**Detection and pattern recognition**

Lecture 1

In this topic, we first try to distinguish between signal processing, their detection and try to identify if there is any pattern in the signal of any kind. This topic would cover very basics of machine learning and not the in-depth mathematical calculation of machine learning.

Signal processing may be defined as the process in which an input signal is fed to a system (or a rule) and the output is obtained. When we talk about signal processing, the task of the computer is to perform the signal processing on an input signal, whereas it is our task to design the (digital) signal processing rule.

There can be various types of signals:

1. 1D signal – which can be represented as vectors – like audio signals, telecommunication signals, etc
2. 2D signals – which can be represented as matrices – like black-white images, geographical maps, etc
3. 3D signals – which can be represented as tensors – like color images which have a 3rd degree of freedom with 3 channels (red, green, blue), CT scans, etc

Depending on the application, the output of the system (quantities of interest) may vary:

1. From signal to **signal** – a vector/matrix/tensor of numbers in - eg time domain to frequency domain
2. From signal to **parameter** – a few numbers in eg 1.23m, 2.44m etc – eg estimate distance from radar signal
3. From signal to **class** – a number in eg. Male/female/others – the class can be further represented as numbers in the natural number line – eg target classification

Here, we would discuss more on the statistical signal processing aspect. This processing technique is a subarea of signal processing which uses a stochastic description of signals and statistical signal processing methods. Thus, a background in basic statistics and statistical formulae and property is very important and essential to understand this topic.

There are two major signal processing approaches that can be used to determine the rule:

1. **Model based approach:**

The input signal’s and the output signal’s properties are known beforehand, i.e. their behaviour with respect to the changes being made to one other. From these properties and the behavioural changes, it is easy to determine the mathematical rule of the model.

A few examples include edge detection, radar distance estimation, channel estimation for digital communication etc.

1. **Learning based or data-driven learning approach:**

The relationship and any knowledge pertaining to it is not known beforehand. It is to be learnt from the examples from the input data. This type of learning approach is often referred to as **machine learning**. Some examples include speech recognition, NLP, etc.

In either of the approaches, there is a need to detect signals and extract information from them. Thus, this brings us to the first part of the topic “DETECTION”. The first focus is mainly on the binary detection which can then easily be extended to multiple classes. The binary detection can be written a bit more mathematically as a **hypothesis test**.

In any hypothesis testing scenario, there are 2 cases:

1. Null hypothesis: H0
2. Alternative hypothesis : H1

Thus there are always 2 unknown true sets H ϵ {H0, H1}.

**Question – What all are we given?**

1. A noisy measurement vector - **x**
2. A probabilistic signal model for **x** : p(**x**|H0)

Here, p(**x**|H0) = probability density function of **x** conditioned on H = H0

p(**x**|H1) = probability density function of **x** conditioned on H = H1

1. Sometimes we are also given the prior information or probability of H0 and H1 : p(H **=** H0). We can easily find out p(H = H1) = 1 - p(H=H0)

**Question : What do we need to design or what is our task?**

We need to design a detector (**x**) ϵ {H0, H1} such that it mimics the true value of H. Thus, we are tasked to estimate the true model or the signal rule or the system as a whole.

There can be cases of binary detection. We assume that H0 is the null hypothesis and H1 is the alternate hypothesis of the model, and we consider H ϵ {H0, H1} is the given, unknown but true state and (**x**)ϵ {H0, H1} is the predicted for some noisy data. Then we can easily come to a conclusion of this table:

|  |  |  |
| --- | --- | --- |
| | H | H1 | H1 |
|  | Correct rejection, **true negative (TN)** | Miss, **false negative (FN)**, type II error |
|  | False alarm, **false positive (FP)**, type I error | Detection, hit, **true positive (TP)** |

Now we need to look at some performance measures for :

1. TNR – True Negative Rate – PTN = P(=H0|H=H0)
2. FPR – False Positive Rate – PFP = P(=H1|H=H0)
3. FNR – False Negative Rate – PFN = P(=H0|H=H1)
4. TPR – True Positive Rate – PTP = P(=H1|H=H1)