Classification

Week 1 - Lecture 2

Dr Matthew Ellis - COM2004/3004 Data Driven Computing 23/24

Goals of today's session

By the end of today's session you should be able to:

1. Explain the process of designing a classifier.
2. Define Bayes' decision rule for classifying a measurement.
3. Understand the difference between the posterior and likelihood probabilities.
4. Explain how generative modelling is used to compute posterior probabilities.

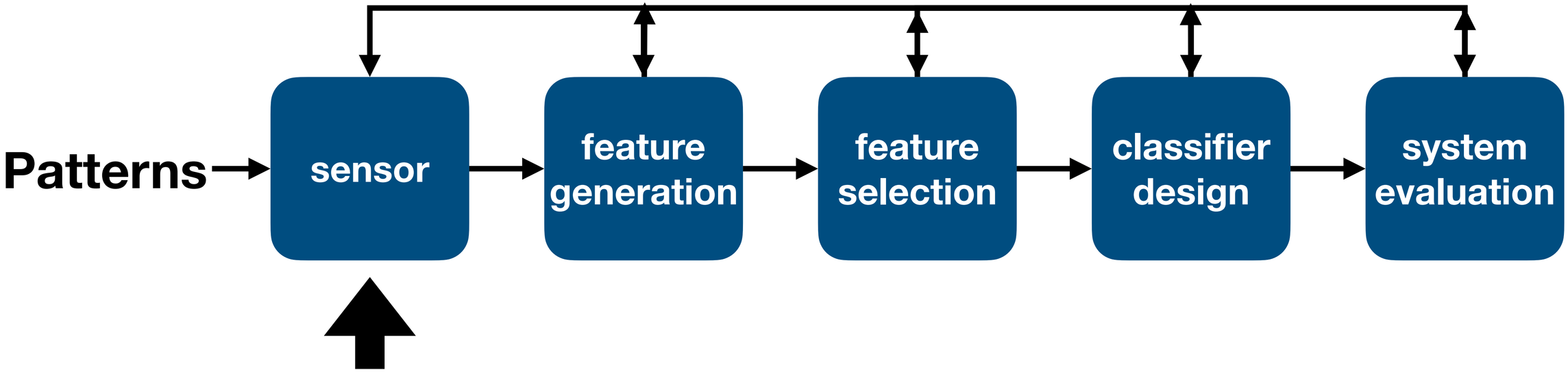
Summary

Types of data driven computing problems:

- Supervised machine learning
    - **Classification** - being able to label new data
    - **Regression** - predicting continuous valued quantities
- Unsupervised machine learning
    - **Clustering** - discovering patterns and structure in data
    - **Dimensionality reduction** - finding smaller representations

The first half of this module will focus on **Classification**.

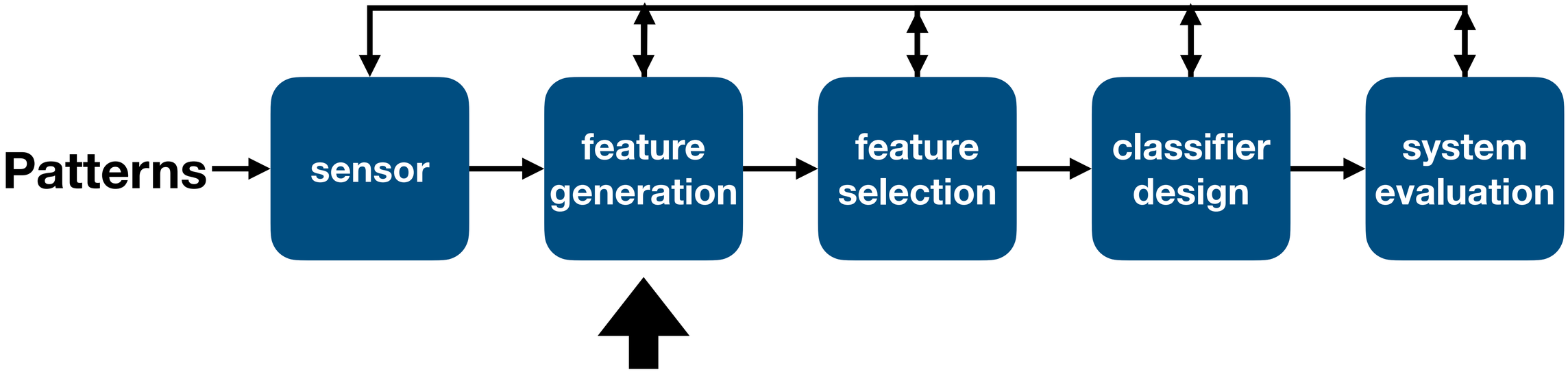
Designing a classifier



Sensor - capture data from physical world

e.g. microphone, camera, temperature sensors, blood pressure monitor etc

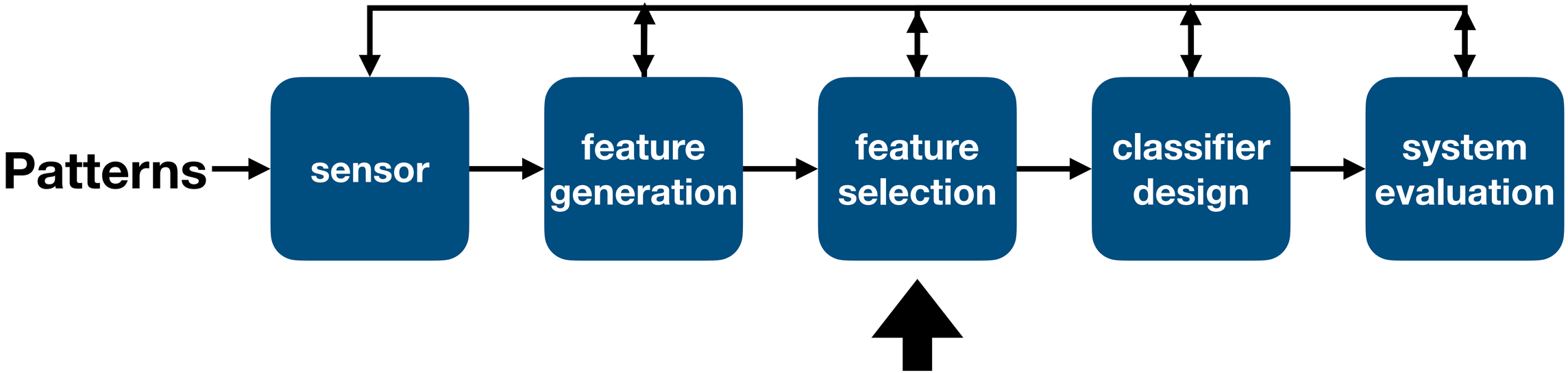
Designing a classifier



Feature generation – generate candidate features from raw data

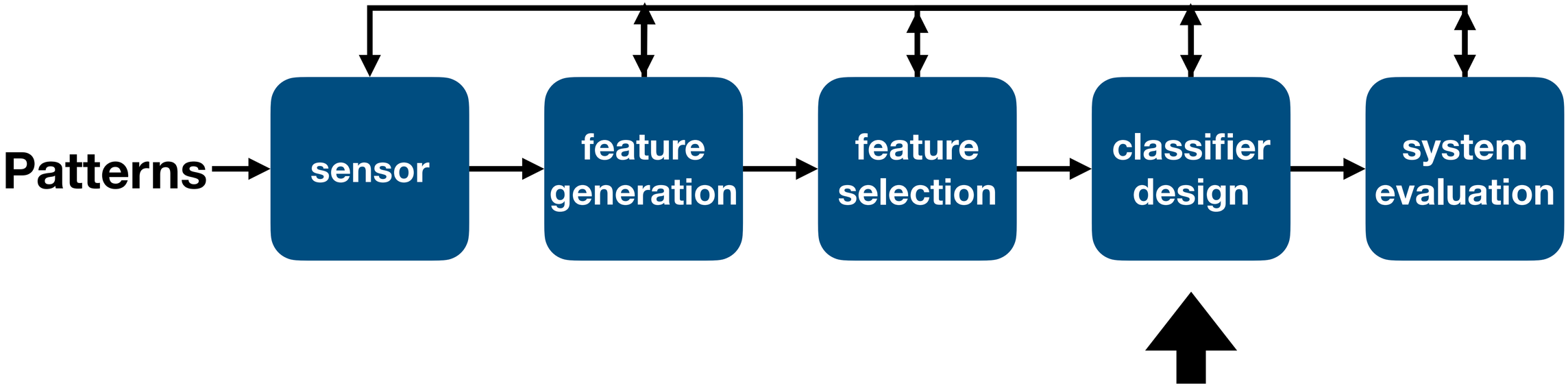
e.g. frequency analysis of sound wave; edge detection in image data

Designing a classifier



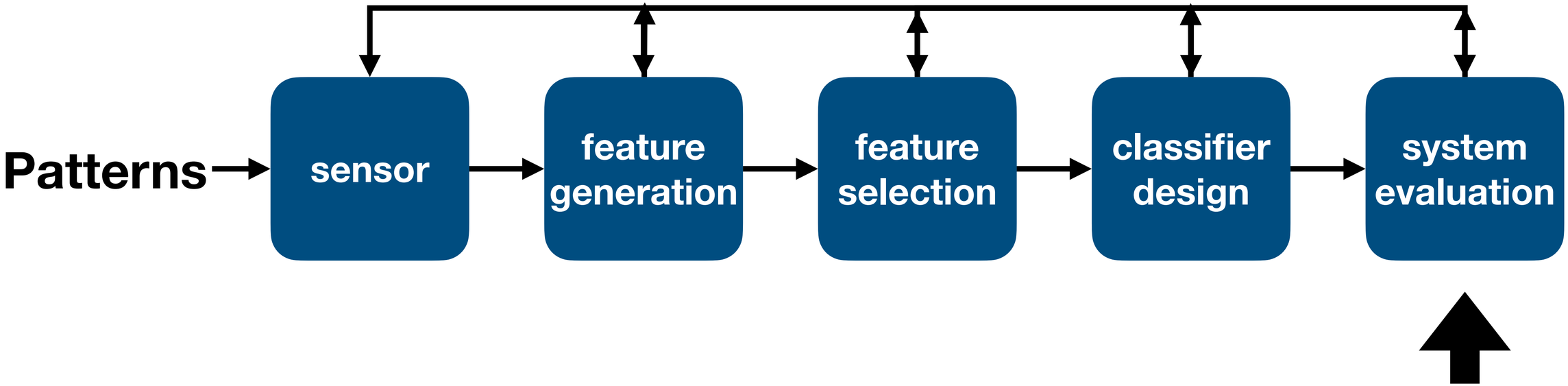
Feature selection – choose subset of features that carry most information

Designing a classifier



Classifier – optimal design depends on statistical properties of features

Designing a classifier



Evaluation – test system to measure performance, redesign

A closer look

Classification example

Cat or Dog?

Let’s consider some ‘rules’ that we might use to classify an animal as either a cat or dog from a photo?

Training data
















Some examples might be ambiguous

Machines perform the task in 2 stages:

- Extract useful **features** (e.g locate eyes, measure their height/width)

Second half of COM2004/3004

- Make a **classification** based on the features

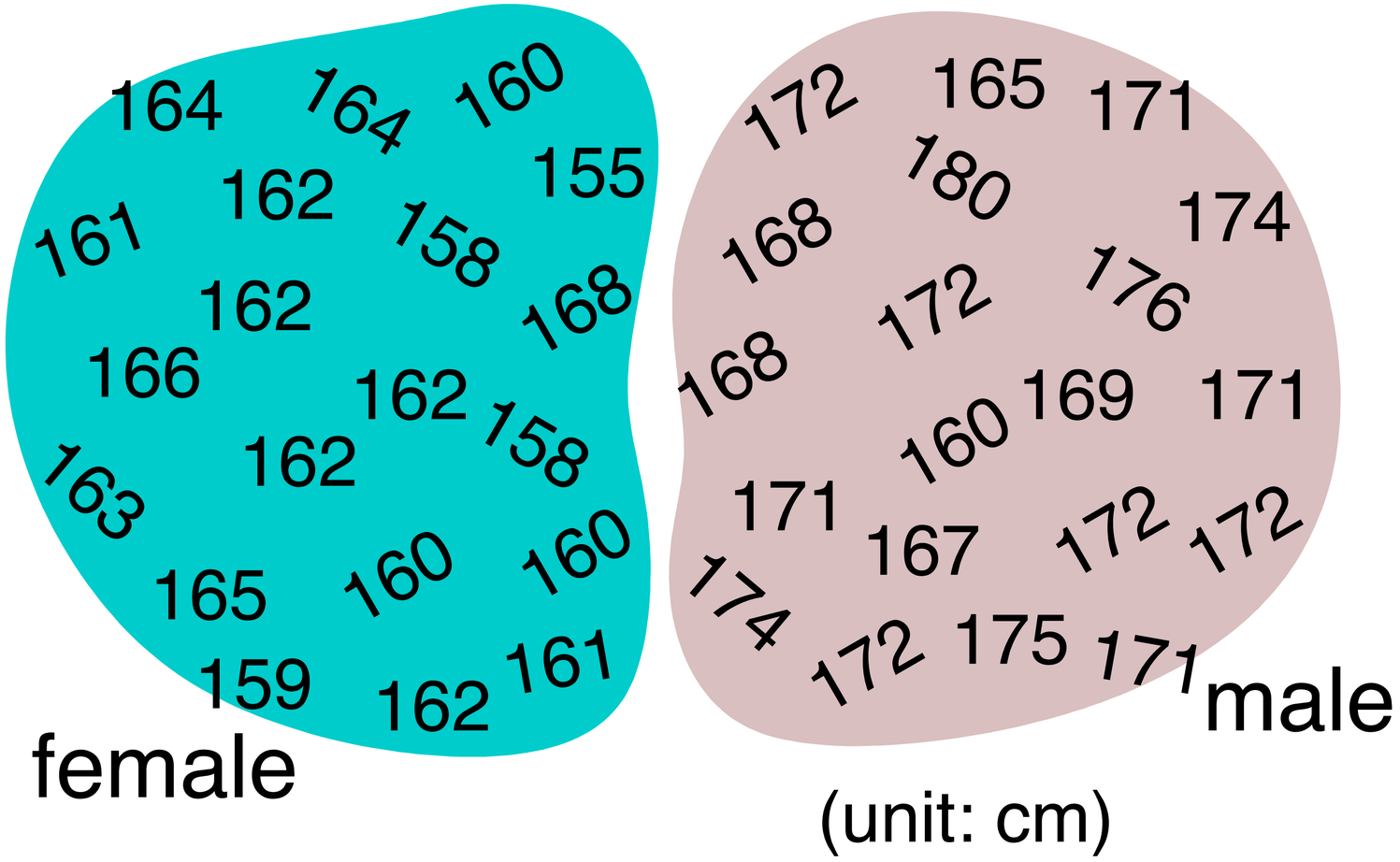
First half of COM2004/3004

Selecting features

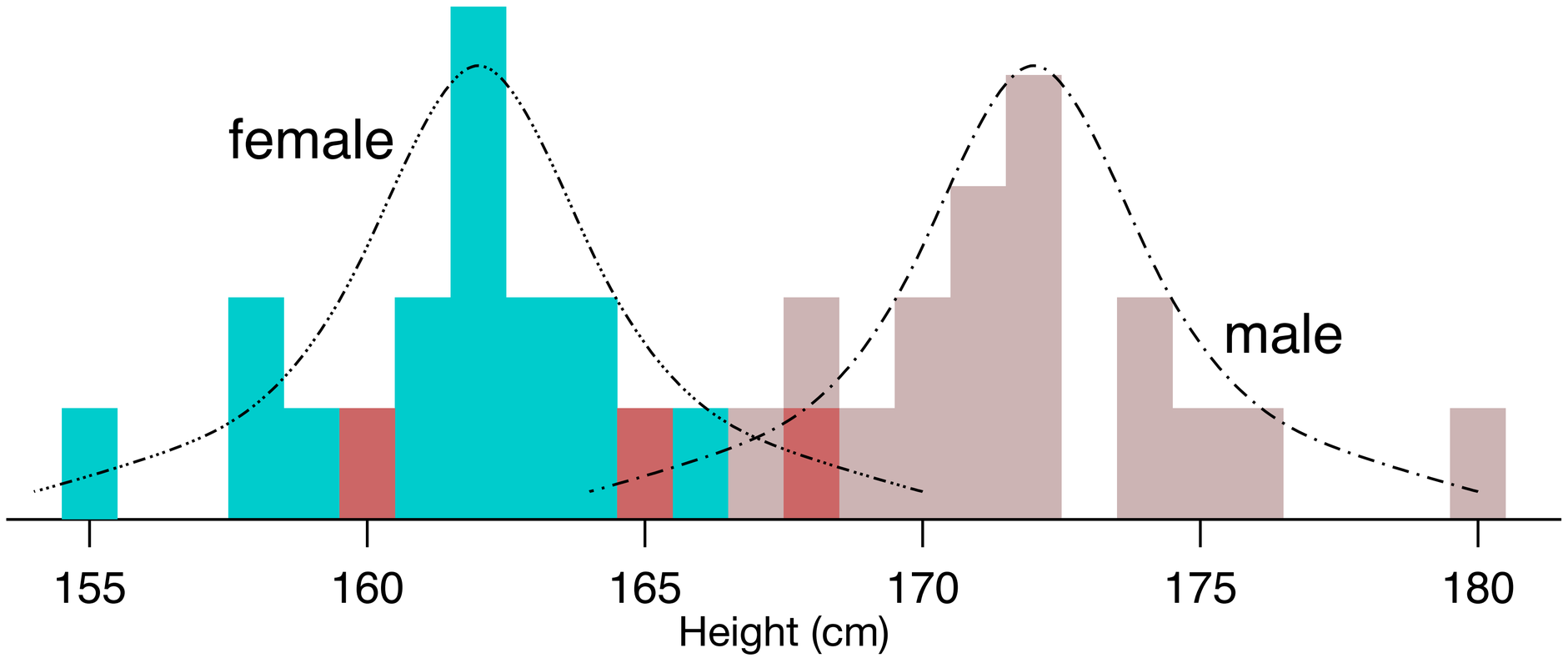
Classification of Male vs Female

Suppose that we have measured the **heights** of a **sample** of male and female people…

Machines can create a **model** of each class from the **features**.



Histograms are one way to ‘model’ the classes



Using multiple features

Better decisions can be made when we have more features.

For example:

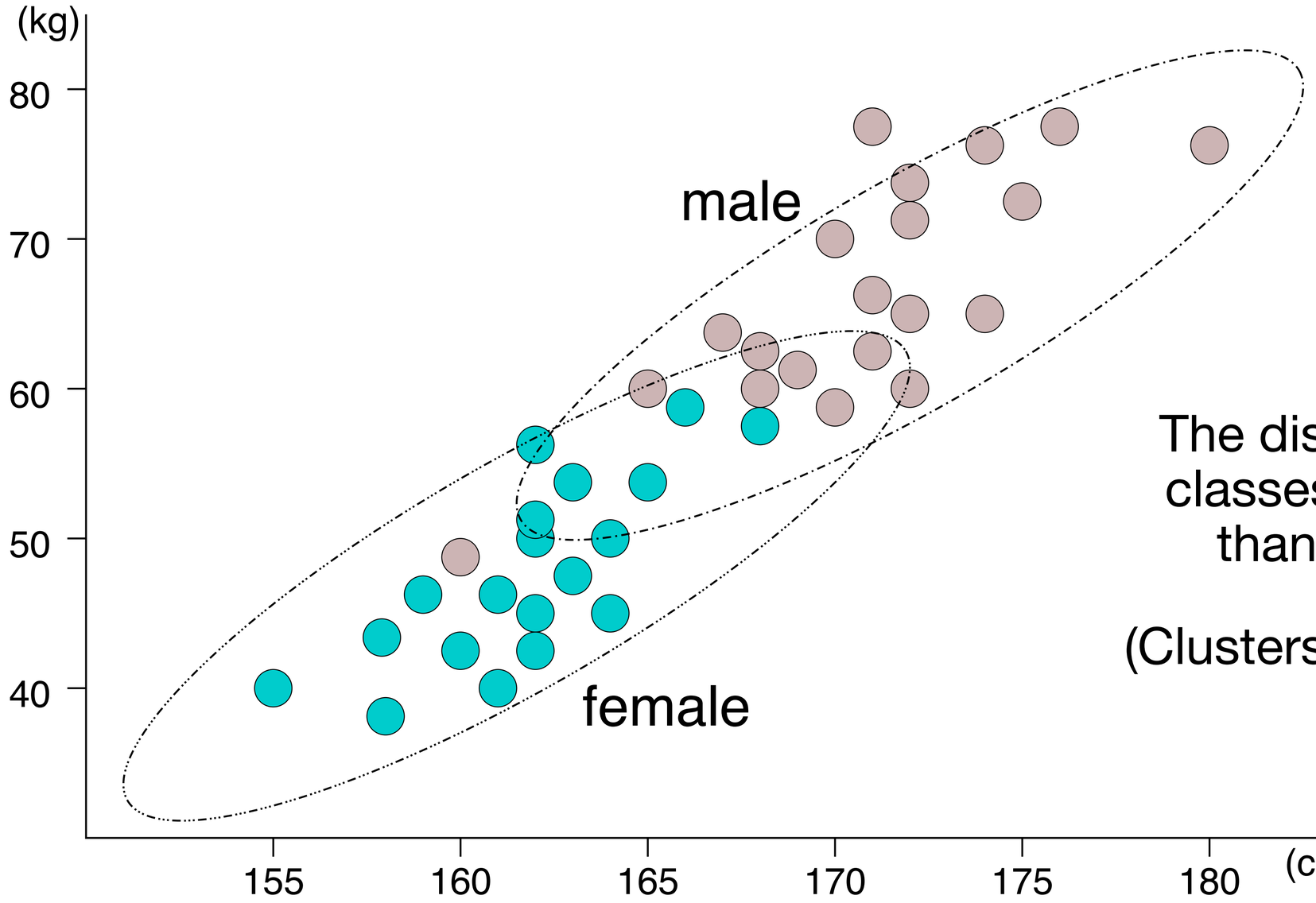
Height and **weight**

Height and **average pitch** of a human voice

(Note) Average pitch

Adult male: 80 to 180 Hz, Adult female: 160 to 260 Hz

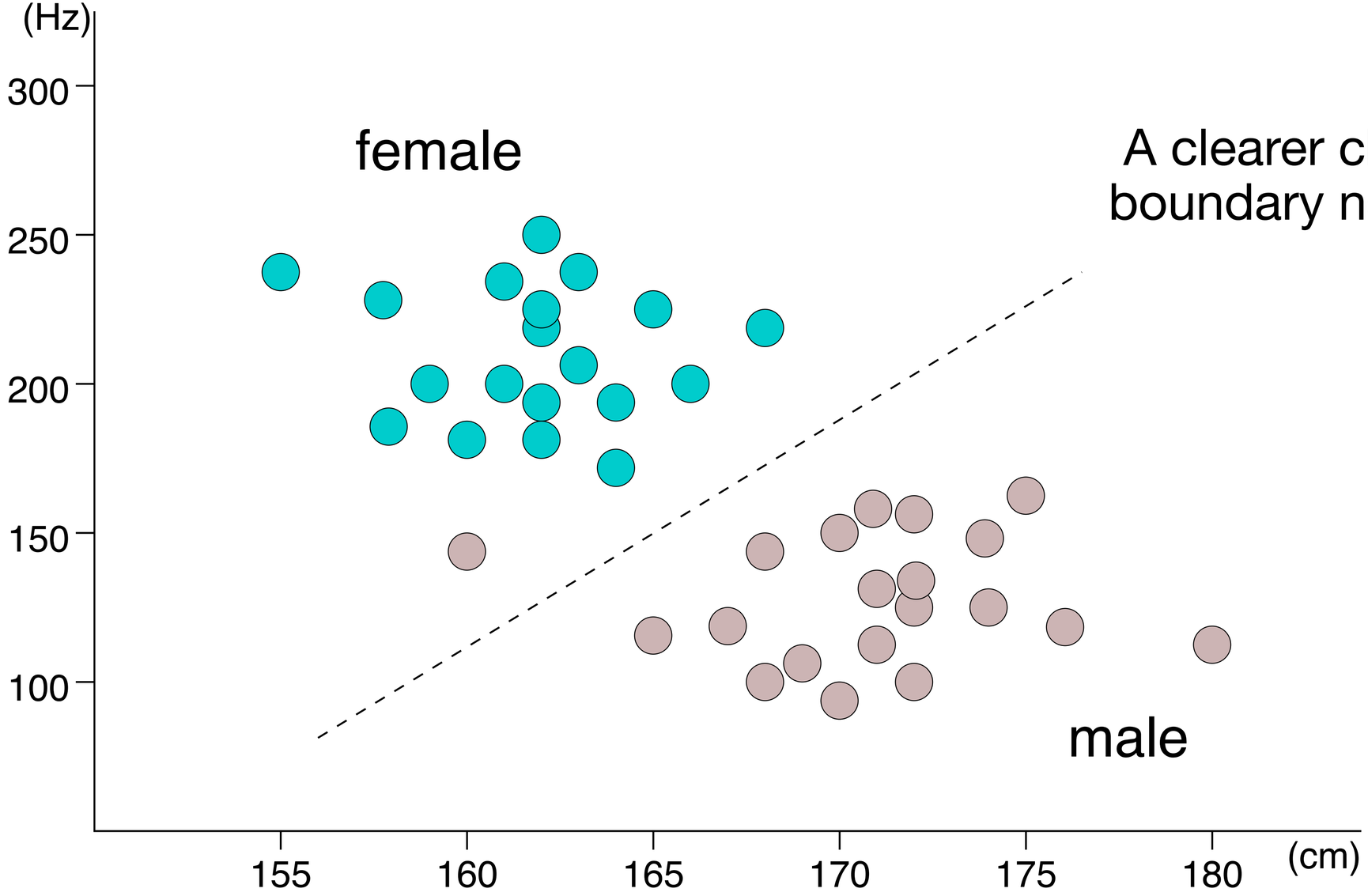
Suppose we have height and weight



The distinction between the classes is not much clearer than with height alone.

(Clusters of points still overlap)

How about height and average pitch?



A clearer classification boundary now appears.

Why is this a better combination?

Summary

A classifier is an algorithm for **attaching a label to an object**.

The classifier is trained from **previously labeled examples** (supervised learning)

The objects are represented using ’**features**’, i.e., one or more measurements made from sensor input.

There are many different ways to **select features** and to **design classifiers**.

To achieve good performance we will need an **iterative design approach** driven by experimentation and robust evaluation.

Classification - formalising the problem

Terminology - class labels

The **class label** will be represented by a **discrete** variable ω.

It can take values from some set of M values, $\Omega = \{\omega_1, \omega_2, \cdots, \omega_M\}$

e.g for classifying people by sex

$$M = 2, \ \Omega = \{\text{Male}, \text{Female}\}$$

e.g for classifying letters of the alphabet

$$M = 26, \ \Omega = \{\text{A}, \text{B}, \cdots, \text{Z}\}$$

Terminology - features

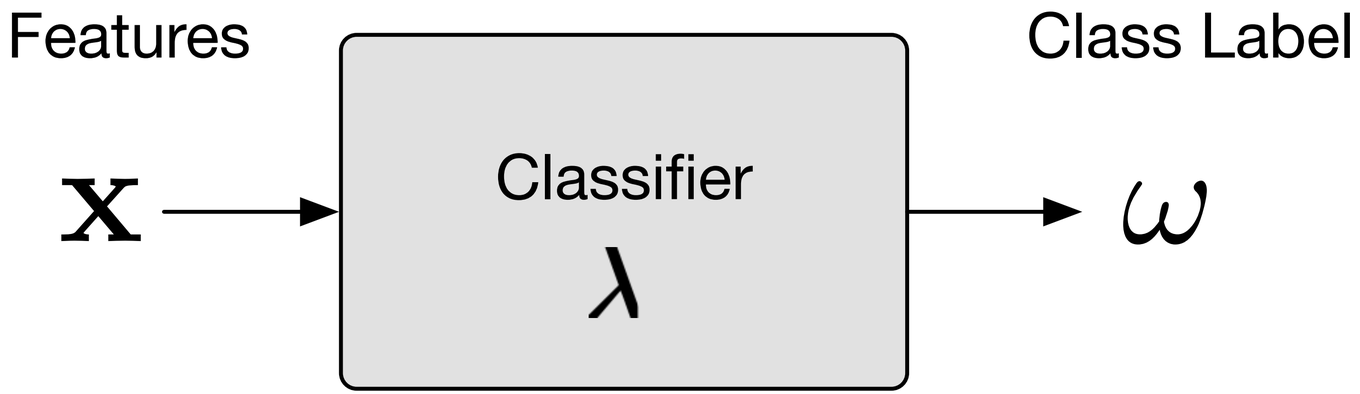
Features are the measurable quantities of the objects to be classified.

Multiple features might be observed e.g height, weight, average pitch.

We store the measurements for a single object in a **feature vector** containing L values:

$$\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_L \end{pmatrix} = \left(x_1, x_2, \cdots, x_L\right)^T$$

The classifier

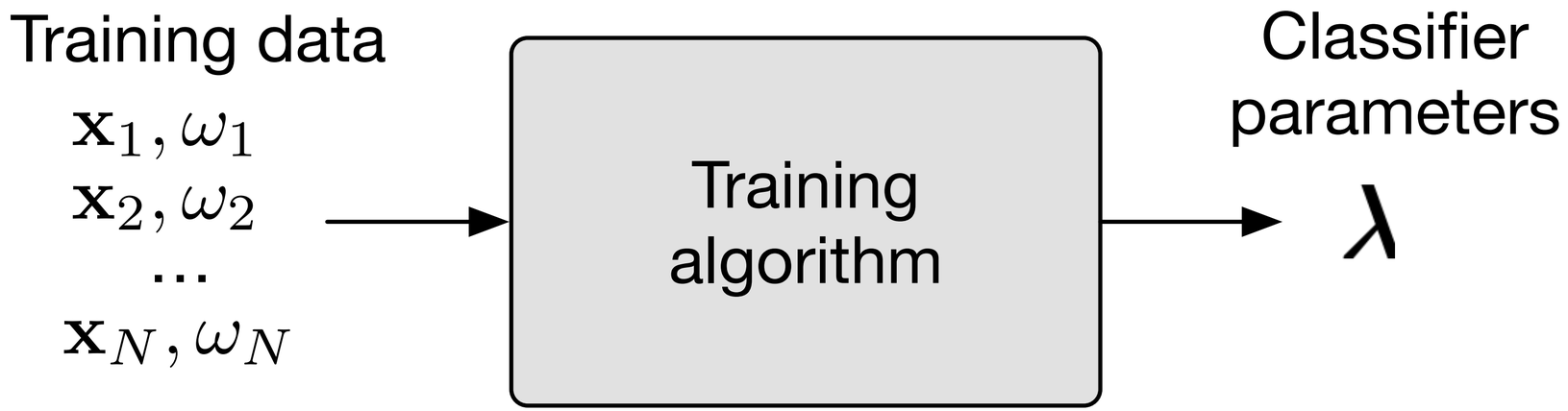


We input the feature vector $\mathbf{x}$ corresponding to some unknown object.

The classifier outputs a class label, ω, determining the class to which the belongs.

The classifier's behaviour is governed by some parameters, λ

Training the classifier



We have labelled training data, i.e. for each object in the training set there is a pair $(\mathbf{x}_i, \omega_i)$.

The training algorithm takes the training data and outputs the 'best' parameters for the classifier, λ

A probabilistic view

When classifying things we can seldom be **100% certain** about the correct class label.

e.g. looking at a photo of a cat I might think,

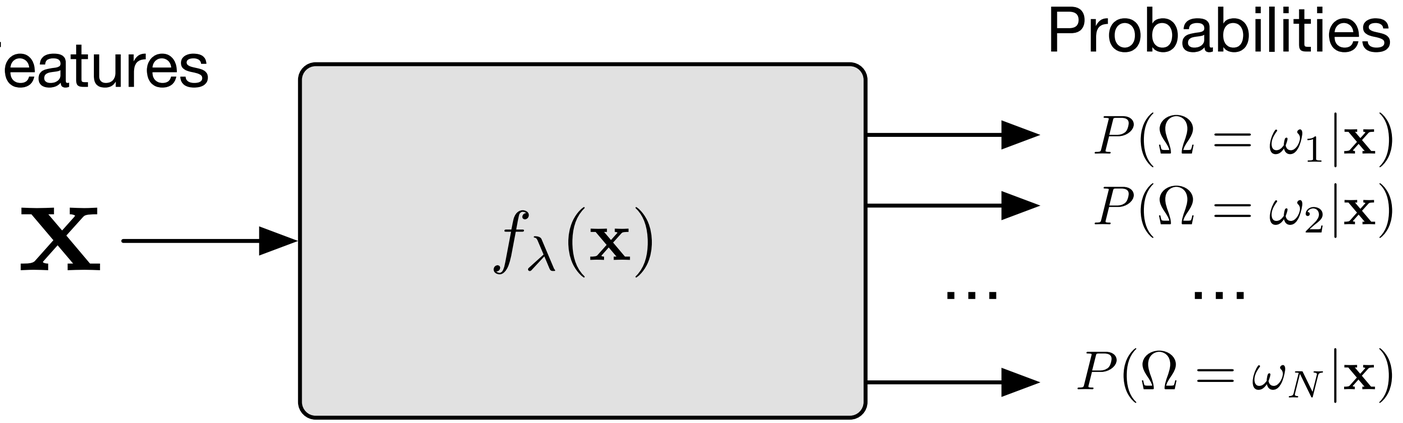
"*I am 90% certain it is a cat but there is a 10% chance it could be a dog*"

To capture this uncertainty we can represent the class label as a **discrete random variable**, i.e., a probability distribution over the possible label values.

e.g, if the label is Ω then its value might be $P(\Omega = \text{cat} \,|\, \mathbf{x}) = 0.9$ and $P(\Omega = \text{dog} \,|\, \mathbf{x}) = 0.1$

A probabilistic view

Our classifier will now be built on a function $f_\lambda(\mathbf{x})$ that estimates the probability of Ω having each label value given the observed features, $\mathbf{x}$.



The function outputs: $P(\Omega = \omega_1 \,|\, \mathbf{x}), P(\Omega = \omega_2 \,|\, \mathbf{x}), \cdots, P(\Omega = \omega_M \,|\, \mathbf{x})$

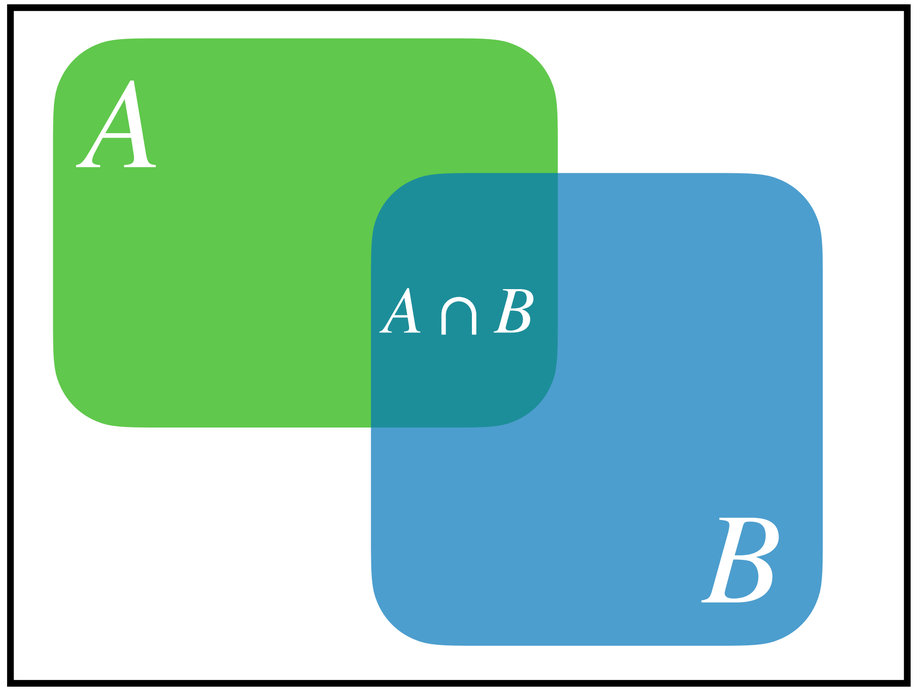
e.g., $P(\Omega = \text{cat} \,|\, \mathbf{x}) = 0.9$, $P(\Omega = \text{dog} \,|\, \mathbf{x}) = 0.1$

These output values are known as **posterior probabilities**.

What are these ‘posterior probabilities’?

The notation here $P(\Omega = \omega_1 \,|\, \mathbf{x})$ means

‘the probability of random variable Ω equalling ω_1 given $\mathbf{x}$ is ….’



$$P(A) = \frac{\blacksquare}{\square} \qquad P(A \cap B) = \frac{\blacksquare}{\square}$$

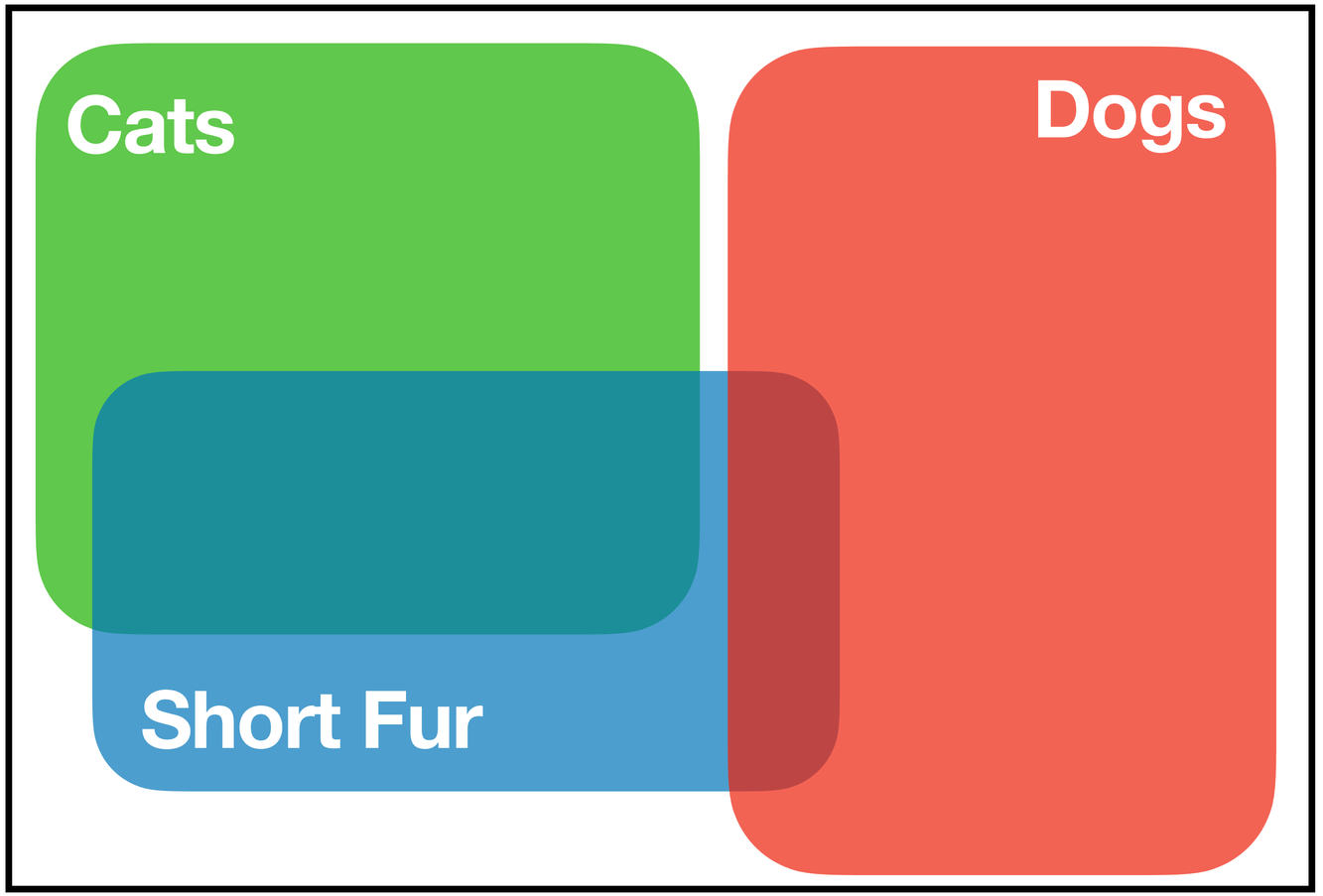
$$P(A \,|\, B) = \frac{\blacksquare}{\blacksquare} = \frac{P(A \cap B)}{P(B)}$$

What are these ‘posterior probabilities’?

For example,

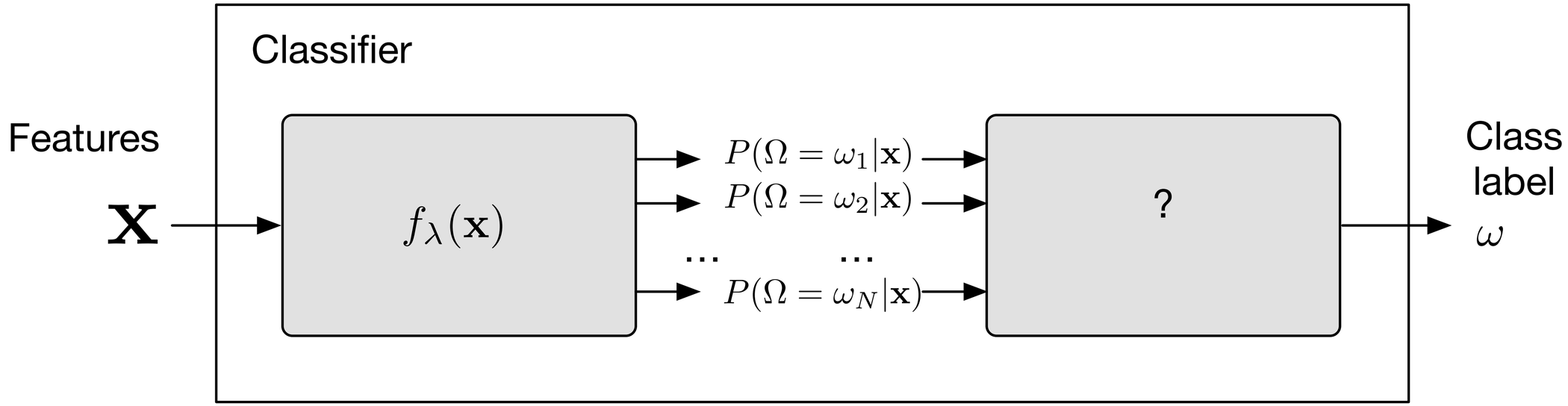
$P(\text{cat} \mid \text{short fur})$ might be quite high given the overlap between the feature and the class.

However, the feature might not be unique to a class.



Bayes' Decision Rule

But for an application to be useful it has to eventually make a decision.



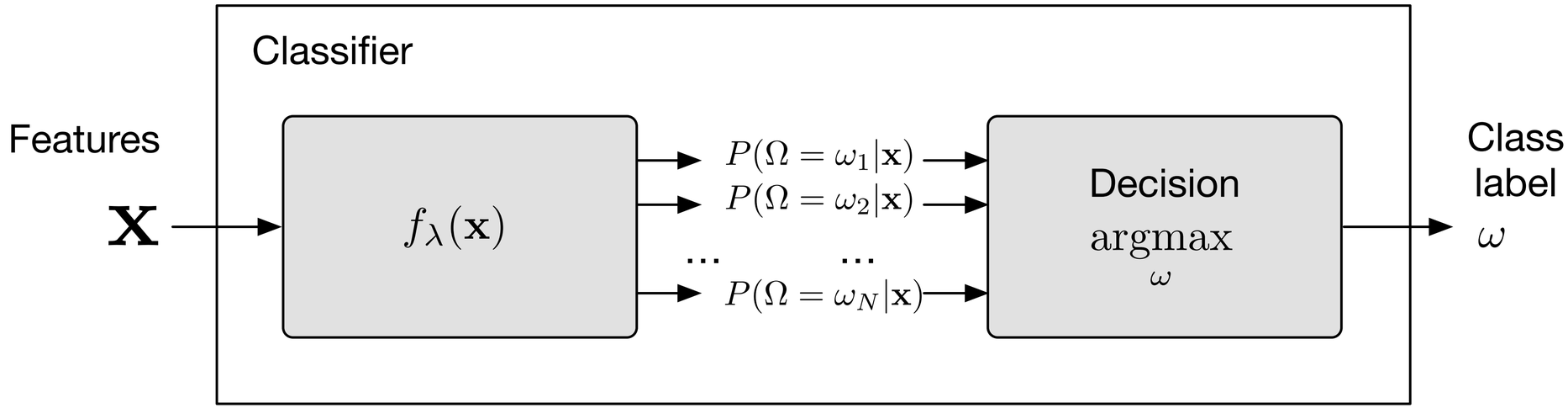
How do we choose the best label?

The 'default' answer will be to use Bayes' Decision Rule…

Bayes' Decision Rule

Bayes decision rule: label $\mathbf{x}$ with the class for which $P(\Omega = \omega \,|\, \mathbf{x})$ is the greatest. i.e., for a given input $\mathbf{x}$, the output label ω is given by

$$\omega^* = \underset{\omega \in \Omega}{\operatorname{argmax}}\, P\left(\Omega = \omega \,|\, \mathbf{x}\right)$$



e.g., if $P(\Omega = \text{cat} \,|\, \mathbf{x}) = 0.9$ and $P(\Omega = \text{dog} \,|\, \mathbf{x}) = 0.1$ then $\omega = \text{cat}$

Summary

A classifier is a function that **maps a feature vector $\mathbf{x}$ onto a class label ω**.

The form of this function is **learnt from training data**.

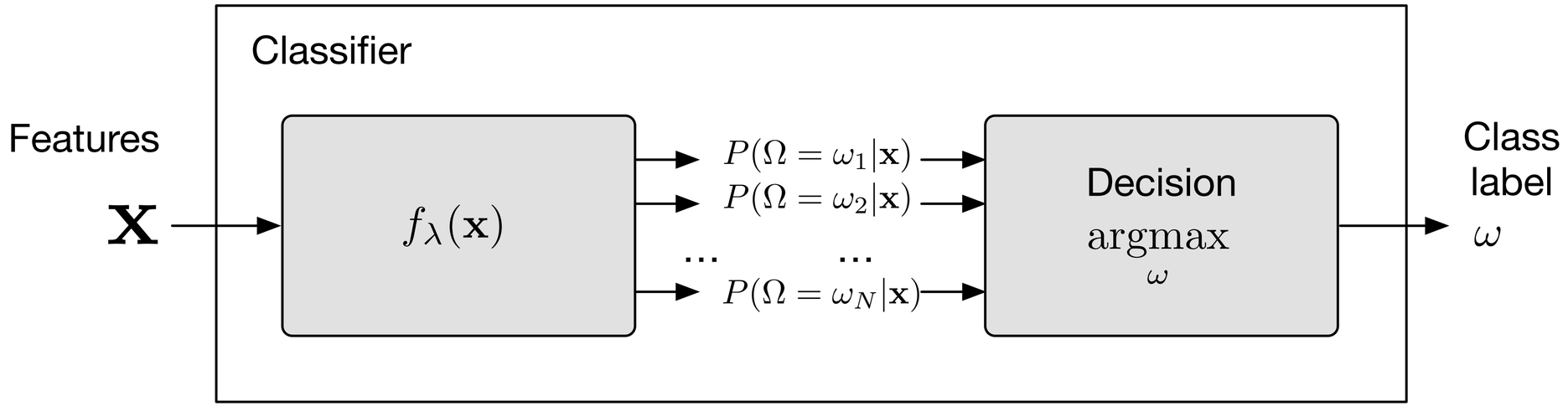
We will be considering classifiers in a **probabilistic** framework.

- A first step computes **posterior probabilities** for all possible classes.
- A decision rule then **chooses which class label** to output.

Bayes' decision rule simply says to output the label of the class with the **highest posterior probability**.

Bayes’ Decision Theory

Posterior probabilities



$P(\omega_1 \,|\, \mathbf{x})$ - this was the posterior probability.

It is the probability of of the class after having made an observation ('posterior' meaning 'after').

How do we calculate $P(\omega_i \,|\, \mathbf{x})$?

Discriminative vs Generative modelling

Discriminative modelling:

- During training we learn a function that estimates $P(\omega \mid \mathbf{x})$ directly
- e.g neural networks, deep learning, etc

Generative modelling:

- During training we learn a function that estimates $P(\mathbf{x} \mid \omega)$
- e.g Gaussian mixture models, hidden Markov models, etc
- Why would $P(\mathbf{x} \mid \omega)$ be helpful? We'll see in a minute.

We will be focusing on generative modelling for the next few weeks.

Bayes' Decision Theory

In the generative modelling approach we will use Bayes' rule. This will allow us to re-express the posterior probability as:

Likelihood **Prior**

Posterior $P\left(\omega_i \mid \mathbf{x}\right) = \dfrac{P\left(\mathbf{x} \mid \omega_i\right) P\left(\omega_i\right)}{P\left(\mathbf{x}\right)}$

Evidence

Note: since we are trying to find the ω_i for which $P\left(\omega_i \mid \mathbf{x}\right)$ is largest, then the evidence is the same for all so we don't need to compute it.

Bayes' Rule

Likelihood Prior

Posterior $P(\omega_i \mid \mathbf{x}) = \dfrac{P(\mathbf{x} \mid \omega_i)\, P(\omega_i)}{P(\mathbf{x})}$

Evidence

Prior: probability of the class **before** observing the data (evidence).

Posterior: probability of the class **after** observing the data (evidence).

Likelihood: probability of **observing** a (random) data point **given** the class label (**compatibility** of the data with the given class).

Evidence or marginal likelihood: probability of **observing** a (random) data point across **all classes.**

Bayes' Decision Theory

Remember, we are using:

$$\omega^* = \underset{\omega \in \Omega}{\operatorname{argmax}} P\left(\Omega = \omega \,|\, \mathbf{x}\right)$$

Using Bayes' rule this becomes:

$$\omega^* = \underset{\omega \in \Omega}{\operatorname{argmax}} P\left(\mathbf{x} \,|\, \omega\right) P\left(\omega\right)$$

We just need to compute $P\left(\mathbf{x} \,|\, \omega\right) P\left(\omega\right)$ (ie., the likelihood times the prior) for all classes ω and find the class which gives the biggest result.

Likelihood

The **likelihood** of feature vector with respect to class can be calculated once x has been observed if the parameters of the distribution p(x|ω) are known.

The parameters of $P\left(\mathbf{x}\,|\,\omega\right)$ will be learnt from the training data.

E.g $P(\text{height}\,|\,\text{male})$ can be the distribution of heights of male people, while $P(\text{height}\,|\,\text{female})$ for females.

These could be modelled as Gaussian distributions. We would use the training data to estimate the parameters (mean, variance) of these distributions.

Probability vs Likelihood

The terms probability and likelihood are used in different contexts,

- Probability is a function of the outcome, given fixed parameter values

  (e.g.) If I flipped an unbiased coin 100 times, we would talk about the **probability** of observing various different sequences.

- Likelihood is a function of the parameter, given a fixed outcome

  (e.g.) If I flipped a coin 100 times and observed a specific sequence, we would talk about the **likelihood** of this sequence occurring for various different degrees of bias.

Bayes' Decision Theory

Consider a simple case with 2 classes, ω_1 and ω_2. Our decision rules implies

$\mathbf{x}$ belongs to ω_1 if $P(\omega_1 \,|\, \mathbf{x}) > P(\omega_2 \,|\, \mathbf{x})$, otherwise it belongs to ω_2

Using Bayes' rule this criterion can be expressed as

$$P(\mathbf{x} \,|\, \omega_1)P(\omega_1) > P(\mathbf{x} \,|\, \omega_2)P(\omega_2)$$

In the case when the priors are equal, $P(\omega_1) = P(\omega_2)$,this reduces to

$$P(\mathbf{x} \,|\, \omega_1) > P(\mathbf{x} \,|\, \omega_2)$$

I.e the maximum of the likelihood

Summary

We want to find a **class label** for a thing after observing its features

We can formalise this as picking the class ω which has the biggest **posterior probability**, $P(\omega \mid \mathbf{x})$

We can't compute posteriors directly so **we use Bayes' rule and compute**, $P(\mathbf{x} \mid \omega)P(\omega)$ instead because

"Posterior is proportional to likelihood times prior"

The class priors and the parameters of the distributions $P(\mathbf{x} \mid \omega)$ can be learnt during a **training stage using labelled data**.

Required reading

Pattern Recognition by Sergios Theodoridis

- Chapter 2, Sections 2.1, 2.2, 2.3

Thanks for listening