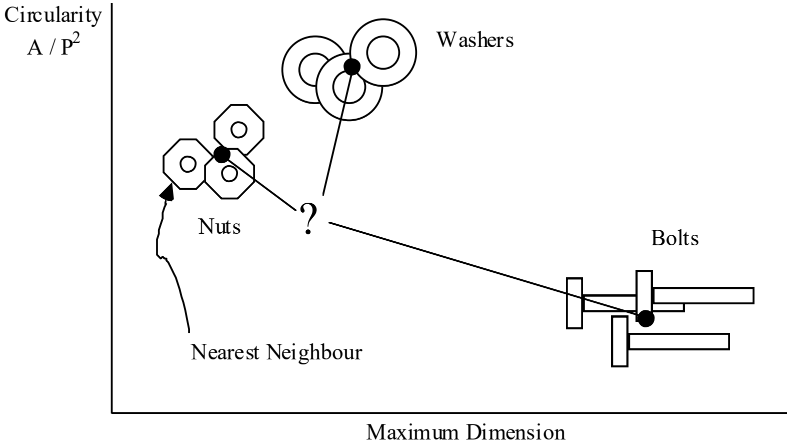
Classification

Minimum distance classifier

- Classifies the object on the basis of the distance of the unknown object vector position from the centre of the three clusters

- Choosing the closest cluster as the one to which it belongs

- The position of the centre of each cluster is simply the average of each of the individual training vector positions



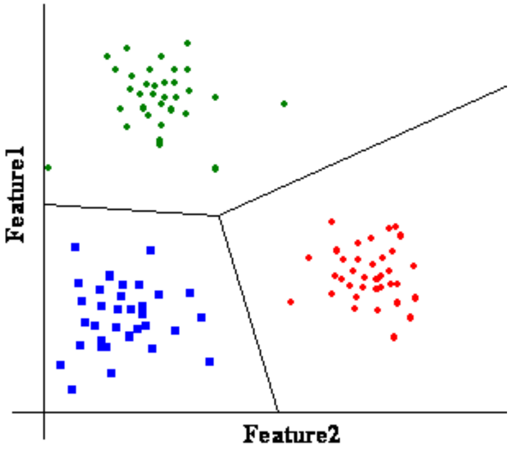
Classification

- Decision rule
  - $w_r = d(x)$ divides the feature space into R disjoint sub-spaces K_r , $r = 1, \dots , R$

- Linear discrimination functions
  - For each class we can define a discrimination function $g_r(x)$ for which $g_r(x) \geq \mathrm{g_s}(x)$ for all values of x for any point in K_r

    $g_r(x) = q_{ro} + q_{r1}x_1 + \dots + q_{rn}x_n$
  - The hyper-surface between any two sub-spaces is defined as $g_r(x) - g_s(x) = 0$

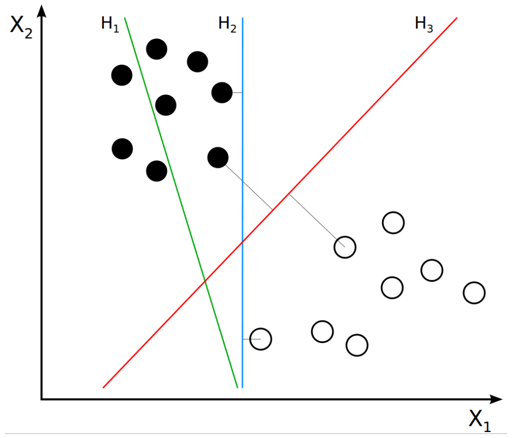
- Unknown class
  - The discrimination function which has the highest value defines which class is selected: $g_r(x) > \text{threshold}$

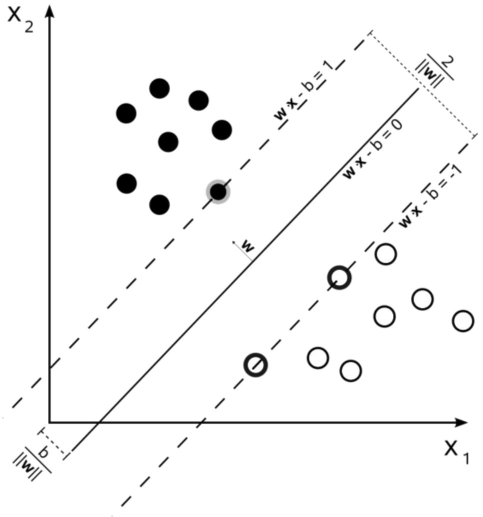


Credit: Kenneth Dawson-Howe, A Practical Introduction to Computer Vision with OpenCV, © Wiley & Sons Inc. 2014

Classification

Support Vector Machine (SVM) classification




X_2
$\mathbf{w}\cdot\mathbf{x}-b=1$
$\frac{2}{\|\mathbf{w}\|}$
$\mathbf{w}\cdot\mathbf{x}-b=0$
$\mathbf{w}\cdot\mathbf{x}-b=-1$
$\mathbf{w}$
X_1
$\frac{b}{\|\mathbf{w}\|}$

- Maximum margin classifier: H3 separates classes “better” than H2
- Training examples defining separating hyperplane: support vectors
- If not linearly separable: use kernel trick to map to higher-dimensional space

Credit: Markus Vincze, Technische Universität Wien

Classification

Classifier Learning

- Training set
  - Must be representative
  - Must be inductive

- Training set size
  - Training set provides the unknown statistical information
  - Size will typically have to be increased several times

- Sample learning strategies
  - Supervised:
    - Probability density estimation – estimating $p(x \mid w_r)$ & $p(w_r)$
    - Training set includes class specification for every instance
  - Unsupervised:
    - Cluster Analysis
    - Look for similarities in feature space

Credit: Kenneth Dawson-Howe, A Practical Introduction to Computer Vision with OpenCV, © Wiley & Sons Inc. 2014

Classification

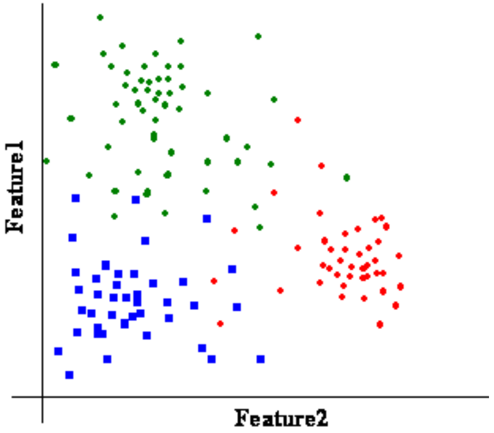
Maximum-likelihood classifier

- Suppose we wish to distinguish between nuts and bolts (no washers this time).
- Circularity measure will suffice
  - one feature and a one-dimensional feature space
  - two classes of object: nuts and bolts.
- Let us refer to these classes as C_n and C_b
- Let us refer to the circularity feature value as x
- We can use Bayes' Theorem to create a good classifier (better than a nearest neighbour classifier)

Classification

Maximum-likelihood classifier

- Use knowledge about the statistical distribution of each class
- Classification maximizes the probability of assigning the unknown object to the correct class



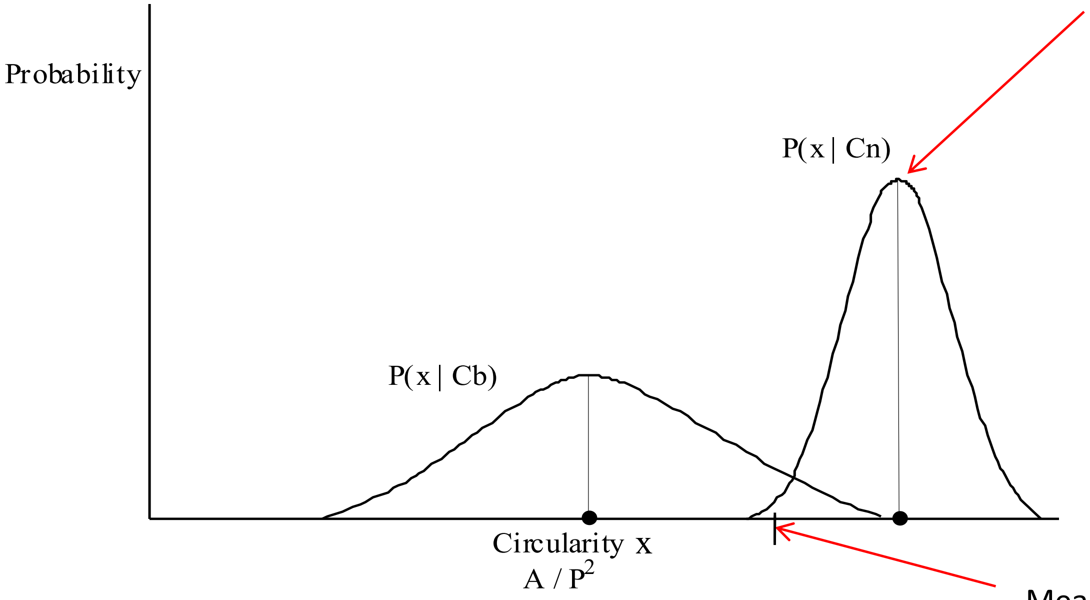
Credit: Kenneth Dawson-Howe, A Practical Introduction to Computer Vision with OpenCV, © Wiley & Sons Inc. 2014

Classification

Example: Maximum Likelihood Classifier

- Let’s design a system that can classify two parts: nuts and bolts
- Two classes $C_b\, C_n$
- Let’s decide to use a feature ‘circularity’ x to distinguish nuts from bolts (nuts are more circular than bolts)

Classification



Nearest neighbour

Measured value

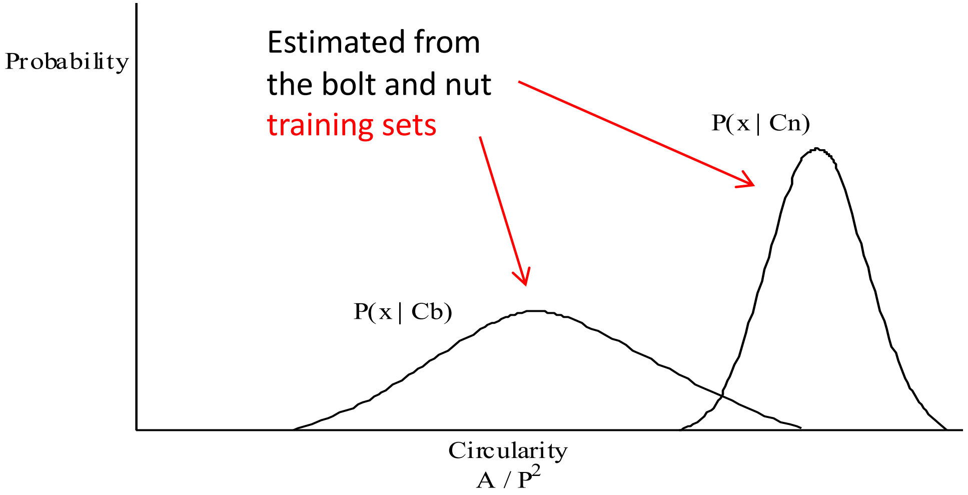
We can do better than nearest neighbour if we use knowledge about the conditional probability of an object having some feature value $P(x|C_i)$ and the prior probability of that object being there $P(C_i)$ in the first place

Classification

Example: Maximum Likelihood Classifier

- The conditional probability, $p(x \mid C_b)$ enumerates the probability that a circularity x will occur, given that the object is a bolt

- $p(x \mid C_n)$ is the conditional probability that a circularity x will occur, given that the object is a nut



Classification

Example: Maximum Likelihood Classifier

- This isn't what we need ...
- We want the posterior probability that an object belongs to a particular class, given that a particular value of x has occurred $P(C_i|x)$

  (Remember: we know the value of x because we measure it)
- We classify the object as a bolt if

$$P(C_b|x) > P(C_n|x)$$

- We use Bayes’ Theorem to convert the probabilities we know (or can estimate),
  i.e. the conditional probabilities and the prior probabilities,
  to the ones we need, i.e. the posterior probabilities

Classification

Example: Maximum Likelihood Classifier

- The posterior probability, $P(C_i|x)$, that the object belongs to a particular class i is given by Bayes' Theorem:

$$P(C_i|x) = \frac{P(x|C_i)P(C_i)}{P(x)}$$

where

$$P(x) = \sum_{i=1}^{2} P(x|C_i)P(C_i)$$

Classification

Example: Maximum Likelihood Classifier

$p(x)$ is a normalisation factor which is used to ensure that the sum of the posterior probabilities sum to one, for the same reasons as mentioned earlier

Classification

Example: Maximum Likelihood Classifier

- How do we determine the conditional probabilities for the two classes?
- Since it is not likely that these are given, we estimate them from a training set

Classification

Example: Maximum Likelihood Classifier

- Let S be the space of circularity values that we can measure with our computer vision system

- Let $X_n(x)$ be the random variable that equals the number of times a given circularity value appears when x is the outcome when the circularity of a nut is measured

- Let $X_b(x)$ be the random variable that equals the number of times a given circularity value appears when x is the outcome when the circularity of a bolt is measured

- We need the probability distribution of random variables X_n and X_b

Classification

Example: Maximum Likelihood Classifier

- Since the circularity value is going to vary continuously (i.e. it won't have a finite set of values), we use a continuous random variable
  - The distribution is called a **Probability Density Function (PDF)**
- If the measured feature had a finite set of discrete values, we would have used discrete random variables
  - The distribution is called a **Probability Mass Function**

Classification

Example: Maximum Likelihood Classifier

- The PDF for nuts can be estimated in a relatively simple manner
  - measuring the value of x for a large number of nuts in a training set
  - plotting the histogram of these values
  - smoothing the histogram
  - normalizing the values so that the total area under the histogram equals 1
    - The normalization step is necessary because certainty has a probability value of 1 and the sum of all the probabilities (for all the possible circularity measures) must necessarily be equal to a certainty of having that object, *i.e.*, a probability value of 1

Classification

Example: Maximum Likelihood Classifier

- The PDF for bolts can be estimated in a similar manner

Classification

Example: Maximum Likelihood Classifier

- Again:

  the two PDFs tell us the probability that the circularity x will occur

  1. given that the object belongs to the class of nuts C_n
  2. given that the object belongs to the class of C_b

- That is, they give us the conditional probability of an object having a certain feature value, given that we know that it belongs to a particular class

Classification

Example: Maximum Likelihood Classifier

- Next problem: the prior probability of each class occurring
  - We may know, for instance, that the class of nuts is, in general, likely to occur twice as often as the class of bolts
  - In this case we say that the prior (or *a priori*) probability of the two classes are :

    $P(C_n) = 0.666$ and $P(C_b) = 0.333$

  - In fact, in this case, it is more likely that they will have the same prior probabilities (0.5) since we usually have a nut for each bolt

Classification

Example: Maximum Likelihood Classifier

- In summary, Bayes' theorem allows us to use
  - the prior probability of objects occurring in the first place
  - the conditional probability of an object having a particular feature value given that it belongs to a particular class and …
  - the actual measurement of a feature value (to be used as the argument in the conditional probability)
  - to estimate the posterior probability that the measured object belongs to a given class

Classification

Example: Maximum Likelihood Classifier

- In summary,
  - Once we estimate
    - the posterior probability that, for a given measurement, the object is a nut
    - and the posterior probability that it is a bolt
  - we can make a decision as to its identity
    - choosing the class with the higher probability

Classification

Example: Maximum Likelihood Classifier

- This is why it is called the maximum likelihood classifier
- Thus, we classify the object as a bolt if :

$$P(C_b|x) > P(C_n|x)$$

Classification

Example: Maximum Likelihood Classifier

- Noting that the normalization factor $p(x)$ is the same for both expressions, we can rewrite this test as

$$P(x|C_b)P(C_b) > P(x|C_n)P(C_n)$$

- These expressions are just the right-hand sides of the corresponding Bayes' Theorem equations

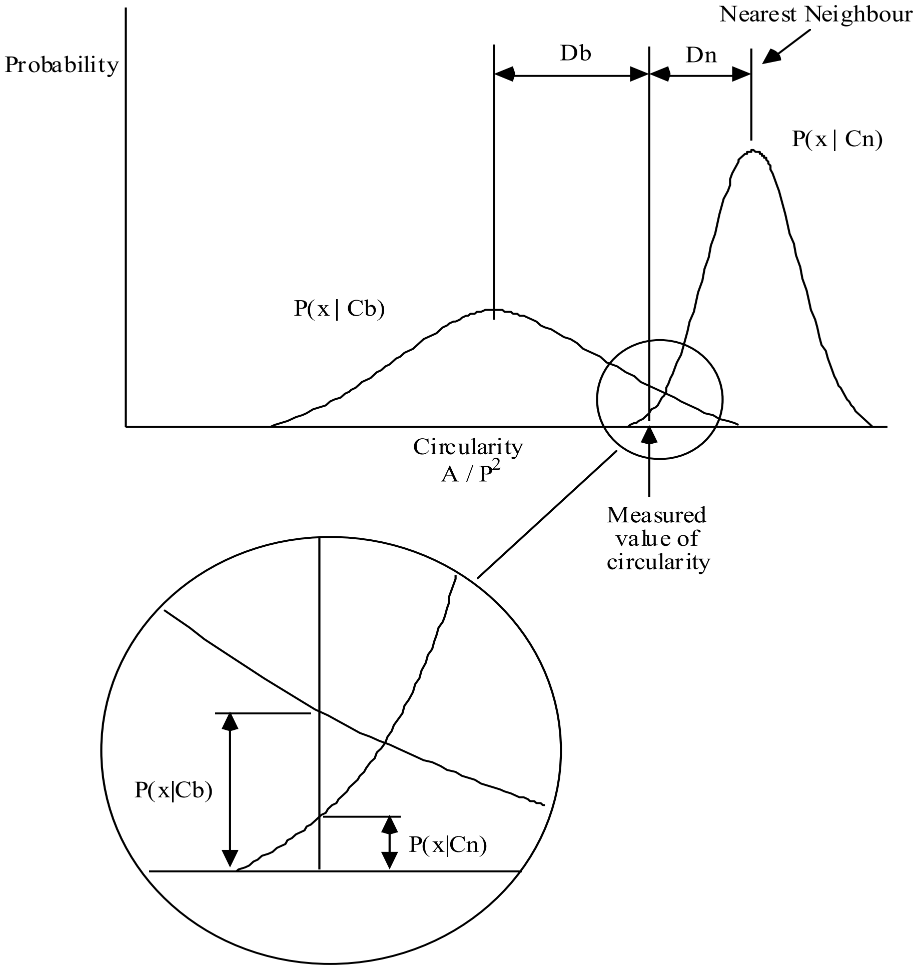
Classification

Example: Maximum Likelihood Classifier

- If we assume that the chances of an unknown object being either a nut or a bolt are equally likely, *i.e.* $P(C_b) = P(C_n)$, then we classify the unknown object as a bolt if

$$P(x|C_b) > P(x|C_n)$$

Classification



Classification

Example: Maximum Likelihood Classifier

- For the example shown $p(x \mid C_b)$ is indeed greater than $p(x \mid C_n)$ for the measured value of circularity and we classify the object as a bolt

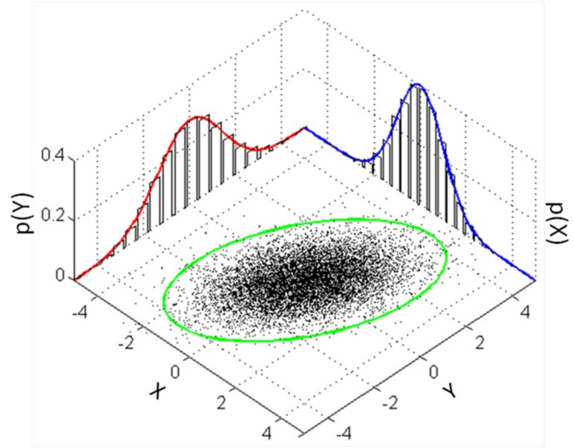
- If, on the other hand, we were to use the minimum distance classification technique, we would choose the class whose mean value “is closer to” the measured value

In this case, the distance D_n from the measured value to the mean of the PDF for nuts is less than D_b, the distance from the measured value to the mean of the PDF for bolts; we would probably classify the object incorrectly as a nut

Classification

Example: Maximum Likelihood Classifier

- This was a simple example with just one feature and a 1-D PDF
- However, the argument generalizes directly to n-dimensions, where we have n features in which case the conditional probability density functions are also n-dimensional



Classification

Example: Maximum Likelihood Classifier

- If we assume that the features are independent then we can use the theory we've just outlined, multiplying together the conditional probabilities for each class
  - This is known a **Naïve Bayes Classifier**
  - It may be naïve, but it works surprisingly well
- If we don't assume independence, then we need a more complex theory

Reading

D. Vernon, Machine Vision, Prentice-Hall International, Section 6.3 Decision-theoretic Techniques